File No. S360(Mod 20)-01
Form A26-5847-5

310

Haupt

IBM Systems Reference Library

IBM System/360 Model 20
Functional Characteristics

This manual contains a description of the complete Model 20 System and a detailed description of the processing unit and console. A detailed description of operation codes and operator information is included.

Sixth Edition (September, 1968)

This is a reprint of A26-5847-4 incorporating changes released in the following Technical Newsletters:

Form Number	Pages Affected	Date
N33-1519	165-172	May 28, 1968
N33-1521	iii-v, 5,6, 9,10, 13-16, 21-34, 41,42,45-50, 79-80.1, 103-104.1,121,122 131-136,147,148,163-178	August 29,1968

Changes are periodically made to the specifications herein; before using this publication in connection with the operation of IBM systems, refer to the latest IBM System/360 Model 20 SRL Newsletter, Form N20-0361, for the editions that are applicable and current.

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

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Laboratories, Product Publications, Department 784, Boeblingen/Wuertt, P. O. Box 210, Germany.

CONTENTS

ILLUSTRATIONS

Figures

Tables

ABBREVIATIONS

ACK	Acknowledgement
ACU	Automatic Calling Unit
ADPREP	Address Prepare
BCD	Binary Coded Decimal
bpi	Bytes (or Bits) per Inch
bps	Bits per second
BSCA	Binary Synchronous Communications Adapter
CA	Communications Adapter
CAN	Cancel
CC	1. Condition Code 2. Cyclic Check
CCW	Channel Command Word
CIO	Control I/O
CIOCS	Communications Adapter Input/Output Control System
CM	Channel Mask
CPL	Control Program Load
cpm	Cards per minute
CPU	Central Processing Unit
CRC	Cyclic Redundancy Checking
CR/LF	Carrier Return/Line Feed
CSW	Channel Status Word
DA	Device Address
DLE	Data Link Escape
EBCDIC	Extended Binary-Coded-Decimal Interchange Code
EIA	Electronics Industry of America
ENQ	Enquiry
EOT	End of Transmission
EOTR	End of Transmittal Record
EPO	Emergency Power Off
ETB	End of Transmittal Block
ETX	End of Text
EXT	External
FS	Function Specification
Hex	Hexadecimal
I/O	Input/Output
IOC	Input/Output Channel
ITB	Intermediate Transmittal Block
lpm	Lines per minute
LRC	Longitudinal Redundancy Checking
MFCM	Multi-Function Card Machine
ms	Millisecond
NAK	Negative Acknowledgement
NOP	No Operation
NRZI	Non-Return-to-Zero-IBM
OM	Overlap Mode
PE	Phase Encoding
PSW	Program Status Word
REC	Receive
REQ	Request
RPQ	Request for Price Quotation (from IBM)
RR	Register-to-Register
RVI	Reverse Interrupt
RX	Register-to-Storage
SAK	Stop Acknowledgement
SI	Storage Immediate
SIOC	Serial I/O Channel
SOB	Start of Block
SOH	Start of Heading
SS	Storage-to-Storage
STR	Synchronous Transmitter/Receiver
STX	Start of Text
SYN	Synchronization
TEL	Telephone
TIC	Transfer in Channel
TIOB	Test I/O and Branch
TSM	Transmit
UCS	Universal Character Set
USASCII	United States of America Standard Code for Information Interchange
usec	Microsecond
VRC	Vertical Redundancy Checking
WABT	Wait Before Transmit
XIO	Transfer I/O

The IBM System/360 Model 20 represents a major advance in low cost, high reliability data processing systems. New dimensions of performance, flexibility, and reliability are offered to the punched card, magnetic tape, or direct access storage user.

The System/360 Model 20 is a powerful, productive, and practical extension of the IBM System/360. The System/360 Model 20 offers users the functional capabilities of large data processing systems at performance and cost levels tailored to the individual user's needs.

The IBM System/360 Model 20 consists of a group of interconnected functional units operating under the control of a series of instructions, called a program, stored in the Central Processing Unit (CPU). Figure 1 shows a typical configuration of units. The IBM 2020 Processing Unit is the central figure in the group, and attached to it are various card reading, card punching, magnetic tape, disk storage drive, and printing units to provide input to the system and output from the system.

The purpose of this reference manual is to provide detailed information on the operation and characteristics of the central processing unit. The Systems Reference Library IBM System/360 Model 20 Bibliography, Form A26-3565, lists all related publications.

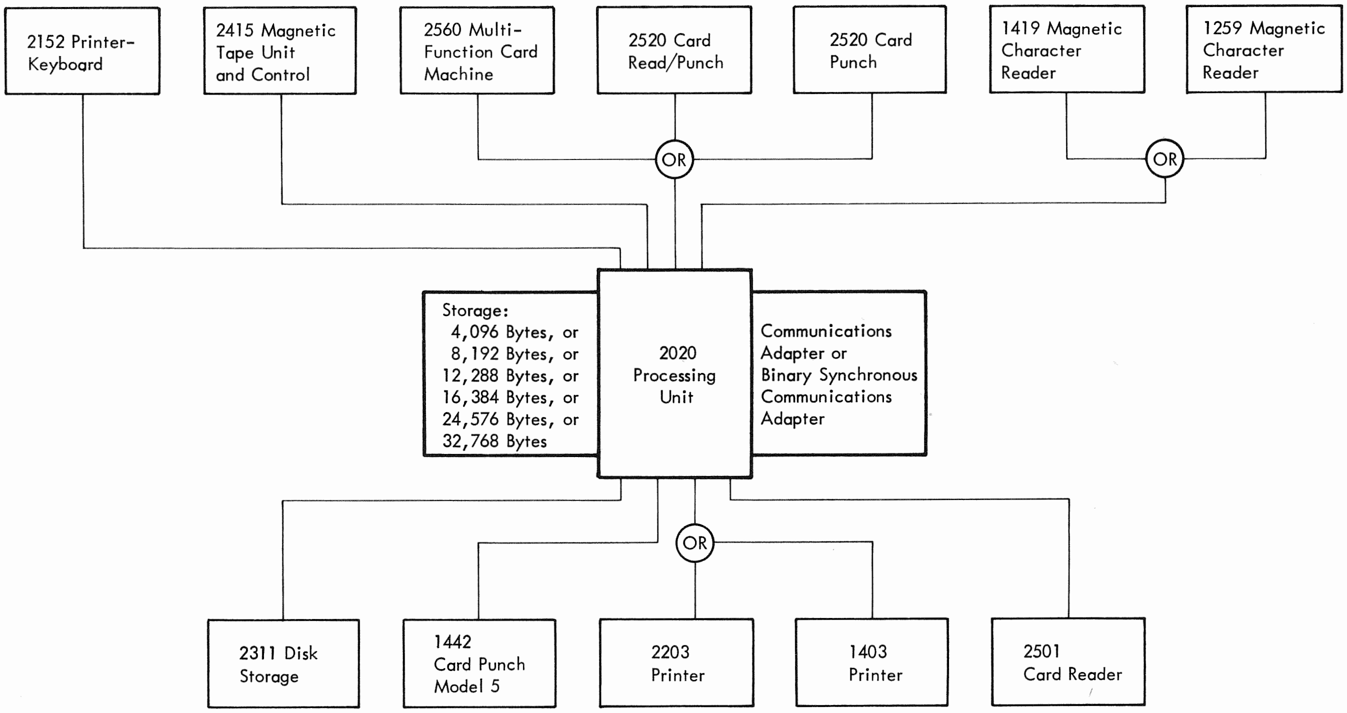
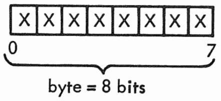


Figure 1. IBM System/360 Model 20 01885

CENTRAL PROCESSING UNIT

Main Storage

Main storage consists of 4,096; 8,192; 12,288; 16,384; 24,576; or 32,768 position of magnetic core storage. Each position has an address and contains an eight-bit unit of information referred to as a byte. Coded combinations of bits within a byte can represent alphabetic, numeric, binary, or logical data.



01886

Main storage is used to hold all the data that is to be operated upon, or processed, at a given time. It also holds the instructions, or program, which control the operation of the system. For each individual job performed by the system, certain portions of main storage are assigned to store instructions, and certain portions to store data to be processed.

General Registers

Eight general purpose registers, small auxiliary storage units, are provided for temporary storage of small amounts of data. Each register is the equivalent of two bytes (one halfword) and is loaded or unloaded under the control of the stored program. Information may flow from register to register, main storage to register, or from register to main storage.

The registers are numbered 8-15 and are selected by the four-bit R or B field of an instruction.

Parity Checking

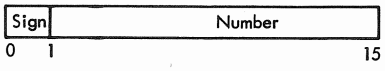
To ensure the accurate transfer of data, an extra (parity) bit is generated for each four bits transferred to or from main storage or register storage. The parity bit is added during transfer, if it is needed, to maintain an odd number of bits. The bit count is monitored continuously, and any missing or extra bits which result in an even number of bits cause a CPU parity error.

Data Formats

The basic unit of addressable data is an eight-bit byte. Each address contains eight bits of data and can be considered a byte boundary.

The byte is divided into two sections of four bits each. A check (parity) bit is provided for each four bits.

Halfword Binary Number



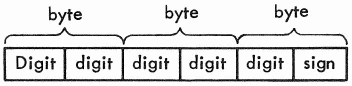
01887

A halfword binary number has a fixed length of two bytes (16 bits). The leftmost bit is reserved for sign (+ or -) control.

Decimal Number

A decimal number may be in either of two forms: packed decimal or zoned decimal.

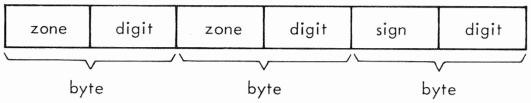
Packed Decimal: This format allows two numeric digits to be stored within one eight-bit byte.



01888

An eight-bit byte may contain two numeric digits, except in the case of the rightmost byte, which has a sign to the right. Variable field length in this format allows for fields up to 16 bytes in length.

Zoned Decimal: This format contains one digit in the rightmost four positions of each byte.



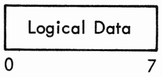
01889

The left four bits of each byte in this format are called zone bits, and are not used except for the low-order (rightmost) byte which uses them for the sign. Zone bits do not affect the value of the numeric digit contained in the right four bits of the byte.

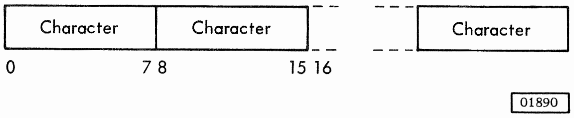
Logical Data

Logical data may be contained in the instruction or it may reside in core storage as an operand (an operand is data contained in or addressed by the instruction and used to execute the instruction). Logical data may have a fixed length of one byte or may be variable in length up to a maximum of 256 bytes.

Fixed-Length Logical Data



Variable-Length Logical Data



Instruction Format

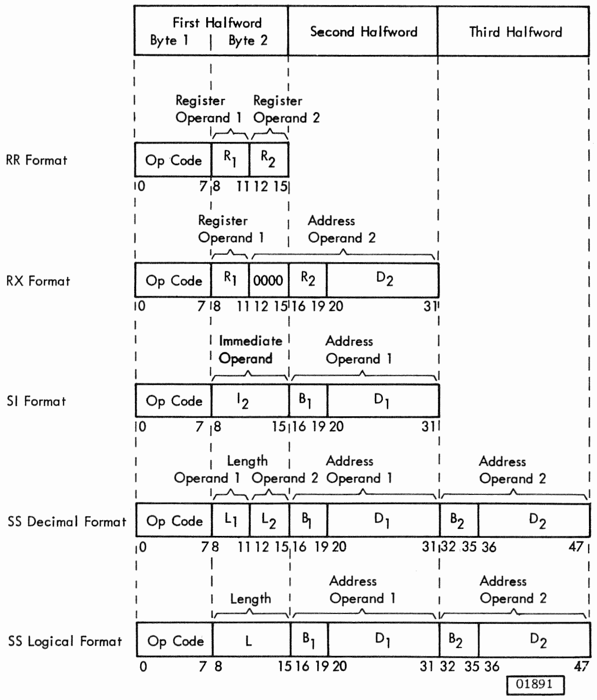
The instruction format specifies the length of the instruction and the type of operation to be performed. The length of the instruction can be one, two, or three halfwords. The types of instruction formats are RR, RX, SI, and SS.

RR Format: Denotes a register-to-register operation.

RX Format: Denotes a register-to-storage or a storage-to-register operation. In this format, bits 12 through 15 must be zero.

SI Format: Denotes a storage-immediate operation. In this format the I2 field of the instruction is the second operand.

SS Format: Denotes a storage-to-storage operation.



In each format, the first instruction halfword consists of two parts. The first byte contains the operation code (op code). The length and format of an instruction are specified by the first two bits of the operation code. The second byte may be used to contain data, specify operand lengths, or specify registers to be used by the operation.

Instruction Execution

For purposes of describing the execution of instructions, operands are designated as first and second operands.

These names refer to the manner in which the operands participate. The operand to which a field in an instruction format applies is generally denoted by the number following the code name of the field, for example, R1, B1, L2, D2. Normally, the operation of the CPU is controlled by instructions taken in sequence. An instruction is fetched from a location specified by the current instruction address. The current instruction address itself is located in the Program Status Word (PSW). After the fetch operation, the current instruction address is increased by the number of bytes in the fetched instruction to enable addressing of the next instruction in sequence.

The instruction is then executed by adding, subtracting, multiplying etc., both operands with each other. The result that is thus obtained usually replaces operand 1. Upon the execution of certain instructions, a condition code,which reflects the nature of the result, is set into the PSW.

Subsequently, the updated address in the PSW is used to read out the next instruction from the main storage and the processing continues.

Information Positioning

Byte locations in storage are consecutively numbered from 0; each number being considered the address of the corresponding byte. Bytes may be handled singly or strung together in fields. A group of two consecutive bytes is called a "halfword." The location of any field or group of bytes is specified by the address of the leftmost byte.

Information positioned in storage may be in fixed-length format or variable-length format. The length of fields is either implied by the operation to be performed or stated explicitly as part of the instruction. When the length is implied, the information is said to have a fixed length, which can be one, two, or four bytes.

Fixed-length fields must be located in main storage on an integral (halfword) boundary (Figure 2) for that unit of information. A boundary is called integral for a unit of information when its storage

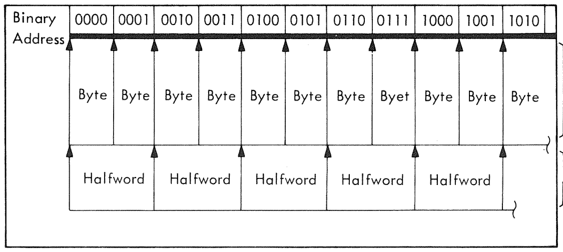


Figure 2. Integral Boundaries 01892

address is a multiple of the length of the unit in bytes. For example, a halfword (two bytes) must have an address that is a multiple of the number 2.

In the 2020, all instructions and all data with fixed word length (operands addressed by RX-format instructions) must begin on a halfword boundary. An instruction (or data with fixed word length) is properly located at a halfword boundary when its address is even or, in other words, when the low order bit of the address is zero. An improperly placed instruction causes an error stop.

When the length of a field is not implied by the operation code, but is stated explicitly, the information is said to have variable-length field.

Within any instruction format or any fixed-length operand format, the bits are consecutively numbered from left to right starting with bit number 0.

Variable-length fields are not restricted to halfword boundaries, and may contain up to 256 bytes. Length is variable in increments of one byte.

ADDRESSING

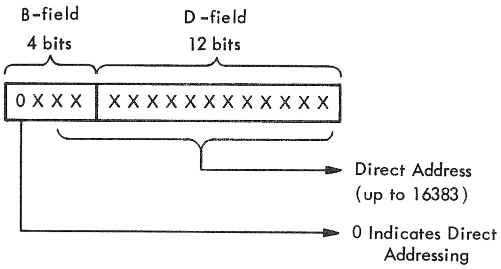
Byte locations in storage are expressed in binary form and consecutively numbered from 0000 to the upper limit of available storage. Appendix D explains the binary number system, and Appendix E contains the hexadecimal representation for addresses 0000 to 4095. The first 144 bytes (Bytes 0000-0143) are reserved for internal CPU control and are not available to the program. The location of any field or group of bytes is specified by the address of the leftmost byte.

An address used to refer to main storage may be specified by either of two methods; direct addressing or effective address generation.

NOTE: Neither a direct address nor an effective address must be zero, because zero is regarded as a specification error. The CPU checks all direct or effective addresses for validity and stops on a specification error if these addresses are zero, or pertain to the protected area or exceed the storage capacity.

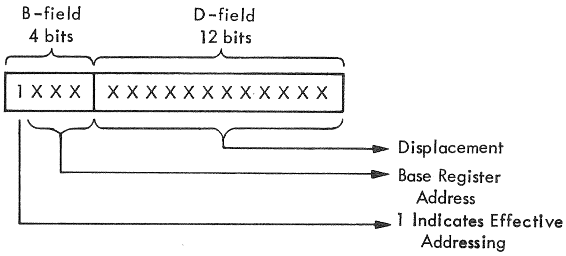
Direct Addressing: Direct addressing is used when the high-order bit in the B-field of an instruction is zero.

When the direct addressing method is employed, the low-order 14 bits of the combined B- and D-fields are used to refer directly to byte locations in main storage. The 12 binary bits in the D-field allow an address specification of up to 4095. To address additional (optional) storage, the adjacent two bits in the low-order position of the B-field are used, allowing address specification of up to 16383.



01893

Effective Addressing: Effective addressing is used when the high-order bit in the B-field of an instruction is one.



In the effective-address generation method, the contents of the general register specified by the B-field of an instruction, are added to the contents of the D-field of the instruction to form the effective address. The content of the general register specified by the B-field is referred to as the "base address." The content of the D-field is referred to as the "displacement." (This type of address modification is commonly referred to as indexing.) Effective addressing may be either in a positive or a negative direction, and is determined by the sign of the base address. Caution should be exercised because a resultant effective address that is negative or that refers to the first 144 bytes of main storage results in an error condition.

Any of the eight general registers, 8 through 15, may be specified in the B-field as the location of the base address for effective-address generation.

If there are zeros in either the general register specified, or in the displacement field of the instruction when effective-address generation is specified, the effective address generated is the same as direct addressing with the nonzero component.

Effective addressing is useful for program routines which require address modification.

Upper Limit of Main Storage: Data can be stored up to (and including) the last available byte of main storage. However, the user must exercise caution when loading a program into the upper available space of main storage. In 2020 Submodels 1 through 4, the last instruction must be located so that at least one byte of main storage remains. (Actually, two bytes of main storage have to remain unused, otherwise the last instruction would not be located on a halfword boundary.) The remaining byte is required to avoid a storage overflow which can occur in 2020 Submodels 1 through 4 because, when the instruction is read out, the internal CPU controls automatically update the current instruction address by an increment equal to the length of this instruction. In 2020 Submodel 5, these restrictions do not apply, and instructions can be loaded to the top limit of main storage without addressing errors occurring.

OPERANDS

For addressing purposes, operands can be grouped in three classes: explicitly addressed operands in main storage, immediate operands placed in main storage as part of the instruction stream, and operands located in the general registers.

Explicitly Addressed Operands

An explicitly addressed operand is selected from a main-storage location not related to the location of the instruction referring to it. It is always specified by means of a storage address. When the operand contains more than one byte, the address gives the location of the first byte of the field, subsequent bytes being located in higher addresses. Both the first and second operands of an instruction can be explicitly addressed.

Explicitly addressed operands can be of fixed length or variable length. The length of variable-field-length operands is specified in the L-field of the instruction. The L-field, either four or eight bits long, specifies the length in terms of the number by bytes to the right of the addressed byte, and thus can specify a maximum field length of 256 bytes.

Immediate Operands

An immediate operand consists of one eight-bit byte of data which is located in the instruction itself. Only the instructions in the SI format contain immediate data. The immediate data is always the second operand; the first operand is located in the main-storage location specified by the B1-D1 field.

Operands in Registers

Information referred to by an instruction may be located in one of eight general registers. The registers are identified by numbers 8-15 and are selected by the four-bit R or B field of an instruction. The registers are not designated by main-storage addresses.

An operand located in a register has a fixed length of one halfword, or 16 bits.

TIME SHARING

The 2020 has the ability to operate in a mode referred to as "time sharing." Time sharing is a means of overlapping input/output operations with each other and with processing. Time sharing is based on a system of monitoring the operation of input/output devices and sequencing the transfer of data to or from the I/O devices so as to make the most efficient use of processing time.

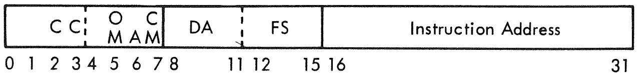
Processing operations in the CPU are time shared with the transfer of data between main storage and the I/O devices. When an I/O device requests service, processing is suspended only for the time required to send or to accept the input/output data.

Time sharing allows the CPU to perform useful processing functions while card or forms movement is taking place. A system of signaling that the I/O device is finished with the data transfer to or from the CPU is referred to as an "interruption" system.

Program Status Word (PSW)

The PSW contains the information necessary for proper program execution. It is located in an internal register in the CPU and is not directly addressable. The programmer can change the PSW by means of a Set PSW instruction. The PSW has a fixed-length format of two halfwords.

PSW Format



0 - 1	Not Used
2 - 3	Condition Code
4	Not Used
5	Overlap Mode (Submodel 5 only)
6	* USASCII Mode Bit
7	Channel Mask
8 - 11	Device Address
12 - 15	Function Specification
16 - 31	Instruction Address

* United States of America Standard Code for Information Interchange

01895

The PSW is used to control instruction sequencing and to hold and indicate the status of the system in relation to the program being executed. The active or controlling PSW is called the "current PSW."

By storing the current PSW during an interrupt, the status of the CPU can be preserved for subsequent inspection. By loading a new PSW or part of a PSW, the state of the CPU can be initialized or changed. The functions of the PSW are described in detail later in this publication.

When the current PSW is stored during an input/output interruption, the status of the CPU and the next sequential address are preserved for use after the interruption is serviced. An instruction to load a new PSW is equivalent to an unconditional branch to the instruction address contained in the new PSW.

Operation of the Program Load key or System Reset key causes the condition code (bits 2-3), the overlap mode bit (bit 5), the USASCII mode bit (bit 6), and the channel mask bit (bit 7) of the current PSW to be reset to zero.

Interruption

In the 2020, an automatic interrupt system is provided to make optimum use of the I/O devices and available processing time. The I/O devices signal the CPU to store an end condition when a data transfer has been terminated. Upon completion of the execution of each instruction, during the time that the CPU is in an interruptible mode, the CPU checks the various end conditions according to a built-in priority sequence. The first end condition thus found causes the actual interrupt. The interrupt is an automatic branch from the main program to a subroutine. This subroutine may be designed to test the received data for validity or to perform some other action. Since the interrupt occurs at the earliest moment possible after a data transfer has been completed, it is the primary means of controlling I/O operations.

The branch to the subroutine is accomplished by replacing the current PSW with a new PSW which contains the start address of the subroutine in its instruction address portion. The former current PSW is stored into a particular core-storage location and thus becomes the old PSW. During the transfer, the old PSW is furnished with the device address and the function specification of the I/O unit that caused the interrupt.

To initiate a branch back to the main program, the old PSW must be fetched from storage to function as the current PSW again. This is done by addressing the old PSW in the B1-D1 fields of a Set PSW instruction placed at the end of the subroutine. The entire exchange of the program status words is fully automatic; however, the channel mask bit in the PSW, which determines whether the CPU is interruptible or not,may be set or reset under program control. The CPU is interruptible when the channel mask bit is set to 1; it is not interruptible if the channel mask bit is 0. The channel mask bit is zero when the CPU is in a reset state.

ARITHMETIC AND LOGICAL OPERATIONS

The arithmetic and logical operations are employed to process binary numbers of fixed length, decimal numbers of variable length, and logical information of either fixed or variable length.

Arithmetic and logical operations performed by the CPU fall into three classes: binary arithmetic, decimal arithmetic, and logical operations. These three classes differ in the data formats used, the registers involved, the operations provided, and in the way the field length is stated.

BINARY ARITHMETIC

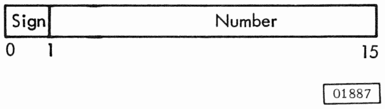
The binary arithmetic operand is the 16-bit binary halfword. The binary instructions perform binary arithmetic on operands serving as addresses, index quantities, and counts, as well as binary data. One operand is always in a general register. The second operand may be either in main storage or in a general register.

Data Format

Binary numbers occupy a fixed length format consisting of a one-bit sign followed by the 15-bit binary field. When held in one of the general registers, a binary quantity has a 15-bit binary field and occupies all 16 bits of the register.

Binary data in main storage also occupies a 16-bit halfword, with a binary field of 15 bits. This data must be located on integral storage boundaries for this unit of information, i.e., the low-order binary address bit is zero.

Halfword Binary Number



Number Representation

All binary operands are treated as signed numbers. Positive numbers are represented in true binary notation with the sign bit set to 0. Negative numbers are represented in two's-complement notation with a 1 in the sign bit. The two's complement of a number is obtained by inverting each bit of the number and adding a one in the low-order bit position.

The number obtained by inverting each binary bit represents the one's complement of the original number. To represent the two's complement, a one must be added to the low-order position of the one's complement.

```
0 1 1 1 0 1 0 1 0 0 1 0 1 1 1 0  =  original number
1 0 0 0 1 0 1 0 1 1 0 1 0 0 0 1  =  one's complement
1 0 0 0 1 0 1 0 1 1 0 1 0 0 1 0  =  two's complement
↑
└── sign position
```

Two's-complement notation does not include a negative zero. It has a number range in which the set of negative numbers is one larger than the set of positive numbers. The maximum positive number consists of an all-one integer field with a sign bit of zero, whereas the maximum negative number consists of an all-zero integer field with a 1 bit for sign.

The CPU cannot represent the complement of the maximum negative number. When an operation, such as a subtraction from zero, produces the complement of the maximum negative number, the number remains unchanged, and a binary overflow condition is recognized. An overflow does not result, however, when the number is complemented and the final result is within the representable range. An example of this case is a subtraction from minus one.

The sign bit is leftmost in a number. An overflow carries into the sign-bit position and changes the sign.

Figure 3 illustrates some sample values of 16-bit binary integers and their equivalents in decimal form.

Number		Signed Decimal		S - - - - - - - - - INTEGER - - - - - - - - -
$2^{15}-1$	=	32767	=	0111 1111 1111 1111
2^{8}	=	256	=	0000 0001 0000 0000
2^{0}	=	1	=	0000 0000 0000 0001
0	=	0	=	0000 0000 0000 0000
-2^{0}	=	-1	=	1111 1111 1111 1111
-2^{1}	=	-2	=	1111 1111 1111 1110
-2^{8}	=	-256	=	1111 1111 0000 0000
$-2^{15}+1$	=	-32767	=	1000 0000 0000 0001
-2^{15}	=	-32768	=	1000 0000 0000 0000

Figure 3. Binary Representation 01896

Condition Code

The results of binary add, subtract, and compare operations are used to set the condition code in the Program Status Word (PSW). All other binary operations leave this code undisturbed. The condition code can be used for decision-making by subsequent branch-on-condition instructions.

The condition code is set to reflect three types of results for binary arithmetic. For most operations, the states 0(00), 1(01), or 2(10) indicate a zero, less than zero, or greater than zero content of the result register. An overflow results in a Binary Overflow Check condition.

For a compare operation, the states 0, 1, or 2 indicate that the first operand is equal, low, or high compared to the second operand. Condition code settings are shown in Figure 4.

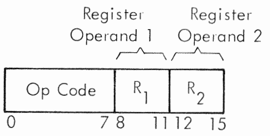
	0	1	2
Compare Halfword	equal	low	high
Add Halfword	zero	< zero	> zero
Subtract Halfword	zero	< zero	> zero
Add	zero	< zero	> zero
Subtract	zero	< zero	> zero

Figure 4. Condition Code Settings for Binary Arithmetic 01897

Instruction Format

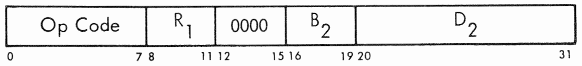
Binary instructions use the following two formats:

RR Format



01898

RX Format



01899

In these formats, R_1 specifies the address of the general register containing the first operand. The second operand location, if any, is defined differently for each format.

In the RR format, the R2 field specifies the address of the general register containing the second operand.

In the RX format, the address which designates the storage location of the second operand is derived from the contents of the B2 and D2 fields of the instruction. As described in the section on Addressing, this address may be taken directly from the B2 and D2 fields, or an effective address may be formed by adding the contents of the general register specified by the B2 field to the contents of the D2 field.

Bits 12 through 15 of the RX format must be zero. The second operand, addressed by B2-D2, must begin on a halfword boundary, i.e., its address must be even, otherwise a program error stop occurs.

Results of operations replace the first operand except for a store operation. The result replaces the second operand for the store operation.

An instruction can specify the same general register both for address modification and for operand location. Address modification is always completed before execution of the operation.

The contents of all general registers and storage locations participating in the addressing or execution part of an operation remain unchanged, except for the storing of the final result.

Instructions

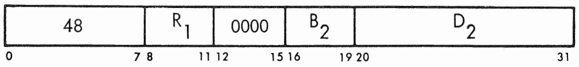
The binary arithmetic instructions and their mnemonics, formats, and operation codes are listed in Table 1.

Table 1. Binary Instructions 01900

Name	Mnemonic	Format	Operation Code
Load Halfword	LH	RX	48
Add Halfword	AH	RX	4A
Subtract Halfword	SH	RX	4B
Compare Halfword	CH	RX	49
Store Halfword	STH	RX	40
Add	AR	RR	1A
Subtract	SR	RR	1B

Load Halfword (RX)

LH R1, D2 (0, B2)



01901

The second operand is placed in the first operand location. The instruction uses RX format and provides for loading a halfword from the main-storage location specified by the B2-D2 fields into the register specified by the R1 field. The condition code is not changed.

Add Halfword (RX)

AH R1, D2 (0,B2)

4A	R_1	0000	B_2	D_2
0 7	8 11	12 15	16 19	20 31

01902

The second operand is added to the first operand and the sum is placed in the first operand location. Both operands are halfword length. The first operand is contained in the register specified by the R1 field. The second operand is located at a main-storage location specified by the B2-D2 fields of the instruction.

Addition is performed by adding all 16 bits of both operands. If the "carries" out of the sign-bit position and the high-order numeric bit position agree, the sum is satisfactory; if they disagree, an overflow occurs. The sign bit is not changed after the overflow. A positive overflow yields a negative final sum, and a negative overflow results in a positive sum. The overflow causes a binary-overflow-check condition. The condition code is set to 00, 01, or to 10 to indicate that the result is zero, less than zero, or greater than zero.

Subtract Halfword (RX)

SH R1, D2 (0,B2)

4B	R_1	0000	B_2	D_2
0 7	8 11	12 15	16 19	20 31

01903

The second operand is subtracted from the first operand and the difference is placed in the first operand location. Both operands are halfword length. The first operand is contained in the register specified by the R1 field. The second operand is located at a main-storage location specified by the B2-D2 fields of the instruction.

Subtraction is performed by adding the two's complement of the second operand to the first operand. All 16 bits of both operands participate, as in ADD. If the "carries" out of the sign-bit position and the high-order numeric bit position agree, the difference is satisfactory; if they disagree, an overflow occurs, resulting in a binary-overflow-check condition. The condition code is set to 00, 01, or 10 to indicate that the result is zero, less than zero, or greater than zero.

Compare Halfword (RX)

CH R1, D2 (0,B2)

49	R_1	0000	B_2	D_2
0 7	8 11	12 15	16 19	20 31

01904

The first operand is compared with the second operand, and the result of the compare is indicated by the setting of the condition code in the PSW. The first operand is contained in the register specified by the R1 field. The second operand is located at a main-storage location specified by the B2-D2 fields of the instruction.

Comparison is algebraic, treating both comparands as 16-bit signed integers. Operands in registers or storage are not changed as a result of the operation.

The condition code is 00 if the operands are equal, 01 if the first operand is lower than the second operand, and 10 if the first operand is higher than the second.

Store Halfword (RX)

STH R1, D2 (0,B2)

40	R_1	0000	B_2	D_2
0 7	8 11	12 15	16 19	20 31

01905

The first operand is stored at the second operand location. The instruction uses RX format and provides for storing a halfword from the register specified by the R1 field into the main-storage location specified by the B2-D2 fields of the instruction. The condition code remains unchanged.

Add (RR)

AR R1, R2

1A	R_1	R_2
3 7	8 11	12 15

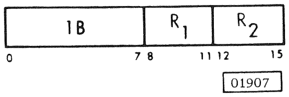
01906

The second operand is added to the first operand, and the sum is placed in the first operand location. The halfword contained in the register specified by the R1 field is added to the halfword contained in the register specified by the R2 field and the sum replaces the contents of the register specified by R1.

Addition is performed by adding all 16 bits of both operands. If the carries out of the sign-bit position and the high-order numeric bit position agree, the sum is satisfactory; if they disagree, an overflow occurs. The sign bit is not changed after the overflow. A positive overflow yields a negative final sum, and a negative overflow results in a positive sum. The overflow causes a binary-overflow-check condition. The condition code is set to 00, 01, or 10 to indicate that the result is zero, less than zero, or greater than zero.

Subtract (RR)

SR R1, R2



The second operand is subtracted from the first operand, and the difference is placed in the first operand location.

The halfword contained in the register specified by the R2 field is subtracted from the halfword contained in the register specified by the R1 field and the difference replaces the contents of the register specified by R1.

Subtraction is performed by adding the two's complement of the second operand to the first operand. All 16 bits of both operands participate, as in ADD. If the "carries" out of the sign-bit position and the high-order numeric bit position agree, the difference is satisfactory; if they disagree, an overflow occurs. The overflow causes a binary-overflow-check condition. The condition code is set to 00, 01, or 10 to indicate that the result is zero, less than zero, or greater than zero.

Binary Arithmetic Error Conditions

Error conditions which may occur during the instruction or execution phase of binary operations are:

1. Operation Code Invalid
2. Addressing Error
   a. An instruction address or an operand address refers to the protected first 144 bytes of main storage (addresses 0 to 143).
   b. An instruction address or an operand address is outside available storage.
   c. In 2020 Submodels 1 through 4, the last (highest) main-storage position contains any part of an instruction to be executed.
   d. The R1 or R2 fields of a binary instruction contain binary values 0 through 7.
3. Specification Error
   a. The low-order bit of an instruction address is one.
   b. The half-word second operand is not located on a 16-bit boundary.
   c. Bits 12 through 15 of an RX format instruction are not all zero.
4. Binary Overflow Check
5. CPU Parity Error

DECIMAL ARITHMETIC

Decimal arithmetic operations include addition, subtraction, multiplication, division, and comparison. Both operands and results are located in storage.

All decimal arithmetic operations are performed with data in the packed mode for optimum use of storage. Since data is often communicated to or from peripheral devices in the zoned form, operations are provided for changing from zoned to packed format and vice versa.

Operand fields can be located on any byte boundary, except for the first 144 bytes of main storage, which are protected. They can have a length of up to 31 digits and sign, except for multiplier and divisor operands which are limited to 15 digits.

Each address specifies the leftmost byte of an operand. Associated with each address is a length field which indicates the number of additional bytes that the operand extends beyond the first byte.

Data Format

Decimal operands reside in main storage only. They occupy fields that may start at any byte address and are composed of one to 16 eight-bit bytes. All operations use a two-address format.

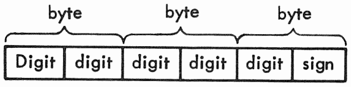
Lengths of the two operands specified in the instruction need not be the same. Regardless of length, the two operands are treated as if the rightmost integers were aligned before the operation begins. For example, in an add operation, the rightmost integers must be aligned to ensure that units are added to units, tens are added to tens, hundreds to hundreds, etc. The programmer may add high-order zeros as needed, to ensure that the result field is always large enough to contain all the digits of the result.

Processing takes place from right to left, as in simple manual arithmetic operations, treating the rightmost position of each operand as the units position, the next position to the left as the tens position, etc.

Lost carries or lost digits from arithmetic operations are signalled by a decimal-overflow condition. Operands are either in the packed or zoned format. Negative numbers are carried in true form.

Packed Decimal Number

In the packed format, each eight-bit byte can contain the binary equivalents to two decimal digits, except for the rightmost four bits of the rightmost byte of the field, which represent the sign. Decimal digits 0 through 9 are equivalent to binary values 0000 through 1001.

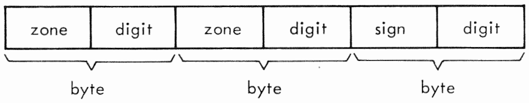


01888

The decimal digits range from 0 to 9 while any value from 10 to 15 (in hexadecimal representation 'A' to 'F') is regarded as a valid sign. In decimal arithmetic, all numbers must be arranged in the packed format. By means of the Pack instruction, it is possible to change zoned decimal numbers into packed numbers. Decimal number fields may overlap only when their rightmost bytes are identical. During the execution of all decimal arithmetic instructions, all data is checked for validity. A sign code found in any other but the sign position leads to an error stop as does a decimal value that is found in the sign position.

Zoned Decimal Numbers

In the zoned format, each eight-bit byte can contain one decimal digit in the right four bits and a zone code in the left four bits. However, the right most byte of a zoned decimal number field contains the digit in the right half and the sign in the left half.



01889

The zone bits act only as fill characters and do not affect the value of the decimal digits. The zoned format is needed for decimal data that is sent to character-set sensitive I/O devices. Similarly, all decimal data that is received from card I/O devices comes in the zoned format. The zone is either 15 (F) in the Extended Binary-Coded-Decimal Interchange Code (EBCDIC) or 5 in the United States of America Standard Code for Information Interchange (USASCII). Since the zone F is identical with one of the general plus signs, all data received from punched cards is recognized as positive unless it is explicitly specified as negative data. Data can be changed from the packed to the zoned format by means of the Unpack instruction. During unpacking, the correct zone is automatically supplied by the CPU. The zone used depends on the mode indicated in bit 6 of the PSW. If the mode bit is zero, the EBCDIC-zone F is used; if the mode bit is one, the USASCII-zone 5 is used.

Automatic Sign Standardization

During the processing of all arithmetic instructions that handle decimal data, the CPU supplies the final result with the correct sign according to the rules of algebra. The resultant sign is either a USASCII plus or minus sign, or an EBCDIC plus or minus sign. Whether the result sign is standardized in the USASCII mode or in the EBCDIC mode depends on mode bit 6 in the PSW. For example, if the result sign is positive and the mode bit is zero, the sign code 1100 is selected. If the result sign is negative and the mode bit is 1, the sign code 1011 is selected. Thus, the result can have only a USASCII- or an EBCDIC sign.

	Valid Sign Codes	USASCII	EBCDIC
Standardization	(A) 1010 +	+	
	(B) 1011 -	-	
	(C) 1100 +		+
	(D) 1101 -		-
	(E) 1110 +		
	(F) 1111 +		

01908

Condition Code

The results of all add-type and comparison operations are used to set the condition code in the PSW. All other decimal arithmetic operations leave the code unchanged. The condition code can be used for decision making by subsequent Branch-on-Condition instructions.

The condition code is set to reflect the types of results for decimal arithmetic. Four different result types can occur:

1. The result is zero, indicated by condition code 00.
2. The result is less than zero (negative), indicated by condition code 01.
3. The result is more than zero (positive), indicated by condition code 10.
4. The result does not fit into the result field (overflow) indicated by condition code 11.

For the comparison operation, the condition code settings 0, 1, and 2 indicate that the first operand compared equal, low, or high as shown in Table 2.

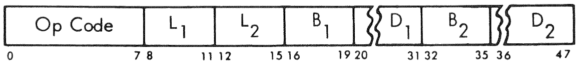
Table 2. Condition Code Settings for Decimal Operations 01909

	0	1	2	3
Add Decimal	zero	< zero	> zero	overflow
Compare Decimal	equal	low	high	
Subtract Decimal	zero	< zero	> zero	overflow
Zero and Add	zero	< zero	>zero	

Instruction Format

Decimal instructions use the following format:

SS Format



01910

In this format, the address which specifies the core storage location of the leftmost byte of the first operand field is derived from the contents of the B1 and D1 fields of the instruction. The number of operand bytes to the right of this byte is specified by the L1 field of the instruction. Therefore, the length in bytes of the first operand field is 1 to 16 corresponding to a length code in L1 of 0000 to 1111. The second operand field is specified similarly by the L2, B2, and D2 instruction fields.

As described in the section on Addressing, the address of each operand may be taken directly from the respective B and D fields of the instruction, or effective addresses may be formed by adding the contents of the general register specified by the B field to the contents of the D field.

Results of operations replace the first operand field. The result is never stored outside the field specified by the address and length. The second operand field remains unchanged, except in those cases where overlapping fields, which are permitted, actually occur. The contents of the general registers remain unchanged.

Instructions

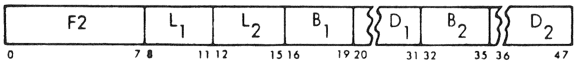
The decimal arithmetic instructions and their mnemonics and operation codes are shown in Table 3.

Table 3. Decimal Instructions 01911

Name	Mnemonic	Format	Operation Code
Pack	PACK	SS	F2
Unpack	UNPK	SS	F3
Move with offset	MVO	SS	F1
Zero and Add	ZAP	SS	F8
Add Decimal	AP	SS	FA
Subtract Decimal	SP	SS	FB
Compare Decimal	CP	SS	F9
Multiply Decimal	MP	SS	FC
Divide Decimal	DP	SS	FD

Pack (SS)

PACK D1 (L1, B1), D2 (L2, B2)



01912

The format of the second operand is changed from zoned to packed, and the result is placed in the first operand location.

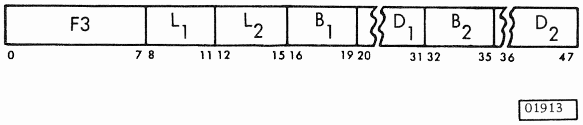
The second operand is assumed to have the zoned format. All zones are ignored, except the zone over the low-order digit, which is assumed to represent a sign. The sign is placed in the right four bits of the low-order byte, and the digits are placed adjacent to the sign and to each other in the remainder of the result field. The sign and digits are moved unchanged to the first operand field and are not checked for valid codes.

The fields are processed right to left. If the second operand does not fill the first operand completely, the remaining high-order positions of the first operand are filled with zeros. If the first operand field is too short to contain all the significant digits of the second operand field, the remaining digits are ignored. Overlapping fields may occur and are processed by storing each result byte immediately after the necessary operand bytes are fetched.

The condition code remains unchanged.

Unpack (SS)

UNPK D1 (L1, B1), D2 (L2, B2)



The format of the second operand is changed from packed to zoned, and the result is placed in the first operand location.

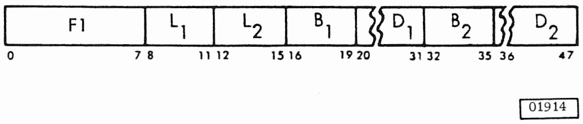
The digits and sign of the packed operand are placed unchanged in the first operand location, using the zoned format. Zones with coding 1111 in the binary-coded-decimal mode and coding 0101 in the USASCII mode are supplied for all bytes except the rightmost byte, which receives (in its high-order four bits) the sign of the packed operand. The operand sign and digits are not checked for valid codes.

The fields are processed right to left. If the second operand does not fill the first operand completely, the remaining high-order bytes of the first operand are each filled with a zero and a zone. If the first operand field is too short to contain all significant digits of the second operand, the remaining digits are ignored. The first and second operand fields may overlap and are processed by storing a result byte immediately after the necessary operand bytes are fetched. Caution must be exercised when overlapping the operands, otherwise bytes which have already been unpacked may be unpacked a second time during a single operation, thus giving an incorrect result.

The condition code remains unchanged.

Move with Offset (SS)

MVO D1 (L1, B1), D2 (L2, B2)



The second operand is placed to the left of and adjacent to the low-order four bits of the first operand.

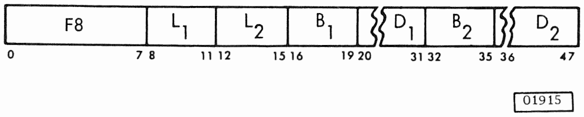
The low-order four bits of the first operand are attached as low-order bits to the second operand, the second operand bits are offset by four bit positions, and the result is placed in the first operand location. The first and second operand bytes are not checked for valid codes.

The fields are processed right to left. If necessary, the second operand is extended with high-order zeros. If the first operand field is too short to contain all bytes of the second operand, the remaining information is ignored. Overlapping fields may occur and are processed by storing a result byte as soon as the necessary operand bytes are fetched.

The condition code remains unchanged.

Zero and Add (SS)

ZAP D1 (L1, B1), D2 (L2, B2)



The second operand is placed in the first operand location.

The operation is equivalent to an addition to zero. The sign code is made 1100 for positive and 1101 for negative results in the binary-coded-decimal mode, and 1010 for positive and 1011 for negative results in the USASCII mode. A zero result is always positive.

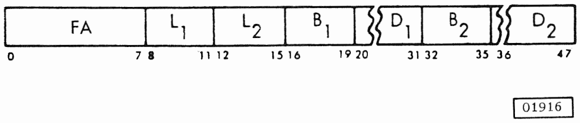
When the length of the second operand (L2) is greater than the length of the first operand (L1) a specification error stop occurs. The instruction is not executed.

The second operand is checked for valid sign and digit codes. Extra high-order zeros are supplied if needed. The first and second operand fields may overlap when the rightmost byte of the first operand field is coincident with or to the right of the rightmost byte of the second operand.

The condition code is set to 00, 01, or 10 to indicate that the result is zero, less than zero, or greater than zero.

Add Decimal (SS)

AP D1 (L1, B1), D2 (L2, B2)



The second operand is added to the first operand, and the sum is placed in the first operand location.

When the length of the second operand (L2) is greater than the length of the first operand (L1), a specification-error stop occurs. The instruction is not executed.

Addition is algebraic, taking into account the sign and all digits of both operands. All signs and digits are checked for validity. If necessary, high-order zeros are supplied for the second operand.

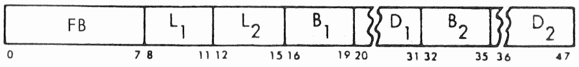
The first and second operand fields may overlap when their low-order bytes coincide; therefore, it is possible to add a number to itself.

The sign of the result is determined by the rules of algebra. A zero sum is always positive. When high-order digits are lost because of an overflow, a zero result has the sign of the correct sum.

The condition code is set to 00, 01, 10 or 11 to indicate that the result is zero, less than zero, greater than zero, or overflow. A decimal overflow is not considered to be an error.

Subtract Decimal (SS)

SP D1 (L1, B1), D2 (L2, B2)



01917

The second operand is subtracted from the first operand, and the difference is placed in the first operand location.

When the length of the second operand (L2) is greater than the length of the first operand (L1) a specification error stop occurs. The instruction is not executed.

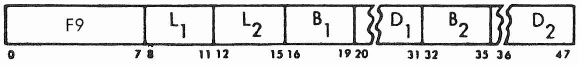
Subtraction is algebraic, taking into account the sign and all digits of both operands. The Subtract Decimal instruction is similar to Add Decimal instruction, except that the sign of the second operand is changed from positive to negative or from negative to positive after the operand is obtained from storage and before the arithmetic operation.

The sign of the result is determined by the rules of algebra. A zero difference is always positive. When high-order digits are lost because of overflow, a zero result has the sign of the correct difference.

The operands of a Subtract instruction may overlap when their low-order bytes coincide even when their lengths are unequal. This property may be used to make an entire field or the low-order part of a field zero. The condition code is set to 00, 01, 10, or 11 to indicate that the result is zero, less than zero, greater than zero, or an overflow condition. A decimal overflow is not considered to be an error.

Compare Decimal (SS)

CP D1 (L1, B1), D2 (L2, B2)



01918

The first operand is compared with the second, and the condition code indicates the comparison result.

When the length of the second operand (L2) is greater than the length of the first operand (L1), a specification-error stop occurs. The instruction is not executed.

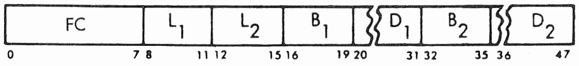
Comparison is right to left, taking into account the sign and all digits of both operands. All signs and digits are checked for validity. If the second operand field is shorter in length than the first operand field, the second operand field is extended with high-order zeros. A positive zero compares equal to a negative zero. Neither operand is changed as a result of the operation. Overflow cannot occur in this operation.

The first and second fields may overlap when their low-order bytes coincide. It is possible, therefore, to compare a number to itself.

A Decimal Compare instruction differs in several respects from a Logical Compare instruction. Signs, zeros, and invalid characters are considered and fields are extended when unequal in length. The condition code is made 00 if the operands are equal, 01 if the first operand is low, and 10 if the first operand is high.

Multiply Decimal (SS)

MP D1 (L1, B1), D2 (L2, B2)



01919

The product of factor one (the first operand) times factor two (the second operand) replaces the first operand. A multiplication can be performed only on data in the packed format. The length codes contained in the L1 and L2 fields specify the number of bytes that extend to the left of the units byte. The units byte is the rightmost byte of a packed decimal operand and it contains one digit and the sign. Factor two is limited to a size of 15 digits plus the sign, or in other words, the length code of factor two must not be greater than 7. Furthermore, the length code of factor two must always be smaller than the length code of factor 1. If L2 is greater or equal to L1, or if L2 exceeds 7, a program error stop occurs.

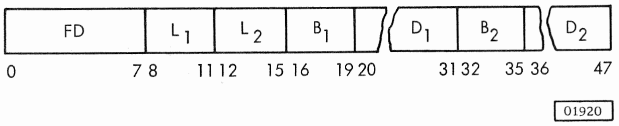
Factor one is limited to a size of 16 bytes (length code 15). The main storage location of factor one is specified by the B1-D1 field according to the rules for direct or effective-address generation. Similarly, the location of factor two is specified by the B2-D2 field.

Since the number of digits in the product (the result) is the sum of the number of digits in both operands, factor one must have as many leading (high order) zero bytes as factor two has data bytes (L2 +1). This requirement for the factor 1 field ensures that no product overflow can occur. If factor 1 has insufficient high order zeros, a program error stop occurs. The sign of the product is developed from the signs of both operands according to the rules of algebra. This is true also when one or both operands are zero.

The factor two field may overlap the product field when the low-order bytes of both fields coincide. The condition code is not changed by a Multiply Decimal instruction.

Divide Decimal (SS)

DP D1 (L1,B1), D2 (L2,B2)



The dividend (the first operand) is divided by the divisor (the second operand) and is replaced by the quotient and remainder. A divide operation must be performed on data in the packed format only. Length codes, contained in the L1 and L2 fields, specify the number of bytes to the left of the units byte. Each byte contains two arithmetic digits, except the units byte which contains one digit and a sign. The location of the dividend is specified by the B1-D1 field, and the location of the divisor is specified by the B2-D2 field. The result consists of the quotient and the remainder.

The quotient field is placed leftmost in the first operand field. The remainder field is placed rightmost in the first operand field and has a size equal to the divisor size. Together, the quotient and remainder occupy the entire former dividend field; therefore, the address of the quotient field is the address of the first operand. The size of the quotient field in eight -bit bytes is L1−L2, and the length code for this field is one less (L1−L2−1). When the divisor length code is larger than seven (15 digits and sign) or larger than or equal to the dividend length code, the operation is not executed and an error stop occurs.

The dividend, divisor, quotient, and remainder are all signed integers, right-aligned in their fields. The sign of the quotient is determined by the rules of algebra from dividend and divisor signs. The sign of the remainder is the same as the dividend sign. These rules are true even when quotient or remainder is zero.

A divide check occurs when the quotient is larger than the number of digits allowed for it, or if the dividend does not have at least one leading zero. If a divide check occurs, the operation is not executed and a decimal-divide-error stop occurs. Divisor and dividend remain unchanged in their storage locations.

The divisor and dividend fields may overlap only if their low-order bytes coincide.

The condition code remains unchanged for division, and overflow cannot occur.

Programming Notes

The maximum dividend size is 30 digits and sign (16 packed bytes). Since the smallest remainder size is one digit and sign, the maximum quotient size is 29 digits and sign.

The condition for a divide check can be determined by a trial subtraction. The leftmost digit of the divisor field is aligned with the leftmost-plus-one digit of the dividend field. When the divisor, so aligned, is less than or equal to the dividend, a quotient overflow is indicated.

By programming a Divide Decimal instruction so that the dividend field (the first operand) has a number of high-order zero digits equal to the field size of the divisor field (the second operand), a divide check occurs only when a division by zero is attempted.

Decimal Arithmetic Error Conditions

The following error conditions may occur during the instruction or execution phases of decimal arithmetic operations:

1. Operation Code Invalid
2. Addressing Error
   a. An instruction address or an operand ad-

dress refers to the protected first 144 bytes of main storage.
   b. An instruction address or an operand address is outside available storage.
   c. An instruction occupies the last two (highest) main-storage positions (2020 Submodels 1 through 4 only).
3. Specification Error
   a. The low-order bit of an instruction address is 1.
   b. For Zero and Add, Compare Decimal, Add Decimal, and Subtract Decimal instructions, the length code L2 is greater than the length code L1.
   c. For Multiply Decimal and Divide Decimal instructions, the length code L2 is greater than 7 or greater than or equal to the length code L1.
4. Data Error
   a. A sign or digit code of an operand in the Zero and Add, Compare Decimal, Add Decimal, Subtract Decimal, Multiply Decimal, or Divide Decimal instruction is incorrect, or the operand fields in these instructions overlap incorrectly.
   b. The factor one field (first operand) in a Multiply Decimal instruction has insufficient high-order zeros.
5. Decimal Divide Check
   a. The resultant quotient in a Divide Decimal instruction exceeds the specified data field instruction (including division by zero) or the dividend has no leading zero.
6. CPU Parity Error.

LOGICAL OPERATIONS

A set of operations is provided for the logical manipulation of data. Generally the operands are treated as characters, eight bits at a time. In a few cases the left or right four bits of a byte are treated separately. The operands are either in storage or are introduced from the instruction stream.

Processing of data in storage proceeds left to right through fields which may start at any byte position.

Except for the Edit instruction, data is not treated as numbers in this section. Editing provides a transformation from packed decimal digits to alphameric characters.

The set of logical operations include move, comparison, editing, testing, and bit-connective operations.

The condition code is set as a result of all logical connective, comparison, editing, and testing operations.

Data Format

Data resides in storage, or is introduced from the instruction itself. The data size may be a single character, or variable in length. When two operands participate they have equal length, except in the Edit instruction.

In storage-to-storage operations, data has a variable field-length format, starting at any byte address (except for the first 144 storage-protected bytes) and continuing to a maximum of 256 bytes. Processing is left to right.

Operations which introduce data from the instruction into storage are restricted to an eight-bit byte. Only one byte is introduced from the instruction, and only one byte in storage participates.

Editing requires a packed decimal field; otherwise no internal data structure is required and all bit configurations are considered valid.

In storage-to-storage operations, the operand fields may be defined in such a way that they overlap. The effect of this overlap depends upon the operation. When the operands remain unchanged, as in compare, overlapping does not affect the execution of the operation. In the case of move and edit operations, one operand is replaced by new data, and the execution of the operation may be affected by the amount of overlap and the manner in which data is fetched or stored. For purposes of evaluating the effect of overlapped operands, it can be considered that data is always handled one eight-bit byte at a time. All overlapping fields are considered valid, but in the edit operations overlapping fields give unpredictable results.

Condition Code

The results of most logical operations are used to set the condition code. The move operations leave this code undisturbed. The condition code can be used for decision-making by subsequent branch on condition instructions.

There are four types of condition code settings for logical instructions. For the Edit instruction the codes 00, 01, and 10 indicate a zero, less than zero, and greater than zero content of the last result field.

For the logical connective operations, the codes 00 and 01 indicate a zero or nonzero result field.

For the Test Under Mask instruction, the codes 00, 01, and 11 indicate a zero, mixed zero and one, and all-one result field.

For the Compare Logical instruction, the codes 00, 01, and 10 indicate that the first operand compared equal, low or high. Table 4 shows the condition code for logical operations.

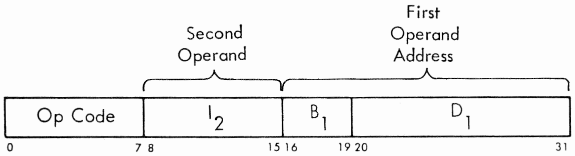
Table 4. Condition Code Settings for Logical Operations 01921

	00	01	10	11
Test under Mask	zero	mixed	- -	one
And	zero	not zero	- -	- -
Compare Logical	equal	low	high	- -
Or	zero	not zero	- -	- -
Edit	zero	< zero	> zero	- -

Instruction Format

Logical instructions use the SI or SS formats.

SI Format

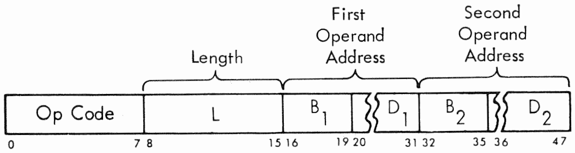


01922

In the SI format, the address which specifies the core storage location of the first operand field is derived from the contents of the B1-D1 fields of the instruction. The address may be taken directly from the B1-D1 Fields (direct addressing) or an effective address may be formed by adding the contents of the general register specified by the B1 field to the contents of the D1 field (effective addressing).

Results replace the first operand. The contents of the general registers are not changed.

SS Format



01923

The address which specifies the core-storage location of the first operand field is derived from the B1-D1 fields of the instruction. The B2-D2 fields specify the leftmost byte of the second operand field. The first and second operand fields are the same length. The number of bytes extending to the right of the first byte is specified by the L field of the instruction.

The address of each operand may be derived by either direct or effective addressing. The result of the operation replaces the first operand, and is never stored outside the field specified by the address and length fields of the instruction. The contents of the general registers remain unchanged.

Instructions

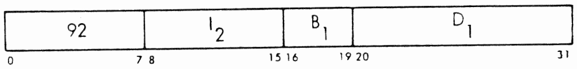
The logical instructions, their mnemonics, formats, and operation codes are shown in Table 5.

Table 5. Logical Instructions 01925

Name	Mnemonic	Format	Operation Code
Move	MVI	SI	92
Move	MVC	SS	D2
Move Numerics	MVN	SS	D1
Move Zones	MVZ	SS	D3
Compare Logical	CLI	SI	95
Compare Logical	CLC	SS	D5
Edit	ED	SS	DE
And	NI	SI	94
Or	OI	SI	96
Test under Mask	TM	SI	91
Halt & Proceed	HPR	SI	99
Translate	TR	SS	DC

Move (SI)

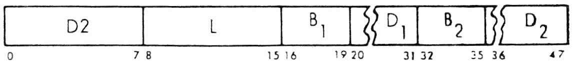
MVI D1 (B1), I2



01924

Move (SS)

MVC D1 (L, B1), D2 (B2)



01926

The second operand is placed in the first operand location.

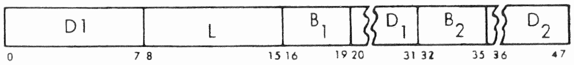
The SS format is used for a storage-to-storage move. The SI format introduces one eight-bit byte from the instruction stream.

In storage-to-storage movement, the fields may overlap in any desired way. Movement is left to right through each field, a byte at a time.

The bytes to be moved are not changed or inspected. The condition code remains unchanged. It is possible to propagate one character through an entire field by having the first operand field start one character to the right of the second operand field.

Move Numerics (SS)

MVN D1 (L, B1), D2 (B2)



01927

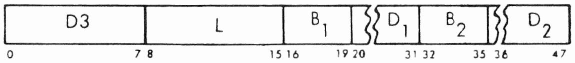
The low-order four bits of each byte in the second operand field, the numerics, are placed in the low-order bit positions of the corresponding bytes in the first operand field.

The instruction is storage-to-storage. Movement is left to right through each field, one byte at a time, and the fields may overlap in any desired way.

The numerics are not changed or checked for validity. The high-order four bits of each byte, the zones, remain unchanged in both operand fields. The condition code remains unchanged.

Move Zones (SS)

MVZ D1 (L, B1), D2 (B2)



01928

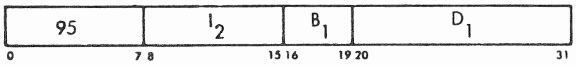
The high-order four bits of each byte in the second operand field, the zones, are placed in the high-order four bit positions of the corresponding bytes in the first operand field.

The instruction is storage-to-storage. Movement is left to right through each field one byte at a time, and the fields may overlap in any desired way.

The zones are not changed or checked for validity. The low-order four bits of each byte, the numerics, remain unchanged in both operand fields. The condition code remains unchanged.

Compare Logical (SI)

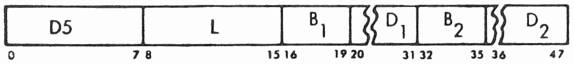
CLI D1 (B1), I2



01929

Compare Logical (SS)

CLC D1 (L, B1), D2 (B2)



01930

The first operand is compared with the second operand, and the result is indicated in the condition code.

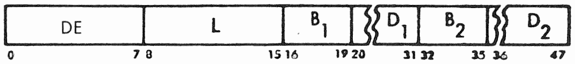
The instructions allow comparisons that are instruction-to-storage, and storage-to-storage.

Comparison is binary, and all codes are valid. The operation proceeds left to right and terminates as soon as an inequality is found. The condition code is made 00 if the operands are equal, 01 if the first operand is low compared to the second operand, and 10 if the first operand compares high.

In the Compare Logical instruction, all bits are treated alike as part of an unsigned binary quantity. In the variable length storage-to-storage operation, comparison is left to right and may extend to field lengths of 256 bytes. This instruction may be used for alphameric comparison.

Edit (SS)

ED D1 (L, B1), D2 (B2)



01931

The format of the source (the second operand) is changed from packed to zoned and is edited under control of the pattern (the first operand). The edited result replaces the pattern.

Editing includes sign and punctuation control and the suppressing and protecting of leading zeros. It also facilitates programmed blanking of all-zero fields. Several numbers may be edited in one operation, and numeric information may be combined with text.

The length field applies to the pattern (the first operand). The pattern has the unpacked format and may contain any character. The source (the second operand) has the packed format and must contain valid decimal digit and sign codes. The left four bits of a byte must be 0000-1001. The right four bits are recognized as either a sign or a digit.

Both operands are processed left to right, one character at a time. Overlapping pattern and source fields give unpredictable results.

The character to be stored in the first operand field is determined by three things; the digit obtained from the source field, the pattern character, and the state of a trigger, called the S trigger. One of three actions may be taken:

1. The source digit is expanded to zoned format and is stored into the first operand.
2. The pattern character is left unchanged.
3. A fill character is stored into the first operand.

S Trigger: The S trigger is used to control the storing or replacing of source digits and pattern characters. Digits to be stored in the result, whether zero or not, are termed significant. Pattern characters are replaced or stored when they are significance-dependent (such as punctuations) or sign-dependent (such as credit symbols). The S trigger is also used to record the sign of the source and it sets the condition code accordingly.

The S trigger is set to the 0 state at the start of the operation and is subsequently changed depending upon the source number and the pattern characters.

Pattern Character: Three pattern characters have a special use in editing. They are the digit-select character, the significance-start character, and the field-separation character. These three characters are replaced, either by a source digit or by a fill character; their encoding is shown in Table 6.

1. The digit-select character causes either a source digit or the fill character to be inserted in the result field.
2. The significance-start character has the same function but also indicates, by setting the S trigger, that the following digits are significant.
3. The field-separation character identifies individual fields in a multiple-field editing operation. The character is replaced by the fill character. The S trigger is set to zero, and testing for a zero-field is reinitiated.
4. All other pattern characters are treated in a common way; if the S trigger is 1, the pattern character is left unchanged; if the S trigger is 0, the pattern character is replaced by the fill character.

If the pattern character is either a digit-select or a significance-start character, the source digit is examined. The source digit replaces the pattern character if the S trigger is 1 or if the source digit is nonzero. If a nonzero digit is inserted when the S trigger is 0, the S trigger is set to 1 to indicate that the subsequent digits are significant. If the S trigger and the source digit are both 0, the fill character is substituted for the pattern character.

Source Digit: When the source digit is stored in the result, its code is expanded from the packed to the zoned format by attaching a zone. The zone code is 1111 in the binary-coded-decimal mode and 0101 in the USASCII mode. The type of zone used depends on the mode bit 6 of the PSW. For example, if the mode bit is 0, 1111 (the EBCDIC zone) is used.

The source digits are examined only once during an editing operation. They are selected eight bits at a time from the second operand field. The leftmost four bits are examined first, and the rightmost four bits remain available for the next pattern character which calls for a digit examination. However, the rightmost four bits are inspected for a sign code immediately after the leftmost four bits are examined.

Any of the plus-sign codes 1010, 1100, 1110, or 1111 sets the S trigger to 0 after the digit is inspected, whereas the minus-sign codes 1011 and 1101 leave the S trigger unchanged. When one of these sign codes is encountered in the four right-most bits, these bits no longer are treated as a digit, and a new character is fetched from storage for the next digit to be examined.

A plus sign sets the S trigger to 0 even if the trigger was set to 1 for a nonzero digit in the same source byte or by a significance-start character for that digit.

Fill Character: The fill character is obtained from the pattern as part of the editing operation. The first character of the pattern is used as a fill character and is left unchanged in the result field, except when it is the digit-select or significance-start character. In the latter cases, a digit is examined and, when nonzero, inserted.

Result Condition: To facilitate the blanking of all-zero fields, the condition code is used to indicate the sign and zero status of the last field edited. All digits examined are tested for the code 0000. The presence or absence of an all-zero source field is recorded in the condition code at the termination of the editing operation.

1. The condition code is made 0 for a zero source field, regardless of the state of the S trigger.

Table 6. Edit Characteristics 01932

Character Code	Name and Purpose	Examine Digit	Trigger Status	Digit Status	Result Character	Trigger Set
0010 0000	digit select	yes	s = 1		digit	
			s = 0	d not 0	digit	s = 1
			s = 0	d = 0	fill	
0010 0001	significance start	yes	s = 1		digit	
			s = 0	d not 0	digit	s = 1
			s = 0	d = 0	fill	s = 1
0010 0010	field separator	no			fill	s = 0
other	message insertion	no	s = 1		leave	
			s = 0		fill	

Notes:

d	Source digit
s	S trigger (1: minus sign, digits, or pattern used; 0: plus sign, fill used)
digit	A source digit replaces the pattern character.
fill	The fill character replaces the pattern character.
leave	The pattern character remains unchanged.

2. For a nonzero source field and an S trigger of 1, the code is made 1 to indicate less than zero.
3. For a nonzero source field and an S trigger of 0, the code is made 2 to indicate greater than zero.

The condition-code setting pertains to fields as specified by the field-separator characters, regardless of the number of signs encountered.

For the multiple-field editing operations, the condition-code setting reflects only the field following the last field-separator character. When the last character of the pattern is a field-separator character, the condition code is made 0.

Table 6 gives the details of the edit operation. The leftmost columns give the pattern character and its code. The next columns show the states of the digit and the S trigger used to determine the resulting action. The rightmost column shows the new setting of the S trigger.

The following example shows the step-by-step editing of a packed field with a length specification of four against a pattern 13 bytes long. The following symbols are used:

Symbol	Meaning
b	blank character
((hexadecimal 21)	significance-start character
) (hexadecimal 22)	field-separation character
d (hexadecimal 20)	digit-select character

Assume:

Loc 1000-1012 (first operand)	bdd, dd(. ddbCR
Loc 1200-1203 (second operand)	02 57 42 6+
Reg 12 (decimal equivalent)	1000

The instruction is:

Op Code	L	B_1	D_1	B_2	D_2
ED	12	12	0	12	200

01933

and provides the following

Pattern	Digit	S Trigger	Rule	Location 1000-1012
b		0	leave(1)	bdd, dd(. ddbCR
d	0	0	fill	bbd, dd(. ddbCR
d	2	1	digit	bb2, dd(. ddbCR(2)
,		1	leave	same
d	5	1	digit	bb2, 5d(. ddbCR
d	7	1	digit	bb2, 57(. ddbCR
(	4	1	digit	bb2, 574. ddbCR
.		1	leave	same
d	2	1	digit	bb2, 574. 2dbCR
d	6+	0	digit	bb2, 574. 26bCR(3)
b		0	fill	same
C		0	fill	bb2, 574. 26bbR
R		0	fill	bb2, 574. 26bbb

Thus:
Loc 1000-1012 (after) bb2, 574. 26bbb

NOTES
1. This character is saved as the fill character.
2. First nonzero digit sets S trigger to 1.
3. Plus sign in this same byte sets S trigger to zero.
Condition code = 2; result greater than zero.

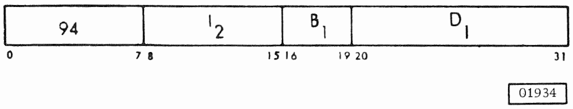
If the second operand in location 1200-1203 is 00 00 02 6-, the following results are obtained:

Loc 1000-1012 (before)	bdd, dd(. ddbCR
Loc 1000-1012 (after)	bbbbbb. 26bCR
Condition code = 1; result less than zero	

In this case the significance-start character in the pattern causes the decimal point to be left unchanged. The minus sign does not reset the S trigger so that the CR symbol is also preserved.

And (SI)

NI D1 (B1), I2



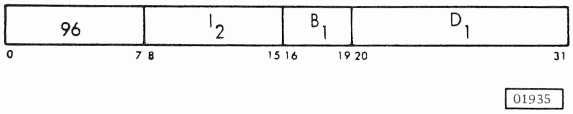
This instruction uses the SI format, which provides a single character instruction-to-storage operation. This instruction may be used to set a bit to 0.

The logical product (And) of the bits of the first and second operand is placed in the first operand location.

Operands are treated as unstructured logical quantities, and the connective And is applied bit by bit. All operands and results are valid. The condition code is set to zero (00) or not zero (01) according to the result of the operation. (See Table 4.)

Or (SI)

OI D1 (B1), I2



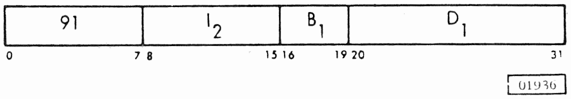
This instruction uses the SI format, which provides a single character instruction-to-storage operation. This instruction may be used to set a bit to 1.

The logical sum (Or) of the bits of the first and second operand is placed in the first operand location. The condition code is set to zero (00) or not zero (01) according to the result of the operation. (See Table 4.)

Operands are treated as unstructured logical quantities, and the connective inclusive Or is applied bit by bit. All operands and results are valid.

Test Under Mask (SI)

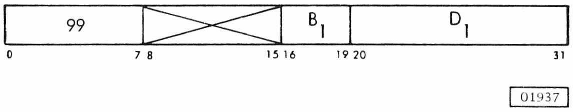
TM D1 (B1), I2



This instruction inspects the first operand bits and compares them with the I2 field, which is the mask. If the first operand has 1 bits in exactly the same positions as the mask, the condition code is set to 11. If the 1 bits in the mask match some of the 1 bits in the first operand, the condition code is set to 01. If none of the 1 bits match or if the mask is zero, the condition code is set to 00. Neither the first operand nor the mask is changed.

Halt and Proceed (SI)

HPR D1 (B1), I2



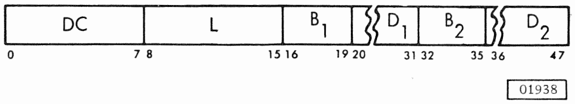
The Halt and Proceed instruction is used to stop the CPU. All input/output operations in progress are continued to completion.

When the CPU has been stopped by the Halt and Proceed instruction, the program may be resumed with the next sequential instruction by operating the Start key on the CPU console.

This instruction uses the SI format in which the I2 field is ignored. The direct or effective address derived from the B1-D1 fields may be used to identify the Halt and Proceed instruction. The condition code remains unchanged.

Translate (SS)

TR D1 (L, B1), D2 (B2)



The first operand contains the data that is to be translated. The second operand represents the translating list. The first operand is selected byte by byte proceeding from left to right. The binary value of each operand 1 byte (the argument) is added to the second operand (left byte) address. The resultant new address is used to select an appropriate byte from the translating list (second operand) which contains the function bytes. The selected function byte replaces the original argument byte in the first operand.

All data is valid. The operation proceeds until the first operand field is exhausted. It is permissible for the list and the first operand field to overlap. The condition code remains unchanged.

Error Conditions

Error conditions which may occur during the instruction or execution phase of logical operations are:

1. Operation Code Invalid
2. Addressing Error
   a. An instruction address or an operand address refers to the protected first 144 bytes of main storage (addresses 0 to 143).
   b. An instruction address or an operand address is outside available storage.
   c. In 2020 Submodels 1 through 4, the last (highest) main-storage position contains any part of an instruction to be executed.
3. Specification Error
   a. The low-order bit of an instruction address is 1.
4. Data Error
   a. An invalid digit code is contained within the second operand field of an Edit operation.
5. CPU Parity Error.

BRANCHING

Instructions are performed by the CPU primarily in the sequential order of their locations. A departure from this normal sequential operation may occur when branching is performed. The branching instructions provide a means for making a two-way choice, to reference a subroutine, or to repeat a segment of coding, such as a loop.

Branching is peformed by introducing a branch address as a new instruction address.

The branch address may be obtained from one of the general registers or it may be the address specified by the instruction. The branch address is independent of the updated instruction address. Branching may be conditional or unconditional. Unconditional branches always replace the updated instruction address with the branch address. Conditional branches may use the branch address or may leave the updated instruction address unchanged. When branching takes place, the instruction is called successful; otherwise, it is called unsuccessful.

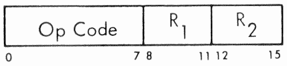
Whether a conditional branch is successful depends on the result of operations preceding the branch. An example is the Branch on Condition instruction, which inspects the condition code that reflects the result of a previous arithmetic or logical operation.

During a branching operation, the rightmost half of the PSW, the updated instruction address, may be stored before the instruction address is replaced by the branch address. The stored information may be used to link the new instruction sequence with the preceding sequence.

Instruction Format

Branching instructions employ the RR, RX, and the SI formats.

RR Format

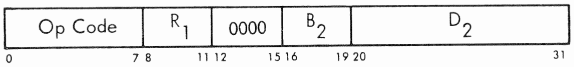


01939

The R1 field may specify a general register into which the updated instruction address is to be stored as link information, or may contain a mask which is employed to identify the bit values of the condition code.

The R2 field specifies the general register which contains the branch address.

RX Format

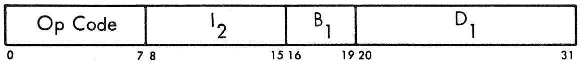


01940

The R1 field may specify a general register into which the updated instruction address is to be stored as link information, or may contain a mask which is employed to identify the bit values of the condition code.

The direct or effective address derived from the B2-D2 fields is the branch address.

SI Format



01941

The SI format is employed by only one branching instruction, Set PSW. The direct or effective address derived from the B1-D1 fields specifies the location of a word in main storage which is to replace the Program Status Word (PSW); the contents of the I2 field are ignored.

Instructions

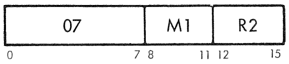
The branch instructions, their operation codes, formats, and mnemonics are shown in Table 7.

Table 7. Branch Instructions 01942

Name	Mnemonic	Format	Operation Code
Branch on Condition	BCR	RR	07
Branch on Condition	BC	RX	47
Branch & Store	BASR	RR	0D
Branch & Store	BAS	RX	4D
Set PSW	SPSW	SI	81

Branch on Condition (RR)

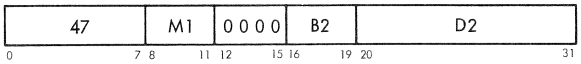
BCR M1, R2



01943

Branch on Condition (RX)

BC M1, D2 (0, B2)



01944

The updated instruction address is replaced by the branch address if the state of the condition code is as specified by M1; otherwise, normal instruction sequencing proceeds with the updated instruction address.

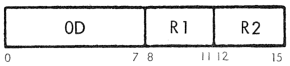
The M1 field is used as a 4-bit mask. The four bits of the mask correspond, left to right, with the four condition codes, 0, 1, 2, and 3, as shown in the following table. The branch is successful whenever the condition code has a corresponding mask bit of one. The condition code is not changed.

Condition Code	Instruction Bits
0 (0 0)	8
1 (0 1)	9
2 (1 0)	10
3 (1 1)	11

When all four mask bits are ones, the branch is unconditional. When two mask bits are ones (for example, bits 8 and 9) the branch will occur when the condition code is either 0 or 1. When all four mask bits are zero or when the R2 field in the RR format contains zero, the branch instruction is equivalent to a no operation. Refer to Appendix G for a table of all operations which affect the condition codes.

Branch and Store (RR)

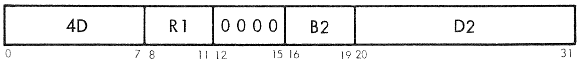
BASR R1, R2



01945

Branch and Store (RX)

BAS R1, D2 (0, B2)



01946

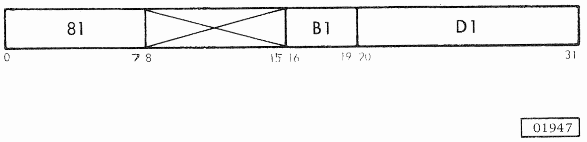
The rightmost 16 bits of the PSW, the updated instruction address, are stored as link information in the general register specified by R1. Simultaneously the instruction address is replaced by the branch address.

The condition code remains unchanged.

When in the RR format, the R2 field contains zero, the link information is stored without branching.

Set PSW (SI)

SPSW D1 (B1)

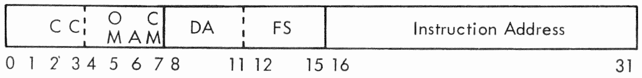


The 32-bit word (four eight-bit bytes), located in main storage with the leftmost byte at the first operand address, replaces the Program Status Word.

Bits 8-15 in the Set PSW instruction are ignored.

The PSW has a fixed length format of one word. It is located in an unaddressable register in the CPU and is employed as an internal control. Since the PSW contains the address of the next sequential instruction, the Set PSW instruction is equivalent to a branch operation.

PSW Format



0 - 1	Not Used
2 - 3	Condition Code
4	Not Used
5	Overlap Mode (Submodel 5 only)
6	USASCII Mode Bit
7	Channel Mask
8 - 11	Device Address
12 - 15	Function Specification
16 - 31	Instruction Address

01895

Programming Notes

1. The instruction address portion of the word which is transferred from main storage to the PSW by the Set PSW instruction should:
   a. Not refer to the protected first 144 bytes of main storage,
   b. Have the least significant bit zero, and
   c. Be within the limits of available storage.

   If these conditions are not satisfied, an addressing or specification error stop will occur.
2. The condition code is set by the Set PSW instruction to the value contained in the word transferred from main storage to the PSW.
3. Main-storage boundaries are not required of the first operand address in the Set PSW instruction.
4. The condition code, USASCII mode bit, channel mask, and overlap mode bit in the PSW are zero when the CPU is in the reset state. The instruction address portion of the PSW is not changed when the CPU is reset.

Error Conditions

Error conditions which may occur during a branching operation are:

1. Operation Code invalid.
2. Addressing Error.
   a. An instruction address or a branch address either refers to the protected first 144 bytes of main storage, or is outside available storage. If an incorrect branch address has been specified, 2020 Submodels 1 through 4 indicate the error during the execution of the branch instruction; 2020 Submodel 5, however, does not indicate the error until an attempt has been made to execute the instruction located at the branch address.

   NOTE: When the branching condition is not met and the branch address is invalid, either an addressing error indication or a no-operation may result.

   b. The R1 field of a Branch and Store instruction contains binary values 0 through 7, or the R2 field of an RR format branch instruction contains binary values 1 through 7.
   c. An instruction part is located in the highest two main-storage positions (2020 Submodels 1 through 4 only).
3. Specification Error.
   a. The low-order bit of an instruction address is 1.
   b. Bits 12 through 15 of an RX format instruction are not all 0.
4. CPU Parity Error.

CPU CONSOLE

The CPU console (Figure 5) provides the switches, keys, and lights necessary to operate and control the system.

Additional switches and indicators are located on the various input/output devices which are included in the system. These switches and indicators control functions or indicate conditions peculiar to the input/output unit on which they are located, and are not considered as system controls.

Operating Keys and Indicators

Register Display Indicators

The eight primary data registers in the CPU are displayed on the console.

System Reset Key

Operation of the System Reset key stops the CPU immediately, including all I/O operations which may be in progress. All error conditions are reset. The condition code, channel mask, USASCII mode bit, and overlap mode bit in the PSW are reset to zero. A system reset sets the mode to EBCDIC. The instruction address portion of the PSW is not changed. All registers displayed on the console are reset to zero, i.e., all bit-lights are off except the parity bits (P). The system reset function is also performed when the power on or program load sequences are initiated.

NOTE (2020 Submodel 5 only): If operation of the System Reset key fails to cause the system reset function to be performed, the operator should reload the control program into the system. The reset function can then be performed by pressing the System Reset key.

Start Key

The Start key is used to start or resume operation of the CPU.

Combined Stop Key and Indicator

Operation of this key stops the CPU at the completion of the execution of the instruction in progress when the key was depressed. All time-shared I/O operations in progress continue to completion.

The Stop key indicator is on when the CPU is stopped by any stop condition.

NOTE: The CPU Stop light may blink on and off during card-document printing operations on the IBM 2560 Multi-Function Card Machine Model A1. This action does not signify any malfunction.

Program Load Key

The following conditions must be established before the program load function is operable:

1. The Mode switch on the CPU console must be in the Process, Address Stop, or Instruction Step position.
2. Card I/O No. 1 device must be in a Ready Condition.

When these conditions are established, operation of the Program Load key initiates the system reset function, followed by a read-card operation starting at the main-storage location set in the Address/Register Data switches.

If the Mode switch is in any position other than Process, Address Stop, or Instruction Step when the Program Load key is operated, 2020 Submodels 1 through 4 perform the system reset function and then stop, whereas 2020 Submodel 5 performs the system reset function and loads.

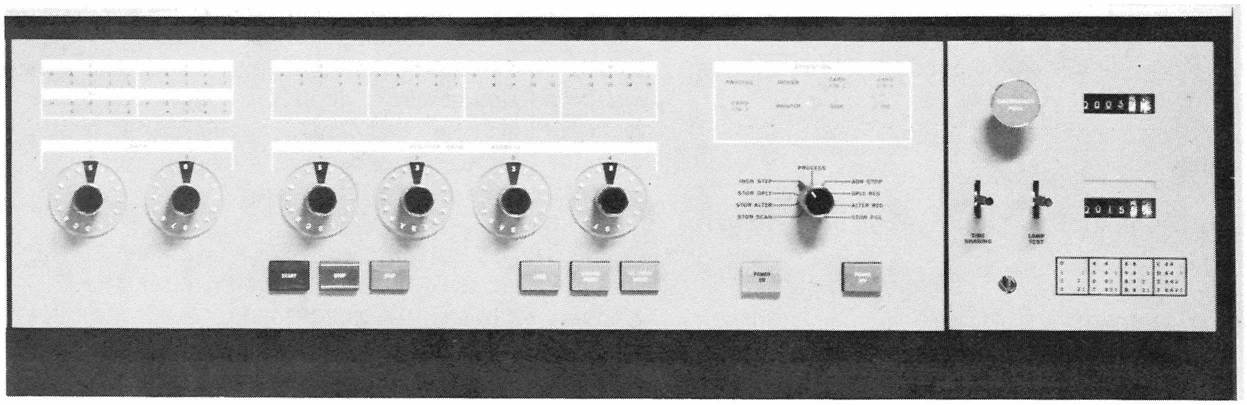


Figure 5. 2020 Console 01948

Two card I/O devices capable of reading cards can be attached to the 2020.

In order that common program load routines may be employed on systems with different card I/O equipment, device 2 responds to the device 1 (2501 Card Reader) instruction when device 1 is not included in the system.

The device 1 instructions to which device 2 responds are the following:

Device 2 Response	Device 1 Instruction
Read (Primary feed if 2560)	Read Card
Branch if Reader Busy	Branch if Reader Busy
Branch if Reader Error	Branch if Reader Error

A device 2 instruction in a system which does not include device 2 is treated as a no-operation.

The Program Load key may be used also for loading the microprogram into the control storage if the CPU is a 2020 Submodel 5. The control program is loaded in the following way:

1. Place the control program card deck into either the card hopper of the IBM 2501 Card Reader or, if this I/O device is not attached, into the primary hopper of the IBM 2560 Multi-Function Card Machine.
2. Turn the Mode switch to the CPL position.
3. Press the Program Load key.

The program will now be loaded and, after completion, the CPU will be stopped in a reset state. After the Mode switch has been turned to Process, the system is ready to load any other program.

I/O Check Reset Key

Operation of the I/O Check Reset key resets all program testable I/O error indicators. Further details are provided in the section on Operating Conditions.

Power On Key

The power-on sequence is initiated when this key is operated if power is not on. Operation of the Power On key has no effect if power is on.

Power Off Key

Operation of the Power Off key removes power from the system.

Attention Indicators

The Attention indicators show the particular condition or the particular I/O device that has caused the system to stop. In this manner the operator is guided quickly to the respective device and an inspection of its console defines the stop condition in detail.

Process: This indicator shows that an error has occurred within the CPU. All operations are halted immediately and usually the attention of a Customer Engineer is required. However, operations can be resumed by first pressing the System Reset key and then loading the program again.

Power (2020 Submodels 1, 2, 3, and 4 only): This indicator shows that a power failure has occurred either in the CPU or in an I/O device such as the IBM 2415 Magnetic Tape Unit. A power failure is defined as the loss of any of the voltages that are generated by the power supply or as an overheating condition in any of the gates in the CPU or I/O devices. Restore power by first pressing the System Reset key and then the Power On key.

Power (2020 Submodel 5 only): This indicator shows that a power failure has occurred either in the CPU or in an I/O device such as the 2415 Magnetic Tape Unit. A power failure is defined as a loss of any of the voltages that are generated by the power supply. Restore power by first pressing the System Reset key and then the Power On key.

Line Failure (2020 Submodel 5 only): This indicator shows that the input voltage to the CPU from the wall outlet has failed. The failure may be either too low a voltage or a missing phase. Restore power by first pressing the System Reset key and then the Power On key.

Thermal (2020 Submodel 5 only): This indicator shows that a gate in the CPU has overheated. Restore power by first pressing the System Reset key and then the Power On key.

Card I/O 1: This indicator shows an unusual condition in the IBM 2501 Card Reader. These conditions can be a filled stacker, an empty hopper, an open machine cover, a read check, a feed check, or a machine check. Machine checks usually require action from a Customer Engineer.

Card I/O 2: This indicator shows an unusual condition in either the IBM 2560 Multi-Function Card Machine or in the IBM 2520 Reader/Punch (whichever is attached). The condition can be filled stacker, empty hopper, open machine cover, feed check, read check, punch check as well as machine check.

Card I/O 3: This indicator relates to the IBM 1442 Card Punch, Model 5. Conditions such as feed check, empty hopper, full stacker etc. cause this indicator to light.

Printer: This indicator shows an unusual condition in either the IBM 2203 Printer or the IBM 1403 Printer. These conditions can be mis-firing of hammers, sync check, or forms check.

SIOC: This indicator shows that either the serial I/O channel or the device that operates through this channel such as the IBM 1419 Magnetic Character Reader has an error condition. For example, a jam or sort check in the 1419 causes the SIOC-light to come on.

IOC: This indicator shows that either the input/output channel or the device that operates through this channel has an error condition. For example, a programing error that involves the IBM 2415 Magnetic Tape Units, or an interface parity error causes the IOC-light to come on.

ST CTRL: This indicator shows that either the storage control feature or the IBM 2311 Disk Storage Drive has an error condition.

TW: This indicator shows an unusual condition in the IBM 2152 Printer-Keyboard. These conditions can be an end-of-forms condition, a P1 check or a P2 check, a 2152 power supply failure, or disconnected cables.

Emergency Off Switch

In an emergency, this switch may be pulled to immediately disconnect all power from the system. This switch must be manually reset by a Customer Engineer before power may be restored to the system.

Control Switches

Mode Switch

Process: In the Process position, the CPU operates under control of the stored program. Further details are provided in the section on Operating Conditions.

Address Stop: The CPU stops when the program has reached the instruction which is located at the main storage address indicated on the Address/Register Data switches, otherwise the same as process mode. When an interrupt occurs and the stop address set up on the Address/Register/Data switches is identical with the instruction address contained in the new PSW the CPU does not stop on this address.

Instruction Step: In this mode of operation, the CPU executes one complete instruction for each operation of the Start key. Since the interrupt condition is always tested prior to the execution of an instruction, it is possible that an interrupt occurs when the Start key is pressed; in this case, the CPU stops after the instruction that is designated by the address in the new PSW is executed.

Storage Display: The eight-bit byte of data located in main storage at the address indicated on the Address/Register Data switches is displayed in the U-L register when the Start key is operated. The address of this byte of data is displayed in the E-S-T-R registers.

Storage Alter: To alter the contents of a main-storage byte, the operator must set the Address/ Register Data switches to the desired location and select the bit configuration of the eight-bit byte of data to be entered by setting the two Data switches. When the Start key is operated, the byte indicated on the Data switches is entered. The byte which was entered is displayed in the U-L registers and the address of this byte is displayed in the E-S-T-R registers.

The eight-bit byte of data is represented in hexadecimal notation on the Data Switches. Data switch 1 selects the configuration of bits 0 1 2 3 and Data switch 2 selects the configuration of bits 4 5 6 7.

Register Display: The contents of the eight general registers and also the contents of certain address registers may be displayed in this mode of operation. The register to be displayed must first be selected by Data switch 1 according to Table 8.

The operator may then display the contents of the selected register in the E-S-T-R registers by operating the Start key. The number of the register selected is displayed as a four-bit binary number in the P register.

The first halfword of the last instruction executed (Op Code plus byte two of the instruction) may be displayed in the E-S-T-R registers in this mode when Register 0 is selected (Table 8).

Register Alter: The contents of the eight general registers and certain address registers may be altered in this mode of operation. The register to be altered must first be selected by Data switch 1 as shown in Table 8. The data to be entered into the selected register is set on the four Address/ Register Data switches. When the Start key is operated, the data is entered into the selected register. The data which was entered is displayed in the E-S-T-R registers and the number of the se-

Table 8. Data Switch 1 01949

Data Switch 1	Register Selected
0 (0000)	First two bytes of previous instruction
1 (0001)	I-Recall Address Register
2 (0010)	PSW (bits 0-15)
3 (0011)	PSW (bits 16-31)
4 (0100)	- -
5 (0101)	- -
6 (0110)	- -
7 (0111)	- -
8 (1000)	General Register 8
9 (1001)	General Register 9
A (1010)	General Register 10
B (1011)	General Register 11
C (1100)	General Register 12
D (1101)	General Register 13
E (1110)	General Register 14
F (1111)	General Register 15

lected register is displayed as a four-bit binary number in the P register.

Storage Scan: In this mode of operation, the CPU starts at the address indicated on the Address/Register Data switches when the Start key is operated. It scans through main storage (from low to high) until a parity error is detected or the Stop key is operated. If the end of main storage is reached during the scan, the scan continues from the beginning of main storage.

NOTE: The only correct method of terminating a scan operation is to press the Stop key.

Storage Fill: In this mode of operation, the CPU enters the contents of Data switches 1 and 2 into all positions of main storage, starting at the address indicated by the Address/Register Data switches. Operation of the Start key causes the loading of storage to begin. The loading proceeds from low address locations to high address locations, and wraps around from the high end of main storage to the low end of main storage. The loading operation is terminated by operation of the Stop key.

NOTE: The only correct method of terminating a fill operation is to press the Stop key.

CPL: The control program load (CPL) position is used in conjunction with the Program Load key for loading the control program into the control storage. (This position in the Mode switch is only effective in 2020 Submodels 5).

Data Switches

Data switches 1 and 2 are physically located on the CPU console below the U and L registers. They are 16-position rotary switches with hexadecimal notation. Data switch 1 selects the configuration of bits 0 1 2 3 and Data switch 2 selects the configuration of bits 4 5 6 7 in an eight-bit byte of data in Storage Alter mode. Data switch 1 is also used for register selection in the Register Display and Register Alter modes. The hexadecimal representation for all bit combinations of an eight-bit byte is shown in Appendix E.

The position of the Data switches may be changed without disturbing CPU operation.

Address/Register Data Switches

The four 16-position rotary Address/Register Data switches are numbered 1, 2, 3, 4 from left to right and are physically arranged on the CPU console directly below the E-S-T-R registers. These switches have hexadecimal notation. A hexadecimal-decimal conversion table is presented in Appendix E. Refer to the Address Stop, Register Display, and Register Alter modes for the functions of the Address/Register Data switches.

The position of the Address/Register Data switches may be changed without disturbing CPU operation, except that a stop may occur if in the Address Stop mode.

Time-Sharing Switch

When this switch is on, the execution of input/output operations is time-shared with other CPU operations. When the switch is off, each input/output operation is completely executed before the CPU continues with the next sequential instruction.

Lamp Test Switch

This switch may be employed to visually check for burned out indicator lamps. All indicators should be on when the Lamp Test switch is on, except the indicators on the IBM 1403 Printer.

Operating Conditions

During normal operation of the CPU under control of the stored program, the Stop indicator on the CPU console is turned off. This indicator is on for all stop conditions of the CPU. Three types of stop conditions may occur.

1. Process check — A process check indicates some

fault in the CPU such as an even parity. It can be reset by operating the System Reset key.

2. A normal stop – In this case, only the Stop indicator is on.
3. A programming error stop – In this case, the Stop indicator as well as an error number display in data register I is on. The error number in data register I is displayed in the binary coded form. For example, the code 0111 represents the number 7 and indicates that the second operand of a decimal instruction is equal to or greater in size (field length) than the first operand.

For normal stop conditions other than a programmed halt operation, the operation code of the Next Sequential Instruction (NSI) is displayed in the U-L registers and the address of the NSI is displayed in the E-S-T-R registers on the CPU console.

For a programmed halt operation, the halt operation code is displayed in the U-L registers and the direct or effective address derived from the B1 and D1 fields of the instruction is displayed in the E-S-T-R registers.

For programming error stop conditions, the operation code of the instruction in which the error condition occurred is displayed in the U-L registers, and the address of this instruction is displayed in the E-S-T-R registers.

Normal Stop Conditions

A normal stop of the CPU occurs as a result of:

1. Operation of the Stop key.
2. A programmed halt operation.
3. An address stop in the Address Stop mode.
4. Operation in the Instruction Step mode.

The CPU is also in a normal stop condition following operation of the System Reset key or when power is first applied, but all register displays are blank (0000).

The CPU starts with the instruction located at the address contained in the program status word (bits 16-31) when the Start key is operated. The Program Status Word (PSW) contains the address of the NSI when the CPU stops as a result of one of the conditions listed above.

All input/output operations in progress when the CPU stops as a result of the conditions listed above are completely executed. All input/output operations in progress are terminated when the System Reset key is operated.

Programming Error Stop Conditions

A programming error stop occurs as a result of the conditions listed below. With each condition is a four-bit binary number which is displayed in the I register when a stop due to the error condition occurs.

1. Operation Code Invalid (0001)
2. Addressing Error
   a. An instruction address or an operand address refers to the protected first 144 bytes of main storage (0100).
   b. An instruction address or an operand address is outside available storage (0101).
   c. The R1 or R2 fields of an RR or an RX format binary instruction contain binary values 0 through 7, the R1 field in a Branch and Store instruction contains binary values 0 through 7, or the R2 field of an RR format branch instruction contains binary values 1 through 7 (0101).
   d. In 2020 Submodels 1 through 4, any instruction part is located at the last available storage position (0101).
3. Specification Error
   a. An instruction address is not located on a half-word boundary of main storage (0110).
   b. A binary operand is not located on a specified boundary of main storage (0110).
   c. For Decimal Add, Decimal Subtract, Zero and Add, and Decimal Compare instructions, the length code L2 is greater than the length code L1 (0110).
   d. For Decimal Multiply and Decimal Divide instructions, the length code L2 is greater than 7 or greater than or equal to the length code L1 (0110).
   e. Bits 12 through 15 of an RX format instruction are not all zero (0110).
   f. A 2560 Write Card instruction is encountered in the program and there is no card in the punch or print station or no print head has been selected (0110).
   g. The field length specified in an input/output instruction is zero or is greater than the maximum allowable number for the I/O device addressed (0110).
4. Binary Overflow Check (1000)
5. Data Error
   a. A sign or digit code of an operand in the decimal instructions Zero and Add, Add, Subtract, Compare, Multiply, or Divide

is incorrect or the operand fields in these operations overlap incorrectly (0111).

b. The multiplicand field (first operand) in a Decimal Multiply instruction has insufficient high-order zeros (0111).

c. An invalid digit code is contained within the second operand field of an Edit operation (0111).

6. Decimal Divide Check (1011)

If the Start key is operated before the error condition is corrected, the CPU again attempts to execute the instruction in which the error occurred.

If the error is corrected without altering the contents of the program status word, the CPU executes the instruction in which the error occurred and continues with the program when the Mode switch is placed in the Process or Address Stop position and the Start key is operated. Caution must be exercised in the case of error conditions that occur during the actual processing of data. A second attempt to execute an instruction which has previously been partially executed without reconstructing the first operand causes erroneous results.

All input/output operations in progress when the central processing unit stops as a result of a programming error condition are completely executed.

If an address which is outside available storage is encountered during an input/output data transfer, the transfer and the input/output operation are terminated as if the complete data field as specified in the input/output instruction had been transferred; an error condition does not occur.

Process Error Stop Conditions

When an internal parity error occurs in the central processing unit, the CPU stops immediately. All input/output operations in progress are terminated.

The Process indicator on the CPU console will be on. The process error condition must be reset by operation of the System Reset key on the CPU console.

After a process error, the restart procedure cannot be used. The program must be loaded again and started either from the beginning or at a check point.

INPUT/OUTPUT OPERATIONS
(except IOC and Storage Control)

Transfers of information to main storage from sources external to the CPU and from main storage to external destinations are referred to as input/output operations. For all card I/O devices, magnetic character readers, communication adapters, and printers, three types of instructions are provided: Transfer I/O, Control I/O, and Test I/O and Branch. The instructions for the Input/Output Channel (IOC) and the Storage Control feature are described in separate sections of this manual.

A Transfer instruction (XIO) controls the transfer of data between main storage and the input/output device.

A Control instruction (CIO) directs an input/output device to perform a specified function; e.g., select a stacker pocket or initiate a carriage skip.

A Test I/O and Branch instruction (TIOB) causes an inquiry to an input/output device for a particular condition (e.g., reader busy or end of form); if the tested indicator is on, the program branches to the specified address.

If the Time-Sharing switch is on, processing operations in the CPU are time-shared with the transfer of data between main storage and the input/output devices. When an input/output device requests service, processing is suspended only for the time required to send or accept the input/output data.

Data Format

Input/output data is located in eight-bit bytes in main storage in variable-length fields. Input/output data may be in the zoned, binary, or packed format. The data format in which an input/output device may accept or present data is, however, a characteristic of the device.

Input data is translated from the code form of the input device to the Extended-Binary Coded Decimal Interchange Code employed by the CPU internally as the data is received. Output data is translated from the internal CPU code to the code form of the output device as the data is transferred.

Condition Code

The status of an I/O device addressed by a Transfer I/O instruction, and under certain conditions, a Control I/O instruction, is used to set the condition code of the PSW at the time the execution of the instruction is completed. The condition code indicates whether or not the I/O device has initiated the operation specified, and if not, the reason for the rejection. The condition code can be used for decision making by subsequent branching operations. The condition code is set to 00, 01, 10, or 11 by an instruction to indicate the status of the I/O device addressed.

Condition Code	Status
00	Available
01	Working
10	Channel Busy
11	Not Operational

Available: Indicates that the addressed I/O device is operational, does not contain data or error check conditions, and is not busy with a previously initiated operation.

Working: Indicates that the addressed I/O device is executing a previously initiated operation.

Channel Busy: This indication is given only in response to an attempt to start a burst-mode operation (IOC, Storage Control, and BSCA High Speed Feature) while a time shared input/output or processing operation was in progress and overlap mode was not set. If overlap mode was set, the indication would pertain only to the channel or to the storage control.

Not Operational: Indicates that the addressed I/O device is in a not ready status, or an error or a data check condition exists on the device.

The operation specified by an XIO or CIO instruction is initiated only when the addressed I/O device is in the available state. If an I/O device is specified which is not a part of the system, a no-operation occurs, and the condition code is not changed.

Instruction Format

The three input/output instructions and their formats are as follows:

Test I/O and Branch

TIOB D1 (B1), UF

9A	D.A.	F.S.	B1	D1
0 7	8 11	12 15	16 19	20 31

01950

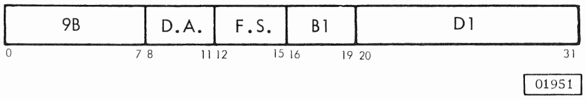
The Device Address (DA) specifies the I/O device in which a condition is to be tested.

The Function Specification (FS) specifies the particular condition or indicator to be tested in the I/O device addressed.

If the condition tested in the addressed I/O device is on, the updated instruction address is replaced by the branch address derived from the B1-D1 fields; otherwise, normal instruction sequencing continues with the updated instruction address.

Control I/O

CIO B1 (D1), UF



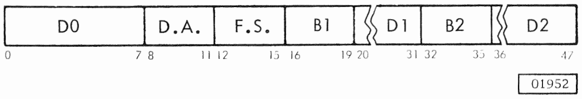
The Device Address (DA) specifies the I/O device in which a control function is to be performed.

The Function Specification (FS) specifies the particular component (it may also specify the primary function of that component) in the I/O device addressed.

A detailed specification of the control function to be performed is derived from the contents of the B1-D1 fields, according to the rules for direct or effective address generation. If the detailed specification derived from the B1-D1 field is all zero, a no-operation occurs.

Transfer I/O

XIO (D1 (UF, B1), D2(B2)



The Device Address (DA) specifies the I/O device to which output data is to be transmitted, or from which input data is to be received.

The Function Specification (FS) specifies the input or output function to be performed on the I/O device addressed, and also the particular component of the addressed device (when required).

The main-storage location of the first byte in the input or output data field is derived from the contents of the B1 - D1 fields according to the rules for direct or effective address generation.

The field or record length of the input or output data in main storage is derived from the contents of the B2 - D2 fields.

The field length specifications for input or output data fields in main storage is the actual number of bytes in the field. Whereas for variable field length processing operations, the field length specification is the number of bytes extending beyond the first byte.

Error Conditions

Error conditions which may occur in the CPU during the instruction or service phases of input/output operations are listed below. Error conditions which may occur in I/O devices are included in the IBM Systems Reference Library pub lication for the respective device.

1. Operation Code Invalid
2. Addressing Error
   a. An instruction address or the address of an input/output data field refers to the protected first 144 bytes of main storage.
   b. An instruction address, the address of an input/output data field, or a branch address is outside available storage. If an incorrect branch address has been specified, 2020 Submodels 1 through 4 indicate the error during the execution of the branch instruction itself; 2020 Submodel 5, however, does not indicate the error until an attempt has been made to execute the instruction located at the branch address.
   c. Any instruction part is located in the last two main-storage positions (2020 Submodels 1 through 4 only).
3. Specification Error
   a. The low-order bit of an instruction address is one.
   b. A 2560 Write Card instruction is encountered in the program and there is no card in the punch or print station, or when no print head has been selected.
   c. The field length specified for an input/output data field is zero or is greater than the maximum allowable number for the I/O device addressed.
4. CPU Parity Error

An input/output instruction containing an I/O device address which specifies a device not attached to the system is treated as a no-operation.

To ensure correct function, only valid specifications as defined in this manual should be used.

Interruption

Interruption is the general term applied to an automatic branch in the CPU program. The branch is automatic in the sense that it occurs when the condition exists, and is independent of a programmed branch instruction.

In the System/360 Model 20, interruption is provided only for the channel end input/output condition. Channel end condition of an input/output device is defined as the time (in the mechanical cycle of the device) at which the data transfer has been completed.

In the time-shared mode of operation of the CPU and the input/output devices, the transfer of data between the input/output devices and main storage in the CPU is asynchronous with respect to processing operations. Thus, the channel end condition of an input/output data transfer operation may occur at any time in the instruction or execution phase of any processing operation. It is also possible that more than one input/output device would reach the channel end condition during a specific processing operation. The channel end conditions are stored in the form of interrupt bits. These bits are reset when the respective interrupts occur or are reset by a system reset or load operation.

The CPU is in an interruptible state when the channel mask bit in the Program Status Word (PSW) is 1, and is in a non-interruptible state when this bit is 0. The channel mask bit is reset to 0 by operation of the System Reset or Program Load key. The channel mask bit may be altered by a Set PSW instruction or by an interruption.

When the CPU is in the interruptible state (channel mask bit is 1), a test is performed by the CPU at the end of each processing operation to determine whether or not a channel end condition exists for any input/output device. (The instruction and start phases of input/output operations are also considered as processing operations.)

The test for a channel end condition is performed in an established priority sequence. The program continues with the next sequential instruction if no channel end conditions exist. When one or more end conditions exist, the first one encountered in the priority sequence causes an interruption to occur. The channel end condition which causes the interruption is reset. The interruption is performed by storing the PSW in fixed main-storage location 144 and obtaining a new PSW from another fixed main-storage location (148) before the program continues. Since the new PSW contains the address of the next sequential instruction, the interrupt is equivalent to a branch operation. The PSW which is stored in main storage location 144 is referred to as the old PSW.

The old PSW (stored at main storage location 144 when an interrupt occurs) contains the device address of the I/O device which caused the interruption (bits 8-11); the primary function which the device was performing (bits 12-15); the address of the next sequential instruction, and the condition code. Primary functions are read, punch, or print; not included are details such as which feed was in use for a read operation on the IBM 2560 Multi-Function Card Machine (MFCM).

The following is a list of device addresses and function specifications contained in bits 8 to 15 of the old PSW stored at main-storage location 144 when an interruption due to the corresponding channel end condition occurs. The list is in the priority sequence for these interruptions.

Channel End Condition	DA	FS
1419 or 1259 Read	6	2
2501 Read	1	2
2520 or 2560 Read	2	2
1403 or 2203 Print	4	0
CA Transmit or Receive	5	6
BSCA Any XIO	5	6
BSCA ITB Received	5	2
2560 Punch	2	4
1442 Punch	3	4
2560 Card Print	2	0
2520 Punch	2	4
Input/Output Channel	7	0
Storage Control	8	0
2152 Read	E	1
2152 Write or Carrier Return	E	2
2152 Inquiry Request	E	3
CC2 (only if non-overlap mode)	F	unspecified

The channel mask bit in the new PSW (obtained from main-storage location 148) may be used to disable further interruptions in the routine which begins at the next sequential instruction address specified in the new PSW. As a means of returning to the point in the program at which the interruption occurred, a Set PSW instruction, in which the specified address is 144, may be used as the last instruction in the routine.

CC2 Interrupt (2020 Submodels 2 and 4 only)

High-speed I/O devices that operate in burst mode only (such as the IBM 2415 Magnetic Tape Unit and the IBM 2311 Disk Storage Drive), cannot operate simultaneously with the other I/O devices that operate in time-sharing mode (such as card machines, communications adapter). However, if the program issues an instruction that addresses a burst-mode device while time-shared operations are in progress, the condition code (CC) is set to 2 (binary 10) and the program continues with the next sequential instruction. Due to this particular condition code, a special CC2 interrupt occurs when all time-shared operations (including stacker selection) are completed, provided that the channel mask bit in the PSW is present and no interrupt request with a higher priority is pending. Because the CC2 interrupt has the lowest priority, the program can branch back to start the previously-ignored burst-mode operation when the CC2 interrupt occurs; this is the earliest moment possible at which the CPU is available for burst-mode operations.

The CC2 interrupt is not requested when overlap mode is set in a 2020 Submodel 5.

Last Card Control

Each card I/O device which has a card reading unit is provided with a testable Last Card indicator.

The last card condition in the 2501 Card Reader is set when a Read Card instruction for the 2501 is encountered in the program and there is no card at the read station. The channel end condition is set and the 2501 is placed in a not-ready status. The Last Card indicator may be tested and is reset by a Test I/O and Branch instruction in which the device address is 1 and the function specification is 4. The Last Card indicator is also reset when a new deck of cards is run in on the 2501 and the first Read Card instruction is encountered in the program; or it can be reset by the system reset function of the CPU.

The Last Card indicator in the IBM 2520 Card Read Punch is set when a Read Card instruction for the 2520 is encountered in the program and there is no card at the pre-read station. The channel end condition is set immediately and a card at the punch station is ejected to the stacker. The 2520 becomes not ready when both the pre-read and the pre-punch stations are empty while a Transfer I/O instruction is in progress. The indicator may be tested, and is reset by a Test I/O and Branch instruction in which the device address is 2 and the function specification is 4. The Last Card indicator is also reset when a new deck of cards is run in on the 2520 and the first Read Card instruction is encountered in the program; or it can be reset by the system reset function of the CPU.

The Last Card indicator in the 2560 MFCM is set when a Read Card instruction for the 2560 is encountered in the program and there is no card at the read station of the specified feed. The channel end condition is set immediately and cards remaining in the specified feed are advanced one station. The feed which is empty remains in a ready status in order that a Write Card instruction may be executed for the last card and in order that a subsequent Read Card instruction may be employed to advance the last card to the stacker. The Last Card indicator may be tested and is reset by a Test I/O and Branch instruction in which the device address is 2 and the function specification is 4. The Last Card indicator is also reset when a new deck of cards is run in on the empty feed of the 2560 and the first Read Card instruction for that feed is encountered in the system program; or it can be reset by the system reset function of the CPU.

COMPATIBILITY

The organization of the System/360 Model 20 is in most respects identical with that of other System/360 models. The data and instruction formats are a compatible subset. The majority of operations in the System/360 Model 20 instruction set are compatible. There are instructions and features which are different in the System/360 Model 20 from those in System/360s. These differences deserve programming consideration and are described in this section.

The System/360 Model 20 CPU exerts direct control over all the I/O devices attached to it. However, I/O operations are initiated, halted, or tested by program instructions, which select the unit to be used. The I/O instructions determine what operation is performed (read, write, etc.) and where the data is stored.

Operation Codes

The operation codes of all System/360 Model 20 instructions other than Halt and Proceed, Set PSW, Branch and Store, and Input/Output controls are the same as those assigned in other System/360 models. The operation codes of these instructions are unassigned in other models.

General Registers

The System/360 Model 20 has eight general registers. They are numbered 8-15 and correspond to the same registers in other System/360 models. They are one halfword in length compared to full word length on other System/360 models. Results of valid System/360 Model 20 binary operations are identical to corresponding operations in other System/360 models except for the recognition of an overflow condition.

Main Storage Addressing

The first 144 bytes of main storage are protected and program reference to this area results in an error condition.

Addressing

An address used to refer to main storage in the System/360 Model 20 may be specified by either of two methods; direct addressing or effective-address generation. Direct addressing is indicated when the high-order bit in the B field of an instruction is zero. The address from this method is derived from the binary value of the low-order 14 positions (for addresses up to 16383) of the combined B and D fields. Effective-address generation is automatically performed when the B field refers to general registers 8-15; in forming this address, the System/360 Model 20 operates identically to other System/360 models.

IBM 1442 CARD PUNCH, MODEL 5

Introduction

The IBM 1442 Card Punch, Model 5 provides punched card output for the System/360 Model 20. The 1442-5 consists of a card hopper, a serial punch station, and one radial stacker. Card punching is done serially at a maximum rate of 160 columns/second. The card punching rate depends upon the number of columns specified in the Punch instruction. Punching speed may range from 265 cards/minute for punching columns 1-10, to 91 cards/minute for punching columns 1-80.

The 1442-5 has a 1200 card hopper capacity and a 1300 card stacker capacity.

1442 Card Punch Instructions

Transfer Instructions (XIO)

Function	Op Code	DA	FS
Punch and Feed	D0	3	6

The direct or effective address (derived from the B1-D1 fields of the instruction) specifies the left-most byte of the output data field in main storage. The length of the output field is derived from the B2-D2 fields of the instruction.

The field length specification is restricted to binary values equal to or less than 80.

Upon completion of a Punch and Feed instruction, the card which was punched is ejected to the stacker and the next card is fed from the hopper to the punch station.

Test I/O and Branch Instructions (TIOB)

Function	Op Code	DA	FS
Test Punch Busy	9A	3	2
Test Punch Error	9A	3	3
Test Feed Error	9A	3	5

Test Punch Busy: The Punch Busy indicator is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Punch Error: The Punch Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The Punch Check indicator is reset by the branch test.

Test Feed Error: The Feed Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

If a feed cycle is in progress when this instruction is encountered in the program, the CPU is interlocked until the cycle is completed, at which time the test is performed.

Condition Code

The condition code is set to 00, 01, or 11 at the time the execution of an XIO instruction is completed to indicate that the 1442 Card Punch is available, working, or not operational.

The data transfer as specified in an XIO instruction for the 1442 is initiated only when the 1442 is in the available state.

The 1442 is available when the Ready indicator is on, the Punch Check indicator is not on, and the data transfer portion of a 2560 MFCM punch or punch and feed operation is not in progress.

The 1442 is working or busy during the execution of a previous 1442 punch operation or during the data transfer portion of a 2560 MFCM punch or punch and feed operation.

The 1442 is not operational when the Ready indicator is off or when the Punch Check or Feed Check indicators are on.

Control I/O Instructions (CIO)

A Control I/O instruction issued to the 1442 results in a no-operation.

IBM 2501 CARD READER, MODELS A1 and A2

Introduction

The 2501 Card Reader consists of one hopper, a serial read station, and one stacker.

The 2501 Card Reader, Model A1 has a maximum rate of 600 cards per minute, and the 2501 Card Reader, Model A2 has a maximum rate of 1,000 cards per minute.

2501 Card Reader Instructions

Transfer Instructions (XIO)

Function	Op Code	DA	FS
Read Card	D0	1	2

The direct or effective address (derived from the B1-D1 fields of the instruction) specifies the leftmost main-storage location of the field where the input data will be located. The length of the input field is derived from the B2-D2 fields of the instruction.

The field length specification is restricted to binary values equal to or less than 80.

Test I/O and Branch Instructions (TIOB)

Function	Op Code	DA	FS
Test Reader Busy	9A	1	0
Test Reader Error	9A	1	1
Test Last Card	9A	1	4

Test Reader Busy: The Reader Busy indicator is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Reader Error: The Read Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The Read Check indicator is reset by the branch test.

Test Last Card: The Last Card condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The Last Card condition is reset by the branch test.

Control I/O Instructions (CIO)

A control instruction which specifies the 2501 is treated as a no-operation.

Condition Code

The condition code is set 00, 01, or 11 when the execution of an XIO instruction is initiated. At this time the status of the respective I/O device is checked and the result of the check determines the setting of the condition code; 00 indicates available status, 01 indicates working status, and 11 indicates a not operational status.

A data transfer specified by an XIO instruction for the 2501 Card Reader is initiated only when the 2501 is in the available state. The 2501 is in the available state when the ready indicator is on, the read check indicator is not on, and a previously initiated Read Card instruction is not in progress.

The 2501 is working (condition code 01) during the execution of a previously initiated Read Card XIO instruction. It cannot accept a new instruction until the previous instruction has been executed and the 2501 has been returned to the available state (condition code 00). The 2501 is considered to be in the working state during the data transfer portion of a Card Read instruction on the 2560 MFCM or the 2520 Card Read Punch.

The 2501 is not operational when the ready indicator is off, or when the feed check or read check indicators are on.

IBM 2520 CARD READ PUNCH and CARD PUNCH

Introduction

The 2520 Card Read Punch, Model A1 reads cards serially at a maximum speed of 500 cards per minute and punches parallel at the same rate. Reading and punching may be overlapped to provide reading, computing, and punching at the maximum throughput speed. Reading without punching and punching without reading are also possible with the 2520-A1. All 2520s have two stackers. The 2520 Card Punch, Models A2 and A3 consist of the same basic mechanical unit as the 2520-A1, but have no read unit. Maximum punching rates are 500 cards per minute, or 300 cards per minute, depending on the model.

2520 Card Read Punch, Model A1	500 cpm read/punch
2520 Card Punch, Model A2	500 cpm punch
2520 Card Punch, Model A3	300 cpm punch

2520 Card Read Punch Instructions

Transfer Instructions (XIO)

Function	Op Code	DA	FS
Read Card	D0	2	2
Punch Card	D0	2	4
Punch and Feed	D0	2	6

Read Card: The direct or effective address, derived from the B1-D1 fields of the instruction, specifies the leftmost core storage location of the field where the input data will be stored. The length of the input field is derived from the B2-D2 fields of the instruction.

The field length specification is restricted to binary values equal to or less than 80.

The data transfer begins with the data read from column 1 of the card and continues until the specified number of bytes have been transferred from the reader to main storage.

When the input-field length is larger than the available space in main storage, the data transfer ends when the highest available storage position has been reached. No data error occurs when data is "left over."

Punch Card or Punch and Feed: The direct or effective address, derived from the B1-D1 fields of the instruction, specifies the leftmost byte of the output data field in the main storage. The length of the output field is derived from the B2-D2 fields of the instruction.

When the input-field length is larger than the available space in main storage, the data transfer ends when the highest available storage position has been reached. No data error occurs when data is "left over."

The field length specification is restricted to binary values equal to or less than 80.

The Card Read Punch has a parallel punch unit. The data transfer consists of a total of 13 scans of the output field, during which the data is not only transferred to the punch unit, but is also compared with echo data to ensure correct punching.

Upon completion of the Punch Card or Punch and Feed instruction, the card which was punched is ejected to the stacker. For the Punch Card instruction, the card at the read station does not advance to the punch station. For the Punch and Feed instruction, however, the card at the read station advances to the punch station without being read, and the next card is fed from the hopper to the read station.

When a Punch Card or a Punch and Feed instruction, which is preceded by a Punch Card instruction, is encountered in the program, the card at the read station advances to the punch station without being read, and the next card is fed from the hopper to the read station before punching begins.

In order to perform simultaneous reading and punching operations on the Card Read Punch, the Punch Card instruction must occur first in sequence in the program.

Test I/O and Branch Instructions (TIOB)

Function	Op Code	DA	FS
Test Reader Busy	9A	2	0
Test Reader Error	9A	2	1
Test Punch Busy	9A	2	2
Test Punch Error	9A	2	3
Test Last Card	9A	2	4
Test Feed Error	9A	2	5

Test Reader Busy: The Reader Busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Reader Error: The Read Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The Read Check indicator is reset by the branch test.

If a read error occurs while a Read instruction is in progress, the Read Error and the Feed Error indicators are both turned on.

Test Punch Busy: The Punch Busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Punch Error: The Punch Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The Punch Check indicator is reset by the branch test.

A card which is punched in error is automatically directed to stacker pocket 2.

Test Last Card: The Last Card condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The Last Card condition is reset by the branch test. When the Last Card indicator is set, the ready status is lost and an interrupt is requested. This action occurs in two cases:

1. The Last Card condition is set when a Read instruction is issued while a card is registered in the prepunch station and the preread station is empty.
2. The Last Card condition is set when a Read instruction is issued while the preread and prepunch stations are empty.

Test Feed Error: The Feed Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

If a feed cycle is in progress when this instruction is encountered in the program, the CPU is interlocked until the cycle is completed, at which time the test is performed.

Error conditions which can occur in the 2520 are read check, punch check, feed check, or machine check. These conditions are indicated individdually on the 2520. The feed check and the machine check conditions turn off the Ready indicator.

The read check or punch check conditions can be reset by the I/O Check Reset key or the Load key on the CPU console or by Test I/O and Branch instructions in the program. The feed check and the machine check condition can be reset only by the Non-Process Runout (NPRO) key on the 2520 after the cards have been removed from the hopper. (A machine check may also require Customer Engineer action.)

The Card I/O No. 2 indicator on the CPU console is turned on when the read check, punch check, feed check or machine check conditions exist on the 2520 or when a Read Card, Punch Card, or Punch and Feed instruction is encountered in the program and the Ready indicator is not on.

NOTE: The feed check condition is set, together with the Attention indicator, when the cover is opened during a 2520 XIO operation or during a card run-in or NPRO cycle. Both conditions can be reset by closing the cover and operating the NPRO key.

Control I/O Instructions (CIO)

Function	Op Code	DA	FS
Stacker Select	9B	2	0

Stacker pocket 2 may be selected for the card in the punch station with this instruction. Stacker pocket 1 is the normal pocket, for which a Stacker Select instruction is not required. The proper location in the program sequence for the Stacker Select instruction is preceding the Read, Punch, or Punch and Feed instruction which causes the card in the punch station to be ejected to the stacker.

The direct or effective detailed specification (derived from the B1-D1 fields of the Stacker Select instruction) is inserted in the stacker control circuits. Only the low-order two bits are involved; the remaining high-order bits are ignored.

Bits		
30	31	Pocket Selected
0	1	1
1	0	2

A stacker select instruction in which bits 30 and 31 are both zero is treated as a no-operation.

Condition Code

The condition code is set to 00, 01, or 11 at the time the execution of a 2520 XIO instruction is completed to indicate that the 2520 is in the available, working, or not operational state.

The I/O operation and the data transfer as specified in a 2520 XIO instruction is initiated only when the 2520 is in the available state.

Available: The 2520 is available for a Read Card XIO instruction when:

1. The Ready indicator is on.
2. The Read Check and Punch Check Indicators are not on.

3. A previous 2520 Read or Punch and Feed instruction is not in progress.
4. The data transfer portion of a 2501 Read instruction is not in progress.

The 2520 is available for a Punch or Punch and Feed XIO instruction when:

1. The Ready indicator is on.
2. The Read Check and Punch Check indicators are not on.
3. A previous 2520 operation is not in progress.

Working: The 2520 is working or busy when a Read Card XIO instruction for the 2520 is encountered in the program if:

1. A previous 2520 Read or Punch and Feed instruction is in progress.
2. The data transfer portion of a 2501 Read instruction is in progress.

The 2520 is working or busy when a Punch or Punch and Feed instruction for the 2520 is encountered in the program if a previous 2520 operation is in progress.

Not Operational: The 2520 is not operational when the Ready indicator is off or when the Read Check, Feed Check, or Punch Check indicators are on.

2520 Card Punch Instructions (2520-A2 and A3)

Transfer Instructions (XIO)

Function	Op Code	DA	FS
Punch and Feed	D0	2	6

The direct or effective address (derived from the B1-D1 fields of the instruction) specifies the leftmost byte of the output data field in main storage. The length of the output field is derived from the B2-D2 fields of the instruction.

The field length specification is restricted to binary values equal to or less than 80.

The Card Punch has a parallel punch unit. The data transfer consists of a total of 13 scans of the output field, during which the data is not only transferred to the punch unit but is also compared with echo data to ensure correct punching.

Upon completion of a punch card operation, the card which was punched is ejected to the stacker and the next card is fed from the hopper to the punch station.

When the input field length is larger than the available space in main storage, the data transfer ends when the highest available storage position has been reached. No data error occurs when data is "left over."

Test I/O and Branch Instructions (TIOB)

Function	Op Code	DA	FS
Test Punch Busy	9A	2	2
Test Punch Error	9A	2	3
Test Feed Error	9A	2	5

Test Punch Busy: The punch busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address: otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Punch Error: The Punch Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The Punch Check indicator is reset by the branch test. A card which is punched in error is automatically directed to stacker pocket 2.

Test Feed Error: The Feed Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

If a feed cycle is in progress when this instruction is encountered in the program, the CPU is interlocked until the cycle is completed, at which time the test is performed.

Control I/O Instructions (CIO)

Function	Op Code	DA	FS
Stacker Select	9B	2	0

Stacker pocket 2 may be selected for the card in the punch station with this instruction. Stacker pocket 1 is the normal pocket, for which a Stacker Select instruction is not required. The proper location in the program sequence for the Stacker Select instruction is preceding the Punch and Feed instruction. This causes the card in the punch station to be ejected to the selected stacker.

The direct or effective detail specification (derived from the B1-D1 fields of the Stacker Select instruction) is inserted in the stacker control circuits.

Only the two low-order bits are involved; the remaining high-order bits are ignored.

Bits 30	Bits 31	Pockets Selected
0	1	1
1	0	2

A stacker select instruction in which bits 30 and 31 are both zero is treated as a no-operation.

Condition Code

The condition code is set to 00, 01, or 11 at the time the execution of a 2520 Card Punch XIO instruction is completed to indicate that the 2520 is in the available, working, or not operational state.

The I/O operation and the data transfer as specified in a 2520 XIO instruction is initiated only when the 2520 is in the available state.

Available: The 2520 is available for a Punch or Punch and Feed XIO instruction when:

1. The Ready indicator is on.
2. Punch Check indicator is not on.
3. A previous 2520 operation is not in progress.

Working: The 2520 is working or busy when a Punch, or Punch and Feed instruction for the 2520 is encountered in the program if a previous 2520 operation is in progress.

Not Operational: The 2520 is not operational when the Ready indicator is off or when the Punch Check or Feed Check indicators are on.

IBM 2560 MULTI-FUNCTION CARD MACHINE, MODELS A1 AND A2

Introduction

The IBM 2560 Multi-Function Card Machine (MFCM) is a versatile input/output unit providing full card-file maintenance abilities. 2560 MFCM Model A1 can be attached to 2020 Submodels 1, 2, and 5 and 2560 Model A2 can be attached to 2020 Submodels 3 and 4. An optional six-line card document printing feature is available for the 2560 Model A1 only. Cards are fed from either of two hoppers through a serial read station and a serial punch station; in the 2560 Model A1, they then pass through the serial print station. Finally, the cards can be directed to any of the stackers; the 2560 Model A1 has five stackers and the 2560 Model A2 has four stackers.

In the 2560 Model A1, the reading speed is 500 cards per minute, punching speed 160 columns per second, and card printing 140 characters per second. In the 2560 Model A2, the reading speed is 310 cards per minute, and the punching speed 120 columns per second.

Unit record functions such as Reproduce, Gang Punch, Summary Punch, Collate and Decollate can all be performed on the 2560. The optional Card Print feature in the 2560 Model A1 may consist of two, four, or six print heads which can be positioned by the operator to print on any of the 25 lines of a card. Up to 64 characters may be printed on each line.

2560 Multi-Function Card Machine Instructions

Transfer Instructions (XIO)

Function	Op Code	DA	FS
Read Primary Card	D0	2	2
Read Secondary Card	D0	2	3
Punch Primary Card	D0	2	4
Punch Secondary Card	D0	2	5
Punch and Feed Primary Card	D0	2	6
Punch and Feed Secondary Card	D0	2	7
Write Card (2560 Model A1 only)	D0	2	0

Read Card: The direct or effective address (derived from the B1-D1 fields of the instruction) specifies the leftmost main-storage location of the field where the input data will be located. The length of the input field is derived from the B2-D2 fields of the instruction.

The field length specification is restricted to binary values equal to or less than 80.

The data transfer begins with the data read from column 1 of the card and continues until the specified number of bytes have been transferred from the reader to main storage.

Punch Card or Punch and Feed: The direct or effective address (derived from the B1-D1 fields of the instruction) defines the leftmost main-storage location of the output data. The length of the output field is derived from the B2-D2 fields of the instruction.

The field length specification is restricted to binary values equal to or less than 80.

A punch operation always begins at column 1 of the card from the specified feed, and continues until the specified number of columns have been punched. The number of columns punched corresponds to the number of bytes transferred from main storage.

Upon completion of a Punch and Feed instruction, all cards in the specified feed advance one station. The card which advances through the read station is not read.

Write Card: The optional Card Print feature, available in the 2560 Model A1 only, may consist of two, four, or six print heads which may be positioned individually by the operator to print on any of the 25 lines of a card. Up to 64 characters may be printed on each line.

The output data which is to be printed on a card is located in adjacent 64-byte blocks of main storage. The number of adjacent 64-byte blocks corresponds to the number of 1s in the detailed specification of the print head select control instruction.

For the instruction which specifies card printing, the direct or effective address derived from the B1-D1 fields defines the leftmost main-storage location of the first 64-byte block of the output data. The first block is printed on the first print head selected, the second block on the next sequential print head selected, etc. The length of the block containing the greatest number of bytes is derived from the B2-D2 fields of the instruction. Each block or line which is shorter than the longest block must contain sufficient blank bytes to the right of the last character to be printed within the block to extend the block to the length of the longest block.

A Write instruction is treated as a no-operation in the 2560 Model A2, and in the Model A1 without the Card Print feature.

The field length specification in the write card instruction is restricted to binary values less than 65.

NOTE: The CPU Stop light may blink on and off during card document-printing operations on the 2560 Model A1. This action does not signify any malfunction.

Test I/O and Branch Instructions (TIOB)

Function	Op Code	DA	FS
Test Reader/Punch Busy	9A	2	0
Test Reader/Punch Error	9A	2	1
Test Card Printer Busy	9A	2	2
Test Last Card	9A	2	4
Test Feed Error	9A	2	5

Test Reader/Punch Busy: The reader or punch busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Reader/Punch Error: The Read Check and the Punch Check indicators are tested. If either indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. Both the Read Check and the Punch Check indicators are reset by the branch test. If a feed error occurs during a Read instruction, the Read Check and the Feed Check indicators are both turned on.

Test Last Card: The last card condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The last card condition is reset by the branch test.

Test Card Printer Busy (2560 Model A1 only): The card printer busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. Test Card Printer Busy results in a no-operation in Model A2.

Test Feed Error: The Feed Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

If a feed cycle or an eject cycle is in progress when this instruction is encountered in the program, the CPU is interlocked until the cycle is completed, at which time the test is performed.

The contents of the 2560 MFCM stacker registers are transferred to main-storage locations 152 through 154 when this instruction is encountered in the program to facilitate program restart procedures.

Control I/O Instructions (CIO)

Function	Op Code	DA	FS
Primary Card Stacker Select	9B	2	0
Secondary Card Stacker Select	9B	2	1
Punch Card Stacker Select	9B	2	2
Print Head Select (2560 Model A1 only)	9B	2	3

Stacker Selection: There are five stackers on the 2560 Model A1 and four stackers on the Model A2. The Stacker Select instructions provide the means for directing cards from the primary feed, secondary feed, and punch station to any of the stackers. Stacker 1 is the normal pocket for cards in the primary feed. The normal pocket for cards in the secondary feed is stacker 5 (Model A1) and stacker 4 (Model A2). Stacker Select instructions are required only when it is desired to direct cards to pockets other than normal. A Stacker Select 5 instruction on the 2560 Model A2 directs the card to stacker 4.

The MFCM stacker controls contain three program addressable stacker registers: the primary feed stacker register, the secondary feed stacker register, and the punch feed stacker register. By use of these registers the destination of a card can be assigned between the time the card has passed through the read station and the time it leaves the punch station.

The direct or effective detailed specification (derived from the B1-D1 fields of the Stacker Select instruction) is inserted in the stacker register specified by the function specification field of the instruction. Only the low-order three bits of the detailed specification are involved; they are the following:

Bits			Stacker Selected	
29	30	31	Model A1	Model A2
0	0	1	1	1
0	1	0	2	2
0	1	1	3	3
1	0	0	4	4
1	0	1	5	4
1	1	0	1	1
1	1	1	1	1

A Stacker Select instruction in which bits 29, 30, and 31 are all zero is treated as a no-operation.

Programming Notes

The proper location in the program sequence to specify a stacker pocket other than normal is after the

Read instruction for that card, if any, but before the next Read, Punch, and Punch and Feed instruction which specifies the same feed. The Stacker Select instruction specifies the destination of a card while the card is in the primary feed, the secondary feed, or the punch station.

The stacker pocket destination for a card from either the primary or secondary feed may be changed by a Punch Card Stacker Select instruction while that card is in the punch unit or requested at the print station.

Print Head Selection: The 2560 Model A1 only has an optional Card Print feature consisting of two, four or six print heads. Any one or any combination of the print heads installed may be selected for a write card operation. This selection is made by a Print Head Select instruction, which must precede the Write Card instruction in the program sequence. If the same combination of lines is to be printed throughout the program, the control instruction must appear only once, before the first Write Card instruction. A Print Head Select instruction is treated as a no-operation in the 2560 Model A2.

The low-order six bits of the detailed specification in the Print Head Select instruction indicate which print heads are selected for a subsequent write card operation as shown below:

Bit Number	26	27	28	29	30	31
Print Head	1	2	3	4	5	6

When a bit corresponding to a print head is 1, the head is selected; when the bit is 0, the head is not selected.

At least one print head must be selected, otherwise the next card print operation results in a programming error stop condition.

Condition Code

The condition code is set to 00, 01, or 11 at the time the execution of a 2560 XIO instruction is completed to indicate that the 2560 is in the available, working, or not operational state.

The I/O operation and data transfer, as specified in a 2560 XIO instruction, is initiated only when the 2560 is in the available state.

Available: The 2560 is available for a Read Card XIO instruction when:

1. The Ready indicator for the specified feed is on.
2. The Read Check and Punch Check indicators are not on.
3. A previous 2560 operation is not in progress.
4. The data transfer portion of a 2501 read card operation is not in progress.

The 2560 is available for a Punch or Punch and Feed XIO instruction when:

1. The Ready indicator for the specified feed is on.
2. The Read Check and Punch Check indicators are not on.
3. A previous 2560 read, punch, or punch and feed operation is not in progress.
4. The data transfer portion of a 1442 punch operation is not in progress.
5. A card is either in the pre-punch or pre-read station of the specified feed.

The 2560 Model A1 with the optional Card Print feature is available for a Write Card XIO instruction when:

1. The Ready indicator for either feed is on.
2. The Read Check and Punch Check indicators are not on.
3. A previous 2560 operation is not in progress.
4. A card is located in the punch or print station.

Working: The 2560 is working or busy when a Read Card XIO instruction for the 2560 is encountered in the program if:

1. A previously issued 2560 operation is in progress.
2. The data transfer portion of a 2501 read card operation is in progress.

The 2560 is working or busy when a Punch or Punch and Feed XIO instruction for the 2560 is encountered in the program if:

1. A previously issued 2560 read, punch, or punch and feed operation is in progress.
2. The data transfer portion of a 1442 punch operation is in progress.

The 2560 Model A1 is working or busy when a Write Card XIO instruction for the 2560 is encountered in the program if a previously issued 2560 operation is in progress.

Not Operational: The 2560 is not operational for a Read, Punch, or Punch and Feed XIO instruction when Ready indicator for the specified feed is not on or when the Read Check, Punch Check, or Feed Check indicators are on.

The 2560 Model A1 is not operational for a Write Card XIO instruction when the Ready indicators for both feeds are off, when the Read Check, Punch Check, or Feed Check indicators are on.

COLUMN BINARY FEATURE (2020 Submodels 2, 4, and 5 only)

The IBM punched card is a permanent data storing device that contains 80 storage positions (card columns). When the standard IBM card code (Figure 6) is used, each punched column contains either one digit or one character. The capacity of a card is increased to 160 storage positions, however, when the column binary code (Figure 6) is used; each punched column can then contain two digits or two characters in six-bit Binary Coded Decimal (BCD) code. A column binary card cannot be read like an ordinary punched card because the meaning of the usual zones and digits (12, 11, 0, 1, 2, etc) is changed to correspond to the bits of the BCD code. The punches in one card column represent the BCD bits B, A, 8, 4, 2, 1, and B, A, 8, 4, 2, 1 from top to bottom.

To enable the card I/O devices that can be attached to the System/360 Model 20 to read column binary cards, the Column Binary feature must be installed. This feature reads each BCD character into one byte of core storage. Since the six-bit BCD character cannot fill a byte, the character is placed into bits 2 - 7 of the byte, and bits 0 and 1 of the byte are forced to zero. Column binary reading is performed from top to bottom of the card column, proceeding by columns from left to right (Figure 7).

The programmer must know that, when the 80 columns of a card punched in Column Binary mode are read, 160 adjacent bytes of core storage are needed. The field length specification derived from the B2 and D2 fields of the XIO instruction (issued to read a column binary card) pertains to the number of BCD characters that are to be read. The field length must therefore be greater than zero and not more than 160.

The programmer can specify an odd number of characters (odd field length). When this occurs, reading always stops after the upper half of the last card column has been read. If the storage capacity is exceeded during a column binary read operation, reading stops when the highest storage position is reached, regardless of whether the field length is exhausted or not. No error indication is given.

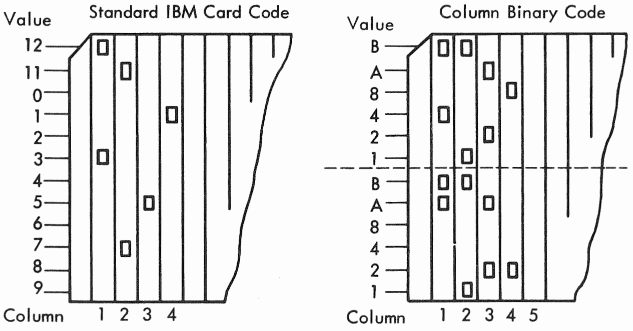


Figure 6. IBM Punched Card Codes 01202

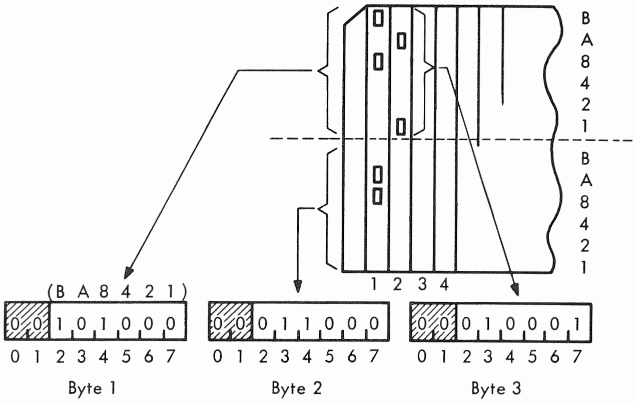


Figure 7. Column Binary Reading 01201

The column binary feature is activated by an additional bit in the FS field of the XIO instruction. When bit 12 of the instruction (this is the 8-bit of the FS field) is present, the reader reads in column binary mode; when bit 12 of the instruction is zero (not present), the reader reads column-by-column. Thus, the column binary mode of reading is specified when the value of the FS is increased by eight.

The following list gives all read instructions, with the regular FS and the column binary FS, for all card I/O devices to which the column binary feature is applicable.

2501 Card Reader

DA	FS	Function	Mode
1	2	Read Card	EBCDIC
1	10	Read Card	Column Binary

2560 MFCM

DA	FS	Function	Mode
2	2	Read Primary Card	EBCDIC
2	3	Read Secondary Card	EBCDIC
2	10	Read Primary Card	Column Binary
2	11	Read Secondary Card	Column Binary

2520 Card Read Punch

DA	FS	Function	Mode
2	2	Read Card	EBCDIC
2	10	Read Card	Column Binary

IBM 2203 PRINTER, MODELS A1 AND A2

Introduction

The IBM 2203 Printer is available in two Models (A1 and A2). The 2203 Printer Model A1 can be attached to 2020 Submodels 1, 2, or 5, and the 2203 Model A2 can be attached to 2020 Submodels 3 and 4. The 2203 Model A1 provides output at speeds up to 750 lines per minute (lpm), and Model A2 provides output at speeds up to 600 lpm. Interchangeable typebars allow the operator to select a type style and character set for a specific printing job. Four character sets are available for the 2203 Printer. The 13-character set has 10 numeric and 3 special characters; the character sets with 39, 52, and 62 characters have 26 alphabetic, 10 numeric and 3, 16, and 26 special characters, respectively.

The printing speed for any one application depends on the total number of lines printed; the amount of processing required for each printed line; and the character set used.

The complete range of speeds available with the 2203 is shown in the table below.

	Model A1		Model A2	
Character Set	lpm	Cycle Time (ms)	lpm	Cycle Time (ms)
13	750	80	600	100
39	425	141	300	200
52	350	171	260	230
62	300	200	230	260

A 120-character line, at 10 characters to the inch, is standard. An additional 24 positions are available as a special feature. Vertical spacing of six or eight lines per inch can be manually selected by the operator. Single, double, and triple spacing of lines, plus skipping to a predetermined point, are performed by the tape-controlled carriage, directed by the CPU. The sequence and arrangement of data printed are also controlled by the stored program; a line to be printed is assembled in core storage in exactly the same sequence it is to appear as output.

The Dual-Feed Carriage special feature permits independent and simultaneous control of two sets of forms.

2203 Printer Instructions

Transfer Instructions (XIO)

Function	Op Code	DA	FS
Print	D0	4	0
Print & Space Suppress	D0	4	1

The direct or effective address (derived from the B1-D1 fields of the instruction) specifies the leftmost byte of the output data field in main storage. The length of the output field is derived from the B2-D2 fields of the instruction.

The field length specification is limited to binary values equal to or less than 144 for the 2203 Printer.

At the completion of the Print instruction, the carriage performs an automatic single space unless otherwise directed by the program. The automatic single space does not occur following a Print and Space Suppress operation.

Test I/O and Branch Instructions (TIOB)

Function	Op Code	DA	FS
Test Printer Busy	9A	4	0
Test Printer Error	9A	4	1
Test Channel 9	9A	4	2
Test Channel 12	9A	4	3
Test Carriage Busy	9A	4	6

Test Printer Busy: The printer busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Printer Error: The Print Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address.

Test Channel 9: The carriage channel 9 condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The carriage channel 9 condition is reset by the branch test, or is reset when the carriage moves to or beyond channel 1.

Test Channel 12: The carriage channel 12 condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The carriage channel 12 condition is reset by the branch test, or is reset when the carriage moves to or beyond channel 1.

If a print operation which includes carriage motion after printing is in progress when either the Test Channel 9 or the Test Channel 12 instructions are encountered in the program sequence, the CPU is interlocked until the completion of the print operation; at this time the test is performed and the program continues.

Test Carriage Busy: The carriage busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise normal instruction sequencing proceeds with the updated instruction address.

Control I/O Instruction (CIO)

Function	Op Code	DA	FS
Control Carriage, Immediate Space	9B	4	4
Control Carriage, Immediate Skip	9B	4	5
Control Carriage, Delayed Space	9B	4	6
Control Carriage, Delayed Skip	9B	4	7

A control carriage instruction in which bits 28, 29, 30, and 31 are all zero is treated as a no-operation.

The direct or effective detailed specification, derived from the B1-D1 fields of the Control Carriage instruction, indicates the particular line or carriage channel to which the carriage is to advance. Only the lowest order four bits of the detailed specification (bits 28-31) are required; the remaining higher order bits are ignored. Carriage channel selection is as follows:

Bits				
28	29	30	31	Specification
0	0	0	1	Single Space or Channel 1
0	0	1	0	Double Space or Channel 2
0	0	1	1	Triple Space or Channel 3
0	1	0	0	Channel 4
0	1	0	1	Channel 5
0	1	1	0	Channel 6
0	1	1	1	Channel 7
1	0	0	0	Channel 8
1	0	0	1	Channel 9
1	0	1	0	Channel 10
1	0	1	1	Channel 11
1	1	0	0	Channel 12

NOTE: After a skip operation (for example, a skip to channel 1), the form is stopped on the particular line to which it was to advance (for example, line 77). If the control tape contains a punch on line 78 in some other channel (for example, channel 2), the programmer can move the form to this next line by issuing a single-space instruction. However, if a Skip to Channel 2 instruction is issued instead, the skip operation is automatically altered into a single-space instruction and the form advances only to the next line.

Dual-Feed Carriage

A dual-feed carriage is available as a special feature for the 2203 Printer. The function specification in the Control Carriage instruction is expanded to permit the upper and lower carriage feeds to be controlled independently or as a single feed, as follows:

Function Specification		Function	Feed
4	(0 1 0 0)	Immediate Space	Lower
5	(0 1 0 1)	Immediate Skip	Lower
6	(0 1 1 0)	Delayed Space	Lower
7	(0 1 1 1)	Delayed Skip	Lower
8	(1 0 0 0)	Immediate Space	Upper
9	(1 0 0 1)	Immediate Skip	Upper
A	(1 0 1 0)	Delayed Space	Upper
B	(1 0 1 1)	Delayed Skip	Upper
C	(1 1 0 0)	Immediate Space	Both
D	(1 1 0 1)	Immediate Skip	Both
E	(1 1 1 0)	Delayed Space	Both
F	(1 1 1 1)	Delayed Skip	Both

The format of the detailed specification (derived from the B1-D1 fields of the Control Carriage instruction) for the dual-feed carriage has six carriage channels. These six channels are numbered 1, 2, 3, 4, 9, and 12, and represent the standard equipment of the upper carriage. Six additional channels (numbered 5, 6, 7, 8, 10, and 11) are available, however, as a feature, thus providing a total of 12 channels for the upper carriage. The upper carriage is controlled by a Carriage Control instruction that has either one of the "upper" or one of the "both" function specifications, plus a channel indication that is derived from the detailed function specification field.

NOTE: If an Upper Carriage Control instruction specifies one of the additional six channels when the additional channel feature is not installed, the upper carriage performs a single space.

A delayed carriage command for either feed (or both) may be cancelled and replaced with another carriage command before the next Print instruction is encountered in the program.

A Print instruction on a 2203 Printer with dual-feed carriage is followed by an automatic single space on either or both carriages in the absence of delayed Carriage Control instructions for the respective carriages.

This automatic single space does not occur on either carriage following a Print and Space Suppress instruction. The Print and Space Suppress instruction does not cancel a previously issued delayed Control Carriage instruction for either carriage.

The testable Carriage Channel 9 and Carriage Channel 12 indicators are completely independent for each carriage. The function specification of the Test I/O and Branch instruction for the 2203 Printer is expanded when the dual-feed carriage is installed to permit testing of these indicators for the upper carriage as well as for the lower carriage. The function specification for the lower carriage of the dual-feed carriage is the same as for a single-feed carriage.

Function Specification		Condition Tested	Carriage
2	(0 0 1 0)	Channel 9	Lower
3	(0 0 1 1)	Channel 12	Lower
4	(0 1 0 0)	Channel 9	Upper
5	(0 1 0 1)	Channel 12	Upper

An "Upper-Lower " Carriage Select switch is provided with the dual-feed carriage to select the carriage to which the Carriage Space and Carriage Restore functions apply when the respective key is operated.

The Carriage Stop key is effective for either carriage at all times.

The Forms Check, End of Forms, and Carriage Interlock indicators are common to both carriages.

Condition Code

XIO Instruction: The condition code is set to 00, 01, or 11 at the time the execution of an XIO instruction for the printer is completed to indicate that the printer is in the available, working, or not operational state.

The I/O operation and data transfer as specified in an XIO instruction for the printer is initiated only when the printer is in the available state.

The printer is available when the Ready indicator is on and a previous print operation, including carriage motion, is not in progress.

The printer is working or busy during the execution of a previous print operation, including carriage motion after printing.

The printer is not operational when the Ready indicator is not on. The Ready indicator is off when the carriage control tape is not installed. If a Print instruction or Control Carriage instruction is issued when the control tape is missing, the condition code is set to 11 (not operational) and the program continues with the next sequential instruction.

On a printer that is equipped with the dual-feed carriage, both carriage control tapes must be installed even if only one feed is used.

CIO Instruction: The condition code is set to 00, 01, or 11 at the time the execution of a Control Carriage instruction is completed to indicate that the carriage is in the available, working, or not operational state.

The carriage operation specified is initiated only when the carriage is in the available state.

The carriage is available when the printer Ready indicator is on if a previous print operation (which includes carriage motion after printing) or a previous carriage operation is not in progress.

The carriage is working or busy during the execution of a previous print operation (which includes carriage motion after printing) or during the execution of a previous carriage operation.

The carriage is not operational when the printer Ready indicator is not on. The Ready indicator is off when the carriage control tape is not installed. If a Print instruction or Control Carriage instruction is issued when the control tape is missing, the condition code is set to 11 (not operational) and the program continues with the next sequential instruction.

On a printer that is equipped with the dual-feed carriage, both carriage control tapes must be installed even if only one feed is used.

IBM 1403 PRINTER, MODELS 2, 7, AND N1

Introduction

The IBM 1403 Printer provides output for the System/360 Model 20 at a rate of 600 lines per minute for the 1403-2 and 1403-7, and 1100 lines per minute for the 1403-N1. If the 1403-N1 is equipped with the Universal Character Set (UCS) special feature, the maximum print speed is 1400 lines per minute. If the 1403-2 is equipped with UCS, the maximum print speed is 750 lines per minute.

The 1403-2 and 1403-N1 have a print line width of 132 characters and the 1403-7 has a print line width of 120 characters. Vertical spacing of six or eight lines to the inch can be manually selected by the operator. Single, double, and triple spacing of lines, plus skipping to a predetermined point are performed by the tape-controlled carriage, under control of the CPU stored program. The 1403-2 and 1403-N1 have a dual-speed carriage that permits high-speed skipping at approximately 75 inches per second on skips over eight lines.

Each print position can print 48 different characters; however, with the UCS special feature, each print position can print up to 240 characters. The printing format is controlled by the stored program. The 1403-2 and 1403-7 have the characters assembled in a chain; the 1403-N1 uses the IBM 1416 Interchangeable Train Cartridge. As the chain (or train) travels in a horizontal plane, each character is printed as it is positioned opposite a magnet-driven hammer that presses the form against the chain (or train).

As each character is printed, checking circuits are set up to ensure that the character printed is correct. Checks are also made to ensure that overprinting does not occur. If an error is detected, the machine stops and the associated check light turns on.

Universal Character Set (1403-2 and -N1)

The UCS special feature allows the printing of any set of graphics (up to 240 different characters) by a 1403-2 or N1 Printer attached to the IBM System/360 Model 20. The graphics can be arranged in any desired sequence on the print chain (1403-2) or print train (1403-N1).

Selective Tape Listing

The Selective Tape Listing special feature allows simultaneous printing on either eight narrow (1.5" width) or four wide (3.1" width) paper tape rolls or folding forms. The 1403-2 and -7 use paper tape rolls and the 1403-N1 uses folding forms. Each tape is controlled individually, one space at a time; spacing occurs after a line has been printed. The operation of this feature is controlled by a switch on the printer. For detailed descriptions of the various printer models, operating procedures, optional features as well as information on character sets, refer to IBM 1403 Component Description (Form A24-3073).

1403 Printer Instructions

Transfer Instructions (XIO)

Function	Op Code	DA	FS
Print	D0	4	0
Print & Space Suppress	D0	4	1
Load Chain Buffer and Disable Folding	D0	4	2
Load Chain Buffer and Enable Folding	D0	4	3

The direct or effective address (derived from the B1-D1 fields of the instruction) specifies the leftmost byte of the output data field in main storage. The length of the output field is derived from the B2-D2 fields of the instruction.

The field length specification is limited to binary values equal to or less than 132 for the 1403-2 and N1, or 120 for the 1403-7.

At the completion of the Print instruction, the carriage performs an automatic single space unless otherwise directed by the program. The automatic single space does not occur following a Print and Space Suppress instruction.

Test I/O and Branch Instructions (TIOB)

Function	Op Code	DA	FS
Test Printer Busy	9A	4	0
Test Printer Error	9A	4	1
Test Channel 9	9A	4	2
Test Channel 12	9A	4	3
Test Carriage Busy	9A	4	6

Test Printer Busy: The printer busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The printer busy condition and the carriage busy condition (see "Test Carriage Busy") are of the same duration.

Test Printer Error: The Print Check indicator is tested. If this indicator is on, the updated instruction address is replaced by the branch address;

otherwise, normal instruction sequencing proceeds with the updated instruction address. With an IBM 1403 Printer attached to the system, the print check indicates misfiring of hammers.

Test Channel 9: The carriage channel 9 condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The carriage channel 9 condition is reset by the branch test, and is reset when the carriage moves to or beyond channel 1.

Test Channel 12: The carriage channel 12 condition is tested. If this condition exists; the updated instruction address is replaced by the branch address; otherwise, normal instruction sequencing proceeds with the updated instruction address. The carriage channel 12 condition is reset by the branch test, and is reset when the carriage moves to or beyond channel 1.

If a print operation which includes carriage motion after printing is in progress when either the Test Channel 9 or the Test Channel 12 instructions are encountered in the program sequence, the CPU is interlocked until the completion of the print operation; at this time the test is performed and the program continues.

Test Carriage Busy: The carriage busy condition is tested. If this condition exists, the updated instruction address is replaced by the branch address; otherwise normal instruction sequencing proceeds with the updated instruction address. The carriage busy condition and the printer busy condition are of the same duration.

Control I/O Instruction (CIO)

Function	Op Code	DA	FS
Control Carriage, Immediate Space	9B	4	4
Control Carriage, Immediate Skip	9B	4	5
Control Carriage, Delayed Space	9B	4	6
Control Carriage, Delayed Skip	9B	4	7

A Control Carriage instruction in which bits 28, 29, 30, and 31 are all zero is treated as a no-operation.

The direct or effective detailed specification (derived from the B1-D1 fields of the Control Carriage instruction) indicates the particular line or carriage channel to which the carriage is to advance. Only the lowest-order four bits of the detailed specification (bits 28-31) are required; the remaining higher-order bits are ignored. Carriage channel selection is as follows:

Bits				
28	29	30	31	Specification
0	0	0	1	Single Space or Channel 1
0	0	1	0	Double Space or Channel 2
0	0	1	1	Triple Space or Channel 3
0	1	0	0	Channel 4
0	1	0	1	Channel 5
0	1	1	0	Channel 6
0	1	1	1	Channel 7
1	0	0	0	Channel 8
1	0	0	1	Channel 9
1	0	1	0	Channel 10
1	0	1	1	Channel 11
1	1	0	0	Channel 12

NOTE: After a skip operation (for example, a skip to channel 1), the form is stopped on the particular line to which it was to advance (for example, line 77). If the control tape contains a punch on line 78 in some other channel (for example, channel 2), the programmer can move the form to this next line by issuing a single-space instruction. However, if a Skip to Channel 2 instruction is issued instead, the skip operation is automatically altered into a single-space instruction and the form advances only to the next line.

Selective Tape Listing: When the switch for this feature is ON, any carriage control I/O instruction has the effect of a delayed single space, regardless of the function specification setting. The function specification, derived from the B1-D1 fields, controls the tapes in the following manner:

Bits	24	25	26	27	28	29	30	31
Tapes	8	7	6	5	4	3	2	1

For example, if bit 24 is a 1 bit, tape 8 is advanced for one space after printing. The wide tape (3.1" width) covers the positions of two narrow tapes requiring that both positions be set to 1 bits to space the double width tape properly. If more than one Control Carriage instruction is interpreted before a Print instruction, only the last one is executed.

Condition Code

XIO Instruction: The condition code is set to 00, 01, or 11 at the time the execution of an XIO instruction for the printer is completed to indicate that the printer is in the available, working, or not operational state.

The I/O operation and data transfer as specified in an XIO instruction for the printer is initiated only when the printer is in the available state.

The printer is available when the Ready indicator is on and a previous print operation, including carriage motion, is not in progress.

The printer is working or busy during the execution of a previous print operation, including carriage motion after printing.

The printer is not operational when the Ready indicator is not on.

CIO Instruction: The condition code is set to 00, 01, or 11 at the time the execution of a Control Carriage instruction is completed to indicate that the carriage is in the available, working, or not operational state.

The carriage operation specified is initiated only when the carriage is in the available state.

The carriage is available when the printer Ready indicator is on if a previous print operation which includes carriage motion after printing, or a previous carriage operation is not in progress.

The carriage is working or busy during the execution of a previous print operation which includes carriage motion after printing, or during the execution of a previous carriage operation.

The carriage is not operational when the printer Ready indicator is not on, or when the carriage control tape is not inserted.

IBM 2152 PRINTER-KEYBOARD

INTRODUCTION

The IBM 2152 Printer-Keyboard consists of an IBM Selectric® Typewriter mounted on a table that is cable-connected to an attachment in the System/360 Model 20 CPU. The 2152 Printer-Keyboard is used mainly as an inquiry station. For example, it allows an operator to retrieve information from a disk file, and to print the information on paper; these inquiries are advantageous when jobs are being run sequentially and the operator needs to know the current working status. The 2152 can also be used for entering certain variable job parameters (such as program decision factors and calculation factors).

The 2152 may also be used as a secondary printer. For example, relatively low frequency messages (such as back-order or re-order information developed during billing or stock control operations) may be printed during a regular run without interfering with the operation of the primary printer of the system. Two separate reports may thus be produced by the same program.

The width of the 2152 print line is 125 characters (maximum) and the vertical spacing is three or six lines per inch (standard), four or eight lines per inch (optional). The 2152 prints at 15.5 characters per second, using a print element of standard System/360 layout, and producing one original print and up to four copies.

The standard equipment includes a pin feed platen (which allows perforated continuous forms to be used) and a paper guide. A friction platen is optional.

General Operation

The 2152 can operate only when the Model 20 has power on. In addition, if the device is to be used as an inquiry station or secondary printer, the keyboard on-line/off-line switch (Figure 8) must be in the on-line position. In off-line mode, the system can neither respond to inquiries nor print unsolicited messages because the attachment is in a reset state.

In on-line mode, the 2152 is under stored program control, thus the system may print unsolicited messages as well as replies to inquiries; therefore the keyboard is locked mechanically during on-line mode. To gain access to the keyboard, the operator must press the Request (REQ) key; this action is acknowledged by the keyboard Request (R) light coming on. The program eventually responds to the request by issuing a Read instruction, the keyboard Proceed (P) light comes on and, at the same time, the keyboard is unlocked. The operator can now enter a message and may key-in as many characters as the field length of the Read instruction permits. The keyboard locks automatically when the allotted field has been filled.

Regardless of whether the field length has been used partially or wholly, the operator must terminate the message by depressing either the End of Transmission (EOT) key, for a valid message, or the Cancel (CAN) key, for an invalid message. When the Cancel key is depressed, a program-testable Cancel indicator is set; however, the message in main storage that has been declared invalid is not destroyed. The program may repeat the operation by issuing another Read instruction.

A special electronic circuit guards against accidental operation of the space bar together with another data key. Should this occur, no data can enter main storage and the keyboard locks. The operator is then forced to cancel the message.

In on-line mode, the program may issue Write instructions to print out messages on the 2152. During write operations, the keyboard is principally locked so that the operator cannot intervene accidentally, however, he can press the Request key at any time.

INSTRUCTIONS

Instruction Set

The 2152 is programmed by the Model 20 instructions shown in Table 9.

Table 9. IBM 2152 Instruction Set 01640

Format	Type	Op Code	DA	FS	Name
SS	XIO	D0	E	1	Read (from Printer)
SS	XIO	D0	E	2	Write without Carrier Return/Line Feed
SS	XIO	D0	E	3	Write with Carrier Return/ Line Feed
SI	CIO	9B	E	1	Carrier Return/Line Feed
SI	CIO	9B	E	2	Enable Request
SI	CIO	9B	E	3	Disable Request
SI	TIOB	9A	E	0	Test Printer Busy
SI	TIOB	9A	E	1	Test Inquiry Request
SI	TIOB	9A	E	2	Test Any Check
SI	TIOB	9A	E	3	Test Cancel
SI	TIOB	9A	E	4	Test P1-Check
SI	TIOB	9A	E	5	Test P2-Check

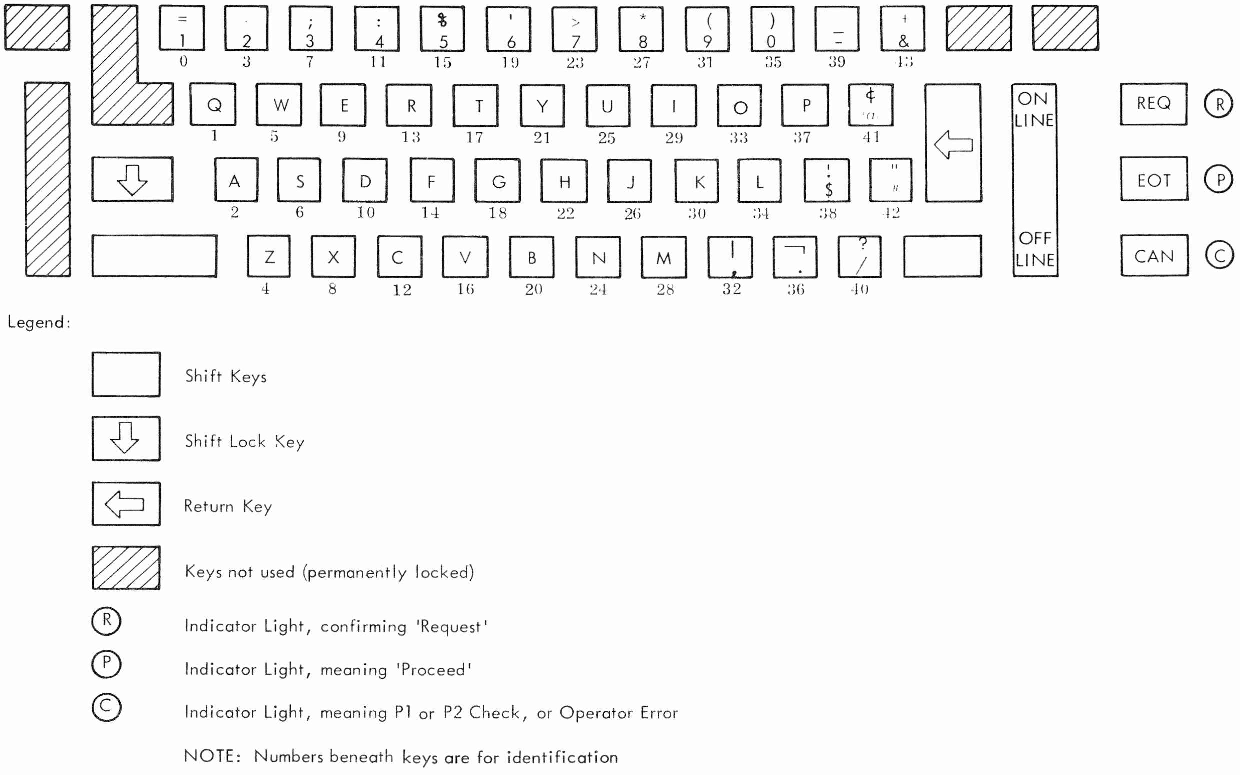


Figure 8. Keyboard Layout 01643

XIO Instructions

The 2152 Transfer I/O (XIO) instructions are executed only when the 2152 is ready. If the device is not ready, the instruction is rejected and a program-testable indication of the reason for the rejection is given in the form of a condition code.

If the 2152 is ready, an XIO instruction is accepted and the device becomes busy for the duration of the operation. The ending of an XIO instruction is indicated to the program by a device end interrupt. The placing of the interrupt request ends the 2152 busy status.

The foregoing characteristics pertain to the Read instruction and both Write instructions, and to the Carrier Return/Line Feed instruction although this is a Control I/O (CIO) instruction. The Read and Write instructions are limited to a field length of 511 bytes; if this rule is violated, a program error stop occurs with a 6 in data register I. The other Model 20 programming rules (even boundary, protected area, main storage limits, direct or effective addressing) must also be observed.

CIO Instructions

The 2152 Control I/O instructions are executed regardless of the status of the device. If the 2152 is off-line or if an unspecified function is stated in the instruction, the execution of a CIO instruction results in no operation. The 2152 cannot become busy while a CIO instruction is being executed, and the termination of this instruction is not accompanied by a request for interrupt. However, these characteristics do not apply to the Carrier Return/Line Feed instruction, which is treated as an XIO instruction. CIO instructions contain a B1-D1 field that may be used to state a detailed function specification; however, this field is not used for the 2152. The B1-D1 field must, nevertheless, not be zero (Model 20 programming rule).

TIOB Instructions

The 2152 Test I/O and Branch (TIOB) instructions are executed regardless of the status of the device. If the tested condition is present, the program branches to the address defined in the B1-D1 field of the instruction. If the tested condition is not present or if the 2152 is off-line or not connected etc., the program continues with the next sequential instruction. TIOB instructions cannot set the 2152 busy status and an interrupt is not requested when such instructions have been completed. The testing of a particular indicator (except Busy, and Any Check) causes it to be reset; the Busy indicator is reset when the busy condition ends, and the Any Check indicator is reset when all conditions covered by this indicator have been reset.

Ready Condition

The 2152 ready condition is a prerequisite for the successful execution of all Read, Write, and Carrier Return instructions. The device is ready when all of the following requirements are met:

1. 2152 attached.
2. On-line mode.
3. No end-of-forms condition.
4. Not busy.
5. 48-volt dc potential present.
6. No P1 or P2 check.

Condition Code

Before a Read, Write, or Carrier Return instruction is executed, the status of the 2152 is examined and a condition code is set.

Condition Code 00 (binary): Indicates that the 2152 is ready.

Condition Code 01 (binary): Indicates that the 2152 is busy executing a previously-issued Read, Write, or Carrier Return instruction.

Condition Code 11 (binary): Indicates that the 2152 is not operational because of one of the following conditions:

1. 2152 not attached (cables disconnected).
2. Off-line mode.
3. 48-volt dc failure.
4. End-of-forms condition.
5. P1 or P2 check present.

The not-operational condition is also indicated by the 2152 light in the attention field of the CPU console; this light comes on when a 2152 XIO instruction is issued while the device is not ready.

2152 Interrupts

The 2152 requests one of two interrupts: device end, or inquiry. The device end interrupt has priority.

Device End Interrupt

The device end interrupt is requested whenever a Read, Write, or Carrier Return instruction has ended. For this interrupt, the device address /E/ and one of two function specifications are stored in the old Program Status Word (PSW) to indicate which operation has ended. Identification is as follows:

Indication	DA	FS
Read End	E	1
Write End, Or Carrier Return End	E	2

At interrupt time, the program may test on errors and on the type of termination. For example, if an operation was ended by off-line switching, the Cancel indicator is on. A check on the residual count positions 162 and 163 can show how much of the allotted field length has been used. The device end interrupt remains pending when the channel mask bit in the PSW is off.

Inquiry Interrupt

This interrupt is identified by device address /E/ and function specification 3. An inquiry interrupt is requested when the operator depresses the Request key. To satisfy the operator's request, the program eventually issues a Read instruction.

However, when the Request key is depressed, the request is only stored; the associated interrupt can occur only when the request is enabled. The request can be enabled either before or any time after operation of the Request key. Inquiry requests can disturb operations in progress and are, therefore, subject to program control via Enable Request and Disable Request instructions. For example, an already enabled request can be disabled provided that the desired inquiry interrupt has not yet occurred. Nevertheless, the request still remains pending until it is enabled, and the inquiry interrupt can then occur.

Because all Model 20 interrupts can be disabled via channel mask zero, a special Test instruction (Test Inquiry Request) is provided by which 2152 inquiry requests can be satisfied without interrupt. However, even in these cases a request is recognized only when it has been enabled.

PRINCIPLES OF OPERATION

All 2152 operations are performed in time-sharing mode with other input/output and processing operations except when burst mode (tape or disk file) operations are in progress.

Read Operation

A read operation is executed only when the 2152 is ready. If the device is ready, a Read instruction can be accepted whereupon the 2152 becomes busy, the keyboard Proceed light comes on, and the keyboard is unlocked. The operator may now key-in data at any speed because there are no timeout conditions to be met. The carrier proceeds from left to right, one step at a time (with each key operation), until it reaches the right-hand end position. At this position, the keyboard locks, except for the space bar. No space character, however, can enter main storage as long as the carrier is positioned at the right-hand end.

To start a new print line, the operator must depress the keyboard Return key. This action causes the carrier to return to the left-hand end, whereupon the form is advanced one step. A lever at the platen allows the operator to select the vertical size of this step; with the standard platen, the line feed ratio can thus be adjusted to either three or six lines per inch.

NOTE: During carrier return, the keyboard is unlocked; if data keys are operated, data enters main storage and is also printed. Data-key operation during carrier return should be avoided.

The Return key can be operated repeatedly to advance the paper the required number of lines. The operation of the space bar causes one blank character to be transferred for each space-bar operation. The operation of function keys, such as Shift (to upper case), EOT or Cancel, initiates the desired action but no function characters are transferred to main storage.

If the field length is used up during a read operation, the keyboard is locked so that the read-in area cannot be overrun. However, the Proceed light remains on until the operation ends. The message is ended when either EOT or CAN is depressed; this action can occur at any time during the read operation. EOT is used when the operator declares the message to be valid; the Cancel key is used when the message must be altered (that is, declared invalid).

NOTE: If the operator accidentally depresses the space bar together with a data key, the keyboard locks and the check light comes on; if this has happened, neither the space nor the character has entered main storage. This situation is termed "operator error". A message, during which an operator error occurs, must be terminated via the Cancel key; EOT has no function in this case. The operator error is thus indicated to the program by the testable Cancel indicator.

The Cancel indicator is also set when a read operation is terminated by the switching of the 2152 to off-line mode. The cancelled message is not destroyed; the program may allow for correction by repeating the read operation. If a message is ended by either EOT, Cancel, or off-line switching, the carrier returns automatically to the left-hand margin and line feed occurs. If either a P1 or P2 check occurs or if the end-of-forms contact operates during a read operation, the operation is not disturbed but can continue to the end. However, the 2152 becomes not ready thereafter, and a new Read, Write, or Carrier Return/Line Feed instruction cannot be executed unless the ready status is restored. At the end of a Read instruction, device end interrupt is requested regardless of the method of termination, and a residual length count is stored in positions 162, 163.

The issue of a Read instruction has no influence on the inquiry interrupt bit, which is reset only by execution of the interrupt. The operator can, therefore, rely on the Request light being on until interrupt occurs. This arrangement is necessary because the program may volunteer a Read instruction (for example, in order to allow the operator to enter the current date) immediately after the Request key has been operated. However, the operator may have pressed the Request key for a different purpose and, therefore, his request is not satisfied by the program-volunteered Read instruction but only by the inquiry interrupt or by a program test on the inquiry request.

Write Operation

Two different write operations can be performed by the stored program: write with carrier return/line feed, and write without carrier return/line feed. In both operations, the keyboard of the 2152 remains locked throughout, and the carrier returns automatically when the right-hand end position is reached. The operations differ only in the action of the carrier when the respective operations terminate. When the write-with-carrier-return operations ends, the carrier returns from the position reached to the left-hand end. When the write-without-carrier-return operation ends, the carrier is left at the position adjacent to the last character that was printed; consequently, any subsequent operation (read or write) begins at this position, with no space between the adjacent texts.

Write operations cannot be stopped by the depression of EOT or CAN, by P1 or P2 checks, or by an end-of-forms condition. However, if a P1 or P2 check or an end-of-forms condition arises, the 2152 becomes not ready and any new Read, Write, or Carrier Return/Line Feed instruction cannot be executed unless the ready status is restored. Only two keys are operational during write operations: REQ and ON LINE/OFF LINE. Requests are stored during write operations and are executed later (see "Inquiry Interrupt" under "2152 Interrupts"). If the 2152 is switched to off-line mode during a write operation, the operation ends, the Cancel indicator is set, and the carrier returns even if the instruction specified no carrier return. At the end of a write operation, device end interrupt is requested regardless of the method of termination. A residual field length count is stored in positions 162, 163.

Control Operations

Carrier Return/Line Feed

The Carrier Return/Line Feed instruction is executed only when the 2152 is ready. This instruction causes the carrier to return to the left-hand margin and the form to advance to the next print line; if the carrier is already positioned at this margin, the only action is line feed. The instruction can be issued repeatedly to move the paper a corresponding number of lines. The 2152 is busy during carrier return until the device end interrupt is requested, which occurs when the carrier has reached the left-hand margin.

Enable Request

Requests are disabled automatically after system power on and system reset. If a request is entered via the Request key, the request is stored and remains pending until it is enabled by the Enable instruction. When a pending request is enabled, it will request an inquiry interrupt. A request that has been entered via the Request key can be extinguished by switching the 2152 to off-line; however, the Enable instruction remains effective until either Disable Request is issued or the system reset key is operated.

Disable Request

Inquiry interrupts may be inconvenient during certain program runs. In order to suppress these interrupts without suppressing all others, the 2152 request can be disabled, either before or after the Request key has been operated.

If Disable Request is issued before the Request key is depressed, a subsequent Request key operation will nevertheless cause a request to be stored; however, no interrupt action will occur until this stored request is enabled.

If Disable Request is issued after a request has been entered via the Request key but before the corresponding inquiry interrupt has been executed, then this inquiry interrupt cannot occur until the request has been enabled. Once a request has been entered, it remains stored unless the 2152 is switched to off-line mode.

TIOB Operations

Test Inquiry Request

The inquiry request can be tested directly to allow the program to branch when interrupts are disabled via channel mask zero. However, the test can find enabled requests only; the branch then occurs and the request is reset. A test on a request that was entered but not enabled cannot cause a branch.

Test Printer Busy

The printer busy condition can be tested to prevent initiation of an operation that cannot be performed successfully. The 2152 becomes busy when it has accepted a Read, Write, or Carrier Return/Line Feed instruction. The busy condition ends when device end interrupt is requested, which occurs at the end of any of these operations. The busy condition is not reset by the test. The program branches when the 2152 is found to be busy; the program continues with the next sequential instruction when the device is found to be not busy.

Test Cancel

The Cancel indicator is set whenever the keyboard Cancel key is depressed during read operations. (The Cancel key is ineffective during write or carrier-return operations.) The indicator is also set whenever the 2152 is switched off-line. The Cancel indicator can be tested to find out whether the previous operation should be repeated. If the indicator is on, the test causes the program to branch and the indicator to be reset. However, the test is not mandatory because the indicator is reset automatically with the next Read, Write, or Carrier Return/Line Feed instruction that is issued after the indicator is set.

Test P1 Check

The P1 check indicates that the attachment to which the 2152 transmits (or from which it receives data) contains an even bit configuration. This parity error can occur during a read or a write operation; however, the operation is not stopped but can continue to completion. The P1 check sets the 2152 not-ready status and turns on the 2152 check light. The 2152 indicator light on the CPU console comes on if a new XIO instruction is issued. The test on this P1 check causes the program to branch, and both lights go out. A new Read, Write, or Carrier Return/Line Feed instruction is rejected with condition code 11 (not operational) as long as the P1 check is present. During operator error, the P1 check is suppressed.

Test on P2 Check

The P2 check indicates that the mechanism of the 2152 has operated a wrong combination of contacts, possibly during a read or write operation. The operation is not, however, stopped but can continue to completion. The P2 check sets the 2152 not-ready status and turns on the 2152 check light. The 2152 indicator light on the CPU console comes on if a new XIO instruction is issued. A new Read, Write, or Carrier Return/Line Feed instruction is rejected with condition code 11 (not operational) as long as the P2 check is present. The test on the P2 check causes the program to branch and the check light, as well as the indicator light, is turned off. The P2 check is operational at all times and is not suppressed when an operator error occurs.

Test Any Check

The Any Check indicator is a summary indication for any of the following indicators: P1 check, P2 check, and Cancel. The program branches when the Any Check indicator is on, but the indicator is not reset; it is reset only when all indicators that have caused it are reset.

PRINTER CHARACTER SET

The 2152 can print all characters shown within the heavy frames in Figure 9. Bit configurations that pertain to the areas outside these frames are valid EBCDIC configurations but, if used, cause unspecified characters to be printed. Unspecified characters do not cause a check. The printable character set corresponds to the keyboard nomenclature and to the print element layout (Figure 10). One blank character is transferred to main storage for each operation of the space bar, except when the carrier is at the right-hand end of its travel; with the carrier in this position, the space bar loses its function (and the rest of the keyboard

Bit positions 0,1 contain	00				01				10				11			
Bit positions 2,3 contain					00	01	10	11	00	01	10	11	00	01	10	11
Bit positions 4,5,6,7, contain																
0000					blk	&	-									0
0001							/		a	j			A	J		1
0010									b	k	s		B	K	S	2
0011									c	l	t		C	L	T	3
0100									d	m	u		D	M	U	4
0101									e	n	v		E	N	V	5
0110									f	o	w		F	O	W	6
0111									g	p	x		G	P	X	7
1000									h	q	y		H	Q	Y	8
1001									i	r	z		I	R	Z	9
1010					¢	!		:								
1011					.	$	,	#								
1100					<	*	%	@								
1101					(	)	_	'								
1110					+	;	>	=								
1111					|	¬	?	"								

Figure 9. IBM 2152 Character Set to EBCDIC

01641

UPPER CASE											LOWER CASE										
=	;	%	>	*	)	<	:	'	(	"	1	3	5	7	8	0	2	4	6	9	//
?	T	V	X	Y	¢	S	U	W	Z	|	/	t	v	x	y	@	s	u	w	z	,
J	L	N	P	Q	_	K	M	O	R	!	¡	l	n	p	q	-	k	m	o	r	$
A	C	E	G	H	+	B	D	F	I	¬	a	c	e	g	h	&	b	d	f	i	.

Figure 10. Print Element Layout (Type 963) 01642

is mechanically blocked). If the program intends to produce spaces, it must issue a Write instruction and provide blank characters in those positions where a space is to appear.

KEYS AND LIGHTS

Refer to Figure 8 for the location of the various keys and lights.

On Line/Off Line Key

The toggle-type On Line/Off Line key must be depressed at ON LINE if the 2152 is to operate under stored program control. Switching from ON LINE to OFF LINE (or vice versa) causes carrier return with line feed. In off-line mode, the 2152 is not ready and all attachment circuits are in a reset state. Switching to off-line cancels any operation in progress at that time. The key also resets a stored request but, if the request has already caused the inquiry interrupt bit to be stored, an inquiry interrupt will occur eventually. The key has no effect on the enable/disable state of requests.

Data Keys

The term data keys refers to all keys by which data is entered. The data keys can be operated only during off-line mode and during on-line mode when a read instruction has started; otherwise the keyboard is locked.

NOTE: Although no special skill is required to operate the device, the operator should avoid these errors:

1. Operation of data keys together with the space bar (this causes operator error with keyboard lock)。
2. Simultaneous operation of data keys and the Return key (characters may appear in already printed lines on paper but will be stored in adjacent main storage positions).
3. Simultaneous operation of data keys and Shift key (the character on paper will not necessarily correspond to the one in main storage; it may be either the upper or lower case character).

Function Keys

Request Key: The Request key must be depressed to notify the program that the operator intends to enter data into main storage. The operation of the Request key causes a request to be stored. This stored request remains ineffective until it is enabled. An enabled request can subsequently initiate inquiry interrupt or it can be tested directly. The request will ultimately cause the issuing of a Read instruction that unlocks the keyboard. A stored request is extinguished only by off-line switching. The Request key is effective only during on-line mode.

EOT Key: The EOT key is used to terminate a read operation when the operator plans no alterations to the message just entered. The key is effective only during read operations when no operator error is present.

Cancel Key: The Cancel key is used to terminate a read operation during which wrong or unsatisfactory data has been entered. The key must be used for termination when the keyboard has locked because of an operator error (simultaneous activation of space bar and data key). Operation of the Cancel key does not destroy the data in main storage. The Cancel key is effective only during read operations.

Indicator Lights

Request Light: The Request light comes on when the Request key is operated. The light remains on until either an inquiry interrupt occurs or until the request is tested by the Test Inquiry Request instruction. The light goes out if the 2152 is switched to off-line mode. Subsequent on-line switching will neither restore the request nor turn on the light.

Proceed Light: The Proceed light comes on whenever the program issues a Read instruction and this instruction is accepted by the 2152. The light goes out only when the operation is terminated via EOT, Cancel or off-line switching.

Check Light: The check light comes on when either a P1 check, a P2 check or an operator error occurs. If the light came on because of a P1 or a P2 check, it goes out when the respective checks are tested. If the check light was caused by an operator error, it goes off after the Cancel key has been operated.

Attention Light: The Attention light is installed in the attention field of the CPU console. The light comes on whenever an XIO instruction that addresses the 2152 is issued and the device is not ready. The 2152 is not ready when the cables are not attached, the power supply has failed (no 48-volt dc potential), a P1 or P2 check is present, the 2152 is in off-line mode, or the end-of-forms contact has operated. The light goes out when these conditions are corrected.

PAPER HANDLING

The 2152 prints on continuous forms of 13-5/8 in (34, 6 cm) width. These forms are transported by a pin-feed platen. The horizontal distance from pin to pin is 13-1/8 in (33, 3 cm); the vertical distance from pin to pin is 9/16 in (1, 4 cm). The line feed mechanism moves the paper in increments of either 3 or 6 lines per inch (2, 54 cm) as determined by the line adjustment lever at the platen. The maximum print line width is 124 ± 1 characters for read operations, therefore the keyboard usually locks when position 124 is reached; however, if the keyboard locks at position 123, then the space bar can still be operated and thus two space characters can be entered. (Beyond position 125, no additional space characters can be entered, even though the space bar remains unlocked.) For write operations, the maximum print line width is 124 characters. The margins are fixed in the extreme left-hand and right-hand positions . One original and up to four copies can be printed. A paper guide is provided as standard equipment.

End-of-Forms Contact

The end-of-forms contact monitors the unprinted paper stock. The contact closes at a point that allows any read or write operation currently in progress to be completed in such a manner that all information is still printed on paper, even if the maximum field length (511 bytes) is used. The operation of the contact does not disturb current operations but the 2152 becomes not ready thereafter.

Ribbon

A single-color ribbon, enclosed in a cartridge, is supplied. Ribbon cartridges can be obtained in a variety of colors through any IBM branch office.

FEATURES

A pin-feed platen with a line feed ratio of 4 or 8 lines per inch (2, 54 cm) is available as a feature. The distances from pin to pin in horizontal and vertical direction are identical with those of the standard 3 to 6 lines per inch platen (see "Paper Handling"). A friction-feed platen (without pins) is a further feature. A dual paper tray for the printed and unprinted stock is available for those customers using fan-fold paper.

INPUT/OUTPUT CHANNEL (FOR 2020 SUBMODELS 2 AND 5)

INTRODUCTION

The Input/Output Channel (IOC) for the 2020 is an electronic device, housed within the CPU, that directs the flow of information between certain I/O devices and main storage. The IOC provides a standard interface for connecting different types of I/O devices to the CPU, and it relieves the CPU of communicating with the devices in the manner appropriate to each; the CPU deals with the IOC only and, therefore, handles each device in a standardized manner.

The channel accepts control information from the CPU in the format supplied by the program and changes this information into a sequence of signals acceptable to a control unit. Thus, the channel deals with control units without regard for the requirements of any specific device, since the control unit must be capable of "understanding" the channel sequence signals. The control units change the information received from the channel into signals that are suitable for operating one or several devices, and distributes these signals to the addressed device.

The IOC can accommodate up to eight control units, however, programming support is available for only one control unit. The number of I/O devices that can share one control unit depends on the design of the control unit only. A Request for Price Quotation is required to provide the emergency power-off connector for the third and any additional control unit.

Because of its design, the IOC for the 2020 is classified as a selector channel. On a selector channel, any one (but only one) device can be handled at a time. Thus, the device stays logically connected on the channel interface for the entire transfer of all data specified by the program for a particular operation.

Normally, the data transfer is performed in burst mode, which means that all bytes specified are transferred in one continuous string; however, the IOC for the 2020 Submodel 5 can operate in either non-overlap mode or overlap mode. In non-overlap mode, the channel occupies all CPU facilities for its data transfer operations and excludes all other I/O or processing operations. In overlap mode, the channel seizes storage cycles each time that at least one byte (but not more than two bytes) are available for transfer. Thus, channel operations can overlap any other I/O or processing operation, thereby providing increased throughput and improved flexibility.

GENERAL OPERATION

Input/output operations are initiated and controlled by two types of formats, i.e., instructions and commands. Instructions are decoded and executed by the CPU and are part of the stored program. Commands are decoded and executed by the channel and initiate I/O operations such as read or write. Both instructions and commands are fetched from main storage and are common to all types of I/O devices attached to the channel.

Functions peculiar to any device, such as rewinding of tape, are specified by modifier bits in the command. The channel is not concerned with these modifier bits beyond transferring them to the addressed device. Each device has an address which is always a combined number, pertaining partly to the control unit (on which the device is assumed to be connected) and partly to the device. A single device must be capable of performing the functions of a control unit and is then identical with it.

To initiate an I/O operation, the IOC places the address of the desired device on the common bus when instructed by the program to do so. (The bus connects the channel with all control units.) Each control unit attempts to decode the address on the bus; however, only one control unit can respond to a specific address. If a control unit recognizes its address, it returns the address to the channel, whereupon the channel compares the sent-out address with the received one. If both addresses match, the channel places a command on the bus. The command specifies the operation to be performed.

The addressed control unit picks up this command and sends status information in response. The status information reveals the state of the channel, the control unit, and the device as it existed when these facilities were addressed. A condition code indicates to the program that status information is available and shows whether the operation was rejected or accepted. Since this acceptance does not guarantee successful execution of the operation, status information is also available at the end of an operation; this is necessary because the ending conditions are too numerous to be indicated by the condition code alone.

The status information is collected and stored in a fixed location in main storage in the form of a channel status word. The channel status word contains the address of the unit involved in the operation, the status of this unit, the status of the channel, a residual count that shows how many bytes were transferred, and the address of the command (or the last command in a chain) that was performed. The channel status word thus allows the program to decide on further action.

The various operations occupy the CPU, the channel, the control unit, and the device each to a varying degree, but only for the time needed by that particular facility; thus it is possible, for example, that the channel is available while the control unit and device are still busy. These conditions are indicated primarily in the channel status word; however, in some cases, a special indication (in the form of a request for interrupt) is given to the program when particular conditions change. Generally, interrupt indicates that a device has changed from the busy to the available state.

The interrupt indication is needed because the order of priority among the various facilities prevents the start of a new operation: if the channel is busy, the state of the control units or devices is unimportant because no new operation can be executed; if a control unit is busy, none of its devices can execute a new operation; if only a device is busy, all others (except this device) can execute operations. Any of these facilities can be busy during the execution of an operation and also when they have requested interrupt while interrupts are disabled (interrupt pending). A pending interrupt must be cleared to get the respective facility into the available state.

Interrupt is handled differently for non-overlap and for overlap modes. In non-overlap mode, interrupt is requested only for certain control operations when they are completed at the device. Interrupt is not requested for pure data transfer operations. In overlap mode, interrupt is requested for all types of operation whenever any one facility or all facilities involved in the operation become available.

INSTRUCTIONS AND COMMANDS

Start Instruction

All channel operations must be initiated by the Start instruction which serves the purpose of addressing the channel, a control unit, a device, and a command word. If the Start instruction is accepted, the specified command is read out from main storage. The bit structure of the Start instruction (Figure 11) is as follows:

Bits 0 - 7: Must contain D 0 (hexadecimal), which is the fixed op code Transfer I/O.

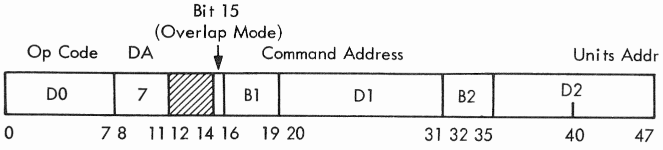


Figure 11. Channel Start or Test I/O Instruction 01953

Bits 8 - 11: Represent the overall device address and must contain 7 if the IOC is to be addressed. (The channel is, to the program, a main device.)

Bits 12 - 14: Are not used, but should be kept zero for future expansion.

Bit 15: Must be zero if overlap mode is specified in the program status word (bit 5 of PSW = 1). If bit 15 is 1 instead of zero when overlap mode is specified, the Start instruction assumes the function of the Test I/O instruction. If non-overlap mode is specified (bit 5 of PSW = 0), bit 15 is ignored.

Bits 16 - 31: Represent the main storage address of a Channel Command Word (CCW). May be used as direct address or an effective address may be derived from them. Neither a direct address nor an effective address may be below 144 or exceed the storage capacity, otherwise a CPU program error stop occurs.

Bits 32 - 47: Represent the combined address of the control unit and individual device; this combined address can be derived from bits 32 through 47 via either direct or effective addressing. However, regardless of the method of addressing, only bits 40 - 47 (of the direct or effective generated value) are used as the combined address. Any low-order portion (bit 47, 46, etc) may be used as the device address with the remaining bits up to bit 40 representing the control unit address. The total value of the combined address must not be zero (program error stop). If a non-existent unit is addressed, condition code 11 (not operational) is set.

Test I/O Instruction (Overlap Mode Only)

The Test I/O instruction is provided specifically for obtaining the current status from a device without starting or attempting any operation on a device. The instruction obtains the status and clears any interrupt conditions that may be pending due to any preceding action in the tested device. (Test I/O does not request interrupt itself.) However, status information is stored and interrupt conditions are cleared only when the addressed device has such information and/or conditions pending. If the device to be tested has not been addressed during a particular program run, it has neither status information nor interrupt conditions pending; in that case, condition code 00 is set, indicating that the status need not be examined because it is "clean". (See "Condition Code Setting" in "Channel-to-Program Response".)

The structure of the Test I/O instruction (see Figure 11) is identical with that of the Start instruction except for bit 15.

Bit 15: Is ignored if non-overlap mode is specified in the program status word (bit 5 of PSW = Ø). Bit 15 of Test I/O must be 1 if overlap mode is specified in the PSW. If bit 15 is zero when overlap mode is specified, the Test I/O instruction assumes the function of the Start instruction.

NOTE: The B1-D1 field in the Test I/O instruction must contain a valid main storage address (not below 144 and not beyond storage capacity) although this address is meaningless for the test I/O function. If this rule is violated, a CPU program error stop occurs.

Programming Note

The Control I/O and Test I/O and Branch instructions used for the native-attached I/O devices have no function for channel-connected devices. A CIO or TIOB instruction with a device address 7 (channel) results in no operation.

Command Word Structure

Commands are specified in a channel command word (Figure 12) that is six bytes long and is located in main storage. The bit structure is as follows:

Bits 0 - 7: Represent the command code. The command code specifies the operation to be performed (Read, Write, Sense, Control).

Bit 8: Represents the chaining flag and specifies whether the next sequential command is to follow If bit 8 is zero, operations end when the present command has been executed. A new command is read out only when a new Start instruction is issued. If bit 8 is 1, the address of the present command is increased by 6, which allows the next command to be fetched when the present command is completed in the normal manner (not ended prematurely by abnormal conditions). The operation specified by this next sequential (chained) command is performed on the same device as was the preceding command. Chaining is possible only between command words that are located adjacent to each other and in an ascending order of address. Chaining is disrupted when bit 8 is zero or when unusual conditions arise. (See also "Programming Notes on Command Chaining" under "Transfer in Channel".)

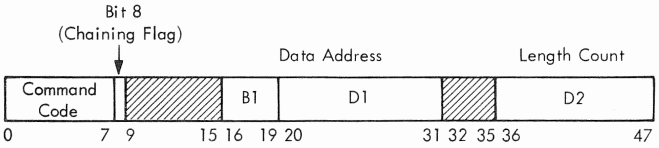


Figure 12. Channel Command Word 01954

Bits 9 - 15: Are not used and should be kept zero for future expansion.

Bits 16 - 31: Represent the data address. This is the address of the first byte of a main storage input or output field that extends from this address in ascending or, in the case of Read Backward, in descending order. In the case of a Transfer in Channel (TIC) command, bits 16 - 31 represent the address of another command word. The address must be valid (i.e., neither below 144 nor beyond storage capacity), otherwise the program check bit in the status word is turned to 1.

Bits 32 - 35: Are not used and should be kept zero for future expansion.

Bits 36 - 47: Represent the length count, i.e., the total number of 8-bit bytes in the input or output data field including the first byte (true field length). The maximum length which may be specified is 4, 095 bytes. However, any length count can lead to encroachment on the upper limit of main storage if the data address is already high; this must be safe-guarded against, otherwise the program check bit in the channel status word is set. A length count of zero is allowed only for the TIC command. If the length count for a data transfer command or a control command is zero, the program check bit is set in the status word. The program check bit is also set if a backward operation runs into the protected area (which could possibly occur due to a low data address and an excessive length count).

COMMAND CODES

The command code (bits 0 through 7 of the command word) specifies the operation to be performed for the IOC and for the device. The channel recognizes only three general types of operation:

1. Output to device: Write operations. Control operations.
2. Input to main storage: Read (forward or backward) operations. Sense operations.
3. Branching: Transfer-in-channel operations.

Therefore, only a small portion of the command code concerns the channel, the other bits of the command code being modifiers that specify how the operation is to be performed by the device. The assignment of these modifier bits (if any) depends only on the design of the device; the channel ignores them. The bit configuration of the command codes is as follows:

	Command Word Bit Pos	
Command	0 1 2 3	4 5 6 7
Read	m m m m	m m 1 0
Read Backward	m m m m	1 1 0 0
Sense	m m m m	0 1 0 0
Write	m m m m	m m 0 1
Control	m m m m	m m 1 1
Transfer in Channel	X X X X	1 0 0 0

The "m" character denotes a modifier bit, and "X" indicates that these bits are ignored by both the channel and the device; they may be used by the programmer for any purpose.

NOTE: A command code in which bits 4 through 7 are all zeros is regarded as invalid and sets the program check bit in the status word, when the command is issued.

A command is executed only when the addressed device has recognized its address, and when neither the device, nor the control unit (if any), nor the channel is busy or not available for any reasons. When an operation has ended, this is indicated by the storing of status information and, in some cases, by a request for interrupt. The following command descriptions show which commands request interrupt upon completion.

Read

The term "read" pertains to a data transfer from the I/O device to the CPU. The read operation is initiated on the device addressed by the Start instruction. The transferred data is stored, beginning at the position defined by the data address, and continuing with successively higher main storage addresses until the number of bytes specified by the length count of the command word has been transferred, or until the device terminates the operation, whichever occurs first. The number of bytes that can be transferred is 4, 095 (maximum), and 1 (minimum).

Interrupt is requested upon termination if Read was performed in overlap mode, and is not requested in non-overlap mode.

Read Backward

The Read Backward command is identical to the Read command except that the transferred data is stored in successively lower main storage positions, starting at the position defined by the data address in the command word. Interrupt is requested upon completion when the operation was executed in overlap mode, and is not requested in non-overlap mode.

Sense

The Sense command is similar to the Read command, except that the data is obtained from status indicators rather than from a record source. By means of Sense, detailed information about unusual conditions detected during the last operation can be obtained, provided the device is designed to furnish such information. The sense information is stored in the main storage location defined by the data address and in successively higher main storage locations as defined by the length count. The number of bits thus read in and their meaning depends on the design of the device; however, the first six bits are designated as follows and described later in "Sense Information":

Bit	Designation
0	Command Reject
1	Intervention Required
2	Bus-Out Check
3	Equipment Check
4	Data Check
5	Overrun

Interrupt is requested upon completion of the Sense command when it was executed in overlap mode. No interrupt is requested for Sense in non-overlap mode.

The execution of a Sense command does not cause the sensed indicators to be reset. However, any following command (regardless of units address) will reset these indicators unless this next command is another Sense. In 2020 Submodels 5, Sense indicators are reset only by the next command issued to the unit from which they originated (unless this next command is another Sense).

Write

The term "write" pertains to a data transfer from the CPU to the I/O device. This data transfer begins at the main storage location defined by the data address in the command word and proceeds with successively higher main storage locations until either the number of bytes specified by the length count has been transferred or until the device ends the operation, whichever occurs first. Interrupt is requested upon termination if the write operation was performed in overlap mode, and is not requested in non-overlap mode. A minimum of 1 byte and a maximum of 4, 059 bytes can be transferred with one Write command.

Control

The Control command is an output command (to the channel devices) similar to Write except that the information transferred is decoded by the device (rather than being recorded) to determine which one of several functions is to be performed. Three types of control operation are possible, and each type generates different responses for the program:

Control Immediate Operation: Occupies the device for significantly longer than it does the channel; therefore, channel end is signalled immediately after the command has been accepted by the control unit. Device end will follow later.

Short Control Operation: Occupies the device for only insignificantly longer than it occupies the channel. Due to the negligible time difference, channel end is delayed until device end occurs and both are presented together.

Data-Transfer Control Operation: Is an operation during which a comparison between main storage data and address registers, etc., is required (such as Seek on an IBM 2311 Disk Storage Drive). This operation may signal channel end when the data transfer portion is completed; it may also delay channel end until device end occurs. The exact response depends on the design of the device.

Interrupt Requests for Controls

The different types of control operations request interrupt as follows:

In Non-Overlap Mode: Interrupt is requested only for control-immediate operations at the time the device terminates the operation (device end after channel end). This can also occur for certain data transfer controls. Short-control operations do not request interrupt.

In Overlap Mode (2020 Submodel 5 only): Control-immediate operations request interrupt when device end is presented. Data-transfer control operations may request interrupt twice, i.e., when channel end is presented and again when device end occurs. However, if channel end is presented already during initial selection, only the device end interrupt is requested.

Transfer in Channel

Command words which are stored in consecutive main storage locations can be chained (via chaining flag), that is, when one command has been executed successfully, the next one in ascending order of address is read out. The Transfer in Channel command provides chaining between commands that are not located adjacent to each other but, instead, are located anywhere outside the protected area. The TIC command specifies the next command to be read out, by means of the data address (bits 16 - 31). The transfer-in-channel operation involves neither the channel nor any device; it is considered to be an unconditional branch. The TIC command itself may be located in a chain, and thus it is possible to skip from one chain to another. The TIC command may contain a chaining flag (bit 8 of command word = 1) but this has no meaning as the command is executed regardless of whether bit 8 is zero or 1.

NOTE: The first command designated by the Start instruction must not be a TIC command; if it is, the program check bit in the channel status word is set. Likewise, a TIC command must not specify TIC again; if it does, program check is set.

Interrupt is not requested when the transfer-in-channel is completed, regardless of whether overlap or non-overlap mode is specified.

Programming Notes on Command Chaining:

A chain of commands is treated as one single operation, which means that status information is available only for the last command of a chain; the program is not notified when any individual command in the chain is completed successfully.

Chaining occurs only, and a new (chained) operation is initiated only, when the command that specifies chaining is completed successfully. If conditions such as unit check, unit exception, control unit end, busy, or incorrect length are detected during the current operation, the operation ends and a new command is not fetched, even if the previous one did specify chaining.

Command chaining is normally performed by increasing the address of the current command by 6; under certain circumstances, however, the current address can be increased by 12, i.e., the next sequential command is skipped and the one after that is issued. This skipping occurs when the device presents the status modifier bit together with the device end signal. The meaning of the status modifier depends only on the design of the device; it may indicate, for example, control unit busy or some other temporary busy condition. (See also "Interrupt during Chaining" in this section.)

NON-OVERLAP/OVERLAP MODE SELECTION (CHANNEL MODE)

The mode in which the IOC operates is determined by bit 5 of the current PSW. If bit 5 is zero, the channel operates in non-overlap mode; if bit 5 is 1, the channel (2020 Submodel 5 only) operates in overlap mode.

Non-Overlap Mode

In non-overlap mode, the channel rejects an instruction with condition code 2 (01 binary) when time-shared operations, except the following, are currently in progress in the 2020:

1. Buffered print operations (on the IBM 1403 Printer only).
2. Tape rewind (on the IBM 2415 Magnetic Tape Unit).
3. Seek (on the IBM 2311 Disk Storage Drive).

All other input/output or processing operations inhibit operations in the non-overlap channel. Conversely, if an operation is in progress in non-overlap mode, time-shared operations cannot be started.

Overlap Mode

In overlap mode, the channel rejects instructions only when they address a channel device that is busy with a previously initiated operation or when either the channel, the addressed control unit or the addressed device has interrupt pending. (See "Pending Channel interrupts".) Time-shared operations can run concurrently with channel operations.

The mode-determining bit 5 is set into the current PSW as desired by a Set PSW instruction. However, if the CPU is a Submodel 2, bit 5 is set to zero even if it was specified as 1, because the 2020 Submodel 2 has no overlap capability. The overlap mode bit (as well as the USASCII mode bit and the condition code) is set to zero when the CPU is in the reset state.

To find out if a 2020 has overlap capability, a Test I/O and Branch instruction is provided. In this instruction, the device address must be specified as F (hexadecimal) and the function specification as zero. The program will branch to the address specified in the B1-D1 field if the CPU has overlap capability, otherwise the program continues with the next sequential instruction.

CHANNEL-TO-PROGRAM RESPONSE

Whenever the program initiates any action in the channel, the channel responds in one of the following ways with information that indicates to the program how it may proceed:

1. Setting a condition code.
2. Storing a channel status word.
3. Requesting interrupt.

Condition Code Setting

The first indication the program receives upon initiation of any action is the setting of a condition code. The condition code gives an approximate breakdown of the situation because it shows whether:

1. The operation has been started successfully.
2. The operation cannot be performed temporarily.
3. The operation definitely cannot be performed.

Since the condition code setting differs for overlap and non-overlap modes, it is described separately. The condition code is represented by bits 2 and 3 of the program status word.

Non-Overlap Mode

In non-overlap mode, only the Start instruction draws a condition code response. Thus, when the Start instruction is issued, the condition code can be set to any of the following values:

<u>01</u>: Indicates that status information has been stored and that the channel status word should be examined, because the initiated operation may have been executed successfully, terminated prematurely, or rejected for any reason. (Condition code 00 is not set.)

<u>10</u>: Indicates that the Start instruction was rejected because the CPU is currently processing time-shared input/output or processing operations. However, when the currently running operations are completed, an interrupt identified by device address / F/ is requested and, if it occurs, the program is notified that the channel will accept the Start instruction.

<u>11</u>: Indicates that the instruction was rejected because of a not-operational condition in either the addressed control unit or the addressed device (or in both). Not-operational indicates that the control unit or the device is unable to recognize its address. No status is available with condition code 11.

Overlap Mode

In overlap mode (2020 Submodel 5 only), two instructions draw a condition code response: Start, and Test I/O. Start and Test I/O instructions set identical condition codes provided that data transfer is involved. Test I/O always involves data (status) transfer; the Start instruction, however, may call for a control operation that does not involve data transfer (control-immediate commands such as Rewind Tape) and, in this case, the condition code setting deviates.

The following codes are set:

00: For a Start instruction, condition code 00 indicates that a data transfer operation or the first command of a chain has been started successfully. For Test I/O, 00 indicates that the addressed control unit and device has a "clean" (all zero) status. Condition code 00 also indicates that interrupt will be requested (unless 00 is given in response to a Test I/O). Control-immediate operations cannot set 00, but set 01 under identical circumstances, i.e., when they were started successfully. However, if a control immediate is the first instruction of a chain and is accepted by the control unit, then 00 is set.

01: Indicates that the channel status word was updated and that the program should examine it to find out why it was stored because, with condition code 01, the status is always significant. For example, most control operations release the channel immediately after initiation; in that case channel end is in the status word, indicating that the channel is free for other operations (on other devices). For data transfer operations, condition code 01 indicates that the operation was not started due to a busy or interrupt-pending condition in the device.

10: Indicates that the channel is busy with a previously initiated operation. The channel appears also busy when it has an interrupt pending and this condition is not cleared.

11: Indicates that the instruction was rejected because of a not-operational condition in either the addressed control unit or the addressed device (or both). Not operational indicates the inability of the control unit to recognize its address because of power off, wrong address, etc. No status information is available when 11 is set.

Storing of Channel Status Word

Since the various conditions that may arise during the execution of a command are too numerous to be represented by the condition code alone, they are collected in the channel status word. This status word is located at main storage positions 156 through 161. (The same area is used also by the storage control to store its own status word.) The status word (or a portion of it) is updated when any of the following operations has been either executed, terminated prematurely, or rejected:

1. Read
2. Read Backward
3. Write
4. Control (See Note)
5. Test I/O
6. Sense

NOTE: For control-immediate operations, the status is updated twice, i.e., during initiation and at termination.

The status word is not updated for the transfer-in-channel operation and for any individual command in a chain; it is updated only for the last command of the chain. If a chained command is terminated prematurely or in error, it becomes the (actual) last one and status is stored to show the reason for the premature ending.

STRUCTURE OF CHANNEL STATUS WORD

The channel status word is six bytes long and contains the unit address, the unit status, the channel status, the residual count, and the command address (Figure 13). Of these parts, the unit status bits reflect the peculiarities of an individual device and its control unit to a large degree. The unit status bits affect the channel, but the design of the device determines which bits are used and under which circumstances they are generated.

The bits of the channel status word are assigned as follows:

Bits 0 - 7, Unit Address

Bits 0 - 7 represent the address of a unit; the term "unit" applies to a control unit as well as to a device in the sense that any low-order portion of this address may represent one of several devices that share a common control unit, while the remaining portion identifies the control unit. The total unit address must not be zero. The address is identical with the unit address in the Start or Test I/O instruction that caused the status to be stored. However, there is one exception; if a control unit signals 'control unit end' alone (to indicate that it has become available), it is allowed to present any address that it is capable of recognizing, regardless of whether a device is actually attached or not.

Unit Address	Unit Status	Chan Status	Residual Count	Address of last command Attempted
0 7	8 15	16 19	20 31	32 47

Figure 13. Channel Status Word 01955

Bits 8 - 15, Unit Status

Each bit within the unit status field represents a condition when it is 1 (turned on). These conditions are:

Attention, Bit 8: This signal is generated by the device when it detects an asynchronous condition that is significant to the program.

The device can present attention during the initiation of an operation or when the operation is completed but not while the operation is in progress. If attention is signaled during initiation, the operation is not started. If attention is signaled upon completion, chaining is stopped. The precise condition indicated by attention (if used at all) depends on the device. The attention signal initiates interrupt regardless of the channel mode when presented outside of the initial selection.

Status Modifier, Bit 9: This signal is generated by the device when the normal sequence of commands must be modified or when some condition prevents the unit from executing the command. The precise condition thus indicated (if used at all) depends on the device. Two meanings are assigned:

1. When the modifier appears together with busy, it indicates that the control unit is busy. 'Control unit busy' disrupts chaining.
2. If the status modifier appears together with device end, the next sequential command in a chain is skipped and the one following after is initiated, unless unusual conditions (such as attention) disrupt chaining.

   If chaining was not specified, the status modifier has no effect beyond interpreting some other condition.

Control Unit End, Bit 10: This signal indicates that the control unit has become available for another operation. However, only control units capable of servicing more than one I/O device will provide the signal.

Control unit end can be generated by a multiple-device control unit either when the unit has detected an unusual condition after it has signaled channel end, or when it is addressed while it is in the busy state. For example, the control unit signals 'channel end' when it has accepted a Backspace File command, then it starts the operation and becomes busy. If the operation cannot continue, the control unit signals 'control unit end' to avoid being tied up by the malfunctioning device. The unusual condition is also indicated.

If the control unit is addressed by a Start instruction while in the busy state, the control unit responds with busy, together with status modifier, and 'control unit end' is sent later, i.e., when the control unit becomes available. The 'control unit end' signal requests interrupt (regardless of the channel mode) when it is sent alone as in the foregoing case. If a busy control unit is addressed by a Test I/O instruction, the response is the same but control unit end will not follow.

Busy, Bit 11: Busy is presented only in response to a Start or Test I/O instruction or when a chain of commands is disrupted by the device signaling 'attention'. Busy alone pertains to the device; busy together with status modifier pertains to the control unit (if control unit busy, the state of the device is of no interest). Thus, busy indicates that the device or control unit is either occupied with a previously initiated operation or that status conditions exist which prevent successful execution of the attempted operation (for example, a pending interrupt).

Channel End, Bit 12: This signal indicates that the channel is free for another operation. Channel end is generated by the control unit or the device when the actual transfer of data or control information is completed.

The precise time at which channel end is generated depends on the design of the device. For data transfer operations, channel end and device end are usually presented together; however, some devices later verify the data and may delay channel end until verification is completed. With buffered devices, channel end is presented when the data transfer between channel and buffer is completed. For control operations, channel end is presented when the control information has been transferred, and device end is usually presented later (when the device has finished with the operation). For control operations which require a negligible execution time, channel end is delayed until the device has completed the operation, and channel end then appears with device end.

Device End, Bit 13: Device end indicates that the device is free for another operation; thus, the signal is generated either when an operation is completed at the device or when the device changes from the not-ready to the ready state (for example, due to a switch being operated manually). During command chaining, device end initiates the next sequential command (and is not made available to the program) unless unusual conditions are present. Device end is given only for the last command of a chain. When device end is sent after channel end, interrupt is requested in overlap and non-overlap modes.

Unit Check, Bit 14: This signal is generated by the device or the control unit to indicate that some type of programming or manual intervention is required. The precise meaning of unit check depends on the design of the device. Usually, unit check is a summary indication for a number of different conditions and is not necessarily an error. With unit check present, sense information is available and can be retrieved by means of a Sense command so that the actual condition causing unit check can be examined. If unit check alone is sent in response to a Start instruction, the device has not started the operation. Unit check disrupts a chain of commands. (See also "Sense Information" in this section.)

Unit Exception, Bit 15: Unit exception indicates an individual condition and needs no interpretation via sense information. The signal indicates a condition that cannot normally occur, its precise meaning depending on the design of the device. Unit exception, sent alone in response to a Start instruction, indicates that the device has not started any action on the command. Unit exception disrupts a chain of commands.

An example of unit exception is when an operation on a tape unit runs into the end of tape area; this cannot normally occur but, if it does, unit exception is signaled.

Channel Status, Bits 16 - 19

The status of the channel is represented by four bits, each of which indicates an individual condition when it is 1: channel data check, interface control check, incorrect length, and program check.

Channel Data Check, Bit 16: This signal indicates that the channel has detected a parity error (even parity) in the information transferred during a read, sense, or write operation. However, channel data check does not terminate the operation prematurely; the operation continues to completion. Data entered into main storage will have correct parity because the CPU forces odd parity on all data to be stored, but this is not necessarily true for the device. Channel data check disrupts chaining.

Interface Control Check, Bit 17: Indicates an invalid signal or sequence of signals on the I/O interface. Such invalid signals are detected by the channel and are usually caused by the device or its control unit. Interface control check is set due to either one or all of the following conditions:

1. The address or the status byte received from the device has invalid parity (even parity).
2. The device responded with an address other than the one specified in the Start or Test I/O instruction.
3. A signal from a device was sent at an invalid time or had invalid duration.
4. A control unit failed to respond with the appropriate tag line sequence (because of a hang up).
5. The channel control circuits detected an error (such as cycle steal control check in overlap mode).

The interface control check signal causes immediate termination of any currently running channel operation; chaining is suppressed. With interface control check present, the status information may be meaningless.

Incorrect Length, Bit 18: Incorrect length indicates that the number of bytes in the main storage data field specified for the particular I/O operation is not equal to the number of bytes requested from (or offered by) the I/O device, i.e., length count and data-field size do not match. However, only the following discrepancies cause incorrect length:

1. The device offers more bytes than were called for in a read (forward or backward) or in a sense operation. In this case, the data field is not overrun by the additional byte or bytes (the residual count is zero).
2. The device offers fewer bytes than were specified in any read or sense operation. (The residual count is not zero.)
3. The device terminates a write operation before the count has gone to zero, i.e., before all bytes specified have been transferred.

NOTE: If the device requests more data than was specified in a Write command, the operation ends when the count goes to zero and incorrect length is not indicated. Incorrect length disrupts chaining.

Program Check, Bit 19: This bit is set when the channel detects programming errors such as invalid data addresses or invalid units addresses, invalid command codes or invalid length counts. If the program check condition is sent in response to a Start or Test I/O instruction, the operation is not initiated. If program check conditions are detected during an already-running operation, the device is ordered to end the operation. Program check disrupts a chain of commands.

The following conditions lead to program check:

1. The unit address derived from the B2-D2 field of a Start or Test I/O (bits 40 - 47) is all zero.
2. The command word address specified in a Start or Test I/O instruction is outside the upper main-storage boundary.

3. The command code of the read-out command word has bits 0 - 7 as all zeros.
4. The length count in a control command or a data transfer type command (Read, Sense, Write) is zero. (For a Transfer in Channel command, the length count is allowed to be zero.)
5. The data address either is above the upper main-storage boundary to begin with or is increased (during execution) beyond the limit because of the length count. On reading backward, program check occurs when the data address decreases below 144, i.e., when the address runs into the protected main-storage area.
6. The command specified by the Start instruction is a TIC command, or a TIC command specifies a further TIC command.

Residual Count, Bits 20 - 31

The residual count bits indicate the number of bytes that were not transferred because of premature ending of the operation or because less bytes were offered than were requested. A residual count is provided for all data transfer operations when chaining is specified (in which case a residual count is provided only for the last command performed or attempted).

Command Address, Bits 32 - 47

The command address bits represent the leftmost main-storage location (the beginning) of the command word used, and allow the status received to be associated with this command. With chaining, the command address pertains to the last command issued.

INTERRUPT

Interrupt is requested to indicate to the program that an operation (or part of it) has ended and that status information relevant to the terminated operation is available. Interrupt is requested under different circumstances for non-overlap and overlap modes.

Interrupt for Non-Overlap Mode

Interrupt is requested only for those operations in which the termination is not obvious to the program. Thus, interrupt is requested whenever device end or control unit end occurs after channel end. Data transfer operations such as Read, Sense, or Write do not request interrupt because the program cannot proceed until they have ended. Control operations which do not occupy the device for significantly longer than they tie up the channel (short controls) do not request interrupt. Control operations (such as rewind tape) which tie up the device for considerably longer than they occupy the channel request interrupt when they are completed at the device; in this case, device end (and possibly control unit end) follows channel end.

If a device signals attention or any other condition outside of initial selection, interrupt is requested. Interrupts are enabled when the channel mask bit (bit 7) of the current PSW is 1. The channel interrupt is identified by channel address (7) being stored in the old PSW. All other information (device address, interrupt conditions, residual count, etc) is stored in the channel status word. If the channel mask bit is zero, all 2020 interrupts are disabled and remain pending.

Interrupt for Overlap Mode

In overlap mode (2020 Submodel 5 only), the program must be informed whenever any of the channel facilities becomes available because this cannot be predicted; however, if availability is indicated, new operations can be started while others are in progress. Interrupt is therefore requested each time the channel, a control unit, or a device (or all of them) becomes available. Thus any one or all of the following signals request interrupt: channel end, control unit end, device end, and attention. Interrupt is requested, however, only for operations that have been started successfully (as indicated by condition code 00 set in response to a Start instruction). If a Start instruction draws condition code 01 in response, interrupt is not requested because status information is already available and should be examined.

Data transfer operations (Read, Sense, Write) request interrupt when they end in normal or abnormal fashion. (Attention, for example, is an abnormal condition.)

Control operations which occupy the channel only for the time required to transfer the control information (control-immediates) request only one interrupt when device end occurs. Control operations which present channel end after initial selection (data transfer controls) request interrupt twice, i.e., when channel end occurs and, again, when device end occurs (unless both occur together). Control operations for which channel end is delayed until device end occurs, because the device-busy time exceeds the channel-busy time for a negligible degree (short controls), request only one interrupt.

The Transfer in Channel commands and Test I/O instructions do not request interrupt.

Interrupt is performed only when the channel mask in the current PSW is 1. The interrupt is identified by the device address 7 (channel) and by the overlap mode bit being stored in the old PSW. If the channel mask bit in the current PSW is zero, interrupts are disabled and remain pending.

Interrupt during Chaining

During a running chain of commands, no interrupt is requested for any command that is completed free of errors. Likewise, no status information is made available to the program for such commands. However, at the end of a chain, status is made available for the last command. (The last command is either the last one designated or the first command during which an error comes up.) Whether interrupt is requested for such a last command depends on two factors, i.e., the mode specified, and the type of command executed or attempted as the last one.

In Non-Overlap Mode

Interrupt is requested only if the last command was a control-immediate; the interrupt is requested when device end for this command is accepted. If the last command is a data transfer command, or if the chain is disrupted by an error condition, the only indication given is condition code 01 (or 11 in the case of an addressing error).

In Overlap Mode (2020 Submodel 5 only)

When a chain is started successfully, the condition code is set to 00 (even if the first command of the chain is a control-immediate) because, at the end of the chain, interrupt will always be requested regardless of the ending reason. If the last command is either a data transfer or a short control command, interrupt is requested when channel end and device end for this command is accepted.

If the last command is a control-immediate, interrupt is requested when channel end for this command is given and a second interrupt request follows when device end for this command is given.

PENDING CHANNEL INTERRUPTS

Pending interrupts are those that have been requested but cannot be performed because the channel mask bit in the current program status word is zero. Pending interrupts make busy the facility for which the interrupt was requested; this condition inhibits new operations on these facilities. Interrupt can be pending in:

1. The device. The condition pending is always device end.
2. The control unit. The condition pending can be control unit end, attention, or device end.

   Although device end is always generated by the device, it is nevertheless sometimes transferred to the control unit. The exact conditions under which device end involves also the control unit depend on the design of the control unit and on the type of operation performed. For example, operations that are performed under supervision of the control unit, such as Write Tape Mark or Forwardspace File, have the effect of a device end transfer to the control unit (which makes this unit busy).
3. The channel. This condition can be pending only when overlap mode was set; the condition is always a channel end signal which a control unit has sent to indicate that data or command transfer is completed.

   A device pending interrupt makes only the device busy. A control unit pending interrupt makes all devices on this unit busy. A channel pending interrupt makes the entire channel busy. Pending interrupts must therefore be cleared (or "worked off" by restoring the channel mask bit to 1) if operations are to continue on the respective facilities.

Clearing of Pending Interrupts

Pending interrupts are cleared by addressing the device (and thereby automatically the control unit) from which the interrupt request originated. This addressing is accomplished by different methods depending on the channel mode (overlap/non-overlap) specified.

In Overlap Mode (2020 Submodel 5 only): Pending interrupts are cleared by addressing the respective device with a Test I/O instruction. Test I/O obtains the status from the device; this action causes any interrupt conditions that may have been generated by this device or its control unit to be cleared. Test I/O does not start any actions beyond obtaining the status; the status will have no busy bit.

NOTE: A Start instruction that addresses a pending device has the same effect as described. However, if it addresses a "clean" device, an operation will be started. If a pending interrupt is cleared via Start, the busy bit in the status will be on.

In Non-Overlap Mode: Pending interrupts are cleared by addressing the pending device with a Start instruction. The Start instruction does not initiate an operation or interrupt but obtains only the status, whereby any pending interrupts that may have been generated by this device or its control unit are cleared. The busy bit in the status will be on.

Clearing Errors

Pending interrupts are not cleared when the Start or Test I/O instruction addresses a device other than the one that caused the pending interrupt. Different responses will be given, depending on how far the address is removed from the unit that requested the interrupt. The following cases are possible:

1. Interrupt pending in device only, and other device on the same control unit is addressed. The operation is started successfully (if Start is used) or a clean status is returned (if Test I/O is used). The interrupt is not cleared.
2. Interrupt pending in control unit, and a device on this control unit (but not the one that requested the interrupt) is addressed. The response is control unit busy; the interrupt is not cleared.
3. Interrupt pending in channel (can occur only in overlap mode). In this case two different responses can occur:
   a. If a device on the same control unit from which the channel end originated is addressed (but not the device that received the previous command and thereby caused channel end), then the response is control unit busy and the interrupt is not cleared.
   b. If another control unit is addressed, the condition code is set to 10 (channel busy); the interrupt is not cleared and no status will be available.

SENSE INFORMATION

A Sense command is issued by the program to obtain status information in addition to, or more detailed than, that which is contained in the status word. Although the type and amount of data received from a unit depends largely on its design, the first 6 bits in the first sense byte are assigned as follows:

Bit	Indication
0	Command Reject
1	Intervention Required
2	Bus-Out Check
3	Equipment Check
4	Data Check
5	Overrun

Command Reject, Bit 0: Indicates that a command has been received which the device is incapable of executing, either because of its design or because some condition prohibits execution (for example, writing on a file-protected tape).

Intervention Required: Indicates that some condition prevented the last operation before the Sense command from being executed, and that an intervention (for example, removing a jam, switching the unit to ready, refilling a hopper) is required to correct the situation.

Bus-Out Check: Indicates that the control unit or device has received one byte with incorrect (even) parity. If this byte was recording data, the write operation was not ended prematurely but continued to completion although faulty data may have been recorded at the device. If the byte was a command code or control information, the operation was not started.

Equipment Check: Indicates that a malfunction has been detected in the device or control unit. The precise meaning depends on the design of the device but, usually, a parity error or an unequal compare between checking circuits will set equipment check.

Data Check: Indicates that the device or control unit has detected a data error during reading or writing which is not covered by the bus-out check. The precise meaning depends on the design of the device but, usually, faults associated with the recording medium (weak signals, loss of a track, redundancy checks, etc) set data check. A read or write operation is not ended prematurely because of data check.

Overrun: Indicates that the time limit for a particular data transfer operation has been exceeded. During a read operation, overrun indicates that the channel has failed to accept a byte that was offered by the device; in this case, the data in main storage is shifted to fill the gap that would otherwise occur. Overrun can occur when the total activity initiated by the program exceeds the channel capability (especially during data transfer between non-buffered control units operating synchronous devices). Overrun occurs also when the new command is received too late during chaining.

IBM 2415 MAGNETIC TAPE UNIT AND CONTROL

INTRODUCTION

The IBM 2415 Magnetic Tape Unit and Control is designed to meet the low-cost magnetic tape requirements of System/360. Six models of the 2415 Magnetic Tape Unit and Control are available for the 2020: the 2415 Models 1, 2, and 3 use the Non-Return-to-Zero-IBM (NRZI) method of recording, and the 2415 Models 4, 5, and 6 use the Phase Encoding (PE) method. The 2020 operates with either NRZI- or PE-type tape drives through the input/output channel.

The user can select from three different models of either NRZI- or PE-type tape drives. The three different models consist of modules containing 2, 4, or 6 tape drives that operate independently but share a common control unit. Each tape drive can be addressed by the program, and has its own keys and indicator lights for manual control.

Table 10 shows the performance of the various models. The tape drives are normally equipped with nine-track read/write heads; however, seven-track heads can be installed. For example, a 2415 Model 2 may be equipped with two 9-track heads and two 7-track heads. The seven-track heads, however, require installation of the Seven Track Compatibility feature in the control unit of that model. (For details, see "Optional Features" in this section.)

All models can read backwards, but the operation is only error-free when the tape that is read backwards was written on a 2415, or on an IBM 2401/2402/2403/2404 Magnetic Tape Unit. Error detection is accomplished by immediate read-back after writing via the second gap of the read/write head. The PE-type tape units are equipped with the automatic in-flight error correction for a single track. The automatic correction becomes effective during read operations when the error is in only one track. All errors (single or multiple track) are detected on all models and, where automatic error correction is not possible, the "Input/Output Control System for Tapes" (an IBM-designed program) uses a special recovery procedure.

Table 10. IBM 2415 Characteristics 01956

	Model	Size	Density	Bytes/Sec	Tape Speed
NRZI (9-Track)	1	two tape drives	800 Bpi	15000	18.75 inch/sec
	2	four tape drives	800 Bpi	15000	18.75
	3	six tape drives	800 Bpi	15000	18.75
PE (9-Track)	4	two tape drives	1600 Bpi	30000	18.75
	5	four tape drives	1600 Bpi	30000	18.75
	6	six tape drives	1600 Bpi	30000	18.75
The 7-Track Compat. Feature for all Models allows densities of 200,556, and 800 Bpi NRZI					
The 9-Track Compat. Feature for Models 4,5,6 allows 800 Bpi NRZI Operation					

Input/Output Channel

The magnetic tape units communicate with the 2020 via the input/output channel. This channel allows the program to operate various different devices in a standardized manner independent of the peculiarities of these devices yet providing utmost flexibility. Since the programming techniques required for operating the tape units are determined by the functional characteristics of the IOC, the further descriptions are limited to the specific requirements of the 2415; details pertaining to the channel are described previously in this manual under "Input/Output Channel (for 2020 Submodels 2 and 5)".

Control Unit

Each individual tape drive represents an I/O device which is identified by an address. The tape drives communicate with the channel through a common control unit. Up to six tape drives share one control unit. The control unit receives the commands from the channel and initiates the required actions in the addressed drive. The control unit provides the channel with status information in response, to indicate to the program whether the operation can be executed and, if not, why it cannot be performed. The control unit also releases the channel as soon as its services are no longer required, and it indicates when an operation has ended and how it was terminated. The channel thus deals with the control unit because this unit "understands" the channel commands and interprets them for the devices (i.e., for the tape drives). The principles of communication between control units and channel are described under "Input/Output Channel (for 2020 Submodels 2 and 5)". These principles govern the operation of the 2415.

2415 COMMANDS

Tape unit operations are initiated when the address of a tape control unit is specified in bits 40 - 43 of the channel Start instruction; bits 44 - 47 then specify a tape drive. The actual addresses are determined by the user and are wired into the control unit at installation time. Details of the Start instruction are described under "Input/Output Channel (for 2020 Submodels 2 and 5)". If the Start instruction is accepted, the specified command word is read out from main storage. The 2415 is capable of performing the following data transfer operations:

Data Operation	Command Code (Bits 0-7 of CCW)
Read	m m m m m m 1 0
Read Backward	m m m m 1 1 0 0
Sense	m m m m 0 1 0 0
Write	m m m m m m 0 1

NOTE: The "m" positions are ignored by the 2415; they can be 0 or 1.

The 2415 is capable of performing the following control operations:

Control Operation	Command Code (Bits 0-7 of CCW)
Rewind	m m 0 0 0 1 1 1
Rewind/Unload	m m 0 0 1 1 1 1
Erase Gap	m m 0 1 0 1 1 1
Write Tape Mark	m m 0 1 1 1 1 1
Backspace Block	m m 1 0 0 1 1 1
Backspace File	m m 1 0 1 1 1 1
Forwardspace Block	m m 1 1 0 1 1 1
Forwardspace File	m m 1 1 1 1 1 1

NOTE: The "m" positions are ignored by the 2415; they may be 0 or 1.

To operate the two compatibility features that can be installed in the control unit, two groups of mode set commands are provided: Mode 1 Set and Mode 2 Set.

Mode 1 Set commands set the addressed tape drives to 200, 556, or 800 bits per inch (bpi) NRZI; at the same time, the commands set the drive to odd or even parity mode, data converter on or off, and translator on or off, depending on the setting of bits 2, 3, and 4 of the Mode 1 Set command. For example, if 200 bpi is specified, it is possible to set simultaneously odd parity, data converter off, and translator on by setting the bits 2, 3, and 4 of the command to 111 (Case H in Table 11).

Mode 2 Set commands set the tape drives to either 800 bpi density for NRZI operation or to 1,600 bpi density for PE operation (Table 12).

Data Transfer Commands

Read, Read Backward

The term "read" refers to data transfer from the device to main storage. The number of bytes to be transferred is specified in the length count (bits 36 - 47) of the command word. Data is stored beginning at the position defined by the data address (bits 16 - 31 of the command word), and continues in ascending order of address for the

Table 11. Function of Mode 1 Set Command 01194

Mode 1 Set

CCW Bits	0	1	2	3	4	5	6	7
Set 200 Bpi	0	0	-	-	-	0	1	1
Set 556 Bpi	0	1	-	-	-	0	1	1
Set 800 Bpi (Reset)	1	0	-	-	-	0	1	1

CCW Bits	2	3	4
Case A	0	0	0
B	0	0	1
C	0	1	0
D	0	1	1
E	1	0	0
F	1	0	1
G	1	1	0
H	1	1	1

Turn Off Translator	Turn On Translator	Turn Off Data Converter	Turn On Data Converter	Set Even Parity	Set Odd Parity	Set Density	Case	
							A	NOP 7 and 9 track (Reserved)
							B	Diagnostic Use (7 and 9 tr)
X			X		X	X	C	*(7-track only)
							D	Sense Status Reset (NOP)
X		X		X		X	E	(7-track only)
	X	X		X		X	F	(7-track only)
X		X			X	X	G	**(7-track only)
	X	X			X	X	H	(7-track only)

Note: 9-track operation overrides but does not reset 7-track mode setting.

All 9-track operations force odd parity. A Read Backward Command overrides a "Data Converter on" Mode Setting.

* Reset condition for control unit with 7-track and data conversion feature.

** Reset condition for control unit with 7-track but without data conversion feature.

Table 12. Function of Mode 2 Set Command 01957

Mode 2 Set Commands	0	1	2	3	4	5	6	7
	CCW Bits							
Set 1600 bpi	1	1	0	0	0	0	1	1
Set 800 bpi	1	1	0	0	1	0	1	1

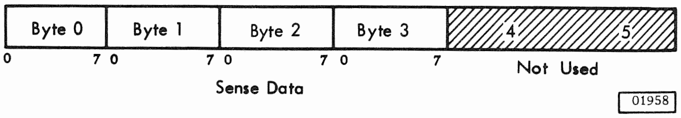
Read command, and in descending order of address for the Read Backward command.

NOTE: Read Backward requires de-skewed characters and, therefore, only tapes written on the 2415/2401/2402/2403/2404 can be successfully read backwards. Read Backward is terminated prematurely if the protected area (byte 143 and lower) is encountered, which can occur if data address and length count are ill-proportioned.

Status is stored when any read operation is terminated. If the operation was performed in overlap mode, interrupt is requested; if it was performed in non-overlap mode, interrupt is not requested. An exception to these rules is when a command specifies chaining. (See "Chaining" in this section.)

Sense

The sense operation is essentially a read operation except that data obtained from status indicators is read into main storage. The sense data is stored in ascending order of address and represents supplementary information about errors, checks, unusual conditions, etc. The 2415 provides four bytes of sense data; however, for compatibility reasons, six bytes should be specified in the length count of the sense command:



The sense data defines conditions that arose during the last operation attempted on the addressed tape drive. The sensed status indicators are reset with the next command issued after the Sense unless this is another Sense command. If Sense was performed in overlap mode, interrupt is requested upon termination; if performed in non-overlap mode, interrupt is not requested. (See "Chaining" for exception, and "Sense Data Description" and Table 13 for details.)

Write

The term "write" pertains to data transfer from main storage to a tape drive. Data is transferred in ascending order of address and recorded on the tape, unless the tape drive is in the file-protected state, in which case the Write command would be rejected with unit check and the sense bit file-protect would be on. During NRZI operations (write

Table 13. Sense Data Table 01959

Sense Data Table

Byte zero (all Models)		Byte two (all Models)	
Bits		Bits	
0 = Command Reject	Unit Check	0 = Not used	
1 = Intervention Required		1 = Not used	
2 = Bus - Out Check		2 = Not used	
3 = Equipment Check		3 = Not used	
4 = Data Check		4 = Not used	
5 = Overrun		5 = Not used	
6 = Word Count Zero		6 =	Unconditionally forced on
7 = Data Converter Check		7 =	
Byte one (Models 1,2,3)		Byte one (Models 4,5,6)	
Bits		Bits	
0 = Interblock Gap Noise		0 = Same (as for Models 1,2,3)	
1 = Tape Unit Status A		1 = Same	
2 = Tape Unit Status B		2 = Same	
3 = 7-Track		3 = Same	
4 = At Load Point →	Unit check during all operations except rewind or rew/unload	4 = Same	
5 = Write Status		5 = Same	
6 = File Protect		6 = Same	
7 = Not used		7 = Not Capable →	Unit Check
Byte three (Models 1,2,3)		Byte three (Models 4,5,6)	
Bits		Bits	
0 = Data Register VRC	Data Check	0 = Same (as for Models 1,2,3)	Data Check
1 = LRCR - Error		1 = Multiple Track Error/LRCR	
2 = Skew Error		2 - Same	
3 = Not used		3 = End Data Check	
4 = Read Register VRC		4 = Read Register VRC/Envelope Check	
5 = Not used		5 = 1600 Bpi	
6 = Backward Memory		6 = Same	
7 = C-compare →	Equipment Check	7 = Same →	Equipment Check

at densities up to and including 800 bpi), a minimum block of 18 bytes per command should be written to facilitate noise recognition. No minimum is required for PE operations (1,600 bpi). Data is written on the tape in the form of blocks. The data blocks are separated by interblock gaps that are created by the start inertia, as the tape requires some time to reach full speed. Interrupt is requested upon termination only if the operation was performed in overlap mode; otherwise interrupt is not requested. (See "Chaining" for exception.)

Control Commands

All control commands for the 2415 are of the immediate type, i.e., they cause the control unit to respond with channel end as soon as it has accepted the command. Device end follows later, when the device terminates the operation; at that time, interrupt is requested regardless of the mode specified. However, the mode set commands differ from this behavior in one respect; they cause the control unit to respond with both channel end and device end during initial selection. Therefore, interrupt is not requested for mode set commands, even when overlap mode is specified. (See "Chaining" for exception.)

Rewind

The Rewind command causes the selected tape drive to rewind to load point. The load point is a reflective marker attached to the tape at the beginning of the tape reel. The control unit responds to the Rewind command with channel end, after which the rewind operation is started. The channel end indicates primarily that the channel is free, however, the tape control unit is also free to accept another command (unless the rewinding drive is specified). When the load point is sensed by the photo cell of the drive, device end is signaled to indicate that the drive is available. This device end is maintained (stacked) in the drive until it is accepted by the channel. Interrupt is requested regardless of the channel mode; if interrupts are disabled, the tape drive appears busy until the interrupt is cleared.

Rewind and Unload

The Rewind and Unload command causes the selected tape drive to rewind the tape to load point, after which the tape is automatically unloaded. Channel end is signaled when the command is accepted by the control unit. After the rewind/unload disconnect time has elapsed, control unit end, device end, and unit check is signaled to indicate the available state of the control unit and the not-ready state of the tape drive. (The tape drive is busy during the rewind/unload operation and becomes not ready when it is unloaded.)

A subsequent manual loading of a new tape causes the tape drive to signal device end when the load point is sensed. This device end requests interrupt, regardless of channel mode, to indicate that the tape drive is ready. If this interrupt remains pending, only the (ready) tape drive appears busy; the control unit is available.

Erase Gap

The Erase Gap command moves the tape of the selected drive in a forward direction and erases the tape for a distance of approximately 3-1/2 in. (88,9 mm). Channel end is signaled immediately upon command acceptance, and device end is signaled when the tape is erased. If the erase operation runs into the end of tape area, unit exception is signaled. The device end signal (which indicates command completion) requests interrupt regardless of the channel mode.

Write Tape Mark

The Write Tape Mark command writes a tape mark after the load point delay has timed out. The tape moves in a forward direction. The tape mark for phase encoding is a block of 40 characters with transitions in all tracks except 1, 3, and 4. For nine-track NRZI and seven-track operations, the tape mark is a single character with an associated check character. The nine-track tape mark consists of a 1-bit in tracks 3, 6, and 7; all others are zero-bits. The seven-track tape mark consists of a 1-bit in tracks 4, 5, 6 and 7 with no bits in the other tracks. Device end is generated when the operation is completed, and requests interrupt regardless of channel mode.

Backspace Block

The Backspace Block command moves the tape in a backward direction to the next interblock gap or to the load point, whichever comes first. The tape unit remains in backward status until a forward-type command, such as Write, is received. No data is transferred during this operation. Device end is generated when the operation is completed. If the load point is encountered, unit check is set and a subsequent Sense command would show that the drive is "at load point". Device end requests interrupt; the preceding channel end does not.

NOTE: On all backward-type commands, the tape unit remains in the backward status until a command that requires forward status is received.

Backspace File

The Backspace File command moves the tape in a backward direction to the interblock gap beyond the next tape mark or to the load point, whichever comes first. No data is transferred and device end is generated when the operation is completed. This command does not cause unit exception but, if the load point is encountered, unit check is set. (A subsequent Sense command would then show that the drive is "at load point".) Device end requests interrupt; the preceding channel end does not.

Forwardspace Block

The Forwardspace Block command moves the tape forward to the next interblock gap without transferring data. Device end is generated when the operation is completed. If a tape mark is read, unit exception is set. Device end requests interrupt; the preceding channel end does not.

Forwardspace File

The Forwardspace File command moves the tape forward to the interblock gap beyond the next tape mark. No data is transferred and device end is generated when the operation is completed. Device end requests interrupt; the preceding channel end does not.

Mode 1 Set Commands

Mode 1 Set commands are used to set all seven-track drives of the addressed control unit to either 200, 556, or 800 bpi density for a seven-track tape; these density settings are valid for read and write operations. A particular density setting is retained for all seven-track drives until a new mode set command is issued for that control unit. The density can be changed at any time, independent of the position of the tape.

In addition to density setting, the data converter, the translator, and the parity can be set. When the mode set command is accepted, channel end and device end occur together. These signals do not request interrupt, regardless of the mode specified. (See "Chaining" for exception.)

The Mode 1 Set commands can be used for all models of 2415, provided that these units are equipped with the Seven Track Compatibility feature and one (or more) 7-track read/write heads. If issued to a control unit without the seven-track compatibility feature, no operation occurs.

The compatibility feature is needed to read seven-track tape; the 2415 Models 1, 2, and 3 normally read nine-track tape at 800 bpi NRZI, and the 2415 Models 4, 5, and 6 normally read nine-track tape at 1,600 bpi PE. When a seven-track tape is read on a 2415 Model 1, 2, or 3 that is equipped with nine-track heads only, or when a nine-track tape is read on a seven-track head, a data check occurs (Table 14). When a Mode 1 Set command specifies data converter on, and the seven-track compatibility feature is installed without a data converter, command reject is set.

Mode 2 Set Commands

The Mode 2 Set commands are used for the 2415 Models 4, 5, and 6 to change the recording density and method from 1,600 bpi PE to 800 bpi NRZI or vice versa. However, the Mode 2 Set commands are only effective when the magnetic tape unit is equipped with the Nine Track (800 BPI NRZI) Compatibility feature. If a nine-track tape (written at 800 bpi NRZI) is read on a 2415 Model 4, 5, or 6 that has no nine-track compatibility feature, unit check occurs and a subsequent Sense command would show that the not-capable bit is set. An attempt to read a seven-track tape on such a unit has the same effect (see Table 14). If a Mode 2 Set command is issued to a control unit without the nine-track compatibility feature, no operation occurs.

The Mode 2 Set commands pertain to all drives on the control unit; however, the mode is transferred

Table 14. Mode Setting Table 01197

Tape Used	Head on Selected Drive	Control Unit Feature	Tape Unit Model	Response
1600 bpi 9-track	9-track	None	4, 5, 6	OK
800 bpi 9-track	9-track	None	4, 5, 6	Not capable
800 bpi 7-track	9-track	None	4, 5, 6	Not capable
556 bpi 7-track	9-track	None	4, 5, 6	Not capable
200 bpi 7-track	9-track	None	4, 5, 6	Not capable
1600 bpi 9-track	9-track	7-track NRZI	4, 5, 6	OK
800 bpi 9-track	9-track	7-track NRZI	4, 5, 6	Not capable
800 bpi 7-track	9-track	7-track NRZI	4, 5, 6	Data check
800 bpi 7-track	7-track	7-track NRZI	4, 5, 6	OK
556 bpi 7-track	7-track	7-track NRZI	4, 5, 6	OK
200 bpi 7-track	7-track	7-track NRZI	4, 5, 6	OK
1600 bpi 9-track	7-track	7-track NRZI	4, 5, 6	Data check
1600 bpi 9-track	9-track	9-track NRZI	4, 5, 6	OK
800 bpi 9-track	9-track	9-track NRZI	4, 5, 6	OK
800 bpi 7-track	9-track	9-track NRZI	4, 5, 6	Data check
1600 bpi 9-track	9-track	None	1, 2, 3	Data check
800 bpi 9-track	9-track	None	1, 2, 3	OK
800 bpi 7-track	7-track	7-track NRZI	1, 2, 3	OK
556 bpi 7-track	7-track	7-track NRZI	1, 2, 3	OK
200 bpi 7-track	7-track	7-track NRZI	1, 2, 3	OK
200 bpi 7-track	9-track	7-track NRZI	1, 2, 3	Data check

only to a drive when its tape leaves from load point (for a write operation). Mode setting is required for write operations only; read operations adjust themselves to the correct mode by sensing the formatting information that is written on the tape immediately after load point. A coded burst of bytes after load point indicates that the data on the tape is recorded at 1,600 bpi. The absence of the burst of bytes indicates the data on the tape is recorded at 800 bpi, and the drive is set automatically to the 800 bpi mode.

Tape drives which have the tape positioned away from load point are not affected by the Mode 2 Set command until the tape has returned to load point and a write operation is started. The density setting is retained in the control unit until a new mode set command is given. However, a system reset or a selective reset results in a recording density setting of 1,600 bpi. Interrupt is not requested for a Mode 2 Set command, regardless of the channel mode specified. (See "Chaining" for exception.)

Interrupt

Interrupt is a branch from the normal flow of the program instructions to a special subroutine. This subroutine checks the cause of the interruption by examining the device address in the old program status word. If magnetic tape units caused the interrupt, the device address will be 7, which is the address of the input/output channel. An examination of the channel status word will show which tape unit or control unit requested interrupt and which conditions caused it. For details on interrupt and on the channel status word, see "Input/Output Channel (for 2020 Submodels 2 and 5)."

Only the unit status bits (described in the following text) in the channel status word reflect specific tape unit conditions; the rest of the status word contains the address of the tape control unit and device causing the interrupt, a residual count, and the address of the last command issued. The channel status portion of the command word is always zero, except in the case of channel-detected malfunctions or programming errors.

Unit Status Bits

Bit 8, Attention: This bit is not set by the 2415.

Bit 9, Status Modifier: The 2415 never sets this bit alone; the bit can only appear together with the busy bit (the meaning of which it modifies). The status modifier cannot appear together with device end, and thus no command in a chain can be skipped. The status modifier, together with the busy bit, indicates control unit busy. If the control unit busy response is given during initial selection, the unit has either an interrupt condition pending (from a previous operation on a device other than the presently-addressed one) or it is busy executing a command.

A typical example is the Backspace File command, which causes the control unit to remain busy after it has signaled channel end. A new command is not accepted in this situation and, therefore, the status modifier and the busy bit are presented. Both bits are also presented when the control unit is addressed while it is executing a command. When the status modifier and the busy bit are on, chaining is stopped.

Bit 10, Control Unit End: This bit is turned on when the control unit detected an unusual condition while busy but after the 'channel end' signal was accepted by the channel. Control unit end is also set when the control unit busy condition ends.

'Control unit end' requests interrupt, regardless of the channel mode (overlap/non-overlap) specified, provided the signal appears after the initial selection is completed.

Bit 11, Busy: Busy can occur only during an initial selection sequence. It indicates that the I/O device or control unit cannot execute a command because it is still executing a previously initiated command or because it has interrupt pending. Busy, alone, pertains to the tape drive; busy, together with status modifier, pertains to the control unit.

Bit 12, Channel End: This bit is turned on as soon as the services of the channel are no longer needed by the device, i.e., when the channel is released. With Read, Readbackward, Write, and Sense operations, the channel is released when the actual data transfer ends. With control operations (Forward-space File, Rewind, etc), the channel is released when the command is accepted by the tape control unit, i.e., during the initial selection. A channel end during initial selection will not request interrupt, except if this channel end is originated from the last command of a chain and overlap mode was specified. When channel end appears after the initial selection and overlap mode were set, interrupt is requested; if interrupts are disabled at that time, the interrupt is pending and thus the entire channel appears busy. Channel end does not request interrupt in non-overlap mode; instead, it sets condition code 01, reminding the program to check on the status word.

Bit 13, Device End: Device end indicates that the tape drive (identified by the unit address) is free for a new operation; device end is generated when a tape drive (device) has completed an operation. Data transfer operations (read, write) cause simultaneous generation of device end and channel end upon completion of the data transfer. The Mode Set commands also generate simultaneous channel end and device end.

The simultaneous presentation of channel end and device end does not generate a request for interrupt in non-overlap mode, but it does so in overlap mode (except during Mode Set commands). When device end appears outside of initial selection (i.e., after channel end), interrupt is requested regardless of the channel mode; this case occurs with all control operations (except during Mode Set commands). For example, channel end is set immediately after a Rewind command is accepted by the control unit; device end is set later when the load point is sensed by the tape drive, i.e., when the tape is rewound. If interrupts are disabled at that time, device end is maintained in the control unit and, thus, interrupt is pending.

Bit 14, Unit Check: This bit is used as a summary indication for nine individual error conditions, any or all of which can set the bit. The meaning of this bit can be obtained by a subsequent Sense command which transfers the sense bytes 0 and 1 to the CPU for inspection. (See Table 13.) For example, any read backward, backspace block, or backspace file operation that runs up to or into the load point causes unit check. Data that was received during such an operation is considered invalid, even when there is no separate data check.

The unit check bit is also set when a rewind/unload operation is completed at the control unit level. (This is no error.)

Bit 15, Unit Exception: This bit is used to indicate an unusual condition, such as a write tape mark, write, or erase gap operation that runs into the end-of-tape area. If that happens, the Tape Indicate light on the respective tape drive is turned on. Unit exception is also set when a tape mark is sensed during a read, read backward, forwardspace block, or backspace block operation.

Clearing of Pending Interrupts

A pending interrupt is cleared when the tape drive which has caused it is addressed by the Start instruction. The Start instruction will, in that case, not initiate a command but will only obtain the status. The busy bit will be on. If overlap mode is on, pending interrupts can be cleared via Start or Test I/O; if Test I/O is used for clearing, the busy bit will be off.

NOTE: If a Test I/O instruction addresses a unit which has a "clean" status and overlap mode is not on, Test I/O acts as a Start I/O instruction and an operation is initiated.

Chaining

In the preceding text under "2415 Commands", the response for each individual command is described, in particular, which status causes a request for interrupt. However, these rules apply only to individual commands; when commands are chained, interrupt is requested only under certain circumstances. The rules for unchained commands are as follows.

Non-Overlap Mode, No Chaining

1. The status 'channel end', either alone or together with device end, does not cause a request for interrupt.
2. When channel end has been accepted by the channel for a particular operation, then interrupt is requested as soon as device end for this operation is accepted.

Overlap Mode, No Chaining (2020 Submodel 5 only)

1. Any status condition that is presented at initial selection does not request interrupt.
2. Any status that is presented after initial selection causes interrupt (such as channel end and device end for a data transfer operation).
3. When channel end has been accepted for a particular command, then interrupt is requested as soon as device end for this operation is accepted.

When the chaining flag of any particular tape command is on, the response and interrupt rules change as follows:

1. When the first command of a chain is accepted by the tape control unit, no status is made available to the program until the chain breaks; this occurs either when an error is detected or when a command chaining flag is off.
2. When the chain "breaks", status is made available. This status pertains only to the last command that was either completed or attempted.

Whether interrupt is requested or not depends on the channel mode and on the type of command that was the last one of the chain, as follows.

Non-Overlap Mode, and Chaining

If the last command in a chain is a control-immediate, interrupt is requested when device end for this command is accepted; for data transfer commands, interrupt is not requested but, instead, the condition code is set to 01.

Overlap Mode, and Chaining (2020 Submodel 5 only)

1. If the first command of a chain is accepted by the control unit, the condition code is set to zero, regardless of whether this command is a control-immediate (which normally does not set this code), or a data transfer, or a Mode Set command.
2. If the last command in a chain is a control-immediate, interrupt is requested twice, i.e., when channel end, and again when device end, for this command is accepted by the channel.
3. If the last command is either a Mode Set or a data transfer command, only one interrupt is requested when channel end and device end for the command are accepted.

SENSE DATA DESCRIPTION

Sense Byte 0

Bit 0, Command Reject: This bit is set when a write, write tape mark, or erase operation addresses a file-protected tape. It is also set when a 'data converter on' mode command is sent to a seven-track tape on a control unit fitted with the seven-track compatibility feature without the Data Conversion special feature.

Bit 1, Intervention Required: This bit indicates that a condition exists in the addressed tape drive which requires manual intervention. This bit is also set if the addressed tape drive is not physically attached.

Bit 2, Bus Out Check: This bit indicates that the bus-out line has even parity which is an error.

Bit 3, Equipment Check: This bit indicates that either a counter check or a C-compare check has occurred. If the C-compare check is not present, the equipment check was caused by a counter check.

Bit 4, Data Check: This bit is a summary indication for data register VRC, LRCR, skew, read register VRC, multiple-track, and end-data checks. Any or all of these conditions result in a data check.

Bit 5, Overrun: Overrun can occur only during a read, read backward, or write operation. The check bit is set when the time limit for data transfer operations is exceeded. (See timing in Appendix H.)

Bit 6, Word Count Zero: This bit is set when the channel stops a write operation before the first character was transferred.

Bit 7, Data Converter Check: During a read operation with data conversion (seven-track), four tape characters are normally transferred as three bytes. When less than four characters are read and the packing effect cannot be achieved due to the structure of the tape character, the packing byte is filled with zeros and the data converter check is set. The check allows the program to take corrective action. (See "Optional Features" in this section.)

Sense Byte 1

Bit 0, Interblock Gap Noise: This bit is set when bits are found in the gap between the data block during a write, write tape mark, or erase gap operation. This condition causes neither a data check nor a unit check.

Bits 1 and 2, Tape Unit Status A and B: These two bits reflect the status of the addressed tape unit and decide on the response to an initial selection. When both bits are zero, the unit check is set because the tape unit is assumed to be non-existent (power off). A off, B on indicates that the device is not ready (which causes unit check when the device is addressed). A on, B off indicates that the unit is ready and not rewinding and, therefore, a 'clear status' is sent in response to an initial selection.

Bit 3, Seven-Track: This bit indicates that the selected tape drive is a seven-track unit.

Bit 4, At Load Point: This bit indicates that the tape in the selected unit ran into the load point. This condition causes a unit check during all operations except during rewind. (Any forward command that starts from load point will not set unit check.)

Bit 5, Write Status: This bit indicates that the selected tape drive is in write status.

Bit 6, File Protect: This bit indicates that the tape in the selected unit is protected from writing or erasing (because the file-protection ring is off the reel).

Bit 7, Not Capable: This bit is used only when PE-type tape units are attached (2415 Model 4, 5, or 6). It is further used only during read or read backward operations. The not-capable bit is set when the following conditions exist:

1. The tape is read on a unit equipped with a nine-track head.
2. No identification burst is detected (either because 800 bpi is specified or because the tape is faulty).

3. The control unit has no compatibility feature that can take care of 800 bpi, nine-track (NRZI) operations.

Sense Byte 2

Bits 6 and 7: Only the bits 6 and 7 of sense byte 2 are used. Both bits are always on because they are forced unconditionally for compatibility reasons. The bits simulate a multiple track error that forces a program (which tests these bits) to automatically repeat the previous operation.

Sense Byte 3

Bit 0, Data Register VRC: This bit indicates a Vertical Redundancy Check (VRC). This means that data with faulty parity was set into the data register during a read or read backward operation.

Bit 1, LRC Error/Multiple Track Error: This bit has a dual function. It indicates a Longitudinal Redundancy Check (LRCR) for NRZI operations and multiple track errors for PE operations.

1. LRCR Error. This bit indicates that the longitudinal parity of the record read does not agree with the check character of that record. The LRCR check monitors write, write tape mark, read, and read backward operations.
2. Multiple Track Error. This bit is set during read operations when more than one track has dropped below the acceptable amplitude for reading. Data is not corrected and therefore, the program must attempt the read operation again. The multiple track error may or may not be accompanied by a vertical redundancy check.

NOTE: If only one track contains an error, the automatic in-flight error correction sends the correct data to the channel without stopping the tape.

Bit 2, Skew Error: This bit indicates that excessive skew has been encountered during a write or write tape mark operation. Skew errors are detected because the two-gap read/write head in the 2415 allows reading immediately after writing for check purposes.

Bit 3, End Data Check: This bit is turned on when the synchronizing burst which follows the data is not recognized (2415 Model 4, 5, or 6 only).

Bit 4, Read Register VRC/Envelope Check: This bit has a dual function. It indicates a vertical redundancy check for NRZI operations or an envelope check for PE operations.

1. Read Register VRC. This indicates that a character with incorrect parity has been read into the read register. The check is effective during write operations.
2. Envelope Check. This check is set during write operations to indicate that at least one track has failed to reach an acceptable amplitude. If this check bit is on, the write operation must be repeated.

NOTE: The data register VRC and the read register VRC are both suppressed after a stop or overrun. Data checks occurring before stop or overrun are registered in the unit status byte as unit check. The LRCR check, however, remains active at all times.

Bit 5, 1600 BPI: This bit indicates that the selected unit has a nine-track read/write head and is not set to any NRZI mode (2415 Model 4, 5, or 6 only).

Bit 6, Backward Memory: This bit indicates that the tape unit is in the backward state from a previous backward operation.

Bit 7, C-Compare: This bit indicates that a parity difference was recognized during either of the following operating modes:

1. Data transfer in nine-track mode or data transfer in seven-track mode with data translator off. (See Appendix M.)
2. Data transfer in data converter mode. The parity difference exists between the three eight-bit bytes and the four six-bit bytes (parity over 24 bits). See "Data Conversion Special Feature" in the following text.

FEATURES

Nine Track Compatibility Special Feature (for 2415 Models 4, 5, and 6 only)

The Nine Track (800 BPI NRZI) Compatibility feature permits a PE-type tape unit to read and write nine-track tape at 800 bpi using the NRZI method of recording. The change from the normal 1,600-bpi PE mode to the 800-bpi NRZI mode is achieved by means of the respective Mode 2 Set command. The Mode 2 Set command affects write operations only. Read operations adjust themselves automatically to the correct mode. All mode setting occurs only when the tape is at load point. A coded burst of bytes on the tape, located between the load point and the first data block, identifies the 1,600-bpi PE mode for read operations. The absence of this burst identifies the 800-bpi NRZI mode. The mode setting for write operations is determined by commands.

Read backward is possible only with tapes generated on the 2415/2401/2402/2403/2404.

Seven Track Compatibility Special Feature (for 2415 Models 1 through 6)

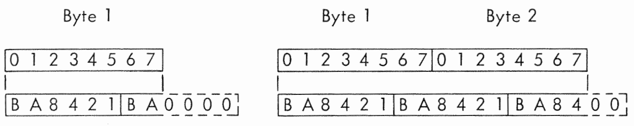
The Seven Track Compatibility feature permits the tape unit to read or write seven-track tape compatible with the current IBM 729/7330 Magnetic Tape Units. The feature provides operation at densities of 200, 556, or 800 bpi, in even or odd redundancy mode with 0.725 in. (18,4 mm) nominal interblock gap using NRZI recording format. A part of this feature is a data translator which translates EBCDIC eight-bit code to BCD six-bit code or vice versa. (See data translator conversion tables in Appendix M.) The seven-track feature may not be used with read backward unless the tape being read was recorded on one of the following tape units - 2415/2401/2402/2403/2404.

Data Conversion Special Feature

The Data Conversion feature expands the capability of the seven-track compatibility feature. With the data converter on, it is possible to write any System/360 data (entire EBCDIC) on seven-track tape with maximum packing efficiency. Data conversion cannot be used, however, during read backward operations.

Eight-bit bytes are written on seven-track tape as six-bit characters with odd parity, in the following manner.

A single byte is converted into two characters, with the four low-order bits of the second character automatically written as zeros. Two bytes are converted to three characters, whereby the two low-order bits of the third character are automatically written as zeros. Three bytes are converted into four characters without filler zeros. When a converted tape is read with the data converter on, the bytes are reassembled in the original form, whereby any filler zeros are automatically removed.

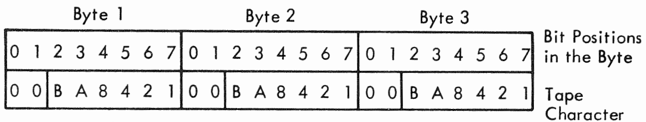


Write Operation with Data Converter On

01196

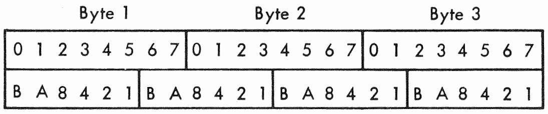
If a seven-track tape was written on another system (for example, IBM 1401 Processing Unit), reading is as follows.

During a read operation with the data converter off, every tape character (BCD) is set into the low-order six bit positions of the byte and the high-order two bits are set to zeros. During a read operation with the data converter on, four tape characters are packed into three bytes.



Tape Characters in Bytes with Data Converter Off.

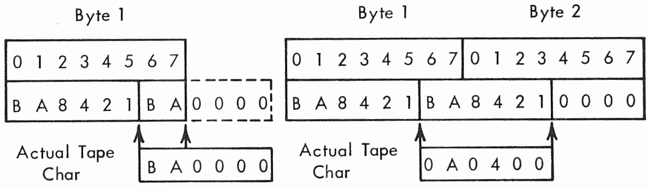
01960



Tape Characters in Bytes with Data Converter On.

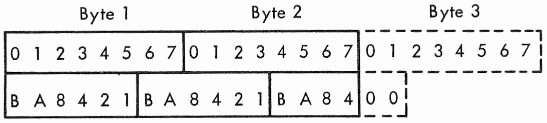
01961

Whenever less than four characters are transferred, a special packing arrangement is employed. For example, two tape characters are packed into one byte if the character bits 8, 4, 2, 1 are all zeros. However, when the bits 8, 4, 2, 1 are not all zero, the two characters are set into two bytes with the bits 4, 5, 6, 7 of the second byte set to zero and the data converter check turned on.



01962

When successfully packed characters are read back, zeros must be added to retrieve the original character. The data converter check indicates that zeros must be deleted to obtain the original character. Three tape characters are packed into two bytes if bits 2 and 1 of the third character are zero; if they are not, a third byte is used with the bits 2, 3, 4, 5, 6, and 7 set to zero. The data conversion check is set in the latter case to indicate that zeros must be deleted to obtain the original character bit configuration. The same holds true for one tape character because it also does not fill one byte completely. One tape character is transmitted as one byte with the bits 6 and 7 of that byte set to zero and the data converter check turned on.



01963

Retrieval (write back) of the packed tape characters is accomplished as shown in Table 15.

Table 15. Retrieval of Packed Tape Characters 01964

Input Record Length in Bytes	Last Bit from Tape resides in Mainstorage		Number of Zeros to be attached or deleted
	Byte	as Bit	
1	2	3	Attach Four
1 with Data Conv Check	1	5	Delete Two
2	3	1	Attach Two
2 with Data Conv Check	2	3	Delete Four
0 or 3	3	7	No Action
0 or 3 with Data Conv Check	3	1	Delete Six

STORAGE CONTROL FEATURE AND IBM 2311 DISK STORAGE DRIVE

INTRODUCTION

The Storage Control feature consists of an electronic attachment, housed within the 2020, which allows IBM 2311 Disk Storage Drives to be connected and operated under stored program control.

The 2311 Disk Storage Drive, with easily changed IBM 1316 Disk Packs, affords the user the programming flexibility of large on-line storage capacity and virtually unlimited off-line storage capacity. As with magnetic tape, an almost-unlimited volume of data can be written on 1316 Disk Packs, detached from the 2311, and stored off-line until needed.

Two models of the 2311 (Models 11 and 12) are available, and both use the 1316 as the storage medium. The two models differ only in the number of track positions available; each track position consists of a stack of ten disk surfaces and read/write heads which form a "cylinder" of data, as defined later; 200 cylinders of data are available in the 2311 Model 11 and 100 cylinders in the Model 12.

Compatibility between the two models is limited to the extent that the 2311 Model 11 reads and writes over the 200 cylinders on the disk pack whereas the Model 12 reads and writes on only the outer 100 cylinders.

The storage control feature is available for 2020 Submodels 2, 4, and 5; however, 2311 Models 11 and 12 may be mixed on storage control units fitted to the 2020 Submodel 5 only. The following disk storage drives can be connected to the CPU:

2020 Submodel 2: up to two 2311's (either two Models 11 or two Models 12).

2020 Submodel 4: up to two 2311's (Model 12 only).

2020 Submodel 5: up to four 2311's (Models 11 and 12; may be intermixed).

IBM 2311 DISK STORAGE DRIVE

Characteristics of the 2311 are shown in Table 16.

Table 16. IBM 2311 Disk Storage Drive Characteristics 01965

	2311 Model 11	2311 Model 12
Storage Capacity	5.4 Million Bytes	2.7 Million Bytes
Data Transfer Rate	156,000 Bytes/Sec	156,000 Bytes/Sec
Data Bytes/Sector	270 Bytes	270 Bytes
Sectors/Track	10	10
Data Bytes/Track	2700 Bytes	2700 Bytes
Data Bytes/Cylinder	27,000 Bytes	27,000 Bytes
Data Cylinders	200	100
Alternate Cylinders	3	3
Maximum Access Time	135 ms	90 ms
Average Access Time	75 ms	60 ms
Minimum Access Time	25 ms	25 ms
Disk Rotation Time	25 ms	25 ms

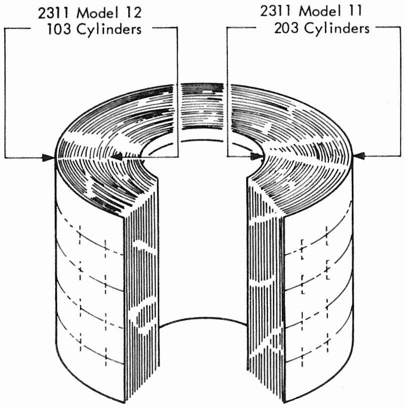


Figure 14. Cylinder Concept 01740

Storage Capacity

Storage capacity for the 2311 Model 11 is 5.4 million bytes. It is based on the use of 200 of the total 203 cylinders (Figure 14) for data and 3 cylinders for use as alternates (spares for use in the event of surface damage to a data track). Storage capacity for the 2311 model 12 is 2.7 million bytes. It is based on the use of 100 of the total 103 cylinders for data and 3 cylinders for alternates.

Storage Medium (1316 Disk Pack)

Each disk pack consists of six 14-inch disks, mounted 1/2 inch apart on a central hub. Data is recorded on the inside ten disk surfaces. The two outer surfaces are covered by protective plates. The entire assembly of disks, hub, and protective plates is rotated at 2,400 revolutions per minute (25 milliseconds per revolution). Each disk pack weighs about 10 lb (4,54 kg).

The two-piece plastic cover is designed to protect the disks against damage. A built-in handle on the top cover makes carrying easy and efficient. A self-locking device in the handle permits removal of the top cover only when the pack is mounted on the disk storage drive.

Access Mechanism

When the disk pack is mounted in the 2311, information is written on and read from the ten disk surfaces by magnetic read/write heads. These ten read/write heads are mounted in pairs between each two disks on a movable comb-like access mechanism. When in operation, the read/write heads float over the disk surfaces on a thin film of air.

The 2311 Model 11 access mechanism can be moved horizontally to any of 203 discrete track positions; the 2311 Model 12 can be moved to 103 track positions. Since all ten read/write heads are moved together by the access mechanism, a vertical cylinder of ten data tracks is formed at each track position.

Access Time

Cylinder-to-cylinder (horizontal) access time varies according to the number of cylinders traversed. Access time from a cylinder to an adjacent cylinder is 25 milliseconds (ms). For the 2311 Model 11, the maximum access time (from cylinder 202 to cylinder 000) is 135 ms, and the average time for "random" accesses is approximately 75 ms. For the 2311 Model 12, the maximum access time (from cylinder 102 to cylinder 000) is 90 ms, and the average time for random accesses is approximately 60 ms (Figure 15).

Once the access mechanism has reached a cylinder position, additional time is required for disk rotation to the desired record. At 2,400 revolutions per minute, rotation time is 25 ms, and one-half revolution (12.5 ms) is the average rotational delay.

Figure 15 shows the approximate time for access mechanism movement in either direction, excluding rotational delay. This may be used as an aid in programming for the most efficient utilization of the storage device.

Operator Controls and Indicators

Start/Stop Switch: This switch is on when it is in the Start position.

With the 2311 attached to a System/360 Model 20, this switch is pressed to the Start position to supply power to the disk drive motor and other 2311 components. When the disk drive motor has come to speed, and other components are ready for operation, the read/write heads are moved into position and the access mechanism performs an automatic seek cycle.

Pressing the Start/Stop switch to the Stop position causes the access mechanism to retract from the disk pack and removes power from the disk drive

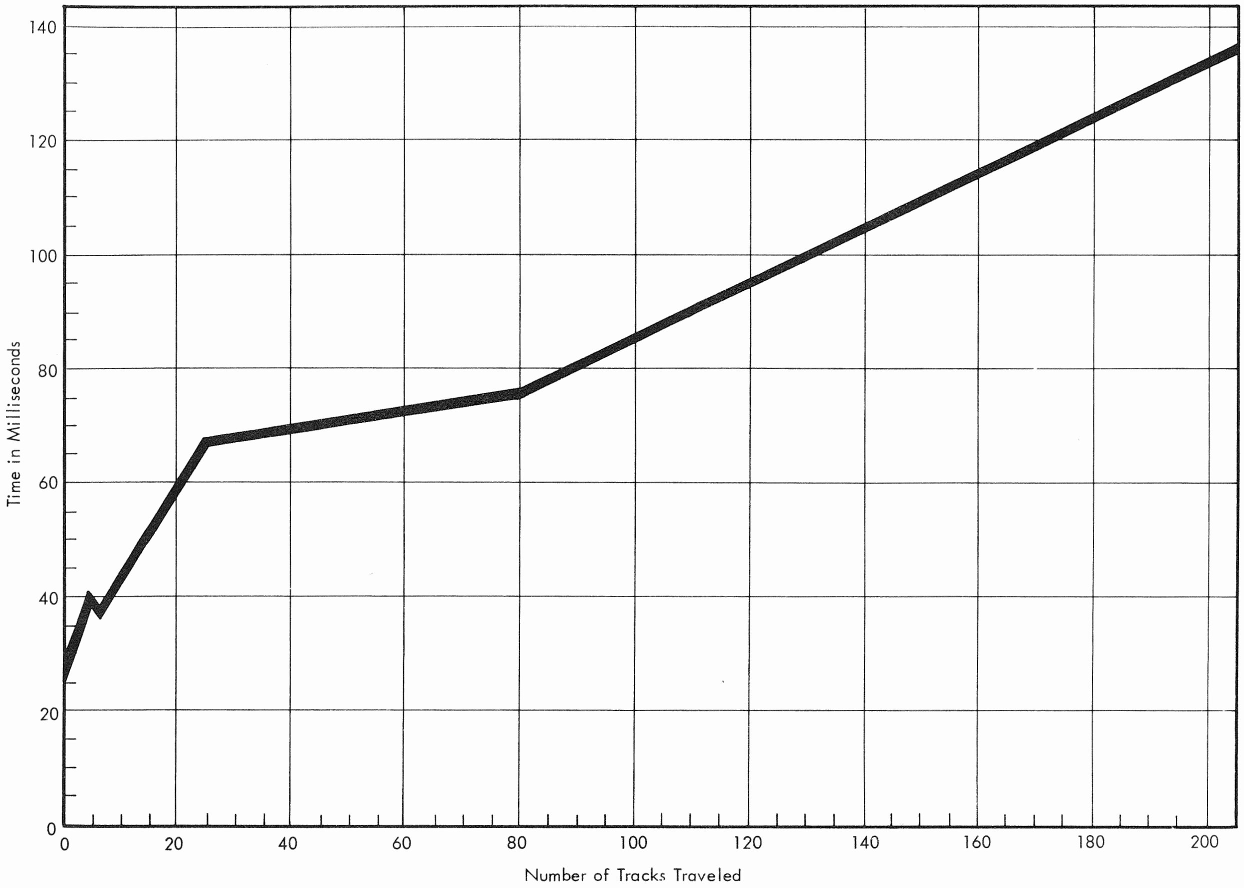


Figure 15. Access Time 01966

motor. Automatic braking stops disk pack rotation in a few seconds. Approximately 2 minutes are required from the time the switch is set to Stop until a disk pack can be removed, replaced, and ready for operation.

Select Lock Indicator: When on, this indicates a machine condition which requires the attention of a customer engineer. This condition causes the disk storage drive to be disabled and stops the usage meter.

Enable/Disable Switch: When the CPU is in the stopped state, this switch enables or disables the communication of the storage drive with the CPU. It also enables or disables the 2311 usage meter.

If the CPU is running when the switch setting is changed, the storage drive and usage meter operating status are not changed until the CPU is placed in the stopped state. (See also "Select Lock Indicator.")

Disk Pack Handling

The disk pack is protected in transit by special carton inserts and special protective material.

When received, examine the carton closely. If its condition is acceptable, remove the disk pack and store it. Keep the carton and inserts; they may be needed later.

If the carton or its contents show any unusual shipping damage, do not use the disk pack. Retain the damaged carton and disk pack in its "as received" condition and notify a customer engineer immediately.

Disk packs have been designed for ease of transport from location to location. For best shipping results:

1. Be sure the pack is secure in its two-piece cover.
2. Use only the specially designed IBM shipping carton with its special protective padding properly inserted. If the original carton is worn or damaged, a new carton may be ordered from an IBM office.

Handle a disk pack only with its cover on.

If the disk pack is accidentally dropped, or receives a sharp impact of any kind, call a customer engineer before using it.

Disk Pack Labeling: For positive identification, small adhesive-backed labels can be placed on the disk pack center hub. Labels in this location can be read through the transparent disk pack cover.

The following operating procedures should be followed when labeling disk packs:

1. Use adhesive-backed labels which can be applied and removed easily.
2. Use a writing implement, such as a pen or felt-tip marker, which does not produce loose residue. Do not use a lead pencil.
3. Write on the label before it is applied to the disk pack.
4. Place the label only on the center hub, not on the disk pack cover or top disk surface.
5. Use a new label if changes are necessary. Never use an eraser, because microscopic eraser particles can damage disk surfaces and read/write heads.

Disk Pack Loading and Unloading: Follow these procedures for rapid, effective disk pack changing.

Loading:

1. Open the 2311 cover.
2. Remove the bottom disk pack cover by turning the bottom locking knob.
3. Place the disk pack (still contained in the top cover) on the 2311 spindle.
4. Turn the top cover in the direction of ON arrow until firm resistance is met.
5. Lift the top cover from the disk pack.
6. Close the 2311 cover.
7. Press the 2311 Start key.
8. Reassemble the top and bottom covers of the disk pack to keep the interior clean.
9. Store the covers in a clean cabinet or on a clean shelf.

CAUTION:
Do not leave the disk pack top cover inside the disk drive.

Unloading:

1. Press the 2311 Stop key.
2. Wait for the disk pack to stop rotating.
3. Separate the top and bottom disk pack covers.
4. Open the 2311 cover.
5. If necessary, move the brushes gently clear of the disk pack.
6. Place the disk pack top cover over the disk pack.
7. Turn the top cover, in the direction of the OFF arrow, at least two full turns.
8. Lift the top cover, now containing the disk pack, from the spindle.
9. Fasten the bottom cover firmly to the disk pack.
10. Close the 2311 cover.
11. Store the disk pack in a clean cabinet or on a clean shelf.

Disk Pack Storage: To ensure maximum disk pack life and reliability:

1. Store the disk packs flat, not on edge.
2. Each pack should rest on a shelf, not on another disk pack.
3. Store in a clean, enclosed metal cabinet or a

similar fire-resistant container, never in direct sunlight.

4. Store disk packs in a machine-room atmosphere, 60° F to 90° F (15,6° C to 32,2° C), 10% to 80% humidity.
5. If disk packs must be stored in a different environment, allow two hours for adjustment to machine room atmosphere before use.

DATA TRANSFER

Information is transferred between the CPU and the storage control feature one byte at a time (in parallel); in the 2020 Submodel 5, however, data transfer is generally one halfword at a time. Information is transferred between the storage control feature and the selected 2311 one bit at a time (in serial). Thus, the storage control feature translates serial data from disk storage into parallel data for internal CPU processing, and translates parallel data from the CPU into serial data for entry into disk storage (Figure 16).

Data Records

A group of related bytes is called a "field". A series of related fields is called a "record." The data length of each record is fixed at 270 bytes. If the actual data length to be written in a record is less than 270 bytes, it is the responsibility of the programmer to add sufficient zeros or fill characters to make the data length equal to 270 bytes.

Data Checking

Storage Control Feature (Cyclic Check)

In direct access storage devices controlled by the 2020 storage control feature, data is stored and retrieved in areas. Checking accuracy in data transfer (to or from attached storage) is done by associating two check bytes with each area.

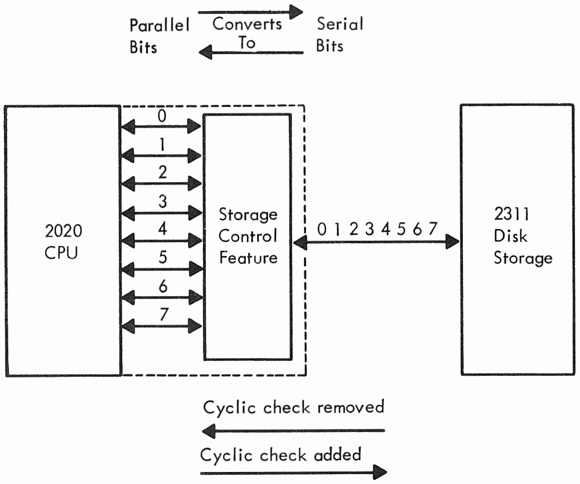
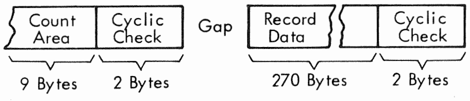


Figure 16. Data Transfer 01967



01968

As data is transferred from the CPU to an attached storage device, the storage control computes two cyclic check bytes which are added to the end of each area. The two cyclic check bytes are arithmetically coded to represent the data in the associated area.

During a transfer from a storage device, all areas read are inspected by the storage control feature. Cyclic check bytes are recalculated for each area and compared with those retrieved from storage. An unequal comparison will set data check error indicators.

The cyclic check code detects the following types of errors:

1. All errors occurring within a 16-bit span.
2. All errors involving an odd number of bits over any span.

DATA RECORDING

The surface of each disk in the 1316 has a uniform oxide coating but data is recorded on only one of the 203 discrete tracks, by precise positioning of the read/write head. The ten tracks thus formed simultaneously on the ten disk surfaces provide the 2311 with a total of 203 cylinders for recording data.

The basic unit of information recorded is the eight-bit byte. A byte is transferred in parallel from the CPU main storage to the storage control, thence serially to the disk storage drive, and

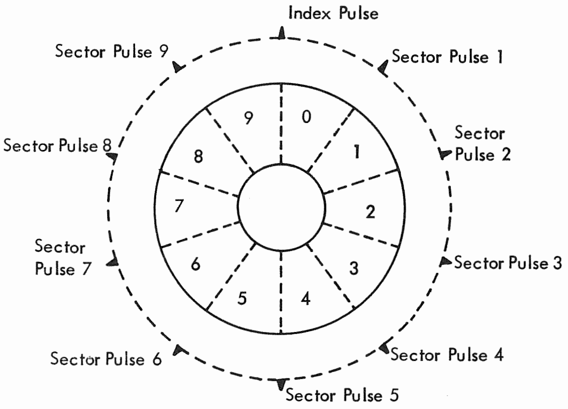


Figure 17. Sector Division of Disk 01969

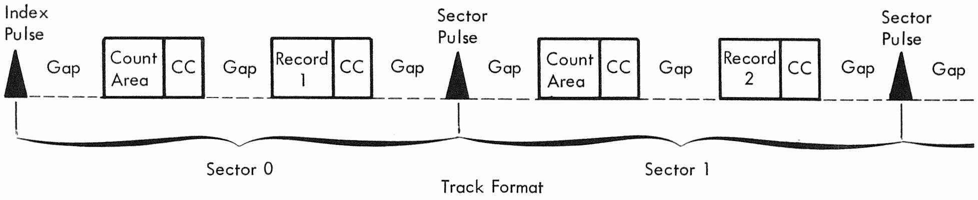


Figure 18. Track Format 01970

recorded on the track bit by bit. Parity bits are not recorded. In the reverse operation, bits are read from a track serially and are reassembled to an eight-bit byte by the storage control, any required parity bit being added before the byte is stored in main storage.

TRACK FORMAT

Each track represents a circular recording area that is divided into ten equal portions termed sectors (Figure 17). The first sector (sector 0) begins at the index pulse and ends at sector pulse 1 (Figure 18). Sector 1 is bounded by sector pulses 1 and 2, and so on, to sector 9 (bounded by sector pulse 9 and the index pulse). The index pulse marks the beginning and end of the track. This description of sector division applies to each of the 203 tracks on each recording surface.

The Sector

The area that comprises one sector is divided into two portions, the count area and the data area. The count area contains all the information required to properly identify the data area; the data area contains the actual data file.

Count Area

The count area begins at the sector (or index) pulse, after which a gap is created. Following the gap, a one-byte flag field, a five-byte identifier field, a one-byte key length, a two-byte data length and a two-byte cyclic check field are recorded (Figure 19). These fields contain numbers in binary notation and are used as follows:

Byte 1, Flag: The flag byte indicates the condition of the track in the following manner:

0000 0000 = Original good track
0000 0010 = Defective track
0000 0001 = Alternate track

NOTE: When 'alternate track' is signaled, the automatic head selection from the count area in main storage (for data commands) is suppressed.

It is the responsibility of the programmer to make certain that all ten count areas of any one particular track have identical flag bytes.

Bytes 2-6, Identifier: The identifier is a program-generated number that identifies the cylinder number, the head number and the record number so that the data in the data area can be retrieved or altered. The identifier is five bytes long and is organized as follows:

Bytes 2 and 3, Cylinder Number: Byte 2 is not checked but must be zero for System/360 compatibility and future expansion. Byte 3 may contain any number from zero through 202 (for a 2311 Model 11) or from zero through 102 (for a Model 12).

Bytes 4 and 5, Head Number: Byte 4 is not checked but must be zero for compatibility and future expansion. Byte 5 may contain any number from zero through 9.

Byte 6, Record Number: Byte 6 may contain any number from zero through 9.

Byte 7, Key Length: The key length byte in the count area is not used by the 2020 storage control and is not checked; however, the key length must be zero for compatibility and future expansion.

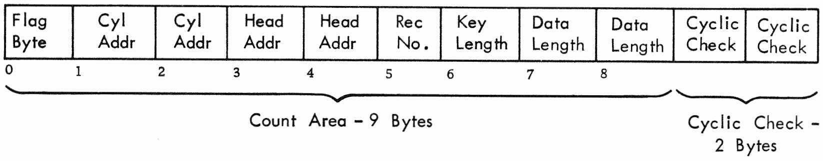


Figure 19. Count Area 01971

Bytes 8 and 9, Data Length: These two bytes are not checked; however, they must always contain the binary equivalent of the decimal number 270 because the data area cannot contain less than 270 bytes.

Data Area

A gap separates the count area of the sector from the data area. The data area is 272 bytes long, 270 bytes containing the actual data to be stored and the other two bytes containing a cyclic check character which is an automatically generated checking polynominal. The data located in the 270-byte area is termed the "record." The record can never be smaller than 270 bytes; if the programmer desires to store less useful information, he must load the remaining bytes with filler characters; the filler characters are not defined and may be any valid EBCDIC bit configuration.

The data area completes the sector. Since there are ten sectors to a track, ten count areas and ten data areas are provided on each track.

STORAGE CONTROL OPERATION

The storage control feature operates essentially in the same way as the input/output channel. Action on disk storage drives is initiated by a Start instruction that addresses the storage control, a specific 2311, and a command word which specifies the operation to be performed. Depending on the state of the addressed 2311, status information will be returned and stored in a storage control status word which can be examined by the program. If the addressed 2311 is in the available state, the command word is fetched from main storage and the operation is started.

Disk-storage operations can be performed in either of two modes, non-overlap or overlap. In non-overlap mode, operations can be started only when no time-shared I/O or processing operations are in progress, except buffered print operations (1403 Printer) or control-immediate commands (input/output channel). Once a disk-storage operation has started in non-overlap mode, the entire CPU is occupied throughout the execution of the operation.

In overlap mode (2020 Submodel 5 only), disk-storage operations can be started at any time (provided the storage control is not busy), and the CPU is released to proceed with the next instruction immediately after initiation of the disk-storage command.

Start Instruction

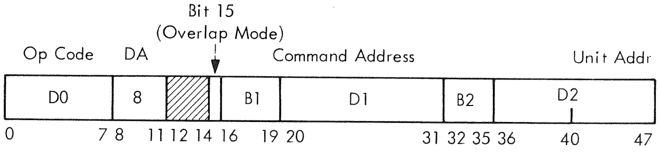


Figure 20. Storage Control Start or Test I/O Instruction 01072

The Start instruction (Figure 20) is six bytes long and contains a fixed op code, a device address, a mode bit, a main-storage address that specifies the location of a command word, and a unit (disk storage drive) address. The bit assignment is as follows:

Bits 0-7, Op Code: These bits represent the fixed op code /D0/ (hexadecimal notation) that specifies transfer I/O as the general operation.

Bits 8-11, Device Address: These bits specify the overall device, i.e., the storage control that is identified by device address 8 (decimal).

Bits 12-14: Are not used and may be employed by the programmer for any purpose.

Bit 15: This bit determines the function of the Start instruction in conjunction with the mode (non-overlap/overlap) set. The mode is determined by bit 5 of the current PSW; if bit 5 is zero, non-overlap mode is set; if bit 5 is one, overlap mode is set. To retain the start function, bit 15 of the Start instruction must be zero when overlap mode is set; if bit 15 is one while overlap is set, the Start instruction assumes the function of the Test I/O instruction. If non-overlap mode is set, bit 15 of the Start instruction is ignored.

Bits 16-31, B1-D1 Field: These bits represent a value from which the address of the leftmost byte of a command word can be derived via either direct or effective addressing. A direct or an effectively-generated command word address must be neither below 144 (decimal) nor above the main-storage upper limit, otherwise a CPU program error stop occurs.

Bits 32-47, B2-D2 Field: These bits represent a value from which the address of a disk storage drive may be derived via either direct or effective addressing. However, only bits 40-47 of a direct or effectively-generated address specify the 2311 to be used; bits 32-39 are ignored as a 2311 address. Only the following 2311 addresses may be used:

Address	Bits							
	40	41	42	43	44	45	46	47
Drive 1	0	0	0	0	0	0	0	1
Drive 2	0	0	0	0	0	0	1	0
Drive 3	0	0	0	0	0	0	1	1
Drive 4	0	0	0	0	0	1	0	0

A disk storage drive address (bits 40-47) which is all zeros causes the program check bit in the status word to be turned on. An address that specifies an unattached 2311 causes condition code 11 (not operational) to be set.

Test I/O Instruction

The Test I/O instruction allows interrogation of any disk storage drive during overlap mode (2020 Submodel 5 only). The interrogated 2311 responds with its status whereby any interrupt conditions that may be pending for this unit are cleared. If a disk storage drive having a "clean" status is tested via Test I/O, the condition code is set to 00 and no status is stored. The Test I/O instruction is six bytes long and its bit structure is identical with that of the Start instruction (see Figure 20). Bit 15, however, is used differently:

Bit 15: This bit determines the function of Test I/O in conjunction with the mode (overlap/non-overlap) set. To retain the test function, bit 15 must be 1 when overlap mode is set (bit 5 of PSW = 1). If bit 15 is zero while overlap is set, Test I/O assumes the function of the Start instruction. If non-overlap mode is set (bit 5 of PSW = 0), bit 15 is ignored.

NOTE: Although the address in the B1-D1 field is meaningless for the Test I/O instruction, the address must nevertheless be valid, i.e., not below 144 (decimal) and not above the main-storage upper limit.

Command Word

The command word (Figure 21) specifies the operation to be performed on a selected disk storage drive and the main-storage fields associated with the particular operation. The command word is six bytes long, and the bits are assigned as follows:

Bits 0-7, Command Code: The command code can specify any one of the following operations:

Operation	Command Code
Read Data	0 0 0 0 0 0 1 0
Read Count and Data	0 0 0 1 0 0 1 0
Read Count	0 0 1 1 0 0 1 0
Write Count	0 0 0 0 0 0 0 1
Write Count and Data	0 0 0 1 0 0 0 1
Verify Data	0 1 0 0 0 0 0 1
Verify Count and Data	1 1 0 0 0 0 0 1
Scan Low or Equal	0 1 0 1 0 0 0 1
Scan Equal	0 1 1 0 0 0 0 1
Scan High or Equal	0 1 1 1 0 0 0 1
Seek	0 0 0 0 0 0 1 1
Recalibrate	0 0 0 0 1 0 1 1
Sense	0 0 0 0 0 1 0 0

		Data Address		Count Field Addr	
Command Code	Sector Count	B1	D1	B2	D2
0 7	8 15	16 19	20 31	32 35	36 47

Figure 21. Storage Control Command Word 01973

Bits 8-15, Sector Count: This field specifies the number of sectors to be processed. For all 'data' operations, the sector count determines how many records (270-byte data areas) are to be processed; the sector count may be any number from 1 to 100. For the 'read count' operation, and for all 'read count and data' operations, the sector count may be any number from 1 to 10. The 'write count and data' operation requires a fixed sector count of ten. With count and data operations, the sector count pertains to both the number of count areas and the (identical) number of data (record) areas. For seek operations and recalibrate operations, the sector count is not used.

NOTE: Although a sector count of 100 is valid for all 'data' operations, actual main-storage capacity must not be exceeded (one cylinder = 100 sectors = 27,000 bytes).

Bits 16-31, Data Address: This field specifies the address of the first (leftmost) byte of a main-storage field that extends, from this address, in ascending order The field differs in length and contains different types of information depending on the command code, as follows:

For read, write, and verify operations, the data address specifies the location of the 270-byte data field for the first sector. If more than one sector is used (multiple sector operation), the data fields for the other sectors are located in sequential 270-byte main-storage blocks in ascending order of address.

For seek operations, the data address specifies the location of a six-byte field that contains the seek address.

For sense operations, the data address specifies the location of a five-byte field into which the sensed data is stored.

For scan operations, the data address specifies the location of a 270-byte field, the contents of which are compared with data on the disk.

NOTE: The data address must neither be below 144 nor exceed the main-storage capacity otherwise the program check bit in the storage control status word is set.

Bits 32-47, Count Field Address: This field specifies the main-storage location for the first (leftmost)

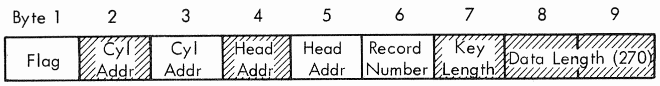


Figure 22. Main Storage Count Field

byte of a nine-byte count field for the first sector (Figure 22). For all "data" and "scan" operations, only one nine-byte count field is required (even if multiple sectors are used) because the storage control automatically updates the count field contents as required. However, for all "count and data" operations that use more than one sector, count fields for sequential sectors must be located in sequential nine-byte main-storage blocks.

NOTE: The count area address must neither be below 144 nor exceed the main-storage capacity, otherwise the program check bit in the storage control status word is set.

Data Commands

All data commands have similar operating characteristics but are used for different purposes. The data commands are:

1. Read Data (from disk into main storage)
2. Write Data (from main storage to disk)
3. Verify Data (compare main-storage data with disk).

Data commands pertain only to the 270-byte data areas (the records) on the disk and are limited to the boundaries of one cylinder (i.e., from 1 up to 100 records can be read, written, or verified for each data command). The number of records to be used is specified in the sector count of the respective command word.

All data operations may start at any specific sector within a given cylinder, and continue (if more than one sector is specified) with sequentially located sectors. The first sector (the start point) is specified by two numbers that are stored in a nine-byte count field in main storage (see Figure 22). These numbers are the record number and the head number; the head number, however, is used only when the flag byte does not indicate that an alternate track is to be used.

The information in the main-storage count field is either program generated or obtained from the disk via a read count operation. The cylinder used (that is, the position of the access mechanism) is defined for all data commands by the last previous seek operation that was issued to the same 2311 designated for the data operation.

All data operations begin with an automatic compare between the count areas (that are read as the disk rotates) and the count field in main storage. If an equal compare occurs (i.e., head and record numbers match), the sector at which the data operation was designated to begin is found and therefore one record (270 bytes of data) is transferred or verified, as appropriate. The main-storage area to which (or from which) this first record is transferred, or with which the first record is compared, is defined by the data address of the command word. The address specifies the leftmost byte of a field that extends from this byte in ascending order of address. For each record, 270 bytes must be provided. If more than one sector is specified (multiple sector operation), a corresponding number of 270-byte blocks must be located in consecutively higher main-storage locations.

For multiple sector data operations, the record number in the main-storage count field is increased by 1 each time a record is transferred (or verified). If the record number changes from 9 to zero, the head number is increased by 1. Each record is processed only when disk-storage and main-storage count fields match; this is an automatic precaution against overrunning any record. The flag bytes in the main-storage and disk-storage count fields must match also. Only one main-storage count field is required regardless of the number of sectors specified; its location is specified by the count address in the command word.

All data operations terminate either when all sectors specified have been processed or at the end of any record in which an error was detected. The residual count of the storage control status word will show how many sectors have not been processed. If an indication is present in the status word (unit check bit), a subsequent sense operation will identify the type of error condition. With the termination of any data command, the main-storage count field that supervised the operation is restored to its original value.

Read Data

The Read Data command is used to read any number of 270-byte data areas (records), from a minimum of one record to a maximum of 100, into main storage; the main storage capacity must not, however, be exceeded (one cylinder = 27,000 bytes). The operation may begin at any sector (one record per sector) but can continue only with sequential sectors.

If a cyclic redundancy check occurs (that is, the checking polynominal on disk does not agree with the one computed during the read operation), the unit check bit in the storage control status word is set. The residual count will show how many sectors have not been read and a subsequent sense operation will show the error as being a data check.

Unit check is also set when the end of a cylinder is reached; in this case, a Sense command will identify the condition as end-of-cylinder. (If the operation is to continue, the access mechanism must be positioned to another cylinder and a new read data command must be issued.)

Write Data

The Write Data command is used to write (to store on disk) any number of records from 1 to 100; the number of records to be written is specified in the sector count (one record per sector) of the command word. However, each record must have a fixed length of 270 bytes. Any portion of these 270 bytes may contain useful information, but the remaining bytes must then be written as filler characters (any valid EBCDIC bit-configuration may be used as filler). For multiple sector write operations, the data must be located in ascending 270-byte data blocks in main storage.

Verify Data

The Verify Data command is used to check the correct execution of any preceding write data operation. The verification is performed by a bit-by-bit compare between the data on disk and the data in main storage; from one up to 100 sectors (records) can thus be verified with one command. If an unequal compare is found, the operation terminates at the end of the sector in which the error was found and the unit check bit in the status word is set. A subsequent sense operation will show data check. The residual count will show how many records have not been verified. If no error is found, the operation ends when all sectors specified have been verified.

Count and Data Commands

All count and data commands have similar operating characteristics. The count and data commands are:

1. Read Count and Data
2. Write Count and Data
3. Verify Count and Data.

All count and data commands process the total contents of the sectors, i.e. the nine-byte count area and the 270-byte data (record) area. All count and data commands are limited to one track. The sector count of a count and data command word specifies simultaneously the number of count and data fields used and may range from 1 to 10, except for the "write count and data" operation for which the sector count must be ten at all times. If the sector count is either zero or above ten, the program check bit in the status word is set.

All count and data operations start at the beginning of the track (i.e., at the index point) and continue until all sectors specified have been processed. If an error is detected, the operation terminates at the end of the faulty record.

Since count and data operations always start at the index point, a count field for supervision is not required. However, for all count and data operations, a nine-byte count field must be in main storage for each sector specified, and these count fields must be located sequentially in ascending order of address, beginning at the location defined by the count address of the command word. Likewise, there must be an identical number of sequentially located 270-byte data fields in main storage. These fields begin at the location defined by the data address of the command word and extend from there in ascending order of address.

The head and the cylinder used for count and data operations is determined by the last previous seek or data operation that was issued to the 2311 designated for the count and data operation.

Read Count and Data

The Read Count and Data command reads into main storage as many count areas and as many data areas as are specified by the sector count of the command. The operation begins always at the index point, and a maximum of ten count areas and ten data areas can be read with one command. A read error causes the operation to terminate at the end of the sector in which the error occurred. An error in the data area causes data check; an error in the count area causes data check and no-record-found to be indicated.

Write Count and Data

The Write Count and Data command writes ten count areas and ten data areas onto the disk (as specified by the sector count of the command). The operation begins always at index point, and not more than ten count areas and ten data areas can be written with one command.

Verify Count and Data

The Verify Count and Data command performs a bit-by-bit compare between the count fields on disk and those in main storage, as well as between the data fields on disk and the data fields in main storage. The sector count specifies how many count fields and data fields are to be compared. The operation starts always at the index point and cannot exceed ten sectors for each command. If an unequal compare is found, the operation stops at the end of the sector containing the error and the unit check bit in the status word is set. A subsequent sense operation will show where the error was detected; if the error was in the count area, data check and no-record-found is indicated. If the error was detected in the data area, data check alone is indicated.

Scan Commands

The scan commands have similar operating characteristics. The scan commands are:

1. Scan Equal
2. Scan High or Equal
3. Scan Low or Equal.

All scan operations compare one 270-byte main-storage data field with as many disk records (270-byte areas) as specified by the sector count of the respective scan command. The purpose is to find one disk record that is equal to, high or equal to, or low or equal to, the data field in main storage.

All scan operations are limited to the boundaries of one cylinder; thus the sector count may be any number from 1 to 100. All scan operations may begin at any sector within a given cylinder but can continue only with sequential sectors. The starting point, i.e., the first sector to be scanned, is specified by the head and record number of a nine-byte count field in main storage. This count field governs the scan operation; if multiple sectors are to be scanned, the count field is updated after each record scanned and, therefore, only one main-storage count field is required regardless of sector count. If an alternate track is flagged, the head selection is determined by the alternate-track procedure.

The cylinder used (position of the access mechanism) is determined for all scan operations by the last (previous) seek operation that was issued to the 2311 selected for scanning.

Scan operations begin when the first designated sector is found (equal compare between main-storage and disk-storage count fields). The operations terminate when a record (270-byte data area) on disk is equal (based on the scan criteria specified in the command) to the 270-byte compare field in main storage. If a "hit" occurs (that is, a matching record is found), the disk-storage count field of the matching record is stored into the main-storage count field that supervised the operation. (The address of this main-storage count field is specified by the count address in the command word.)

If all specified sectors have been scanned without finding a matching record, the main-storage count field will contain the count area of the last sector scanned, and the operation ends. If a data error is found during scanning, the operation terminates at the end of the sector in which the error was detected.

The location of the main-storage compare field is defined by the data address of the command. The compare field must always be 270 bytes long. However, the programmer can limit the comparison to any number of bytes by masking that portion of the compare field that is not to be compared. All bytes that are to be spared must contain the mask character /F/ (hexadecimal).

The results of the scan operations are indicated by bits 8 and 9 of the storage control status word. The meaning of these bits is contained in the following descriptions of the commands.

Scan Equal

The Scan Equal command attempts to find one record that is identical to the 270-byte main-storage data field designated as the compare field. If an identical record is found, bit 9 of the storage control status word (the status modifier bit) is turned on and the count area of the matching record is stored in the main-storage count field. If nothing is found, the main-storage count field contains the count of the last record scanned, and no other indication is given.

Scan High or Equal

The Scan High or Equal command attempts to find one record that is either equal to or higher (in binary value) than the 270-byte main-storage field designated as the compare field. Two results are possible and whichever condition is found first stops the scan operation. If an equal condition is found, bit 9 of the storage control status word is turned on. If a high condition is found, bit 9 and bit 8 (the not-equal compare bit) are both turned on. In either case, the count area of the equal or high record is stored into the main-storage count field. If neither condition is found, only the count of the last sector scanned will be stored.

Scan Low or Equal

The Scan Low or Equal command attempts to find one record that is either equal to or lower (in binary value) than a 270-byte main-storage field that has been designated as the compare field. If an equal "hit" occurs, bit 9 of the status word is turned on; if a low "hit" occurs, bits 9 and 8 are turned on. The condition that is found first stops the operation, and the count of the equal or low record is stored into the main-storage count field. With no "hit", only the count of the last sector scanned is stored.

Function Commands

The function commands are not directly associated with the data but serve for preparatory and control purposes; the function commands are:
1. Read Count
2. Seek
3. Recalibrate
4. Sense.

Read Count

The Read Count command is used to transfer one error-free count area from any given track to a nine-byte main-storage field in order to establish a count field for supervisory purposes. The main-storage location of the first (leftmost) byte of this count field is specified by the count address of the command word.

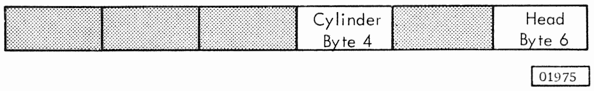
The read count operation always starts at the index point and attempts to read the count area in all ten sectors of a track. The sector count in the command is not used for the operation but must be valid (i.e., not zero and not beyond ten), otherwise the program check bit in the storage control status word is turned on. The data address of the command is also not used but must be valid (i.e., not below 144 and not beyond main-storage capacity).

The operation terminates when the first error-free count area is found; this area is stored into main storage. The residual count in the status word will then show how many sectors (in this case how many count areas) have not been read. If all ten sectors of the track have been read without finding an error-free count field, the unit check bit in the storage control status word is set. A subsequent sense operation will then show the no-record-found condition.

The head and the cylinder (i.e., the track) to be used for the read count operation is determined by the last data command or Seek command issued to the selected disk storage drive.

Seek

The Seek command is used to move the access mechanism to any desired cylinder and to select any particular head, thus selecting a track. The data address of the Seek command specifies the location of a six-byte seek field in main storage. This seek field contains the cylinder number in byte 4 and the head number in the four rightmost bits (low-value positions) of byte 6. All other bits must be zero for compatibility and future expansion.



In the seek field, the cylinder number (access position selected) pertains to all operations; the head number, however, is used only for the 'count and data' commands. Data commands select a head from the count field in main storage that governs this operation. Thus, any intervening data operation re-selects the head and will therefore determine the head for any count and data operation that may follow.

The cylinder number specified in the main-storage seek field may be any number from 0 to 102 (for the 2311 Model 12) and any number from 0 to 202 (for the Model 11). The head number may vary from 0 to 9 (both models). If the head or cylinder numbers are beyond those allowed, the program check bit in the status word is set.

Count address and sector count are not used for seek operations and are ignored. Seek operations can overlap processing or other seek operations (on other disk storage drives) because the CPU is released when the access motion begins. At the completion of the seek operation, interrupt is requested.

Recalibrate

The Recalibrate command is used to set the access mechanism to cylinder zero and to select head zero, regardless of the previous position of the mechanism, and thus allows re-orientation when the position of the mechanism is unknown. Neither data address, nor count address, nor sector count are used for the operation; these are ignored. The CPU is released when the access mechanism recalibrate motion begins. Any disk-storage command, however, is delayed until the first 15 milliseconds of a recalibrate operation have elapsed. All other operations can completely overlap the access mechanism recalibrate motion. When cylinder zero is reached, interrupt is requested.

Sense

The Sense command is used to transfer a five-byte field from the storage control to a five-byte main storage field defined by the data address in the sense command word (see Figure 23). The information thus stored indicates the conditions that caused premature ending of an operation. In non-overlap mode, the sensed data always reflects the conditions for the operation last terminated. The unit address in the Start instruction that reads out the Sense command may, therefore, be the address of any attached disk storage drive. In overlap mode, the sensed data always pertains to the disk storage drive addressed by the Start instruction. The five bytes of information are divided into status, data address, and count information.

Sensed Status	Data Address	Count Information
0 7	8 23	24 39

Figure 23. Sense Data 01976

Status (Figure 23)

Sensed Status (Bits 0-7)

Bit 0: Not used (is set to zero).

Bit 1, Intervention Required: Indicates that the last operation could not be executed, either because the selected 2311 was not switched to on-line (that is, if the switch next to the use meter was off) or because the 2311 was not running up to speed.

Bit 2, End of Cylinder: Indicates that the last operation was terminated because the end of the cylinder was reached.

Bit 3, Equipment Check: Indicates that a hardware malfunction was detected during the last operation.

Bit 4, Data Check: Indicates that a read error (disagreement between the cyclic check bytes on disk and the cyclic check bytes computed in storage control) was detected during the last read operation. This bit is also set by an unequal compare during verify operations.

Bit 5, Seek Check: Indicates that the last seek operation to be attempted was executed in a faulty manner, for example, the detent of the access mechanism failed and the heads may therefore be positioned between cylinders, or the access mechanism did not halt but ran into the crash stop.

Bit 6, No Record Found: Indicates an error associated with the count fields. The bit is set either:

1. When the first record specified (by a main-storage count field) for a data operation cannot be found, or
2. Should the first record be found, when any subsequent record (in a multiple sector data operation) is out of sequence.

The bit is also set under the following conditions:

1. If a count area cannot be read without error during a 'read count and data' operation.
2. If any of the ten count fields attempted cannot be read without error during a 'read count' operation.

Bit 7, Track Condition Check: Is set during a data or a scan operation when the flag byte of the main-storage count field is not identical with the flag bytes found on the track; this condition causes the operation to terminate. The track condition check cannot occur during 'count and data', or read count operations.

NOTE: For any operation (other than sense) that terminates with unit check, the entire status portion (bits 0-7) of the sense field will be zero. A program check that occurs without channel end or device end does not alter the sense data.

Data Address (Bits 8-23)

This area of the sense field contains the address of the first (leftmost) byte of that 270-byte main-storage field into (or from) which the last record was transferred. The last record may be either the last one specified or, in case of premature ending due to an error, the record in which the error was detected.

NOTE: The data address is unpredictable for seek and recalibrate operations, or for any operation that was rejected (i.e., not started) because of errors, busy, or not-ready conditions.

Count Information (Bits 24-39)

The contents of this area depend on the type of operation successfully initiated or completed immediately prior to the Sense command.

For All Data Operations: The count information area contains the head and record number of the last disk storage count area involved in the operation. (The main-storage count field contains the original data.)

For All Scan Operations: The count information area contains the head and record number originally specified in the main-storage count field. (The main-storage count field meanwhile contains the count of the matching record.) If nothing was found during the compare, the count information shows the last record scanned. If errors are detected, the count information identifies the faulty record.

For All Count and Data Operations: The count information shows the main storage address of the count field for the last sector transferred or attempted.

NOTE: The count information is unpredictable for all seek, recalibrate and unsuccessful read count operations, and for all operations not initiated due to busy or not-ready conditions.

DEFECTIVE SURFACES OF DISKS

With the high density magnetic recording techniques used in the 2311, minute particles of contamination can cause disk surface damage during the life of the disk pack. While experience with magnetic recording surfaces indicates that surface damage resulting in unreadable areas is a rare occurrance, a means for identifying these defective areas is provided in the flag byte of the count area on disk.

An entire track can be flagged as defective by setting the track condition bits in the flag byte of the count area. The three possible settings of the track condition bits are:

For Usable Original Track: The track condition bits are set to 00. In this case, the track address portion of the count area is identical to the actual address of the track in disk storage.

For Defective Original Track: The track condition bits are set to 10. In this case, the track address portion of the count area is identical to the actual address of the alternate track. All ten sectors on the track are flagged as defective, and all ten sectors contain the alternate track address in the count area.

For Usable Alternate Track: The track condition bits are set to 01. In this case, the track address portion of the count area is identical to the actual address of the original defective track.

Data Recovery and Flagging Procedures

If a track contains a defective area, either during initial formatting of the track or on a subsequent attempt to read or verify data written on the track, the following procedures may be used to recover data, flag the track, and assign an alternate track.

Data Recovery

If the defective track was detected during a verify operation after a 'write count and data' operation, the data to be written should still be available in main storage. Sectors which were written and verified successfully, prior to the sector in which the error occurred, can be recovered by a read operation.

If the defective track was detected during a read operation, the data for the sector in which the error occurred cannot be recovered from disk storage. In this case, data recovery may require the use of an audit trail procedure to reconstruct the data.

If the defect was in the count area, the data may be recovered by using a 'read count and data' operation. If bit position 6 (no record found) of the status byte has been set to 1 and all other bit positions in the byte have been set to 0 after the 'read count and data' operation, the count area is unreliable, but the data area is good. The count area for that sector may be reconstructed, using an audit trail procedure, and the remaining sectors may be recovered by a read data operation.

Flagging a Defective Track

The defective track can be flagged by setting up the count and data areas in main storage to write 270 bytes of zeros for the data area; for the count area in main storage, the track condition bits in the flag byte are set to 10 and set in the alternate-track address. All of the ten sectors on the track must be written in this format, using a 'write count and data' operation.

Assigning an Alternate Track

The assignment of data tracks and alternate tracks is flexible and the user can allocate data and alternate tracks as needed within the total available. The 2311 Model 11 has 203 tracks available, and the Model 12 has 103 tracks available. Typical assignments of data tracks for 2311 Models 11 and 12 are:

	2311 Model 11	2311 Model 12
Data Cylinder Address	0-199	0-99
Alternate Cylinder Address	200-202	100-102

Any data track found to be defective can be assigned an alternate-track address. Spare tracks, not already assigned as data or alternate tracks, can be determined by keeping a table of available tracks.

Copying Data on an Alternate Track

The data from the defective track should be set up in main storage with the count area identical to the actual address of the original track. The track condition bits in the associated flag byte should be set to 01. Since heads are selected from the count area in main storage during data operations, care must be taken that the head address in the count area written on the alternate tracks is identical to the head address in use. 'Write count and data' operations can be used to copy the data on to the alternate track. If the head used for the alternate track differs from that used for the defective track, the programmer must store the defective track head address in main storage to ensure that the program can resume disk-storage operations in the right location.

Typical Alternate-Track Procedure

During normal disk-storage operations, the track condition bits of the flag byte in main storage should be set to 00 for a usable original track. If bit position 7 (track condition check) in the status byte has been set to 1 during the operation, the track has been flagged as a defective track. In this case, the programmer should make a read count operation to recover the alternate-track address, then make a seek operation to the alternate address derived from the count area of the defective track. The data on the alternate track can now be processed as if it were on the original track.

The track condition bits of the flag byte in main storage must be set to 01 to process the data on an alternate track, and the head address in the count area in main storage should be identical to the head address in use. After processing an alternate track, disk-storage operations can continue with a seek operation on the next track to be processed.

STORAGE CONTROL TO PROGRAM RESPONSE

Whenever a Start instruction addresses the storage control, a response is given to the program to indicate how the program may proceed. Three types of responses are given: a condition code is set, status information is provided, and interrupt may be requested. The first response given is always the condition code, which indicates whether a command has been rejected, initiated, or terminated; the indication, however, does not show whether the command has been executed successfully. Ending conditions are indicated in a status word. The interrupt request informs the program that a storage control facility or a disk storage drive has become available. Condition code settings and interrupt requests differ for non-overlap and overlap modes.

Condition Code Setting

Condition Code in Non-Overlap Mode

Condition Code 01: Indicates that status information has been stored and that the status word, located in bytes 156-161 (decimal) of main storage, should be examined. The status will then show whether the command specified in the Start instruction has been rejected, started, or terminated, and why it was rejected or how it was completed.

Condition Code 10: Indicates that the command has been rejected because the CPU is busy processing time-shared operations (other than 1403 buffered print or control-immediate commands in the input/output channel). With condition code 10, status is not stored but a CC2 interrupt request is placed. This request causes an interrupt, identified by device address 15, when all currently running time-shared operations are completed (provided interrupts are enabled).

Condition Code 11: Indicates that the command has been rejected because the address of an unattached disk storage drive was specified in the unit address portion of the Start instruction. Status is not stored when condition code 11 is set.

NOTE: If the storage control feature is not attached, a Start instruction for it causes no operation and no condition code is set. Condition code 00 is not set in non-overlap mode.

Condition Code in Overlap Mode (2020 Submodel 5 only)

Condition Code 00: Indicates that a data transfer (any data, scan, or count and data) operation has been initiated and that the storage control is executing it. (Sense, seek, or recalibrate operations are not included; these set condition code 01 under similar circumstances.)

Condition Code 01: Indicates that the status word should be examined because it is significant. For example, condition code 01 indicates:

1. A reject condition, in data transfer operations
2. A successful initiation, in seek and recalibrate operations
3. A completion, in sense operations.

Condition Code 10: Indicates that the storage control is busy, either with the execution of a previously issued command or because a channel-end interrupt condition is pending in a disk storage drive (other than the unit currently addressed). A pending channel-end condition indicates that a seek or recalibrate operation on another disk storage drive has attempted to release the storage control.

Condition Code 11: Indicates a not-operational condition, that is, the address of an unattached disk storage drive was specified in the unit address of the Start instruction.

Interrupt Request

Interrupt is requested for disk-storage operations whenever their termination cannot be predicted, thus necessitating an indication that either the storage control or a disk storage drive has become available. Interrupt is requested differently for non-overlap and overlap modes.

Interrupt in Non-Overlap Mode

Interrupt is requested only for seek and for recalibrate operations, provided they cause the access mechanism to move. If a movement of the access mechanism is required (i.e., the mechanism is not already positioned at the desired cylinder), the disk storage drive signals 'channel end' when the motion begins; when the access mechanism stops, the drive signals 'device end', which requests interrupt. However, the interrupt is performed only when the channel mask bit of the current program status word is on. The interrupt is then identified by device address 8 (which refers to the storage control) being stored in the old PSW. The conditions causing the interrupt are identified in the status word.

If interrupts are disabled, the device-end interrupt remains pending, causing the device to appear busy. A pending interrupt can be cleared by addressing the disk storage drive that caused it, via a Start instruction. The Start instruction will not, in this case, initiate a command but will obtain the status of the pending device only; this status contains the busy bit. This fetching of the status clears the pending interrupt, so that the device becomes available thereafter. All commands that terminate with both channel end and device end do not request interrupt.

Interrupt in Overlap Mode (2020 Submodel 5 only)

In overlap mode, all disk-storage commands request interrupt at the time they terminate. The one exception is in seek and recalibrate operations that do not require access motion; if this is the case, these operations do not request interrupt because, with no access motion, the channel end, device end, and any other condition have been signaled already during the initial selection and are stored in the status word, the action being indicated by condition code 01.

Interrupt is performed only when the channel mask of the current program status word is on. If interrupts are disabled, they remain pending. A pending device-end interrupt causes the pending device to appear busy; a pending channel-end interrupt causes the entire storage control to appear busy. A pending interrupt must be cleared if operations are to continue.

In overlap mode, pending interrupts are cleared by a Test I/O instruction that addresses the device that caused the pending interrupt. Test I/O obtains the status of the device, whereby any pending interrupt conditions are cleared. If the tested device has a "clean" status, the condition code is set to 00 and no status is stored.

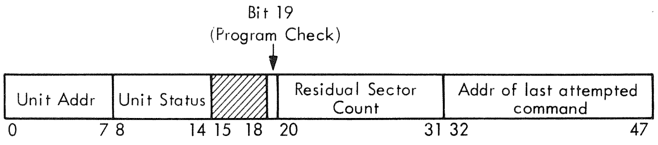


Figure 24. Storage Control Status Word 01977

If a channel-end interrupt is pending and a disk storage drive other than the one that requested the interrupt is addressed, the condition code is set to 10 (storage control busy) and no status is stored.

NOTE: A pending interrupt is also cleared when the device that caused it is addressed by a Start instruction. However, the Start instruction is not suitable for interrogating purposes because, if a device with "clean" status is addressed, an operation is initiated on that device.

STORAGE CONTROL STATUS WORD

The storage control status word (Figure 24) contains information that allows the program to decide on further action. The status word reflects the conditions that caused the last operation to be rejected or terminated. The status word is six bytes long and is located in main-storage positions 156-161. The bit assignment is as follows.

Bits 0-7, Unit Address

The unit address identifies the disk storage drive for which the status is stored.

Bits 8-14, Unit Status

Each bit of the unit status shows a specific condition when on.

Bit 8, Not Equal Scan: Indicates either a high- or a low-compare result of a scan; the bit is zero for all equal compares.

Bit 9, Status Modifier: Is on whenever a "hit" of any type is made during a scan operation. If bit 9 alone is on (bit 8 off), it indicates that an equal compare occurred during the scan. Bit 9 is off if no "hit" of any kind was made.

Bit 10: Is not used and is set to zero.

Bit 11, Busy: Indicates that the addressed disk storage drive is either busy executing a command or has interrupt pending. If the busy indication originated from a pending interrupt, this interrupt can be considered cleared.

Bit 12, Channel End: Indicates that the storage control is free for another operation, either because a command (Seek or Recalibrate with access mechanism motion) has been started successfully, or because a data transfer type of command has been terminated.

Bit 13, Device End: Indicates that the addressed disk storage drive is free for another operation. Device end, alone, indicates that a seek or recalibrate operation (with access mechanism motion) has been completed, or has been terminated via off-line switching. Device end, together with channel end (bit 12), indicates that any data transfer operation or a seek or recalibrate operation (without access mechanism motion) has been completed.

Bit 14, Unit Check: Is a summary indication for the following conditions:

1. Data Check
2. No Record Found
3. Intervention Required
4. Track Condition Check
5. Equipment Check
6. End of Cylinder
7. Seek Check.

Sense data obtained via a Sense command will show which of these conditions actually occurred.

Bits 15-18

Bits 15-18 are not used and are set to zero.

Bit 19, Program Check

The program check bit indicates one of the following errors:

1. Bits 40 through 47 (unit address) of the number derived from the B2-D2 field of the Start or the Test I/O instruction contains all zeros and any one of the bits 32 through 39 was on.
2. The command code is zero.
3. The data address specified in the command word is either invalid (below 144, or above the upper limit of main storage) initially or the main-storage upper limit is violated during execution of the command. In the latter case, the operation is terminated with channel end and device end.
4. The count field address in the command word is either above the main-storage upper limit or is below 144, or the upper limit is violated during multiple sector count and data operations.
5. The main-storage count field contains an invalid head number or cylinder number. (The cylinder number must not exceed 102 for the 2311 Model 12, or 202 for the Model 11. The head number must not exceed 9.)

Bits 20-31, Residual Sector Count

The residual sector count indicates the actual number (in binary notation) of sectors not satisfactorily completed. If eight sectors, for example, were specified in the sector count and an error was detected in the fifth sector, the residual count will be 4.

NOTE: If a scan operation was terminated due to a "hit", the residual count includes the sector in which the hit was made. However, the residual count is valid only for operations that terminate with channel end and device end. For seek, recalibrate, and sense operations, the sector count is of no significance and should not be used.

Bits 32-47, Command Address

The command address indicates the main-storage location of the command that caused storing of the status, that is, the last command performed or attempted.

IBM 1419 MAGNETIC CHARACTER READER

INTRODUCTION

The 1419 Magnetic Character Reader can be attached to a System/360 Model 20 through a Serial Input/Output Channel (SIOC). The 1419 reads into the system the magnetically inscribed information on checks and other banking documents at speeds as high as 1,600 documents per minute. Documents can be sorted into as many as 13 classifications as they are read. All magnetic inscriptions can be checked for validity.

Documents read by the 1419 Magnetic Character Reader may be of intermixed sizes and thicknesses, as typically encountered in check-handling operations. The standard minimum length is six inches; shorter documents, such as the 51-column postal money order, can be read into the System/360 Model 20 at a maximum rate of 1,960 documents per minute. Shorter documents (which may be intermixed with standard-length documents) can be sorted if the optional (no charge) feature for this purpose is installed.

Many special features are available for the 1419, including an endorser that prints a full endorsement on the back of each document at no reduction in operating speed.

INSTRUCTIONS

The 1419 Magnetic Character Reader operates through the Serial I/O Channel (SIOC) of the 2020. The SIOC is identified by a device address of 6.

An Engage 1419 instruction must be given to start document feeding. A Read From 1419 instruction notifies the CPU to accept information as it is available from the document. A Select Stacker Pocket instruction must precede the arrival of a document at photocell number 6. A Disengage 1419 instruction stops the flow of documents.

Transfer Instruction (XIO)

OP Code	DA	FS	Function
D0	6	2	Read From 1419

The direct or effective address is derived from the B1-D1 fields of the instruction and specifies the rightmost data location in core storage. Length of the data transferred is derived from the B2-D2 fields of the instruction.

When the Read From 1419 instruction is terminated, the low order byte (residual count) of the field length specification is stored in main-storage location 155.

A Read From 1419 instruction causes a serial transfer of characters from the 1419 to the CPU. The CPU accepts each character as the 1419 makes it available. The CPU is free for other processing during a read operation except for 55 microseconds per character.

The first character or symbol transferred is placed in the rightmost (highest) core-storage position specified by the instruction and subsequent characters enter successively lower core-storage locations.

The read operation is terminated by an end of data signal which occurs when the trailing edge of a document passes the read head (or when a specified character count has been exceeded) whichever occurs first. An end of data signal also occurs as a result of a jam or stop delay during a read operation. Therefore, a separate Read From 1419 instruction is required for each document.

Failure to provide a Read From 1419 instruction before the leading edge of a document is sensed at the read head causes the document to be rejected. An interrupt is requested when a read operation is terminated.

An early interrupt occurs when the leading edge of a document is sensed at Photo Document Sensor 5 (PDS 5) if a read operation is still in progress for that document. This interrupt indicates that approximately six inches of a document have passed the read head, but this does not end the read operation. The early interrupt is used for establishing the stacker pocket destination for the document.

Condition Code Setting

The condition code is set to 00, 01, or 11 at the time the execution of an XIO instruction for the SIOC is completed to indicate that the SIOC is available, working, or not operational.

The transfer of data, as specified in the XIO instruction, is initiated only when the SIOC is in the available state. The SIOC is available when the 1419 is attached, no previous read or write operation is in progress, no read transmission error condition exists, and the 1419 is ready for operation.

The 1419 is ready for operation when a Read From 1419 instruction is issued and no jam or stop delay is present, and at least one of the following conditions exists;

1. Separator on.
2. Document between separator and PDS 3.
3. Document recognized at PDS 4.

The SIOC is working or busy when a previous read or write operation is in progress.

The SIOC is not operational if the I/O device (the 1419) is not attached or is not ready for operation.

Test I/O and Branch Instruction (TIOB)

OP Code	DA	FS	Function
9A	6	1	Document to be read
9A	6	2	Document under read head
9A	6	3	Amount field valid
9A	6	4	Process control field valid
9A	6	5	Account No. field valid
9A	6	6	Transit field valid
9A	6	7	Serial No. field valid
9A	6	8	Auto Select
9A	6	9	Read transmission error
9A	6	A	Jam/sort check

By means of the TIOB instruction, the line defined by the function specification is tested; if the condition indicated by the line exists, the updated instruction address is replaced by the branch address. If the condition does not exist, normal instruction sequencing proceeds with the updated instruction address.

Document to be Read: This indicator is on when there is a document between the separator and the read head. It turns off when a document leaves the read head, but no trailing document is between the separator and the read head. The indicator remains off until another document leaves the separator. The indicator is also turned off by a right or left feed jam.

Document Under the Read Head: This indicator turns on when the leading edge of a document is sensed by photocell 4 (at the read head). It turns off when the trailing edge of the same document passes the read head. It is also turned off when a document jams between the separator and the read head.

Valid Amount Field: This indicator turns on when the amount field has been completely read without errors. The indicator turns off when the leading edge of the next document is sensed at the read head.

The indicator does not turn on if:

1. Any of the characters in the amount field (including the amount-field special symbols) are unreadable.
2. Amount field symbols are missing or out of sequence.
3. The field is missing.
4. The field length is invalid.

Valid Process-Control Field: The indicator turns on when the process-control field has been completely read without errors. It turns off when the leading edge of the next document is sensed at the read head.

The indicator does not turn on if:

1. Any of the characters in the process-control field (including the special symbol) are unreadable.
2. The field is missing.
3. Special symbols are out of sequence or missing.
4. The field length is invalid (fixed-field lengths only).

Valid Account-Number Field: This indicator turns on when the account-number field is read correctly. It turns off when the leading edge of the next document is sensed at the read head.

The indicator does not turn on if:

1. Any of the characters in the account-number field (including the special symbol) are unreadable.
2. Special symbols are missing or out of sequence.
3. The field is missing.
4. The field length is invalid (fixed-lengths only).
5. The self-checking account-number device (special feature) indicates the account number is in error.

Valid Transit-Number Field: This indicator turns on when the transit-number field is read correctly. It turns off when the leading edge of the next document is sensed at the read head.

The indicator does not turn on if:

1. Any of the characters in the transit-number field (including the special symbols) are unreadable.
2. Special symbols (except the dash) are missing or out of sequence.
3. The field is missing.
4. The field length is invalid.

Valid Serial-Number Field: This indicator turns on when the serial-number field is read correctly. It turns off when the leading edge of the next document is sensed at the read head.

The indicator does not turn on if:

1. Any of the characters in the serial-number field (including the special symbols) are unreadable.
2. Special symbols (except the dash) are missing or out of sequence.
3. The field is missing.

Auto Select: The Auto Select indicator is off if a Stacker Selection command can be executed correctly. The auto select bit is set on when a document is not acceptable due to spacing (too close together), a late read, or extreme mutilation.

The auto select bit is also turned on when a document reaches the select station prior to the issuing of a Stacker Select command. A good programming practice is to test the setting of the auto select bit immediately prior to the issuing of a Stacker Select command. After the Stacker Select command is accepted, the auto select bit is turned off, thus preventing two Stacker Select commands from being issued for the same document.

All auto select documents are routed to the reject pocket.

Read Transmission Error: This indicator is turned on when the data transmitted from the 1419 to the CPU has incorrect parity. The indicator is reset by the Test instruction.

Jam or Sort Check: This check bit (and its associated indicator light) indicates a right or left feed jam or a sort check condition. The bit remains on until the operator clears the jam and resets the condition using the Stop/Restore key.

Control I/O Instructions (CIO)

OP Code	DA	FS	DS	Function
9B	6	0	0	No operation
9B	6	0	1	Select or Light Pocket A
9B	6	0	2	Select or Light Pocket B
9B	6	0	3	Select or Light Pocket 0
9B	6	0	4	Select or Light Pocket 1
9B	6	0	5	Select or Light Pocket 2
9B	6	0	6	Select or Light Pocket 3
9B	6	0	7	Select or Light Pocket 4
9B	6	0	8	Select or Light Pocket 5
9B	6	0	9	Select or Light Pocket 6
9B	6	0	A	Select or Light Pocket 7
9B	6	0	B	Select or Light Pocket 8
9B	6	0	C	Select or Light Pocket 9
9B	6	0	D	Select Reject or Batch Number Update
9B	6	0	E	Pocket Light and/or Batch Number Control
9B	6	0	F	No operation
9B	6	1	0	No operation
9B	6	1	1	Engage 1419
9B	6	1	2	Disengage 1419

The CPU sends control information on the line specified by the function specification and detail specification fields of the control instruction.

NOTE: A Disengage command given when the separator is off is effective immediately and it terminates the previous Engage instruction.

Program Sort Mode: The purpose of this mode is to provide CPU control of pocket selection and to permit transmission of data from documents to the CPU.

1. An Engage 1419 instruction starts document feeding.
2. A Disengage 1419 instruction stops the flow of documents.
3. A read from 1419 prepares the CPU for accepting information as it becomes available from each document.
4. A Select Stacker instruction should precede the arrival of each document at PDS6.
5. Information on validity of transmitted data and end of document is presented to the CPU as soon as available.
6. The Read Field keys on the 1419 determine which fields of data are to be transmitted.

Pocket Selection Time

The program must make a stacker selection before the document reaches the select station. The pocket selection must not be executed until it is known that the document is no longer under the read head. There are 23.62 (-0.95, + 0.0) milliseconds available between the time the early interrupt is generated and the latest point at which a Stacker Selection command must be given. If a pocket decision is not made before the document reaches the select station, the document is rejected and the Auto-Select indicator is turned on.

Program Control for Pocket Lights: When the Program Control for Pocket Lights special feature is installed, it is possible to light the pocket lights (A through 9) in the 1419 with the use of program instructions. The correct sequence of operations is as follows:

1. Disengage.
2. Handle follow-up items in normal manner.
3. Issue Pocket Light and/or Batch Number Control command.
4. Issue Stacker Select commands for the appropriate pocket lights.
5. Engage to restart document feeding.

NOTE: During program testing, avoid unnecessary stops. Do not issue CIO instructions for the Program Control for Pocket Lights (PCPL) special feature when it is not installed on a 1419 with Batch Numbering special feature. If the PCPL feature is inadvertently instructed, document feeding is stopped; however, the motor continues to run. To resume normal operation, press the 1419 Stop key, turn the power off and then back on, and press the 1419 Start key. No error condition should be indicated.

Batch Numbering: The installation of this special feature allows documents to be numbered on the reverse side for audit purposes. This number may be advanced by one under program control. The same timing restrictions that apply to proper stacker selection apply to batch number updating. When the batch number update is issued in conjunction with the stacker select for a given document, the updated number is printed on the document. Two CIO instructions are required for the operation as follows:

1. The Pocket Light and/or Batch Number control instruction which prepares the numbering device to accept the updating instruction.
2. The Reject and/or Batch Number updating instruction which advances the batch number.

NOTE: these instructions must be given in the sequence shown in order for the 1419 to properly recognize the dual functions of these instructions.

Document feeding is suspended if more than one advance sequence is given within three seconds. Feeding is suspended, resulting in a reduction of throughput, for the remainder of the three-second interval. The 1419 resumes feeding automatically, unless a Disengage 1419 instruction is issued. An advance check stop occurs if the Advance Check On switch is on and the batch numbering device has failed to respond to the updating instruction.

System Stops

1. System stops should be initiated by depressing the 1419 Stop/Restore key prior to stopping the CPU.
2. When the system is stopped by any other key or by a system error stop, document feeding is terminated. Documents in flight that have not been stacker selected are directed to the reject pocket.
3. Jams prior to the read head do not affect previously read documents.
4. For jams after the read head, the operator must examine the print-out and properly distribute any items which stop prior to or in the selector unit, based on the print-out.
5. Sort check stops are caused by a malfunction in the equipment and cannot be caused by the program. This stop occurs if no stacker has been activated for a document that reaches the select station, or if more than one stacker has been activated for a document at the select station. When a sort check stop occurs, the 1419 stops feeding and the remaining documents in process (one to five may be in the feed) are read, stacker selected, and processed. The 1419 Sort Check indicator and the Stop/Restore key are turned on. The Jam/Sort Check indicator bit is turned on also.

   When a sort check occurs, the error document is found in the reject pocket.

   The operator must examine the print-out of the items and properly distribute the item involved.

   The 1419 Stop/Restore key must be depressed to reset the sort check condition.
6. End-of-Transport Stop: An End-of-Transport Sense switch stops document feeding if a document in the transport fails to enter a pocket. When an end-of-transport stop occurs, both the 1419 Left Feed light and the Ready light come on.

   The operator removes the document and determines its proper distribution in the manner prescribed by the operation currently in progress.

   The Left Feed light is reset by depressing the Stop/Restore key.
7. Film Stop: The write head and the read head are each protected from excessive wear and damage by thin metal tapes called head-film tapes. When either film tape breaks, or when its supply runs out, the Film light turns on and an immediate stop occurs.

   The operator corrects the film stop condition and follows the jam error recovery procedures.
8. Non-Error Stops: If the following 1419 stops occur, the operator should inspect the machine and take corrective action.
   a. Full pocket stop.
   b. Empty feed hopper stop.
   c. Hopper feed failure stop.
   d. Operator stops.
9. Unreadable Documents: If for any reason the 1419 cannot read and transmit any selected information correctly, the indicators provided can be tested to determine the reason.
10. Endorse Stop Condition: This condition can occur only when the endorser feature (optional) is installed and the customer has the Endorse Checking switch turned on.

    An endorse stop condition causes the 1419 to stop with the Endorse Stop indicator and the Stop/Restore key lighted.

    The operator must examine the last few documents processed for proper endorsements and handle them in a manner prescribed by the operation currently in progress.

    The 1419 Stop/Restore key must be pressed to reset the endorse stop condition.
11. Advance Check Stop: This condition can occur only when the Batch Numbering special feature is installed and the Advance Check switch is in the On position. The advance check stop condition causes the 1419 to stop feeding with the Advance Check indicator and the Stop/Restore key lighted. This stop is caused by a failure to advance the number device.

IBM 1259 MAGNETIC CHARACTER READER

INTRODUCTION

The IBM 1259 Magnetic Character Reader is a medium-speed reader/sorter that is capable of reading and sorting magnetic-ink-inscribed banking documents of intermixed sizes and paper weights at speeds of up to 600 documents per minute.

The machine is connected to the System/360 Model 20 through the Serial I/O Channel of this system, and can thus operate either on-line (that is, under control of the Model 20 stored program) or off-line (that is, under control of its own circuitry and operator's panel). The operating mode (on-line or off-line) is selected by a switch.

The data made available from the banking documents can be transferred to the Central Processing Unit (CPU) and can thus be used for calculations and analysis, and for the generation of management reports that may be produced on the printer used with the Model 20.

Three models of the 1259 Magnetic Character Reader are available, each being customized according to the main requirements of the banking business in the United States, the United Kingdom and Australia, and European countries. The models are as follows:

1259 Model 1:
Reading E13B type font; 60 Hertz (Hz) power.
1259 Model 31:
Reading E13B type font; 50 Hz power.
1259 Model 32:
Reading CMC-7 type font; 50 Hz power.

These type fonts and their equivalent conversions are shown in Figures 25 and 26. All models are equipped with eleven sort pockets, one for each digit (zero through nine), and one reject pocket. All models accept documents of intermixed dimensions as follows:

Length : 5.50 to 8.75 in (140 to 222 mm). Documents longer than 3.50 in (89 mm) and shorter than 5.50 in (140 mm) are not read and are rejected without causing a machine stop. Documents shorter than 3.50 in (89 mm) are not read and cause a machine stop.

Width : 2.50 to 4.14 in (64 to 105 mm). When the document width is 4.0 in (102 mm) or more, the length must be 7.0 to 8.75 in (178 to 222 mm).

Thickness : 0.003 to 0.007 in (0,076 to 0,178 mm) Carrier documents up to 0.011 in (0,279 mm) enclosing a document are fed, transported and stacked.

Paper Stock: Short and long grains 20 lb to 44 lb (75 to 165 grams per square meter).

Card documents with 51 columns can be fed through the 1259. Because these documents are shorter than standard length, they are automatically routed to the reject pocket. The documents are counted during either off-line or on-line operation, as part of the total documents being processed.

When operating on-line, stacker selection is controlled by the system to which the 1259 is attached. After having been started, the magnetic character reader reads the specified information from the inscribed documents and sends this information to the system. The system responds with a command which directs the document just read to the desired stacker. When operating off-line, the 1259 reads the specified data field sort position and directs the document to the desired stacker. Regardless of whether the magnetic character reader is operating

Clear Text	Amount Symbol	On-Us Symbol	Transmit Symbol	Dash Symbol	Digits (0 to 9)	Unreadable Character
1259 Name	SS 1	SS 2	SS 3	SS 4		
E13 B Font	⑇	⑈	⑆	⑉	0123456789	
Converted into	$	'	⌑ or <	—	0 1 2 3 4 5 6 7 8 9	*
Hexadecimal Value	/5B/	/6B/	/4C/	/60/	/F0/ to /F9/	/5C/

Figure 25. E13B Type Font and its Equivalent Conversion 01430

Clear Text	Special Symbols (See Note)					Digits (0 to 9)	Unreadable Character
1259 Name	SS 1	SS 2	SS 3	SS 4	SS 5		
CMC-7 Font						0123456789	
Converted into	$	,	⌑ or <	—	&	0 1 2 3 4 5 6 7 8 9	*
Hexadecimal Value	/5B/	/6B/	/4C/	/60/	/50/	/F0/ to /F9/	/5C/

NOTE: Use and meaning of the special symbols are at the customer's discretion.

Figure 26. CMC-7 Type Font and its Equivalent Conversion 01429

on-line or off-line, various checking facilities ensure the integrity of data being read.

INSTRUCTIONS

Refer to Table 17 for the 2020 instruction set that is used with the 1259.

Transfer I/O Instructions

Transfer I/O (XIO) instructions occupy six bytes of main storage and consist of:

1. The Operation (Op) code that specifies the operation to be performed.
2. The Device Address (DA) which specifies the I/O device that is selected for the operation.
3. The Function Specification (FS) that specifies the primary function of the operation.
4. The B1-D1 field that contains the start address of the main storage input or output field, and the B2-D2 field that specifies the true length of the input or output field. The B1-D1 field and the B2-D2 field can be used either as direct addresses, or an effective address may be derived from them.

Read From 1259

The Read From 1259 instruction causes a serial transfer of the characters from the document to the CPU. The B1-D1 field of this instruction must specify the rightmost (low-order) position of the

Table 17. 2020 Instruction Set used with IBM 1259 Magnetic Character Reader 01431

Format	Type	Op Code	DA	FS	Detailed FS	Function
SS	XIO	DO	6	2	None	Read From 1259
SI	CIO	9D	6	1	0	No Op
SI	CIO	9B	6	1	1	Engage 1259
SI	CIO	9B	6	1	2	Disengage 1259
SI	CIO	9B	6	0	0	No Op
SI	CIO	9B	6	0	1	Select Pocket 0
SI	CIO	9B	6	0	2	Select Pocket 1
SI	CIO	9B	6	0	3	Select Pocket 2
SI	CIO	9B	6	0	4	Select Pocket 3
SI	CIO	9B	6	0	5	Select Pocket 4
SI	CIO	9B	6	0	6	Select Pocket 5
SI	CIO	9B	6	0	7	Select Pocket 6
SI	CIO	9B	6	0	8	Select Pocket 7
SI	CIO	9B	6	0	9	Select Pocket 8
SI	CIO	9B	6	0	A	Select Pocket 9
SI	CIO	9B	6	0	D	Select Reject Pocket
SI	CIO	9B	6	0	F	No Op
SI	TIOB	9A	6	1	None	Test Document To Be Read
SI	TIOB	9A	6	2	None	Test Document Under Read Head
SI	TIOB	9A	6	3	None	Test Amount Field (Field 1) Valid
SI	TIOB	9A	6	4	None	Test Process-Control (Field 2) Valid
SI	TIOB	9A	6	5	None	Test Account-Number (Field 3) Valid
SI	TIOB	9A	6	6	None	Test Transit-Number (Field 4) Valid
SI	TIOB	9A	6	7	None	Test Serial-Number (Field 5) Valid
SI	TIOB	9A	6	B	None	Test Field 6 Valid
SI	TIOB	9A	6	C	None	Test Field 7 Valid
SI	TIOB	9A	6	8	None	Test Auto Select
SI	TIOB	9A	6	9	None	Test Read Transmission Error
SI	TIOB	9A	6	A	None	Sorter is Stopped

main-storage input field. The B2-D2 field must specify the true number of characters that are to be read. Digits, special symbols, and dashes are characters, but spaces are not transmitted. The rightmost input field position must be specified because data is read right-to-left.

Read operations may be time-shared with other CPU operations. Thus, during a read operation, the CPU remains free for other processing operations except for 58 microseconds per character. A Read instruction ends either when the trailing edge of the document is sensed at the read head, or a specific character count has been reached, whichever occurs first. The end of a Read instruction is signalled by a request for interrupt. The 1259 interrupt is identified by the device address 6 and the function specification 6 that are stored in the old program status word. To prevent a CPU hangup condition, a Read instruction is ended automatically if it is issued while the 1259 drive motor is not running (no document feed). As a further precaution against a CPU hangup, a Read instruction should be terminated before a CPU Halt command is issued.

Programming Note: The latest time at which a Read From 1259 instruction ends is the moment when the trailing edge of the document is sensed at the read head. When the Read instruction ends in the middle of the document (because a specific character count has been reached), more time is available for processing and stacker selection. However, the programmer must exercise caution when using this approach because certain conditions can, nevertheless, limit the decision time that is available. For example, when a number of variable fields on the documents are read, and all fields are of minimum length, the Read instruction terminates at the end of the document and less time is available than was anticipated.

Condition Code Setting

When a Read From 1259 instruction is issued by the program, the condition code is set to inform the program of how it may proceed. Condition code 00 (binary) means that the 1259 is available to execute the instruction; condition code 01 (binary) indicates that the 1259 is busy, that is, occupied with processing a previously-issued Read instruction; condition code 11 (binary) indicates a not-operational condition. On a not-operational condition, the 1259 should be disengaged to prevent loss of documents when the machine is made ready again.

The 1259 is not operational when:

1. The start key is held depressed.
2. A document jam exists.
3. A pocket-full condition exists.
4. A late-stacker-select condition exists.
5. A short-document stop condition exists.
6. A document-gap stop condition exists.
7. A disengage operation or a runout has occurred.
8. Any motor-stop condition exists.

Control I/O Instructions

Control I/O (CIO) instructions occupy four bytes of main storage and consist of:

1. The op code.
2. The device address.
3. The function specification that defines the primary control function to be performed in the selected I/O device.
4. A B1-D1 field that defines the detailed control function. This detailed function may be taken directly from the B1-D1 field or it may be generated effectively by adding the content of a base register (addressed by B1) to the displacement (D1).

Pocket Select Instructions

The Pocket Select (CIO) instructions are accepted by the serial I/O channel as soon as they are issued and no condition code is set. The program must issue such an instruction for each document. From the time a Read From 1259 instructions ends, 24 milliseconds (minimum) are available to issue a Select Pocket 0 instruction, and 52 milliseconds (minimum) are available to issue a Select Pocket 1 through Reject instruction. Failure to issue the instruction within that time results in a late-stacker-select condition which stops the machine. This condition lasts until the Start key is pressed.

NOTE: If burst-mode devices such as tape or disk units are operating together with the 1259, burst-mode operations should not be started (even though the CC2 interrupt announces that the CPU is free for burst-mode operations) until the 1259 has been disengaged and Stacker Select instructions have been issued for all follow-up documents.

In normal operation, any pocket can be selected for any document. However, when the Auto Select indication is given for a document, this document is routed to the reject pocket and a Stacker Select command may be omitted. If a command is given for an auto select document, the command must be Select Reject Pocket, otherwise a mis-sort occurs. The auto-select condition arises:

1. When the program has failed to issue a Read From 1259 instruction for a document.
2. When a document of less than 5.50 in (140 mm) but more than 3.55 in (90 mm) length is detected.

3. When a document longer than 8.75 in (222 mm) is detected.

Engage/Disengage Instructions

The Engage 1259 instruction must precede all others to obtain document feeding. This feeding, however, begins only when the Start key has been pressed. The Engage instruction may be given either before or after the Start key is pressed. If given prior to start, the instruction will be stored in the machine and is executed when the Start key is pressed.

The Disengage 1259 instruction stops document feeding. One unread document will be in motion and will require processing after the Disengage instruction is given. A delay ensures that all documents which are under, or have passed, the read head are routed to their pockets before the machine stops.

Test I/O and Branch Instructions

Test I/O and Branch (TIOB) instructions occupy four bytes of main storage and consist of:

1. The op code.
2. The device address.
3. The function specification that specifies which condition is to be tested in the addressed I/O device.
4. The B1-D1 field from which a branch address may be derived via either direct address generation or effective address generation. If a particular condition is found to be present, the program branches to the address defined by B1-D1; if the condition is absent, the program continues with the updated instruction address located in the program status word.

The TIOB instructions are used to check on the various Field Valid indications as well as on special conditions such as auto select, document to be read, etc. If an error is detected, the valid condition for a given field is suppressed and thus the program will not branch. The valid condition lasts until the leading edge of the next document is sensed by the read head. The sorter-is-stopped condition continues until the error is corrected and the Start key is pressed.

Read Transmission Error: This error denotes parity deficiency. If even parity is set, odd parity will be treated as read transmission error; if odd parity is set, even parity will be treated as read transmission error. The parity setting is determined by a wire strap within the I/O interface connector. A parity error renders the I/O not operational to the program; this means that, after a transmission error, any XIO instruction is rejected with condition code 11 (binary) unless the error is tested. The test (TIOB instruction) resets the error condition.

Document To Be Read

The Document To Be Read indicator is turned on when a document is in motion between the separator and the read head. The indicator is turned off when the document is under the read head, provided that there are no other documents between separator and read head; the indicator is also turned off by a sorter-is-stopped condition. When the CPU is operating in non-time-sharing mode, the indicator can be used to determine the need for more read instructions, as it gives a preview of further documents which will eventually reach the read head. When the CPU is in time-sharing mode, the Read command should be issued unconditionally. In this case, the condition code signals whether or not the Read command was accepted.

Auto Select

The Auto Select indicator can be tested by the program because, when the auto select condition exists, the program must not issue a Pocket Select instruction other than Reject or a mis-sort will occur. Auto select documents enter the reject pocket automatically.

The Auto Select indicator is turned on when a document is sensed at the read head and no Read From 1259 instruction has been given. Likewise, this indicator is turned on for documents longer than 8.75 in (222 mm), or when documents of more than 3.55 in (90 mm) but less than 5.50 in (140 mm) are detected at the read station. The indicator remains on until the next document arrives, or if the main drive motor stops.

Document Under Read Head

The Document Under Read Head indicator is turned on when the leading edge of a document is sensed at the read head; the indicator is turned off when the trailing edge of the same document passes the read head. If a document reaches the read head before a Read instruction is given, the document is auto-selected. An auto select document is not read and, therefore, instructions concerning the reading and sorting of this document need not be issued.

The Document Under Read Head indicator should be used to determine the earliest possible pocket-select timing. When the indicator turns off, the document has passed the read head and a pocket can be selected.

Sorter-Is-Stopped Indicator

This indicator is turned on when either a jam or late-stacker-select condition exists, or if the drive motor is stopped. The indicator can be utilized to facilitate on-line correction of these errors. When the 1259 Start key is pressed, the indicator turns off.

In a sorter-is-stopped condition, the 1259 should be disengaged to prevent loss of documents when the machine is made ready again.

Field Valid Indicators for 1259 Models 1 and 31 (E13B Font)

Process-Control Field Valid: This indicator turns on when the process-control field has been completely read without errors. The indicator turns off when

the leading edge of the next document is sensed at the read head. The indicator does not turn on if:

1. Any of the characters in the process-control field (including the special symbols) are unreadable.
2. Special symbols (except the dash) are missing or out of sequence.
3. The processing unit does not store each character or symbol transmitted to it.
4. The process-control field is not selected for transmission to the processing unit.
5. The main-storage read-in area in the processing unit becomes filled before the trailing symbol of the process-control field is transmitted.

Amount Field Valid: This indicator turns on when the amount field has been completely read, without errors. The indicator turns off when the leading edge of the next document is sensed at the read head. The indicator does not turn on if:

1. Any of the characters in the amount field (including the special symbols) are unreadable.
2. Special symbols are missing or out of sequence.
3. The field is missing.
4. The field length is invalid.
5. The processing unit does not store each character or symbol transmitted to it.
6. The amount field is not selected for transmission to the processing unit.
7. The main-storage read-in area in the processing unit becomes filled before the trailing symbol of the amount field is transmitted.

Account-Number Field Valid: This indicator turns on when the account-number field is read correctly. The indicator turns off when the leading edge of the next document is sensed at the read head. The indicator does not turn on if:

1. Any of the characters in the account-number field (including the special symbols) are unreadable.
2. Special symbols (except the dash) are missing or out of sequence.
3. The field is missing.
4. The field length is invalid (fixed-lengths only).
5. The processing unit does not store each character transmitted to it.
6. The account-number field is not selected for transmission to the processing unit.
7. The main-storage read-in area in the processing unit becomes filled before the trailing symbol of the account-number field is transmitted.

Transit-Number Field Valid: This indicator turns on when the transit-number field is read correctly. It turns off when the leading edge of the next document is sensed at the read head. The indicator does not turn on if:

1. Any of the characters in the transit-number field (including the special symbols) are unreadable.
2. Special symbols (except the dash) are missing or out of sequence.
3. The field is missing.
4. The field length is invalid (the transit-number field must contain eight digits).
5. The processing unit does not store each character and symbol transmitted to it.
6. The transit-number field is not selected for transmission to the processing unit.
7. The main-storage read-in area in the processing unit becomes filled before the trailing symbol of the transit-number field is transmitted.

Serial-Number Field Valid: This indicator turns on when the serial-number field is read correctly. The indicator turns off when the leading edge of the next document is sensed at the read head. The indicator does not turn on if:

1. Any of the characters in the serial-number field (including the special symbols) are unreadable.
2. Special symbols (except the dash) are missing or out of sequence.
3. The field is missing.
4. The processing unit does not store each character transmitted to it.
5. The serial-number field is not selected for transmission to the processing unit.
6. The main-storage read-in area in the processing unit becomes filled before the trailing symbol of the serial-number field is transmitted.

Field Valid Indicators for 1259 Model 32 (CMC-7 Font)

The CMC-7 type font allows for seven data fields. A valid-field indicator that can be tested by the program is associated with each of these fields; these indicators are labeled Field 1 Valid, Field 2 Valid, Field 3 Valid, Field 4 Valid, Field 5 Valid, Field 6 Valid, and Field 7 Valid. Each of these indicators turns on when its respective field (including bracketing symbols) is read without error, transmitted to the processing unit, and received by the processing unit.

The field must contain the correct bracketing symbols, and the proper symbol sequence must exist for the entire document. The symbols and symbol sequence are checked according to the code-line configuration installed on the machine.

Machine Stops

The machine stops automatically and is made not operational when:

1. A document of less than 3.55 in (90 mm) length is detected.
2. The read station photocell detects the absence of a document for 0.5 seconds (feed failure, empty hopper).
3. A jam is detected at the read head or further along the document path.
4. A pocket is full.
5. Improper document spacing occurs.
6. Any of the cover interlocks of the transport area is open.
7. A Pocket Select instruction for any document has not been issued.

The machine is made operational (ready) when the stop condition is corrected manually and the Start key is then pressed.

COMMUNICATIONS ADAPTER

INTRODUCTION

The Communications Adapter (CA) special feature enables the System/360 Model 20 processor to function as a remote terminal capable of point-to-point transmission under stored-program control. The device allows communication with all standard synchronous transmitter receiver units such as the IBM 1009 Data Transmission Unit, the IBM 7701 and 7702 Magnetic Tape Transmission Terminals, the IBM 1013 Card Transmission Terminal as well as the IBM 7710 and 7711 Data Communication Units.

The System/360 Model 20 with the Communications Adapter special feature is fully compatible with the synchronous transmitter receiver devices used with System/360 equipment and is the primary means of data interchange between the System/360 Model 20 and another System/360 Model 20, or System/360 Models 30, 40 etc. equipped with an IBM 2701 with a synchronous data adapter type 1.

Data transmission can only be between two stations at a time (point-to-point). Stations can be connected either by permanent private or leased lines or by common carrier (dial) network. The Communications Adapter operates in half duplex mode on either two- or four-wire telephone lines. Four-wire operation results in a shorter line turn around time.

The Communications Adapter is programmed in much the same way as any other I/O device used with the System/360 Model 20. Transmission operations are time-shared with other primary input/output and processing operations. The communications adapter also uses the System/360 Model 20 interrupt feature to signal the CPU that an instruction has been completed or that acknowledgment for a transmitted record has been received. The transmission time rates depend on the line facilities available and on the type of modem used. Up to 600 bytes per second (4800 baud) can be transmitted or received. Data is transmitted in the 4-of-8 code and it must therefore be translated from or into EBCDIC. Either the Translate instruction or an IBM utility program can be used for this translation. Transfer instructions must be issued for each record transmitted. The instruction specifies the direction of transfer, the length of the assigned record area in main storage, and the leftmost address of the record in storage. Keys, lights, and switches are located on an operator panel adjacent to the CPU console.

CA INSTRUCTIONS

The following instructions are used to program the CA:

Type	Op Code	DA	FS
CIO	9B	5	1 Set Transmit Mode CIO
CIO	9B	5	0 Set Receive Mode
CIO	9B	5	2 Send EOT
CIO	9B	5	3 Inhibit Audible Alarm
XIO	D0	5	4 Transmit
XIO	D0	5	2 Receive
TIOB	9A	5	1 Test for current message in error
TIOB	9A	5	0 Test for CA busy
TIOB	9A	5	5 Test for EOT received
TIOB	9A	5	4 Test and Branch on BCD

Summary of TIOB Instructions

The use of the TIOB instructions is summarized in the following text and later under "Operating Principle" and "Transmit Operations".

Test for Current Message in Error

An error condition in the CA indicates that a faulty record has been either sent or received. However, the error condition does not stop the CA or render it non-operational; the only effect is that the faulty data is not set into core storage. To facilitate immediate retransmission, the error condition is testable. If the error condition is found to be on, the program branches to a subroutine that initiates retransmission; if the error condition is off, the program continues with the next sequential instruction (the address of this instruction is contained in the PSW).

If the programmer intends to check each message, he must use the Test for Current Message in Error instruction at interrupt time. If the CA is transmitting (send run status), the error condition means that a non-4-of-8 code was sent out. If the CA is receiving (receive run status), the error condition means that one of the following faults has occurred:

1. A non-4-of-8-code character was received
2. The odd/even counts do not match
3. The received record exceeded the length that was specified in the Receive instruction
4. The incoming character was not serviced by the micro-program
5. The longitudinal redundancy check shows an error.

If the same faulty message is transmitted (or received) two more times, the CA stops and the ready status is lost. At that time, the check lights indicate the type of error for the last transmission (or reception). The status of the CA (send or receive) is not affected.

The error condition is reset with the next XIO instruction (Transmit or Receive) that is issued, regardless of whether the error is corrected or not. The error condition is also reset by the next error-free message that is received.

Test for CA Busy

The busy condition is testable so that the program may avoid issuing one instruction while another is still in progress. The CA becomes busy when an instruction is issued and it remains busy until that instruction has ended. When the busy condition is found to be on, the program branches to a subroutine; if the busy condition is off, the program continues with the next sequential instruction.

When either of the Set Mode (CIO) instructions is issued, or when the Transmit (XIO) instruction is issued, the CA remains busy until a reply from the other station is received. However, character phase (synchronization) must be established between both stations before a reply can be received. Thus it is possible for the CA to remain busy indefinitely because of loss of character phase. The CA remains busy indefinitely also when the other station has received the same message three times in error and has, therefore, a check stop condition.

When a Receive (XIO) instruction is issued, the CA remains busy until either the End of Transmittal Record (EOTR) signal or an acknowledge signal is received from the transmitting station. In all cases, the busy condition is reset at interrupt time.

Note, however, the following exception: When the Set Receive Mode instruction is issued while the Auto Answer/Disconnect switch is in the auto answer position, the CA gets no busy condition; the program can then proceed with the next instruction. The Test on CA Busy instruction has a different function to that described previously when the switch is in the auto answer position; the instruction disables the Auto Answer/Disconnect feature (when issued) and no branch occurs.

Test for EOT Received

The End of Transmission (EOT) condition is testable so that the program can take appropriate action when an EOT signal is received. The Test for EOT Received instruction should be part of the interrupt routine because an EOT signal can be expected at any time and the interrupt time is the earliest possible moment at which the reply from the other station can be tested. When an EOT signal is received, the run status of the CA (either send or receive) is dropped and the program must change its normal course; for example, it must restore the run status if continued transmission (or reception) is desired. The EOT condition indicates that the other station has terminated the data exchange. If one station sends the EOT signal, the other station responds with EOT automatically and, therefore, the EOT condition can also indicate a positive acknowledgement.

When the EOT condition is found to be on, the program branches to a subroutine; when the EOT condition is off, the program continues with the next sequential instruction. The EOT condition is reset when the next Set Mode instruction is issued.

Test and Branch on BCD

To allow the CA to communicate with Synchronous Transmitter/Receiver (STR) devices that use only 63 characters (such as the IBM 1030 Data Collection System), the Binary/BCD switch must be in the BCD position. When the CA is transmitting data to such a device, the EBCDIC codes 0100 0000 (space) and 0111 1010 (colon) must both be translated into the same 4-of-8 code, that is, code 0842; when the CA is receiving data from an STR device, the 4-of-8 code NXOR is recognized as invalid (data check) and code 0842 must be translated into EBCDIC code 0100 0000 (space). In this way, the 64- and 63-character sets are made compatible.

The translating program caters for these deviations from the normal one-for-one translating pattern by using the Test and Branch on BCD instruction to find out whether the Binary/BCD switch is in the BCD position. If the switch is so, the translating program branches to a routine which translates the codes "colon" and "space" as described.

Operating Principle

To explain the basic operating principle an inquiry/answer routine is described. One CA is termed "local", another "remote", and automatically produced control signals are denoted "auto" in this example.

Description: When line connection is established between two CAs, both devices send idle characters (auto) for the purpose of obtaining synchronization. The local program issues a Set Transmit Mode (CIO) instruction to get the local CA into send run status. Send run status is only obtained when synchronization is established (indicated by the Character Phase light).

Meanwhile, the remote program has issued a Set Receive Mode (CIO) instruction to get the remote station into receive run status. The receive run status can also only be obtained when synchronism exists. Therefore, the Set Mode instructions can be issued either before or after synchronization because they are only completed if and when synchronization exists.

When the local CA is in send run status, an inquiry signal (auto) goes to the remote station. This inquiry signal initiates an interrupt in the remote station.

The remote station subsequently issues a Receive (XIO) instruction which causes the acknowledge 2 (ACK 2) signal (auto) to be sent back. The ACK 2 signal represents the answer to the inquiry and it initiates an interrupt in the local CA.

The local program then issues a Transmit (XIO) instruction which causes a start of record 1 (SOR 1) signal (auto) to go to the remote station. SOR 1 is immediately followed by data. The end of the data record is marked by an end of transmittal record (EOTR) signal which contains the Longitudinal Redundancy Check (LRC) character (auto).

The reception of the EOTR causes interrupt in the remote CA. The remote program then issues another Receive (XIO) instruction which causes the acknowledge 1 signal (auto) to be sent back as a reply if the record was received correctly. Otherwise, an error signal is sent.

Reception of ACK 1 causes an interrupt in the local CA and thus either a new Transmit (XIO) or a Send End of Transmission (CIO) instruction may be issued. If EOT is received at the remote CA, the remote CA answers also with EOT and that ends all transmissions. The automatic signals that precede each record alternate with each transmission, that is, the first record is preceded by SOR 1, the next by SOR 2, the following by SOR 1 again and so on. The acknowledgement alternates correspondingly with ACK 1, ACK 2, ACK 1. Since both stations keep track of this odd/even count no record can be "lost" unnoticed. The control signals (SOR 1, ACK 1, EOTR etc.) are not transferred to the main storage.

Transmit Operations

Set Transmit Mode Instruction (CIO)

The CA must be in send run status to be able to transmit. To get it into send run status, the Set Transmit Mode instruction must be issued. Normally this instruction is used only once per transmittal batch. An inquiry signal is sent out to the remote terminal and the instruction can be completed only when the ACK 2 signal is received. When this signal is received, the CA is in the send run state and interrupt is requested.

The Set Transmit Mode instruction can be issued before or after synchronization (character phase) is established, however, the instruction can only be processed when synchronization exists. Synchronization is indicated by the turned on Character Phase light (not by interrupt). The CA busy condition is turned on when the instruction is started and it is turned off upon completion.

Transmit Instruction (XIO)

The interrupt after the Set Transmit Mode instruction indicates to the program that now the transmit instruction can be issued. All odd numbered records are automatically preceded by the SOR 1 signal and all even numbered records are preceded by SOR 2. After the execution of the Transmit instruction has been initiated, the CA busy condition is turned on and it stays on until the record has been transmitted and a reply (ACK 1, ACK 2) has been received at the local CA.

When a reply is received, interrupt is requested. At interrupt time, the CA error condition and the EOT condition should be tested to find out what type of reply has been received. The conditions can be tested by the respective TIOB instructions. If none of these conditions exist, the last message was received free of errors at the remote station and therefore the program can issue the next Transmit instruction, or the End of Transmission instruction. One of these instructions must be issued within 3 seconds or a processor time out occurs.

If the error bit is on, the CA retains the send run status and the previous record can be transmitted again by issuing the previous Transmit instruction. It is possible that the remote station operator has stopped his CA and pressed the EOT-key. In that case the EOT bit is on and the local CA is out of send run status. To resume transmitting, the program must issue Set Transmit Mode again. The Set Mode instruction resets the EOT condition. Similarly, the error condition is reset by the next Transmit instruction.

Send EOT Instruction (CIO)

The Send EOT instruction is used to indicate the end of the transmission of a single batch of messages (records).

The local CA must be in send run to be able to transmit the EOT signal.

The CA busy condition is turned on when the instruction is executed and it stays on until an EOT signal is received in response.

When it is received, the local CA drops out of send run and an interrupt is requested.

At interrupt time, the EOT and the error condition should be tested by the respective Test I/O and Branch instruction.

A turned on EOT condition confirms the correct reception of the EOT signal at the remote station.

If the error condition is on, the send EOT instruction must be issued again because the previous was not received correctly.

Receive Operations

Set Receive Mode Instruction (CIO)

The CA must be in receive run status to be able to receive.

To get into the receive run condition, the Set Receive Mode instruction must be issued.

The instruction can be issued before or after synchronization is established but it can be processed only when synchronization (character phase) exists.

The Set Receive Mode instruction is completed when an inquiry signal is received. When the inquiry signal is received, the CA is in receive run status and interrupt is requested.

Receive Instruction (XIO)

The interrupt, caused by the reception of an inquiry signal, informs the program that now a Receive instruction can be issued. This Receive instruction automatically sends an ACK 2 signal to the inquiring station. The CA busy condition is turned on when the Receive instruction is started and it stays on until the instruction is completed. The instruction is completed when the End of Transmittal Record (EOTR) signal is received. This EOTR signal is sent automatically when the field length counter in the transmitting station goes to zero.

When the EOTR signal is received, interrupt is requested. In the interrupt program the Error as well as the EOT condition should be tested. If none of these conditions are on, the record was received correctly and the program can proceed with the data translation. When the received data has been processed, a Receive instruction must be issued to send back the acknowledgement which the transmitting station is expecting. If the received record was odd numbered (the first, third, fifth etc.) the ACK 1 signal is sent back; otherwise, it is the ACK 2 signal.

If the EOT condition is on, an EOT signal was received and therefore the receive run status is dropped. However, an EOT signal is sent back automatically as acknowledgement.

If the error condition is found to be on, a faulty record was received. The receiving station keeps the receive run status when the error bit is on. Next, a Receive instruction must be issued to send back the expected acknowledgement. The acknowledgement (in that case) is an Error signal which informs the transmitting station that a repeat of the previous record is expected.

Error Conditions	Detected by
● Invalid Character Transmitted (non - 4 of 8 - character)	Transmitting CA
● Invalid Character Received (non - 4 of 8 - character)	Receiving CA
● Lost Record (odd/even counts do not match)	Receiving CA
● Wrong Length Record (more characters were received than specified in the Receive instruction)	Receiving CA
● Overrun (character was not serviced)	Receiving CA
● LRC Error (longitudinal redundancy check error)	Receiving CA

If the error occurs within the transmitting station, an error counter is set to one.

If the program repeats that same faulty transmission two more times (the error counter is at 3), the transmitting station stops with the Check Stop light and the alarm on but it keeps its run status. The ready status is lost.

When the same message has been received in error three times in succession, the error counter is at 3 and a check stop condition occurs. The Check Stop light and a combination of other check lights is on as well as the alarm. The ready status is lost, however, the run status is retained.

When the receiving station has a check stop condition (due to faulty transmission lines etc.), the transmitting station remains busy if the fault occurred somewhere outside the transmitter.

Programming Errors

The CA instructions are subject to the normal restrictions, that is, addresses that are derived from the B/D fields must not refer to the protected area in main storage (0-143), or exceed storage capacity. No instruction part may be located in the last main-storage position. Either the B-field of an instruction or the D-field can be zero but not both. Violation of these rules leads to a CPU stop.

The CA instructions must be issued in the correct sequence and at the correct time. When a sequence error occurs, the CPU is stopped and the CA Interlock light, the audible alarm and the error number 0110 in data register I is on. The error number identifies the CA. An interlock is caused by:

- Failure to issue Set Transmit Mode command prior to the first Transmit instruction.
- Failure to issue Set Receive Mode command prior to the first Receive instruction.
- Issuing a Set Mode instruction when the CA is in a run status.
- Issuing a Set Mode instruction while a message is still being received, or transmitted.

Other Errors: When a CA instruction is issued and the CA-Start key has not been pressed, no operation occurs and the alarm is sounded due to a not operational condition. When the CE-switches are not in the normal position, the audible alarm is sounded and data errors are incurred. When the Binary/BCD switch is not set to the mode in which the other terminal transmits, errors can occur. An instruction causes an error if the previous instruction resulted in a storage wrap-around.

NOTE: The program may neglect the testing of the error bit after a record has been received. However, it must be understood that only the "good" portion of a record is transferred to main storage. Since the CA keeps its run status when the error bit is on, a new instruction can be issued and that will reset the error bit. To avoid a processor time out condition, an instruction must be issued within three seconds after acknowledgement.

Condition Code

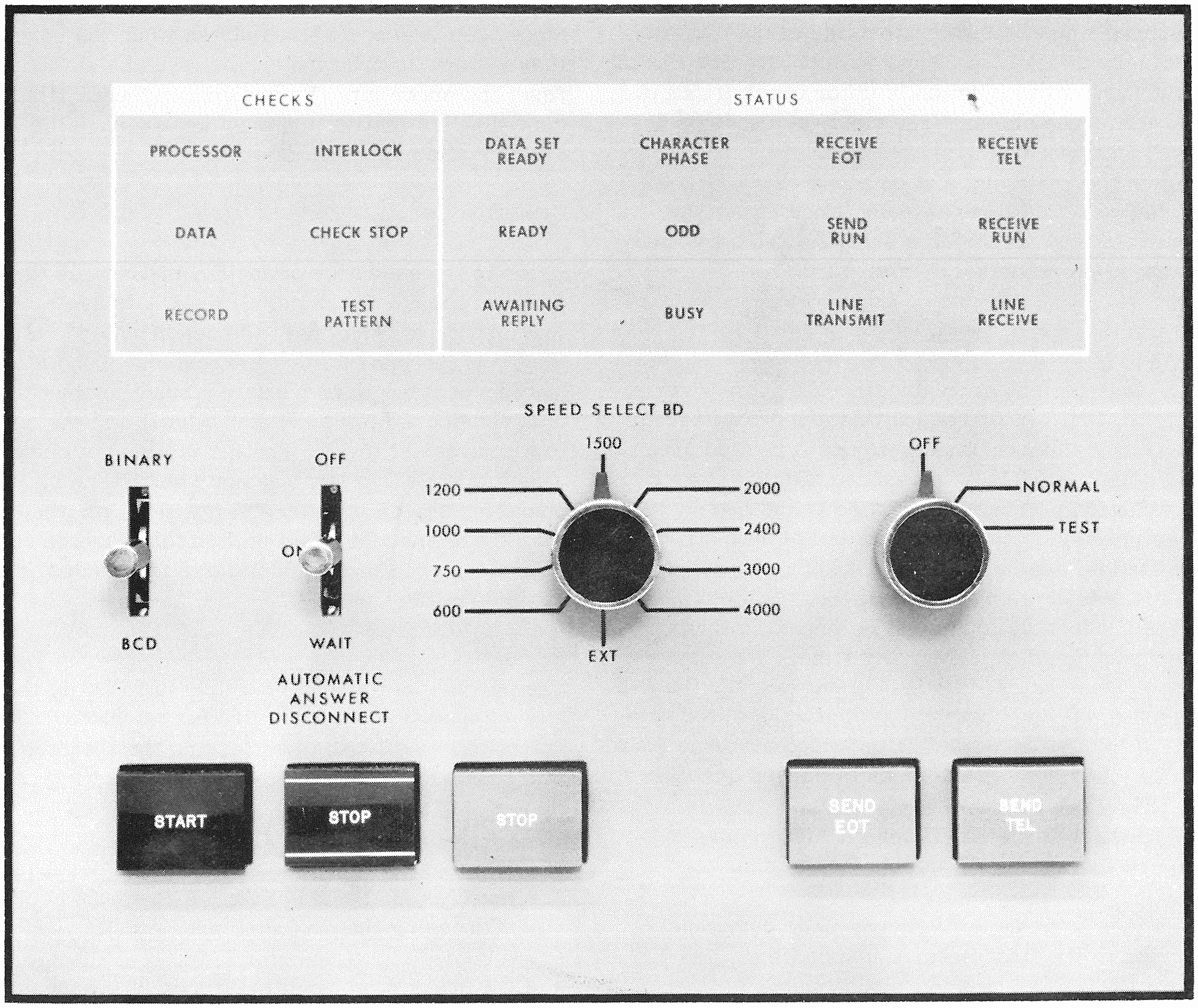
Prior to the execution of any CA instruction the condition of the CA is tested. If the CA is available, i.e., ready and not busy, the condition code is set to 00 and the instruction is executed. If the CA is busy (processing a previously issued instruction), it is considered to be not operational and the condition code is set to 11. The CA is also not operational when the CA-Start key was not pressed, when the operation switch is in the Off position, or when the data set or modem is not on. In that case no operation occurs and the audible alarm is sounded.

Audible Alarm

The purpose of the audible alarm is to notify the operator of conditions that require his attention. The alarm can be silenced by pressing the CA-Stop key. The ready status is then lost. The following conditions sound the alarm:

- CA programming error stop condition of the CPU (such as B/D-field of the CA instruction all zero, addressing the protected area etc.).
- CA check stop condition (three faulty messages transmitted or received).
- More than three seconds interval between instructions (processor time out).

  This alarm condition can be suppressed by the Inhibit Alarm instruction.
- CPU Process Check (indicates an error, such as faulty parity etc. within the CPU).
- Received EOT signal.
- Received TEL signal.
- CA in run status but character phase (synchronization) is lost. Alarm sounds until synchronization is obtained or until the CA-Stop key is pressed.
- CA not ready when any instruction that requires data transmission over the lines is issued.



01979

Figure 27. CA Console

NOTE: When the alarm is sounded the ready status is lost.

An exception is the processor time out condition which retains the ready status. The ready status is restored and the check lights are reset by pressing the CA-Start key.

Inhibit Audible Alarm (CIO)

The Inhibit Audible Alarm instruction suppresses the automatic alarm, which normally occurs three seconds after the completion of a CA operation if another is not initiated within that interval. This suppression is cancelled when the next Receive Record, Transmit Record, or Send EOT instruction is issued.

The instruction does not silence the audible alarm under any manually-resettable error conditions, or during an auto answer disconnect cycle. The execution of this instruction does not cause an interrupt and it does not turn off the busy condition.

Check Lights

Figure 27 shows the various check lights and Table 18 gives their interpretation.

Interlock Light

The Interlock light indicates a CA programming error stop. When a program error stop has occurred, the display lights of data register U and L (on the CPU console) show the Op code and the display lights E, S, T, R show the address of the CA instruction in which the error has occurred. Data register 1 displays the error number 0110 to identify the stop as a CA-program error.

Processor Check Light

NOTE: This light is not to be confused with the Process Check light on the CPU. The processor check condition has no effect on the operation of the CPU.

A processor check occurs under the following conditions:

- When the CA has received a message and is prepared to send a reply, it initiates an interrupt. If after three seconds (processor timeout) the program has not come up with a Receive instruction (which constitutes the reply), the Processor Check light is turned on.

- The same condition arises when the CA has transmitted a message and received a reply upon which it initiates an interrupt. If after three seconds the program has failed to respond with a Transmit instruction, the Processor Check light is turned on.

In addition, the audible alarm is normally turned on. If the Inhibit Audible Alarm instruction is executed before the processor check condition signal drops, the audible alarm is suppressed. However, as soon as the processor check signal drops, the Processor Check light is turned on. The CA does not loose its ready status when a processor check occurs.

The Processor Check light is turned off only when the required program instruction (Receive, or Transmit) is executed.

Table 18. Check Lights Interpretation 01980

Interlock Light	Check Stop Light	Data Light	Processor Light	Record Light	Interpretation
	X	X			Invalid character transmitted to remote terminal.
		X			Invalid character received in message.
				X	Incorrect longit. Redundancy check comparison
		X	X	X	Overrun of main storage area for incoming data.
		X	X		Overrun of received character (char. not serviced).
	X	X		X	Message Parity (Start of Record signal fails to match odd/even count).
X			X		Previous Transmit Instruction caused Storage Wrap-Around

Data Light/Record Light

These two lights are used in combination with the other lights to distinguish the various errors from each other.

Check Stop Light

During transmissions, the validity of all messages is constantly being monitored. An error count of 3 messages turns on the Check Stop light and the audible alarm while the start latch is being turned off. (Ready goes off.)

The audible alarm can be silenced by operating the Stop key. Pressing the Start key will turn off the Check Stop light and reset the error counter to zero. Reception of a good message has the same effect, i.e., the ready status is restored. Since the CA can attempt three repetitions of the bad message before it turns on the Check Stop light, this light in combination with the Data, Processor, or Record lights indicates the type of error for the third attempt.

Test Pattern Light

Indicates that a continuous sequence of control leader signals from the remote terminal has been received.

Status Indicators (Figure 27)

Data Set Ready

Indicates that power is on in the data set, the line is connected, and the switches are not in Test position.

Ready

Indicates that power is on in the System/360 Model 20 CPU, the CA CE switches are in the normal operating position, the data set is ready (not required for a Set Receive Mode instruction), and that the CA Start key has been operated.

The CA Ready indicator is turned off by any of the following:

- CA Stop Key. Any operation that is in progress when the stop key is pressed continues to completion.
- CA Check stop.
- Received TEL signal. (The alarm is sounded.)

Awaiting Reply

Indicates that the CA has transmitted an EOTR signal and is waiting for a reply from the receiver.

Character Phase

Indicates that synchronism with the remote station, a prerequisite for all operations, has been achieved. The CA attempts to achieve character phase when system power is turned on if the CA Operational Switch is in the Normal position. Character Phase is achieved when the data set is ready.

Busy

Indicates that an operation which causes data exchange over the line is in progress.

Odd

Indicates the odd/even count of records received or transmitted; it provides the primary method of checking for the loss of a complete record.

Receive EOT

Indicates that an EOT signal originated remotely or a remote reply to a locally originated EOT signal has been received. With the Auto Answer Disconnect switch on, the line is disconnected 30 seconds after this light comes on unless the program continues the operation before this time has elapsed. The alarm is sounded when an EOT signal is received.

Receive TEL

Indicates that a telephone (TEL) signal from a remote station or a remote reply to a TEL signal (originated at the CA) has been received. This means that one operator wants to get into voice communication with the other operator. All operations are halted and the ready status is lost when the Receive TEL light is on. The audible alarm is sounded.

Send Run

Indicates that the CA is set as a transmitting terminal.

Receive Run

Indicates that the CA is set as a receiving terminal.

Line Transmit and Line Receive

Indicate that data or control characters are actually on the line.

Stop

Indicates that the CA is not in a ready status.

Keys and Switches (Figure 27)

Operation Switch

A rotary switch with three positions, NORMAL, OFF, and TEST. NORMAL and OFF are the "in use/out of use" setting for the CA. In Test position, a continuous sequence of control leader signals is transmitted. The CA can be reset by turning this switch to Off and back to ON. This also silences the alarm.

Start Key

Used to place the CA in a ready status and under certain conditions to turn off the audible alarm.

Stop Key

The Stop key is used to remove the CA from the ready status and under certain conditions it also silences the audible alarm. If the Stop key is operated while the CA is in the process of transmitting or receiving a message, the transmission is allowed to continue to completion. However, the CA does not send a reply in response to the last message and it does not start a new message; the CA keeps its run condition.

After the Stop key has been operated, the CA cannot be automatically disconnected. This allows check stop analysis or other manual interventions, such as changing programs etc.

Speed Select Switch

This switch determines the transmission rate. It must be set to a speed which is agreeable to the line facilities that are available. The speed ratings are calibrated in terms of baud.

Baud Rate	Characters Per Second
600	75
750	93.75
1000	125
1200	150
1500	187.5
2000	250
2400	300

The EXT position must be selected if a data set with internal clock is to be used.

NOTE: Although the CA is capable of receiving and transmitting at a rate of 4,800 Baud, i.e., 600 characters/second, the speeds above 300 char/sec can at present not be utilized due to limitations on available common carrier line facilities.

Auto Answer/Disconnect Switch

Enables the Auto Answer/Disconnect special feature.

Binary/BCD Switch

Setting determines the number of valid 4-of-8 code combinations for transmission or reception and the performance of intermediate Longitudinal Redundancy Checks (LRC).

In the Binary position, 64 code combinations are valid and no intermediate LRC is performed.

In the BCD position, 63 code combinations are valid for reception; code ROXN is invalid and code 2480 is decoded as "space". Intermediate LRC is performed after line codes 148N or 280N. For transmission, both codes ROXN and 2480 are encoded as 2480.

Refer to the Tables in the Appendix for code conversion. The purpose of the BCD mode is to avoid as much line turn around time as possible. This is accomplished by sending as long a record as possible. The intermediate LRCs are transmitted to allow the program on the receiving end to use portions of a long record while the transmission is still in progress. Any portion that checks out as valid can thus be processed.

Send EOT Key

Permits the operator to manually initiate transmission of an EOT signal. This function is independent of the run condition and is operative when the CA is ready but not busy. In normal operation the program terminates the transmission by using the send EOT instruction. In certain cases an operator may wish to initiate the same function manually. If due to the program, the not busy periods between CA instructions are very short, it is advisable that the operator first press the CPU-Stop key and then the EOT key to get the signal through. However, it must be understood that the transmitting CA stays in the run condition and does not get an interrupt when the EOT response from the receiving station is received.

Send TEL Key

When operated, originates a TEL signal to inform the remote operator that verbal communication is requested. This signal is returned automatically to the originating CA; the returned signal turns on the TEL indicator.

Reset

The CA can be reset by turning the Operation switch to the Off position. The CA is also reset when the CPU-Reset key is pressed. Otherwise the execution of the Set Receive Mode or Set Transmit Mode instructions establishes all necessary originating conditions in the CA.

Code Translation

The 4-of-8 code is translated to the internal machine code (the EBCDIC in the System/360 Model 20) by means of the IBM program "Communications Adapter Input/Output Control System (CIOCS)." This macro routine is an IBM designed and tested program which relieves the customer of the translation task. The CPU presents the translated character to the CA for transmission and the CA checks the character for 4-of-8 validity. During reception, the reverse operation takes place.

Auto Answer/Disconnect Operation

This feature is used for three different purposes, for automatically answering an incoming call (auto answer), for automatically disconnecting the line 30 seconds after transmissions are ended (auto disconnect), and for keeping the CPU in a wait state allowing processing only after a call has come in (wait).

Auto Answer

The auto answer operation is enabled by the program when the Auto Answer Disconnect switch is in the On position and a Set Receive Mode instruction is issued. The program can continue with the next instruction immediately since the CA is not busy although the Set Receive Mode instruction has not been completed.

When a call comes in, the data set answers automatically and completes the connection. The CA then attempts to obtain synchronization for 30 seconds. If synchronization is not obtained within that time, the data set disconnects the line and the alarm is sounded. Interrupt is not requested after disconnection. When synchronization is established, the Set Receive Mode instruction is completed and interrupt is requested.

After the program has enabled the auto answer feature by issuing a Set Receive Mode instruction (with the switch in Auto Answer position) it can also disable the feature. To do this the program must issue a Test CA Busy (CIO) instruction.

If the data set has not started to answer a call when the Test For Busy instruction is executed, the CA prevents the data set from answering and the program does not branch. If the data set has started to answer a call during the execution of the instruction, the program branches because it is too late to stop the auto answer feature and the call must be accepted. In the On position, the CA is disconnected 30 seconds after it has received a call if it could not get into the receive run status within that time. It is also disconnected 30 seconds after issuing or receiving an EOT signal. Disconnection also occurs when check conditions that require manual intervention arise.

If an auto answer receiver wants to permit other calls to come through after a previous operation, his program must provide an idle loop for 35 seconds after EOT and then issue another Set Receive Mode instruction. The loop makes certain that the previous line is disconnected.

Auto answer receive programs must make use of the System/360 Model 20 interrupt feature to detect that a call has come in. A Test and Branch on Busy command cannot be used for this purpose because of its modified meaning.

Auto Answer Wait

When the Auto Answer/Disconnect switch is in the Wait position, operation is the same as when this switch is in the Auto Answer position except that the program cannot continue with the next sequential instruction after executing a Set Receive Mode instruction. The CPU enters a waiting state and awaits an incoming call. When the call is received, the program continues.

The CPU process meter does not run during the waiting state. The CPU may also be removed from the waiting state by operating the Stop key or System Reset key on the CPU console. The CA does not automatically branch back into the wait state. If desired, the programmer may cause a branch back to the Set Receive Mode instruction to enter the wait state.

When the Auto Answer/Disconnect switch is in the Off position, the auto answer/disconnect functions are disabled.

Operating Procedures

Start for Transmitting Without Auto Answer

1. Press CPU Power On key, press all I/O Start keys to get all I/O devices ready. CA Operation switch must be in Normal position.

2. Load program, then press CPU Start key to execute program.
3. Turn on data set, press CA Start key, check all switch settings.
4. Dial receiving party, when voice communication has been established, switch data set from talk to data mode. Hang up telephone receiver.

Start for Transmitting with Auto Answer On at Remote CA

1. When transmitting, the Auto Answer switch must always be OFF at the local CA.
2. Get CPU and all I/Os Ready. Load program and start execution just as before.
3. Turn on data set, press CA Start key, check all switch settings.
4. Dial receiving party but do not use voice communication.
5. Avoid pressing TEL key because this causes a Stop at the remote terminal. (Remote station may be unattended.)
6. After dialing, switch data set to data mode and hang up telephone receiver without using voice communication.

Start for a Receive Operation Without Auto Answer

1. Press CPU power On key, get I/Os ready.
2. When call comes in, get into voice communication with the remote operator.
3. Turn on data set, set CA switch to Normal, make certain that the Auto Answer switchis OFF.
4. Switch data set from talk to data mode.
5. Load program, press CPU and CA Start keys.

Start for Receive Operation with Auto Answer On at Local CA

1. Press CPU Power On Key, get system ready.
2. Load Program.
3. Start CA, turn on data set, check all switches, set Auto Answer switch to ON.
4. Press CPU Start key to execute program.

Wait with Auto Answer On at Local CA

Starting procedure is the same as for receive with auto answer except the Auto Answer switch must be in Wait position.

CA Timing Conditions

Instruction and Execute Phase

XIO	Transmit	190 usec
XIO	Receive	190 usec
TIOB	Busy	120 usec
TIOB	Error	120 usec
TIOB	EOT	115 usec
CIO	Set Receive Mode	150 usec
CIO	Set Transmit mode	135 usec
CIO	Send EOT	135 usec
CIO	Inhibit Aud. Alarm	120 usec

The service phase time is 70 μsec per character (maximum). The code translation which is handled by the CIOCS can be calculated as follows: $250 + 54 N = \mu$sec.

250 is a factor which takes care of address calculations (indexing). N represents the number of characters in the record (254 characters maximum). Thus the maximum time for translating 254 characters is 14 ms.

Time Sharing

All standard speeds which can be selected by the Speed Selection switch allow the CA to operate simultaneously with the other I/O devices and processing operations.

Line Turn-Around Delay

In half duplex lines, the effective data transmission speed depends on turn-around delay. This turn-around delay varies with the lines used, but turn-around delays of 250 ms per change of direction can be considered standard. After a record has been transmitted, the remote station "turns around" to acknowledge after which the local station turns around to transmit again.

During this total turn-around time of 500 ms, processing or I/O functions can be performed.

The CA also allows "full duplex" operation, which is, however, actually half duplex on four wires. Even in this case there is a maximum time loss of two characters where each change of direction is from 3.3 ms up to 26.0 ms depending on the line speed selected. With the higher speed lines in "full duplex", turn-around time can practically be disregarded in connection with long records.

BINARY SYNCHRONOUS COMMUNICATIONS ADAPTER

INTRODUCTION

Telecommunication, the ability to exchange up-to-date information between remote locations, is a valuable asset in the fast-paced modern business world of industry and commerce. Therefore, powerful tools such as electronic data processing systems should not be limited to strictly local input and output. Nowadays, as business activities expand and companies establish branches further afield, the rapid exchange and processing of vital data is of ever-increasing importance. Thus, more and more emphasis is being placed on Tele-processing, which offers interesting prospects to all users of data processing systems.

The Binary Synchronous Communications Adapter (BSCA) for the System/360 Model 20 is a Teleprocessing device which is virtually independent of such restricting factors as the necessity for code translation (due to a limited character set), complex synchronizing procedures, slow transmission speeds, etc. The adapter can be field installed in the Model 20 and adds to the capabilities of this system, since any type of binary data can be transmitted and received in time-sharing mode (at speeds up to 2,400 bits per second) with other input/output and processing operations. At transmission speeds above 2,400 bits per second, the BSCA operates in burst mode.

The BSCA meets the requirements of medium- and high-speed communication systems by offering the following:

1. Transmission of the entire Extended Binary-Coded-Decimal Interchange Code.
2. Transmission of the entire United States of America Standard Code for Information Interchange.
3. Transmission of any binary data (entire programs in any language, packed decimal data, random data, etc).
4. Transmission speeds from 600 to 50,000 bits per second.
5. Block-dividing of messages to adapt to optimal conditions on telephone lines of a given quality.
6. Powerful error-checking methods.
7. Optional features that allow a variety of different operating modes.

The BSCA can communicate with the synchronous adapter units of the IBM 2701 Data Adapter Unit and the IBM 2703 Transmission Control (equipped with the corresponding synchronous features) that provide the primary means of data exchange with System/360 Models 30, 40, 50, etc. A Model 20 equipped with the BSCA can also communicate with another Model 20 that is fitted with the BSCA.

The BSCA operates over two-wire or four-wire telephone lines that may be either privately-owned, leased, or switched (that is, part of a common carrier dial-up network*). The transmission technique is half-duplex (that is, in one direction only at a time). The adapter can communicate in point-to-point or in multipoint fashion. In point-to-point operation, data exchange is between two stations; in multipoint operation, a master station can select or poll one of a number of slave stations that are interconnected on a leased or private line.

The Binary Synchronous Communications Adapter is fully program-controlled by ten basic instructions. These instructions are similar to the other I/O instructions of the Model 20 which are described in this manual and, to the Model 20, the BSCA is a further I/O device. However, considerable flexibility is introduced into the program by the binary synchronous philosophy; this philosophy features precisely-defined control character sequences that "frame" each message at its beginning and end. Such a message, with its control sequences at both ends, is moved byte-by-byte from main storage in the Central Processing Unit (CPU) to a buffer from where it is transmitted bit-by-bit (serially) to the remote terminal. Here, the stream of bits enters a similar buffer where the data is re-assembled into characters and moved eventually into main storage. The control characters trigger certain actions in both stations so that each message actually controls its own mode of transmission. Thus, the main controlling elements are the control character sequences which enable each message to function as a self-controlling entity.

Transmission Codes and Interfaces

When ordering a BSCA for the System/360 Model 20, the customer must specify either the USASCII or the EBCDIC as his line code; one line code only is permissible on any adapter.

Two interfaces, the data set interface and the data station interface, are available for connecting the BSCA to a modem. (The modem converts dc signal output from the BSCA into modulated carrier signals suitable for transmission over telephone lines.) The data set interface complies with the Electronic Industry of America (EIA) requirement RS 232B; the interface allows U.S. data sets, the IBM 3977 Modem or equivalent equipment to be connected. These devices allow operation on voice-grade telephone lines at speeds ranging from 600 to 2,400 bits per second.

The data station interface (also termed digital interface) is part of the BSCA High Speed Feature and is used only when that feature is installed.

* All references in this section to switched networks apply at present to U.S.A. only.

NOTE: The BSCA does not impose on the customer the use of any particular modem, but he must consider certain aspects when planning his BSCA installation. Modems that do not provide bit timing can be used without restriction. Modems that are equipped with a clock and use relative phase angle modulation (4-phase modulation) offer the advantage of operating on narrow-band communication channels; however, these modems are sensitive to certain steady bit patterns. In particular, modems that transmit the serial BSCA bits in pairs of bits (dibits) can lose synchronization when a sequence of more than 40 consecutive one-zero (or zero-one) patterns is transmitted. Likewise, high speed data stations cannot transmit a series of 500 (or more) zero bits unless the stations are equipped with a universal scrambler. For guidance, refer to Systems Reference Library General Information - Binary Synchronous Communications, Form A27-3004.

United States of America Standard Code for Information Interchange

This code consists of seven data bits and one parity bit (Figure 28). Thus, an eight-bit character is formed which must have odd parity. The parity bit is supplied automatically by the BSCA as needed and, therefore, the parity bit position must always be zero in all USASCII characters that are to be transmitted from main storage. The automatically-supplied parity bit is transmitted over the line; in the receiving BSCA, however, the bit is removed automatically and entered into main storage as a zero bit. The USASCII character is transmitted low-order bit (bit 1) first, bit 2 next and so on, ending with the parity bit. The USASCII table in Appendix L shows all bit configurations; the characters within thick lines are control characters, while those in the shaded areas cause line turnaround (change of transmission direction) whenever they are preceded by the control character DLE. This operation is explained later under "Control Sequences (Basic BSCA)."

Extended Binary Coded Decimal Interchange Code

This code consists of eight data bits (Figure 28). EBCDIC characters are transmitted low-order bit (bit 7) first, bit 6 next and so on, ending with bit 0. The EBCDIC table in Appendix L shows all bit configurations; the thick lines enclose tne control characters, while the shaded areas contain characters that cause line turnaround (change of transmission direction) when they are preceded by the control character DLE. See "Control Sequences (Basic BSCA)."

High-Order Position ↓ Low-Order Position ↓

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

Figure 28. Sequential Numbering of Bit Positions in USASCII and EBCDIC 01256

Table 19. BSCA Instruction Set 01262

Type	Format	Op Code	DA	FS	Function
XIO	SS	D0	5	0	Transmit and Receive
XIO	SS	D0	5	1	Receive Initial
XIO	SS	D0	5	4	Receive only
XIO	SS	D0	5	8	Transmit only
CIO	SI	9B	5	2	Enable BSCA
CIO	SI	9B	5	3	Disable BSCA
CIO	SI	9B	5	6	Store Current Address
CIO	SI	9B	5	7	Store Sense Information
TIOB	SI	9A	5	0	Test and Branch on Any Indicator Set
TIOB	SI	9A	5	8	Test and Branch on Busy

DESCRIPTION (BASIC BSCA)

The information in this section pertains to the functional characteristics of a BSCA without features. Features are described separately because they significantly influence certain functions.

Instruction Set and Control Characters (Summary)

The instruction set for the basic BSCA is shown in Table 19, while EBCDIC and USASCII control character codes are given in Tables 20 and 21 respectively.

Table 20. EBCDIC Control Character Codes 01263

Bit Position 0 1 2 3 4 5 6 7	Mnemonic	Function
0 0 0 0 0 0 0 1	SOH	Start of Heading
0 0 0 0 0 0 1 0	STX	Start of Text
0 0 0 0 0 0 1 1	ETX	End of Text
0 0 1 0 0 1 1 0	ETB	End of Transmittal Block
0 0 1 1 0 1 1 1	EOT	End of Transmission
0 1 1 *	Column 3	Change of Direction Modifier
0 0 1 1 1 1 0 1	NAK	Negative Acknowledgement
0 0 1 0 1 1 0 1	ENQ	Enquiry
0 0 0 1 0 0 0 0	DLE	Data Link Escape
0 0 1 1 0 0 1 0	SYN	Synchronization Character

* "Do not care" positions.
These bits are used at the discretion of the programmer

Table 21. USACII Control Character Codes 01273

Bit Position P 7 6 5 4 3 2 1	Mnemonic	Function
P 0 0 0 0 0 0 1	SOH	Start of Heading
P 0 0 0 0 0 1 0	STX	Start of Text
P 0 0 0 0 1 0 0	EOT	End of Transmission
P 0 0 0 0 0 1 1	ETX	End of Text
P 0 0 1 0 1 1 1	ETB	End of Transmittal Block
P 0 1 1 *	Column 3	Change of Direction Modifier
P 0 0 1 0 1 0 1	NAK	Negative Acknowledgement
P 0 0 0 0 1 0 1	ENQ	Enquiry
P 0 0 1 0 1 1 0	SYN	Synchronization Character
P 0 0 1 0 0 0 0	DLE	Data Link Escape

* "Do not care" positions.
These bits are used at the discretion of the programmer

Operating Principle

Operations in the BSCA are started by initialization, that is, by bidding for the line. The initialization assigns the right to begin transmission to one or the other station. If the BSCA gets the line, it can begin to transmit; if, however, it does not get the line, it must first receive the bid from the other station before it can start its transmission. The actual data exchange is accomplished with Transmit and Receive instructions that are issued in both stations. Before each XIO instruction (Transmit/Receive) is executed, the condition code is set to indicate to the program whether the instruction can be executed or not. After each XIO instruction is executed, an interrupt is requested to provide a means for checking the results. If an error occurs, details of the cause for rejection (in the form of sense information) are available at interrupt time.

Initialization at Transmitting Station

The actual wire connection between two stations is either permanent (on private and leased lines) or it must be established by a dialing procedure (in switched networks). Dialing can also be accomplished under program control. (See "Automatic Calling Feature.")

After connection is established, an initialization procedure must be used. For reasons that are explained later in "XIO Instructions" under "Transmit and Receive", the initialization is accomplished by two different methods that ensure the proper rhythm of data exchange between the two stations. The method used depends on the intentions of the user of a particular station. If he intends first to send data to a remote station, he must use one method; if he intends first to receive data from the remote station, he must use the other method. Thus, initialization can be likened to starting on "one foot or the other", one station using one method and the other station using the opposite method; neither method represents an advantage or a disadvantage because, after initialization, each station transmits in turn.

Before any other action, the BSCA is enabled by issuing an Enable BSCA instruction. This instruction produces a positive level on the signal line 'data terminal ready' that runs from the BSCA to the modem or data set; this positive level is needed for the modem to become operational. No other preparation is needed because the BSCA has power on when the CPU has power on.

The transmitting BSCA then bids for the telephone line by issuing a Transmit and Receive instruction. The instruction reads out the actual bid from the CPU main storage and transmits it to the remote station. In the simplest form, the bid consists of an Enquiry (ENQ) character that is interpreted by the recipient as "Who are you?" A more sophisticated bid consists of some identification followed by the ENQ character which is interpreted as "I am xxx, who are you?"

The BSCA reacts to the Transmit and Receive instruction that is issued to send the bid by setting a condition code to inform the program how it may proceed:

Condition Code	Function
0 (binary 00)	BSCA is "available" and has accepted the instruction
1 (binary 01)	BSCA is "working" (it is still processing a previously-issued XIO instruction)
3 (binary 11)	Manual intervention is required because the modem or data set is not operational

If the BSCA was not enabled, a program error stop occurs when the Transmit and Receive instruction is issued. In this example, it is assumed that the BSCA has accepted the instruction; therefore, the BSCA activates the modem interface line 'request to send'. This causes the modem to set its carrier on the line and subsequently to respond with 'clear to send'. At this point, transmission over the telephone line begins with two automatically-supplied Synchronization (SYN) characters that provide synchronization with the other station, the actual bid follows next, and an automatically-supplied pad character (sent behind the ENQ character in the bid) ends the transmission. At this point, however, the instruction has not ended because the ENQ character causes the BSCA to turnaround, and to "listen" for a response from the remote station. Normally, the Transmit and Receive instruction ends only when a reply is received.

However, the BSCA spends only a limited amount of time on awaiting a reply. As soon as turnaround occurs, a receive timeout delay of 3-sec duration starts running. This timeout delay ensures that the BSCA cannot be hung-up indefinitely and unnoticed. The timeout delay is reset by every nonsynchronizing character that is received immediately after a SYN character. Since the BSCA automatically inserts a SYN character after a certain time interval during transmit operations, the 3-sec timeout delay (started in the receiving BSCA) normally cannot elapse when SYN characters are detected in the data stream. It is only if either all SYN characters, or all data, or nothing, is received that a timeout occurs to end the instruction. The SYN characters are not entered into main storage.

The end of a Transmit and Receive instruction is indicated by a request for an interrupt. As explained later, tests performed in the interrupt routine reveal the ending status of the instruction. With the acceptance of the Transmit and Receive instruction, the BSCA becomes busy; the busy condition ends when interrupt is requested.

Initialization at the Receiver

The receiving station prepares its BSCA to receive a bid from the transmitting station by issuing a Receive Initial instruction. Prior to this instruction, the BSCA must have been enabled, otherwise the Receive Initial instruction is rejected, whereby condition code 3 (binary 11) is set to indicate that the BSCA is not operational. The BSCA reacts to the Receive Initial instruction by setting a condition code to inform the program how it may proceed.

NOTE: If the modem has power off, the condition code for a Receive Initial instruction is set to 00 (available) and, therefore, modem power off is not noticed immediately.

In this example, it is assumed that the Receive Initial instruction is accepted. Thus, the BSCA listens for a bid. However, the BSCA does not become busy; the busy condition begins only with the actual arrival of data from the other station. With switched networks and the Receive Initial instruction, the timeout delay of 3 sec is started when the line connection is established, that is, when the modem sends the 'data set ready' signal to the BSCA; at this moment, the BSCA becomes busy and searches automatically for synchronization; if this cannot be obtained within 3 sec, a timeout occurs. With leased or privately-owned lines, the 3-sec timeout delay is started when synchronization (character phase) is established between two basic BSCA stations; however, a timeout occurs under any of the following conditions to avoid a permanent hang-up of the BSCA:

1. If no data is received
2. If a constant stream of SYN characters arrives
3. If data is received that does not contain a character (such as ENQ) to cause turnaround.

Since the BSCA is not busy when it awaits a bid (after the Receive Initial instruction was issued), a burst operation (tape, or disk storage) can interfere.

The end of a Receive Initial instruction is indicated by a request for an interrupt. This interrupt request is made under any of the following conditions:

1. When a bid has been received and stored in main storage
2. When a timeout occurs
3. When an interfering burst operation ends.

Interrupt

The interrupt indicates that an XIO instruction has ended. The BSCA interrupt is identified by the device address 5 and the function specification 6 that are stored in the old program status word. Since it is not known how a particular instruction has ended, the program must test certain indicators at interrupt time to find out whether the instruction has ended in normal or in abnormal fashion. The testing is simplified by use of the TIOB On Any Indicator Set instruction. If no indicator is set, the previous instruction has ended without errors, which means that either a bid, or a positive or negative acknowledgement, or a regular message, has been received. If one (or more) indicators are on, it is necessary to retrieve them from the unit in which they were generated and to store them into a main storage location where they can be inspected by the program. The Store Sense Information instruction is used for this task.

If an instruction has ended with timeout, the timeout flag in the sense information would be on. If a Receive Initial instruction has been interfered with by a burst operation that coincided with an incoming bid, the overrun flag in the sense information would be on. For details of all error situations, see "Sense Information."

If the error testing at interrupt time reveals that the initialization was successful, the data exchange may begin with the issue of a Transmit and Receive instruction.

The individual transmit and receive portions in corresponding instructions of both stations mutually overlap. This overlapping creates the proper rhythm of data exchange which is shown later in Figure 32.

Contention

On privately-owned or leased lines, it is possible for two stations to vie for the line at the same time. Therefore, priority is assigned to one station. Priority can be allotted by different methods; for example, the program of the priority station may ignore inquiries from the other station and react only to positive acknowledgements. Likewise, a higher bid-repetition rate may be used by the priority station than by the other station. Either method ensures that the non-priority station is forced to "listen" eventually to the priority station.

In switched networks, contention is normally not possible because, on other than on leased lines, the two stations are only connected when transmission is desired and in switched networks, the calling station has priority. However, if one station replies to a receive bid with an acknowledgement which indicates that reception cannot continue, contention is possible on switched lines too, because both stations remain connected after such a reply. For details, see "Control Sequences."

INSTRUCTION (BASIC BSCA)

The information in this section pertains to the functional characteristics of a BSCA without features. Features are described separately because they significantly affect certain functions.

XIO Instructions

Transfer I/O instructions occupy six bytes of main storage and consist of the operation code, the device address, the function specification, the address of the first (leftmost) byte of the main-storage input or output field, and the field length. The B1-D1 field can be used as the direct address of the input/output field, or an effective address may be generated by adding the contents of a general register (defined by B1) to the displacement (D1). The address thus derived must not exceed the storage capacity or violate the protected area, and must not be equal to zero as this causes a specification error. The field length is similarly derived from the B2-D2 field of the instruction and represents the true field length. The field length is variable from 1 to 4,095 bytes.

All XIO instructions set the condition code prior to execution and request an interrupt after execution.

Receive Initial

This instruction prepares the BSCA to receive an initial sequence (a bid) from the other station. When this instruction is issued, the BSCA remains available until synchronization is achieved (on leased lines) or until a bid arrives (on switched lines). When synchronization is established, or when the bid arrives, the BSCA becomes busy and the bid is stored into the main-storage field defined by the B1-D1 address. An interrupt is requested and the busy condition ends when the bid is stored or when a timeout occurs due to a faulty message (all SYN characters, or all data, or nothing) being received.

If the BSCA has accepted a Receive Initial instruction but a burst mode operation (tape, or disk storage) is started by the CPU before an initial sequence is received, it is possible that an incoming bid coincides with that burst operation. The instruction ends with an interrupt that occurs when the burst mode operation is completed. At that time, the sense information overrun flag is on and the first character of the prevented line bid is saved in main storage. However, if a timeout occurs in addition to overrun, this first character cannot be stored. A timeout during Receive Initial can occur only in leased- or private-line operations because the 3-sec timeout delay is started when synchronization is achieved, as opposed to switched-line operations where the timeout delay starts only when data is received. A receive initial operation can be cancelled by the Disable BSCA instruction, provided that this instruction is issued before an initial sequence is received.

Wait State: The Receive Initial instruction can be used to place the 2020 into a wait state, during which the CPU is stopped but is prepared to receive and, subsequently, to process an incoming call. As a prerequisite, the toggle switch on the BSCA console (see Figure 40) must be set to WAIT STATE; this can be done anytime before or during processing operations of any kind, because the CPU enters the wait state only when a Receive Initial instruction is issued. The REC INITIAL OR ADPREP light on the BSCA console comes on when the wait state begins. The light goes out when a call comes in (on switched lines) or when synchronization is achieved (on leased lines); at this point, the 3-sec timeout is started to prevent hangup. The wait state allows data exchange with an unattended station.

Transmit and Receive

In the half-duplex mode of data exchange that is used by the BSCA, transmission occurs in one direction only at a time. To avoid confusion, note that each station fulfils a primary function during data exchange, independent of the individual actions that occur in each station: the primary function of one station is to serve as a transmitter, while the primary function of the other station is to serve as a receiver. Because both stations employ the Transmit and Receive instruction to correspond with each other, two names are used in the following explanation to distinguish clearly between the primary functions and the actual operations of a specific station; one station is referred to as the "donor", the other is called the "recipient."

This clarification is especially necessary when two stations are both System/360 Model 20's equipped with the BSCA. The situation is somewhat different when one station is a Model 20 but the other station is a System/360 Model 30, 40, 50 etc; these latter models operate through the 2701 Data Adapter Unit or the 2703 Transmission Control, both of which feature the channel concept. With the channel concept, actions are started by a Start I/O instruction that basically reads out a channel command word which has a read or write function. These functions, however, are identical with the transmit and receive functions of the BSCA. Further explanations are, therefore, based on Model 20 operations.

The Transmit and Receive instruction serves a dual purpose as the name indicates, that is, it causes the BSCA to transmit and subsequently to turn around to receive. This dual function is needed because of the extremely fast response coming from the other station. The B1-D1 field of the Transmit and Receive instruction defines the leftmost byte of a main-storage field that represents an output area as well as an input area, while the field length (defined by B2-D2) pertains to the total length of this combined output/input field. The length is variable from 1 to 4,095 bytes, any portion of which may be chosen as the output field, with the remaining portion representing the input field. The donor station assigns the message that is to be transmitted to the output portion of the combined field, and reserves the input portion for the reply from the recipient. The recipient usually replies with any of the standard acknowledgements. See "Control Sequences (Basic BSCA)."

The recipient uses the Transmit and Receive instruction in a similar way, but for a different purpose; the recipient assigns its reply to the output area of the combined field and reserves the input area for the next message that it expects to receive.

Any message that is to be transmitted or received must be framed by control characters on both ends. These control characters activate certain functions in the BSCA when the message is read out from the main storage of the donor and, subsequently, they activate the same functions in the recipient upon arrival. See "Control Characters (Basic BSCA)." Figure 29 shows a combined field as it appears in the main storage of the donor and Figure 30 shows the corresponding field as it is used by the recipient.

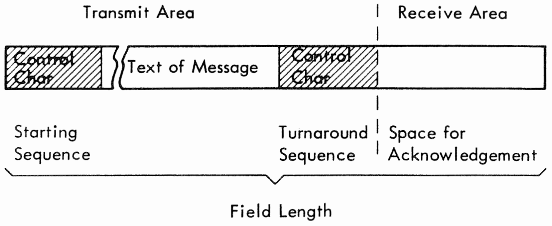


Figure 29. Transmit and Receive Area used by Donor 01264

When a Transmit and Receive instruction is issued, the output portion (transmit area) of the combined field is always dealt with first. For the donor, this means that the message is sent out bit-by-bit until the control character at the end of the message is detected. This control character triggers the turnaround function in the donor BSCA which means that, from that moment onwards, the donor temporarily becomes a receiver. The turnaround character thus marks the boundary between output and input areas.

With the detection of the turnaround character, the 3-sec timeout is started to prevent a hang-up situation in the event that nothing is received. The BSCA becomes busy when the Transmit and Receive instruction is accepted and this condition lasts until interrupt occurs. Interrupt is normally requested only when the reply has been received and stored into the input portion (receive area) of the combined field. As explained later, interrupt is also requested when certain abnormal conditions are detected.

A Transmit and Receive instruction is processed in exactly the same way in the recipient as in the donor. The only difference is in the logical purpose for which the instruction is used by the recipient; the recipient sends an acknowledgement first after which it turns around to receive the next message. Again, an interrupt is requested when this new message has been stored or when the timeout delay has elapsed if nothing was received.

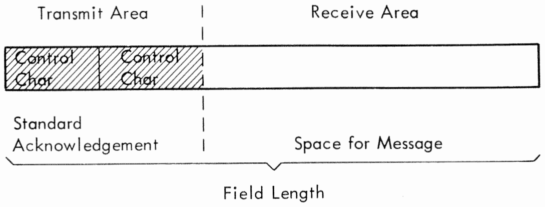


Figure 30. Transmit and Receive Area used by Recipient 01265

From the foregoing explanation, it may seem that the donor has an advantage over the recipient, in that the donor may send messages of any length while the recipient is limited to sending standard acknowledgements. However, this is not so; the recipient is allowed to send a message of any length in reply, provided that it has received the previous donor-message free of errors. This type of operation is termed "tête-à-tête" to distinguish it from the method of replying with standard acknowledgements.

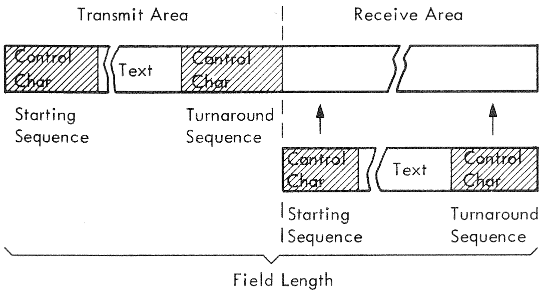


Figure 31. Transmit and Receive Area used in Tête-à-Tête Operations 01266

Tête-à-tête is a French expression meaning "head-to-head" in reference to a private conversation. In the operation of the BSCA, the term characterizes the full data exchange in the foregoing sense. The tête-à-tête operation speeds up overall throughput, because a message sent by the recipient implies that the previous reception was free of errors, thus omitting the actual acknowledgement. Only when a previous reception was faulty must the recipient send a negative acknowledgement to the donor. Figure 31 shows the main-storage field for a Transmit and Receive instruction that is used in tête-à-tête operations.

Since Transmit and Receive instructions are used by both stations, a conflict must be avoided; this is achieved by using two different initialization procedures. The donor starts with a Transmit and Receive instruction while the recipient begins with a Receive Initial instruction. Thus, an offset in the rhythm of data exchange is produced to ensure that the transmit portion of a given instruction is always faced with a receive portion at the other station (Figure 32). The Transmit and Receive instructions

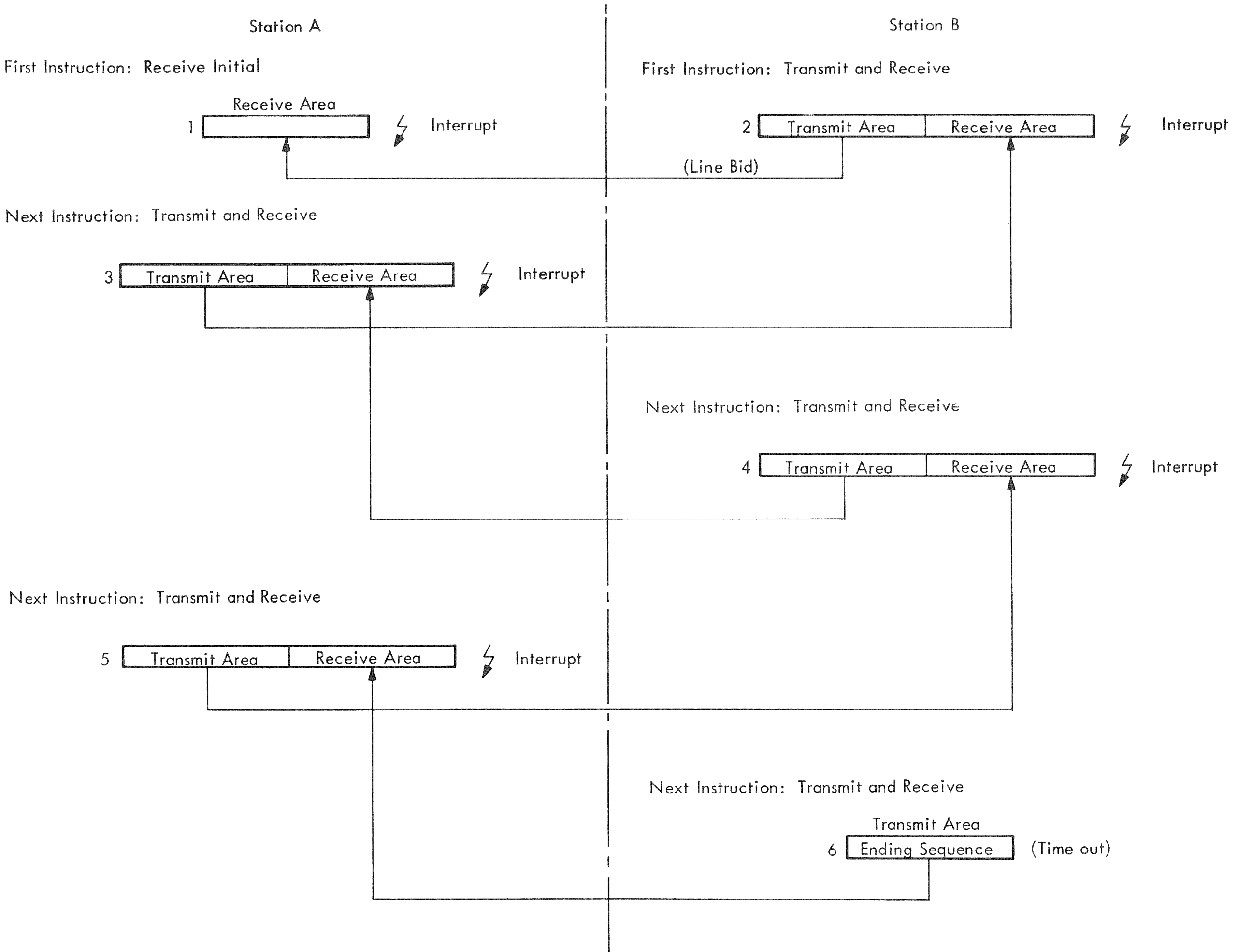


Figure 32. Data Exchange via Transmit and Receive 01274

used in both stations are not necessarily mirror images of each other. However, two types of errors (short record, and storage wraparound) must be avoided.

The short record error arises when the message that is to be transmitted, or the message that is being received, is longer than the field length specified in the Transmit and Receive instruction. When a short record is detected, an interrupt request is generated immediately. The storage wraparound error arises when the storage capacity is exceeded during a transmit or receive operation. For this error, the interrupt request is also generated immediately. Both error cases are registered in the sense information, which is available at interrupt time.

Receive Only

The Receive Only instruction sets the BSCA to receive mode and the 3-sec timeout is started. The instruction is used when a previously-issued Transmit and Receive instruction has ended with timeout and the receiving BSCA assumes that the transmitter is only delayed temporarily, but will start sending again eventually. Thus, the Receive Only instruction serves as a timeout extender. When the instruction ends with timeout again, nothing is received. The Receive Only instruction can be issued as often as desired. The BSCA becomes busy when it has accepted this instruction, and the busy condition lasts until an interrupt occurs.

Transmit Only

The Transmit Only instruction is used for sending a message to which a reply is either not expected or not desirable. For example, when it is necessary to send a negative acknowledgement because operations are held up by certain circumstances, it may be advantageous to use a Transmit Only instruction to avoid receiving any further data from the other station. The BSCA becomes busy when it has accepted this instruction, and the busy condition ends when interrupt is requested.

CIO Instructions

Control I/O instructions occupy four bytes of main storage and consist of the operation code, the device address, the function specification, and the B1-D1 field; this field further specifies the control function that is to be performed in the BSCA according to the rules for direct or effective addressing. These rules are given previously in this manual in "Central Processing Unit" under "Addressing". When a Control I/O instruction is issued, it is executed immediately and no condition code is set. However, if the BSCA is working (that is, processing a previously-issued XIO instruction), the Control I/O instruction enters the I/O-busy loop and is executed when the BSCA becomes available.

Enable BSCA

The Enable BSCA control instruction has a general "turn on" function, and is a prerequisite for all BSCA operations. When the instruction is issued, the control signal line 'data terminal ready' that runs from the BSCA to the modem or data set becomes positive; this positive level makes the modem operational. The Enable BSCA instruction must be issued before the first Transmit and Receive, Receive Only, or Receive Initial instruction, or otherwise a program error stop occurs when any of these instructions are issued.

Disable BSCA

The Disable BSCA control instruction, when issued, makes the 'data terminal ready' signal line negative. The instruction can be used to withdraw from an intended data exchange that was started, provided it is issued prior to the reception of the bid or acknowledgement from the other station. If Disable BSCA is issued while a transmission is in progress, the current operation continues to completion and all operations then cease because the BSCA enters a reset state.

Store Current Address

The current address is the address of the last-received byte, updated by plus one. Thus, it defines the beginning of the next combined input/output field in main storage or the end of the previous message. The Store Current Address instruction moves the current address to the storage location addressed by the B1-D1 field of the instruction. By means of this instruction, search operations can be avoided and, therefore, inspection of the last-received message is simplified.

Store Sense Information

At interrupt time, information on the ending status of the previously-executed XIO instruction is available. When a TIOB on Any Indicator Set instruction shows that the previous XIO instruction was ended by an abnormal condition, the Store Sense Information instruction can be used to retrieve the error indications from the inaccessible auxiliary storage and to store these indications into that byte of main storage that is addressed by the B1-D1 field of the

Store Sense Information instruction. The following conditions can thus be stored:

1. EOT received.
2. Transmit/receive mode error.
3. Short record.
4. Storage wraparound.
5. Overrun.
6. Timeout.
7. Cyclic or longitudinal redundancy (CRC/LRC) check.
8. Parity check.

For explanations of these checks, see "Sense Information."

TIOB Instructions

Test I/O and Branch instructions occupy four bytes of main storage and consist of the operation code, the device address, the function specification, and a B1-D1 field from which the branch address can be derived according to the rules for direct or effective addressing. The TIOB instructions test for a particular condition in the BSCA and if the condition is present, the program branches to the address defined by B1-D1. If the particular condition is not present, the program continues with the next sequential instruction address located in the program status word. When TIOB instructions are issued, no condition code is set.

TIOB on Any Indicator Set

The TIOB on Any Indicator Set instruction can be used to find out whether an instruction was ended by an abnormal condition. If any of the check conditions that are contained in the sense information (timeout, overrun, etc.) are present, the program branches. In this manner, the program can determine whether it is necessary to store the sense information. The instruction is executed immediately, except when the BSCA is processing a previously-issued XIO instruction. In that case, the TIOB on Any Indicator Set instruction enters the I/O-busy loop and is executed as soon as the BSCA becomes available.

TIOB on Busy

The TIOB on Busy instruction is used to find out whether the BSCA is busy (that is, processing a previously-issued XIO instruction); if so, the program branches. The BSCA becomes busy when it has accepted an XIO instruction. The busy condition ends when an interrupt occurs. Thus, the BSCA-busy condition is identical with the BSCA-working condition, except for the following difference in the use that is made of both indications:

1. The condition code for "working" is set, after the XIO instruction has been attempted, to inform the program why the attempt was unsuccessful.
2. The TIOB on Busy instruction can be performed to find out whether an XIO instruction would be successful if it were issued.

The TIOB on Busy instruction is always executed immediately, regardless of the BSCA condition.

Control Characters (Basic BSCA)

The control characters are used either singly or in groups of two consecutive characters (thus forming a control sequence) to cause two actions in the BSCA: setting the BSCA to text mode, and setting the BSCA to control mode. Data can be exchanged in either mode; however, text mode is reserved strictly for text transmission, while control mode is used for preparatory and administrative purposes. For example, when the BSCA has called another station or when it has been addressed by another station, it is in control mode.

In control mode, all control characters are effective so that, if used, they activate certain functions (such as the setting of text mode) in both stations.

In text mode, control characters must not be used because they make the message invalid. Only specific control characters (and/or sequences) cause the BSCA to return to control mode when they are used at the end of the text. The return to control mode is always accompanied by turnaround so that the other station has an opportunity to reply.

The actions of the individual control characters are described in the following paragraphs, and the bit configurations of the control characters are listed in Tables 20 and 21.

Start of Heading (SOH)

The SOH character sets the BSCA to text mode and indicates that the data that follows is an identification of the message that will follow eventually. The heading can be a name, a text, or a number. The SOH character cannot be used in the message.

Start of Text (STX)

The STX character sets the BSCA to text mode and indicates that the data that follows is the actual text of the message. STX cannot be used in the text.

End of Text (ETX)

The ETX character causes the BSCA to leave text mode and to enter control mode. This mode change causes turnaround and both stations remain connected. ETX is commonly used to mark the boundary between the core storage transmit and receive areas.

End of Transmittal Block (ETB)

The ETB character is an alternate for ETX. ETB can be used to divide a message into blocks of optimal length for a telephone line of a given quality. The function of ETB is identical to that of ETX; ETB causes the BSCA to leave text mode but it indicates to the program that the next transmission will be the logical continuation of the one that has just ended.

End of Transmission (EOT)

The EOT character is used by the transmitting station to tell the recipient that the current transmission will not be continued. EOT can only be used in control mode; the character causes turnaround so that the receiver has an opportunity to start its own transmission. Both stations remain connected after EOT is sent. EOT must not be used in text mode.

Negative Acknowledgement (NAK)

The NAK character is used as a reply to tell the transmitter that the received message cannot be accepted because it was faulty. NAK can only be used in control mode; the character causes turnaround and both stations remain connected. The NAK reply is likely to be received from a station to which text containing control characters was sent in text mode.

Enquiry (ENQ)

The ENQ character can be used in text mode as well as in control mode. ENQ causes turnaround and both stations remain connected. In control mode, ENQ is used as the initial sequence or as a request for information on the status of the other station. The program should, therefore, react to a received ENQ by sending information that indicates whether transmission can continue or not. When ENQ appears in the data stream during text mode, both stations return to control mode and turnaround occurs.

Data Link Escape (DLE)

The DLE character is used in control sequences only; it cannot stand alone and must not be used in text mode. For details, see "Control Sequences (Basic BSCA)".

Synchronization Character (SYN)

The SYN character is automatically inserted into the data stream at the proper time to establish and maintain synchronization. The program has no influence on this SYN insertion but can use SYN in text mode or in control mode as a filler character, provided that such fillers are sent for less than 3 seconds. If SYN is sent for more than 3 seconds, a timeout occurs. SYN characters do not enter main storage.

Control Sequences (Basic BSCA)

Control sequences are formed by two consecutive control characters with no data between them. The sequences are: standard affirmative replies, turnarounds, and ending sequences. Control sequences can be used in control mode only in the basic BSCA (BSCA without features); they must not be used in text mode.

Standard Affirmative Replies

In normal data exchange, it is customary to acknowledge each received message with an affirmative reply. If the reply cannot be positive because the received message was faulty, a NAK character must be used as reply.

Odd-numbered messages (first, third, fifth, etc.) are acknowledged by ACK-1 and even-numbered messages are acknowledged by ACK-0. The abbreviation ACK stands for Acknowledgement, but the actual acknowledgement is realized by the DLE character as the leader, followed by specific bit combinations as the trailer. A third positive acknowledgement, which has a special meaning is Wait before Transmit (WABT); WABT indicates to the transmitting station that the receiver has a temporary not-ready condition.

Figure 33 shows the three standard affirmative replies in USASCII and EBCDIC. The question mark shown is a regular USASCII character, while the digits in the EBCDIC configuration are not numbers but represent a bit configuration in hexadecimal notation. The actual bit configuration for the question mark is the same as that for WABT (see Figure 41 in Appendix L). For the actual bit configuration of the hexadecimal expressions, see Appendix E.

All standard replies cause turnaround. Standard affirmative replies, however, need not necessarily be used; the customer may prefer to operate in tête-à-tête mode. In this case, another message is used as the positive reply, provided the previous transmission was received correctly. However, when the previous transmission was received in error, the NAK character must be used as the negative reply. Standard affirmative replies as well as NAK can only be used outside of text mode.

Function	USASCII		EBCDIC	
ACK-0	DLE	0	DLE	/70/
ACK-1	DLE	1	DLE	/61/
WABT	DLE	?	DLE	/7F/
	Leader	Trailer	Leader	Trailer

Figure 33. Standard Affirmative Reply Sequences 01258

Start of Block Sequences

Since there are no alternating acknowledgements in tête-à-tête operations, two sequences are used as a programming convention to identify each message. These sequences can either precede the heading or they can be used in the heading. They represent the functions Start of Block (SOB)-1 and SOB-2. The sequences consist of the DLE character followed by a trailer character; for the bit configuration of the trailer, see the USASCII and EBCDIC tables in Appendix L. Both sequences activate no function in the BSCA, but concern only the program.

Turnaround Sequences

Turnaround sequences are used to effect turnaround when the BSCA is in control mode. These sequences must not be used in text mode. The sequences consist of the DLE character followed by any of the "Column 3" characters. Column-3 characters are located in the third column of the USASCII table in Appendix L, and are shown also in the shaded areas of the EBCDIC table in the same appendix. The Column-3 characters alone have no effect.

The character sequence DLE/Column 3 can be used as an acknowledgement to which a special meaning may be assigned by agreement. In addition, the low-order bits of these characters are "do not care" bits, that is, the bits can be used by the programmer, provided that the total character retains odd parity in USASCII and in EBCDIC. If the total character is turned to even parity (by insertion of bits into the "do not care" area), it loses its control function.

Special Turnaround Sequences

Stop Acknowledge (DLE/SAK): The SAK character is a reply which indicates that the previous message was received free of errors but that further messages cannot be received due to interfering or not-ready conditions (such as programmed halts, empty hoppers, forms end) in the I/O equipment. SAK is a Column-3 trailer character and must be preceded by DLE. The sequence can be used only outside of text mode.

Reverse Interrupt (DLE/RVI): The RVI character is a reply which indicates that the previous message was received error free; however, RVI requests that the data transfer is reversed so that the receiving station is given an opportunity to send an important message. The recipient of an RVI must then provide storage space into which the message from the other station may enter. RVI is commonly used in non-tête-à-tête types of data exchange where one station primarily receives messages and refrains from transmitting data other than acknowledgements. To RVI, the former donor answers with ACK, whereupon the situation is reversed. RVI is a Column-3 trailer character and must be preceded by DLE. The sequence can only be used outside of text mode.

Ending Sequences

Ending sequences can only be used in control mode. Two ending sequences are provided, one for ending the transmission without breaking the line connection, and the other for ending with line disconnection. The character sequence DLE/SAK is an acknowledgement which indicates to the transmitter that the last message was received free of errors, but that reception cannot continue; both stations remain connected after DLE/SAK is sent and turnaround occurs. The character sequence DLE/EOT is used to announce End of Transmission; after DLE/EOT, the line is disconnected (in switched networks) or operations cease (on private and leased lines).

	Valid Combination	Valid Combination	Valid Combination	Valid Combination
High Speed Feature	Installed	No	No	No
Station Selection Feature	No	Installed	No	Allowed if no Auto Calling Feature
Automatic Calling Feature	No	No	Installed	Allowed if no Station Selection Feature
Internal Clocking Feature	No	Allowed	Allowed	Installed
Full Transparency Feature	Allowed	Allowed	Allowed	Allowed
Intermediate Block Checking Feature	No	Allowed	Allowed	Allowed

Figure 34. Optional Features Configuration 01269

Programmed Halts

Since the Model 20 equipped with the BSCA is a rather complex terminal, it is quite possible that a programmed halt occurs due to a not-ready condition in one of its I/O devices. Such halts can be caused by card jams, forms check, empty hopper, etc., and they interfere with BSCA operations. To avoid stopping the BSCA without giving any indication of the reason for this, the Model 20 program must send an ending sequence to the remote BSCA prior to issuing the halt instruction.

If the local BSCA is the donor, STX/ENQ (or data ending with ENQ) must be sent. To this, the recipient responds with NAK. When NAK is recognised, the donor must send EOT (or DLE/EOT on switched lines) and then the halt instruction can be issued.

If a condition requiring a halt comes up in the recipient, the recipient must send EOT (or DLE/EOT) after it has received the ending character (ETB or ETX) of the current message. After that, the halt instruction can be given.

The BSCA can receive the EOT even after the CPU has stopped. The received EOT causes the BSCA to become not-operational, and any new BSCA-XIO instruction is rejected with condition code 3 (binary 11). This condition code reminds the program to check the EOT bit that is on in the sense information. The EOT bit and the not-operational condition are reset by the test.

Initialization Sequence

The initialization sequence is used to bid for the line and is actually an indirect control sequence since it only initiates data exchange. The sequence consists either of an ENQ character alone which is interpreted as "Who are you?", or of some identification followed by ENQ which is interpreted as "I am xxx, who are you?" The sending of an initialization sequence does not set the BSCA to text mode.

BSCA OPTIONAL FEATURES

To accommodate special needs, optional features are provided for the Binary Synchronous Communications Adapter. All of these features can be field-installed, but the installation of some features excludes others. Figure 34 lists features and shows valid combinations for the same BSCA; the shaded areas provide a guide to which features can be added when a specific feature is already installed.

With the installation of certain features, instructions and control characters, additional to those described previously for the basic BSCA, are provided to operate them. The additional instructions (Figure 35) can be used only when the corresponding feature is installed. If an instruction is issued when the feature is not installed, the condition code is set to 3 (binary 11), meaning not operational, and a programming error stop occurs with error number 6 displayed in data register I on the CPU console.

Type	Format	Op Code	DA	FS	Function
XIO	SS	DO	5	2	Address Prepare
XIO	SS	DO	5	3	Auto Call
CIO	SI	9B	5	0	Disable ITB
CIO	SI	9B	5	1	Enable ITB
CIO	SI	9B	5	8	Store ITB Address

EBCDIC

Bit Position 0 1 2 3 4 5 6 7	Mnemonic	Function
0 0 0 1 1 1 1 1	ITB	Intermediate Block (Unit Separator)

USASCII

Bit Position P 7 6 5 4 3 2 1	Mnemonic	Function
P 0 0 1 1 1 1 1	ITB	Intermediate Block (Unit Separator)

Figure 35. Additional Instructions and Control Characters for Optional Features 01268

Internal Clocking Feature

The Internal Clocking Feature supplies the clock pulses that are required for establishing and maintaining bit synchronization. The feature provides for transmission speeds of 600, 1,200, 2,000 or 2,400 bits per second, the required speed being set by the customer engineer when the feature is installed.

This feature is required whenever the modem or data set does not provide the clock pulses. The feature is required also if the other station (or stations) on the same network use internal clocking; conversely, if the BSCA is equipped with the internal clocking feature, the other stations must be similarly equipped. When the feature is installed, four SYN characters are automatically supplied in addition to the two regular SYN characters to establish and maintain bit synchronization.

High Speed Feature

The High Speed Feature is required for transmission speeds above 4,800 bits per second (such as 19,200 or 40,800 bits per second), up to a maximum of 50,000 bits per second. When this feature is installed, the data set interface is changed to a data station interface (digital interface).

The high speed feature significantly alters the operational characteristics of the BSCA if the host CPU is a 2020 Submodel 2. With the high speed feature installed in this submodel, any BSCA - XIO instruction is executed in burst mode, that is, all bytes called for are transferred in one continuous string at the selected speed and the CPU is occupied exclusively with this transfer. When a BSCA burst-mode operation ends, interrupt is requested.

With a burst-mode operation in progress, only 1403 buffered-print operations, 2415 tape-rewind operations, or 2311 seek-access operations can be performed concurrently, provided they were initiated before the burst-mode operation commenced; all other input/output or processing operations are excluded. Conversely, an interlock prevents the start of a burst-mode operation as long as time-shared operations are in progress; if time-shared operations are being executed, a BSCA high-speed XIO instruction is rejected with condition code 2 (binary 10). Following the setting of this condition code, a special interrupt (identified by device address 15) is requested when all currently running time-shared operations have ended; the burst-mode operation may then be repeated because the CPU is now free to accept it.

If the high speed feature is installed in a 2020 Submodel 5, all BSCA - XIO instructions are executed in overlap mode; interrupt is requested at the termination of each XIO instruction. In overlap mode, BSCA high-speed operations overlap with other input/output operations and are at no time rejected unless errors have occurred. (The device address 15 interrupt will not be requested.) BSCA high-speed operations are executed in overlap mode independent of the setting of the overlap mode bit in the PSW.

Automatic Calling Feature

The Automatic Calling Feature enables the BSCA to originate dial calls by program control. The program-generated call is delivered to an externally-attached automatic calling unit that executes the dialing. Automatic calling units are used only in switched networks.

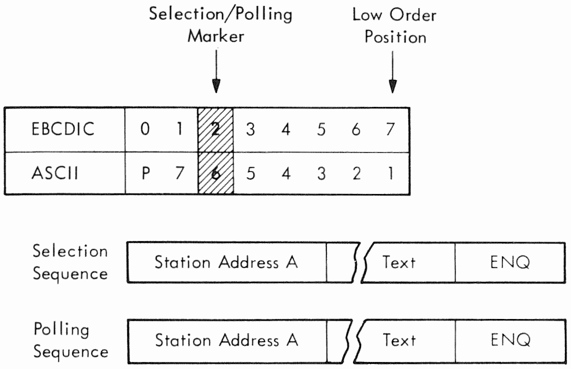
Prior to initiating a call, the Enable BSCA instruction must be issued to make the modem operational. The dialing information (the number of the desired subscriber) is transmitted to the automatic calling unit by means of the Auto Call instruction. When this instruction is issued, the condition code is set to 1 (binary 01) if the BSCA is processing a previously-issued XIO instruction. If the automatic calling unit has power off, or if the data key on the connected telephone set was not pressed, the condition code is set to 3 (binary 11).

When the Auto Call instruction is accepted, the four low-order bits of each byte of the output field (addressed by B1-D1) are transmitted to the calling unit. The number of half-bytes that are thus transmitted is specified by the field length. After transmission, an interrupt is requested. If no error flag is on, connection is established; if the timeout flag is on, the connection cannot be made because either the line is busy, or the wrong number was used.

Station Selection Feature

The Station Selection Feature is required whenever the BSCA is to be incorporated into a centralized multipoint network on leased or privately-owned lines. In a centralized multipoint network, all operations are initiated by the master station. The master station transmits to one or more slave stations by selecting them, and receives from a slave station by polling it. Thus, the initiative rests with the master station; the slave stations cannot begin to transmit without having been requested to do so. In a multipoint network, all stations involved must have the same speed, transmission code, and optional features. The Model 20 has no master station capability.

At initialization time, and always after the end of a transmission from a slave station, the master station has the sole right to transmit. The master



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

Figure 36. Selection and Polling 01267

station uses two different initial sequences, either a selection sequence or a polling sequence. The selection sequence causes a specific slave station to serve as the receiver, and the polling sequence requests a specific slave station to transmit to the master station. The configurations of both sequences are alike and consist of a station address, followed by data (that represents an identification) and by the ENQ character (Figure 36). To distinguish between polling and selection, bit position 2 (in EBCDIC) and bit position 6 (USASCII) are treated as "do not care" bit positions by the address decoding circuits. These bit positions are used to mark polling or selection. The BSCA is equipped with an address decoder for the station address. The station address is set up by the customer engineer when the feature is installed.

NOTE: When station addresses are set up, bit configurations identical to those of control characters must not be used, otherwise the receiving station will recognize both address and control character simultaneously. As Figure 36 shows, the bit configuration associated with a station address differs slightly, depending on whether polling or selection is specified. Neither of these two addresses (polling or selection) may be identical with a control character.

Address Prepare Instruction

To gain access to the multipoint network, the Address Prepare (Adprep) instruction must be issued after the BSCA has been enabled and the modem has power on. The instruction can be issued either before or after line connection has been established. With the BSCA console switch set to WAIT STATE, Adprep allows the CPU to enter the wait state, and also causes the BSCA to search for synchronization. Once synchronization is achieved, the 3-sec timeout begins to run; if synchronization is lost, however, this is not noticeable because the BSCA recuperates automatically. The BSCA monitors the line for a single EOT (sent in control mode), because this character indicates that currently-running transmissions have ended.

Throughout the synchronizing and monitoring operations, the BSCA remains available; it becomes busy only when a single EOT is recognized. The single EOT sets the monitoring BSCA to control mode and enables it to listen for either of its addresses (selection or polling). If either address is received within 3 seconds, the BSCA enters addressed mode, which means that it actively enters the multipoint network. The addressed state is indicated by a light on the BSCA console (see Figure 40). No data, other than the selection or polling address is transferred to main storage during monitoring operations; the transfer of the address is indicated by interrupt. The interrupt program must examine the marker bit in the address to find out whether the station has been polled or selected. (The meaning of the marker bit must be agreed between the subscribers.) If the station was polled, it must transmit to the master; if it was selected, it must receive from the master.

Because the BSCA becomes busy only when a single EOT is recognized, a burst mode operation can interfere with the Address Prepare operation during the not-busy period. The BSCA recognizes a single EOT also during a concurrent burst mode operation. However, when the incoming polling or selection address coincides with the burst mode operation, an overrun condition occurs, but the first character of the address is stored in main storage. A timeout occurs if the address is not received within 3 seconds after recognition of the single EOT; the timeout prevents a hangup in case this EOT was not sent for the explicit purpose of addressing a new station.

Once the BSCA is addressed, it enters text mode with every starting sequence (SOH or STX) and returns to control mode with every turnaround. An EOT sets all stations on the network to control mode.

Intermediate Block Checking Feature

The Intermediate Block Checking Feature allows the programmer to divide a message into small blocks, so that each block is checked separately. Turnaround (with acknowledgement of each intermediate block) is not needed and, therefore, overall throughput is increased. Without this feature, a check is performed only after the entire message has been transmitted.

The length of the intermediate blocks can be freely chosen. A block-divided message can be processed by the receiver while transmission is still in progress because each block (that is received free of errors) is made available by means of an intermediate interrupt. This interrupt is requested only in the receiving BSCA and it is identified by the function specification 2. The final interrupt (function specification 6) occurs only at the end of the entire message, that is, when turnaround occurs; this final interrupt resets any pending intermediate interrupt.

NOTE: If the channel mask bit in the program status work is temporarily disabled (zero) while Intermediate Block (ITB) messages are being received, intermediate interrupts are lost. However, the final interrupt remains pending and will occur when the channel mask is set to one.

The intermediate block checking feature is automatically disabled after power on or system reset. If the feature is to be used, the Enable ITB instruction must be issued. The feature is operational until either Disable ITB or system reset is performed. The Enable/Disable instructions are executed immediately, provided that the BSCA is not busy. In the busy condition, the instructions are held in the busy loop and are executed as soon as the BSCA becomes available.

The message is divided into blocks by the ITB character as shown in Figure 37. The ITB characters are recognized as normal data with no other function when the feature is not enabled.

When the intermediate interrupt occurs in the receiving BSCA, the last-received block can be used directly without testing for any errors. The error testing is superfluous because the intermediate interrupt is requested only when the received block is error-free; in addition, the TIOB on Any Indicator Set instruction cannot be used because the BSCA remains busy for the duration of the entire message and, therefore, the test instruction would be locked in the I/O-busy loop. If a block is faulty, no intermediate interrupt occurs for this and all following blocks. At final interrupt time (end of message), error testing can be performed.

During ITB operations, the receiving BSCA stores automatically the address of each ITB character. This address is updated by every block that is received free of errors, and thus it shows where the ITB character of the last-received error-free block is located in the main storage. If an error occurs, the ITB address is not updated but remains unaltered so that the address of the last "good" block can be made available at final interrupt time. The address is made available by issuing the Store ITB Address instruction. The instruction is executed immediately (even though the BSCA is busy) and moves the ITB address to the main-storage position addressed by B1-D1.

The block in which an error is detected is stored into main storage as if it were a good block, but any further blocks that belong to the message are not stored, even if they are received free of errors. However, the last character of the message is stored into the byte next to the last byte of the faulty block.

All blocks that belong to a message must be transmitted with one Transmit and Receive instruction. Transmitting a single block defeats the purpose of block checking and, if a single block does not end with the ETX character, a timeout occurs because there is no turnaround after ITB.

Intermediate block checking is inhibited by the Disable ITB instruction. Disable ITB, as well as Enable ITB, causes the ITB address to be reset to zero. Intermediate block checking can be used only at speeds up to 2,400 bits per second. Above this speed, there is insufficient time for the interrupt requests to be serviced.

Full Transparency Feature

The Full Transparency Feature allows the BSCA to transmit (free of restriction) any kind of binary data. The feature is required for the transmission of packed decimal data, entire programs, random data, etc. Transmission of this binary data is accomplished by making all control characters available for use as data; thus, a message becomes transparent.

The feature is activated by sending or receiving the starting sequence DLE/STX. This sequence places the BSCA into data mode and into transparent mode at the same time. When transparent mode is set, the following characteristic changes occur:

1. The BSCA recognizes neither individual control characters nor control sequences as anything but data with no other associated function.

2. All inserted SYN characters are automatically preceded by a DLE character.

3. A second DLE is automatically attached to every "data-DLE" to mark the data-DLE as such.

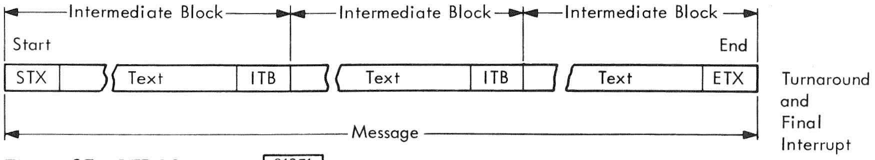


Figure 37. ITB Message 01271

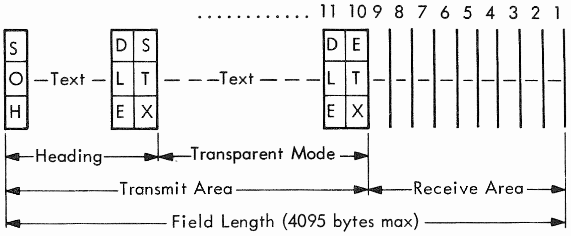


Figure 38. Transparent Message 01270

This second DLE, and the inserted DLE/SYN's, are automatically deleted upon reception and do not enter main storage. (Therefore, the program must not insert SYN characters as "fillers" as these will enter main storage.)

To leave transparent mode, the following ending sequences are available: DLE/ETX, DLE/ETB, DLE/ITB, and DLE/ENQ. These sequences are recognized as escape signals only when they coincide with two specific values in the field length counter of the transmitting station. DLE must coincide with the counter value 11 (decimal) and the associated ETX, ETB, ITB, or ENQ must coincide with the counter value 10 (decimal). As Figure 38 shows, a transparent message may be of any length up to 4,095 bytes; however, due to the fixed location at which the escape sequence must appear, a reply of not greater than nine-bytes length is required. Longer replies, such as used in tête-à-tête operations, are not permitted, but a tête-à-tête-type reply that is limited to nine bytes in length is allowed.

All escape sequences (except DLE/ITB) that are used to leave transparent mode cause the BSCA to return to control mode. In addition, all of them (except DLE/ITB) cause turnaround. DLE/ITB is used to end a transparent block when the intermediate block checking feature is enabled together with the full transparency feature. When transparent intermediate blocks are to be transmitted, there must be a Transmit and Receive instruction for each transparent block.

Each transparent intermediate block is started with DLE/STX and ended with DLE/ITB. After each transparent intermediate block, the transmitting BSCA sends SYN characters for 3 sec to retain synchronization. To avoid timeout, the transmitter must send the next block before 3 sec have elapsed.

The sequence DLE/SOH does not set transparent mode; it is accepted strictly as SOH. The text of the heading must not end with DLE when the adjacent text is to be sent in transparent mode. If the heading ends with DLE, the transparency start signal is changed into DLE/DLE/STX; the BSCA expects STX or ITB directly after DLE (since it does not know where the heading ends) and, therefore, DLE/DLE/STX is not recognized.

The sequence DLE/EOT is not a transparent sequence. DLE/EOT is the disconnect sequence for switched networks.

SENSE INFORMATION

The sense information is a collection of check bits which indicates the type of error that occurred. These bits are available at final interrupt time and must be moved from auxiliary storage (where they are generated) to a main-storage position where they can be examined by the program. The Store Sense Information instruction is designed explicitly for the retrieval of this information. After the execution of this instruction, or after acceptance of the next XIO instruction, the sense information is reset. (See Figure 43 in Appendix L.)

Check Bits

The individual bits are described in the following text.

Bit Zero - Transmit/Receive Mode Error

Bit zero indicates whether the error occurred during transmission or during reception. When this bit is present (binary 1), the error occurred during receive operations. When it is not present (binary 0), the error occurred in transmit operations.

Bit One - EOT Received

This bit indicates that the remote BSCA station was forced to stop transmission or reception due to an I/O condition that required a programmed halt.

Bit Two - CRC/LRC/VRC Check

Bit two is set when either the Cyclic Redundancy Checking (CRC) circuits or the Longitudinal Redundancy Checking (LRC) circuits have detected an error. Both checking circuits can be installed in the BSCA. The checking circuits are activated in both stations with each starting sequence (SOH. STX) and they accumulate a block check character by way of arithmetic operations. After each ending sequence (ETX, ETB, or ITB), the transmitter sends its Block Check Character (BCC) to the receiver where it is compared with the BCC generated by the receiver. Thus, only the receiver detects CRC/LRC errors. These errors are caused by noise on the telephone line. (The efficiency of a noisy telephone line can be greatly increased when messages are divided into blocks that reduce the error probability by way of smaller individual lengths.)

Bit two is set in the receiving BSCA only when USASCII is used as the basic code and the full transparency feature is not installed. The bit indicates a vertical redundancy error (invalid USASCII character) detected during receive operations.

Bit Three - Timeout

Bit three is turned on in the receiving BSCA only when the BSCA is in receive mode and if either:

1. The BSCA does not receive a SYN character, interspersed at the proper time interval
2. The BSCA does not receive a turnaround sequence before 3 sec have elapsed
3. The BSCA receives constant SYN characters for 3 sec.

If the timeout bit is set during an Auto Call instruction, it indicates that the telephone line was busy or that the connection cannot be made (wrong number, etc).

Bit Four - Short Record

Bit four indicates that the field length specification of a Transmit and Receive instruction is smaller than the size of the combined output/input field in main storage that is addressed by the instruction. During processing of an instruction with insufficient field length, the field length counter in the BSCA runs down to zero prior to the detection of the turnaround character (ETX or ETB).

The short record bit is also turned on under the following circumstances:

1. When the field length specification is correct but the turnaround character is either not detected or missing.
2. If the escape sequence in transparent data does not coincide with field length counter values 11 (decimal) and 10 (decimal).
3. If a dial digit is greater than 9.

Bit Five - Storage Wraparound

Bit five is turned on when the combined output/input field (addressed by a Transmit and Receive instruction) exceeds the upper limit of the main storage of the CPU. Depending on the main-storage size, the upper limit is either 4,095, 8,191, 12,287, 16,383, 24,575, or 32,767 positions; beyond these limits, storage wraparound occurs.

Bit Six - Parity Check

Bit six is turned on when a parity error is detected within the data path of the transmitting BSCA. This parity error is indicated only in the transmitting BSCA, and requires the attention of a customer engineer.

Bit Seven - Overrun

Bit seven is turned on when a burst mode operation (tape, or disk) overlaps the reception of an initial sequence (a bid) that was invited by a Receive Initial instruction or an Address Prepare instruction. The overrun bit is also set in the event that a BSCA malfunction prevents the micro-program from servicing BSCA requests; this fault must be corrected by a customer engineer.

When a burst mode operation ends, a BSCA interrupt can follow in which the overrun bit is on, which indicates that an initial sequence was received during the burst mode operation. Therefore, an acknowledgement must be sent back within 3 seconds, otherwise timeout occurs.

BSCA REJECT CONDITIONS

To avoid program error stops during BSCA operation, the following basic requirements must be met:

1. The BSCA must be enabled (to supply the 'data terminal ready' signal for the modem)
2. The modem must have power on and must not be in a test state (so that it can supply the 'data set ready' signal for the BSCA)
3. Instructions must not be issued for optional features which are not installed.

The BSCA instructions are subject to the same rules and restrictions that apply to Model 20 instructions, that is, they must be located on even boundaries in main storage, and addresses must neither violate the protected area nor exceed storage capacity. Field length specifications in excess of 4,095 bytes can be stated; however, only the 12 low-order bits of the field length are used. Burst mode operations (tape, or disk) that interfere with either a Receive Initial instruction or an Address Prepare instruction must not interfere more than once, and new burst mode operations must be inhibited until the initial sequence is received and a reply has been sent.

Figure 39 shows the reject conditions of the BSCA.

Affected Instructions	Error Type	Result
Enable ITB Store ITB Address Auto Call Address Prepare	Feature not installed	Program error stop (6 displayed in Data Register I)
Auto Call	Auto Call Unit power off or Data key on connected telephone set not pressed	Condition code 3 (not operational)
Receive Initial	Modem power off	Condition code 00 (available)
Transmit and Receive, Receive, or Transmit, Address Prepare	Modem not operational (Data Set Ready off)	Condition code 3 (not operational)
All XIO instructions	BSCA not enabled (Data Terminal Ready off)	Condition code 3
	BSCA busy with processing of previous XIO instruction	Condition code 1 (working). See Note
Receive Initial, Transmit and Receive, Receive, or Transmit	Received EOT/DLE-EOT sequence in selected mode but not busy mode	Condition code 3 (not operational) and sense bit 1 are set
Transmit and Receive, or Transmit	Data Set Ready and character phase in receive mode	Enter I/O-Busy loop, wait for available, then execute instruction
Receive, or Receive Initial		Sense bits 4 and 0 are set and the instruction is executed
All CIO and TIOB instructions, except Test on Busy and Store ITB Address	BSCA busy with processing of previous XIO instruction	Enter I/O-Busy loop, wait for available, then execute instruction
All BSCA instructions (2020 Submodel 5 only)	BSCA special voltage off	Condition code 3, and check condition indicator stays on

Note: Condition code 1 is also set when an XIO instruction is issued following a Receive Initial or Address Prepare instruction, even if the BSCA has not yet become busy.

Figure 39. BSCA Reject Conditions 02833

ERROR CHECKING METHODS

Three error checking methods are used in the BSCA: CRC, LRC, and Vertical Redundancy Checking (VRC). The methods are used in the following circumstances:

Circumstances	Checking Methods
When EBCDIC is used as the transmission code.	CRC
When USASCII is used as the transmission code and the full transparency feature is not installed in the BSCA.	LRC and VRC
When USASCII is used as the transmission code and the full transparency feature is installed in the BSCA.	CRC

BSCA CONSOLE

The BSCA console (Figure 40) is located adjacent to the CPU console and contains all indicator lights that are required to show the current state of the BSCA.

Indicator Lights

The following indicator lights are provided on the BSCA console:

BSCA READY: Indicates that the BSCA has been enabled and has activated the data terminal ready line to the modem.

DATA SET READY: Indicates that the modem has power on, is not in test mode, and/or has placed its carrier on the line. The exact modem condition thus indicated depends also on the design of the modem; some modems provide the ready signal only when line connection is established.

ACU POWER: Indicates that the automatic calling unit has power on.

CHECK CONDITION: Indicates that any of the testable BSCA indicators has been set by an abnormal condition.

CHARACTER PHASE: Indicates that the BSCA has achieved synchronization with another station.

AUTO CALL REQUEST: Indicates that the BSCA is sending dialing information to the automatic calling unit.

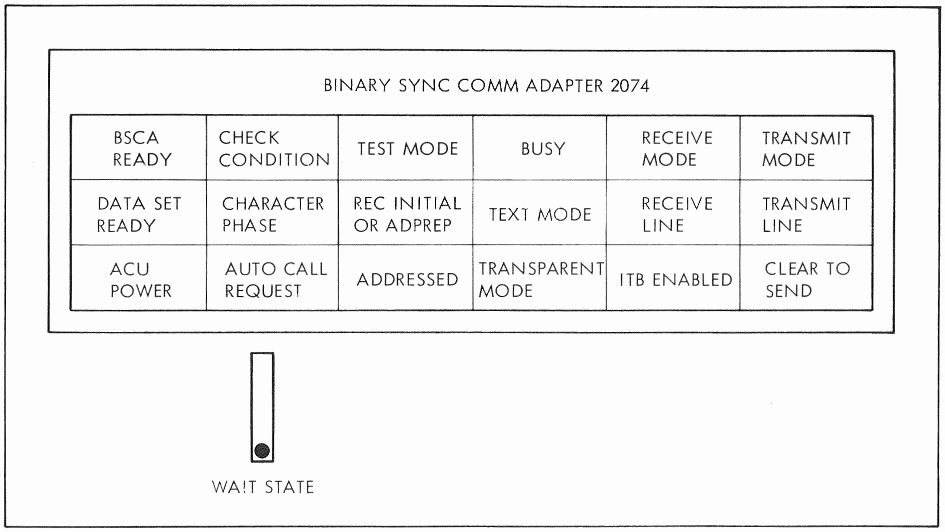
TEST MODE: Indicates that the test switch located in the interface plug is in the test position. In test mode, the BSCA operates in a closed loop with the CPU simulating a remote station.

REC INITIAL OR ADPREP: Indicates that the BSCA is awaiting either an initial sequence or its polling or selection address.

ADDRESSED: This light is active only when the station selection feature is installed. The light indicates that the BSCA has either been polled or selected by a master station.

BUSY: Indicates that the BSCA is processing an XIO instruction.

TEXT MODE: Indicates that a control character (or sequence) has set the BSCA to text mode.



BINARY SYNC COMM ADAPTER 2074

BSCA READY	CHECK CONDITION	TEST MODE	BUSY	RECEIVE MODE	TRANSMIT MODE
DATA SET READY	CHARACTER PHASE	REC INITIAL OR ADPREP	TEXT MODE	RECEIVE LINE	TRANSMIT LINE
ACU POWER	AUTO CALL REQUEST	ADDRESSED	TRANSPARENT MODE	ITB ENABLED	CLEAR TO SEND

Figure 40. BSCA Console 01260

TRANSPARENT MODE: Indicates that the BSCA has been set to transparent mode.

RECEIVE MODE: Indicates that the BSCA is prepared to (or has turned around to) listen.

RECEIVE LINE: Indicates that there is incoming information on the line.

ITB ENABLED: Indicates that the intermediate block checking feature is operational.

TRANSMIT MODE: Indicates that the BSCA is prepared to (or has turned around to) transmit.

TRANSMIT LINE: Indicates that outgoing data is on the line.

CLEAR TO SEND: Indicates that the modem responds to a request to send, via its ready-for-sending interface line. This light must remain on during transmission; if it goes out, a modem or line failure is indicated.

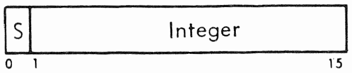
Wait State Switch

The wait state toggle switch on the BSCA console can be flipped to WAIT STATE at any time to prepare the CPU for entering this state. The CPU enters wait state when either a Receive Initial or an Address Prepare instruction has been accepted; this is indicated by REC INITIAL OR ADPREP. The wait state is characterized by the CPU and the use meter being stopped with power on. In wait state, no I/O or processing operation can be started unless the wait state switch is moved to the central (i.e. normal) position. However, the CPU is ready to receive a call, and starts when this comes in.

DATA FORMATS

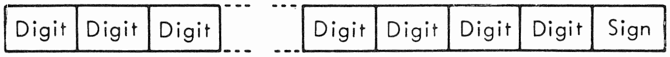
Binary Number

Halfword Binary Number

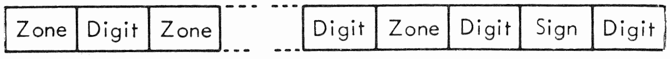


Decimal Numbers

Packed Decimal Number

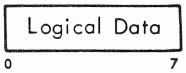


Zoned Decimal Number

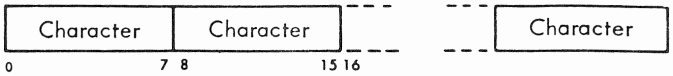


Logical Information

Fixed-Length Logical Information

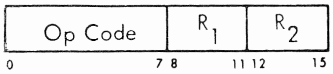


Variable-Length Logical Information

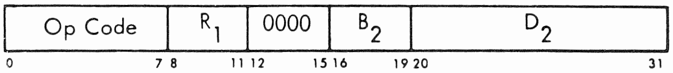


INSTRUCTIONS BY FORMAT TYPE

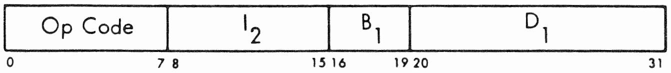
RR Format



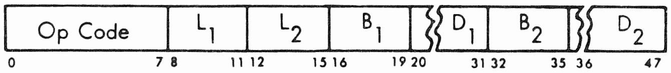
RX Format



SI Format



SS Format



APPENDIX B. EBCDIC AND USASCII CHARTS

Extended Binary-Coded-Decimal Interchange Code (EBCDIC)

Bit Positions of a Byte 0, 1, 2, 3, 4, 5, 6, 7

4567	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
0000	NUL				BLANK	&	–									0	0
0001							/						A	J		1	1
0010													B	K	S	2	2
0011													C	L	T	3	3
0100													D	M	U	4	4
0101													E	N	V	5	5
0110													F	O	W	6	6
0111													G	P	X	7	7
1000													H	Q	Y	8	8
1001													I	R	Z	9	9
1010					¢	!		:									A
1011					.	$	,	#									B
1100					<	*	%	@									C
1101					(	)	–	'									D
1110					+	;	>	=									E
1111					|	¬	?	"									F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

Hexadecimal Representation for Bits 0 1 2 3 (Data Switch 1)

Hexadecimal Representation for Bits 4567 (Data Switch 2)

American Standard Code for Information Interchange (ASCII)
Extended to Eight Bits

Bit Positions 76	00				01				10				11			
X5 / 4321	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0000	NUL	DLE			SP	0					\	P			@	p
0001	SOH	DC1			!	1					A	Q			a	q
0010	STX	DC2			"	2					B	R			b	r
0011	ETX	DC3			#	3					C	S			c	s
0100	EOT	DC4			$	4					D	T			d	t
0101	ENQ	NAK			%	5					E	U			e	u
0110	ACK	SYN			&	6					F	V			f	v
0111	BEL	ETB			'	7					G	W			g	w
1000	BS	CAN			(	8					H	X			h	x
1001	HT	EM			)	9					I	Y			i	y
1010	LF	SS			*	:					J	Z			j	z
1011	VT	ESC			+	;					K	[			k	{
1100	FF	FS			,	<					L	~			l	¬
1101	CR	GS			-	=					M	]			m	}
1110	SO	RS			.	>					N	∧			n	|
1111	SI	US			/	?					O	_			o	DEL

01981

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

01982

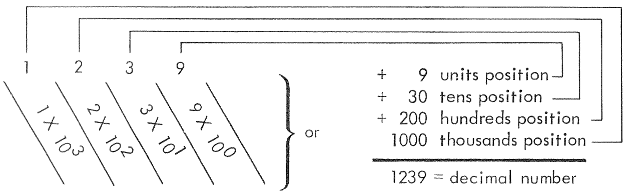
APPENDIX D. BINARY AND HEXADECIMAL NUMBER NOTATION

Binary Number Notation

A binary number system, such as is used in System/360 uses a base of two. The concept of using a base of two can be compared with the base of ten (decimal) number system:

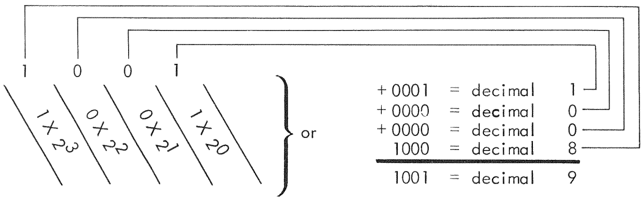
Decimal		Binary
0	=	0
1	=	1
2	=	10
3	=	11
4	=	100
5	=	101
6	=	110
7	=	111
8	=	1000
9	=	1001

Example of a decimal number



As shown above, the decimal number system allows counting to ten in each position from units to tens to hundreds to thousands, etc. The binary system allows counting to two in each position. Register displays in the System/360 are in binary form: a bit light on is a "one"; a bit light off is a "zero".

Example of a binary number



01983

Hexadecimal Number System

It has been noted that binary numbers require about three times as many positions as decimal numbers to express the equivalent number. This is not much of a problem to the computer; however, in talking and writing or in communicating with the computer, these binary numbers are bulky. A long string of 1's and 0's cannot be effectively transmitted from one individual to another. Some shorthand method is necessary.

The hexadecimal number system fills this need. Because of the simple relationship of hexadecimal to binary, numbers can be converted from one system to another by inspection. The base or radix of the hexadecimal system is 16. This means there are 16 symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. The letters A, B, C, D, E, and F represent the 10-base system values of 10, 11, 12, 13, 14, and 15, respectively.

Four binary positions are equivalent to one hexadecimal position. The following table shows the comparable values of the three number systems.

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

At this point, all 16 symbols have been used, and a carry to the next higher position of the number is necessary.

16	0001 0000	10
17	0001 0001	11
18	0001 0010	12
19	0001 0011	13
20	0001 0100	14
21	0001 0101	15

and so on.

Remember that as far as the internal circuitry of the computer is concerned, it only understands binary. But an operator can look at a series of lights on the computer console showing binary 1's and 0's, for example: 0001 1110 0001 0011, and say that the lights represent the hexadecimal value 1E13 which is easier to state than the string of 1's and 0's.

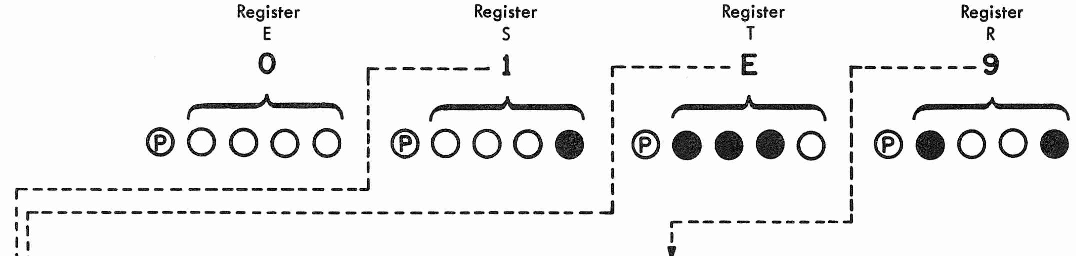
The table in this appendix provides for direct conversion of decimal and hexadecimal numbers in these ranges:

Hexadecimal	Decimal
000 to FFF	000 to 4095

For numbers outside the range of the table, add the following values to the table figures:

Hexadecimal	Decimal
1000	4096
2000	8192
3000	12288
4000	16384

Hexadecimal	Decimal
5000	20480
6000	24576
7000	28672
8000	32768
9000	36864
A000	40960
B000	45056
C000	49152
D000	53248
E000	57344
F000	61440



	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

01295

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20_	0512	0513	0514	0515	0516	0517	0518	0519	0520	0521	0522	0523	0524	0525	0526	0527
21_	0528	0529	0530	0531	0532	0533	0534	0535	0536	0537	0538	0539	0540	0541	0542	0543
22_	0544	0545	0546	0547	0548	0549	0550	0551	0552	0553	0554	0555	0556	0557	0558	0559
23_	0560	0561	0562	0563	0564	0565	0566	0567	0568	0569	0570	0571	0572	0573	0574	0575
24_	0576	0577	0578	0579	0580	0581	0582	0583	0584	0585	0586	0587	0588	0589	0590	0591
25_	0592	0593	0594	0595	0596	0597	0598	0599	0600	0601	0602	0603	0604	0605	0606	0607
26_	0608	0609	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	0620	0621	0622	0623
27_	0624	0625	0626	0627	0628	0629	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639
28_	0640	0641	0642	0643	0644	0645	0646	0647	0648	0649	0650	0651	0652	0653	0654	0655
29_	0656	0657	0658	0659	0660	0661	0662	0663	0664	0665	0666	0667	0668	0669	0670	0671
2A_	0672	0673	0674	0675	0676	0677	0678	0679	0680	0681	0682	0683	0684	0685	0686	0687
2B_	0688	0689	0690	0691	0692	0693	0694	0695	0696	0697	0698	0699	0700	0701	0702	0703
2C_	0704	0705	0706	0707	0708	0709	0710	0711	0712	0713	0714	0715	0716	0717	0718	0719
2D_	0720	0721	0722	0723	0724	0725	0726	0727	0728	0729	0730	0731	0732	0733	0734	0735
2E_	0736	0737	0738	0739	0740	0741	0742	0743	0744	0745	0746	0747	0748	0749	0750	0751
2F_	0752	0753	0754	0755	0756	0757	0758	0759	0760	0761	0762	0763	0764	0765	0766	0767
30_	0768	0769	0770	0771	0772	0773	0774	0775	0776	0777	0778	0779	0780	0781	0782	0783
31_	0784	0785	0786	0787	0788	0789	0790	0791	0792	0793	0794	0795	0796	0797	0798	0799
32_	0800	0801	0802	0803	0804	0805	0806	0807	0808	0809	0810	0811	0812	0813	0814	0815
33_	0816	0817	0818	0819	0820	0821	0822	0823	0824	0825	0826	0827	0828	0829	0830	0831
34_	0832	0833	0834	0835	0836	0837	0838	0839	0840	0841	0842	0843	0844	0845	0846	0847
35_	0848	0849	0850	0851	0852	0853	0854	0855	0856	0857	0858	0859	0860	0861	0862	0863
36_	0864	0865	0866	0867	0868	0869	0870	0871	0872	0873	0874	0875	0876	0877	0878	0879
37_	0880	0881	0882	0883	0884	0885	0886	0887	0888	0889	0890	0891	0892	0893	0894	0895
38_	0896	0897	0898	0899	0900	0901	0902	0903	0904	0905	0906	0907	0908	0909	0910	0911
39_	0912	0913	0914	0915	0916	0917	0918	0919	0920	0921	0922	0923	0924	0925	0926	0927
3A_	0928	0929	0930	0931	0932	0933	0934	0935	0936	0937	0938	0939	0940	0941	0942	0943
3B_	0944	0945	0946	0947	0948	0949	0950	0951	0952	0953	0954	0955	0956	0957	0958	0959
3C_	0960	0961	0962	0963	0964	0965	0966	0967	0968	0969	0970	0971	0972	0973	0974	0975
3D_	0976	0977	0978	0979	0980	0981	0982	0983	0984	0985	0986	0987	0988	0989	0990	0991
3E_	0992	0993	0994	0995	0996	0997	0998	0999	1000	1001	1002	1003	1004	1005	1006	1007
3F_	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
40_	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039
41_	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1050	1051	1052	1053	1054	1055
42_	1056	1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1070	1071
43_	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081	1082	1083	1084	1085	1086	1087
44_	1088	1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103
45_	1104	1105	1106	1107	1108	1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119
46_	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129	1130	1131	1132	1133	1134	1135
47_	1136	1137	1138	1139	1140	1141	1142	1143	1144	1145	1146	1147	1148	1149	1150	1151
48_	1152	1153	1154	1155	1156	1157	1158	1159	1160	1161	1162	1163	1164	1165	1166	1167
49_	1168	1169	1170	1171	1172	1173	1174	1175	1176	1177	1178	1179	1180	1181	1182	1183
4A_	1184	1185	1186	1187	1188	1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199
4B_	1200	1201	1202	1203	1204	1205	1206	1207	1208	1209	1210	1211	1212	1213	1214	1215
4C_	1216	1217	1218	1219	1220	1221	1222	1223	1224	1225	1226	1227	1228	1229	1230	1231
4D_	1232	1233	1234	1235	1236	1237	1238	1239	1240	1241	1242	1243	1244	1245	1246	1247
4E_	1248	1249	1250	1251	1252	1253	1254	1255	1256	1257	1258	1259	1260	1261	1262	1263
4F_	1264	1265	1266	1267	1268	1269	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279
50_	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289	1290	1291	1292	1293	1294	1295
51_	1296	1297	1298	1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	1310	1311
52_	1312	1313	1314	1315	1316	1317	1318	1319	1320	1321	1322	1323	1324	1325	1326	1327
53_	1328	1329	1330	1331	1332	1333	1334	1335	1336	1337	1338	1339	1340	1341	1342	1343
54_	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353	1354	1355	1356	1357	1358	1359
55_	1360	1361	1362	1363	1364	1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375
56_	1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	1386	1387	1388	1389	1390	1391
57_	1392	1393	1394	1395	1396	1397	1398	1399	1400	1401	1402	1403	1404	1405	1406	1407
58_	1408	1409	1410	1411	1412	1413	1414	1415	1416	1417	1418	1419	1420	1421	1422	1423
59_	1424	1425	1426	1427	1428	1429	1430	1431	1432	1433	1434	1435	1436	1437	1438	1439
5A_	1440	1441	1442	1443	1444	1445	1446	1447	1448	1449	1450	1451	1452	1453	1454	1455
5B_	1456	1457	1458	1459	1460	1461	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471
5C_	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	1482	1483	1484	1485	1486	1487
5D_	1488	1489	1490	1491	1492	1493	1494	1495	1496	1497	1498	1499	1500	1501	1502	1503
5E_	1504	1505	1506	1507	1508	1509	1510	1511	1512	1513	1514	1515	1516	1517	1518	1519
5F_	1520	1521	1522	1523	1524	1525	1526	1527	1528	1529	1530	1531	1532	1533	1534	1535

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
60_	1536	1537	1538	1539	1540	1541	1542	1543	1544	1545	1546	1547	1548	1549	1550	1551
61_	1552	1553	1554	1555	1556	1557	1558	1559	1560	1561	1562	1563	1564	1565	1566	1567
62_	1568	1569	1570	1571	1572	1573	1574	1575	1576	1577	1578	1579	1580	1581	1582	1583
63_	1584	1585	1586	1587	1588	1589	1590	1591	1592	1593	1594	1595	1596	1597	1598	1599
64_	1600	1601	1602	1603	1604	1605	1606	1607	1608	1609	1610	1611	1612	1613	1614	1615
65_	1616	1617	1618	1619	1620	1621	1622	1623	1624	1625	1626	1627	1628	1629	1630	1631
66_	1632	1633	1634	1635	1636	1637	1638	1639	1640	1641	1642	1643	1644	1645	1646	1647
67_	1648	1649	1650	1651	1652	1653	1654	1655	1656	1657	1658	1659	1660	1661	1662	1663
68_	1664	1665	1666	1667	1668	1669	1670	1671	1672	1673	1674	1675	1676	1677	1678	1679
69_	1680	1681	1682	1683	1684	1685	1686	1687	1688	1689	1690	1691	1692	1693	1694	1695
6A_	1696	1697	1698	1699	1700	1701	1702	1703	1704	1705	1706	1707	1708	1709	1710	1711
6B_	1712	1713	1714	1715	1716	1717	1718	1719	1720	1721	1722	1723	1724	1725	1726	1727
6C_	1728	1729	1730	1731	1732	1733	1734	1735	1736	1737	1738	1739	1740	1741	1742	1743
6D_	1744	1745	1746	1747	1748	1749	1750	1751	1752	1753	1754	1755	1756	1757	1758	1759
6E_	1760	1761	1762	1763	1764	1765	1766	1767	1768	1769	1770	1771	1772	1773	1774	1775
6F_	1776	1777	1778	1779	1780	1781	1782	1783	1784	1785	1786	1787	1788	1789	1790	1791
70_	1792	1793	1794	1795	1796	1797	1798	1799	1800	1801	1802	1803	1804	1805	1806	1807
71_	1808	1809	1810	1811	1812	1813	1814	1815	1816	1817	1818	1819	1820	1821	1822	1823
72_	1824	1825	1826	1827	1828	1829	1830	1831	1832	1833	1834	1835	1836	1837	1838	1839
73_	1840	1841	1842	1843	1844	1845	1846	1847	1848	1849	1850	1851	1852	1853	1854	1855
74_	1856	1857	1858	1859	1860	1861	1862	1863	1864	1865	1866	1867	1868	1869	1870	1871
75_	1872	1873	1874	1875	1876	1877	1878	1879	1880	1881	1882	1883	1884	1885	1886	1887
76_	1888	1889	1890	1891	1892	1893	1894	1895	1896	1897	1898	1899	1900	1901	1902	1903
77_	1904	1905	1906	1907	1908	1909	1910	1911	1912	1913	1914	1915	1916	1917	1918	1919
78_	1920	1921	1922	1923	1924	1925	1926	1927	1928	1929	1930	1931	1932	1933	1934	1935
79_	1936	1937	1938	1939	1940	1941	1942	1943	1944	1945	1946	1947	1948	1949	1950	1951
7A_	1952	1953	1954	1955	1956	1957	1958	1959	1960	1961	1962	1963	1964	1965	1966	1967
7B_	1968	1969	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979	1980	1981	1982	1983
7C_	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
7D_	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015
7E_	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031
7F_	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
80_	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063
81_	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079
82_	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095
83_	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111
84_	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127
85_	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143
86_	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159
87_	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175
88_	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191
89_	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207
8A_	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223
8B_	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239
8C_	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255
8D_	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271
8E_	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287
8F_	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303
90_	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319
91_	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335
92_	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351
93_	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367
94_	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383
95_	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399
96_	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415
97_	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431
98_	2432	2433	2434	2435	2436	2437	2438	2439	2440	2441	2442	2443	2444	2445	2446	2447
99_	2448	2449	2450	2451	2452	2453	2454	2455	2456	2457	2458	2459	2460	2461	2462	2463
9A_	2464	2465	2466	2467	2468	2469	2470	2471	2472	2473	2474	2475	2476	2477	2478	2479
9B_	2480	2481	2482	2483	2484	2485	2486	2487	2488	2489	2490	2491	2492	2493	2494	2495
9C_	2496	2497	2498	2499	2500	2501	2502	2503	2504	2505	2506	2507	2508	2509	2510	2511
9D_	2512	2513	2514	2515	2516	2517	2518	2519	2520	2521	2522	2523	2524	2525	2526	2527
9E_	2528	2529	2530	2531	2532	2533	2534	2535	2536	2537	2538	2539	2540	2541	2542	2543
9F_	2544	2545	2546	2547	2548	2549	2550	2551	2552	2553	2554	2555	2556	2557	2558	2559

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A0 –	2560	2561	2562	2563	2564	2565	2566	2567	2568	2569	2570	2571	2572	2573	2574	2575
A1 –	2576	2577	2578	2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591
A2 –	2592	2593	2594	2595	2596	2597	2598	2599	2600	2601	2602	2603	2604	2605	2606	2607
A3 –	2608	2609	2610	2611	2612	2613	2614	2615	2616	2617	2618	2619	2620	2621	2622	2623
A4 –	2624	2625	2626	2627	2628	2629	2630	2631	2632	2633	2634	2635	2636	2637	2638	2639
A5 –	2640	2641	2642	2643	2644	2645	2646	2647	2648	2649	2650	2651	2652	2653	2654	2655
A6 –	2656	2657	2658	2659	2660	2661	2662	2663	2664	2665	2666	2667	2668	2669	2670	2671
A7 –	2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684	2685	2686	2687
A8 –	2688	2689	2690	2691	2692	2693	2694	2695	2696	2697	2698	2699	2700	2701	2702	2703
A9 –	2704	2705	2706	2707	2708	2709	2710	2711	2712	2713	2714	2715	2716	2717	2718	2719
AA –	2720	2721	2722	2723	2724	2725	2726	2727	2728	2729	2730	2731	2732	2733	2734	2735
AB –	2736	2737	2738	2739	2740	2741	2742	2743	2744	2745	2746	2747	2748	2749	2750	2751
AC –	2752	2753	2754	2755	2756	2757	2758	2759	2760	2761	2762	2763	2764	2765	2766	2767
AD –	2768	2769	2770	2771	2772	2773	2774	2775	2776	2777	2778	2779	2780	2781	2782	2783
AE –	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	2795	2796	2797	2798	2799
AF –	2800	2801	2802	2803	2804	2805	2806	2807	2808	2809	2810	2811	2812	2813	2814	2815
B0 –	2816	2817	2818	2819	2820	2821	2822	2823	2824	2825	2826	2827	2828	2829	2830	2831
B1 –	2832	2833	2834	2835	2836	2837	2838	2839	2840	2841	2842	2843	2844	2845	2846	2847
B2 –	2848	2849	2850	2851	2852	2853	2854	2855	2856	2857	2858	2859	2860	2861	2862	2863
B3 –	2864	2865	2866	2867	2868	2869	2870	2871	2872	2873	2874	2875	2876	2877	2878	2879
B4 –	2880	2881	2882	2883	2884	2885	2886	2887	2888	2889	2890	2891	2892	2893	2894	2895
B5 –	2896	2897	2898	2899	2900	2901	2902	2903	2904	2905	2906	2907	2908	2909	2910	2911
B6 –	2912	2913	2914	2915	2916	2917	2918	2919	2920	2921	2922	2923	2924	2925	2926	2927
B7 –	2928	2929	2930	2931	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941	2942	2943
B8 –	2944	2945	2946	2947	2948	2949	2950	2951	2952	2953	2954	2955	2956	2957	2958	2959
B9 –	2960	2961	2962	2963	2964	2965	2966	2967	2968	2969	2970	2971	2972	2973	2974	2975
BA –	2976	2977	2978	2979	2980	2981	2982	2983	2984	2985	2986	2987	2988	2989	2990	2991
BB –	2992	2993	2994	2995	2996	2997	2998	2999	3000	3001	3002	3003	3004	3005	3006	3007
BC –	3008	3009	3010	3011	3012	3013	3014	3015	3016	3017	3018	3019	3020	3021	3022	3023
BD –	3024	3025	3026	3027	3028	3029	3030	3031	3032	3033	3034	3035	3036	3037	3038	3039
BE –	3040	3041	3042	3043	3044	3045	3046	3047	3048	3049	3050	3051	3052	3053	3054	3055
BF –	3056	3057	3058	3059	3060	3061	3062	3063	3064	3065	3066	3067	3068	3069	3070	3071

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
C0 –	3072	3073	3074	3075	3076	3077	3078	3079	3080	3081	3082	3083	3084	3085	3086	3087
C1 –	3088	3089	3090	3091	3092	3093	3094	3095	3096	3097	3098	3099	3100	3101	3102	3103
C2 –	3104	3105	3106	3107	3108	3109	3110	3111	3112	3113	3114	3115	3116	3117	3118	3119
C3 –	3120	3121	3122	3123	3124	3125	3126	3127	3128	3129	3130	3131	3132	3133	3134	3135
C4 –	3136	3137	3138	3139	3140	3141	3142	3143	3144	3145	3146	3147	3148	3149	3150	3151
C5 –	3152	3153	3154	3155	3156	3157	3158	3159	3160	3161	3162	3163	3164	3165	3166	3167
C6 –	3168	3169	3170	3171	3172	3173	3174	3175	3176	3177	3178	3179	3180	3181	3182	3183
C7 –	3184	3185	3186	3187	3188	3189	3190	3191	3192	3193	3194	3195	3196	3197	3198	3199
C8 –	3200	3201	3202	3203	3204	3205	3206	3207	3208	3209	3210	3211	3212	3213	3214	3215
C9 –	3216	3217	3218	3219	3220	3221	3222	3223	3224	3225	3226	3227	3228	3229	3230	3231
CA –	3232	3233	3234	3235	3236	3237	3238	3239	3240	3241	3242	3243	3244	3245	3246	3247
CB –	3248	3249	3250	3251	3252	3253	3254	3255	3256	3257	3258	3259	3260	3261	3262	3263
CC –	3264	3265	3266	3267	3268	3269	3270	3271	3272	3273	3274	3275	3276	3277	3278	3279
CD –	3280	3281	3282	3283	3284	3285	3286	3287	3288	3289	3290	3291	3292	3293	3294	3295
CE –	3296	3297	3298	3299	3300	3301	3302	3303	3304	3305	3306	3307	3308	3309	3310	3311
CF –	3312	3313	3314	3315	3316	3317	3318	3319	3320	3321	3322	3323	3324	3325	3326	3327
D0 –	3328	3329	3330	3331	3332	3333	3334	3335	3336	3337	3338	3339	3340	3341	3342	3343
D1 –	3344	3345	3346	3347	3348	3349	3350	3351	3352	3353	3354	3355	3356	3357	3358	3359
D2 –	3360	3361	3362	3363	3364	3365	3366	3367	3368	3369	3370	3371	3372	3373	3374	3375
D3 –	3376	3377	3378	3379	3380	3381	3382	3383	3384	3385	3386	3387	3388	3389	3390	3391
D4 –	3392	3393	3394	3395	3396	3397	3398	3399	3400	3401	3402	3403	3404	3405	3406	3407
D5 –	3408	3409	3410	3411	3412	3413	3414	3415	3416	3417	3418	3419	3420	3421	3422	3423
D6 –	3424	3425	3426	3427	3428	3429	3430	3431	3432	3433	3434	3435	3436	3437	3438	3439
D7 –	3440	3441	3442	3443	3444	3445	3446	3447	3448	3449	3450	3451	3452	3453	3454	3455
D8 –	3456	3457	3458	3459	3460	3461	3462	3463	3464	3465	3466	3467	3468	3469	3470	3471
D9 –	3472	3473	3474	3475	3476	3477	3478	3479	3480	3481	3482	3483	3484	3485	3486	3487
DA –	3488	3489	3490	3491	3492	3493	3494	3495	3496	3497	3498	3499	3500	3501	3502	3503
DB –	3504	3505	3506	3507	3508	3509	3510	3511	3512	3513	3514	3515	3516	3517	3518	3519
DC –	3520	3521	3522	3523	3524	3525	3526	3527	3528	3529	3530	3531	3532	3533	3534	3535
DD –	3536	3537	3538	3539	3540	3541	3542	3543	3544	3545	3546	3547	3548	3549	3550	3551
DE –	3552	3553	3554	3555	3556	3557	3558	3559	3560	3561	3562	3563	3564	3565	3566	3567
DF –	3568	3569	3570	3571	3572	3573	3574	3575	3576	3577	3578	3579	3580	3581	3582	3583

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
E0 _	3584	3585	3586	3587	3588	3589	3590	3591	3592	3593	3594	3595	3596	3597	3598	3599
E1 _	3600	3601	3602	3603	3604	3605	3606	3607	3608	3609	3610	3611	3612	3613	3614	3615
E2 _	3616	3617	3618	3619	3620	3621	3622	3623	3624	3625	3626	3627	3628	3629	3630	3631
E3 _	3632	3633	3634	3635	3636	3637	3638	3639	3640	3641	3642	3643	3644	3645	3646	3647
E4 _	3648	3649	3650	3651	3652	3653	3654	3655	3656	3657	3658	3659	3660	3661	3662	3663
E5 _	3664	3665	3666	3667	3668	3669	3670	3671	3672	3673	3674	3675	3676	3677	3678	3679
E6 _	3680	3681	3682	3683	3684	3685	3686	3687	3688	3689	3690	3691	3692	3693	3694	3695
E7 _	3696	3697	3698	3699	3700	3701	3702	3703	3704	3705	3706	3707	3708	3709	3710	3711
E8 _	3712	3713	3714	3715	3716	3717	3718	3719	3720	3721	3722	3723	3724	3725	3726	3727
E9 _	3728	3729	3730	3731	3732	3733	3734	3735	3736	3737	3738	3739	3740	3741	3742	3743
EA _	3744	3745	3746	3747	3748	3749	3750	3751	3752	3753	3754	3755	3756	3757	3758	3759
EB _	3760	3761	3762	3763	3764	3765	3766	3767	3768	3769	3770	3771	3772	3773	3774	3775
EC _	3776	3777	3778	3779	3780	3781	3782	3783	3784	3785	3786	3787	3788	3789	3790	3791
ED _	3792	3793	3794	3795	3796	3797	3798	3799	3800	3801	3802	3803	3804	3805	3806	3807
EE _	3808	3809	3810	3811	3812	3813	3814	3815	3816	3817	3818	3819	3820	3821	3822	3823
EF _	3824	3825	3826	3827	3828	3829	3830	3831	3832	3833	3834	3835	3836	3837	3838	3839
F0 _	3840	3841	3842	3843	3844	3845	3846	3847	3848	3849	3850	3851	3852	3853	3854	3855
F1 _	3856	3857	3858	3859	3860	3861	3862	3863	3864	3865	3866	3867	3868	3869	3870	3871
F2 _	3872	3873	3874	3875	3876	3877	3878	3879	3880	3881	3882	3883	3884	3885	3886	3887
F3 _	3888	3889	3890	3891	3892	3893	3894	3895	3896	3897	3898	3899	3900	3901	3902	3903
F4 _	3904	3905	3906	3907	3908	3909	3910	3911	3912	3913	3914	3915	3916	3917	3918	3919
F5 _	3920	3921	3922	3923	3924	3925	3926	3927	3928	3929	3930	3931	3932	3933	3934	3935
F6 _	3936	3937	3938	3939	3940	3941	3942	3943	3944	3945	3946	3947	3948	3949	3950	3951
F7 _	3952	3953	3954	3955	3956	3957	3958	3959	3960	3961	3962	3963	3964	3965	3966	3967
F8 _	3968	3969	3970	3971	3972	3973	3974	3975	3976	3977	3978	3979	3980	3981	3982	3983
F9 _	3984	3985	3986	3987	3988	3989	3990	3991	3992	3993	3994	3995	3996	3997	3998	3999
FA _	4000	4001	4002	4003	4004	4005	4006	4007	4008	4009	4010	4011	4012	4013	4014	4015
FB _	4016	4017	4018	4019	4020	4021	4022	4023	4024	4025	4026	4027	4028	4029	4030	4031
FC _	4032	4033	4034	4035	4036	4037	4038	4039	4040	4041	4042	4043	4044	4045	4046	4047
FD _	4048	4049	4050	4051	4052	4053	4054	4055	4056	4057	4058	4059	4060	4061	4062	4063
FE _	4064	4065	4066	4067	4068	4069	4070	4071	4072	4073	4074	4075	4076	4077	4078	4079
FF _	4080	4081	4082	4083	4084	4085	4086	4087	4088	4089	4090	4091	4092	4093	4094	4095

APPENDIX F. OPERATION CODES

Operation Codes

Name	Mnemonic	Op Code		Class	Format
		Hexadecimal	Binary		
Branch on Condition	BCR	07	0000 0111	Branching	RR
Branch and Store	BASR	0D	0000 1101	Branching	RR
Add	AR	1A	0001 1010	Binary	RR
Subtract	SR	1B	0001 1011	Binary	RR
Store Halfword	STH	40	0100 0000	Binary	RX
Branch on Condition	BC	47	0100 0111	Branching	RX
Load Halfword	LH	48	0100 1000	Binary	RX
Compare Halfword	CH	49	0100 1001	Binary	RX
Add Halfword	AH	4A	0100 1010	Binary	RX
Subtract Halfword	SH	4B	0100 1011	Binary	RX
Branch and Store	BAS	4D	0100 1101	Branching	RX
*Diagnostic		83	1000 0011	Special	SI
Test under Mask	TM	91	1001 0001	Logical	SI
Move	MVI	92	1001 0010	Logical	SI
Set PSW	SPSW	81	1000 0001	Branching	SI
And	NI	94	1001 0100	Logical	SI
Compare	CLI	95	1001 0101	Logical	SI
Or	OI	96	1001 0110	Logical	SI
Halt & Proceed	HPR	99	1001 1001	Logical	SI
Test I/O and Branch	TIOB	9A	1001 1010	Input/Output	IO
Control I/O	CIO	9B	1001 1011	Input/Output	IO
Transfer I/O	XIO	D0	1101 0000	Input/Output	IO
Move Numerical	MVN	D1	1101 0001	Logical	SS
Move Characters	MVC	D2	1101 0010	Logical	SS
Move Zone	MVZ	D3	1101 0011	Logical	SS
Compare	CLC	D5	1101 0101	Logical	SS
Edit	ED	DE	1101 1110	Logical	SS
Move with Offset	MVO	F1	1111 0001	Decimal	SS
Pack	PACK	F2	1111 0010	Decimal	SS
Unpack	UNPK	F3	1111 0011	Decimal	SS
Zero and Add	ZAP	F8	1111 1000	Decimal	SS
Compare Decimal	CP	F9	1111 1001	Decimal	SS
Add Decimal	AP	FA	1111 1010	Decimal	SS
Subtract Decimal	SP	FB	1111 1011	Decimal	SS
Multiply Decimal	MP	FC	1111 1100	Decimal	SS
Divide Decimal	DP	FD	1111 1101	Decimal	SS
Translate	TR	DC	1101 1100	Logical	SS

* The Diagnostic instruction is used in special programs by the IBM Customer Engineer to verify the proper functioning of the CPU and to assist in locating faulty components. It is not intended for use in programs for normal applications and its use should be avoided, because the status of the CPU, the condition code, the contents of registers and main storage, and the progress of I/O operations may be affected.

01984

APPENDIX G. CONDITION CODE

Instruction	Condition Code Setting			
	00	01	10	11
Add	zero	< zero	> zero	--
Subtract	zero	< zero	> zero	--
Compare Halfword	equal	low	high	--
Add Halfword	zero	< zero	> zero	--
Subtract Halfword	zero	< zero	> zero	--
Zero & Add	zero	< zero	> zero	--
Compare Decimal	equal	low	high	--
Add Decimal	zero	< zero	> zero	overflow
Subtract Decimal	zero	< zero	> zero	overflow
Test under Mask	zero	mixed	--	all one
And	zero	not zero	--	--
Compare Logical	equal	low	high	--
Or	zero	not zero	--	--
Edit	zero	< zero	> zero	--
Transfer I/O*	available	working	channel busy	not operational
Control I/O**	available	working	--	not operational

* The Transfer I/O and the corresponding condition codes do not pertain to the input/output channel or storage control feature

** 1403, 2203, and 2152 Control Carriage instructions only

01985

APPENDIX H. SYSTEM/360 MODEL 20 TIMING

CPU SUBMODELS 1 AND 2

Processing Operation	Format	Mnemonic	Timing (average in microseconds with time sharing switch ON)
Branch on Condition	RR	BCR	78
Branch and Store	RR	BASR	94
Add	RR	AR	174
Subtract	RR	SR	183
Store Halfword	RX	STH	117
Branch on Condition	RX	BC	110
Load Halfword	RX	LH	124
Compare Halfword	RX	CH	223
Add Halfword	RX	AH	209
Subtract Halfword	RX	SH	216
Branch and Store	RX	BAS	124
Test Under Mask	SI	TM	132
Move	SI	MVI	104
Set PSW	SI	SPSW	149
AND	SI	NI	123
Compare Logical	SI	CLI	122
OR	SI	OI	130
Halt and Proceed	SI	HPR	105
Move Numerical	SS	MVN	$137 + 16N$
Move Characters	SS	MVC	$137 + 16N$
Move Zone	SS	MVZ	$137 + 16N$
Compare Logical	SS	CLC	$126 + 24B$
Edit	SS	ED	$165 + 26N_1$
Move with Offset	SS	MVO	$170 + 10.2N_1 + 7.8N_2$ (if $N_1 > N_2$) $170 + 18.0N_1$ (if $N_1 \leq N_2$)
Translate	SS	TR	$148 + 53N$
Pack	SS	PACK	* $182 + 27N_2(\frac{N_2 + P_3}{2} - 1) + 16.8P_1 + 9.6N_1 - 4.8N_2$ * $\left.\begin{matrix}\text{Use 182, if } 2N_1 - (N_2 + 1) \text{ is } > 0 \\ \text{Use 161, if } 2N_1 - (N_2 + 1) \text{ is } < 0\end{matrix}\right\}$
Unpack	SS	UNPK	$178 + 9.6N_1 + 8.4N_2 + P_4(5 + 4.4N_1 - 8.4N_2)$
Compare Decimal	SS	CP	$312 + 13.2N_2 + (N_2 - 1)(69 + 15P_8) + (N_1 - N_2)$ $(38 + 9P_8) + 12P_6$
Add Decimal	SS	AP	$312 + 13.2N_2 + (N_2 - 1)(69 + 15P_5) + (N_1 - N_2)$ $(38 + 9P_5) + 12P_6 + P_7(15 + 57N_1)$
Subtract Decimal	SS	SP	(The formula is the same for both instructions)
Multiply Decimal	SS	MP	$215 + 20N_2 + (N_1 - N_2)(70 + 9.6N_2) + \Sigma Op1_{dig}$ $(39 + 9.6N_1 + 44N_2)$
Divide Decimal	SS	DP	$325 + 250N_2 + (N_1 - N_2)(150 + 250N_2) + \Sigma Q$ $(100 + 50N_2)$
Zero and Add	SS	ZAP	$210 + 10.2N_1 + 13.2N_2$

Notes: With time sharing switch OFF, add 22 usec to each instruction.
When effective address generation (Indexing) is used, add 56 usec for each generated address.

Symbols used in the timing formulas:

N	= Number of bytes in field (defined by coupled field length)
N_1	= Number of bytes in the first operand
N_2	= Number of bytes in the second operand
B	= Total number of processed bytes in the first operand field
ΣQ	= Sum of quotient digits
$\Sigma Op1_{dig}$	= Sum of operand 1 digits

The following constants must be placed into the respective formulas in the positions indicated by P_1 through P_8:

$P_1 = 1$ if $2N_1 - (N_2 + 1)$ is odd 0 if $2N_1 - (N_2 + 1)$ is even	$P_4 = 1$ if $N_1 + 1 - 2N_2$ is < 0 0 if $N_1 + 1 - 2N_2$ is ≥ 0	$P_6 = 1$ if $L_1 > L_2$ 0 if $L_1 = L_2$
$P_2 = 1$ if $N_1 = N_2 = 1$ 0 if other	P_5 = For Add: 1 if the signs of oprnd 1 and oprnd 2 differ 0 if the signs of both oprnds are alike For Subtract: 1 if the signs of both oprnds are alike 0 if the signs of oprnd 1 and 2 differ	P_7 = 1 if Recomplement occurs 0 if No Recomplement occurs
$P_3 = 1$ if N_2 is odd 0 if N_2 is even		P_8 = 1 if the signs of oprnd 1 and oprnd 2 are alike 0 if the signs of the oprnds differ

01205

CPU SUBMODELS 1 AND 2 (CONTINUED)

I/O Device	Input/Output Operation	Mnemonic	Timing (Average in Microseconds
2501	Read Card	XIO	186
	Test Reader Busy	TIOB	112
	Test Reader Error	TIOB	112
2520	Read Card	XIO	190
	Punch Card	XIO	172+24F+7(80-F)
	Punch and Feed	XIO	172+24F+7(80-F)
	Test Reader Busy	TIOB	112
	Test Reader Error	TIOB	112
	Test Punch Busy	TIOB	112
	Test Punch Error	TIOB	112
	Test Last Card	TIOB	112
	Test Feed Error	TIOB	112
	Stacker Select	CIO	112
2560 Model A1	Read Card	XIO	220
	Punch Card	XIO	216
	Punch and Feed	XIO	216
	Write Card	XIO	224
	Test Rdr/Pch Busy	TIOB	112
	Test Rdr/Pch Error	TIOB	112
	Test Printer Busy	TIOB	112
	Test Last Card	TIOB	112
	Test Feed Error	TIOB	136
	Stacker Select	CIO	118
	Print Head Select	CIO	116
1442	Punch Card	XIO	183
	Test Punch Busy	TIOB	112
	Test Punch Error	TIOB	112
	Test Feed Error	TIOB	112
2203 Model A1	Print	XIO	2100+45.2F
	Test Printer Busy	TIOB	115
	Test Printer Error	TIOB	115
	Test Chan 9 or 12	TIOB	115
	Test Carriage Busy	TIOB	115
	Control Carriage	CIO	125
1403	Print (48 Char)	XIO	2400+6.6F
	Print (Numeric)	XIO	2400+10.5F
	Test Printer Busy	TIOB	115
	Test Printer Error	TIOB	115
	Test Chan 9 or 12	TIOB	115
	Control Carriage	CIO	120
1419	Read I/O Device	XIO	187
	Test I/O Trans Line	TIOB	112
	Unit Select	CIO	112
	I/O Select	CIO	110
1259	Read I/O Device	XIO	187
	Test I/O Trans Line	TIOB	112
	Unit Select	CIO	112
	I/O Select	CIO	110

I/O Device	Input/Output Operation	Mnemonic	Timing (Average) in Microseconds
Comm Adapter	Transmit	XIO	190
	Receive	XIO	190
	Test for Busy	TIOB	120
	Test for Error	TIOB	120
	Test EOT	TIOB	115
	Test on BCD	TIOB	115
	Receive Mode	CIO	150
	Transmit Mode	CIO	135
	Send EOT	CIO	135
	Inhibit Aud Alarm	CIO	120
BSCA	Transmit	XIO	194
	Transmit and Receive	XIO	194
	Receive	XIO	200
	Receive Initial	XIO	198
	Adprep	XIO	192
	Auto Call	XIO	192
	Enable/Disable BSCA	CIO	106
	Enable/Disable ITB	CIO	106
	Store Current Address	CIO	149
	Store Sense	CIO	115
	Store ITB Address	CIO	149
	Test any Check Condition	TIOB	106
	Test Busy	TIOB	103

01789

Average CPU Time Required for Time-Shared Input/Output Operations (Time in Milliseconds)

Machine	Operation	Timing Formula
2501	Card Read	0.065·F
2520	Card Read Card Punch	0.065·F 8.0
2560 Model A1	Card Read Card Punch Card Print	0.065·F 0.065·F H·0.065·F
1442	Card Punch	0.065·F
2203 Model A1	Print	TB·0.0047·F
1419	Read	0.055·F
1259	Read	0.055·F
Comm Adapter	Transmit or Receive	0.070·F
BSCA	Transmit Transmit and Receive Receive Receive Initial Adprep Auto Call	0.054·F 0.054·F 0.053·F 0.053·F 0.053·F 0.045·F

Symbols: F = Input/Output field length specified in XIO instruction.

H = Number of print heads selected for printing.

TB = Number of characters in the typebar.
Note: 65 if typebar with 63-character set is used.

01790

CPU SUBMODELS 1 AND 2 (CONTINUED)

2311 Disk Timing

Operation	Timing (Average)
Read Data	145 usec + 180 usec + 2.5 (N) ms + 12.5 ms*
Read Count and Data	145 usec + 180 usec + 2.5 (N + 1) ms + 12.5 ms*
Read Count	145 usec + 180 usec + 12.5 ms* + 40 usec + 2.5 (C) ms
Write Data	145 usec + 180 usec + 2.5 (N) ms + 12.5 ms*
Write Count and Data	40.325 ms for ten sectors (See Note)
Verify Count and Data	40.325 ms for ten sectors (See Note)
Verify Data	145 usec + 180 usec + 2.5 (N) ms + 12.5 ms*
Seek	145 usec + 220 usec + Access Motion Time
Recalibrate	145 usec + 100 usec + Access Motion Time
Sense	275 usec
Scan Low or Equal	145 usec + 180 usec + 12.5 ms* + 40 usec + 2.5 (N_1) ms
Scan Equal	145 usec + 180 usec + 12.5 ms* + 40 usec + 2.5 (N_1) ms
Scan High or Equal	145 usec + 180 usec + 12.5 ms* + 40 usec + 2.5 (N_1) ms

N = Number of sectors. Each sector requires 2.5 ms
* = This factor of 12.5 ms represents the average rotational delay incurred during a large number of operations
C = Number of count areas unsuccessfully attempted
N_1 = Number of sectors unsuccessfully scanned

NOTE: Minimum time delay between two consecutive full-track operations is 25 ms when performed on the same disk drive.

2415 Tape Timing

Initial Selection: 15.6 usec (Min); 31.0 usec (Max)

Time required to Read, Read Backward, and Write =
Number of bytes multiplied by time per byte plus inter-block gap time

2415 Model	Time/Byte (9-Track)	Time/Byte (7-Track)			Inter-Block Gap Time (Nominal)	
		800 bpi	556 bpi	200 bpi	9-Track	7-Track
1	66.6 usec	66.6 usec	95.9 usec	266.4 usec	32.0 ms	40.0 ms
2	66.6 usec	66.6 usec	95.9 usec	266.4 usec	32.0 ms	40.0 ms
3	66.6 usec	66.6 usec	95.9 usec	266.4 usec	32.0 ms	40.0 ms
4	33.3 usec	66.6 usec	95.9 usec	266.4 usec	32.0 ms	40.0 ms
5	33.3 usec	66.6 usec	95.9 usec	266.4 usec	32.0 ms	40.0 ms
6	33.3 usec	66.6 usec	95.9 usec	266.4 usec	32.0 ms	40.0 ms

Times tc be added to the above calculations:
When Reading or Writing from Loadpoint, add 204 ms
When changing from Forward to Backward status (or vice versa), add 205 ms

Timing of Non-Variable Operation (Total time including initiation)

Operation (All Models)	7-Track (All Densities)	9-Track		Data Rates
		800 bpi	1,600 bpi	Models 1, 2, 3: 15,000 Bytes/sec at 800 bpi 10,452 Bytes/sec at 556 bpi 3,750 Bytes/sec at 200 bpi Models 4, 5, 6: 30,000 Bytes/sec (1,600 bpi)
Write Tape Mark	219 ms	219 ms	220 ms	
Rewind Full Reel	4.0 minutes			
Rewind/Unload Full Reel	4.0 minutes			Tape Speed (All Models) 18.75 inches/sec
Order Disconnect Time for Rewind, or Rewind/Unload: 30 usec				Inter-Block Gap (All Models) 0.5 to 0.8 inch

01791

CPU SUBMODELS 3 AND 4

Processing Operation	Format	Mnemonic	Timing (Average in microseconds)
Branch on Condition	RR	BCR	78
Branch and Store	RR	BASR	94
Add	RR	AR	174
Subtract	RR	SR	183
Store Halfword	RX	STH	117
Branch Condition	RX	BC	110
Load Halfword	RX	LH	124
Compare Halfword	RX	CH	223
Add Halfword	RX	AH	209
Subtract Halfword	RX	SH	216
Branch and Store	RX	BAS	124
Test under Mask	SI	TM	132
Move	SI	MVI	104
Set PSW	SI	SPSW	149
And	SI	NI	123
Compare Logical	SI	CLI	122
Or	SI	OI	130
Halt and Proceed	SI	HPR	105
Move Numerical	SS	MVN	403 + 16N
Move Characters	SS	MVC	403 + 16N
Move Zone	SS	MVZ	403 + 16N
Compare Logical	SS	CLC	392 + 24B
Edit	SS	ED	431 + 26N
Move with Offset	SS	MVO	$436 + 10.2N_1 + 7.8N_2$ (if $N_1 > N_2$) $436 + 18N_1$ (if $N_1 \leq N$)
Pack	SS	PACK	$448 + 27N_2(N_2 + \frac{P_3}{2} - 1) + 16.8P_1 + 9.6N_1 - 4.8N_2$ if $2N_1 - (N_2 + 1) \geq 0$ $427 + 27N_1$ if $2N_1 - (N_2 + 1) < 0$
Unpack	SS	UNPK	$444 + 9.6N_1 + 8.4N_2 + P_4 (5 + 4.4N_1 - 8.4N_2)$
Zero and Add Decimal	SS	ZAP	$476 + 10.2N_1 + 13.2N_2$
Compare Decimal	SS	CP	$578 + 13.2N_2 + (N_2 - 1)(69 + 15P_8) + (N_1 - N_2) \times (38 + 9P_8) + 12P_6$
Add Decimal	SS	AP	$578 + 13.2N_2 + (N_2 - 1)(69 + 15P_5) + (N_1 - N_2) \times (38 + 9P_5) + 12P_6 + P_7 (15 + 57N_1)$
Subtract Decimal	SS	SP	(The formula is the same for both instructions)
Multiply Decimal	SS	MP	$481 + 20N_2 + (N_1 - N_2)(70 + 9.6N_2) + \Sigma Op1_{dig} \times (39 + 9.6N_1 + 44N_2)$
Divide Decimal	SS	DP	$591 + 250N_2 + (N_1 - N_2)(150 + 250N_2) + \Sigma Q (100 + 50N_2)$
Translate	SS	TR	414 + 53N

Note: If the time-sharing switch is off, add 22 usec to the execution time of each instruction.
Indexing requires 56 usec per effectively-generated address.

Symbols used in the timing formulas:

B	= Total number of processed bytes in the first operand field
N_1	= Number of bytes in the first operand
N_2	= Number of bytes in the second operand
N	= Number of bytes in field (defined by coupled field length)
ΣQ	= Sum of quotient digits
$\Sigma Op1_{dig}$	= Sum of operand 1 digits

Constants:

P_1 = 1 if $2N_1 - (N_2 + 1)$ is odd 0 if $2N_1 - (N_2 + 1)$ is even	P_4 = 1 if $N_1 + 1 - 2N_2 < 0$ 0 if $N_1 + 1 - 2N_2 \geq 0$	P_6 = 1 if $L_1 > L_2$ 0 if $L_1 = L_2$
P_2 = 1 if $N_1 = N_2 = 1$ 0 otherwise	P_5 = 1 if Op Code Add and signs of operand 1 and operand 2 differ or if Op Code Sub and signs are alike 0 if Op Code Add and signs are alike or if Op Code Sub and signs differ	P_7 = 1 if Recomplement occurs 0 if no Recomplement occurs
P_3 = 1 if N_2 odd 0 if N_2 even		P_8 = 1 if signs of operand 1 and operand 2 are alike 0 if signs differ

CPU SUBMODELS 3 AND 4 (CONTINUED)

I/O Device	Input/Output Operation	Mnemonic	Timing (Average) in Microseconds
2560 Model A2	Read Card	XIO	486
	Punch Card	XIO	482
	Punch and Feed	XIO	482
	Test Reader/Punch Busy	TIOB	115
	Test Reader/Punch Error	TIOB	115
	Test Last Card	TIOB	115
	Test Feed Error	TIOB	136
	Stacker Select	CIO	120
2203 Model A2	Print	XIO	2366 + 45.2•F
	Test Printer Busy	TIOB	115
	Test Printer Error	TIOB	115
	Test Channel 9 or 12	TIOB	115
	Test Carriage Busy	TIOB	115
	Control Carriage	CIO	125
BSCA	Transmit Only	XIO	460
	Transmit and Receive	XIO	460
	Receive Only	XIO	466
	Receive Initial	XIO	464
	Adprep	XIO	458
	Auto Call	XIO	458
	Enable/Disable BSCA	CIO	106
	Enable/Disable ITB	CIO	106
	Store Current Address	CIO	149
	Store Sense	CIO	115
	Store ITB Address	CIO	149
	Test Any Check Condition	TIOB	106
	Test Busy	TIOB	103

01793

Average CPU Time Required for Time-Shared Input/Output Operations (in Milliseconds)		
Machine	Operation	Timing Formula
2560 Model A2	Read Card	0.065·F
	Punch Card	0.065·F
	Punch and Feed	0.065·F
2203 Model A2	Print	0.0047·F·C
BSCA	Transmit	0.054·F
	Transmit and Receive	0.054·F
	Receive	0.053·F
	Receive Initial	0.053·F
	Adprep	0.053·F
	Auto Call	0.045·F

Symbols: F = Input/Output field length specified in XIO instruction.

C = Number of characters in typebar
Note: 65 if typebar with 63-character set is used.

01795

2311 Disk Timing	
Operation	Timing (Average)
Read Data	12.5 ms* + 591 usec + 2.5 (N) ms
Read Count and Data	12.5 ms* + 591 usec + 2.5 (N) ms
Read Count	12.5 ms* + 631 usec + 2.5 (C) ms
Write Data	12.5 ms* + 591 usec + 2.5 (N) ms
Write Count and Data	38.091 ms for ten sectors (See Note)
Verify Count and Data	38.091 ms for ten sectors (See Note)
Verify Data	12.5 ms* + 591 usec + 2.5 (N) ms
Seek	631 usec + Access Motion Time
Recalibrate	511 usec + Access Motion Time
Sense	541 usec
Scan Low or Equal	12.5 ms* + 631 usec + 2.5 (N_1) ms
Scan Equal	12.5 ms* + 631 usec + 2.5 (N_1) ms
Scan High or Equal	12.5 ms* + 631 usec + 2.5 (N_1) ms

N = Number of sectors. Each sector requires 2.5 ms.

* = This factor of 12.5 ms represents the average rotational delay incurred during a large number of operations.

C = Number of count areas unsuccessfully attempted.

N_1 = Number of sectors unsuccessfully scanned.

Note: Minimum time delay between two consecutive full-track operations is 25 ms when performed on the same disk drive.

01794

CPU SUBMODEL 5

Processing Operation	Format	Mnemonic	Timing (Average) in Microseconds	Remarks
Branch on Condition	RR	BCR BCR BCR	24 34 38	If R2 is 0 If no branch If branch
Branch and Store Register	RR	BASR BASR	46 42	If R2 is 0 If R2 is not 0
Add Register	RR	AR	48	
Subtract Register	RR	SR	50	
Store Halfword	RX	STH	54	
Branch on Condition	RX	BC BC	36 44	If no branch If branch
Load Halfword	RX	LH	50	
Compare Halfword	RX	CH CH	54 60	If no overflow If overflow
Add Halfword	RX	AH	58	
Subtract Halfword	RX	SH	56	
Branch and Store	RX	BAS	56	
Set PSW	SI	SPSW	90	
Test under Mask	SI	TM TM TM	39 42 48	If condition code is 00 If condition code is 01 If condition code is 11
Move Immediate	SI	MVI	42	
AND Immediate	SI	NI	44	
OR Immediate	SI	OI	44	
Compare Logical Immediate	SI	CLI	36	
Halt and Proceed	SI	HPR	104	
Move Numerics	SS	MVN	54 + 6N	
Move Characters	SS	MVC	56 + 4N	
Move Zones	SS	MVZ	54 + 6N	
Compare Logical Characters	SS	CLC	48 + 4B	
Translate	SS	TR	38 + 18N	
Edit	SS	ED	60 + 46N digit sel + 54N signif start + 30N field sep + 22N other + 8N sign	
Move with Offset	SS	MVO	If $N_1 - N_2 > 0$, then $= 84 + 8N_1 + 12N_2$ If $N_1 - N_2$ is 0 or < 0, then $= 74 + 20N_1$	
Pack	SS	PACK	If $2N_1 - N_2 - 1 > 0$, then $= 90 + 22 \frac{(N_2 - P_3)}{2} - 12P_3 + 15 (2N_1 - N_2 - 1 - P_1) - 6P_1$ If $2N_1 - N_2$ is 0 or < 0, then $= 68 + 22N_1 - 2P_{10}$	
Unpack	SS	UNPK	If $N_1 + 1 - 2N_2 > 0$, then $= 74 + 8N_1 + 10N_2 - 2P_9$ If $N_1 + 1 - 2N_2$ is 0 or < 0, then $= 92 + 26 \frac{(N_1 - P_{11})}{2} - 4P_{11} - 2P_{10}$	
Zero Add Packed	SS	ZAP	$114 + 4N_{max}$	
Compare Packed (Decimal)	SS	CP	$176 + 10N_{max} + P_7 (20 + 4N_1) + 6P_5 + 6P_{12}$	
Add Packed (Decimal)	SS	AP	$126 + 6N_{max} + P_7 (20 + 4N_1) + 6P_5 + 6P_{12}$	
Subtract Packed (Decimal)	SS	SP	$128 + 6N_{max} + P_7 (20 + 4N_1) + 6P_5 + 6P_{12}$	
Multiply Packed (Decimal)	SS	MP	$176 + 82N_{max} - 56N_2 + (6N_2 + 22) Dig_{sum} + 9 (N_1 - N_2 - 2) (N_1 - N_2 - 1)$	
Divide Packed (Decimal)	SS	DP	$202 + 82N_{max} - 54N_2 + (6N_2 + 28) (Q_{sum} + 2N_1 - 2N_2 - 1)$	

Symbols used in the timing formulas:

N = Number of bytes in field (defined by common field length L)

N_1 = Number of bytes in operand 1 (defined by field length L1 + 1)

N_2 = Number of bytes in operand 2 (defined by field length L2 + 1)

N_{max} = Use the largest operand (either N_1 or N_2)

B = Total number of bytes processed in the first operand field

Dig sum = Sum of operand 1 digits

Q sum = Sum of quotient digits

P_1 = Use 1 if $2N_1 - (N_2 + 1)$ is odd; use 0 if $2N_1 - (N_2 + 1)$ is even

P_3 = Use 1 if N_2 is odd; use 0 if N_2 is even

P_5 in AP = Use 1 if signs of operands differ; use 0 if signs are equal

P_5 in SP/CP = Use 1 if signs of operands are equal; use 0 if signs differ

P_7 = Use 1 if recomplementation occurs; use 0 if no recomplementation

P_9 = Use 1 if $N_2 = 1$; use 0 if $N_2 > 1$

P_{10} = Use 1 if $N_1 = 1$; use 0 if $N_1 > 1$

P_{11} = Use 1 if N_1 is odd; use 0 if N_1 is even

P_{12} = Use 1 if overflow; use 0 if no overflow

N digit sel = Number of digit select characters

N signif start = Number of significant start characters

N field sep = Number of field separator characters

N other = Number of message insert characters

N sign = Number of signs

01986

CPU SUBMODEL 5 (CONTINUED)

I/O Device	Input/Output Operation	Mnemonic	Timing (average) in Microseconds	Remarks
2501	Read Card	XIO	190 + 22F + 84	F = Field length in XIO
	Test Reader Busy	TIOB	If branch, then = 86; if no branch, then = 84	
	Test Reader Error	TIOB	If branch, then = 90; if no branch, then = 82	
	Test Last Card	TIOB	If branch, then = 98; if no branch, then = 86	
2520	Read Card	XIO	115 (or 118 if col bin) x 2 + 22F + End	For End use:
	Punch Card	XIO	268 x 2 + 6F + 2183 x 2 + End	32 x 2 for last column read
	Punch and Feed	XIO	270 x 2 + 6F + 2183 x 2 + End	22 x 2 if punch and read
	Test Reader Busy	TIOB	41 x 2 (42 x 2 if branch)	16 x 2 if punch only
	Test Read Error	TIOB	40 x 2 (44 x 2 if branch)	75 x 2 if read only
	Test Punch Busy	TIOB	44 x 2 (45 x 2 if branch)	F = Field length in XIO
	Test Punch Error	TIOB	40 x 2 (44 x 2 if branch)	
	Test Last Card	TIOB	40 x 2 (48 x 2 if branch)	
	Test Feed Error	TIOB	43 x 2 (61 x 2 if branch)	
	Stacker Select	CIO	41 x 2	
2560	Read Card	XIO	224 + (F + 6) x 22	F = Field length in XIO
	Punch Card	XIO	194 + 256 + 98 x F	H = Number of print heads selected for printing
	Punch and Feed	XIO	194	
	Write Card	XIO	154 + 36 + (386 + 18 x H) x F	
	Test Reader/Punch Busy	TIOB	84	
	Test Read/Punch Error	TIOB	84	
	Test Card Printer Busy	TIOB	90	
	Test Last Card	TIOB	90	
	Test Feed Error	TIOB	136	
	Stacker Select	CIO	106	
	Print Head Select	CIO	94	
1442	Punch Card	XIO	74 + 88F + 154	F = Field length
	Test Punch Busy	TIOB	If branch, then = 24; if no branch, then = 22	
	Test Punch Error	TIOB	If branch, then = 32; if no branch, then = 26	
	Test Feed Error	TIOB	If branch, then = 62; if no branch, then = 30	
2203	Print	XIO	208 + 22F + 17CH + F x CH x 2	F = Field length of XIO
	Test Printer Busy	TIOB	50 x 2	CH = Character set on typebar such as 13,39,52, or 65
	Test Printer Error	TIOB	50 x 2	
	Test Channel 9 or 12 (overflow)	TIOB	50 x 2	
	Test Carriage Busy	TIOB	50 x 2	
	Control Carriage	CIO	87 x 2	
1403	Print with UCS	XIO	[(4.3 x F) + (240 - F) x 4 + 100] x 2	F = Field length in XIO
	Print without UCS	XIO	[(4.3 x F) + F + 100] x 2	
	Test Printer Busy	TIOB	31 x 2 + 13 x 2	
	Test Print Error	TIOB	31 x 2 + 13 x 2	
	Test Channel 9 or 12 (overflow)	TIOB	31 x 2 + 13 x 2	
	Control Carriage	CIO	30 x 2 + 45 x 2 + 12	
2152	Print with CR/LF	XIO	90 x 2 + (F + N_{shift} + 2) x 60 x 2	F = Field length in XIO
	Print without CR/LF	XIO	90 x 2 + (F + N_{shift} + 1) x 60 x 2	N_{shift} = Number of shift cycles required (lower to upper or vice versa)
	Carrier Return/Line Feed	CIO	80 x 2 + 60 x 2	
	Enable Request	CIO	46 x 2	
	Disable Request	CIO	46 x 2	
	Test Printer Busy	TIOB	90 x 2	
	Test Inquiry Request	TIOB	90 x 2	
	Test Any Check	TIOB	90 x 2	
	Test Cancel	TIOB	90 x 2	
	Test P1 Check	TIOB	90 x 2	
	Test P2 Check	TIOB	90 x 2	
	Read	XIO	90 x 2 + (F + 2) x 60 x 2	
BSCA	Transmit and Receive	XIO	86 + 216 + 56 + T + Turnaround-delay of modem	T = Time per block (in seconds) use:
	Transmit Only	XIO	86 + 240 + 56 + T	$(8 + F) \times \frac{C}{S}$ if internal clock;
	Receive Only	XIO	86 + 236 + 56 + TX	$(4 + F) \times \frac{C}{S}$ if external clock
	Receive Initial	XIO	86 + 216 + 56 + TX	
	Address Prepare	XIO	86 + 220 + 56	C = Number of bits per character
	Auto Call	XIO	86 + 204 + 56	S = Transmission speed in bits per second
	Enable/Disable BSCA	CIO	58 + 32	
	Enable/Disable ITB	CIO	58 + 34	TX = Time per block of remote station
	Store Current Address	CIO	54 + 92	
	Store Sense	CIO	54 + 30	F = Field length in XIO
	Store ITB Address	CIO	54 + 30	
	Test any Check	TIOB	52 + 30	
	Test Busy	TIOB	52 + 30	
SIOC (Serial I/O Channel)	Read Time-shared	XIO	200 + 12 x F + 20	F = Field length in XIO
	Write Time-shared	XIO	200 + 12 x F + 20	
	Read Burst Mode	XIO	200 + (request spacing x F) + 20	
	Write Burst Mode	XIO	200 + (request spacing x F) + 20	
	I/O Select	CIO	20	
	Test Transfer Line	TIOB	50	
IOC (Input/Output Channel)	Start I/O		25 + 250 (up to initial status)	
	Test I/O		25 + 250 (up to status presentation)	
	CPU interference in Overlap Mode		2 usec per halfword transferred	
	CPU interference in Non-Overlap Mode		100%	
	Additional time during Chaining		400 usec between each command	
	Data Transfer Rate		Depends upon attached unit only	
2311	Start or Test I/O		0.77 ms	N = Number of sectors
	Scan/Read/Write/Verify Data		0.9 + 2.5N + 12.5 (average in ms)	
	Read/Write/Verify Count and Data		0.9 + 2.5 (N+ 1) + 12.5 (average in ms)	
	Read Count		0.9 + 2.5N + 0.4 + 12.5 (average in ms)	
	Seek (including motion)		0.45 + 12.5 (average in ms)	
	Recalibrate (including motion)		0.3 + 400 (ms)	
	CPU interference in Overlap Mode		0.34 ms per record for all operations except Read Count. (For Read Count: 0.06 ms)	
	CPU interference in Non-Overlap Mode		100%	

01987

The following examples illustrate the use of System/360 instructions. Note that these examples closely approximate machine language to best illustrate the operation of the system. For clarity, the mnemonic for each operation code is used instead of the actual machine code. In addition, whenever possible, the contents of registers, storage locations, and so on, are given in decimal notation rather than the actual binary formats. When binary formats are used, they are segmented into bytes (eight bits) for ease of visual comparison.

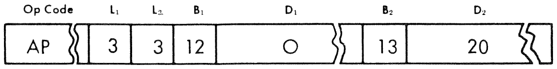
Decimal Add

The signed, packed decimal field at location 500-503 is to be added to the signed, packed decimal field at location 2000-2003.

Assume:

Reg 12	20 00
Reg 13	04 80
Loc 2000-2003 (before)	00 38 46 0-
Loc 500-503	01 12 34 5+

The instruction is:



Op Code	L_1	L_2	B_1	D_1	B_2	D_2
AP	3	3	12	0	13	20

01990

Loc 2000-2003 (after)	00 73 88 5+

Condition code = 2; sum is greater than zero.

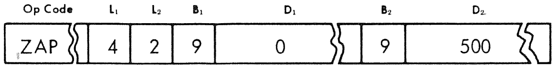
Zero and Add

The signed, packed decimal field at location 4500-4502 is to be moved to location 4000-4004 with four leading zeros in the result field.

Assume:

Reg 9	40 00
Loc 4000-4004 (before)	12 34 56 78 90
Loc 4500-4502	38 46 0-

The instruction is:



Op Code	L_1	L_2	B_1	D_1	B_2	D_2
ZAP	4	2	9	0	9	500

01991

Loc 4000-4004 (after)	00 00 38 46 0-

Condition code = 1; sum is less than zero.

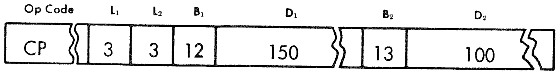
Compare Decimal

The contents of location 700-703 are to be compared algebraically with the contents of location 500-503.

Assume:

Reg 12	05 50
Reg 13	04 00
Loc 700-703	17 25 35 6+
Loc 500-503	06 72 14 2+

The instruction is:



Op Code	L_1	L_2	B_1	D_1	B_2	D_2
CP	3	3	12	150	13	100

01992

Condition code = 2; first operand is high.

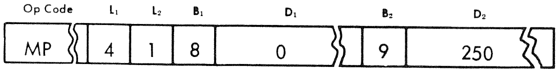
Multiply Decimal

The signed, packed decimal field in location 1200-1204 is to be multiplied by the signed, packed decimal field in location 500-501, and the product is to be placed in location 1200-1204.

Assume:

Reg 8	12 00
Reg 9	02 50
Loc 1200-1204 (before)	00 00 38 46 0-
Loc 500-501	32 1-

The instruction is:



Op Code	L_1	L_2	B_1	D_1	B_2	D_2
MP	4	1	8	0	9	250

01993

Loc 1200-1204 (after)	01 23 45 66 0+

Condition code: unchanged.

Divide Decimal

The signed, packed decimal field at location 2000-2004 is to be divided by the packed decimal field at location 3000-3001.

Assume:

Reg 12	18 00
Reg 13	25 00
Loc 2000-2004 (before)	01 23 45 67 8+
Loc 3000-3001	32 1-

The instruction is:

Op Code	L_1	L_2	B_1	D_1	B_2	D_2
DP	4	1	12	200	13	500

01994

Loc 2000-2004 (after)   38 46 0- 01 8+
where the quotient is 38460-and the remainder is 018+
Condition code: unchanged.

Pack

Assume locations 1000-1004 contain the following:

Z1 Z2 Z3 Z4 S5

where Z = four-bit zone code
S = four-bit sign code

The field is to be in packed format with two leading zeros and placed in location 2500-2503.

Reg 12	10 00
Reg 13	25 00
Loc 1000-1004	Z1 Z2 Z3 Z4 S5
Loc 2500-2503 (before)	A B C D

The instruction is:

Op Code	L_1	L_2	B_1	D_1	B_2	D_2
PACK	3	4	13	0	12	0

01995

Loc 2500-2503 (after)   00 12 34 5S
Condition code: unchanged.

Unpack

Assume locations 2501-2503 contains the following fields:

12 34 5S

This field is to be put into zoned format and placed in the locations 1000-1004 where: S is a four-bit sign code.

Reg 12	10 00
Reg 13	25 00
Loc 2501-2503	12 34 5S
Loc 1000-1004 (before)	A B C D E

The instruction is:

Op Code	L_1	L_2	B_1	D_1	B_2	D_2
UNPK	4	2	12	0	13	1

01996

and results in
Loc 1000-1004 (after)   Z1 Z2 Z3 Z4 S5
where Z is a four-bit zone code.
Condition code: unchanged.

Move with Offset

The unsigned three-byte field at location 4500-4502 is to be moved to location 5600-5603 and given the sign of the one byte field location at 5603.

Assume:

Reg 12	50 00
Reg 15	40 00
Loc 5600-5603 (before)	77 88 99 0+
Loc 4500-4502	12 34 56

The instruction is:

Op Code	L_1	L_2	B_1	D_1	B_2	D_2
MVO	3	2	12	600	15	500

01997

Loc 5600-5603 (after)   01 23 45 6+
Condition code: unchanged.

Move Immediate

A dollar sign ($) is to be placed in location 2100, leaving locations 2101-2105 unchanged. Let Z represent a four-bit zone.

Assume:

Reg 12	20 00
Loc 2100-2105 (before)	Z0 Z1 Z2 Z3 Z5 Z0

The instruction is:

Op Code	I_2	B_1	D_1
MVI	$	12	100

01998

Loc 2100-2105 (after) $ Z1 Z2 Z3 Z5 Z0
Condition code: unchanged.

Move Numeric

Let Z and Y represent four-bit zones. The numeric parts of the eight-bit characters in the field at locations 6070-6074 are to be replaced by the numeric parts of eight-bit characters at locations 8080-8084.

Assume:

Reg 12	60 00
Reg 15	80 00
Loc 6070-6074 (before)	Y1 Y2 Y3 Y4 Y5
Loc 8080-8084	Z3 Z6 Z9 Z7 Z8

The instruction is:

Op Code	L	B_1	D_1	B_2	D_2
MVN	4	12	70	15	80

01999

Loc 6070-6074 (after)   Y3 Y6 Y9 Y7 Y8
Condition code: unchanged.

Move Zones

Let Z and Y represent four-bit zones in the eight-bit characters making up the fields at location 2006-2010 and 3007-3011, respectively. The zones of the field at 2006-2010 are to be replaced by the zones from location 3007-3011.

Assume:

Reg 12	20 00
Reg 15	30 00
Loc 2006-2010 (before)	Z1 Z4 Z7 Z8 Z5
Loc 3007-3011	Y8 Y7 Y4 Y6 Y8

The instruction is:

Op Code	L	B_1	D_1	B_2	D_2
MVZ	4	12	6	15	7

02000

Loc 2006-2010 (after)	Y1 Y4 Y7 Y8 Y5

Condition code: unchanged.

Test Under Mask

Test bit positions 0, 2, 3, and 6 of a given byte in storage to determine if all of these bit positions contain 1s. A Test Under Mask instruction with a mask of 10110010 = 178_{10} is used. The byte to be tested is stored at location 1250 and contains 01101101.

Assume:

Reg 10	12 00

The instruction is:

Op Code	I_2	B_1	D_1
TM	178	10	50

02001

Mask from TM	10110010
Byte tested	01101101
Selected result	00100000

Condition code = 1; some selected bits are 0, some selected bits are 1 (mixed).

Translate

Assume a stream of 20 characters comes into location 2100 in EBCDIC. Translate to USASCII-8.

Assume:

Reg 12	20 00
Reg 15	10 00
Loc 2100-2119 (before)	JOHN JONES 257 W.95

The instruction is:

Op Code	L	B_1	D_1	B_2	D_2
TR	19	12	100	15	0

02002

Loc 2100-2119 (after)   JOHN JONES 257 W. 95
where the overbar means the same graphic in USASCII-8
Condition code: unchanged.

Translate Table

1000 … 1015

1000																
1064	BI										$\overline{\ }$	$\overline{\leftarrow}$	$\overline{(}$	$\overline{+}$		
1080	$\overline{\&}$										$\overline{\$}$	$\overline{*}$	$\overline{)}$			
1096	$\overline{-}$	$\overline{/}$									$\overline{,}$	$\overline{\%}$				
1112											$\overline{\#}$	$\overline{@}$	$\overline{\blacktriangledown}$	≡		
1128		$\overline{a}$	$\overline{b}$	$\overline{c}$	$\overline{d}$	$\overline{e}$	$\overline{f}$	$\overline{g}$	$\overline{h}$	$\overline{i}$						
1144		$\overline{j}$	$\overline{k}$	$\overline{l}$	$\overline{m}$	$\overline{n}$	$\overline{o}$	$\overline{p}$	$\overline{q}$	$\overline{r}$						
1160			$\overline{s}$	$\overline{t}$	$\overline{u}$	$\overline{v}$	$\overline{w}$	$\overline{x}$	$\overline{y}$	$\overline{z}$						
1176																
1192		$\overline{A}$	$\overline{B}$	$\overline{C}$	$\overline{D}$	$\overline{E}$	$\overline{F}$	$\overline{G}$	$\overline{H}$	$\overline{I}$						
1208		$\overline{J}$	$\overline{K}$	$\overline{L}$	$\overline{M}$	$\overline{N}$	$\overline{O}$	$\overline{P}$	$\overline{Q}$	$\overline{R}$						
1224			$\overline{S}$	$\overline{T}$	$\overline{U}$	$\overline{V}$	$\overline{W}$	$\overline{X}$	$\overline{Y}$	$\overline{Z}$						
1240	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$	$\overline{5}$	$\overline{6}$	$\overline{7}$	$\overline{8}$	$\overline{9}$						

1255

02003

Branch-On Condition

Assume a prior operation has been performed which resulted in setting the condition code in the PSW. The program is to branch when the condition code has any value above zero.

To cover all possible condition code settings (only one of four possible settings can be present at a time), the mask is set to 7 (binary 0111). This forces the program to branch when the condition code is either 01, 10, or 11 (binary). If the value in the B2 field of the instruction is 12 (for example), an effective branch address is generated by adding the contents of the D2 field to the contents of general register 12. Therefore, if register 12 contains a bit configuration that represents the decimal value of 400 and the D2 field contains, likewise, a value of 26, the resulting branch address is 426. The condition code is not changed.

Op Code	M1		B2		D2
BC	7	☒	12		26

Reg 12	0 0 0 0 0 0 0 1 1 0 0 1 0 0 0 0	= 400
Displacement (D2)	0 0 0 0 0 0 0 1 1 0 1 0	= 26
Result in D2 (Branch Addr)	0 0 0 1 1 0 1 0 1 0 1 0	= 426

01639

APPENDIX J. 4-OF-8/EBCDIC CONVERSION TABLE

Graphic	4 of 8		EBCDIC	
	N X O R	8 4 2 1	0 1 2 3	4 5 6 7
'	0 0 0 0	1 1 1 1	0 1 1 1	1 1 0 1
7	0 0 0 1	0 1 1 1	1 1 1 1	0 1 1 1
#	0 0 0 1	1 0 1 1	0 1 1 1	1 0 1 1
"	0 0 0 1	1 1 0 1	0 1 1 1	1 1 1 1
=	0 0 0 1	1 1 1 0	0 1 1 1	1 1 1 0
X	0 0 1 0	0 1 1 1	1 1 1 0	0 1 1 1
,	0 0 1 0	1 0 1 1	0 1 1 0	1 0 1 1
?	0 0 1 0	1 1 0 1	0 1 1 0	1 1 1 1
:	0 0 1 0	1 1 1 0	0 1 1 1	1 0 1 0
+	0 0 1 1	0 1 1 0	0 1 0 0	1 1 1 0
>	0 0 1 1	1 0 1 0	0 1 1 0	1 1 1 0
;	0 0 1 1	1 1 0 0	0 1 0 1	1 1 1 0
p	0 1 0 0	0 1 1 1	1 1 0 1	0 1 1 1
$	0 1 0 0	1 0 1 1	0 1 0 1	1 0 1 1
¬	0 1 0 0	1 1 0 1	0 1 0 1	1 1 1 1
-	0 1 0 0	1 1 1 0	0 1 1 0	0 0 0 0
(	0 1 0 1	0 1 1 0	0 1 0 0	1 1 0 1
_	0 1 0 1	1 0 1 0	0 1 1 0	1 1 0 1
)	0 1 0 1	1 1 0 0	0 1 0 1	1 1 0 1
C	0 1 1 0	0 0 1 1	1 1 0 0	0 0 1 1
E	0 1 1 0	0 1 0 1	1 1 0 0	0 1 0 1
F	0 1 1 0	0 1 1 0	1 1 0 0	0 1 1 0
I	0 1 1 0	1 0 0 1	1 1 0 0	1 0 0 1
¢	0 1 1 0	1 0 1 0	0 1 0 0	1 0 1 0
<	0 1 1 0	1 1 0 0	0 1 0 0	1 1 0 0
A	0 1 1 1	0 0 0 1	1 1 0 0	0 0 0 1
B	0 1 1 1	0 0 1 0	1 1 0 0	0 0 1 0
D	0 1 1 1	0 1 0 0	1 1 0 0	0 1 0 0
H	0 1 1 1	1 0 0 0	1 1 0 0	1 0 0 0
G	1 0 0 0	0 1 1 1	1 1 0 0	0 1 1 1
.	1 0 0 0	1 0 1 1	0 1 0 0	1 0 1 1
|	1 0 0 0	1 1 0 1	0 1 0 0	1 1 1 1

Graphic	4 of 8		EBCDIC	
	N X O R	8 4 2 1	0 1 2 3	4 5 6 7
&	1 0 0 0	1 1 1 0	0 1 0 1	0 0 0 0
3	1 0 0 1	0 0 1 1	1 1 1 1	0 0 1 1
5	1 0 0 1	0 1 0 1	1 1 1 1	0 1 0 1
6	1 0 0 1	0 1 1 0	1 1 1 1	0 1 1 0
9	1 0 0 1	1 0 0 1	1 1 1 1	1 0 0 1
0	1 0 0 1	1 0 1 0	1 1 1 1	0 0 0 0
@	1 0 0 1	1 1 0 0	0 1 1 1	1 1 0 0
T	1 0 1 0	0 0 1 1	1 1 1 0	0 0 1 1
V	1 0 1 0	0 1 0 1	1 1 1 0	0 1 0 1
W	1 0 1 0	0 1 1 0	1 1 1 0	0 1 1 0
Z	1 0 1 0	1 0 0 1	1 1 1 0	1 0 0 1
none	1 0 1 0	1 0 1 0	1 1 1 0	0 0 0 0
%	1 0 1 0	1 1 0 0	0 1 1 0	1 1 0 0
/	1 0 1 1	0 0 0 1	0 1 1 0	0 0 0 1
S	1 0 1 1	0 0 1 0	1 1 1 0	0 0 1 0
U	1 0 1 1	0 1 0 0	1 1 1 0	0 1 0 0
Y	1 0 1 1	1 0 0 0	1 1 1 0	1 0 0 0
L	1 1 0 0	0 0 1 1	1 1 0 1	0 0 1 1
N	1 1 0 0	0 1 0 1	1 1 0 1	0 1 0 1
O	1 1 0 0	0 1 1 0	1 1 0 1	0 1 1 0
R	1 1 0 0	1 0 0 1	1 1 0 1	1 0 0 1
!	1 1 0 0	1 0 1 0	0 1 0 1	1 0 1 0
*	1 1 0 0	1 1 0 0	0 1 0 1	1 1 0 0
J	1 1 0 1	0 0 0 1	1 1 0 1	0 0 0 1
K	1 1 0 1	0 0 1 0	1 1 0 1	0 0 1 0
M	1 1 0 1	0 1 0 0	1 1 0 1	0 1 0 0
Q	1 1 0 1	1 0 0 0	1 1 0 1	1 0 0 0
1	1 1 1 0	0 0 0 1	1 1 1 1	0 0 0 1
2	1 1 1 0	0 0 1 0	1 1 1 1	0 0 1 0
4	1 1 1 0	0 1 0 0	1 1 1 1	0 1 0 0
8	1 1 1 0	1 0 0 0	1 1 1 1	1 0 0 0
blank	1 1 1 1	0 0 0 0	0 1 0 0	0 0 0 0

01200

Graphic	EBCDIC		4 of 8	
	0123	4567	NXOR	8421
blank	0100	0000	1111	0000
¢	0100	1010	0110	1010
.	0100	1011	1000	1011
<	0100	1100	0110	1100
(	0100	1101	0101	0110
+	0100	1110	0011	0110
|	0100	1111	1000	1101
&	0101	0000	1000	1110
!	0101	1010	1100	1010
$	0101	1011	0100	1011
*	0101	1100	1100	1100
)	0101	1101	0101	1100
;	0101	1110	0011	1100
¬	0101	1111	0100	1101
-	0110	0000	0100	1110
/	0110	0001	1011	0001
,	0110	1011	0010	1011
%	0110	1100	1010	1100
–	0110	1101	0101	1010
>	0110	1110	0011	1010
?	0110	1111	0010	1101
:	0111	1010	0010	1110
#	0111	1011	0001	1011
@	0111	1100	1001	1100
'	0111	1101	0000	1111
=	0111	1110	0001	1110
"	0111	1111	0001	1101
A	1100	0001	0111	0001
B	1100	0010	0111	0010
C	1100	0011	0110	0011
D	1100	0100	0111	0100
E	1100	0101	0110	0101

Graphic	EBCDIC		4 of 8	
	0123	4567	NXOR	8421
F	1100	0110	0110	0110
G	1100	0111	1000	0111
H	1100	1000	0111	1000
I	1100	1001	0110	1001
J	1101	0001	1101	0001
K	1101	0010	1101	0010
L	1101	0011	1100	0011
M	1101	0100	1101	0100
N	1101	0101	1100	0101
O	1101	0110	1100	0110
P	1101	0111	0100	0111
Q	1101	1000	1101	1000
R	1101	1001	1100	1001
none	1110	0000	1010	1010
S	1110	0010	1011	0010
T	1110	0011	1010	0011
U	1110	0100	1011	0100
V	1110	0101	1010	0101
W	1110	0110	1010	0110
X	1110	0111	0010	0111
Y	1110	1000	1011	1000
Z	1110	1001	1010	1001
0	1111	0000	1001	1010
1	1111	0001	1110	0001
2	1111	0010	1110	0010
3	1111	0011	1001	0011
4	1111	0100	1110	0100
5	1111	0101	1001	0101
6	1111	0110	1001	0110
7	1111	0111	0001	0111
8	1111	1000	1110	1000
9	1111	1001	1001	1001

02004

APPENDIX K. CARD CODE TABLE

Extended Binary-Coded Decimal Interchange Code (EBCDIC) Card Code Table

Bit Positions of a Byte 0, 1, 2, 3, 4, 5, 6, 7

4567	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
0000	TO9 18	TE9 18	EO9 18	TEO9 18	Blank	T &	E -	TEO	TO 18	TE 18	EO 18	TEO 18	TO	EO	O 28	O 0	0
0001	T9 1	E9 1	O9 1	9 1	TO9 1	TE9 1	01 /	TEO9 1	TO 1	TE 1	EO 1	TEO 1	T1 A	E1 J	EO9 1	1 1	1
0010	T9 2	E9 2	O9 2	9 2	TO9 2	TE9 2	EO9 2	TEO9 2	TO 2	TE 2	EO 2	TEO 2	T2 B	E2 K	02 S	2 2	2
0011	T9 3	E9 3	O9 3	9 3	TO9 3	TE9 3	EO9 3	TEO9 3	TO 3	TE 3	EO 3	TEO 3	T3 C	E3 L	03 T	3 3	3
0100	T9 4	E9 4	O9 4	9 4	TO9 4	TE9 4	EO9 4	TEO9 4	TO 4	TE 4	EO 4	TEO 4	T4 D	E4 M	04 U	4 4	4
0101	T9 5	E9 5	O9 5	9 5	TO9 5	TE9 5	EO9 5	TEO9 5	TO 5	TE 5	EO 5	TEO 5	T5 E	E5 N	05 V	5 5	5
0110	T9 6	E9 6	O9 6	9 6	TO9 6	TE9 6	EO9 6	TEO9 6	TO 6	TE 6	EO 6	TEO 6	T6 F	E6 O	06 W	6 6	6
0111	T9 7	E9 7	O9 7	9 7	TO9 7	TE9 7	EO9 7	TEO9 7	TO 7	TE 7	EO 7	TEO 7	T7 G	E7 P	07 X	7 7	7
1000	T9 8	E9 8	O9 8	9 8	TO9 8	TE9 8	EO9 8	TEO9 8	TO 8	TE 8	EO 8	TEO 8	T8 H	E8 Q	08 Y	8 8	8
1001	T9 18	E9 18	O9 18	9 18	T 18	E 18	O 18	18	TO 9	TE 9	EO 9	TEO 9	T9 I	E9 R	09 Z	9 9	9
1010	T9 28	E9 28	O9 28	9 28	T28 ¢	E28 !	TE	28 :	TO 28	TE 28	EO 28	TEO 28	TO9 28	TE9 28	EO9 28	TEO9 28	A
1011	T9 38	E9 38	O9 38	9 38	T38 .	E38 $	O38 ,	38 #	TO 38	TE 38	EO 38	TEO 38	TO9 38	TE9 38	EO9 38	TEO9 38	B
1100	T9 48	E9 48	O9 48	9 48	T48 <	E48 *	O48 %	48 @	TO 48	TE 48	EO 48	TEO 48	TO9 48	TE9 48	EO9 48	TEO9 48	C
1101	T9 58	E9 58	O9 58	9 58	T58 (	E58)	O58 -	58 '	TO 58	TE 58	EO 58	TEO 58	TO9 58	TE9 58	EO9 58	TEO9 58	D
1110	T9 68	E9 68	O9 68	9 68	T68 +	E68 ;	O68 >	68 =	TO 68	TE 68	EO 68	TEO 68	TO9 68	TE9 68	EO9 68	TEO9 68	E
1111	T9 78	E9 78	O9 78	9 78	T78 |	E78 ¬	O78 ?	78 "	TO 78	TE 78	EO 78	TEO 78	TO9 78	TE9 78	EO9 78	TEO9 78	F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

Hexadecimal Representation for Bits 0 1 2 3 (Data Switch 1)

Hexadecimal Representation for Bits 4567 (Data Switch 2)

The numbers contained in the blocks represent row punches in the card.
"T" indicates a twelve zone punch.
"E" indicates an eleven zone punch.
EBCDIC graphic characters are shown in the shaded areas.

01207

Row	Column	0	1	2	3	4	5	6	7
	Bits → 765								
	↓	0	0	0	0	1	1	1	1
		0	0	1	1	0	0	1	1
	4 3 2 1	0	1	0	1	0	1	0	1
0	0 0 0 0	NUL	DLE	SOB-0 Trailer	ACK-0 Trailer	\	P	@	p
1	0 0 0 1	SOH	DC1	SOB-1 Trailer	ACK-1 Trailer	A	Q	a	q
2	0 0 1 0	STX	DC2	"	2	B	R	b	r
3	0 0 1 1	ETX	DC3	#	3	C	S	c	s
4	0 1 0 0	EOT	DC4	S	4	D	T	d	t
5	0 1 0 1	ENQ	NAK	%	5	E	U	e	u
6	0 1 1 0	ACK	SYN	&	6	F	V	f	v
7	0 1 1 1	BEL	ETB	'	7	G	W	g	w
8	1 0 0 0	BS	CAN	(	8	H	X	h	x
9	1 0 0 1	HT	EM	)	9	I	Y	i	y
10	1 0 1 0	LF	SS	*	:	J	Z	j	z
11	1 0 1 1	VT	ESC	+	SAK Trail.	K	[	k	{
12	1 1 0 0	FF	FS	,	RVI Trail.	L	~	l	¬
13	1 1 0 1	CR	GS	—	=	M	]	m	}
14	1 1 1 0	SO	RS	.	>	N	^	n	|
15	1 1 1 1	SI	ITB	/	WABT Trailer	O	_	o	DEL

▨ = Column 3 Characters

Note: The trailer charactes have a normal data function when not preceded by DLE; the following trailers represent graphics as well:

ASK - 0	= Digit 0
ACK - 1	= Digit 1
SAK	= Graphic ;
RVI	= Graphic <
WABT	= Graphic ?

Figure 41. USASCII Table 01259

	Bit positions 0,1 contain	00				01				10				11			
	Bit positions 2,3 contain	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
	Bit positions 4,5,6,7 contain																
0	0 0 0 0	NUL	DLE	DS		SOB-0 Trailer	&	-	▨ ACK-0 Trailer					\	~		0
1	0 0 0 1	SOH	DC1	SOS			SOB-1 Trailer	▨ ACK-1 Trailer		a	j			A	J		1
2	0 0 1 0	STX	DC2	FS	SYN			▨		b	k	s		B	K	S	2
3	0 0 1 1	ETX	DC3						▨	c	l	t		C	L	T	3
4	0 1 0 0	PF	RES	BYP	PN			▨		d	m	u		D	M	U	4
5	0 1 0 1	HT	NL	LF	RS				▨	e	n	v		E	N	V	5
6	0 1 1 0	LC	BS	EOB	UC				▨	f	o	w		F	O	W	6
7	0 1 1 1	DEL	IL	PRE	EOT			▨		g	p	x		G	P	X	7
8	1 0 0 0		CAN					▨		h	q	y		H	Q	Y	8
9	1 0 0 1		EM						▨ `	i	r	z		I	R	Z	9
10	1 0 1 0	SMM	CC	SM		¢	!		▨ :								
11	1 0 1 1	VT	CU1	CU2	CU3	.	$	▨ SAK Trail.	#	{	}						
12	1 1 0 0	FF	FLS		DC4	<	*	%	▨ RVI Trail.								
13	1 1 0 1	CR	GS	ENQ	NAK	(	)	▨ _	'			[	]				
14	1 1 1 0	SO	RDS	ACK		+	;	▨ >	=								
15	1 1 1 1	SI	ITB	BEL	SUB	|	¬	?	▨ WABT Trailer								

▨ = Column 3 Characters

Notes: 1. EOB is called ETB in the section "IBM Binary Synchronous Communication Adapter" of the manual.

2. The trailer characters have a normal data function when not preceded by DLE; the following trailers represent graphics as well:

SAK = Graphic ;
RVI = Graphic @

Figure 42. EBCDIC Table 01261

Bit Position		Valid on XIO Insn during		Valid with Auto Call Insn
		TSM	REC	
0	TSM/REC Mode Error		X	
1	EOT Received		X	
2	CRC/LRC Check		X	
3	Timeout		X	X
4	Short Record (Field Length Zero)	X	X	
5	Storage Wraparound	X	X	X
6	Parity Check (VRC or BSCA Check)	X	X	
7	Overrun	X	X	

Figure 43. BSCA Sense Byte Table 01257

APPENDIX M. DATA TRANSLATOR CONVERSION TABLES (2415 WITH DATA TRANSLATOR)

	--	-A	B-	BA
----	bl	ƀ	-	&+
---1	1	/	J	A
--2-	2	S	K	B
--21	3	T	L	C
-4--	4	U	M	D
-4-1	5	V	N	E
-42-	6	W	O	F
-421	7	X	P	G
8---	8	Y	Q	H
8--1	9	Z	R	I
8-2-	0	ǂ	!	?
8-21	#=	,	$	.
84--	@'	%(	*	⌑)
84-1	:	γ	]	[
842-	>	\	;	<
8421	✓	⧻	Δ	‡

Figure 44. Conversion Table, BCD Code of Graphics 02830

To find the translated equivalent of any character, look up the graphic in Figure 44, find the identical graphic in Figure 45, or vice versa, then check the bit configuration of the graphic in the respective figures. Some of the graphics shown in Figure 45 are not those printed when BCD is translated into EBCDIC; to find the printed EBCDIC graphic, refer to Appendix B.

NOTE: If even redundancy is set, the EBCDIC blank is converted to the BCD substitute blank, and vice versa. With the data translator turned off, BCD characters are set into the byte as 00BA8421.

Figure 46 lists the graphics by name.

Bit Positions of Byte → 01	00				01				10				11			
→ 23 → 4567	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0000					bl	&+	-						?	!	ǂ	0
0001							/						A	J		1
0010													B	K	S	2
0011													C	L	T	3
0100													D	M	U	4
0101													E	N	V	5
0110													F	O	W	6
0111													G	P	X	7
1000													H	Q	Y	8
1001													I	R	Z	9
1010								ƀ								
1011					.	$	,	#=								
1100					⌑)	*	%(	@'								
1101					[	]	γ	:								
1110					<	;	\	>								
1111					‡	Δ	⧻	✓								

Figure 45. Conversion Table, EBCDIC to BCD 02831

Symbol	Name
bl	Blank (Space)
ƀ	Substitute Blank
-	Minus Sign, Hyphen
&	Ampersand
/	Slash
\	Backslash
<	Less-Than Sign
>	Greater-Than Sign
#	Number Sign
@	At Sign
*	Asterisk
⌑	Lozenge
'	Prime, Apostrophe
(	Left Parenthesis
)	Right Parenthesis
[	Left Bracket
]	Right Bracket
‡	Record Mark
⯒	Group Mark
△	Mode Change
⧻	Segment Mark
√	Tape Mark (Radical)
⋎	Word Separator
=	Equal Sign

Figure 46. Names of Graphics 02832

GLOSSARY

Acknowledgment: In Teleprocessing, a message from the addressee sent in response to a communication received.

Adapter: A device used to interconnect two machines of different characteristics.

Address: 1. A number which designates a register, a storage location or an input/output device.
2. The part of an instruction which specifies the location of an operand.

Available: A condition in which part of a data processing system is operational, does not contain data or error check conditions, and is not busy with a previously initiated operation.

Bid: A procedure which establishes the right to start the first transmission to one or the other Teleprocessing terminal of a point-to-point data line.

Binary: A number system whose successive digits are interpreted as coefficients of the successive powers of the base 2.

Binary Coded Decimal: A decimal notation in which the individual decimal digits are each represented by four binary digits having a position value of 8-4-2-1. For example, in binary coded decimal notation, the number 23 is represented as $\begin{matrix}(8 & 421 & 8 & 421)\\ 0 & 010 & 0 & 011\end{matrix}$, whereas binary notation 23 is represented as $\begin{matrix}(16 & 8 & 421)\\ 1 & 0 & 111\end{matrix}$

Bit: The contraction of "binary digit", the smallest unit of information in the System/360 Model 20. A bit can have either of the two binary values: zero or one.

Bit stream: A binary signal without regard to grouping by character.

Block (Blocking): 1. To group records for the purpose of conserving storage space or increasing the efficiency of access or processing.
2. A physical record so constituted, or a portion of a telecommunications message defined to be a unit of data transmission.

Branching: A deviation from the strict sequence of a program.

Buffer: A storage device needed to compensate for a difference in rate of data flow, when transmitting data from one device to another.

Burst Mode: An operating mode in which the operation of a high-speed I/O device excludes all other input/output or processing operations.

Busy: A condition in which part of a data processing system is not available for use.

Byte: The basic unit of information in the System/360 Model 20. Every byte consists of eight bits, each having a value of zero or one.

CC2 Interrupt: An interrupt which is requested on completion of all time-shared operations to notify high-speed I/O devices that the CPU is free for burst-mode operations. The term "CC2" indicates that the condition code is set to 2 when a burst-mode operation is attempted at a time when time-shared operations are in progress. The prerequisites for the execution of a CC2 interrupt are that the channel mask bit in the PSW is on and that no interrupt request with a higher priority is pending.

Channel Command: A command initiated for an I/O operation such as read or write. It is stored in main storage and, unlike an instruction, is decoded and executed by the channel. (A similar command exists for the storage control feature.)

Channel Command Word: An area of information in main storage that specifies the operation that a channel-connected I/O device is to perform. (A similar structure exists for the storage control feature.)

Channel End: A signal sent by a channel-connected I/O device to indicate that the channel facilities are no longer needed. (A similar signal is sent by storage control connected devices.)

Channel Mask Bit: A bit in the PSW which determines whether the CPU is interruptible or not. The CPU is interruptible when the channel mask bit is set to 1, and is not interruptible if the bit is 0.

Channel Status Word: A group of 48 bits in a fixed main-storage location (009C-00A1) which represents status information on the input/output channel and the unit addressed via the channel. (Storage location 009C-00A1 is also used by the storage control feature to store a similar status word.)

Character: One of a set of elementary signals which may include decimal digits 0 through 9, the letters A through Z, punctuation marks, and any other symbols acceptable to a computer for reading, writing, or storing.

Column Binary: A coding system that allows the capacity of an IBM punched card to be increased from 80 to 160 storage positions by using the six-bit BCD code. Every punch position represents a bit (=1 if hole, =0 if no hole). Sets of bits are read by card column, two 6-bit sets per column. The first set consists of the 12-11-0-1-2-3 positions; the second set consists of the 4-5-6-7-8-9 positions.

Command Chaining: The uninterrupted execution of a sequence of commands applying to the same device. The commands must be stored adjacent to each other in ascending order of address. (Chaining occurs only in the input/output channel.)

Command Code: This specifies for the input/output channel and addressed device the exact operation (such as read, write, sense, control) to be performed. The command code is located in bits 0 through 7 of the CCW. (Command codes are also used with the storage control feature.)

Common Carrier: A company which furnishes communications services to the general public, and which is regulated by the appropriate local or central authorities.

Communication: The transmission of messages between points of origin and reception without alteration of the sequence or structure of the information contents.

Condition Code: A code, located in bits 2 and 3 of the PSW, to reflect certain conditions in the central processing unit. The two binary bits permit four possible combinations (00, 01, 10, or 11) which indicate the result of successful operations or report on the conditions leading to unsuccessful operations. The condition code can be tested by the program to allow branching decisions.

Configuration: The specific combination of data processing units forming a system.

Contention: A condition on a point-to-point line that exists when the right to the first transmission is not established, and thus both stations may attempt to transmit at the same time.

Control Character: A character used in BSCA transmission to initiate, modify, or stop operations.

Control Instruction: This directs an I/O device to perform a specified function, e.g., to select a stacker pocket or to initiate a carriage skip. (Control instructions are used for all devices except channel-connected or storage control-connected devices; these use control commands.)

Control Station: See "Master Station".

Control Unit: A device specifically designed to communicate with the input/output channel, and to start and supervise operations in the individual I/O devices connected.

Control Unit End: A signal sent by a control unit when the control-unit-busy condition ends.

Cylinder: A set of ten vertically arranged disk tracks.

Data Conversion: A feature that allows any System/360 Model 20 data to be written on seven-track tape with maximum packing efficiency. Three eight-bit bytes are written on seven-track tape as four six-bit characters with odd parity, and vice versa.

Data Link: The communication lines, modems, and communication controls of all stations connected to the line, used in the transmission of information between two or more stations.

Data Set: A device which converts digital dc signals into frequency-encoded signals to allow transmission over telephone lines. The device also reconverts frequency-encoded signals received from a remote station into digital dc signals and performs certain control functions required to establish the data link. (Also called "modem".)

Decimal Number: A number in which individual decimal digits ranging from 0 to 9 are each represented by four binary bits. Decimal numbers have a variable field length from 1 to 16 bytes and may be in zoned format or packed format. In zoned format, the sign (+ or -) is in the left half of the low-order byte; in packed format, the sign is in the right half of the low-order byte.

Demodulation: The process of retrieving intelligence (data) from a modulated carrier wave; the reverse of modulation.

Device End: A signal sent by a channel-connected device when the device-busy condition ends.

Diagnostic Routine: A routine designed to locate a malfunction, either in other routines or in the computer.

Direct Access: The retrieval or storage of data by a reference to its location on a volume, rather than relative to the previously retrieved or stored data.

Direct Addressing: A method of addressing main storage by which the 14 low-order bits of the combined base and displacement field are used to refer directly to byte locations in main storage.

Displacement: A value contained in the D1 or D2 field of an instruction. This value may be added to the so-called base address (contained in the register specified by the B1, B2, or R2 field of an instruction) to form an effective address.

Donor: The designation of a data station when its primary function is that of transmitter.

Duplex Transmission: The simultaneous two-way independent transmission in both directions. (Also called "full-duplex" transmission.)

EBCDIC: Extended binary-coded-decimal interchange code. This code is based on eight binary bits (byte), allowing any one of 256 characters to be represented.

Effective Addressing: A method of addressing main storage by which the contents of the general register, specified by the B-field of an instruction, are added to the contents of the D-field of the instruction. (Also known as "indexing".)

Explicitly Addressed Operand: An operand specified by means of a main-storage address, and not located in the instruction itself (immediate operand) or in a general register.

Field: A contiguous set of digits or characters forming a meaningful entity.

Fixed-Length Record: A record having a specific length that cannot be altered.

Format: The general makeup of data, instructions, records, or files.

General Register: An auxiliary storage unit in the CPU, used for temporary storage of data. The System/360 Model 20 has eight general registers, each accommodating two bytes (one halfword) of data.

Graphic: The visual representation of a character or symbol.

Halfbyte: The leftmost or rightmost four bits of an eight-bit byte. It can contain the representation of a digit, a zone, or the sign of a number.

Half-duplex Transmission: In communications facilities, this is an alternate, one-way-at-a-time transmission.

Halfword: Two adjacent bytes where the left byte is on a halfword boundary.

Halfword Binary Number: A whole number in binary notation with a fixed length of two bytes, having its sign (+ or -) in the leftmost bit.

Hexadecimal: A number system whose successive digits are interpreted as coefficients of the successive powers of the base 16. The values 0 through 15 are written as digits 0 through 9 (for values 0 - 9) and alphabetic characters A through F (for values 10-15); thus a hexadecimal position may have any value from zero to F (see Appendix D).

High-Order Bit: The bit position of a given storage area representing the highest numeric value in binary notation. (Also referred to as the "leftmost bit".)

Immediate Operand: A byte of data located in the instruction.

Indexing: See "Effective Addressing".

Input: The data that is fed into a computer. The input may be data that is to be processed or may be variable parameters or instructions that are supplied to implement certain operations.

Input/Output Channel: An electronic device for processing instructions and commands which are standardized for a number of different I/O devices that communicate with the CPU through the channel facilities exclusively.

Input/Output Operations: The transference of information to main storage from sources external to the CPU and/or from main storage to external destinations.

Integral Boundary: An address boundary which is a multiple of the length of the unit of information stored. A halfword (two bytes) binary number, for example, is addressed at an integral boundary that is a multiple of the number 2.

Interblock Gap: A blank length of tape that separates two physical data records, and generated by the start inertia of the tape drive.

Interface: The line of demarcation between two pieces of equipment having different functions.

Interruption (Interrupt): An automatic branching in the stored program which alerts the system to the end of a data transfer. The object of the interrupt system is to indicate the availability of data received from I/O devices at the earliest moment possible, without forcing the program to continually check on the I/O device to find its end condition.

Leased Line: A communications facility reserved for the sole use of a single leasing customer.

Leftmost Bit: See "High-Order Bit".

Logical Data: This consists of alphabetic or numeric character codes and may be in a one-byte fixed length or in variable length ranging from one to 256 bytes. There is no sign.

Logical Functions: Non-arithmetic data processing steps in the CPU, such as moving and comparing operands, editing, and testing.

Logical Record: A record from the standpoint of its contents, function, and use, rather than its physical attributes; i. e. , a record that is defined in terms of the information it contains.

Low-Order Bit: The bit position of a given storage area representing the lowest numeric value in binary notation. (Also referred to as the "rightmost bit".)

Main Storage: All addressable storage, from which instructions or data can be read out and executed or into which instructions or data can be loaded. General-purpose registers are outside of main storage.

Master Station: A unit having control of all other terminals on a multipoint network for the purposes of polling and/or selection. (Also termed "control station".)

Mnemonic: In programming terminology, a contraction or abbreviation whose characters are suggestive of the full expression and are therefore easy to remember.

Mode Set Command: A command issued to set recording density, parity mode, data converter and translator on or off in magnetic tape units. A particular setting is retained until a new mode set command is issued for the control unit concerned.

Modem: See "Data Set".

Modulation: The process by which some characteristic of one wave or signal is varied in accordance with another wave or signal. This technique is used in data sets (modems) to make data-processing machine signals suitable for transmission over telephone lines.

Multipoint Data Exchange System: A Teleprocessing system in which one station always has control and the other stations act as tributaries.

Next Sequential Instruction: The instruction due to be read out immediately following the current instruction, and addressed by the rightmost two bytes of the PSW.

Non-Overlap Mode: A mode of operation in which an input/output operation excludes other input/output operations and internal CPU processing.

Non-Return to Zero IBM (NRZI): A tape recording method in which bits are recorded by reversing the magnetic flux polarity. A series of one bits is therefore represented by flipping successively plus-to-minus, minus-to-plus, plus-to-minus, and so on. For zero bits, the polarity maintains its current level. As the polarity is never zero, the method is termed "non-return to zero".

Not Operational: A condition in which part of a data processing system is in a not-ready status, or has an error, or a data check condition, or power off.

Offline: Pertaining to equipment or devices not under direct control of the central processing unit.

Online: Pertaining to equipment or devices under direct control of the central processing unit.

Operand: An item of information which is to be processed by an instruction. Instructions will therefore always contain addresses that refer to the location of one or two operands so that these may be fetched from their location and processed. An operand can also be located in the instruction.

Operation Code: The leftmost byte of an instruction which specifies the length of the instruction, the data format, and the operation to be performed.

Output: The resultant data produced by a computer program.

Overlap Mode: An extension of the System/360 Model 20 time-sharing principle that allows a maximum number of I/O devices to participate in overlapping. When the overlap mode bit (bit 5) of the PSW is on (CPU Submodel 5 only), the regular timesharing ability of the System/360 Model 20 is extended to include tape and disk operations.

Pack: To convert decimal data from zoned to packed format.

Packed Decimal: A data format in which two numeric digits, or one digit and a sign, are stored in one eight-bit byte.

Parity Bit: A binary digit appended to (or omitted from) an array of bits to make the sum of all one-bits in the array always odd (or always even).

Pending Channel Interrupt: An interrupt that has been requested but cannot be performed because the channel mask bit in the current PSW is zero.

Phase Encoding: A tape recording method in which all binary states (logical zeros and logical ones) are recorded by magnetic flux reversal. However, zero bits are represented by a flux direction that is identical with that of an oscillator, whereas one bits are represented by a flux direction opposite to that of the oscillator; thus the encoding method is by phase (either in phase or out of phase with the oscillator).

Physical Record: A record from the standpoint of the manner or form in which it is stored, retrieved and moved; i.e. a record that is defined in terms of physical qualities.

Point-to-Point Data Exchange System: A Teleprocessing system connecting only two stations for any one transmission.

Polling: A technique by which each of the terminals sharing a communications line is periodically interrogated to determine whether it requires servicing. The control station sends a poll which, in effect, requests the terminal to transmit.

Program Status Word: A four-byte area of information in auxiliary storage used by the internal CPU control to effect instruction sequencing and to hold and indicate the status of the system in relation to the program being executed.

Protected Area: The main-storage locations 0 through 143 which are reserved for internal CPU control and are not available to the program. Addresses which refer to this area are rejected as invalid.

Read Operation: This pertains to data transfer from an I/O device to the CPU.

Recipient: The designation of a data station when its primary function is that of receiver.

Record: A general term for any unit of data that is distinct from all others when considered in a particular context.

Rightmost Bit: See "Low-Order Bit".

RR Format: This denotes a register-to-register operation, i.e., both operands are located in general registers.

RX Format: This denotes a register-to-storage or storage-to-register operation, i.e., one operand is located in core storage and the other is located in a general register.

Scan Operation: In a disk storage drive, an operation that compares one 270-byte main-storage data area with as many disk records (270-byte areas) as specified by the sector count of the respective Scan command. The purpose of the operation is to find one disk record that is equal to, high or equal to, or low or equal to, the specified data area in main storage.

Sector: On a disk, one of the ten equal portions of recording area which together form a (circular) track. A sector accommodates 270 bytes of recording data.

Seek: A command that positions the access mechanism of a direct-access device on a specified location, and selects a specified head.

Sense Bytes: Bytes in main storage that contain status information on tape units and disk files, additional to the information located in the channel status word. Information is stored upon issuance of a Sense command.

Sense Operation: An operation similar to a read operation, except that the information read into main storage originates from status indicators and not from data files.

Serial I/O Channel: An electronic device that allows I/O devices using 6-bit BCD code to communicate with the 2020. The data transfer is parallel by bit, serial by character, whereby BCD characters are automatically translated to EBCDIC characters and vice versa.

SI Format: This denotes a storage-immediate operation, i.e., the first operand is located in core storage, and the second operand is located in the instruction.

SS Format: This denotes a storage-to-storage operation, i.e., both operands are located in main storage.

Storage Control Feature: An electronic attachment housed in the 2020 to permit 2311's to be connected and operated under stored program control.

Storage Control Status Word: A group of 48 bits stored in a fixed main storage location (009C-00A1). These bits reflect the condition of the storage control and the devices that operate through it. The status word allows the program to decide on further proceedings.

Storage Wraparound: An error arising when the storage capacity is exceeded during a transmit or receive operation or during I/O operations. (Also termed "Storage overflow.")

Switched Network: A term that denotes common carrier (public) telephone lines.

Synchronization Character: A character that is automatically inserted into the data stream of a communications facility to establish and maintain synchronization.

Synchronize: To lock one element of a system into step with another. The term usually refers to locking a receiver to a transmitter, to allow the receiver to recognize the beginning and end of a data unit within the bit stream.

Synchronous: Having a constant time interval between bits or characters. The term implies that all equipment in the system is in step.

Telecommunications: A general term expressing information exchange over long distances.

Teleprocessing: A form of information handling which is characterized by the fact that data generated at one location is transmitted to another location via telephone lines for remote processing, with subsequent retransmission of the results.

Tête-à-Tête Mode: In Teleprocessing, the communication between stations in which each entry from one station elicits a response from the other, and vice versa. (Also called "limited conversation mode".)

Text Mode: A mode reserved for the information portion of a message transmitted by a communications facility.

Time Sharing: A means of overlapping input/output operations with each other and with processing.

Timeout: In Teleprocessing, the time interval allotted between operations before system operation is interrupted and must be restarted.

Track: On tape, one of the nine parallel recording areas (nine-track tape) or seven parallel recording areas (seven-track tape) arranged along the length of the tape. On disk, one of the 203 circular recording areas, arranged concentrically on the disk surface and each divided into ten equal portions termed "sectors". A disk track accommodates 2,700 bytes of recording data.

Turnaround Time: In modems or communication adapters, the actual time required to reverse the direction of transmission from send to receive, or vice versa, when operating in half-duplex mode.

Two's Complement: This pertains to the format of a halfword binary number which represents a negative value. The two's complement of a number is obtained by inverting each bit of the number and adding the value one to the low-order position.

Unblock: To change the format of a file so that a physical record comprises only one logical record.

Universal Character Set: A special feature allowing the printing of any set of graphics (up to 240 different characters) by a 1403 Model 2 or N1.

Unpack: To convert decimal data from packed to zoned format.

Update: To modify a unit of information or a data file so that it reflects the most recent events.

USASCII: United States of America Standard Code for Information Interchange. A code for data transfer adopted by the American Standards Association to achieve compatibility between data devices. USASCII is a seven-bit code, which is also available in an extended eight-bit version then termed USASCII-8.

Variable-Length Records: Logical records in which the number of bytes in each record is not fixed but may vary within prescribed limits.

Working: A condition in which part of a data processing system is executing a previously initiated operation.

Write Operation: This pertains to data transfer from the CPU to an I/O device.

Zone: The four leftmost bits of an unpacked (zoned) decimal number as represented in the EBCDIC or USASCII code. In EBCDIC, the zone has the value 15 and, in USASCII, it has the value 5. Zone bits do not affect the value of the numeric digit contained in the four rightmost bits of the byte.

Zoned Decimal: A data format in which the four rightmost bits of each 8-bit byte represent a decimal digit and the four leftmost bits represent a zone or sign.

READER'S COMMENT FORM

IBM System/360 Model 20 Functional Characteristics **Form A26-5847-5**

How did you use this publication?

As a reference source □

As a class-room text □

As a self-study text □

Based on your own experience, rate this publication:

As a reference source—Very Good □ Good □ Fair □ Poor □ Very Poor □

As a text—Very Good □ Good □ Fair □ Poor □ Very Poor □

What is your occupation?

We would appreciate your specific comments; please give page and line references where appropriate. If you wish a reply, be sure to include your name and address.

● Thank you for your cooperation. No postage necessary if mailed in U.S.A.

YOUR COMMENTS PLEASE . . . .

This SRL bulletin is one of a series which serves as reference sources for systems analysts, programmers and operators of IBM systems. Your answers to the questions on the back of this form, together with your comments, will help us produce better publications for your use. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Please note: requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

fold fold

FIRST CLASS
PERMIT NO. 1359
WHITE PLAINS, N.Y.

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM CORPORATION
112 EAST POST ROAD,
WHITE PLAINS, N.Y. 10601.

Attention: Publishing Operations (B)

fold fold

IBM®

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]

Printed in U.S.A. A26-5847-5

IBM

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]