Reference Manual

IBM 1410 Data Processing System

MINOR REVISION (November, 1961)

This edition, A22-1407-2, obsoletes A24-1407-0 but not A24-1407-1. Minor revisions have been made to bring information up to date. This edition also obsoletes these publications:

G24-1409-0	IBM 1410 Data Processing Systems Bulletin: New Features
G24-1430-0	IBM 1401 and 1410 DPS Bulletin: Numerical Print Feature for IBM 1403 Printer

Comments about this manual may be addressed to:

PRODUCT PUBLICATIONS
IBM CORPORATION
P.O. Box 390
Poughkeepsie, N. Y.

Contents

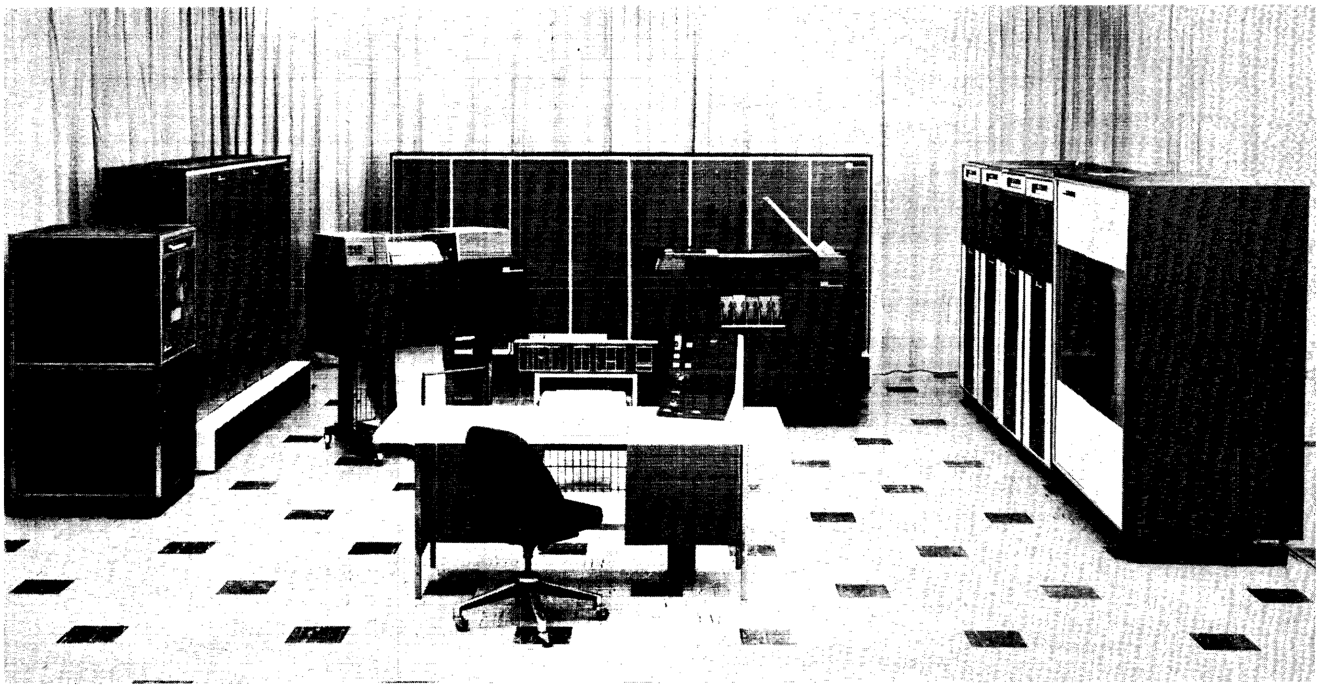


Figure 1. IBM 1410 Data Processing System

IBM 1410 Data Processing System

The first IBM 1400 Series Data Processing System, with its variable-word length, powerful editing ability, add-to-storage logic, program simplicity, and other features, filled a long-standing data-processing need for those businesses that could not justify the speed and cost of a large data processing system. This system can also serve as auxiliary equipment for large-scale computers because it can efficiently handle editing and tape preparation, off-line.

Now, IBM offers a new IBM 1400 Series System – the IBM 1410 Data Processing System (Figure 1). The basic concepts of the 1410 are similar to those of the 1401. Some of the standard features of the 1410 are:

1. Independent input-output synchronizer areas
2. Large storage capacity (10, 20, or 40 thousand storage positions)
3. Fast internal speed
4. Table lookup
5. Fifteen index registers

These features make it possible to process the larger volumes of data in the applications that characterize the intermediate data-processing area.

The 1410 is available in a variety of system configurations (Figure 2). This flexibility enables the user to select the components and features that satisfy his particular requirements and to expand the system as his needs increase.

Compatibility between the IBM 1401 and the IBM 1410 is designed into the 1410 system. This compatibility offers several advantages to the 1400-series customer:

1. It permits the transition from a 1401 to a 1410 with a minimum of system development costs.
2. It eliminates complete retraining of programmers because the basic system concepts are the same. The programmers need only learn additional 1410 features.
3. It enables machine operators to learn to use the 1410 with minimum retraining.
4. It permits the processing of many 1401 programs on the 1410 without altering the 1401 program.

SYSTEM COMPONENTS	CARD SYSTEM CONFIGURATIONS				TAPE SYSTEM CONFIGURATIONS		RAMAC CONFIGURATION	COMPONENT ATTACHMENTS	
	CARD ALONE	CARD—TAPE	CARD—RAMAC	CARD—TAPE—RAMAC	TAPE ALONE	TAPE—RAMAC		1011 PAPER TAPE READER	1412 MAG. CHAR. READER
IBM 1415 Console	Required	Required	Required	Required	Required	Required	Required	Required	Required
IBM 1411 Processing Unit Model 1, 2, 3	Required	Required	Required	Required	Required	Required	Required	Required	Required
IBM 1414 Input/Output Synchronizer									
Model 1		Model 1—729		Model 1—729	Model 1—729	Model 1—729	Not Available		Not Available
Model 2		Model 2—7330		Model 2—7330	Model 2—7330	Model 2—7330			
Model 3	Model 3	Model 3	Model 3	Model 3					
Model 4								Model 4	
IBM 1402 Card Read Punch Model 2	Model 2	Model 2	Model 2	Model 2	Not Available	Not Available	Not Available	Chart shows the Components that are required or available for all possible 1410 System Configurations. In addition, the Components needed for attaching other specific devices are also shown.	
IBM 1403 Printer Model 1 or 2	Model 1 or 2 Available	Model 1 or 2 Available	Model 1 or 2 Available	Model 1 or 2 Available	Not Available	Not Available	Not Available		
IBM 729 Mag. Tape Unit II and/or IV	Not Available	Model II and/ or IV Available	Not Available	Model II and/or IV Available	Model II and/or IV Available	Model II and/or IV Available	Not Available		
IBM 7330 Magnetic Tape Unit	Not Available	Available	Not Available	Available	Available	Available	Not Available		
IBM 1405 Disk Storage Unit Model 1 and/or 2	Not Available	Not Available	Model 1 and/or 2 Available	Model 1 and/or 2 Available	Not Available	Model 1 and/or 2 Available	Model 1 and/or 2 Available		

IBM 1411 PROCESSING UNIT SPECIAL FEATURES								NOTES
Card Read Punch Adapter	Required	Required	Required	Required	Not Available	Not Available	Not Available	Channel one only.
Tape Input/Output Adapter	Not Available	Required	Not Available	Required	Required	Required	Not Available	One for each channel used for magnetic tapes.
Disk Storage Adapter	Not Available	Not Available	Required	Required	Not Available	Required	Required	One for each channel used for 1405's.
Mag. Character Reader Adapter	Available	Available	Available	Available	Available	Available	Available	Channel one, Channel one and two.
Processing Overlap	Available	Available	Available	Available	Available	Available	Available	Channel one, Channel one and two.
Dual Synchronizer Adapter	Not Available	Available	Available	Available	Available	Available	Available	Prerequisite: Processing overlap. Provides two channel operation.

Figure 2. IBM 1410 Data Processing System Components

SYSTEM COMPONENTS	CARD ALONE	CARD—TAPE	CARD—RAMAC	CARD—TAPE—RAMAC	TAPE ALONE	TAPE—RAMAC	RAMAC CONFIGURATION	NOTES
IBM 1414 MODEL 3 AND 4 INPUT/OUTPUT SYNCHRONIZER SPECIAL FEATURES								
Printer Synchronizer Storage (Model 3)	Available	Available	Available	Available	Not Available	Not Available	Not Available	
Printer Synchronizer Storage—Additional (Model 3)	Available	Available	Available	Available	Not Available	Not Available	Not Available	
Paper Tape Reader Adapter (Model 4)	Available	Available	Available	Available	Not Available	Not Available	Not Available	
51-Col. Feed Adapter	Available	Available	Available	Available	Not Available	Not Available	Not Available	
Numeric Print Control	Available	Available	Available	Available	Not Available	Not Available	Not Available	
IBM 1405 DISK STORAGE UNIT SPECIAL FEATURES								
Disk Storage Control	Not Available	Not Available	1 Required For Each Channel	1 Required For Each Channel	Not Available	1 Required For Each Channel	1 Required For Each Channel	
Successive Disk Storage	Not Available	Not Available	1 Required For Each 1405	1 Required For Each 1405	Not Available	1 Required For Each 1405	1 Required For Each 1405	
Additional Access Arms	Not Available	Not Available	Available*	Available*	Not Available	Available*	Available*	

*Maximum of 2 Additional Arms Per Unit, a Total of 12 (Standard and Additional) Per System

NOTES:

1411 Processing Unit Model 1 has 10,000 storage positions
1411 Processing Unit Model 2 has 20,000 storage positions
1411 Processing Unit Model 3 has 40,000 storage positions

1403 Printer Model 1 has 100 print positions
1403 Printer Model 2 has 132 print positions } Either Model can be equipped with the numeric print feature

729 Model II moves tape at 75 inches per second
729 Model IV moves tape at 112.5 inches per second } Can be intermixed on same channel

1405 Model 1 has storage capacity of 10 million characters
1405 Model 2 has storage capacity of 20 million characters
This Chart is intended only as a guide for 1410 System Configurations.
For specific details contact your IBM sales representative.

Figure 2. (Continued)

Features

Solid State Circuitry

Transistorization of components contributes to the total efficiency of the IBM 1410 Data Processing System. Standard Modular System (SMS) cards with printed circuits are easy to maintain, are relatively inexpensive, and increase reliability in the system (Figures 3 and 4). Space requirements, heat dissipation, and power and air-conditioning requirements are carefully controlled.

Card Input and Output

The IBM 1402 Card Read-Punch, Model 2 (Figure 5), can read and punch cards. The 1402 has a read feed and punch feed.

The read feed can read as many as 800 cards per minute. Its standard file feed device can be loaded with as many as 3,000 cards. Because of the tremendous reading speed, the ability to keep this file feed loaded increases operating efficiency. The punch feed has a rated speed of 250 cards per minute and a feed hopper capacity of 1200 cards.

The stored program controls the card reading, punching, and stacking operations. Data passes from the card read unit to the input synchronizer and back to the card punch unit from the output synchronizer, allowing the system to continue processing during the actual reading and punching operation.

Radial Stackers

The IBM 1402 Card Read-Punch, Model 2 has 5 radial-type stackers (Figure 6), with a capacity of 1,000 cards each. Cards from each feed can be directed to one of three stackers.

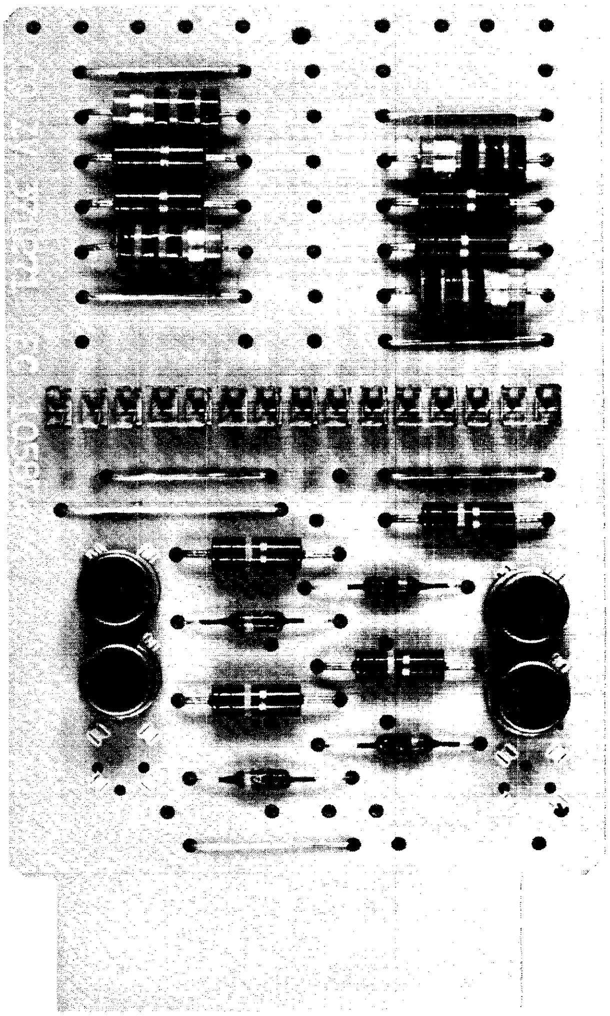


Figure 3. SMS Card, Front Side, Actual Size

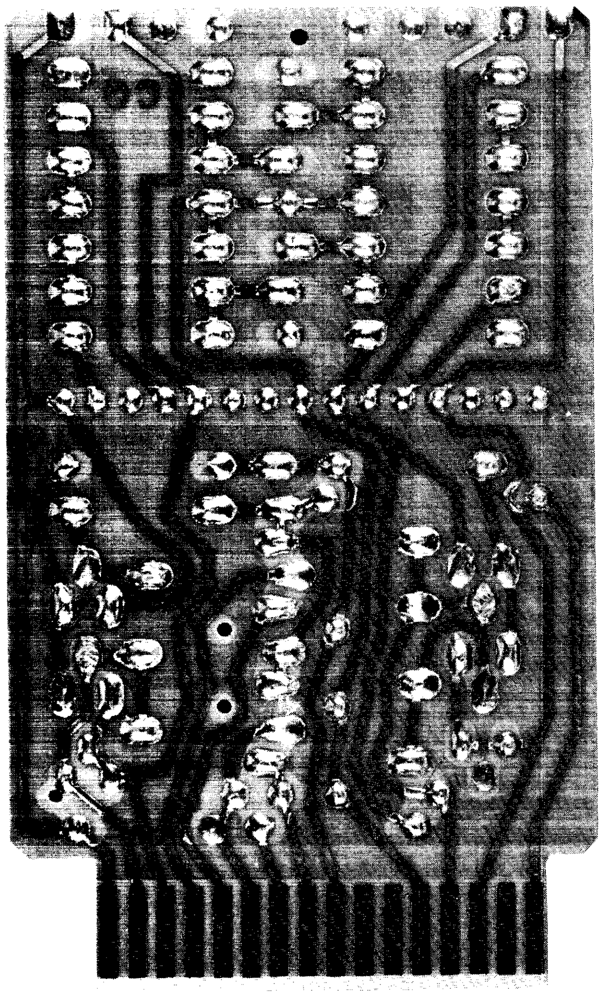


Figure 4. SMS Card, Reverse Side, Actual Size

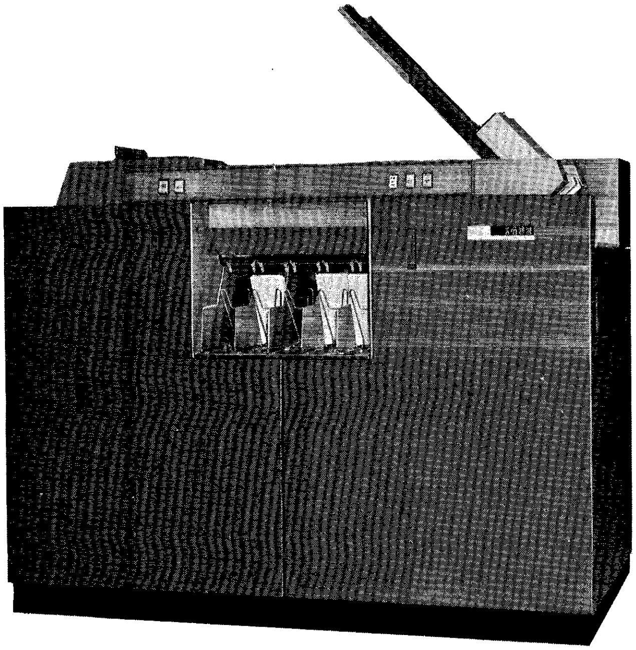


Figure 5. IBM 1402 Card Read-Punch, Model 2

Interchangeable 51-Column Read Feed

The interchangeable 51-column read feed (including file feed) permits feeding either 51-column cards or standard 80-column cards in the read feed of the IBM 1402 Card Read-Punch.

Printed Output

The IBM 1403 Printer (Figure 7) can produce output documents at 600 lines per minute. Data are transferred to a printer synchronizer before printing. A standard dual-speed, tape-controlled carriage permits high-speed skipping at 75 inches per second for skips of more than 8 lines. Because document preparation involves a high degree of form skipping, this feature provides faster document throughput – the true measure of printer efficiency.

Each of the 100 standard print positions can print 48 different characters: 26 alphabetic, 10 numerical; and 12 special characters. It is possible, by special feature, to increase the number of print positions to 132.

If large volumes of numerical printing are encountered, the numerical print feature can be utilized. This feature consists of an interchangeable numerical chain cartridge, and an interchangeable alphamerical chain cartridge. The numerical chain contains 16 characters (digits 0 through 9 $. , * – ⌑). With this feature installed, the IBM 1403 can print as many as 1285 lines per minute, allowing it to print over twice as fast as a 1403 equipped with an alphamerical chain cartridge.

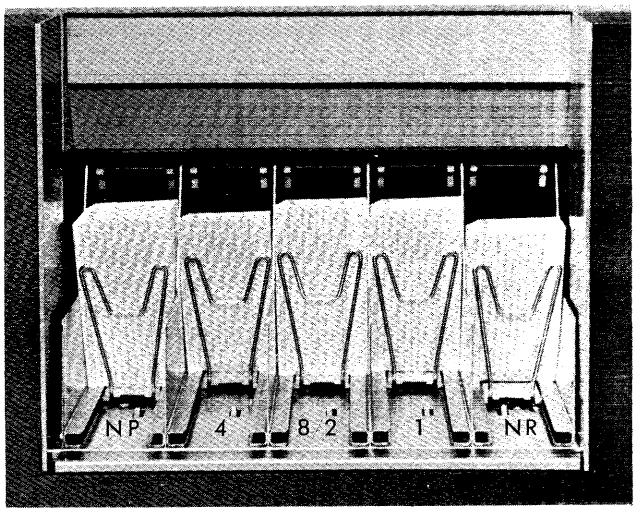


Figure 6. Radial Stackers

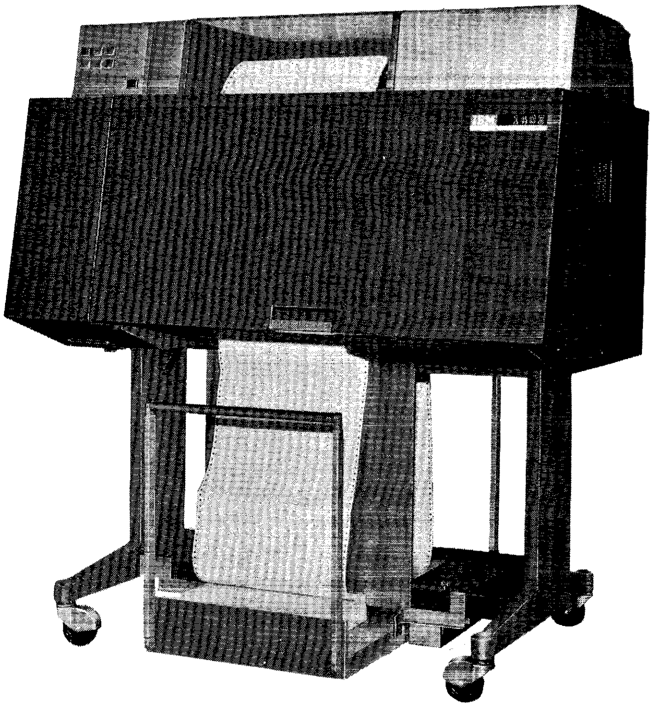


Figure 7. IBM 1403 Printer

Magnetic-Core Storage

The IBM 1410 Data Processing System uses magnetic-core storage for storing instructions and data. All data in core storage are instantly available, and the special design of the core-storage unit makes each position individually *addressable.* Thus, an instruction can designate the exact storage locations that contain the data needed for that step.

The physical make-up of each core-storage location makes it possible for the IBM 1410 to perform arithmetic operations directly in the storage area. (This is called *add-to-storage logic.)*

Magnetic-Tape Storage

The IBM 1410 Data Processing System permits the attachment of IBM 729 Model II or IV tape units (Figure 8), or IBM 7330 tape units (Figure 9). IBM 729 tape units are used when high volume and high speed are the most important considerations in the applications to be processed. The IBM 7330 tape units provide the same accuracy and reliability as the IBM 729 tape units in areas where applications, volumes, and speed justify a lower cost tape system.

The magnetic tape that is used is a plastic material, coated with a metallic oxide. It can be easily magnetized in tiny spots, and patterns of these magnetized spots are codes for digits, alphabetic characters, and special characters.

Data can be read from a variety of sources and written on the tape. Magnetic spots, representing information written on the tape, remain until erased. Thus, besides being used as data storage, the data can be part of the system's input and output.

Magnetic tape is an ideal storage medium for a large volume of data, because a vast amount of information can be kept permanently. The reels of tape can be removed from the system and filed, or they can be shipped and used on other systems.

Data stored on magnetic tape are read sequentially. The data processing system can search the tape to find the data to be used. These tapes, commonly used to set up a library or file of procedures, can store program steps.

Another advantage of magnetic-tape storage is that a reel of tape, produced as an output of a procedure, can be removed from the data processing system. Reports can be written using an independent unit, while the data processing system proceeds with the next programmed operation.

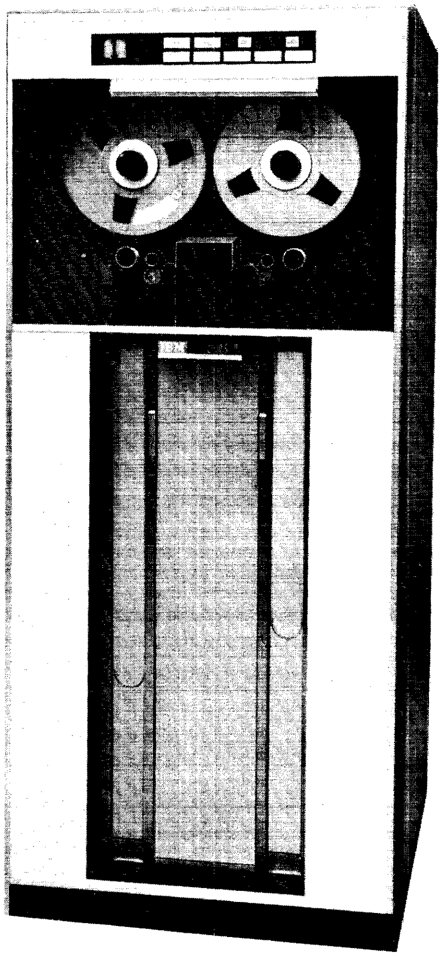


Figure 8. IBM 729 Magnetic Tape Unit, Model II

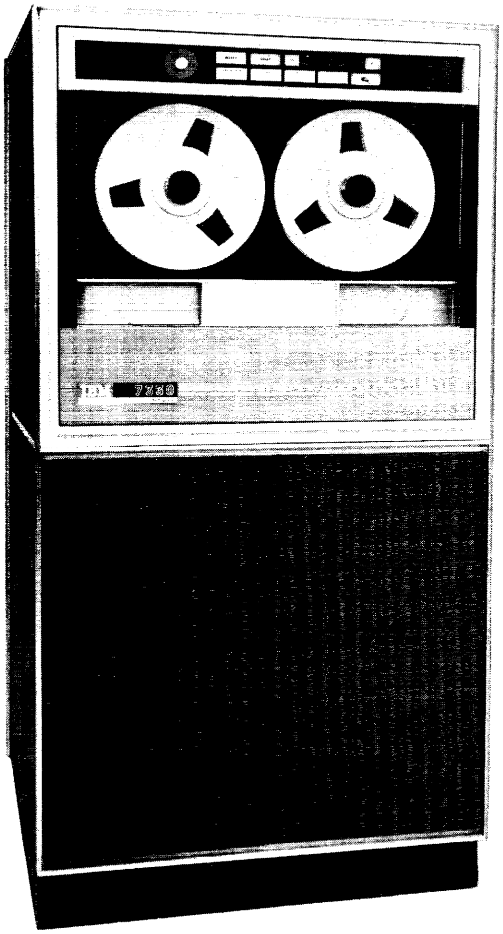


Figure 9. IBM 7330 Magnetic Tape Unit

Magnetic-Disk Storage

As many as 10 million characters of information can be retained by Model 1 of the IBM 1405 Disk Storage Unit. Model 2 (Figure 10) can retain as many as 20 million characters of information.

Data are stored in the form of magnetized spots on the surface of a circular disk that is coated with a magnetic oxide material. Storage disks are mounted on a vertical shaft, which turns at a speed of 1200 revolutions per minute. Data can be recorded on both sides of each disk.

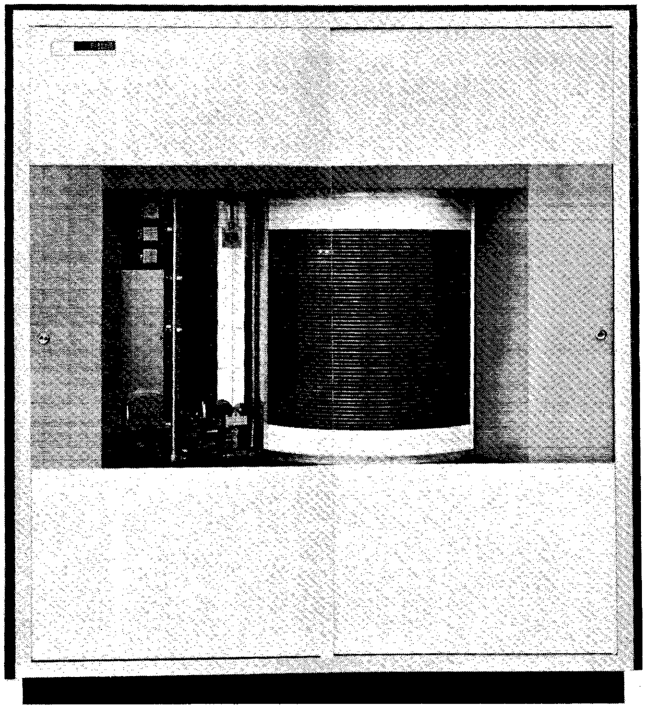


Figure 10. IBM 1405 Disk Storage Unit

Read-write heads, mounted on movable access arms, perform the reading and writing on the disk faces. The arms seek specified records in the disk-storage unit by moving up or down to the specific disk, and across that disk face to the particular record or track.

Once data have been stored in the disk-storage unit, they remain there until a new record is written. Special checking circuits assure accuracy of reading and writing operations.

Records in the disk-storage unit can be updated during processing so that the information contained in disk storage reflects the result of current transactions.

The development of the IBM Management Operating System (MOS) adds significance to the random access feature of the IBM 1410 Data Processing System.

Console

The IBM 1415 Console (Figure 11) provides operator communication with the IBM 1410 Data Processing System. Through the console, the operator can enter information, display storage contents, and read records from the magnetic-tape and/or disk storage units. The console is also designed to simplify program testing. Sufficient desk space is provided for operating ease.

A console I/O printer is used to enter or display storage data. It can also be used as an output device for printing special messages on an exception basis.

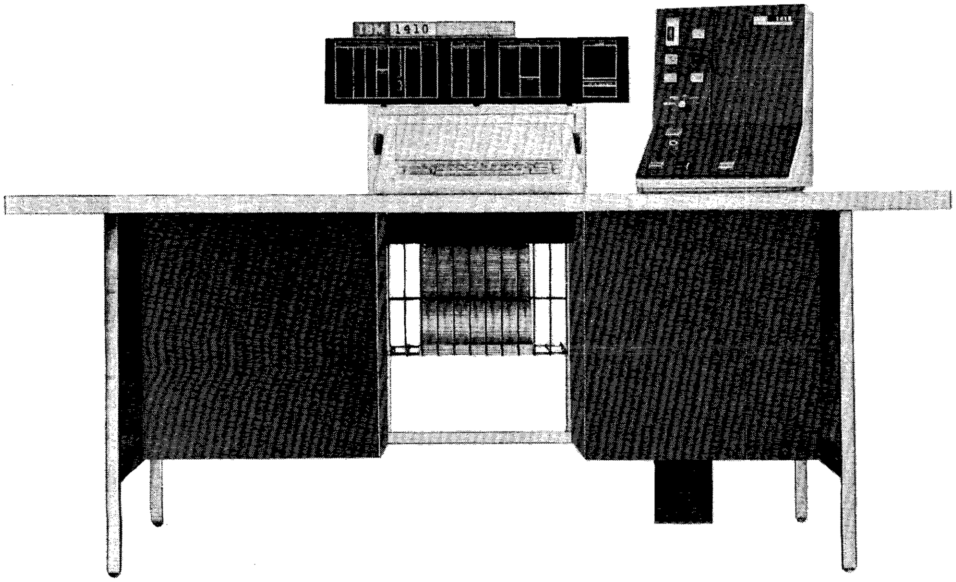


Figure 11. IBM 1415 Console

The console I/O printer can print 64 characters (10 numerical, 26 alphabetic, and 28 special), a word-mark symbol, and an underscore symbol (invalid bit parity print-out).

Input-Output Synchronizers

The IBM 1414 Input-Output Synchronizer (Figure 12) contains the circuitry necessary for transmitting data between the various I/O units and the processing unit.

Model 1 of the 1414 contains the circuitry necessary for controlling up to ten IBM 729 Magnetic Tape Units, Model II or IV, on the same data transmission channel.

Model 2 of the 1414 contains the circuitry necessary for controlling up to ten IBM 7330 Magnetic Tape Units on one data transmission channel.

The 80-character read and punch synchronizers, and a 100- or 132-character printer synchronizer and all the associated controlling circuitry is housed in a 1414, Model 3.

Model 4 of the 1414 is the same as a Model 3, but has the additional facility of being able to control a paper tape input, Teletype, 1009 Data Transmission Unit, and 1014 Remote Inquiry Station. Models 1, 2, and 3 of the 1414 are one-module units. Model 4 of the 1414 is a two-module unit.

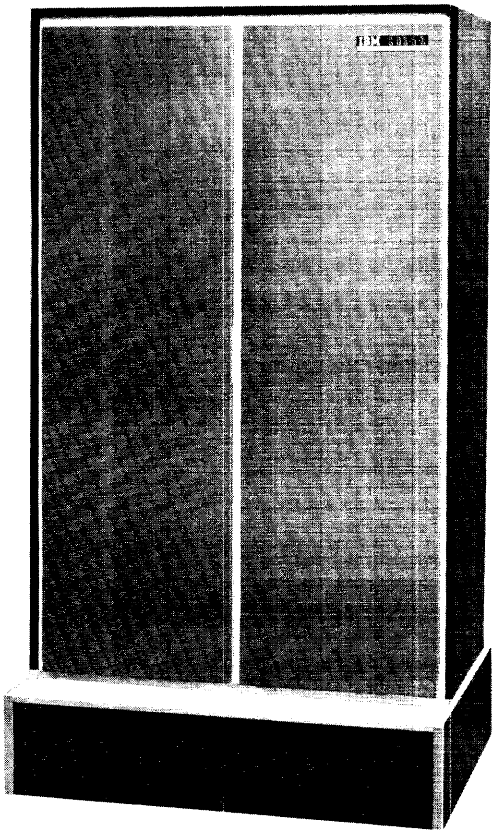


Figure 12. IBM 1414 Input/Output Synchronizer

Processing

Processing is the manipulation of data from the time it is introduced to the system as input until the desired results are ready for output. The IBM 1411 Processing Unit (Figure 13) houses the magnetic-core storage and circuitry that perform these operations.

The storage capacity is either 10,000 (Model 1), 20,000 (Model 2), or 40,000 (Model 3) core-storage positions. All data entering or leaving the 1410 must pass through the 1411 where a series of data checks assure accuracy and reliability.

Processing consists of logic, arithmetic, table lookup, and editing functions.

Logic

The logic of the system is its ability to execute the program in the proper sequence under any circumstances that can occur during the processing. A well-designed program uses this logic by making tests whenever processing conditions (including input data) should change the normal sequence of program step execution. For example, if an input card contains a 2 in column 1, fields A and B should be combined. If the card column contains a 3, fields A and C should be combined. A programmed logical test can be made to determine which digit the card column contains, and then cause the proper program step to be executed.

Arithmetic

The 1410 can add, subtract, multiply, divide, and modify addresses. These are the arithmetic functions performed by the IBM 1411 Processing Unit under program control. They are all standard functions. Modifying an address in the 1410 is an add or subtract function with or without using an index register.

Table Lookup

Table lookup instructions make it possible to search the contents of an entire table for a single desired factor. A table lookup instruction is used whenever factors such as rates, constants, etc., used for processing must be selected under control of a factor in the input information.

The table can contain as many characters as storage capacity of the system permits.

Editing

Programmed editing in the 1410 is a powerful operation that punctuates and inserts special characters in data lines before they are printed. One instruction loads a control constant into the print area, and another edits the raw data automatically as specified by this constant. Floating dollar signs, asterisk protection, decimal control, and special sign control are included to satisfy special editing requirements.

actually needed for an instruction or data, and because the contents of an entire storage area can be moved with a single instruction.

Word Marks

A special eighth plane bit, called a *word mark,* is used to define the length of each instruction and data field in core storage. This word mark makes it possible to employ the variable-word-length concept in the 1410. It can be set and cleared, when necessary, by stored-program instructions. If word-mark identification is needed in magnetic tape records, in disk storage, or in cards, special instructions translate the word marks to word-separator characters during write operations, and translate the word-separator characters to word marks when the data are read back into storage from the tape, disk unit, or card. A special instruction makes it possible to print the digit 1 in the print position that corresponds to a word-mark position in the print area. When data are transferred to the console I/O printer, each character that contains a word mark in storage has the word mark printed above that character. The letter A with an associated word mark is printed $\overset{\vee}{\text{A}}$.

In this manual, each character that has an associated word mark has the inverted circumflex (v) above it.

The word mark serves several functions:

1. It indicates the first character of an instruction.
2. It defines the size of a data word, or
3. It signals the end of an instruction.

The rules governing the use of word marks are:

1. Word marks are assigned specific storage locations in the program planning stage. They are set by a program load routine, or by specific instructions within the program itself.
2. Word marks remain in their original locations unless cleared or set by a positive action such as a CLEAR STORAGE, CLEAR WORD MARK or SET WORD MARK instruction, or by a data move operation that specifies that they should be altered.
3. A word mark must be associated with the first character of each instruction (Op-code position) and are usually associated with the high-order character in a data field.
4. Every instruction in the program must be followed by a word mark. This word mark stops the reading of the instruction.

Addressing

The instructions and data used by the 1410 are kept in core storage. Each core-storage position is addressable and has its own 5-character address. Valid addresses range from 00000 to 39999 (this is for 40K storage; system may have 10K or 20K storage). An address with zone bits in the units, thousands, or the ten-thousands position is invalid. Some characters are not accepted as valid addresses. These characters are: bl, 8-3, 8-4, 8-5, 8-6, 8-7. If an invalid address is given in an instruction, the machine stops and signals an error.

Address Registers

To read out an address from storage, five storage readout cycles are needed. In addition, a device is needed to accept the address characters and keep them until the whole address has been read out. The devices used to do this are the address registers.

I-ADDRESS REGISTER

The instruction address register is a 5-character register. The address read into it specifies the initial address of an instruction in core storage (Op-code position) other than the next instruction in sequence. The number is increased by one each storage cycle as the instruction is being read out.

A-ADDRESS REGISTER

The A-address register is a 5-character register. The register accepts a 5-character address that specifies the core-storage location of the units position of the data field (see note for exception). The number is decreased by one when each position of the data field is acted upon.

B-ADDRESS REGISTER

The B-address register is a 5-character register. Usually, the address read into it specifies the core-storage location of the units position of a data field (see note for exception). In this instance, the number is decreased by one when the field is acted upon.

C-ADDRESS REGISTER

The C-address register is a 5-character register. It is used in store address register, multiply, divide, table lookup, and other operations, but is not accessible to the programmer.

D-ADDRESS REGISTER

The D-address register is a 5-character register. It is used in multiply, divide, recomplement, and other operations, but is not accessible to the programmer.

Note: In most cases, data fields are addressed by specifying the location of the low-order character. This is done so that addition, subtraction, and other operations start with the units position and end with the high-order position. However, data are addressed by specifying the location of the high-order character whenever: (1) Data movements originate or terminate outside the core-storage area; (2) Record moves originate and terminate within the core-storage area.

Single-Character Registers

The A- and B-data registers, Op and Op-modifier registers, I/O channel select register, unit select register, and unit number register are single-character registers used to store data during the execution of various instructions.

B-DATA REGISTER

The B-data register accepts each character as it leaves the 1410 core-storage area. The character is stored in an 8-bit form (BCD code, check bit, and a word-mark bit). This register is reset and filled with a character from core storage during every storage read-out operation. The character can be entered back into storage from the output of the B-data register.

A-DATA REGISTER

The A-data register is reset and filled with the 8-bit character output from the B-data register whenever the operation requires it.

OP-REGISTER

The Op- (operation) register is reset and filled with a 7-bit character (the WM bit is dropped) output from the B-data register whenever the character is an operation code character. The Op-register stores the operation code of the instruction in process for the duration of the operation.

OP-MODIFIER REGISTER

The Op-modifier register is reset and filled with a 7-bit character (the WM bit is dropped) output from the B-data register whenever the character is a d-character of an instruction. The Op-modifier register stores the modifier of the instruction in process for the duration of the operation.

I/O CHANNEL SELECT REGISTER

The I/O channel select register accepts the hundreds position of an x-control field. This position specifies the data transmission channel to be used (1 channel standard – Ch 1; additional channel available – Ch 2) and whether the operation will be performed in an overlapped or non-overlapped mode (refer to *System Features and Considerations* section for more detail on processing overlap).

UNIT SELECT REGISTER (CHANNEL 1 AND 2)

There is a unit select register associated with each data transmission channel. The register accepts the tens position of the x-control field. The tens position specifies the input or output unit being used for the operation.

UNIT NUMBER REGISTER (CHANNEL 1 AND 2)

There is a unit number register associated with each data transmission channel. The register accepts the units position of the x-control field. The units position specifies:

1. the printing of characters or word marks on the IBM 1403 Printer
2. the pocket selection of a card on the IBM 1402 Card Read-Punch
3. the magnetic tape unit that is involved in the operation
4. the operation to be performed by the disk storage unit.

Addressing Example (Figure 15)

Instruction address 02000 contains the operation code for the following instruction that adds A-address field to B-address field.

Op Code	*A-address*	*B-address*
$\overset{\vee}{A}$	05389	05399

The Stored Program

The IBM 1410 is called a stored-program data processing system because it internally stores the instructions used to solve a particular program.

The IBM 1410 has a sequential method of program execution. Thus, instruction 1 is followed by instruction 2 and so on, unless special circumstances arising during processing make it necessary to alter this sequence. Branch instructions make it possible to test for special conditions and change the sequence of program execution, repeat an instruction or group of instructions, or transfer to special subroutines. These tests can be made at any time during processing and control the logic of the program.

Stored-Program Instructions

The 1410 uses stored-program instructions to cause all the input and output devices to operate, and to perform all arithmetic, logical, general data, and miscellaneous operations.

The actual operation performed is indicated by the format and contents of the instruction itself.

The programmer can write instructions in actual machine language, or he can use the IBM 1410 *Autocoder* system. If the programmer chooses to write in actual machine language, he must write the actual

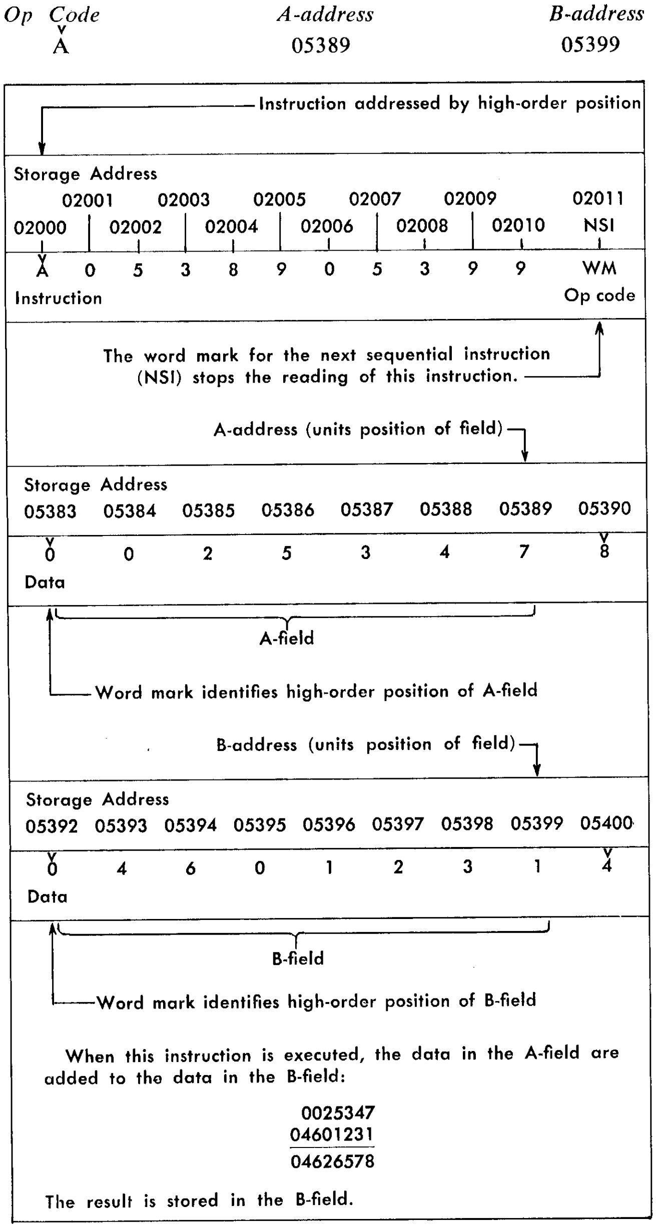


Figure 15. Addressing Example

instructions and storage addresses of all instructions and data used in the program. If he chooses the *Autocoder* system, he uses a set of mnemonic operation codes and assigns labels and symbols for data fields, record areas, and instructions. The *Autocoder* assembly program translates these symbolic entries into actual machine-language instructions.

Once the machine-language program has been produced, it can be kept permanently in punched cards, in the disk-storage unit, or on magnetic tape.

INSTRUCTION FORMAT

The basic instruction format for the IBM 1410 Data Processing System is divided into four parts – the operation code, the A- or I-address (or x-control field), the B-address, and a d-character modifier to the operation code. Because of the variable length instruction format, the total length of an instruction can vary from one to 12 positions

Op Code	*A- or I-address or X-control field*	*B-address*	*d-character*
v x	xxxxx or xxx	xxxxx	x

Op Code. This is always a single character that specifies the basic machine operation to be performed.

A-Address. The A-address is always five characters and specifies a core-storage location.

I-Address. This is always five characters and specifies the address of an instruction in storage other than the next instruction in sequence.

X-Control Field. If the field between the Op code and the B-address is made up of three characters, it is called the x-control field. The x-control field specifies the channel and I/O unit involved in an input-output operation.

B-Address. The B-address is always five characters and specifies a core-storage location.

d-Character. The d-character is a single alphabetic, numerical, or special character used to specify a particular operation within the control of the operation code of the instruction.

Instructions are arranged in sequential locations within core storage. The machine normally executes instructions sequentially from left to right (operation code position through the d-character position) unless this sequential execution is changed by an instruction that causes a transfer (branch) to another storage area for the next instruction.

WORD MARKS

Each instruction must have a word mark set over the operation code, and must not contain word marks in any other position. Also, a word mark must be set in the core-storage location immediately to the right of the last character of an instruction. This is normally the word mark associated with the operation code of the next sequential instruction.

INSTRUCTION LENGTH VALIDITY

Instruction length checking is incorporated in the system to insure that each instruction read contains a valid number of characters for the operation code specified.

Valid instruction words vary in length from one to twelve characters depending on the amount of information required for the operation. The general instruction format consists of a single character operation code followed by one or two 5-character addresses, or a 3-character input-output operation specification (called the x-control field) and, in some cases, a single character operation modifier. Valid instruction word lengths are:

O
Od
Oxxxd
OAAAAA
OAAAAAd
Oxxx BBBBBd
OAAAAA BBBBB
OAAAAA BBBBBd

The O specifies the single-character operation code. The five A's specify the 5-character address of the A-field, and five B's specify the 5-character address of the B-field. The three x's specify the x-control field and the d specifies the operation modifier character.

INSTRUCTION DESCRIPTIONS

Instructions are described in this manual by using a standard format. Instructions that perform similar functions are incorporated in one general description. In these cases, individual instructions and details are shown in chart form.

Descriptions follow this format:

Instruction Format. This contains the mnemonic operation code, the actual operation code, the A-address or I-address or x-control field, the B-address and the d-modification character.

Function. This is the description of the operation performed.

Word Marks. This is the information the programmer must have to determine the effect of word marks on the operation. All instructions must have a word mark associated with the operation code.

Timing. This is the formula used in calculating the time required for executing the instruction (in microseconds). Refer to Figure 16 for symbols that are used in the formulas.

Because there are several valid lengths for some instructions, these lengths, as they apply to the timing formula, are included in the section. Some operations also have extra steps, depending on the data, etc. When this condition is present, the numbers assigned to these steps (D, E, R, etc.) are also given.

Notes. These are special notations or additional information that pertain to the operation.

Address Registers after Operation. The contents of the address registers are represented by the codes described in the *Chaining Instructions* section.

A — is the A-field length.
B — is the B-field length.
C — is 1 if the branch is taken. It is zero otherwise.
D — is the number of characters in the B-field from the start of zero suppression to the place where the dollar sign is inserted. If no dollar sign is inserted, D is zero.
E — is 2 on a single-character multiply or divide operation. It is 1 on a single-character add, subtract, reset add, reset subtract, table search, or a 6-character multiply or divide operation. It is zero otherwise.
I/O is the time used by input/output device to accept or send data and the Synchronizer access time, when applicable.
L — is the instruction length.
M — is the multiplier length.
N — is the number of fields actually compared on a table search. The B-field length on a table search includes only those argument fields actually compared and the intervening function values.
Q — is the quotient length.
R — is 1 if a recomplement is taken on an add or subtract operation. It is zero otherwise.
Z — is the number of characters in the B-field from the start of zero suppression (as indicated in the control field) to the left end of the B-field.

Figure 16. Timing Formula Symbols

CHAINING INSTRUCTIONS

If the A- and B-address registers contain addresses of the next fields to be processed during the execution of a program, another complete instruction is not necessary. The operation code, alone, can be given and the contents of the address registers are used to specify the A- and B-fields. Connecting instructions together in this manner is called *chaining*. This method not only conserves storage space, but also saves time, because the address registers do not have to be reloaded during the reading of the instruction. For example, an ADD instruction, $\overset{\vee}{\text{A}}$ 05985 06985, is executed. Field A has five characters and field B has six characters. The address registers after the operation contain:

A-Add. Reg.	*B-Add. Reg.*
05980	06979

The A- and B-addresses of the fields to be used in the next operation are 05980 and 06979 respectively. The next instruction need contain only the operation code, because the A- and B-address registers are already at the desired locations. The instruction $\overset{\vee}{\text{S}}$ causes the data at location 05980 to be subtracted from the data at 06979. This chaining technique can be used to link several instructions together. The only restriction is that the fields remain in sequence and that the address registers contain valid addresses.

The descriptions of the instructions include the contents of the address registers after the operation has been performed. The programmer can use this information to determine which instructions can be chained in particular situations. Figure 17 shows the symbols that represent the contents of the address registers.

Abbreviation	*Meaning*
A	A-address of the instruction
B	B-address of the instruction
NSI	Address of the next sequential instruction
BI	Address of the next instruction if a branch is taken
LA	The number of characters in the A-field
LB	The number of characters in the B-field
LW	The number of characters in the A- or B-field, whichever is shorter
Ap	The previous contents of the A-address register
Bp	The previous contents of the B-address register

Figure 17. Address Register Symbol Chart

Special Considerations.

1. Only 1410 storage addresses can be chained.
2. All no-address instructions (Op code only) use the contents of the A- and B-address registers after the previous operation.
3. Branch instructions cause the B-address register to be loaded with the address of the next instruction in normal sequence (NSI), if a branch occurs. The I-address register always contains the address of the next sequential instruction. However, machine circuits set a special latch. The latch causes the address of the next program step to be taken from the A-address register after a branch is taken. If a branch does not take place, the B-address register contains its contents as a result of the previous operation.

Note: BRANCH ON WORD MARK OR ZONE EQUAL, BRANCH IF BIT EQUAL, and BRANCH IF CHARACTER EQUAL instructions have B-addresses. In this case, the B-address register contains the B-address minus one.

The A-address register contains the I-address of the instruction just executed.

Arithmetic Operations

The add, subtract, zero and add, zero and subtract, multiply, and divide operation codes are used to perform the system's arithmetic operations. The use of add-to-storage logic in the IBM 1410 eliminates the need for special-purpose accumulators or counters in the system. Because any group of storage positions can be used as an accumulating field, the capacity for arithmetic functions is not limited by a predetermined number of counter positions.

All arithmetic functions are performed under complete algebraic sign control. The sign of a factor is determined by the combination of zone bits in the units position of the fields specified by the instruction being executed.

Figure 18 shows the four possible combinations of zone bits and the values of the signs they represent.

The standard machine method of signing a field is to indicate a positive factor with A- and B-bits (12 zone) or No A- or B-bits (No Zone), and to indicate a negative factor with a B-bit (11 zone).

The arithmetic operations in the IBM 1410 Data Processing System are performed by using one of two types of add cycles incorporated in the system. The two types of add cycles are:

1. true add
2. complement add

The type of add cycle performed depends on the arithmetic operation and the signs and values of the two factors involved (Figure 19).

SIGN	BCD CODE BIT CONFIGURATION	CARD CODE CONFIGURATION
Plus	No A- or B-Bit	No Zone
Plus	A- and B-Bits	12 Zone
Minus	B-Bit Only	11 Zone
Plus	A-Bit Only	0 Zone

Figure 18. Sign Bit Equivalents

True Add

A true add cycle is specified when the total number of minus signs is an even number (0 or 2). The signs considered are the signs of the factors and the sign of the operation.

TYPE OF OPER.	A-FLD. SIGN	B-FLD. SIGN	TYPE OF ADD CYCLE	SIGN OF RESULT
ADD +	+	+	True Add	+
		−	Compl. Add	Sign of Greater Value
	−	+	Compl. Add	
		−	True Add	−
SUBTRACT −	+	−	True Add	−
		+	Compl. Add	Sign of Greater Value
	−	−	Compl. Add	
		+	True Add	+

Figure 19. Types of Add Cycles and Sign of Result for Add and Subtract Operations

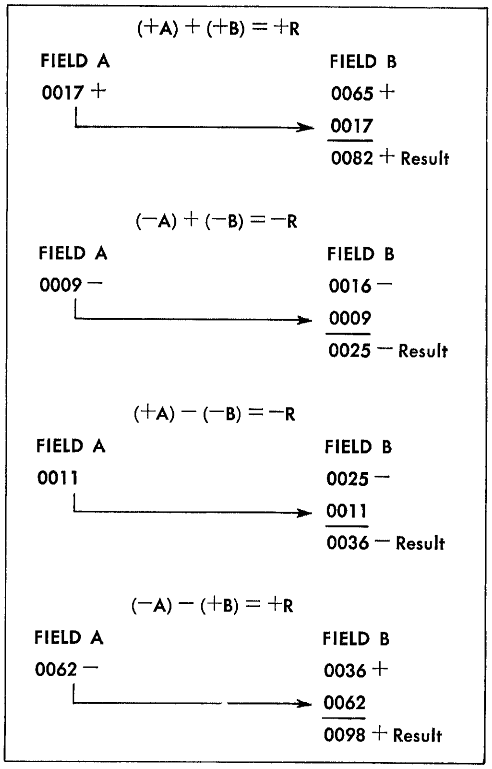


Figure 20. True-Add Cycles Examples

The sign of the result after a true-add cycle carries the original sign of the B-field when either an add or a subtract operation is performed (Figure 20). The sign of the result on a zero and add or zero and subtract operation is covered in the individual description.

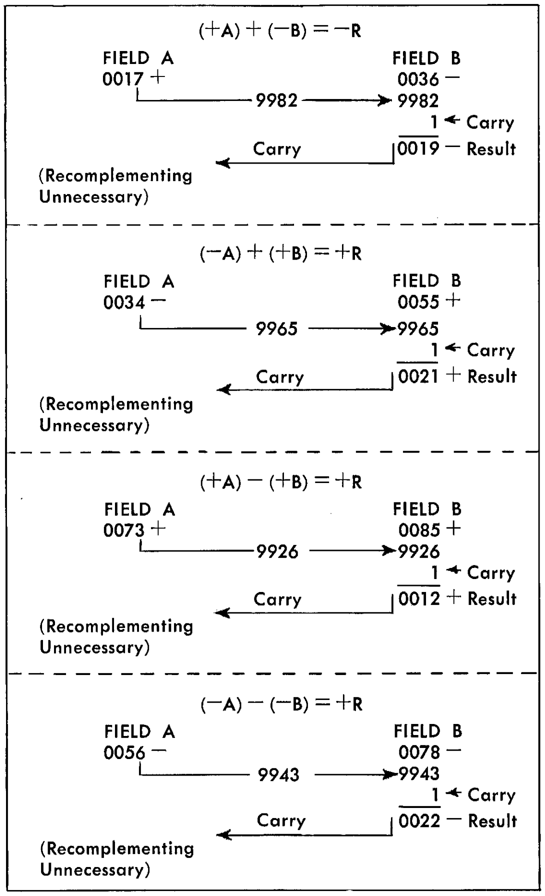


Figure 21. Complement-Add Cycle Examples

Complement Add

An uneven number of minus signs (1 or 3) specifies a complement add cycle. The system converts the A-field factor to its nines complement figure and adds it to the B-field factor (plus one initial carry). The system then initiates a carry test to determine whether a carry occurred from the high-order position of the B-field. The presence of a carry indicates that the result in the B-field is a true figure (Figure 21). The original sign of the B-field is the sign of the result. The sign of the result on a zero and add or zero and subtract operation is covered in the individual description.

If there was no carry from the high-order position of the B-field, the result in the B-field is not a true figure. A recomplement cycle is performed to convert the result to a true figure. In an add operation that results in a negative figure, the sign of the result is always changed during a recomplement cycle (Figure 22). The system generates the new sign automatically. A positive factor is indicated by the presence of an A- and B-bit over the units position of the factor. After a complement add cycle, the sign of the result carries the sign of the greater value factor.

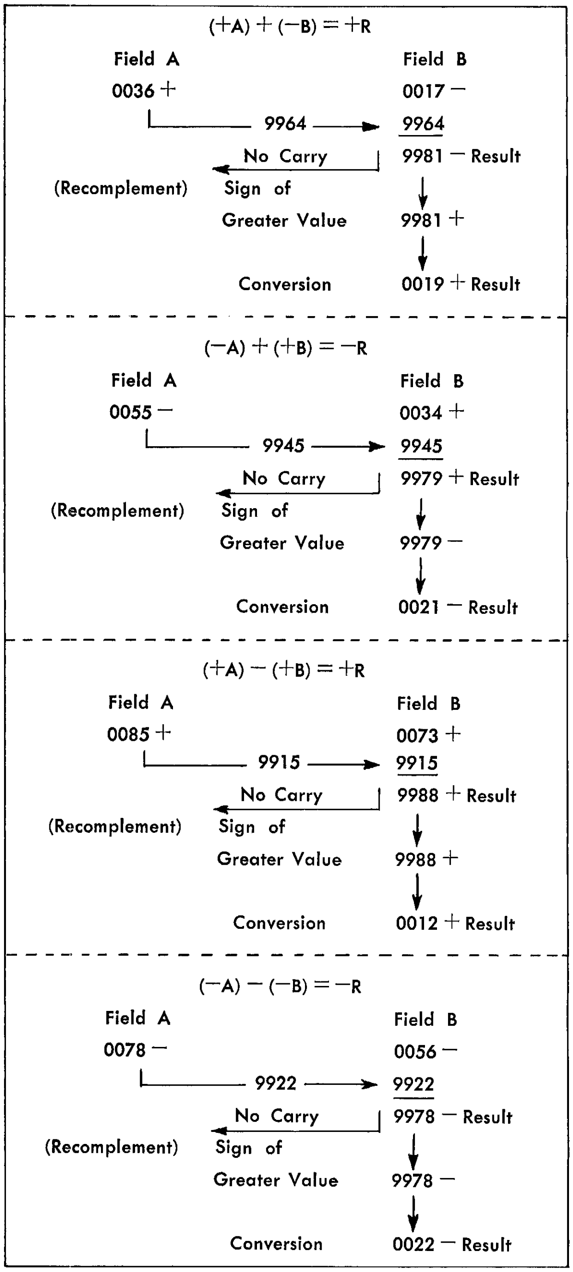


Figure 22. Complement-Add (with Recomplementing) Cycle Examples

Digit Coding

In all arithmetic operations, the presence of the codes: blank, 8-3, 8-4, 8-5, 8-6, and 8-7 in the numerical portion of a field are treated as zero, 3, 4, 5, 6, and 7 respectively.

Overflow

If the result exceeds the limit of the B-field (determined by the B-field word mark), the carry is lost and the arithmetic overflow indicator turns on. A TEST AND BRANCH instruction, $\overset{\vee}{J}$(I)Z, tests and turns OFF this indicator.

Zero Balance

If the result of any add, subtract, multiply, zero and add, or zero and subtract operation is a zero balance, the zero balance indicator is turned on. This indicator can be tested by a TEST AND BRANCH instruction — $\overset{\vee}{\text{J}}$(I)V. The indicator is turned off by the next ADD, SUBTRACT, MULTIPLY, ZERO AND ADD, or ZERO AND SUBTRACT instruction that does not result in a zero balance.

Arithmetic Operation Codes

Add (Two Fields)

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
A	$\overset{\vee}{\text{A}}$	xxxxx	xxxxx

Function. This instruction causes the numerical data in the A-field to be added algebraically to the numerical data in the B-field. The result is stored in the B-field. Zone bits remain undisturbed in the B-field (except for the sign position, which may be required to change). A-field zone bits (except for the sign position) are ignored. The resultant B-field is in true form at the end of the operation.

Word Marks. The B-field word mark stops the operation and must be set to define the high-order position. If the A-field is shorter than the B-field, it must also have a defining word mark to stop transmission of data from the A-field to the B-field. In this case, the machine automatically adds zeros to the extra high-order positions of the B-field until it detects the B-field word mark.

If the A-field is longer than the B-field, the high-order positions of the A-field, that exceed the limits imposed by the B-field word mark, are not processed.

Timing. $T = 4.5\,(L + 1 + E + A + 1.5B + 1.5RB)$.

When $L = 1$, $E = 1$; when $L = 11$, $E = 0$.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LW	B-LB

Add (One Field)

Instruction Format.

Mnemonic	*Operation Code*	*A-address*
A	$\overset{\vee}{\text{A}}$	xxxxx

Function. This format of the ADD instruction is used to double the A-field. The A-field is added to itself and the result is stored in the A-field. This operation is always performed with a true-add cycle and the sign-bit configuration of the result is always the same as the original sign of the A-field.

Word Marks. The A-field must have a word mark associated with its high-order position.

Timing. $T = 4.5\,(L + 1 + A + 1.5A)$.

$L = 1$ or 6.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	A-LA

Subtract (Two Fields)

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
S	$\overset{\vee}{\text{S}}$	xxxxx	xxxxx

Function. The numerical data in the A-field is *subtracted algebraically* from the data in the B-field. The result is stored in the B-field. Zone bits remain undisturbed in the B-field except for the sign position that may be changed. A-field zone bits are ignored in all positions except the sign position. The B-field result is in true form.

Word Marks. The B-field word mark stops the operation and must be set over the high-order position of that field. If the A-field is shorter than the B-field, it, too, must have a defining word mark to stop transmission of data from A to B. When the A-field is shorter than the B-field, the machine *subtracts* zeros from the extra high-order positions of the B-field up to and including the word-mark position.

If the A-field is longer than the B-field, the high-order positions of the A-field that exceed the limits imposed by the B-field word mark are not processed.

Timing. $T = 4.5\,(L + 1 + E + A + 1.5B + 1.5RB)$.

When $L = 1$, $E = 1$; when $L = 11$, $E = 0$.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LW	B-LB

Subtract (One Field)

Instruction Format.

Mnemonic	*Op Code*	*A-address*
S	$\overset{\vee}{\text{S}}$	xxxxx

Function. The A-field is subtracted from itself, and the result is stored in the A-field. The numerical portion of the A-field is always zero after the operation, but zones in the A-field are unchanged and the A-field sign bit configuration is the same as it was before the operation.

Word Marks. A word mark must be associated with the high-order position of the A-field.

Timing. T = 4.5 (L + 1 + A + 1.5A).
L = 1 or 6.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	A-LA

Zero and Add (Two Fields)

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
ZA	$\overset{\vee}{?}$	xxxxx	xxxxx

Function. This instruction causes the numerical data in the A-field to be stored in the B-field. The sign of the result field (B-field) is the same as the sign of the A-field. The final B-field sign will be in standard form. All other zone positions in the B-field are set to No Zone (No A- and No B-bits).

Word Marks. The B-field must have a defining word mark to stop the operation. The A-field requires a word mark only if it is shorter than the B-field. If the A-field is shorter than the B-field, extra high-order B-field positions are set to zero. If the A-field is longer than the B-field, the high-order positions of the A-field, that exceed the limits imposed by the B-field word mark, are not processed.

Timing. T = 4.5 (L + 1 + E + A + 1.5B).
When L = 1, E = 1; when L = 11, E = 0.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LW	B-LB

Zero and Add (One Field)

Instruction Format.

Mnemonic	*Op Code*	*A-address*
ZA	$\overset{\vee}{?}$	xxxxx

Function. This format of the ZERO AND ADD instruction causes the A-field to be added to itself. This instruction is used to strip the A-field of all zones, except in the units (sign) position, and to change non-numerical codes (blanks, 8-3, 8-4, 8-5, 8-6, and 8-7 codes) to their numerical equivalents (zero, 3, 4, 5, 6, and 7 respectively). The sign of the A-field is retained. However, the bit configuration of the plus sign may change. If the A-field plus sign bit configuration is not an A- and B-bit, it is changed to the A- and B-bit configuration.

Word Marks. The A-field must have a word mark set over its high-order position.

Timing. T = 4.5 (L + 1 + A + 1.5A).
L = 1 or 6.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	A-LA

Zero and Subtract (Two Fields)

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
ZS	$\overset{\vee}{!}$	xxxxx	xxxxx

Function. This operation causes the numerical data in the A-field to be moved and stored in the B-field with the opposite sign as shown in Figure 23. The standard machine method of signing a field is used.

All other zone positions in the B-field are set to No Zone (no A- and no B-bits). If the A-field is shorter than the B-field, extra high-order B-field positions are set to zero.

Word Marks. The B-field must have a defining word mark to stop the operation. The A-field requires a word mark only if it is shorter than the B-field. In this instance, the system inserts zeros in the extra high-order positions of the B-field up to, and including, the word-mark position.

If the A-field is longer than the B-field, the high-order positions of the A-field that exceed the limits imposed by the B-field are not processed.

Timing. T = 4.5 (L + 1 + E + A + 1.5B).
When L = 1, E = 1; when L = 11, E = 0.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LW	B-LB

A-FIELD SIGN	B-FIELD SIGN AT END OF OPERATION
No A- and No B-bits (plus)	B-bit (minus)
B-bit (minus)	A- and B-bits (plus)
A- and B-bits (plus)	B-bit (minus)
A-bit (plus)	B-bit (minus)

Figure 23. Sign Changes for Zero and Subtract Operation

Zero and Subtract (One Field)

Instruction Format.

Mnemonic	*Op Code*	*A-address*
ZS	$\overset{v}{!}$	xxxxx

Function. This format of the ZERO AND SUBTRACT instruction causes no change to the numerical data in the A-field. This instruction is used to strip the A-field of all zones, except in the units (sign) position, and to change the A-field sign. If the A-field was positive before the operation, it is negative after the operation, and vice versa. The standard machine method of signing a field is used. Non-numerical codes are replaced by their corresponding numerical equivalents. See ZERO AND ADD.

Word Marks. The A-field requires a word mark in its high-order position.

Timing. $T = 4.5 (L + 1 + A + 1.5A)$.
$L = 1$ or 6.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	A-LA

Multiply

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
M	$\overset{v}{@}$	xxxxx	xxxxx

Function. The multiply instruction causes the numerical data in the A-field (multiplicand) to be repetitively added and stored in the B-field (product), starting with the low-order positions. The multiplier is initially located in the B-field, but in the high-order positions.

The conditions that must be considered before and during a multiply operation are:

1. *B-field length.* Because the product is developed in the B-field, the field must be big enough to accommodate the repetitive additions of the A-field and still not interfere with the multiplier position. Therefore, the length of the B-field is determined by adding *1* to the sum of the number of digits in the multiplicand and the multiplier.

Example:

$\overset{v}{1}432$ 4-digit multiplicand
× 302 3-digit multiplier
+1
8 positions must be allowed
in the B-field — $\overset{v}{3}02xxxx\overset{\pm}{x}$

2. *Zone Bits and Sign.* Zone bits that appear in any position of the multiplicand (A-field) are undisturbed by the multiply operation. Zone bits that appear in the assigned product area are eliminated before product development starts. Zone bits that appear in the multiplier (high-order position of the B-field) are eliminated during product development. However, before zone-bit elimination starts, both the units position of the multiplicand and the multiplier are checked for zone bits (the sign of the factor). The presence or absence of zone bits is used to determine the sign of the product. Like signs in the units position of the multiplicand and multiplier result in a plus sign. Unlike signs result in a minus sign. At the end of the operation, the sign of the product is placed in the units (sign) position of the B-field.

3. *Multiplier Factor.* As the product is developed, the multiplier is eliminated, digit by digit. If it is required for later use in the program, it must be retained in another storage area.

Word Marks. Word marks must be set to identify the high-order positions of both the multiplier and multiplicand fields.

Example. Multiplication in the example shown (Figure 24) is a series of repetitive true or complement additions. The number and type of additions is determined by the multiplier digit. A multiplier digit from one to four causes the A-field (multiplicand) to be true-added that number of times. The multiplier

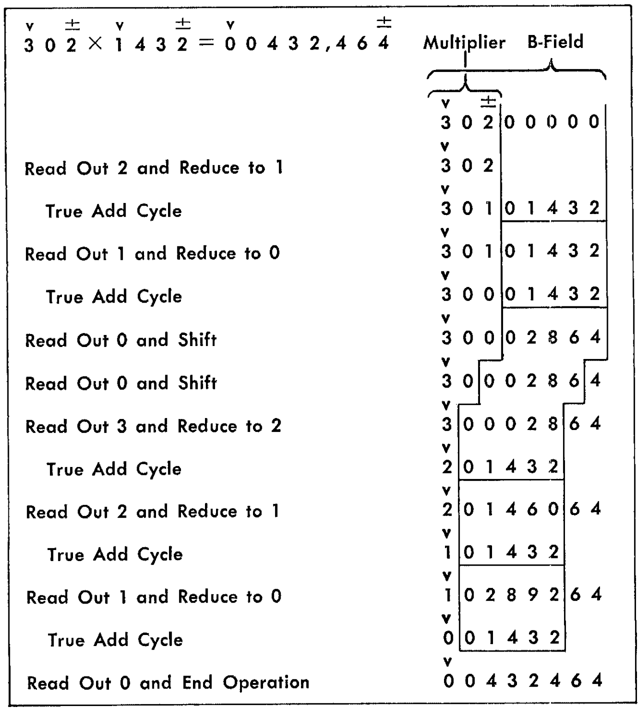


Figure 24. Multiply Example

digit is reduced by one each true-add cycle until zero is reached. The recognition of the zero ends the true-add cycles, and shifts the portion of the product field, being developed, one position to the left.

A multiplier digit of five or more causes the A-field to be complement-added. The number of complement-add cycles depends on the multiplier digit. If the multiplier digit is 8, two complement-add cycles occur (the tens complement of 8 is 2).

Timing. $T \approx 4.5\ [L + 1 + E + 2.5M + (2.5M + 1.5)(2.5A + 3)]$

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	B-LB

Divide

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
D	$\overset{v}{\%}$	xxxxx	xxxxx

Function. During a divide operation, the dividend located in the B-field is divided by the divisor located in the A-field. The result (quotient) is stored in the high-order positions of the B-field. The dividend is destroyed during the operation, except for any remainder, which remains in the low-order positions of the field.

The conditions that must be considered before and during a divide operation are:

1. *Addressing of factors.* The A-address specifies the units position of the divisor. The B-address specifies the high-order position of the dividend. The dividend itself is located in the low-order positions of the quotient-dividend (B) field.

2. *B-field length.* Because the quotient is developed in the B-field, the field must be big enough to accommodate the repetitive complement-additions of the A-field, and still not interfere with the quotient position being developed. Therefore, the length of the B-field is determined by adding *1* to the sum of the number of digits in the divisor and dividend fields.

Example:

v
12 ⌐147 3-digit dividend
← 2-digit divisor
+1
—
6 positions must be allowed
v ±
in the B-field — 000DDD

3. *Sign.* The divisor located in the A-field can be signed or unsigned. If there are no bits in the units position of the divisor factor, the system assumes that the divisor factor is positive.

The dividend factor located in the B-field must have a sign in the units position of the factor. The sign bit configuration must be an A- and B-bit for plus, or a B-bit for minus. If the dividend is moved into the B-field by a ZERO AND ADD instruction it insures the proper signing of the B-field and the presence of zeros in the high-order position (quotient positions) of the B-field.

4. *Quotient.* The quotient field should contain zeros when the divide operation begins. These zeros can be automatically inserted by moving the dividend into the B-field with a ZERO AND ADD instruction (condition 3).

Word Marks. A word mark must be set to define the high-order position of the divisor (A-field). If a ZERO AND ADD instruction was used to move the dividend into the B-field, the high-order position of the field contains a word mark. This B-field word mark, however, is not needed during the divide operation; it is ignored, but retained.

Example. Division in the example shown (Figure 25) is a series of complement additions, and, in the case of an overdraw, a true addition. Complement-add cycles take place until signalled by a no carry condition. The divisor factor is true-added, and then the shift to the next position takes place. The carry

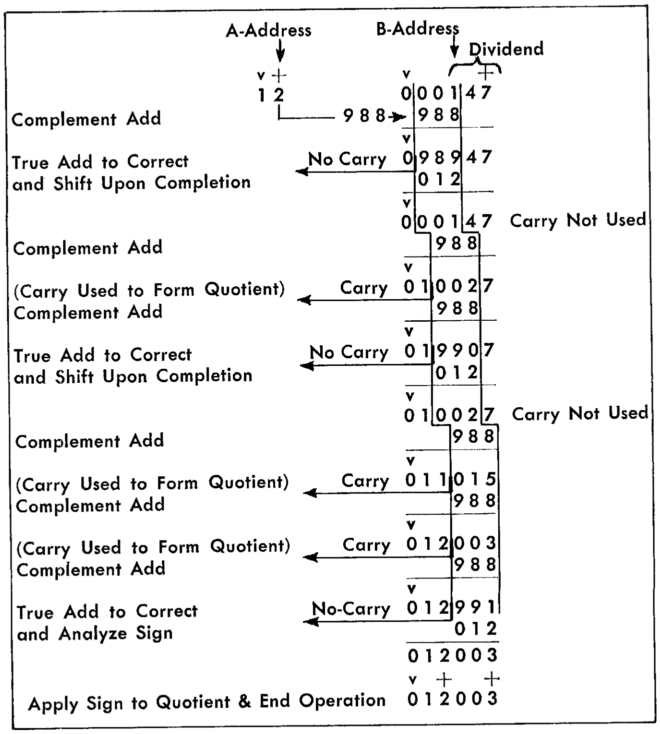


Figure 25. Division Example

resulting from a successful complement-add cycle is used to develop the quotient. This type of operation continues until the true-add cycle that corrects the units position of the dividend occurs. During this cycle, the signs of the dividend and the divisor are analyzed. The resultant sign is then applied to the units position of the quotient and the operation is ended.

At the completion of the operation:

1. The quotient is in the high-order positions of the B-field. The location of the units position of the quotient is the address of the units position of the dividend, minus the length of the divisor, minus one.
2. The remainder is located in the low-order positions of the B-field.
3. The sign of the quotient follows algebraic sign rules (Figure 26), and appears over the units position of the quotient at the end of the operation.
4. The sign of the remainder is the sign of the original dividend (Figure 26).

Divisor Sign	+	+	−	−
Dividend Sign	+	−	+	−
Remainder Sign	+	−	+	−
Quotient Sign	+	−	−	+

Figure 26. Divide Sign Control

Notes.

1. Because only one quotient digit can be developed at a time, it is important to address the high-order position of the dividend (B-address of the DIVIDE instruction). This insures that the first divide operation will result in a single high-order quotient digit. An improperly addressed dividend can cause a divide overflow condition if the result of the first divide operation is greater than 9.

2. If the quotient field is not large enough, a resulting overflow may or may not be indicated. If the field is one position too small, there will not be an overflow indication, even though the units position of the adjacent field is changed. If the field is two or more positions short, the divide operation will result in a divide overflow condition. Too small a quotient field is a programming error, and is not checked by the system. The divide overflow condition, however, can be tested by a TEST AND BRANCH instruction — $\overset{\vee}{\text{J}}$(I)W.

3. Division by zero always results in a divide overflow indication.

4. If a larger quotient is required, extra zeros can be added to the dividend before the divide operation starts. For each additional quotient digit desired, insert one zero to the right of the dividend as shown in Figure 27.

Timing. $T \approx 4.5\{L + 1 + E + 6.5Q\ [A + 1.5\ (A + 2)]\}$

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	tens position of the quotient field.

```
Regular
    147 ÷ 12 = 12 + 3 Remainder          Factors Before Division
           1 2                          Divisor   Quot/Dividend Fld
    1 2 ) 1 4 7                          v ±       v          ±
          1 2                            1 2       0 0 0 1 4 7
          ---
            2 7                           Factors After Division
            2 4                          v ±       v     ±    ±
            ---
              3  Remainder               1 2       0 1 2 0 0 3


Additional Quotient Digits Required
    147.00 ÷ 12 = 12.25                   Factors Before Division
           1 2.2 5                      Divisor   Quot/Dividend Fld
    1 2 ) 1 4 7.0 0                      v ±       v              ±
          1 2                            1 2       0 0 0 1 4 7 0 0
          ---
            2 7
            2 4                           Factors After Division
            ---
              3 0                        v ±       v       ±      ±
              2 4                        1 2       0 1 2 2 5 0 0 0
              ---
                6 0
                6 0
                ---
                  0
```

Figure 27. Additional Quotient Digits

Indexing

The IBM 1410 Data Processing System is equipped with fifteen indexing registers that can be used to modify the A-, I-, or B-address of most instructions. Each index register is assigned five storage locations where the index factor is kept (Figure 28).

Addresses that cannot be indexed:

1. X-control field of M, L, or U instruction
2. G instruction.

To modify addresses, the index register containing the index factor must be selected. To do this, the A-address, the B-address, or both addresses must be *tagged.* A tag is a zone bit over the hundreds position, the tens position, or both the hundreds and tens positions of the address to be modified (Figure 29).

The index factor of the index register selected is added algebraically to the address of the instruction after the address has entered the address register from storage and before the instruction is executed. The address is modified in the appropriate address register. Thus, the actual instruction in storage itself is not changed; but, because data are read from storage under control of the address registers, the *effect* of the original instruction is changed. The index factor also remains unchanged as a result of indexing.

During an indexing operation, the tagged address is considered to be positive regardless of the bit configuration of the sign position; however, the sign position of the index factor is considered. If it has a plus sign, it is added to the tagged address. If it has a minus sign, it is subtracted from the tagged address. The arithmetic overflow latch is not set as a result of overflows incurred during indexing. The result of this modification *must be a valid storage address,* or the machine will stop on an error when the instruction is executed. The validity of addresses must be considered when altering or interchanging programs between 10K, 20K, and 40K systems.

Storage positions 00025 to 00099 can be used for general storage if they are not required for indexing purposes. Word marks can be set in this area at any time because they do not affect the indexing operation. Zone bits are undisturbed in the index registers and have no effect on indexing except when they appear in the sign position of an index factor.

EXAMPLES

1. Modify the A-address of this instruction:

Op Code	*A-address*	*B-address*
$\overset{\text{v}}{\text{A}}$	009Z6	00961

The A-address is tagged by an A-bit over the tens position (Z = A81). Index register 1 is selected

INDEX REGISTER	INDEX FACTOR STORAGE LOCATIONS
1	00025 to 00029
2	00030 to 00034
3	00035 to 00039
4	00040 to 00044
5	00045 to 00049
6	00050 to 00054
7	00055 to 00059
8	00060 to 00064
9	00065 to 00069
10	00070 to 00074
11	00075 to 00079
12	00080 to 00084
13	00085 to 00089
14	00090 to 00094
15	00095 to 00099

Figure 28. Index Register Location

B-BIT OVER HUNDREDS POSITION	A-BIT OVER HUNDREDS POSITION	B-BIT OVER TENS POSITION	A-BIT OVER TENS POSITION	TAG INDEX REGISTER
				NONE
			A	1
		B		2
		B	A	3
	A			4
	A		A	5
	A	B		6
	A	B	A	7
B				8
B			A	9
B		B		10
B		B	A	11
B	A			12
B	A		A	13
B	A	B		14
B	A	B	A	15

Figure 29. Zone-Bits Used to Tag Index Registers

(Figure 21). Because the index register 1 factor is minus it is subtracted from the A-address:

A-address = 009Z6 = 00996
Index register 1 factor = 0001J = −00011

Effective A-address = 00985

	Op Code	*A-address*	*B-address*
Effective instruction:	$\overset{\vee}{\text{A}}$	00985	00961

Valid address on 10K, 20K, and 40K systems.

2. Modify the B-address of this instruction:

Op Code	*A-address*	*B-address*
$\overset{\vee}{\text{A}}$	00459	01MT1

The B-address is tagged by a B-bit over the hundreds position and an A-bit over the tens position (M = B4 and T = A21). Index register 9 is selected. Because the index register 9 factor is plus, it is added to the B-address:

B-address = 01MT1 = 01431
Index register 9 factor = 0010C = +00103

Effective B-address = 01534

	Op Code	*A-address*	*B-address*
Effective instruction:	$\overset{\vee}{\text{A}}$	00459	01534

Valid address on 10K, 20K, and 40K systems.

3. Modify the A- and B-addresses of this instruction:

Op Code	*A-address*	*B-address*
$\overset{\vee}{\text{S}}$	00V51	00W50

Both the A- and B-addresses are tagged by an A-bit over the hundreds position (V = A41 and W =A42). Index register 4 is selected. Because the index register 4 factor is unsigned, it is added to both addresses.

A-address = 00V51 = 00551
Index register 4 factor = +00100

Effective A-address = 00651

B-address = 00W50 = 00650
Index register 4 factor = +00100

Effective B-address = +00750

	Op Code	*A-address*	*B-Address*
Effective instruction:	$\overset{\vee}{\text{S}}$	00651	00750

Valid address on 10K, 20K, and 40K systems.

4. Modify the A-address of this instruction:

Op Code	*A-address*	*B-address*
$\overset{\vee}{\text{A}}$	126A8	06429

The A-address is tagged by an A- and B-bit over the tens position (A = BA1). Index register 3 is selected (Figure 29). Because the index register 3 factor is plus, it is added to the A-address.

A-address 126A8 = 12618
Index register 3 factor = 0643E = 06435

Effective A-address = 19053

	Op Code	*A-Address*	*B-address*
Effective instruction:	$\overset{\vee}{\text{A}}$	19053	06429

Valid address on 20K and 40K systems.

5. Modify the B-address of this instruction:

Op Code	*A-address*	*B-address*
$\overset{\vee}{\text{S}}$	10687	26UB5

The B-address is tagged by an A-bit over the hundreds position and a B- and A-bit over the tens position (U = CA4 and B = BA2). Index register 7 is selected (Figure 29). Because the index register 7 factor is plus, it is added to the B-address.

B-address = 26UB5 = 26425
Index register 7 factor = 0273D = 02734

Effective B-address = 29159

	Op Code	*A-address*	*B-address*
Effective instruction:	$\overset{\vee}{\text{S}}$	10687	29159

Valid address on 40K system.

Timing for indexing operations:
T = 34.5 for each single address indexed.

Note: For proper operation of indexing on a 10K system, the high-order position of the address being indexed must *always* contain a zero.

Logic Operations

The 1410 program can examine conditions that can arise during processing, and transfer the program to a predetermined set of instructions or subroutines as a result of specific tests. This is called the logical ability of the IBM 1410 Data Processing System. A transfer from one instruction to another instruction or set of instructions to alter the sequential execution of program steps is called a *program branch.* A branch instruction can be one of two types:

1. A branch that occurs as a direct result of the execution of the instruction itself is called an *unconditional branch.* Thus, no special condition (other than the execution of the program step) is needed to transfer the program out of its normal sequential execution.
2. A branch that occurs as a result of a particular condition such as an arithmetic overflow, zero balance, etc., is called a *conditional branch.* If the condition is present at the time a conditional branch instruction is executed, sequential execution of program steps is bypassed. Then, the program branches to the address of the instruction specified by the I-address of the conditional branch instruction. If the condition is not present, the machine takes the instruction that appears at the immediate right of the conditional branch instruction (next sequential instruction).

All branch instructions have a d-character that is used to specify the conditions necessary for a program transfer.

Logic Operation Codes

Branch Unconditional

Instruction Format.

Mnemonic	*Op Code*	*I-address*	*d-character*
B	$\overset{\vee}{\text{J}}$	xxxxx	blank

Function. This is an unconditional branch instruction. Whenever it is executed, it causes a program branch to the location specified by the I-address.

Word Marks. Word marks are not affected.

Timing. T = 4.5 (L + 2).
L = 1 or 7

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	BI	NSI

Test and Branch (Conditional)

Instruction Format.

Mnemonic	*Op Code*	*I-address*	*d-character*
xxx (See figure 30)	$\overset{\vee}{\text{J}}$	xxxxx	x

Function. This is a conditional branch instruction. It allows the program to test the conditions that can arise during the processing. The d-character specifies the condition or internal indicator that is examined for an ON or OFF condition. If the indicator is ON, the program branches to the I-address for the next instruction. If the indicator is OFF, the program continues with the next sequential instruction. The various indicators and the d-characters used to test them are shown in Figure 30.

The carriage channel 9 and carriage channel 12 indicators are turned on whenever the corresponding holes in the carriage tape are sensed. They are turned off whenever another carriage tape channel is sensed. The compare indicators are turned off by the next COMPARE, TABLE LOOKUP or TEST CHARACTER AND BRANCH instruction and are set to the result of

DESCRIPTION	MNEMONIC	d-CHARACTER
Branch Unconditional	B	Blank
Branch on Carriage Channel 9	BC9	9
Branch on Carriage Overflow (Channel 12)	BCV	@
Branch if Compare Unequal	BU	/
Branch if Compare Equal (B = A)	BE	S
Branch if Compare Low (B < A)	BL	T
Branch if Compare High (B > A)	BH	U
Branch if Zero Balance	BZ	V
Branch if Divide Overflow	BDV	W
Branch if Arithmetic Overflow	BAV	Z
Branch Inquiry	BNQ	Q
Branch if Overlap in Process on Channel 1	BOL 1	1
Branch if Overlap in Process on Channel 2	BOL 2	2
Branch if Printer Carriage Busy	BPCB	R

Figure 30. Test and Branch Conditions, Mnemonics, and d-Characters

that operation. A computer reset operation turns on the unequal and low compare indicators. The overflow indicators are turned off by the TEST AND BRANCH instruction or a computer reset operation. A computer reset operation also turns off the zero result indicator.

Word Marks. Word marks are not affected.

Timing. $T = 4.5 (L + 1 + C)$.
$L = 1 \text{ or } 7$

Address Registers after Operation.

	I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
Branch	NSI	BI	NSI
No branch	NSI	BI	B_p

Branch if I/O Channel Status Indicator On

Instruction Format.

Mnemonic	*Op Code*	*I-address*	*d-character*
xxx	$\overset{\vee}{R}$ (Ch. 1)	xxxxx	x
xxx	$\overset{\vee}{X}$ (Ch. 2)	xxxxx	x
(See figure 31)			

Function. This is a conditional branch instruction. Branching to the specified I-address occurs if I/O channel status indicators for channel 1 or channel 2, when tested, are ON. These indicators are set ON as a result of conditions that arise during an operation of one of the input-output units serviced by that channel. The bit configuration of the d-character used in the BRANCH IF I/O CHANNEL STATUS INDICATOR ON instruction determines which test or tests are made. This instruction with a ⧧ d-character must be given prior to the execution of another input or output instruction on the same channel to avoid interlocking the system. The system is interlocked if the status test is not satisfied. This status test is satisfied if either:

1. A BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction $\overset{\vee}{R}$ (I) ⧧ is given prior to the next I/O unit instruction, or;

2. A specific $\overset{\vee}{R}$ (I) d instruction (see Figure 31) is given and *results in a branch* prior to the next I/O unit instruction. Figure 31 is a chart of the I/O channel status indicators, the d-character bits that test them, and a brief write-up of their operation. For more detailed information concerning these indicators, refer to the specific I/O unit descriptive section of this manual.

DESCRIPTION	MNEMONIC	INDICATOR	d–CHARACTER BIT	OPERATION
Branch if input/output unit Not Ready	BNR 1 or 2	Not Ready	1	The indicator is internally set during instructions involving input/output devices, if these devices or their associated synchronizers are in a not ready condition, but before any data transfer takes place. If the indicator is set ON, the operation is terminated and no data are transferred.
Branch if input/output unit Busy	BCB 1 or 2	Busy	2	The indicator is internally set during instructions involving input/output devices, if these devices or their associated synchronizers are in a busy condition, but before any data transfer takes place. If the indicator is set ON, the operation is terminated and no data are transferred.
Branch if input/output Data Check	BER 1 or 2	Data Check	4	The indicator is set ON, after the transfer of data involving input/output devices, their associated synchronizers or the processing unit, if a parity error was detected during the data transfer.
Branch if input/output Condition	BEF 1 or 2	Condition	8	The indicator is normally set during the move or load instruction, before any data transfer takes place. As an example, the indicator is set ON if an end of file (last card stacked) has occurred in the card reader. If the indicator is set ON, the operation is terminated and no data are transferred.
Branch if input/output No Transfer	BNT 1 or 2	No Transfer	A	No Transfer. The indicator is normally set before any data transfer takes place. If it is set ON, it indicates that no data was available to transfer.
Branch if input/output Wrong Length Record	BWL 1 or 2	Wrong Length Record	B	The indicator is set ON, if the record written from storage or written in storage is not the correct length.

Note: These indicators are reset at the beginning of the next I/O operation

Figure 31. Branch If I/O Channel Status Indicator On Instruction

If an Ř (I) ǂ (BA 8 4 2 1 bits in the d-character) instruction is given following a channel 1 input-output operation, this instruction tests *all* the channel 1 indicators, and if any of them are on, the program branches to the specified I-address. Then the program can test the indicators (individually or in groups, determined by the bit structure of the d-characters) to determine the exact condition present. This technique saves total program execution time because individual test instruction need be given only if processing conditions call for them.

If the system is equipped with the input-output overlap feature, the program should test the overlap-in-process indicator with a TEST AND BRANCH instruction. This is done to insure that the overlapped I/O function is complete before the BRANCH IF I/O CHANNEL STATUS INDICATOR ON instruction is given. A BRANCH IF I/O CHANNEL STATUS INDICATOR ON instruction, executed while the machine is performing an I/O overlap operation, causes the machine to interlock until the overlapped function is complete.

Word Marks. Word marks are not affected.

Timing. T = 4.5 (L + 1 + C).

Address Registers after Operation.

	I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
Branch	NSI	BI	NSI
No branch	NSI	BI	B_p

Branch if Character Equal

Instruction Format.

Mnemonic	*Op Code*	*I-address*	*B-address*	*d-character*
BCE	B̌	xxxxx	xxxxx	x

Function. This instruction causes the bit configuration (BA 8 4 2 1 bits) of the character at the B-address to be compared to the bit configuration of the d-character. If the comparison is equal, the program branches to the I-address for the next instruction. If the two characters are not exactly the same, the program continues with the next sequential instruction. This instruction also results in the setting of the high, low, or equal indicator. The high indicator is set if the B-address character is higher than the d-character (collating sequence).

Word Marks. Word marks do not affect this operation. The nature of the instruction specifies that only one character is to be included in the test.

Timing. T = 4.5 (L + 2.5 + C).
L = 1, 6 or 12.

Address Registers after Operation.

	I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
Branch	NSI	BI	NSI
No branch	NSI	BI	B-1

Branch if Bit Equal

Instruction Format.

Mnemonic	*Op Code*	*I-address*	*B-address*	*d-character*
BBE	W̌	xxxxx	xxxxx	x

Function. This instruction causes the character at the B-address to be compared, bit by bit, with the d-character. If any bit in the character at the B-address matches any bit in the configuration of the d-character, the program branches to the I-address (WM and C bits not compared). For example, if position 0 5 8 9 6 (B-address) contains a Z (CBA 8 1 bits) and the d-character contains a 3 (C 2 1 bits), the program branches.

Word Marks. Word marks cannot be tested with this instruction and have no effect on the operation.

Timing. T = 4.5 (L + 2.5 + C).
L = 1, 6 or 12.

Address Registers after Operation.

	I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
Branch	NSI	BI	NSI
No branch	NSI	BI	B-1

Branch on Word Mark or Zone Equal

Instruction Format.

Mnemonic	*Op Code*	*I-address*	*B-address*	*d-character*
xxx (See Figure 32)	V̌	xxxxx	xxxxx	x

Function. This instruction examines the character located at the B-address for the zone or word mark combinations specified by the d-character. A correct comparison branches the program to the specified I-address. A 1-bit in the d-character examines the B-address for a word mark. A 2-bit compares the zone bits of the B-address character against the zone bits in the d-character. A combination of the 1-bit and 2-bit allows either a word mark or a correct zone bit comparison to initiate a branch to the specified I-address. If the program does not branch to the I-address, it continues with the next sequential instruction. The d-characters, and the conditions they test, are shown in Figure 32.

Word Marks. A word mark is not required at the B-address to stop transmission because this is always a one-character operation.

Timing. $T = 4.5 (L + 2.5 + C)$.

$L = 1, 6$ or 12

Address Registers after Operation.

	I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
Branch	NSI	BI	NSI
No branch	NSI	BI	B-1

DESCRIPTION	MNEMONIC	d-CHARACTER
Branch on Word Mark	BW	1
Branch on Zone	BZN	2, B, K, S
Branch on Word Mark or Zone	BWZ	3,C,L,T

Figure 32. Branch On Word Mark or Zone Equal Instruction

These operations are used to manipulate data within core storage during processing. They include data moving, comparing, table lookup, and editing.

Data Moving

This concerns moving data, either left-to-right or right-to-left, from the A-field to the B-field (with or without word marks). Data can be moved by fields or by records. If a data field is moved, the operation can be programmed to stop at:

1. a word mark in the A-field; or
2. a word mark in the B-field; or
3. a word mark in either field.

If a record is moved, the operation can be programmed to stop at:

1. a record mark in the A-field; or
2. a group-mark with a word-mark in the A-field; or
3. either a record mark or group-mark with a word-mark in the A-field.

The operation code for the move instruction is $\overset{\vee}{\text{D}}$. The bit structure of the d-character used with the move instruction determines the type of operation that will be performed (Figure 33 — all 64 characters composed of the bits shown in the figure are valid; each one accomplishes a special purpose). These operations are:

1. The transfer of the numerical portion of the data field
2. The transfer of the zone portion of the data field
3. The transfer of word marks from the A-field to the B-field
4. The scanning of the A-field and B-field for word marks, record marks, or group-marks with word-marks (this operation is used when the storage positions containing record marks, group marks, and word marks can vary from one record to another — no data are transferred)

Move Instructions

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*	*d-character*
x - - x (See Figure 36)	$\overset{\vee}{\text{D}}$	xxxxx	xxxxx	x

Function. Data are moved from left to right or from right to left, serial by character, from the A-field to the B-field under control of the d-character (Figure 33).*

Only the portion of the A-field that is transferred replaces the corresponding portion of the B-field (the remainder of the B-field is unchanged). If data are moved from left to right, the A-address specifies the high-order position of the A-field; the B-address specifies the high-order position of the B-field. If data are moved from right to left, the A-address specifies the low-order position of the A-field; the B-address specifies the low-order position of the B-field. The position that contains the terminating

d–CHARACTER CONTROL BITS		CONTROL
1		Transfer of numerical portion of data field
2		Transfer of zone portion of data field
4		Transfer word marks from A–field to B–field
Blank (No. 1, 2, or 4 Bit)		Scan for word marks, record marks, or group-marks with word-marks
8–BIT (LEFT TO RIGHT MOVE)	No A– and No B–Bits	Stop transfer or scan at first word mark sensed in either field
	* A–Bit Only	Stop transfer or scan at A–field record mark
	B–Bit Only	Stop transfer or scan at A–field group-mark with word-mark
	A– and B–Bits	Stop transfer or scan at A–field record mark or group-mark with word-mark
NO 8–BIT (RIGHT TO LEFT MOVE)	No A– and No B–Bits	Transfer or scan only one storage position
	* A–Bit Only	Stop transfer or scan at A–field word mark
	B–Bit Only	Stop transfer or scan at B–field word mark
	A– and B–Bits	Stop transfer or scan at first word mark sensed in either field

*Whenever the A-bit d-character modifier is used in instructions to write programs on tape, the odd parity mode should be used. See Figure 107.

Figure 33. d-Character Control Bits for Move Instructions

character is moved or replaced the same as the rest of the field.

This same instruction, with the appropriate d-characters, is also used for scan operations (no data transferred).

Word Marks. See Figure 33.

Timing. $T = 4.5 (L + 1 + A + 1.5B)$.
$L = 1, 6 \text{ or } 12$

Address Registers after Operation. See Figure 34.

Mnemonics

Because each mnemonic character has a special meaning (Figure 35), it is possible to construct the entire mnemonic for any of the sixty-four move instructions by applying certain rules. These rules are:

DATA TRANSFERRED

1. The first character of the mnemonic is M.
2. The second character of the mnemonic specifies the direction of data movement, either left-to-right or right-to-left (L is right-to-left; R is left-to-right).
3. The third section of the mnemonic specifies the portion of data moved. If only one portion of data is moved, this section contains a single mnemonic character (W, Z, N, or C). If word marks and one other portion of data are moved, this section contains two mnemonic characters (ZW, NW, or CW).
4. The fourth section of the mnemonic specifies the terminal point. If more than one data character is moved, the terminal point character is A, B, blank, R, G, or M. If only one data character is moved, the terminal point character is S.

NO DATA TRANSFERRED (SCAN)

1. The first three characters of the mnemonic are SCN.
2. The fourth character of the mnemonic specifies the direction of scan, either L or R.
3. The fifth character of the mnemonic specifies the

CONTROL	MNEMONIC CHARACTER	MEANING	DESCRIPTION
Direction or Type of Operation	M	Move	Move data serial by character
	SCN	Scan	Affect A- and B-address registers only, do not move data
	L	Left	Right to left operation
	R	Right	Left to right operation
Portion of Data Transferred	N	Numerical	Move only numerical portion of data
	Z	Zone	Move only zone portion of data
	C	Character	Move character(s) (zone and numerical portions of data)
	W	Word Mark	Move word mark(s)
TERMINAL POINT	A (L)	A-Field Word Mark	Stop at A-field word mark
	B (L)	B-Field Word Mark	Stop at B-field word mark
	blank (L or R)	Either A- or B-Field Word Mark	Stop at first word mark sensed in either A- or B-field
	S (L)	One Position	Affect only one position
	R (R)	Record Mark	Stop at A-field record mark
	G (R)	Group Mark	Stop at A-field group-mark, word-mark
	M (R)	Record or Group Mark	Stop at A-field record mark or group-mark, word-mark

Figure 35. Mnemonic Characters

CONTROL	DIRECTION	ADDRESS REGISTERS		
		I–Add. Reg.	A–Add. Reg.	B–Add. Reg.
Stop at first word mark sensed in either field	L to R	NSI	A + LW	B + LW
Stop at A–field record mark	L to R	NSI	A + LA	B + LA
Stop at A–field group-mark with word-mark	L to R	NSI	A + LA	B + LA
Stop at A–field record mark or group-mark with word-mark	L to R	NSI	A + LA	B + LA
Stop after one storage position	R to L	NSI	A − 1	B − 1
Stop at A–field word mark	R to L	NSI	A − LA	B − LA
Stop at B–field word mark	R to L	NSI	A − AB	B − LB
Stop at first word mark sensed in either field	R to L	NSI	A − LW	B − LW

Figure 34. Address Registers after Move Operations

Direction of Move	Condition Which Ends Operation	No Portion Moved	Move Numeric Portion of A-Field to B-Field	Move Zone Portion of A-Field to B-Field	Move Numeric and Zone from A-Field to B-Field	Move WM in A-Field to B-Field	Move Numeric and WM from A-Field to B-Field	Move Zone and WM from A-Field to B-Field	Move Numeric, Zone, and WM from A-Field to B-Field	BCD Coding (Bits)
RIGHT TO LEFT	Move data one position	blank SCNLS	1 MLNS	2 MLZS	3 MLCS	4 MLWS	5 MLNWS	6 MLZWS	7 MLCWS	NONE
	Move data to 1st WM in A-field	¢ SCNLA	/ MLNA	S MLZA	T MLCA	U MLWA	V MLNWA	W MLZWA	X MLCWA	A
	Move data to 1st WM in B-field	– SCNLB	J MLNB	K MLZB	L MLCB	M MLWB	N MLNWB	O MLZWB	P MLCWB	B
	Move data to 1st WM in either A- or B-field	& SCNL	A MLN	B MLZ	C MLC	D MLW	E MLNW	F MLZW	G MLCW	A B
LEFT TO RIGHT	Move record to 1st WM in either A- or B-field	8 SCNR	9 MRN	0 MRZ	# MRC	@ MRW	: MRNW	> MRZW	√ TM MRCW	8
	Move record to 1st RM in A-field	Y SCNRR	Z MRNR	⧧ MRZR	, MRCR	% MRWR	= WS MRNWR	apostrophe ' MRZWR	" MRCWR	A 8
	Move record to 1st GM-WM in A-field	Q SCNRG	R MRNG	! MRZG	$ MRCG	* MRWG	) MRNWG	; MRZWG	Δ MRCWG	B 8
	Move record to 1st RM or GM-WM in A-field	H SCNRM	I MRNM	? MRZM	• MRCM	⌑ MRWM	(MRNWM	< MRZWM	⯒ MRCWM	A B 8
	BCD CODING (Bits)	NONE	1	2	1, 2	4	1, 4	2, 4	1, 2, 4	

Figure 36. Data Move d-Characters and Mnemonics

terminal point. The terminal point character is A, B, blank, R, G, or M. If only one position of storage is scanned, the terminal point character is S.

Figure 36 is a complete chart of the data move d-characters and mnemonics.

Move Characters and Suppress Zeros

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
MCS	v Z	xxxxx	xxxxx

Function. This instruction causes the data in the A-field to be moved to the B-field. The A-field remains unchanged after the operation. High-order zeros and commas in the B-field are replaced by blanks, and zone bits in the units (sign) position are removed. Refer to Figure 37 for an example of MOVE CHARACTERS AND SUPPRESS ZEROS.

Figure 38 is another example of the MOVE CHARACTERS AND SUPPRESS ZEROS instruction involving a

Example	Op Code	A–address	B–address
Move Char. and Suppress Zeros	v Z	xxxxx	xxxxx
Storage before		A–field (data) v ± 001206	B–field (data) v bbbbbbb
Storage after		v ± 001206	v bbb1206

Figure 37. Move Characters and Suppress Zeros Example

Example	Op Code	A–address	B–address
Move Char. and Suppress Zeros	$\overset{v}{Z}$	xxxxx	xxxxx
Storage before		A–field (data)	B–field (data)
		$\overset{v}{0}010b@00.2\overset{\pm}{5}$	$\overset{v}{b}bbbbbbbbbbb$
Storage after		$\overset{v}{0}010b@00.2\overset{\pm}{5}$	$\overset{v}{b}bb10b@bb.25$

Figure 38. Move Characters and Suppress Zeros Example, Multiple Field

multiple field transfer. In this operation there are effectively two groups of high-order zeros. The @ sign is recognized as not being a significant digit or a zero, blank, comma, decimal, or minus sign. Thus, not only are the two high-order zeros suppressed, but also the two zeros to the right of the @ sign.

Word Marks. The A-field must have a defining word mark. It is this word mark that specifies the length of the data moved to the B-field. B-field word marks within this specified area are removed during the operation.

Timing. $T = 4.5 (L + 1 + 4A)$.
$L = 1, 6 \text{ or } 11.$

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A – LA	B + 1

Comparing

The IBM 1410 compares data fields by testing the bit structure of each character in the B-field with the bit structure of each character in the A-field. The result of the compare operation is determined by the collating sequence of 1410 characters (see Figure 14). B can be equal to, unequal to, higher than, or lower than A.

Compare

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
C	$\overset{v}{C}$	xxxxx	xxxxx

Function. The data in the B-field are compared to the data in the A-field. The operation does not change either field. The result of the compare sets the high ($B > A$), equal ($B = A$), or low ($B < A$) indicator, depending on whether the B-field data are high, equal, or low with respect to the A-field. These indicators can be tested by a subsequent TEST AND BRANCH instruction.

Word Marks. The compare operation is terminated by either an A-field or a B-field word mark. If the A-field is shorter than the B-field, it must also have a defining word mark. In this case, the high-compare indicator ($B > A$) is turned on.

Note: The compare indicators must be tested before the next COMPARE, BRANCH IF CHARACTER EQUAL or TABLE LOOKUP instruction is executed.

Timing. $T = 4.5 (L + 1 + A + B)$.
$L = 1, 6 \text{ or } 11.$

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A – LW	B – LW

Table Lookup

Many commercial and scientific applications are characterized by the need to search through a table for rates, mathematical factors, or other types of information that vary with the requirements of the input data.

The IBM 1410 Data Processing System has a powerful instruction that causes the machine to search through the table and find the *function* (desired factor or address of desired factor) – the TABLE LOOKUP instruction.

To do this, the machine requires two arguments in addition to the function. They are the search argument and the table argument.

Search Argument

The search argument is a data field that has been generated internally or read into the system from a card, tape, disk record, or other input medium. It is used to find the table argument.

Table Argument

The table argument is kept in a table of arguments in core storage. It is exactly the same number of characters as the search argument. If it is shorter, it signifies the end of the table and ends the table search.

Function

The function is kept in core storage with the table argument. If the desired factor is five positions or less, it is often practical to store the factor itself in this place. In this case, the desired factor is the function. If the desired factor is more than five characters, it is usually kept in another area of core storage. In this case, the function is the 5-character address of the desired factor. Because the timing of the table lookup operation is determined by the number of characters in the table that are read before a table argument is found, it is desirable to have the least possible number of characters in the function.

Another suggested method for reducing the number of characters in the function is to store the desired factors in a separate table and store the starting address of this table in an accumulator field. If the function contains a factor (less than five characters) that can be added to the starting address to give the actual address of the desired factor, the lookup operation takes less time.

Function values may also be stored a fixed number of core-storage positions from their arguments. Thus, having found the location N of the argument, the function is located at $N + C$, where C is the fixed separation of the functions from the arguments.

FINDING THE FUNCTION

The operation can be programmed to stop when a table argument is found that is equal to the search argument. The program can then move the desired factor to a working area for processing. If the function is the desired factor, it can be moved directly to the working area. If not, it is necessary to bring out the address of the desired factor and then move the factor to the working area.

A table lookup operation can also be stopped if a table argument is found that is higher than, or lower than, the search argument, or when the B-field (table argument) is shorter than the A-field (search argument). The latter condition results in setting the high compare indicator ON.

Table Lookup

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*	*d-character*
xxx (See Figure 39)	$\overset{v}{\text{T}}$	xxxxx	xxxxx	x

Function. This instruction causes the machine to search for a table argument that is equal to, lower than or higher than the search argument as specified by the d-character. A table lookup operation stops on the first table argument that satisfies the table lookup command. Refer to Figure 39 for valid d-characters used for table lookup instructions.

The A-field contains the search argument. The B-address is the address of the low-order character of the entire table. The number of characters in the table argument must be equal to the number of characters in the search argument. However, the field in the table that contains the table argument and the function can be longer than the search argument.

At the end of the operation, one of the high, low, or equal compare indicators is turned on as a result of the last field compared.

Word Marks. The search argument (A-field) must have a word mark set to define the high-order position.

DESCRIPTION	MNEMONIC	d–CHARACTER	BIT CONFIGURATION	TABLE SEARCH RESULT
Lookup to End	—	b̸	—	Search to end of table
Lookup Low	LL	1	1	Lower than search argument
Lookup Equal	LE	2	2	Equal to search argument
Lookup Low or Equal	LLE	3	21	Equal to or lower than search argument
Lookup High	LH	4	4	Higher than search argument
Lookup Low or High	LLH	5	41	Lower than or higher than search argument
Lookup Equal or High	LEH	6	42	Equal to or higher than search argument
Lookup to Any	—	7	421	Stop on any

Figure 39. Valid d-Characters for Table Lookup Instructions

The table field (including the argument in the low-order positions and the function in the high-order position) must have a defining word mark in its high-order position. The A-field word mark stops the comparison against the table argument. The machine starts to compare again at the position immediately at the left of the word mark in the table field (low-order position of next table argument).

Timing. $T = 4.5 (L + 1 + B + NA)$.

$L = 1, 6$ or 12.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	Address of the function at immediate left of the table argument that stopped the operation.

Note: If a table field is found that is shorter than the search argument, the B-address register will contain the address of the position at the immediate left of the short table field. Thus, a short table field can be used to signal the end of the table. This condition results in setting the high compare indicator ON.

EXAMPLE. Find the unit price of part number 1002. The unit price is the desired factor, and the part number is the search argument and the table argument (Figure 40).

The table is searched from low-order to high-order position. In this case, the search starts at position 1243. When the search argument equals the table argument (1002), the search stops.

Search Argument	TABLE	
	Function (Desired Factor)	Table Argument
1 0 0 2	$\overset{v}{5}$ 9 8	1 0 0 0
	$\overset{v}{6}$ 9 8	1 0 0 2
	$\overset{v}{2}$ 9 7	1 0 0 3
	$\overset{v}{1}$ 9 7	1 0 0 4
	$\overset{v}{1}$ 9 8	1 0 0 5
	$\overset{v}{2}$ 9 8	1 0 0 6
	$\overset{v}{3}$ 9 5	1 2 4 1
	$\overset{v}{4}$ 9 5	1 2 4 2
	$\overset{v}{6}$ 9 5	1 2 4 3

Figure 40. Table Lookup Operation

Editing

The IBM 1410 Data Processing System has a powerful EDIT instruction that can cause all desired commas, decimals, dollar signs, asterisks, credit symbols, and minus signs to be inserted automatically in a numerical output field. Also, unwanted zeros to the left of significant digits can be suppressed (Figure 41). A step-by-step editing process of this example is shown in Figure 47. Thus, editing in the IBM 1410 is the automatic control of zero suppression, inserting of identifying symbols, and punctuation of an output field.

In editing, two fields are needed: the data field and a control field. The data field is the data to be edited for output. The control field specifies the conditions for the edit operation (how the data field is to be edited). It specifies the location of punctuation and condition of special characters, and indicates where zero suppression is to occur.

The control field is divided into two parts: the body (used for punctuating the A-field) and the status portion (containing the special characters). The *body* of the control word is that portion beginning with the right-most blank or zero and continuing to the left until the A-field word mark is sensed. The remaining portion of the control field is the *status* portion. Sign printing is partly controlled by the sign of the A-field.

Example	Op Code	A-address	B-address
Edit Inst.	$\overset{v}{E}$	12163	04685
		A-field (data)	B-Field (control word)
Storage		$\overset{v}{0}$0257426	$\overset{v}{\$}$bbb,bb0.bb&CR&**
Result of Edit			B-field
Storage		$\overset{v}{0}$0257426	$ 2,574.26 **

Figure 41. Editing

An edit operation requires two instructions. A MOVE instruction is used to transfer the control word and its word mark to the output area; the EDIT instruction moves the data to the output area and performs the editing function.

Move Characters and Edit

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
MCE	$\overset{v}{E}$	xxxxx	xxxxx

Function. The data field (A-field) is modified by the contents of the edit control field (B-field), and the result is stored in the B-field. The data field and the control field are read from storage alternately, character by character, under control of the word marks and the editing specifications. See *Editing Specifications.* Any sign in the units position of the data field is removed during the operation.

Word Marks. A word mark must be set in the high-order position of the B-field to control the edit operation. The A-field must also have a defining word mark. When the A-field word mark is sensed, the remaining commas in the B-field are set to blanks. The edited output field does not contain any A-field data that have not been moved before the word mark for the control field is sensed. The data field can contain fewer, but should not contain more, positions than the number of blanks and zeros in the body of the control word.

Timing. $T = 4.5 (L + 1 + A + 1.5B + 1.5Z + 1.5D)$.
$L = 1, 6$ or 11.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	A-LA	Varies with result of edit.

Editing Specifications

All numerical, alphabetic, and special characters can be used in the control word. However, some of these have special meanings as listed in the following:

Control Character	Function
b (blank)	Replaced with the character from the corresponding position of the A-field.
0 (zero)	Used for zero suppression. Replaced with a corresponding character from the A-field. The right-most 0 in the control word indicates the right-most limit of zero suppression.
. (decimal)	This remains in the edited field in the position where written, unless decimal control was in effect, and the data field did not contain a significant digit. (See *Decimal Control.*)

, (comma)	Undisturbed in the output data field in the position where written, unless zero suppression takes place and no significant numerical character is found at the left of the comma.
CR (credit)	Body portion: undisturbed in the position where written. Status portion: if sign of the data field is plus, these two positions are replaced by blanks. If the sign of the data field is minus they are undisturbed in the output field in the positions where written. (See also *Sign Control Left.*)
– (minus)	Same as CR.
& (ampersand)	Causes a blank space in the output field. It can be used in multiples.
* (asterisk)	Status Portion: undisturbed in the position where written. Body Portion: (See *Asterisk Protection.*)
$ (dollar)	Status Portion: undisturbed in the position where written. Body Portion: (See *Floating Dollar Sign.*)

Zero Suppression

Zero suppression is the deletion of unwanted zeros at the left of significant digits in an output field (Figure 42).

A special 0 is placed (in the body of the control word) in the right-most limit of zero suppression.

FORWARD SCAN:

1. The positions in the output field at the right of this special zero are replaced by the corresponding digits from the A-field.
2. When the special zero is detected in the control field, it is replaced by the corresponding digit from the A-field.
3. A word mark is automatically set in this position of the B-(output) field.
4. The scan continues until the B-field (high-order) word mark is sensed and removed.

REVERSE SCAN:

1. All zeros and punctuation at the left of the first significant character (up to and including the zero suppression code position) are replaced by blanks in the output field.
2. When the automatically set zero suppression word mark is sensed, it is erased and the operation ends.

```
EXAMPLE:
                                      v
    A-field                           0010900
                                   v
    Control word (B-field)         $bb,bb0.bb
                                         v
    Forward scan                   $00,109.00

    Reverse scan                   $bbb109.00

    Results of edit                $   109.00
```

Figure 42. Zero Suppression

Asterisk Protection

When it is necessary to have asterisks appear at the left of significant digits, the asterisk protection feature is used (Figure 43).

The control word is written with the asterisk in the body to the left of the zero suppression code (if the asterisk appears in the body to the right of the zero suppression code, it is treated as a blank).

FORWARD SCAN:

1. The normal editing process proceeds until the asterisk is sensed.
2. The asterisk is replaced (in the output field) by the corresponding digit from the A-field.
3. The editing process continues normally until the B-field word mark is sensed and removed.

REVERSE SCAN:

1. Zeros, blanks, and punctuation to the left of the first significant digit are replaced by asterisks.
2. The word mark (set during the forward scan) signals the end of editing. It is erased, and the operation stops.

Note: Asterisk protection and floating dollar sign cannot be used in the same control word.

```
EXAMPLE:
                                     v      -
    A-field                          00257426
                                   v
    Control word (B-field)         bbb,b*0.bb&CR
                                         v
    Forward scan                   002,574.26 CR

    Reverse scan                   **2,574.26 CR

    Results of edit                **2,574.26 CR
```

Figure 43. Asterisk Protection

Floating Dollar Sign

This feature causes the insertion of a dollar sign in the position at the left of the first significant digit in an amount (Figure 44).

The control word is written with the "$" in the body to the left of the zero suppression code (if the dollar sign appears in the body to the right of the zero suppression code, it is treated as a blank).

EXAMPLE:

A-field	$\overset{v}{0}0257426$
Control word (B-field)	$\overset{v}{\text{b}}\text{bb,b\$0.bb}$
First forward scan	$\text{002,57}\overset{v}{4}\text{.26}$
Reverse scan	bb2,574.26
Second forward scan	$2,574.26
Results of edit	$2,574.26

Figure 44. Floating Dollar Sign

Three scans are necessary to complete this editing operation.

FIRST FORWARD SCAN:

1. The editing proceeds until the "$" is sensed.
2. The "$" is replaced (in the output field) by the corresponding digit from the A-field.
3. Editing continues until the B-field word mark is sensed and removed.

REVERSE SCAN:

1. Zeros and punctuation to the left of the first significant digit are replaced by blanks.
2. The reverse scan continues until the word mark (set during the first forward scan) signals the start of the second forward scan.

SECOND FORWARD SCAN:

1. The word mark is erased, and the scan continues until the first blank position is sensed. This blank position is replaced by "$", and the operation stops.

Note: Floating dollar sign cannot be used at the right of the decimal point.

Sign Control Left

CR or — symbols can be placed at the left of a negative field (Figure 45).

The control word is written with the CR or — symbols in the high-order status position.

FORWARD SCAN:

1. The scan proceeds until the zero suppression character in the control field is sensed.
2. The corresponding character from the A-field is placed in this position of the output field.
3. A word mark is automatically inserted in this position in the output field.
4. The scan proceeds until the A-field word mark is sensed, indicating the end of the body of the control word.
5. Editing continues and the CR or — symbols are undisturbed in their corresponding positions in the output field, only if the sign of the A-field is minus. If the sign is plus, they are blanked.

EXAMPLE:

A-field	$\overset{v}{0}037894\overline{0}$
Control word (B-field)	$\overset{v}{\text{C}}\text{R\&bbb,bb0.bb}$
Forward scan	$\text{CRb003,78}\overset{v}{9}\text{.40}$
Reverse scan	CRbbb3,789.40
Results of edit	CR 3,789.40

Figure 45. Sign Control Left

REVERSE SCAN:

1. Zeros and punctuation are replaced by blanks in the output field. The scan continues until the automatically set word mark is sensed.
2. This word mark is erased and the operation ends.

Decimal Control

This feature insures that decimal points print only when there are significant digits in the A-field (Figure 46).

The control word is written with a decimal in the body to the left of the zero suppression code.

Two scans are sufficient to complete this editing operation *unless* the field contains no significant digits Then three scans are required.

FIRST FORWARD SCAN:

1. When the zero suppression code (0) is sensed during editing, this position is replaced by the corresponding digit from the A-field.
2. A word mark is set automatically in this position in the B-(output) field.

EXAMPLES:

1.	A–field	$\overset{v}{0}0000$
	Control word (B–field)	$\overset{v}{\text{b}}\text{bb.b0}$
	First forward scan	$000.0\overset{v}{0}$
	Reverse scan	bbb.00
	Second forward scan	bbb
	Results of edit	(Blank Field)
2.	A–field	$\overset{v}{2}9437$
	Control word (B–field)	$\overset{v}{\text{b}}\text{bb.b0}$
	First forward scan	$294.3\overset{v}{7}$
	Reverse scan	294.37
	Result of edit	294.37
3.	A–field	$\overset{v}{0}0001$
	Control word (B–field)	$\overset{v}{\text{b}}\text{bb.b0}$
	First forward scan	$000.0\overset{v}{1}$
	Reverse scan	bbb.01
	Results of edit	.01

Figure 46. Decimal Control

STEP	TYPE OF CYCLE	ADDRESS REGISTERS			DATA REGISTER		PUT BACK INTO STORAGE	B-FIELD AT END OF CYCLE	REMARKS
		I	A	B	B	A			
1	Iop	00002	?????	?????	$\overset{v}{E}$	$\overset{v}{E}$	$\overset{v}{E}$	$\overset{v}{\$}$bbb,bb0.bb&CR&**	Read Instruction Op Code
2	I1	00003	1????	?????	1	1	1	Same	Load A-address register
3	I2	00004	12???	?????	2	2	2	Same	Load A-address register
4	I3	00005	121??	?????	1	1	1	Same	Load A-address register
5	I4	00006	1216?	?????	6	6	6	Same	Load A-address register
6	I5	00007	12163	?????	3	3	3	Same	Load A-address register
7	I6	00008	12163	0????	0	0	0	Same	Load B-address register
8	I7	00009	12163	04???	4	4	4	Same	Load B-address register
9	I8	00010	12163	046??	6	6	6	Same	Load B-address register
10	I9	00011	12163	0468?	8	8	8	Same	Load B-address register
11	I10	00012	12163	04685	5	5	5	Same	Load B-address register
12	I11	00012	12163	04685	$\overset{v}{O}$p	$\overset{v}{O}$p	$\overset{v}{O}$p	Same	Op Code & next instruction
13	A	00012	12162	04685	6	6	6	Same	Execute EDIT instruction
14	B	00012	12162	04684	*	6	*	Same	
15	B	00012	12162	04683	*	6	*	Same	
16	B	00012	12162	04682	&	6	Blank	$\overset{v}{\$}$bbb,bb0.bb&CRb**	
17	B	00012	12162	04681	R	6	Blank	$\overset{v}{\$}$bbb,bb0.bb&Cbb**	
18	B	00012	12162	04680	C	6	Blank	$\overset{v}{\$}$bbb,bb0.bb&bbb**	
19	B	00012	12162	04679	&	6	Blank	$\overset{v}{\$}$bbb,bb0.bbbbbb**	
20	B	00012	12162	04678	b	6	6	$\overset{v}{\$}$bbb,bb0.b6bbbb**	
21	A	00012	12161	04678	2	2	2	Same	
22	B	00012	12161	04677	b	2	2	$\overset{v}{\$}$bbb,bb0.26bbbb**	
23	A	00012	12160	04677	4	4	4	Same	
24	B	00012	12160	04676	•	4	•	Same	
25	B	00012	12160	04675	0	4	$\overset{v}{4}$	$\overset{v}{\$}bbb,bb\overset{v}{4}$.26bbbb**	Zero Suppress
26	A	00012	12159	04675	7	7	7	Same	
27	B	00012	12159	04674	b	7	7	$\overset{v}{\$}$bbb,b7$\overset{v}{4}$.26bbbb**	
28	A	00012	12158	04674	5	5	5	Same	
29	B	00012	12158	04673	b	5	5	$\overset{v}{\$}$bbb,57$\overset{v}{4}$.26bbbb**	
30	A	00012	12157	04673	2	2	2	Same	
31	B	00012	12157	04672	,	2	,	Same	
32	B	00012	12157	04671	b	2	2	$\overset{v}{\$}$bb2,57$\overset{v}{4}$.26bbbb**	

Figure 47. Step-by-Step Editing Process

STEP	TYPE OF CYCLE	ADDRESS REGISTERS			DATA REGISTER		PUT BACK INTO STORAGE	B-FIELD AT END OF CYCLE	REMARKS
		I	A	B	B	A			
33	A	00012	12156	04671	0	0	0	Same	
34	B	00012	12156	04670	b	0	0	$\overset{v}{\$}\text{b02,57}\overset{v}{4}\text{.26bbbb**}$	
35	A	00012	12155	04670	$\overset{v}{0}$	$\overset{v}{0}$	$\overset{v}{0}$	Same	
36	B	00012	12155	04669	b	$\overset{v}{0}$	0	$\overset{v}{\$}\text{002,57}\overset{v}{4}\text{.26bbbb**}$	
37	B	00012	12155	04668	$\overset{v}{\$}$	$\overset{v}{0}$	$	$\text{\$002,57}\overset{v}{4}\text{.26bbbb**}$	Sense Word Mark — Rev. Scan
38	B	00012	12155	04669	?	$\overset{v}{0}$	?	$\text{\$002,57}\overset{v}{4}\text{.26bbbb**}$	Units Position of next Field
39	B	00012	12155	04670	$	0	$	Same	
40	B	00012	12155	04671	0	$\overset{v}{0}$	Blank	$\text{\$b02,57}\overset{v}{4}\text{.26bbbb**}$	
41	B	00012	12155	04672	0	$\overset{v}{0}$	Blank	$\text{\$bb2,57}\overset{v}{4}\text{.26bbbb**}$	
42	B	00012	12155	04673	2	$\overset{v}{0}$	2	Same	
43	B	00012	12155	04674	,	$\overset{v}{0}$	,	Same	
44	B	00012	12155	04675	5	$\overset{v}{0}$	5	Same	
45	B	00012	12155	04676	7	$\overset{v}{0}$	7	Same	
46	B	00012	12155	04677	$\overset{v}{4}$	$\overset{v}{0}$	4	$bb2,574.26bbbb**	

Figure 47. (Continued)

3. Editing continues normally until the B-field word mark is sensed and removed.

REVERSE SCAN:

1. Zeros and punctuation are replaced by blanks in the output field until the decimal point is sensed.
2. The decimal point and the digits at its right are unaltered. The automatically set word mark is erased. If there are no significant digits in the field, the second forward scan is initiated. Otherwise, the edit operation stops.

SECOND FORWARD SCAN:

1. The zeros at the right of the decimal point and the decimal point itself are replaced by blanks.
2. The operation stops at the decimal column.

Figure 47 is a step-by-step editing process of the example shown in Figure 41.

Miscellaneous Operations

This section describes the STORE ADDRESS REGISTER, SET WORD MARK, CLEAR WORD MARK, CLEAR STORAGE, HALT and NO-OPERATION instructions that are used to facilitate programming and prepare storage areas for processing data fields.

Store Address Register

This operation makes it possible to store the contents of the A-, B-, E-, and F-address registers after any operation. It is particularly useful when fields or records of variable length are being processed, or when a method of linking a main routine with a subroutine is desired. For example, the address of the next sequential instruction is stored in the B-address register after a program branch to the I-address occurs. If the first step of the subroutine stores the contents of the B-address register in the last step of the subroutine (BRANCH UNCONDITIONAL instruction), the program branches back to the next instruction of the main routine after the subroutine is executed.

Instruction Format.

Mnemonic	*Op Code*	*C-address*	*d-character*
xxx	$\overset{\vee}{\text{G}}$	xxxxx	A, B, E, F

(See Figure 48.)

Function. The contents of the register specified by the d-character are stored in the C-field. The C-address specifies the low-order position of the field in core storage where the register contents will be stored.

Word Marks. Word marks in the C-field have no effect on the operation. *Note:* If there are zones in the C-field, they are not disturbed.

OPERATION	MNEMONIC	d-CHARACTER
Store A-Address Register	SAR	A
Store B-Address Register	SBR	B
Store E-Address Register	SER	E
Store F-Address Register	SFR	F

Figure 48. Store Address Register Mnemonics and d-Characters

Timing. T = 69.75.

Note: This instruction cannot be indexed.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	Bp

Set Word Mark

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
SW	$\overset{\vee}{,}$	xxxxx	xxxxx

Function. If this instruction is given as shown in the instruction format, a word mark is set in the specified A-address location and in the specified B-address location. The data characters in the specified locations are not disturbed.

If this instruction is given with only one address (A-address), a word mark is set in the specified A-address location only. The data character in the specified location is not disturbed.

If this instruction is given with no address specified (a no-address chained instruction), word marks are set in the address locations that are specified by the A- and B-address registers (contents from the previous operation).

Word Marks. Word marks are explained in previous paragraph.

Timing. T = 4.5 (L + 4).

L = 1, 6, or 11.

Address Registers after Operation.

	I-Add. Reg.	*A-Add. Reg.*	*B-address*
Two Addresses	NSI	A − 1	B − 1
One Address	NSI	A − 1	A − 1
No Addresses	NSI	Ap − 1	Bp − 1

Clear Word Mark

Instruction Format.

Mnemonic	*Op Code*	*A-address*	*B-address*
CW	$\overset{\vee}{⌑}$	xxxxx	xxxxx

Function. If this instruction is given as shown in the instruction format, a word mark is cleared, if present, from the specified A-address location and from the specified B-address location. The data characters in the specified locations are not disturbed.

If this instruction is given with only one address (A-address), a word mark, if present, is cleared from the specified A-address location only. The data characters in the specified location are not disturbed.

If this instruction is given with no address specified (a no-address chained instruction), word marks, if present, are cleared from the address locations that are specified by the A- and B-address registers (contents from the previous operation).

Word Marks. Word marks are explained in previous paragraph.

Timing. T = 4.5 (L + 4).
L = 1, 6, or 11.

Address Registers after Operation.

	I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
Two addresses	NSI	A − 1	B − 1
One address	NSI	A − 1	A − 1
No addresses	NSI	Ap − 1	Bp − 1

Clear Storage

Instruction Format.

Mnemonic	*Op Code*	*B-address*
CS	$\overset{\text{v}}{/}$	xxxxx

Function. A storage area is cleared of data and word marks, right-to-left, from the specified B-address location to, and including, the nearest hundreds position. For example, to clear storage from 12590 to 12500, use a $\overset{\text{v}}{/}$ 12590 instruction.

If this instruction is given with no address specified (a no-address chained instruction), the contents of the B-address register are used as the B-address location. (In this case, the A-address register is not loaded at instruction loading time, and is undisturbed at the end of the clear-storage operation.) By chaining the instruction in this manner, several blocks of 100 core-storage positions can be quickly cleared.

For clearing larger blocks of core storage, a simple program loop (as shown in Figure 49) proves more efficient. This example clears the core-storage area from positions 00500 to 36199. The first instruction sets a word mark in the low-numbered position of the core-storage area being cleared.

INSTRUCTION ADDRESS					INSTRUCTION											
					OP	A/I FIELD					B–FIELD					*d*
						d	x–ctrl fld		*d*		*d*					
0	0	1	2	3	,	0	0	5	0	0						
0	0	1	2	9	/						3	6	1	9	9	
0	0	1	3	5	G	0	0	1	3	4	B					
0	0	1	4	2	V	0	0	1	2	9	0	0	5	0	0	1

Figure 49. One Method for Clearing Core Storage

The second instruction starts clearing the specified core-storage area at the high-numbered position. This instruction clears the core-storage area from 36199 to 36100. (At the end of the operation, the B-address register contains the number 36099.)

The third instruction stores the B-address register contents in core storage, starting at the address specified by the C-address register. The C-address register contains the core-storage address that is the units position of the CLEAR STORAGE B-field. After the operation, core-storage positions 00130-00134 contain the number 36099.

The fourth instruction tests core-storage position 00500 for the word mark that was previously put there. If the word mark is still there, the program branches to the specified I-address. The I-address is the core-storage address that contains the CLEAR STORAGE operation code. The next 100 positions of core storage are cleared.

When the last group of 100 core-storage positions is cleared, the word mark is removed from core-storage position 00500. The test instruction is performed, but no branch occurs. This signifies that the clear operation is complete, and the program proceeds with the next sequential instruction.

Word Marks. Word marks are cleared in the area specified.

Timing. T = 4.5 (L + 1 + B).
L = 1 or 6.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	B	xxx00 − 1

Clear Storage and Branch

Instruction Format.

Mnemonic	*Op Code*	*I-address*	*B-address*
CS	$\overset{\text{v}}{/}$	xxxxx	xxxxx

Function. This instruction has the same effect as CLEAR STORAGE except that the next instruction is taken from the I-address. This is an *unconditional branch* instruction because the branching does not depend on any condition.

Word Marks. Word marks are cleared in the storage area specified by the B-address.

Timing. T = 58.5.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	BI	NSI

Halt

Instruction Format.

Mnemonic	*Op Code*
H	$\overset{\vee}{.}$

Function. The machine stops. Pressing the start key starts system operation with the next sequential instruction.

Word Marks. Word marks are not affected. If this is the last instruction in the program, a word mark must be preset in the storage location immediately to the right of the operation code.

Timing. $T = 4.5$.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	Bp

Halt and Branch

Instruction Format.

Mnemonic	*Op Code*	*I-address*
H	$\overset{\vee}{.}$	xxxxx

Function. The machine stops. When the start key is pressed, the program resumes with the instruction located at the I-address. This is an *unconditional branch* instruction, because the branching does not depend on any condition.

Word Marks. Word marks are not affected. If this is the last instruction in the program, a word mark must be preset in the storage location immediately to the right of the HALT AND BRANCH instruction.

Timing. $T = 36$.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	BI	NSI

No Operation

Instruction Format.

Mnemonic	*Op Code*
NOP	$\overset{\vee}{N}$

Function. This operation code can be substituted for the operation code of any instruction to make that instruction ineffective.

Word Marks. Word marks are not affected.

Timing. $T = 4.5\,(L + 1)$.

$L = 1, 2, 3$ .... NO LIMIT

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	Bp

IBM 1415 Console

The IBM 1415 Console (Figure 50) contains the operating and controlling keys, lights, and switches of the IBM 1410 Data Processing System. The console comprises an I/O printer, a control section, and an indicator-light panel.

I/O Printer

The I/O printer on the IBM 1415 Console has the ability to:

1. Provide an operating log of all major manual console I/O printer operations (reset key operations are not logged). All alterations to internal data and their addresses are logged by the I/O printer. Before the alteration is made, the data must be displayed through the use of the console I/O printer.
2. Provide display facilities for some registers and all storage locations.
3. Provide an inquiry mode of operation under control of console operator.
4. Provide messages under program control. A programmed print-out can be overlapped with compute if the overlap option is included.
5. Provide print-out of the instruction-address register, A- and B-address register, Op-register, Op-modifier register, A-data register, B-channel contents, assembly register, and the unit-select and unit-number register for both channels 1 and 2 on manual-stop, programmed-stop, and error-stop operations (Figure 51).

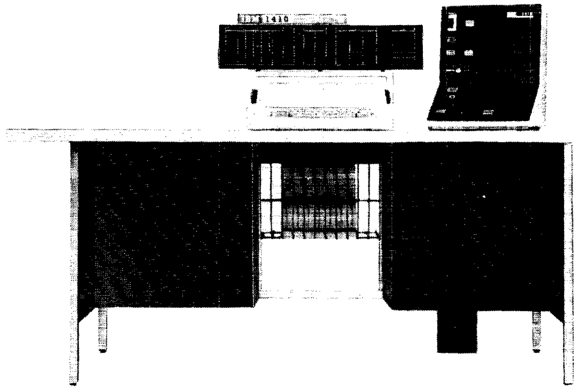


Figure 50. IBM 1415 Console

The console I/O printer (Figure 52) can print 64 characters (10 numerical, 26 alphabetic, 28 special), a word-mark symbol, and an underscore symbol (invalid bit parity print-out). This printer has no type bars or movable carriage. Instead, it has a sphere-shaped element containing the 64 characters. The element moves from left to right across the paper during a printing operation. Maximum speed of the console I/O printer is 932 characters per minute.

Because this unit is used as an I/O printer, the normal functions of carriage return, index, backspace, and tabulation are inoperative from the keyboard.

I/O Printer Control Keys and Levers

INQUIRY KEYS

The IBM 1415 Console can be used as an inquiry station by using the console inquiry keys (Figure 52). The entry of inquiries and the print-out of their replies are under program control, and can occur while the system is operating in either the *run* mode or the *I/E-cycle* mode.

The use of each key is discussed by the order of use during an inquiry request operation.

REQUEST KEY

A console inquiry is initiated by pressing the inquiry-request key. A signal, requesting permission to process a console inquiry (turns on the inquiry-status latch in the 1410) is sent to the central processing unit.

The inquiry request is discovered by the 1410 system when the program tests the inquiry status latch with a TEST AND BRANCH instruction ($\overset{\vee}{\text{J}}$ (IIIII) Q). If the latch is set ON, the program branches to a subroutine that contains the READ CONSOLE PRINTER instruction $\overset{\vee}{\text{M}}$ (% T 0) (BBBBB) R (Figure 53). Acknowledgement of the inquiry request by the 1410 system causes the character *I* to be printed by the console I/O printer. The I/O printer is then impulsed to take a single space. As soon as the space operation is completed, the I/O printer keyboard is unlocked. The manual entry of the inquiry request, character by character, can now proceed. The first inquiry character is placed in the storage address specified by the READ CONSOLE instruction B-address. Subsequent characters are placed in the next higher storage positions.

If a request is made but has not yet been recognized, pressing the cancel key resets the inquiry status latch. If an error is recognized while printing the message,

	PRINTOUT IDENT.	IAR	AAR	BAR	OPERATION CODE OP MODIFIER	A DATA REGISTER B CHANNEL CONTENTS ASSEMBLY CHANNEL	UNIT SEL REG UNIT NUM REG } CH. #1	UNIT SEL REG UNIT NUM REG } CH. #2
NORMAL STOP (Double Space)	S	XXXXX	XXXXX	XXXXX	X X	X X X	X X	X X
HALF-CYCLE (Double Space)	C	XXXXX	XXXXX	XXXXX	X X	X X X	X X	X X
ERROR STOP (Double Space)	E	XXXXX	XXXXX	XXXXX	X X	X X X	X X	X X
ADDRESS SET (Single Space)	B	XXXXX						
STORAGE SCAN SET (Single Space)	#	XXXXX						
DISPLAY (Single Space)	D D	XXXXX XXXXXXXX						
ALTER (Single Space)	A	XXXXXXXX						
CONSOLE INQUIRY (Single Space)	I	XXXXXXXX						
CONSOLE REPLY (Single Space)	R	* XXXX<u>X</u>XXX						

* ____ Indicated Invalid Character (Underlined)

Figure 51. IBM 1415, Printing Layout

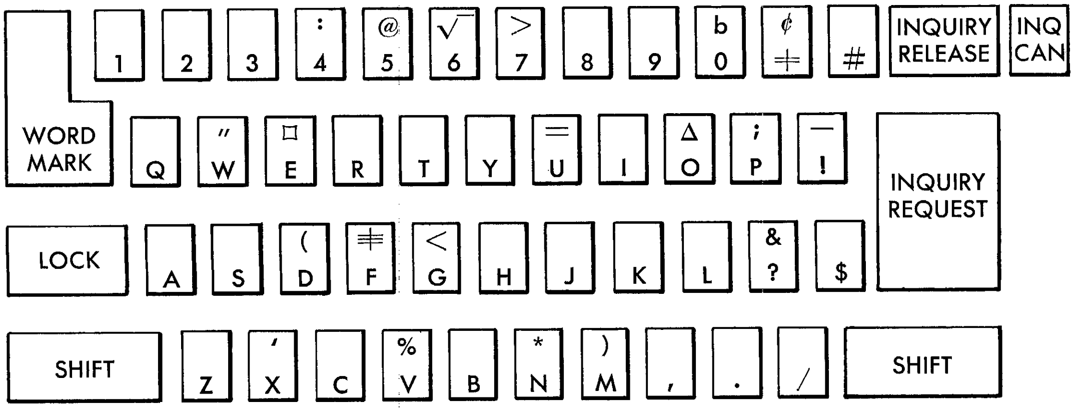


Figure 52. I/O Printer Keyboard

I/O UNIT	X-CTRL FIELD	DESCRIPTION	MNE-MONIC	d-CHARACTER		OPERATION	NOTES
				CHAR-ACTER	CONTROL		
Console I/O Printer	% T0	Read from console I/O printer	RCP	R	Read	Transfer data direct from the console I/O printer to storage	1. Data transfer is operator-controlled. See console operating features. 2. Word marks in storage are undisturbed in move. 3. Word marks in storage are erased and entered during a load operation.
		Write on console I/O printer	WCP	W	Write	Transfer data direct from storage to the console I/O printer and print	1. Group-mark, word-mark is not printed with message. 2. Positions containing word marks are indicated in output line only in a load operation.

Figure 53. Console I/O Printer Control Instructions

pressing the cancel key sets the condition I/O channel-status indicator ON, and the program continues. Figure 54 shows the conditions that set the I/O channel-status indicators and that turn ON their associated lights during an I/O printer read operation.

Error Condition. Any system error stops the system and initiates an error print-out operation, and the inquiry request operation is ended.

RELEASE KEY

The inquiry is released to the processing unit by pressing the release key after the correct number of characters have entered storage. The programmer has already specified the length of the inquiry (a certain number of characters that occupy specific storage locations). The next higher storage location must contain a previously-inserted group-mark with a word-mark over it. As the last inquiry character is entered in its storage location, the addressing circuitry is set up to read out the group-mark with a word-mark. The operator must press the release key at this time to obtain a correct-length record, and to insure the processing of the inquiry.

INDICATOR	d-CHARACTER BIT	CONDITION
Not Ready	1	Never set
Busy	2	Never set
Data Check	4	Processing unit detects input character validity error
Condition	8	Cancel Key operated during inquiry
Wrong Length Record	B	Wrong length record
No Transfer	A	No message request — Cancel Key operated before inquiry

Figure 54. I/O Channel Status Indicators Set During IBM 1415 Read Operation

If it is desirable to request a second inquiry while entering a first inquiry, this can be accomplished by holding down the inquiry-request key while pressing the inquiry release key to release the first inquiry. This causes the inquiry-status latch to remain ON. After initiating a request, the inquiry-status latch can be reset before any characters are entered, by pressing the release key.

Operating the release key also initiates a carriage-return and line-space operation, and locks the keyboard.

Wrong-Length Record (Inquiry). Operating the release key, when the number of characters printed is less than the prescribed format on an inquiry request, causes:

1. a carriage-return and line-space operation
2. the wrong-length record I/O channel status indicator to be set ON
3. the program to go to the next instruction.

When the number of characters printed is more than the prescribed format on an inquiry request, it causes:

1. the I/O printer to lock-up and an immediate carriage-return and line-space operation. The last character that caused the lock-up prints, but does not enter storage.

2. the wrong-length record I/O channel status indicator to be set ON. The operator must then press the release key or the cancel key to continue programming.

CANCEL KEY

Operating the cancel key terminates the inquiry routine in process at that time. Operating the cancel key during the inquiry-request-message printing sets the condition I/O channel-status indicator ON, releases the system, initiates a carriage-return and line-space operation, and allows the normal program to resume.

If it is desirable to request a second inquiry while cancelling a first inquiry, this can be done by holding down the inquiry-request key while pressing the cancel key to cancel the first inquiry. This causes the inquiry-status latch to remain ON.

After initiating a request, the inquiry-status latch can be reset off, by pressing the cancel key, before any characters are entered.

WORD-MARK KEY

Pressing this key prints a word mark and backspaces the carriage after printing. Pressing a character key prints the character under the word mark, and enters both the word mark and the character into storage. The word-mark key must be pressed first when entering a character with a word mark into storage.

SHIFT KEYS

Pressing either one of the two shift keys shifts the printer into upper case. Figure 52 illustrates the I/O printer keyboard. The characters shown at the top of the keys are upper-case characters and require a shift key to be operated before pressing the character key. The printer automatically returns to lower-case shift when the key is released.

LOCK KEY

Pressing this key activates the shift keys, and locks the printer in upper-case shift until released by pressing one of the shift keys.

COPY-CONTROL LEVER

Operating the copy-control lever (located at the left end of the carriage) positions the carriage forward or backward so that various thicknesses of printing material are accommodated. The copy-control lever can be set in five different positions. Moving the lever forward decreases the distance between the platen and printing mechanism, and moving the lever to the rear increases this distance.

PAPER-RELEASE LEVER

Pulling forward on the paper-release lever (the inner lever located at the right end of the carriage) releases the pressure of the front and rear feed rolls from the platen. This permits more accurate paper positioning and easier paper removal. This lever should be left in the forward position when the pin-feed platen is used. It should only be pushed back when it is desired to move the paper backwards through the platen.

MARGIN-SET LEVERS

The left and right margins are determined by the position of the margin stops on the margin rack. The left or right margin is set by operating the associated margin-set lever (located at the rear of the keyboard). The margin-set lever is operated by exerting pressure toward the rear of the printer and sliding the lever to the right or left.

INDEX-SELECTOR LEVER

When the index-selector lever (the outer lever located at the right end of the carriage), is set toward the rear, the platen double spaces for each line of printing (three lines per inch). With the index-selector lever set toward the front of the machine, the platen single spaces for each line of printing (six lines per inch).

CONSOLE REPLY ROUTINE

A reply routine or programmed print-out can occur at any time. The CONSOLE WRITE instruction, $\overset{\vee}{\mathrm{M}}/\overset{\vee}{\mathrm{L}}$ (% T 0) (BBBBB) W causes:

1. the character R to print
2. a single carriage-space operation
3. data to be transferred from storage and printed by the console I/O printer until a group-mark with a word-mark is sensed in storage.

A valid blank in storage causes the I/O printer to space.

4. a carriage return and line space operation
5. the program to continue with the next instruction.

Figure 55 shows the conditions that set the I/O channel-status indicators ON and that turn ON their associated lights during an I/O printer write operation.

INDICATOR	d-CHARACTER BIT	CONDITION
Not Ready	1	Never set
Busy	2	Carriage returning
Data Check	4	I/O Printer detects output character validity error
Condition	8	Never set
Wrong Length Record	B	Never set
No Transfer	A	Never set

Figure 55. I/O Channel Status Indicators Set During IBM 1415 Write Operation

CPU Processing Error. A CPU processing error during the data transfer ends the reply routine and initiates an error print-out operation. Operation of the start key is necessary to complete the programmed print-out.

I/O Printer Error. If a parity error is sensed in the I/O printer, the error character is printed and underlined. The data-check I/O channel status indicator is also set ON. The reply routine continues until a group-mark with a word-mark is sensed in storage.

CONSOLE LOAD READ AND WRITE OPERATIONS

If the console I/O printer is addressed on a *load-write* operation ($\check{\text{L}}$ Op code), blank characters in storage are printed as small *b*'s, and word marks are printed as inverted circumflexes over the character associated with the word mark.

It is possible to enter word marks into storage during a console-inquiry routine if the instruction calls for a *load-read* operation. The word mark prints on the log sheet and enters storage. A printer space operation generates a blank character in storage.

Control Section

The control section (Figure 56 and Figure 57) contains the power keys and lights and other keys and switches that control the 1410.

Power Keys, Lights, and Switches

The power keys, lights, and switches (Figure 56) control the application of power to the 1410.

EMERGENCY POWER OFF

This pull switch should be used *only* in case of emergency, when all power must be shut off immediately to prevent injury to an individual or damage to the machine. If this switch is used, only a customer engineer should turn on the power again.

Pulling the switch removes all power from all the units.

POWER ON

Operating this switch normally provides full operating power to the 1410 system, either from a power-off or a dc-off condition. (Power is applied to the 1414, the tape adapter unit, and the disk-storage control unit only if their respective CE panel power local-remote switches are set to REMOTE.) Also, a *power-on reset* operation is initiated whenever power is initially applied or reapplied to the system, and causes all system registers, latches, rings, etc., to be reset. The initial operation of the switch starts the internal sequencing of power to the system and turns on the illuminated portion of the power-on key. The key light remains ON until the emergency power-off switch or power-off key is operated.

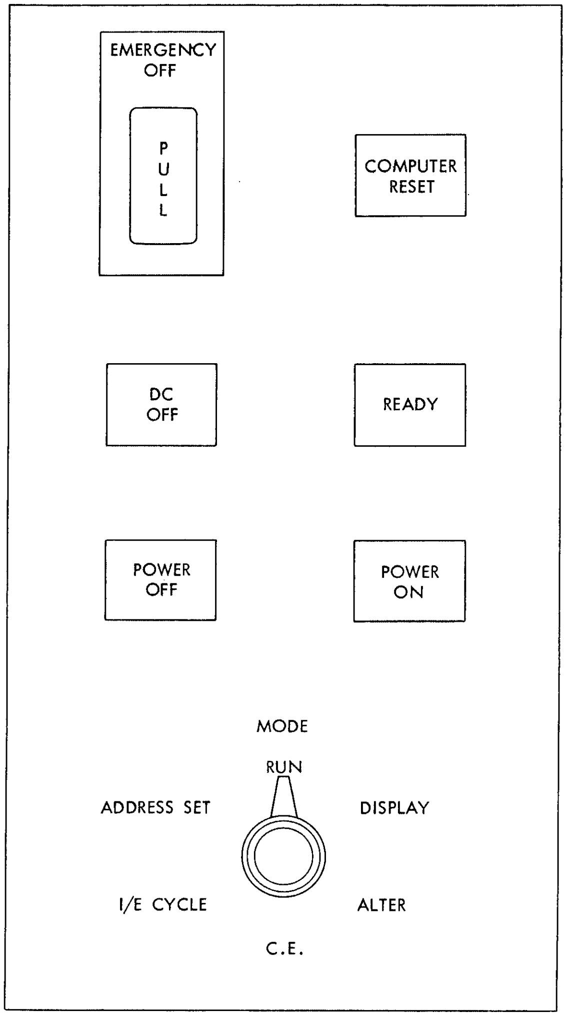


Figure 56. Power Keys, Lights, and Switches

When the system reaches full operating power, the ready light is turned on.

POWER OFF

Pressing the power-off key removes all power from the system, except those units with CE panel power local-remote switch set to LOCAL. The removal of system power also turns off the *ready light* and the illuminated portion of the power-on key.

To restore full operating power to the system, the power-on key must be operated.

DC OFF

Pressing this key turns off the system dc power only, except in those units with CE panel power local-remote switch set to LOCAL. The key is used when the machine will be idle for a short time. Operating the key turns off the *ready light,* but the light comes ON again as soon as full power is restored to the system. Power-on light remains ON.

COMPUTER RESET

Operating this key resets the check circuits, resets the program to 00001, resets all timing clocks, and resets all machine indicators (that is, overflow latches, compare triggers, etc.). The inquiry latches (except the console-inquiry latch) and the tape-density latch are not reset.

READY LIGHT

The ready light is turned on when full operating power is applied to the system. It takes a short time for the machine to reach ready status because the power is supplied to the various system components in a specified sequence. The ready light turns ON immediately if the power-on key is pressed while the machine is in the DC-OFF mode.

Mode Switch

The six modes of machine operation are selected by the *mode switch.* The six modes are modified by the CE controls. Usually, these CE controls are set to the normal or off operating mode. Anytime the mode switch setting is changed, it initiates a stop print-out operation as soon as the execution of the previous setting is complete.

CE (CUSTOMER ENGINEER)

When the mode switch is set to CE, the customer engineering function of storage scan is available for use.

I/E CYCLE

With the mode switch set to I/E CYCLE, the first operation of the start key causes the machine to read one complete instruction from storage, then stop and print out. Because this print-out occurs while the machine is in the I/E CYCLE mode, the print-out is preceded by the printing of a C. The carriage then spaces and prints out the contents of the instruction-address register, A- and B-address registers, Op-register, Op-modifier register, A-data register, B-channel contents, assembly-channel output, and the unit-select and unit-number registers for channel 1 and channel 2.

The second operation of the start key causes the execution of that instruction (called the execution phase), and then machine operation stops. Another C print-out operation occurs exactly as previously described.

Subsequent operation of the start key results in the machine's going through alternate instruction and execution cycles.

ADDRESS SET

This setting of the mode switch is used to start a program at a specific place in storage. Operation of the stop key or turning the mode switch to the ADDRESS SET position causes a normal stop print-out operation.

The start key is then pressed, and a B-character is printed on the console I/O printer. The carriage then takes a single space operation. The address that is now printed by operating the I/O printer keys enters the instruction-address register and is followed by an automatic carriage return and line space (index) operation.

The mode switch may then be positioned at either the RUN setting or the I/E CYCLE setting. Pressing the start key starts the program with the instruction located at the printed address.

This switch setting, when used with the address-entry switch on the console CE panel, permits altering the contents of A-, B-, C-, D-, E-, or F-register, depending on the setting of the switch. With this switch in the NORMAL position, the contents of the instruction-address register are altered. If any address register is altered other than the instruction-address register, the address-entry switch must be returned to the NORMAL position before pressing the start key. Pressing the start key starts the program at the unaltered address in the instruction-address register.

RUN

When the mode switch is set to RUN, pressing the start key causes the system to run continuously under control of the stored program.

DISPLAY

Any portion of storage may be displayed on the console I/O printer log sheet by using the DISPLAY setting of the mode switch. The display may be of any length, from one field to a multiple line print-out.

Operation. During a display operation, this sequence of events takes place:

1. The system is stopped by operating the *stop key* or setting the mode switch to DISPLAY.

2. With the mode switch set to DISPLAY, operating the start key results in printing a character D followed by a single carriage-space operation.

3. The high-order address of the field to be displayed is manually printed on the console I/O printer by operating the appropriate keys.

4. An automatic carriage-return and index operation takes place following the printing of the fifth address character. *Note:* If an extra key is pressed when keying in the address, the extra character overrides the automatic carriage return and leaves the

carriage on the same line. The next print-out takes place on the same line with no error indication or machine stop.

5. A character D is automatically printed, followed by a single-space operation.
6. The contents of storage, starting at the high-order position previously printed, are printed until a word mark is recognized. The character and its associated word mark are printed (first character of the adjacent field).

The adjacent field can be displayed if the start key is pressed again. A continuous display results from holding the start key in its operated position. The display operation can be ended at any time by pressing the stop key.

The display operation is momentarily held up if an end-of-printing-line signal is encountered. An automatic carriage return and index operation takes place, followed by a resumption of the display operation. If the last character in storage is printed during a display operation, internal circuitry ends the display and causes a carriage return.

Usually, a display operation must precede an alter operation. If the mode switch is reset to the RUN or I/E CYCLE setting, operation of the start key allows the program to proceed, beginning with the next sequential instruction.

Error Conditions. If a system error occurs during a display operation, an error print-out occurs. Channel errors are ignored; address errors cause a stop. The carriage returns when the stop key is pressed or the error is reset.

ALTER

By using the ALTER setting of the mode switch, in combination with the console I/O printer, it is possible to alter the information in any storage location. However, a display operation of the specific storage location must be completed before an alter operation can be performed. This display operation prerequisite insures having a record of the storage location contents before the alteration takes place.

Upon completion of the display operation, the alter operation is initiated by rotating the mode switch from DISPLAY to ALTER and pressing the start key. The character A is printed, signifying an alter operation, followed by a carriage space operation and the unlocking of the keyboard. Unlocking the keyboard allows the manual-alter printing operation to proceed.

If one or more fields (but less than a full line) were previously displayed, only the first displayed field can be altered. Only the first line from a multiple-line display can be altered.

The correct characters are printed and replace the previously-displayed incorrect data. Correct data are kept by reprinting all the correct characters. Any previously-displayed word mark must be re-entered into storage. Valid blanks are entered in storage by operating the space bar, or the blank-character key (b).

The alter operation continues until a word mark is sensed if one or more fields (but less than one line) were displayed. An alter operation ends when the end-of-line condition is sensed if a multi-line display preceded the alter operation. Either one of these conditions locks the keyboard and initiates a carriage return and index operation. When a character is entered into the last location of storage during an alter routine, internal circuitry terminates the alter routine and causes a carriage return.

If an error other than a data error occurs, the alter routine terminates and the carriage returns.

Control Keys

Control keys (Figure 57) include start, stop, and program reset.

START KEY

With the mode switch set to RUN, the operation of the start key causes the system to begin executing instructions and resets the parity-check circuits. Also, the start key initiates the operation when the mode switch is set to I/E CYCLE or DISPLAY or ALTER. It is also active in some customer engineering operating modes.

STOP KEY

Operation of the stop key, while the program is running, stops the program after executing the current instruction. The S character is printed, followed by a single carriage-space operation. Following the space operation, the contents of the instruction-address register, A- and B-address register, Op-register, Op-modifier register, A-data register, B-channel contents, assembly-output register, and the unit-select and unit-number registers for channels 1 and 2 are printed out.

PROGRAM-RESET KEY

Operating this key resets the check circuits, resets the program to 00001, and resets the A- and B-data registers, Op-register, Op-modifier register, and the console-inquiry latch.

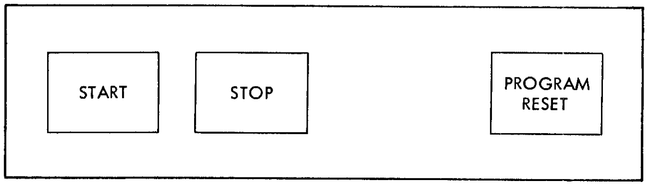


Figure 57. Control Keys

Indicator Light Panel

CPU Control Indicator Lights

Refer to Figure 58.

I RING

These lights indicate the 13 steps of the instruction ring (OP and 1-12).

A RING

These lights indicate the 6 steps of the A ring (1-6).

CLOCK

These lights indicate the 10 steps of the main clock (A-K).

SCAN

These lights indicate what type of address modification is taking place.

The N light indicates that a storage location is operating in a + 0 modification cycle.

The 1 light indicates that the CPU is operating in a — 1 address-modification cycle.

The 2 light indicates that the CPU is operating in a + 1 address-modification cycle.

The 3 light indicates that storage is being readdressed and the CPU is operating in a — 1 address-modification cycle.

SUB SCAN

These lights indicate what portion of a field is being addressed during arithmetic operation and certain other system executions.

U (Units). This light indicates that the units position of the field is being addressed.

B (Body). This light indicates that the body of the field (excluding units position of the field) is being addressed.

E (Extension). This light indicates that the extension portion of the field is being addressed.

MQ (Multiplier-Quotient). This light indicates that the multiplier or quotient is being addressed during a multiply or divide operation. It is also used to indicate special conditions during an edit operation.

CENTRAL PROCESSING UNIT						
I RING		A RING	CLOCK	SCAN	CYCLE	ARITH
OP		1	A	N	A	
1	6	2	B	1	B	CARRY IN
2	7	3	C	2	C	CARRY OUT
3	8	4	D	3	D	
4	9	5	E		E	A COMPL
5	10	6	F	SUB SCAN	F	B COMPL
	11		G	U		
	12		H	B	I	
			J	E	X	
			K	MQ		

Figure 58. CPU Control Indicator Lights

CYCLE

These lights indicate the 8 types of cycles in which the CPU can operate (A, B, C, D, E, F, I, X).

ARITHMETIC LIGHTS

Carry In. This light indicates that the carry latch has been set ON.

Carry Out. This light indicates that the adder has a carry output.

A Compl (A Complement). This light indicates that channel A data are being complemented.

B Compl (B Complement). This light indicates that channel B data are being complemented.

Status Lights

Refer to Figure 59.

B < A (LOW)

This light indicates that the B-field is less than the A-field. A computer-reset operation or a power-on reset operation turns the light ON. The light remains ON until the condition is reset by a stored-program operation.

B = A (EQUAL)

This light indicates that the B-field is equal to the A-field. The light remains ON until the condition is reset by a stored-program operation, a computer-reset operation, or a power-on reset operation.

B > A (HIGH)

This light indicates that the B-field is greater than the A-field. The light remains ON until the condition is reset by a stored-program operation, a computer-reset operation, or a power-on reset operation.

STATUS
B>A
B=A
B<A
OVERFLOW
DIVIDE OVERFLOW
ZERO BALANCE

Figure 59. Status Indicator Lights

OVERFLOW

This light indicates that an arithmetic-overflow condition has been detected. The overflow condition can be detected only during an add or subtract operation, but not during a zero and add, zero and subtract, multiply or divide operation. The light remains ON until the condition is reset OFF by a stored-program test operation, a computer-reset operation, or a power-on reset operation.

DIVIDE OVERFLOW

This light indicates the occurrence of a divide-overflow condition. The light remains ON until the condition is reset OFF by a stored-program test operation, a computer-reset operation, or a power-on reset operation.

ZERO BALANCE

When ON, this light indicates the occurrence of a zero-balance condition. It is set by the result (which is zero) of any add, subtract, zero and add, zero and subtract, or multiply operation. The light remains ON until the condition is reset by the computer reset, or power-on reset. It is also reset by the result (which is not zero) of any add, subtract, zero and add, zero and subtract, or multiply operation.

I/O Channel Control

Refer to Figure 60.

There are two sets of I/O channel control lights. One set indicates channel 1; the other set indicates channel 2, if channel 2 is present. The description applies to both channels.

INTERLOCK

This light indicates that either an I/O read or write operation has been called for. The light is turned off when the status test is satisfied following a read or write operation. The status test is satisfied if either:

1. a BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction $\overset{\vee}{R}$ (I) ⋕ is given before encountering the next I/O unit instruction on the same channel, or;
2. a specific $\overset{\vee}{R}$ (I) d instruction (see Figure 31) is given, which results in a branch before encountering the next I/O unit instruction.

If the status test is not satisfied before the next I/O instruction for that particular channel is called for, the system is interlocked and the interlock light remains ON.

RBC INTERLOCK (READ BACK CHECK INTERLOCK)

This light indicates that the system has completed a successful write operation, but has not called for a read-back check (write-disk check) operation.

I/O CHANNEL CONTROL	
CH 1	CH 2
INTERLOCK	INTERLOCK
RBC INTERLOCK	RBC INTERLOCK
READ	READ
WRITE	WRITE
OVERLAP IN PROCESS	OVERLAP IN PROCESS
NOT OVERLAP IN PROCESS	NOT OVERLAP IN PROCESS

Figure 60. I/O Channel-Control Indicator Lights

READ

This light indicates that an I/O read operation has been called for. The light remains ON until the next I/O operation.

WRITE

This light indicates that an I/O write operation has been called for. The light remains ON until the next I/O operation.

OVERLAP IN PROCESS

This light is turned on at the beginning of any I/O operation that is performed in the OVERLAP mode. If the system stops because of an error during the I/O operation, the light remains ON to indicate what type of I/O operation was in process when the error occurred. The light turns off at the end of the data transfer, when no error occurs.

NOT OVERLAP IN PROCESS

This light turns on at the beginning of any I/O operation that is *not* performed in the OVERLAP mode. It is turned off at the end of the data transfer. The light signifies what type of I/O operation was in process when the system stopped because of an error.

I/O Channel Status Indicator Lights

Refer to Figure 61.

The I/O channel-status indicator lights indicate the setting of their associated indicators. The indicators were set as a result of the last I/O operation on that particular I/O unit. Whenever the *not ready, busy, data check, condition, wrong-length record, or no transfer* I/O channel status indicator is set ON, the corresponding indicator light is also turned ON. One set of indicator lights is associated with channel 1; another set of identical lights is available for use with the channel

I/O CHANNEL STATUS	
CH 1	CH 2
NOT READY	NOT READY
BUSY	BUSY
DATA CHECK	DATA CHECK
CONDITION	CONDITION
WRONG LENGTH RECORD	WRONG LENGTH RECORD
NO TRANSFER	NO TRANSFER

Figure 61. I/O Channel-Status Indicator Lights

2 special feature. Figures 54 and 55 show the conditions that set the indicators ON and turn ON their associated lights during an I/O printer read or write operation.

NOT READY

The not ready light indicates that one of the input or output units on that channel is not capable of taking a cycle. Refer to the individual I/O unit write-up for the specific conditions that turn on the not ready light.

BUSY

The busy light indicates that one of the input or output units on that channel has not completed a previous operation. Refer to the individual I/O unit write-up for the specific conditions that turn on the busy light.

DATA CHECK

The data check light, when ON, indicates that one of the input or output units on that channel has detected a data parity condition. Refer to the individual I/O unit write-up for the specific conditions that turn on the data check light.

CONDITION

The condition light indicates that one of the input or output units on that channel has encountered an end-of-file condition or a data-transfer control error condition relating to that unit. Refer to the individual I/O unit write-up for the specific conditions that turn on the condition light.

WRONG-LENGTH RECORD

The wrong-length record light indicates that one of the input or output units on that channel has encountered or sent a wrong-length record. Refer to the individual I/O unit write-up for the specific conditions that turn on the wrong-length record light.

NO TRANSFER

The no-transfer light indicates that an operation of one of the input or output units on that channel has resulted in a no-transfer condition. Refer to the individual I/O unit write-up for the specific conditions that turn on the no-transfer light.

Systems Check Indicator Lights

Refer to Figure 62.

PROCESS

A-Channel. This light indicates that an A-channel parity error has been detected.

B-Channel. This light indicates that a B-channel error has been detected.

Assembly Channel. This light indicates an error at the assembly output or an error when merging zones, numerical information, and word marks during any operation.

Address Channel. This light indicates that a validity error has been detected on the channel that supplies data to the various address registers.

Address Exit. This light is only active during an indexing or store address-register operation, and indicates that a validity error has been detected at the address-register exit channel.

A-Register Set. This light indicates that the A-data register has failed to reset.

B-Register Set. This light indicates that the B-data register has failed to reset.

Op-Register Set. This light indicates that the Op-register has failed to set.

Op-Modifier Set. This light indicates that the Op-modified register has failed to set.

SYSTEM CHECK		
PROCESS		PROGRAM
A CHANNEL	A REGISTER SET	I/O INTERLOCK
B CHANNEL	B REGISTER SET	ADDRESS CHECK
ASSEMBLY CHANNEL	OP REGISTER SET	RBC INTERLOCK
ADDRESS CHANNEL	OP MODIFIER SET	INSTRUCTION CHECK
ADDRESS EXIT	A CHARACTER SELECT	
	B CHARACTER SELECT	

Figure 62. System-Check Indicator Lights

A-Character Select. This light indicates that no character is, or extra characters are, gated on the A-channel.

B-Character Select. This light indicates that a malfunction in the storage-character selection and regeneration circuitries has been detected.

PROGRAM LIGHTS

I/O Interlock. This light indicates that the program has failed to test the I/O channel status-indicator prior to the next I/O instruction on that channel.

Address Check. This light indicates that an improper storage address has been given by the programmer, or that an operation goes beyond the capacity of core storage.

RBC Interlock (Read-Back Check Interlock). This light indicates that the read-back check (write-disk check) operation had not been completed before another operation of that disk-storage channel was called for.

Instruction Check. This light indicates that an improper instruction has been given by the programmer.

Power Indicator Lights

Refer to Figure 63.

I/O OFF-LINE

This light indicates:

1. One or all the synchronizer areas are operating in an off-line operation (for CE use only).
2. I/O synchronizer power is shut down (CE panel local-remote switch on LOCAL and the synchronizer power-off button is pressed).

THERMAL

When the internal temperature of the system has exceeded the allowable limit or a blower circuit breaker has tripped, the power turns off and the light turns on.

CB TRIP (CIRCUIT-BREAKER TRIP)

When one of the circuit breakers in the system has tripped, all dc power turns off and the light turns on.

TAPE OFF-LINE

This light indicates:

1. That either one, or both, tape-transmission channels is operating off-line (CE use only).
2. That power is shut down under local control for either one, or both, tape-transmission channels.

POWER	
THERMAL	I/O OFF LINE
CB TRIP	TAPE OFF LINE
	DISK OFF LINE

Figure 63. Power Indicator Lights

DISK OFF-LINE

This light indicates:

1. That either one, or both, disk-transmission channels is operating off-line (CE use only).
2. That power is shut down under local control for either one, or both, disk-transmission channels.

System Controls Indicator Lights

Refer to Figure 64.

1401 COMPATIBILITY LIGHT

This light indicates that the system is in the 1401 mode of operation (capable of running IBM 1401 programs). It is turned on when the *compatibility* switch is in the 1401 (ON) position.

OFF NORMAL LIGHT

This light indicates that certain CE switches on the console are not in the proper position for normal operation. The light is ON if:

1. Print-out-control switch is set to INHIBITED.
2. Asterisk-insert switch is set OFF.
3. Cycle-control switch is *not* set OFF.
4. Check-control switch is *not* set to STOP NORMAL.
5. Storage-scan switch is *not* set OFF.
6. Address-entry switch is *not* set to NORMAL.

STOP

This light, when ON, indicates that the system has stopped and that operator intervention is required to start a new operation.

PRIORITY ALERT

This light, when ON, indicates that the machine is operating in the PRIORITY ALERT mode and is ready for an interruption (priority feature must be installed).

SYSTEM CONTROLS	
1401 COMPAT	OFF NORMAL
PRIORITY ALERT	STOP

Figure 64. System-Controls Indicator Lights

IBM 1415 Console CE Test Panel

The IBM 1415 Console customer engineering test panel (Figure 65) is provided primarily for customer engineers when diagnostic testing and performing preventive maintenance routines. However, certain functions of the panel can be used advantageously by customer personnel when checking new program routines. Only the switches and functions, of use to the customer, are described in this section of the manual.

Compatibility Controls

COMPATIBILITY SWITCH

This switch (Figure 65), when set to the 1401 setting, makes it possible to run IBM 1401 programs on the 1410. Usually, the switch is set to the 1410 setting.

I/O CHECK STOP SWITCH

This switch is operative *only* when the 1410 is operating in 1401 mode. The I/O check stop switch, when set to the ON position, stops programming at the completion of an I/O operation if the error occurs during that operation. Error conditions that can cause this are: hole-count check in the card reader or card punch, validity error in the card reader, print check, or any one of a number of control errors.

The stored program controls the system in the event of an I/O error when this switch is in the OFF position.

1401 I/O CHECK-RESET SWITCH

This switch is operative only when the 1410 is in 1401 mode and is used in conjunction with the I/O check

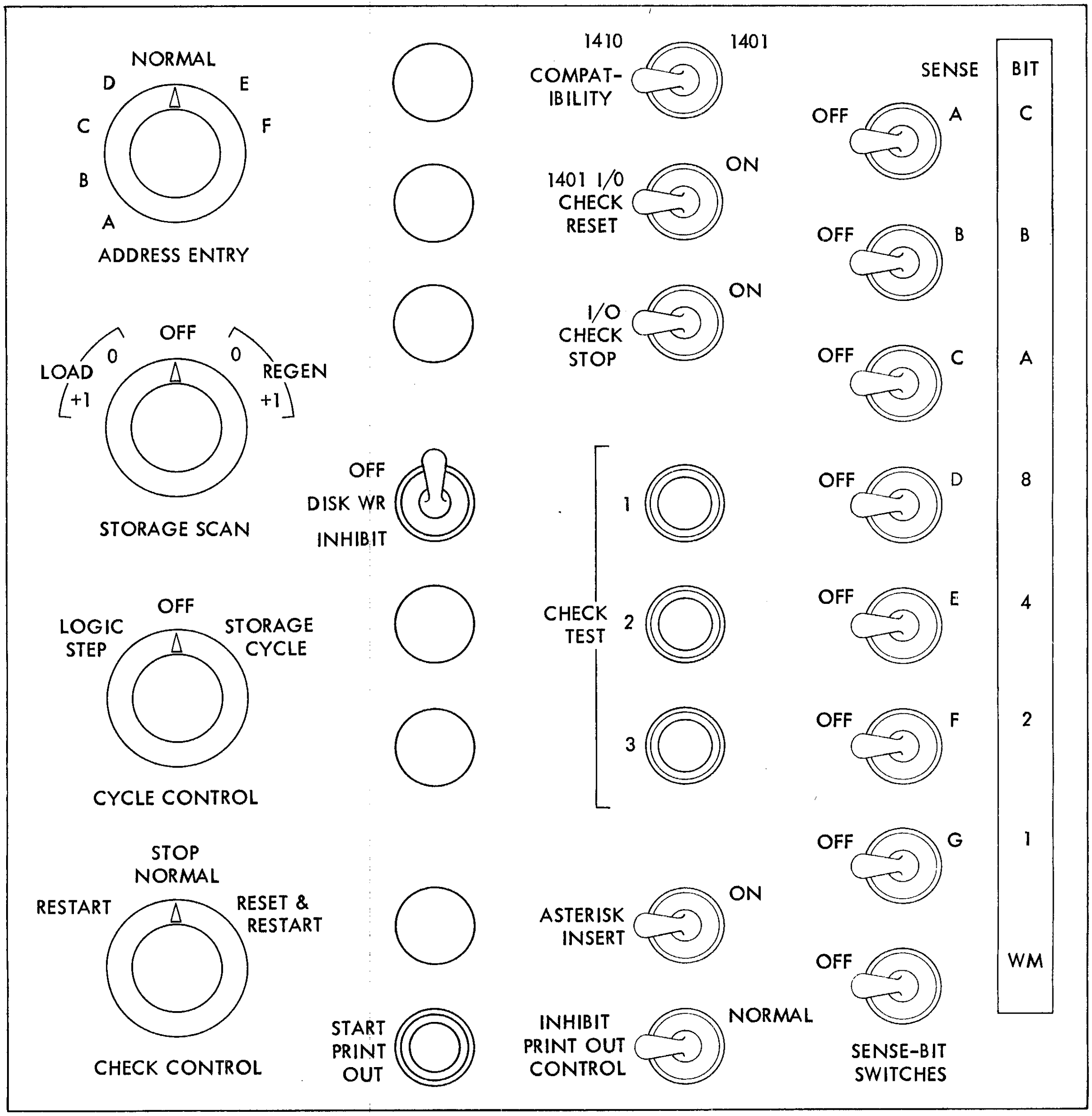


Figure 65. Compatibility and C.E. Controls

stop switch. Operating this switch resets any I/O unit error conditions that are sensed when the I/O check stop switch is set OFF. (The switch is primarily used by customer engineers for diagnostic testing.)

SENSE-BIT SWITCHES

Sense switches are tested by the program. When ON, they can cause a change (branch) in program operation.

The sense-bit switches are active as sense switches (A-G) when operating in the 1401 mode.

CHECK-CONTROL SWITCH

The check-control switch is a 3-position rotary switch. When it is set to STOP NORMAL, any CPU error or input parity error, with the asterisk-insert switch set OFF, results in an immediate stop and an error print-out operation. For normal operation, this switch is set to the STOP NORMAL position, with the asterisk-insert switch ON.

When set to RESTART, any of the previously mentioned errors also result in an immediate stop. Following the error print-out operation, the program is restarted automatically. If the error print-out is bypassed (print-out-control switch), the program is restarted immediately following the stop.

When set to RESET AND RESTART, any of the previously mentioned errors also results in an immediate stop in the same manner as the RESTART setting. An error print-out operation is followed by a computer-reset operation. When the computer-reset operation is completed, the program is restarted. If the error print-out is bypassed (print-out control switch), computer-reset and the program-start follow the stop.

PRINT-OUT CONTROL SWITCH

This toggle switch controls all stop print-out operations, including error print-out. When set to NORMAL, the print-out takes place. When set to INHIBITED, the print-out does not take place.

START PRINT-OUT SWITCH

This switch is used when the program routine fails to advance, and a stop print-out operation cannot be initiated by pressing the stop key. Operating this switch initiates a stop print-out operation. Printing the contents of the various registers aids in determining the cause of failure. This switch can also be used to initiate occasional print-outs while single cycling with print-out control OFF.

ASTERISK-INSERT SWITCH

This toggle switch, when ON, converts any input unit character of incorrect parity to an asterisk, and enters it into storage in place of the invalid character. When the toggle switch is set to the OFF position, a wrong-parity character from any input unit stops the operation and initiates an error print-out operation, unless inhibited by the print-out control switch. If the check-control switch is set to STOP NORMAL, the data transfer stops. If the check-control switch is set to RESTART, the full record can be entered into storage and used for diagnostic purposes or for the reconstruction of the incorrect record.

CYCLE-CONTROL SWITCH

This is a rotary 3-position switch that is used in conjunction with any setting of the mode switch.

When the cycle-control switch is set to OFF, system operation is not controlled by this switch.

When the cycle-control switch is set to STORAGE CYCLE, pressing the start key advances the program by single storage cycles. A print-out operation as described in the I/E CYCLE mode switch setting, occurs at the end of each cycle, unless inhibited.

When the cycle-control switch is set to LOGIC STEP, pressing the start key advances the program by single logic steps.

ADDRESS-ENTRY SWITCH

This is a 7-position rotary switch (A, B, C, D, E, F, and NORMAL). This switch enables a console-printed address to enter the selected address register (A, B, C, D, E, F, or IAR if the switch is set to the NORMAL position). To activate this switch, the console mode switch must be positioned to the ADDRESS SET setting.

For normal system operation, the switch must be set to NORMAL.

DISK-WRITE SWITCH

The disk-write switch (Figure 65) facilitates testing programs on an IBM RAMAC®-oriented 1410 system. It prevents writing test data on permanent records in disk storage. When this switch is set to the OFF position, normal disk-storage operations can be performed.

When the switch is set to the INHIBIT position, all disk-storage instructions, with the exception of WRITE DISK and WRITE DISK WITH WORD MARKS, are performed normally. When these two instructions are encountered, data are transferred from core storage to disk storage, parity and record length are checked; however, no data are written on the surface of the disk. Automatic comparison of the record address in core storage and the address on the disk record is performed however, and the unequal-address compare indicator turns ON if an unequal condition occurs.

A WRITE-DISK-CHECK instruction must follow the write operation, which results in an error condition because no data was written on the disk.

The various types of IBM 1410 Data Processing Systems are:

1. Card oriented
2. Tape oriented
3. RAMAC oriented; or
4. Combinations of preceding.

Card-Oriented IBM 1410 System

Components that are present in every card-oriented system are an IBM 1411 Processing Unit, an IBM 1414 Input-Output Synchronizer, and the IBM 1415 Console.

Input-output units that can be added to these components to form a card-oriented 1410 system (Figure 66) are the IBM 1402 Card Read-Punch, Model 2, and the IBM 1403 Printer (if printed results are desired).

IBM 1402 Card Read-Punch, Model 2

The card read-punch used with the IBM 1410 Data Processing System is a modified 1402-2 (Figure 67). The card reader enters data into the system through an 80-position read synchronizer, and the card punch receives data from the system through an 80-position punch synchronizer (these two synchronizers are located in an IBM 1414 Input-Output Synchronizer).

The card reader has a rated speed of 800 cards per minute (actual card speed realized is governed by the stored-program instructions). The card reader is equipped with a large-capacity, card-loading device, called a *file feed.* With the file feed device, the read feed can be loaded with as many as 3,000 cards.

Cards pass through the 1402-2 read feed face down, 9-edge first, from right to left, past two sets of reading brushes and a stacker-selector station (Figure 68). The read-check brushes read the card to establish a hole-count check. The read brushes also read the entire card for a hole-count check (comparison of the same card as read by the read-check brushes and the read brushes), and direct the data into the read synchronizer for later transmission to storage.

The card punch has a rated speed of 250 cards per minute (actual card speed realized is governed by the stored-program instructions). Cards pass through the

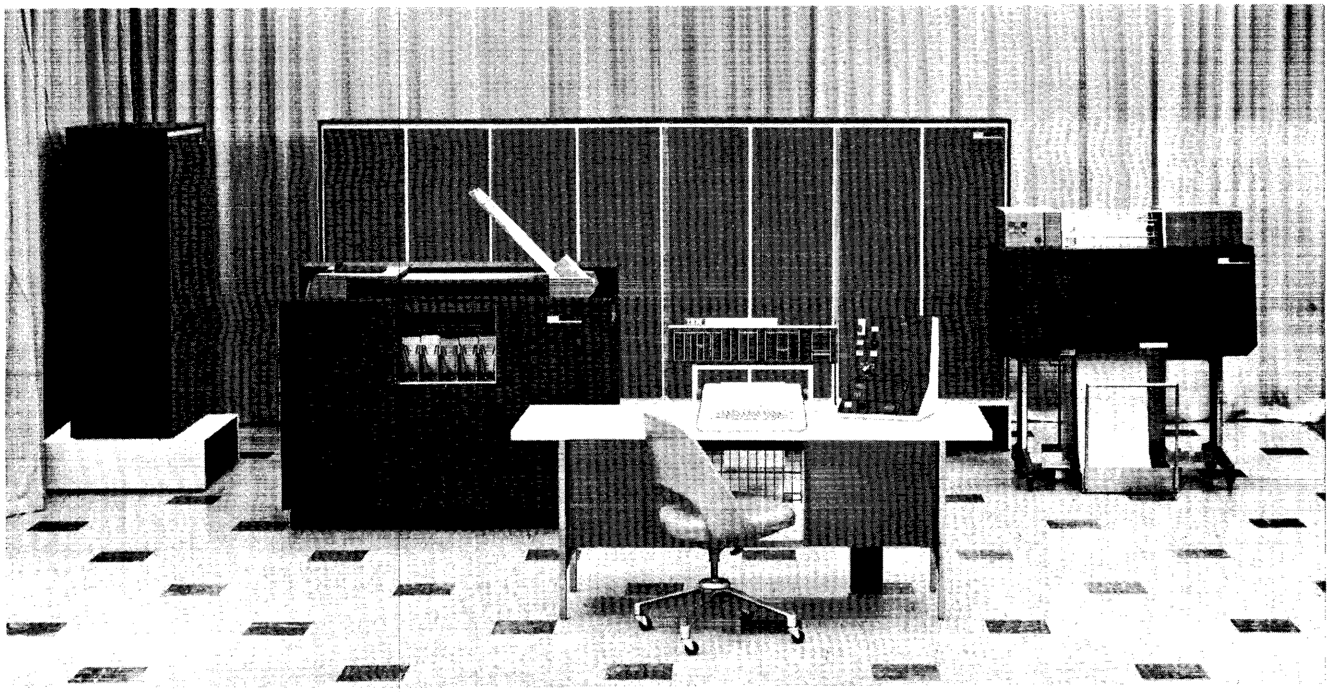


Figure 66. Card-Oriented IBM 1410 System

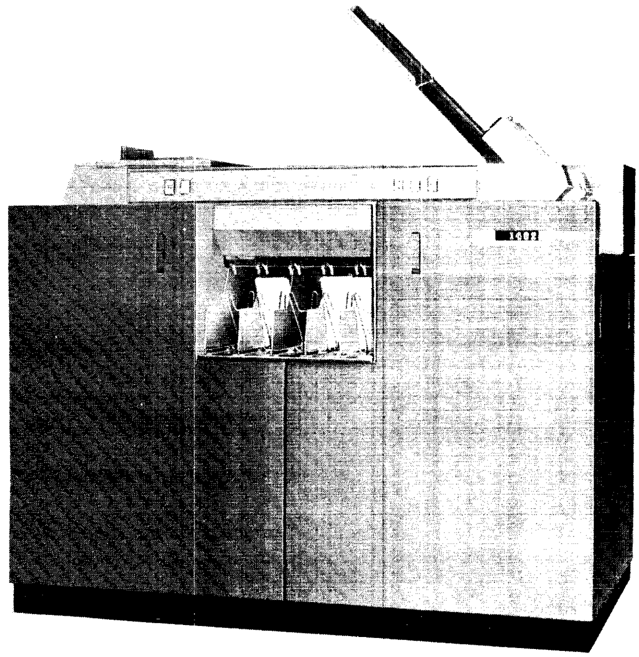


Figure 67. IBM 1402 Card Read-Punch, Model 2

1402-2 punch feed face down, 12-edge first, from left to right, past a blank station, the punch station, the punch-check brushes, and a stacker-selector station (Figure 68). The punch-check brushes read the entire card to establish a hole-count check (comparison of the same card as read by the punch-check brushes against the impulses received by the punch magnets).

The IBM 1402 Card Read-Punch, Model 2, is equipped with five radial-type stackers (Figure 69), with a capacity of 1,000 cards each. Cards from each feed can be directed, under program control, to three of the five pockets.

The cards in the card reader can be directed to the NR (normal read) pocket, the 1 pocket, or the 8/2 pocket. The cards in the card punch can be directed to the NP (normal punch) pocket, the 4 pocket, or the 8/2 pocket.

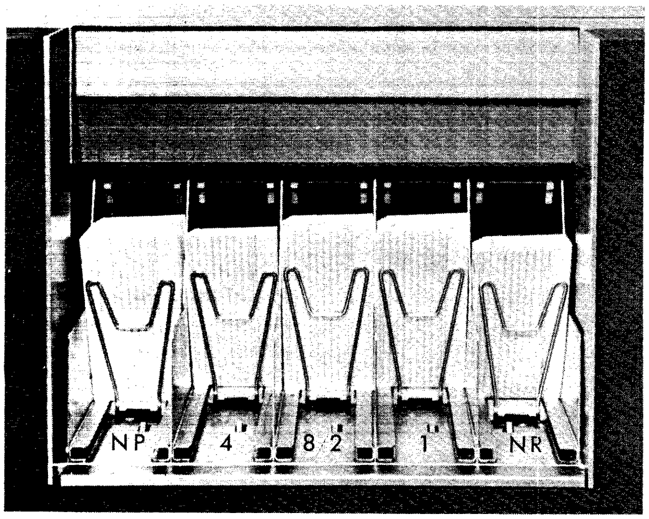


Figure 69. Radial Stackers

Note: Cards in either the punch or reader which result in validity errors or a hole-count check are *automatically* stacked in the NP or NR pocket.

Card Read-Punch Lights

The IBM 1402 Card Read-Punch, Model 2, has several lights (Figure 70) that refer to the machine rather than to one of the two units. These lights are:

STACKER

This light indicates that one or more pockets are full. Both the reader and the punch units stop.

FUSE

This light indicates that a fuse has blown in the reader or punch unit.

POWER

This light indicates that power is being supplied to the 1402.

TRANSPORT

This light indicates that a card jam has occurred in the stacker area. Card feeding is stopped in the rest of the 1402 until the jam is removed.

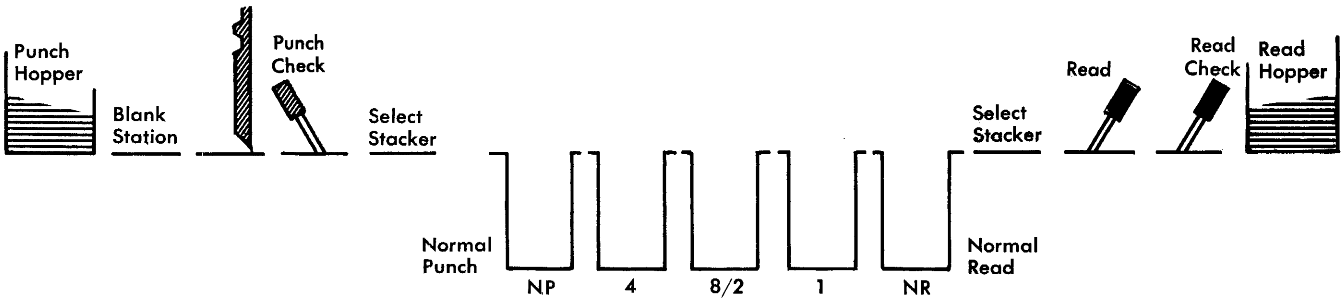


Figure 68. IBM 1402, Card-Transport Schematic

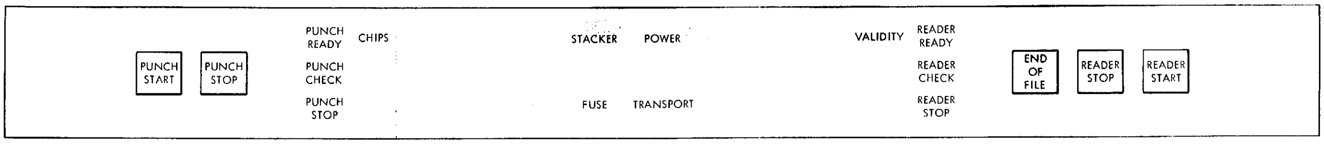


Figure 70. IBM 1402 Card Read-Punch Keys, Lights and Switches

Reader Keys and Lights

The card reader unit has certain keys and lights (Figure 70), which are:

READER START

Operating this key feeds three cards into the read feed, fills the reader synchronizer with the contents of the first card, and turns on the reader-ready light.

When the reader has been stopped, pressing the start key turns on the reader-ready light, and allows the cards to continue feeding under program control.

When the cards are removed from the read feed hopper and the end-of-file key is not operated, pressing the start key moves the remaining two or three cards to the stacker area unprocessed.

READER STOP

Operating this key stops the reader and turns off the reader-ready light.

END-OF-FILE

Operation of this key activates circuits that signal a last-card condition in the central processing unit. The last-card condition can be used by the stored program to initiate an end-of-file routine. The end-of-file latch is turned on following the data transfer of the last card. The next card-read instruction is interpreted as a NO OP.

The end-of-file key, which can be pressed at any time, causes the card reader to operate in one of these ways:

1. With four or more cards in the read hopper, all the cards are processed and run into a stacker. Operating the stop key or processing the last card causes the end-of-file condition to be reset.
2. With three cards remaining in the feed, a card-read or card-feed instruction before the operation of the end-of-file key causes the program to set the not-ready I/O channel status indicator. Pressing the end-of-file key and then the start key allows the last three cards to be processed and run into a stacker. Operating the stop key or processing the last card causes the end-of-file condition to be reset.
3. With the one, two, or three cards to be processed in the read hopper, pressing the end-of-file key and then the start key feeds the card or cards and turns on the reader-ready light after the first card passes the second read station. The card or cards are processed and run into a stacker. Operating the stop key or processing the last card causes the end-of-file condition to be reset.

READER READY

This light indicates that the reader is under stored-program control.

VALIDITY

This light indicates that an invalid character has been detected during a feed operation. The light remains ON until the next feed instruction is started. During the read instruction, the invalid character is transferred from synchronizer to storage.

READER STOP

This light indicates a feed failure or card jam during a feed operation. This error stops the reader and turns off the reader-ready light.

READER CHECK

This light indicates the detection of a hole-count error, parity error, or synchronizer-timing error during a feed operation. The light remains ON until the next feed instruction is started. During the read instruction, the data are transferred from synchronizer to storage, and the CPU sets the data check I/O channel status indicator ON and the program can test it.

Punch Unit Keys and Lights

The card punch unit has certain keys and lights (Figure 70), which are:

PUNCH START

Operating this key feeds two cards into the punch feed and turns on the punch-ready light.

When the punch has been stopped, pressing the start key turns on the punch-ready light, and allows card punching to resume under program control.

When the cards have been removed from the punch feed hopper, pressing the start key moves the three cards remaining in the punch feed to the normal-punch pocket. The first card that enters the normal-punch pocket is unchecked.

PUNCH STOP

Operating this key stops the punch and turns off the punch-ready light.

PUNCH READY

This light indicates that the punch is under stored-program control.

PUNCH STOP

This light indicates a feed failure or card jam during a punch operation. This error stops the punch and turns off the punch-ready light.

PUNCH CHECK

This light indicates the detection of a hole-count error, parity error, or synchronizer-timing error during a punch operation.

CHIPS

This light indicates that the chip receptacle is full or not in place.

IBM 1402 Card Read-Punch Operation Codes

Refer to Figure 71 for IBM 1402 operation codes.

Read a Card

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
R	$\overset{\text{v}}{\text{M}}$	%ln	xxxxx	R
RW	$\overset{\text{v}}{\text{L}}$	%ln	xxxxx	R

Function. This instruction causes the transfer of eighty characters in the card-read synchronizer to core storage. If the units position (n) in the x-control field is 0, 1, or 2, a card feed is initiated and the card from which this data came (which is now at the select station) is directed to the NR, 1, or 8/2 pocket. During the feed cycle the card-read synchronizer is refilled and this card is positioned at

I/O UNIT	CODE	X-CTRL FIELD	DESCRIPTION	MNE-MONIC	d-CHARACTER		OPERATION	NOTES
					CHAR-ACTER	CONTROL		
CARD READER	$\overset{\text{v}}{\text{M}}$ or $\overset{\text{v}}{\text{L}}$	%10	Read a card	R or RW	R	Read	Initiate feed cycle. Transfer 80 characters from read synchronizer to core storage. Read card into read synchronizer. Stack card in the NR pocket.	If $\overset{\text{v}}{\text{L}}$ op code is used, word-separator characters are read into storage as word marks, and are associated with the following data character.
		%11					Same as above, except card is stacked in the 1 pocket.	
		%12					Same as above, except card is stacked in the 8/2 pocket.	
		%19					Transfer 80 characters from read synchronizer to core storage. THERE IS NOT A CARD FEED AND STACKER SELECT OPERATION	
CARD READER	$\overset{\text{v}}{\text{K}}$		Select stacker and feed	SSF	0 1 2	Select NR stacker Select read stacker 1 Select read stacker 8/2	Initiate feed cycle. Read card into read synchronizer. Stack card in the designated pocket.	Should be used only after a Read a Card instruction with a 9 in the units position of the X-control field.
CARD PUNCH	$\overset{\text{v}}{\text{M}}$ or $\overset{\text{v}}{\text{L}}$	%40	Punch a card	P or PW	W	Write	Transfer 80 characters from core storage to punch synchronizer. Punch a card. Stack card in the NP pocket.	If $\overset{\text{v}}{\text{L}}$ op code is used, word-marks are translated to word-separator characters and are punched ahead of its associated character.
		%44					Same as above, except card is stacked in the 4 pocket.	
		%48					Same as above, except card is stacked in the 8/2 pocket.	

Figure 71. IBM 1402 Card Read-Punch Op Codes

the select station. If the units position (n) in the x-control field is 9, the contents of the card-read synchronizer are transferred to core storage, but there is no card feed and stacker-select operation.

The B-address specifies the high-order (starting) position of the data record in core storage. Data records are transferred from high-order to low-order position (left-to-right). The operation is stopped by the first group-mark with a word-mark sensed in core storage.

Word Marks. A group-mark with a word-mark must appear in the core-storage position to the immediate right of the data record. If the $\overset{\vee}{\text{L}}$ Op code is used, word-separator characters are read into storage as word marks. The character in the adjacent card column is also stored in the core-storage position that contains the word mark. This combination of word marks and their associated characters affects the record length.

Timing. T = 49.5 + I/O.

(See IBM *1402, Model 2, Timing Considerations.)*

Note. A BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction, $\overset{\vee}{\text{R}}$ (I) ⧧, must be given before the next channel 1 I/O unit operation, to be sure that the data was properly transferred.

(A specific $\overset{\vee}{\text{R}}$ (I) d instruction may be substituted for the $\overset{\vee}{\text{R}}$ (I) ⧧ instruction. This will not cause a system interlock as long as a branch to the I-address, specified in the $\overset{\vee}{\text{R}}$ (I) d instruction, occurs.)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	B + LB + 1

Select Stacker and Feed

Instruction Format.

Mnemonic	*Op Code*	*d-character*
SSF	$\overset{\vee}{\text{K}}$	0, 1, or 2

Function. This instruction (used after a READ A CARD instruction that had a 9 in the units position of the x-control field) stacks the card that was read on the last card-read cycle into the NR, 1, or 2 pocket, depending on the d-character of 0, 1, or 2. A card feed is initiated that refills the card-read synchronizer and positions this card at the select station.

Word Marks. Word marks are not affected.

Timing. T = 13.5 + I/O.

(See IBM *1402, Model 2, Timing Considerations.)*

Note. A BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction, $\overset{\vee}{\text{R}}$ (I) ⧧, must be given before the next channel 1 I/O unit operation, to be sure that the data was properly transferred.

(A specific $\overset{\vee}{\text{R}}$ (I) d instruction may be substituted for the $\overset{\vee}{\text{R}}$ (I) ⧧ instruction. This will not cause a system interlock as long as a branch to the I-address, specified in the $\overset{\vee}{\text{R}}$ (I) d instruction, occurs.)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	Bp

Punch a Card

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
P	$\overset{\vee}{\text{M}}$	%4n	xxxxx	W
PW	$\overset{\vee}{\text{L}}$	%4n	xxxxx	W

Function. This instruction causes the transfer of eighty characters in core storage to the punch synchronizer. the units position of the x-control field (n) can contain 0, 4, or 8, which directs the card to the NP, 4, or 8 pocket.

INDICATOR	d-CHARACTER BIT	CONDITION
Not Ready	1	Card jam Reader out of cards (not EOF) Reader not on line Reader power off Reader stacker full Cover interlock open Feed clutch failure (clutch chk) Joggle switch open (file feed door) Input/Output Synchronizer off line Input/Output Synchronizer power off
Busy	2	Read synchronizer being filled Card being stacked
Data Check	4	Hole count check Input/Output Synchronizer detects parity error Input/Output Synchronizer detects timing error Processing Unit detects parity error *Never set on Stacker Select and Feed Instruction*
Condition	8	EOF (last card has been stacked) (EOF latch turned off as this indicator turned on) *Never set on Stacker Select and Feed Instruction*
Wrong Length Record	B	Wrong length record *Never set on Stacker Select and Feed Instruction*
No Transfer	A	Card has been transferred previously. This indicator will be set ON if two Select Stacker and Feed Instructions are given without an intervening Read a Card Instruction with a 9 in the units position (n) of the X-control field. It will also be set ON if two Read a Card Instructions with a 9 in the units position of the X-control field are given without an intervening Select Stacker and Feed Instruction.

Figure 72. I/O Channel-Status Indicators Set During Card Reader Operations

The B-address specifies the high-order (starting) position of the data record in core storage. Data records are transferred from the high-order to the low-order position (left-to-right). The operation is stopped by the first group-mark with a word-mark sensed in core storage. At the end of the data transfer, the punch is started and punches a card with the data just transferred.

Word Marks. A group-mark with a word-mark must appear in the core-storage position to the immediate right of the data record. If the Ľ Op code is used, word marks are translated to word-separator characters for punching. Use of the LOAD mode causes the word-separator character to be punched ahead of its associated character, and affects the result field length.

Timing. T = 49.5 + I/O.

(See IBM *1402, Model 2, Timing Considerations.*)

Note. A BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction, Ř (I) ǂ, must be given before the next channel 1 I/O unit operation, to be sure that the data was properly transferred.

(A specific Ř (I) d instruction may be substituted for the Ř (I) ǂ instruction. This will not cause a system interlock as long as a branch to the I-address, specified in the Ř (I) d instruction, occurs.)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	B + LB + 1

Refer to Figures 72 and 73 for the I/O channel status indicators set during IBM 1402 Card Read-Punch operations.

INDICATOR	d-CHARACTER BIT	CONDITION
Not Ready	1	Card jam Punch out of cards Punch stacker full Punch power off Punch not on line Chip basket full or not in place Cover interlock open
Busy	2	Previous card still being punched
Data Check	4	Input/Output Synchronizer detects parity error (card not punched)
Condition	8	Hole count check-detected during the punch cycle of the following card. If error card is stacked, a blank card (second card behind error card) is fed. Parity error detected during punching
Wrong Length Record	B	Wrong length record (this card not punched)
No Transfer	A	Never set

Figure 73. I/O Channel-Status Indicators Set During Card Punch Operations

IBM 1402, Model 2, Timing Considerations

The I/O units are BUSY transferring data to or from their input-output synchronizers for these times:

1. READ A CARD instruction, when the units position (n) of the x-control field is 0, 1, or 2.
   I/O = 0-75,000μ (access time) + 880μ (time needed by synchronizer to accept 80 positions of data at 11 microseconds per position) + 65,000μ (read cycle).
2. READ A CARD instruction, when the units position (n) of the x-control field is 9.
   I/O = 880μ (time needed by synchronizer to accept 80 positions of data at 11 microseconds per position).
3. SELECT STACKER AND FEED instruction.
   I/O = 0-75,000μ (access time) +65,000μ (read cycle).
4. PUNCH A CARD instruction.
   I/O = 0-60,000μ (access time) + 880μ (time needed by synchronizer to accept 80 positions of data at 11 microseconds per position) + 217,500μ (punch cycle).

Special Feature — Interchangeable 51-Column Read Feed

The interchangeable 51-column read feed (including file feed) permits feeding either 51-column cards or standard 80-column cards in the read feed of the IBM 1402 Card Read-Punch, Model 2.

The 51-column card is commonly used for charge sales slips, postal money-order forms, installment payments, inventory cards, and many other applications.

Using an interchangeable feed allows direct entry to the data processing system from the stub card. This eliminates the need for reproducing 51-column cards into standard 80-column cards.

To adapt the read feed for 51-column-card operation, the operator installs a tray and hopper side plates on the read file feed, and adjusts the stackers on the read side.

Normal operations of the IBM 1402 Card Read-Punch, Model 2, can be performed with 51-column cards in the read feed. For example, a file of 51-column cards can be processed in the read feed while the results are punched in 80-column cards in the punch feed. However, when the stackers are adjusted to accept 51-column cards, no cards from the read feed can be *selected* into stacker 8/2.

Machine Features

Modifying the read file feed and stackers readily adapts the IBM 1402 Card Read-Punch, Model 2, for processing 51-column cards.

MODIFYING THE FILE FEED

An *adapter tray* (Figure 74), placed on the file-feed magazine, accommodates the 51-column cards. A modified card weight enables feeding the last cards from the hopper. Inserting two *hopper side plates* (Figure 74) positions the 51-column cards at the center of the feed. Thumbscrews fasten the side plates to the hopper. Jogglers align the cards in the hoppers, as in standard operation.

In 51-column-card operation, the first column of the card corresponds to column 15 of an 80-column card, and is therefore read by brush 15; the last column corresponds to column 65 and is read by brush 65. A factor of 14 relates the card column to the reading brush. A switch for regulating the storing of information from a 51-column card is physically located in the 1402-2. It is automatically turned on when the stacker guide is pulled forward for stacking of 51-column cards.

When the switch is ON, the information from a 51-column card is read into positions 15 through 65 of the read synchronizer. Positions 1-14 and 66-80 of the read synchronizer are filled with valid blanks.

To check for proper transfer of the data from the card synchronizer to core storage, a group-mark with a word-mark must be inserted in the 52nd position of the core storage read-in area.

ADJUSTING THE STACKERS

The operator adjusts the stacker guide (Figures 75 and 76) at the rear of stackers NR and 1 to accommodate 51-column cards. A finger hole permits pulling the guide forward to reduce the depth of the stacker. A spring latch holds the guide securely in either the 51- or 80-column-card position.

A *pivot-plate* assembly (Figures 75 and 76) adapts the front of stackers NR and 1 for stacking either 51- or 80-column cards. The 51-column pivot plate with card-retaining levers swings down and fastens to the stacker separators. This assembly provides a lower pivot for properly stacking the 51-column cards.

For standard 80-column operation, the operator pulls each auxiliary pivot-plate assembly forward and then places it under the cover.

Modified *card-deck supports* (Figures 75 and 76) for stackers NR and 1 permit stacking 51-column cards, standard cards, and the scored cards processed by the machine. The capacity of each of these stackers is 800 cards.

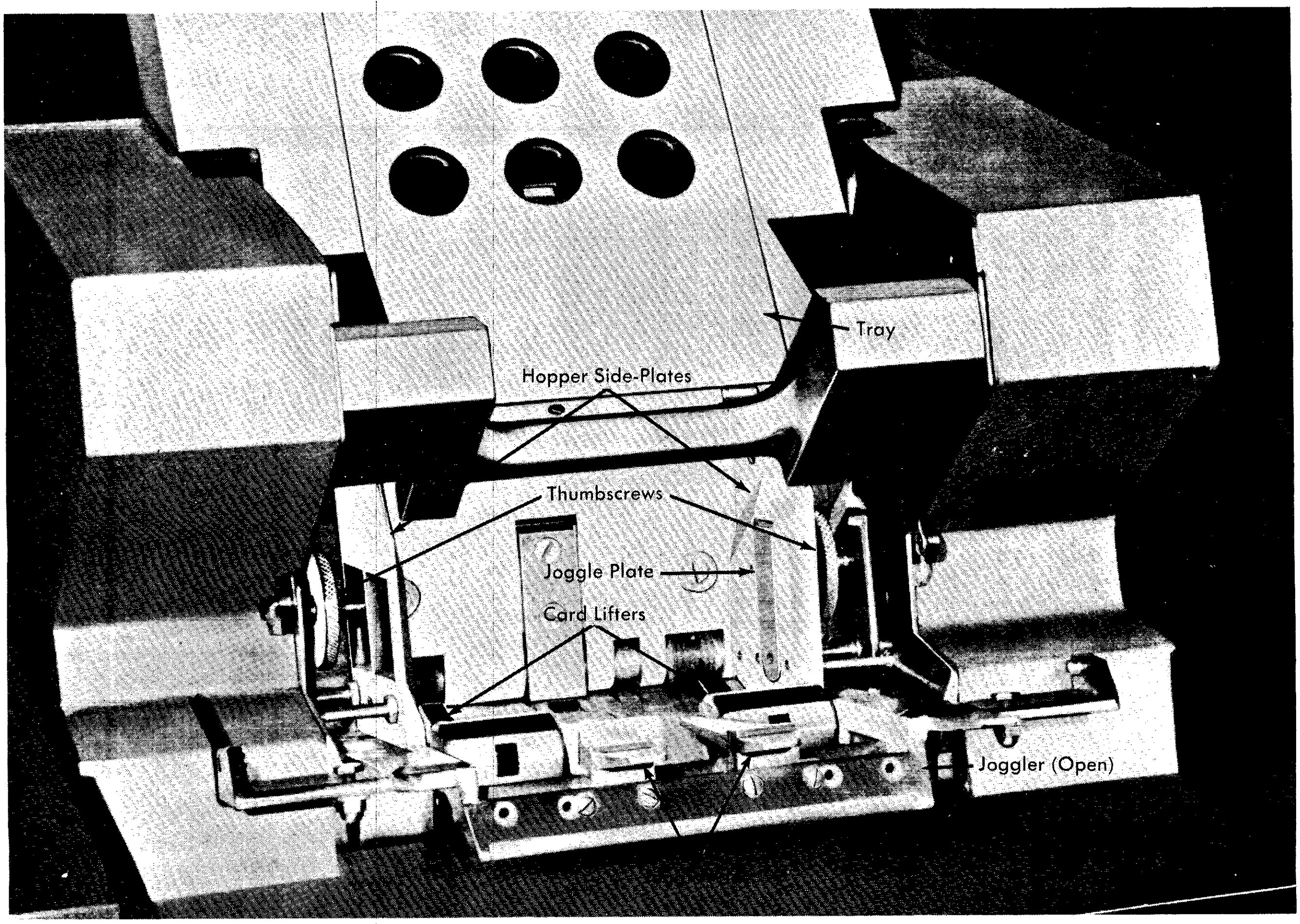


Figure 74. Interchangeable 51-Column Read Feed

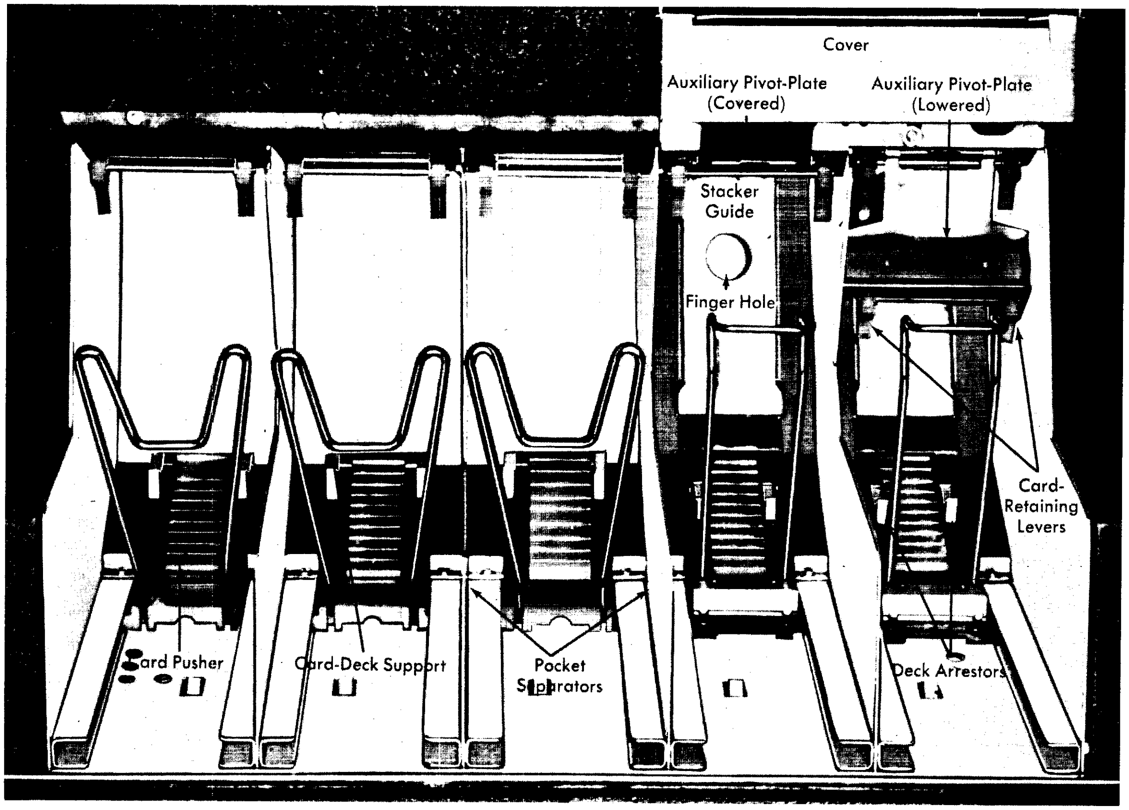


Figure 75. 51-Column Adjustable Stackers

SETUP OPERATION

To set up the IBM 1402 Card Read-Punch, Model 2, to feed 51-column cards in the read feed:

1. Position the side plates in the hopper, and fasten firmly by turning the knurled thumbscrews. Be careful not to interfere with the card lifters.
2. Place the 51-column-card tray over the file-feed magazine.
3. Reach into stackers NR and 1 and, using the finger hole, pull the guide forward until it latches.
4. Raise the cover over the auxiliary pivot-plate assemblies, lower one assembly partially, and then slide the main pivot-plate to the rear until it latches.
5. Swing the auxiliary pivot-plate assembly down until it latches to the stacker separators. (Repeat steps 4 and 5 for the other pivot-plate assembly.)

Reverse this procedure to return to standard card feeding. *Note:* Handle and store the adapter tray and hopper side plates carefully to avoid damaging them.

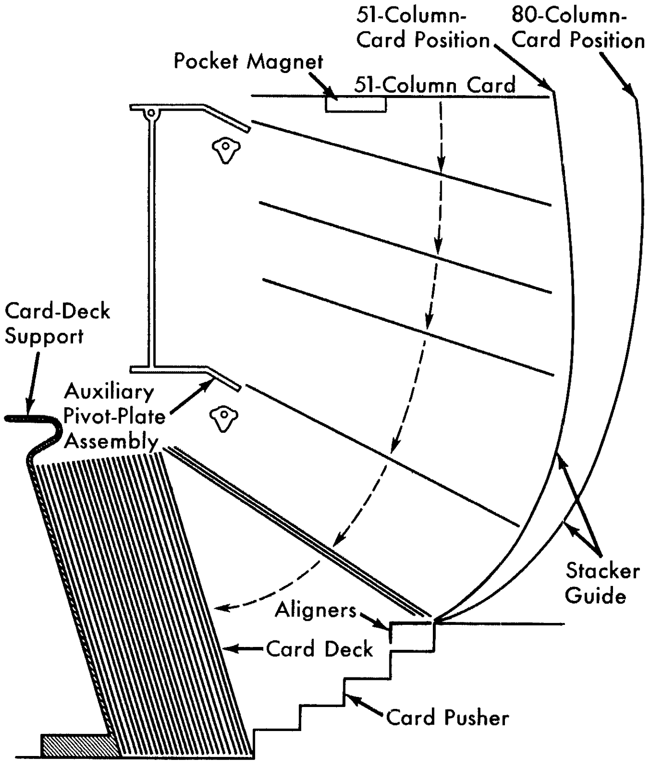


Figure 76. 51-Column Stacker Schematic

IBM 1403 Printer

The IBM 1403 Printer (Figure 77) is another output unit for the IBM 1410 Data Processing System. The

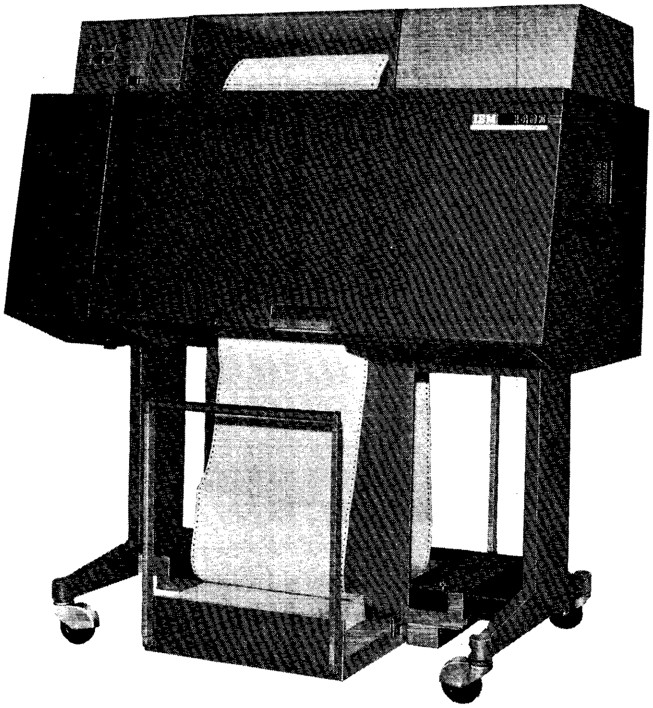


Figure 77. IBM 1403 Printer

standard printing capacity is 100 positions, with an additional 32 positions available as a special feature. Each position can print 48 different characters: 26 alphabetic, 10 numerical, and 12 special characters (& , . ⌑ – $ * / % # @ +). For information pertaining to the numerical print feature, refer to 1403 *Special Features* at end of this section.

METHOD OF PRINTING

The alphabetic, numerical, and special characters are assembled in a chain (Figure 78). As the chain travels in a horizontal plane, each character is printed when it is positioned opposite a magnet-driven hammer that presses the form against the chain.

When each character is printed, it is checked against the corresponding position in the print synchronizer to insure that printed output is accurate. Also, the machine checks to insure that the character is printed in the correct print position, that only valid characters are printed, and that over-printing does not occur.

Printer Keys and Lights

These keys and lights are shown in Figures 79 and 80.

PRINT START (FRONT AND BACK)

Operating this key turns on the ready light.

PRINT STOP (FRONT AND BACK)

Operating the stop key turns off the ready light. If the stored program attempts to execute a print instruction, the program automatically sets the not-ready I/O channel status indicator ON in the CPU and turns on its associated light.

CHECK RESET

This key resets a printer error indication. The print-start key is then pressed to resume operation.

PRINT READY

This light indicates that the printer is ready to print.

END-OF-FORMS

This light indicates an end-of-forms condition and the machine stops.

FORMS CHECK

This light indicates that there is paper-feed trouble in the forms tractor or that the carriage stop has been used. This light must be cleared by the check-reset key before the print-start key is effective.

PRINT CHECK

This light indicates a print error.

SYNC CHECK (SYNCHRONISM CHECK)

This light comes ON to show that the chain was not in synchronism with the printer-compare counter. The timing is automatically corrected. The light is extinguished by operating the printer-start key.

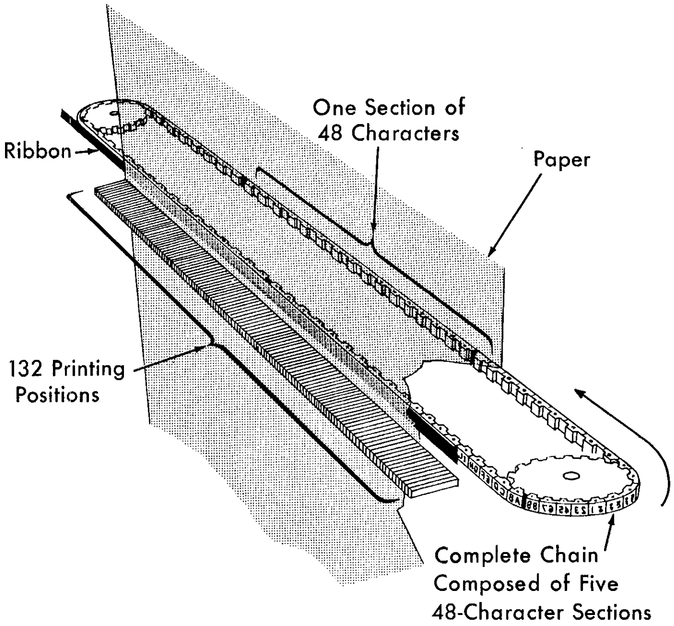


Figure 78. Printing Mechanism, Schematic

IBM 1403 Carriage Controls

The carriage controls are shown in Figure 79.

CARRIAGE RESTORE

Pressing this key positions the carriage at channel 1 *(home position)*. If the carriage feed clutch is disengaged, the form does not move. If it is engaged, the form moves in synchronization with the control tape.

CARRIAGE STOP

Pressing this key stops carriage operation and turns ON the forms-check light.

CARRIAGE SPACE

Each time it is pressed, this key causes carriage tape and the form to advance one space.

SINGLE CYCLE

This key initiates the operation of the printer for one print cycle on each pressing of the key when the end-of-form light is ON and no paper jam exists. This allows printing of the last line of a form.

IBM 1403 Manual Controls

The manual controls are shown in Figure 81.

FEED CLUTCH

The feed clutch controls the carriage-tape drive and form-feeding mechanism. If it is set to neutral, automatic form-feeding cannot take place. It is also used to select six- or eight-lines-to-the-inch spacing.

PAPER-ADVANCE KNOB

This knob positions the form vertically. It can be used only when the feed clutch is disengaged.

VERTICAL-PRINT ADJUSTMENT

This knob makes possible fine spacing adjustments of forms at the print line. Carriage tape is not affected by this knob.

LATERAL-PRINT VERNIER

This knob obtains fine horizontal positioning.

PRINT-DENSITY CONTROL LEVER

As many as six forms can be printed at one time, and the print hammer unit is designed to adjust automatically for different thicknesses of forms. However, to provide a vernier control for print impression, a print-density control lever is used. When this lever is set at position E, print impression is lightest. When this lever is set at position A, print impression is darkest. Between these two settings are intermediate settings. Position C is considered the normal setting. This lever moves the type chain closer to or farther from the hammer unit.

The setting of this lever must be considered together with the forms thickness, to determine the normal setting of the print-timing dial (Figure 82). A chart is provided to determine the normal setting (Figure 83).

PRINT-TIMING DIAL

A movable dial is set to a fixed indicator. Numbers around the dial provide a means of setting the print timing for a specific operation. The setting of the print-density control lever must be set before the print-timing dial is set. The nominal setting is read from a chart (Figure 83).

The chart (Figure 83) should give the correct setting of the print-timing dial. However, this setting can be

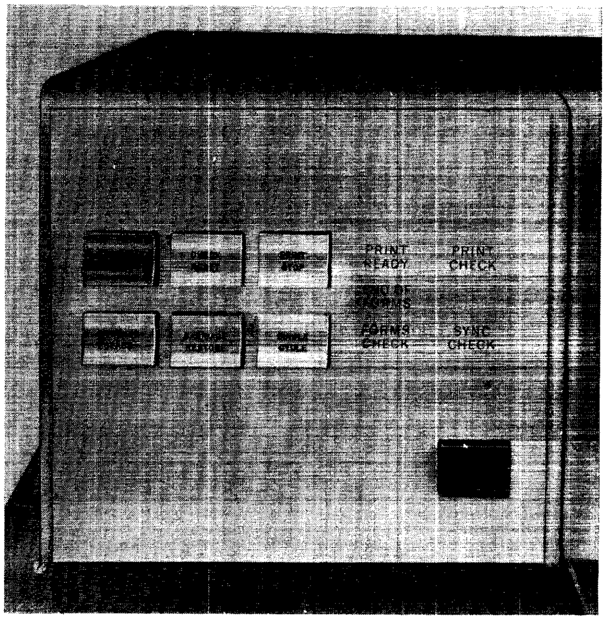


Figure 79. IBM 1403 Printer, Operating Keys and Lights

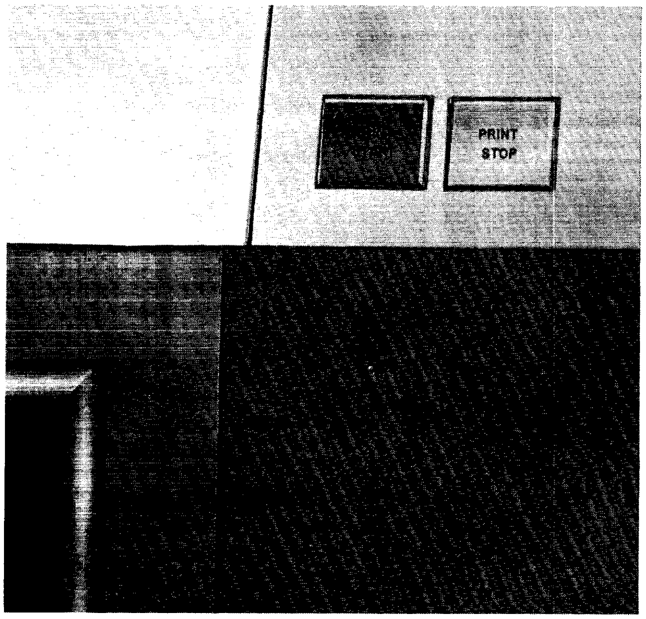


Figure 80. Printer Keys (Rear)

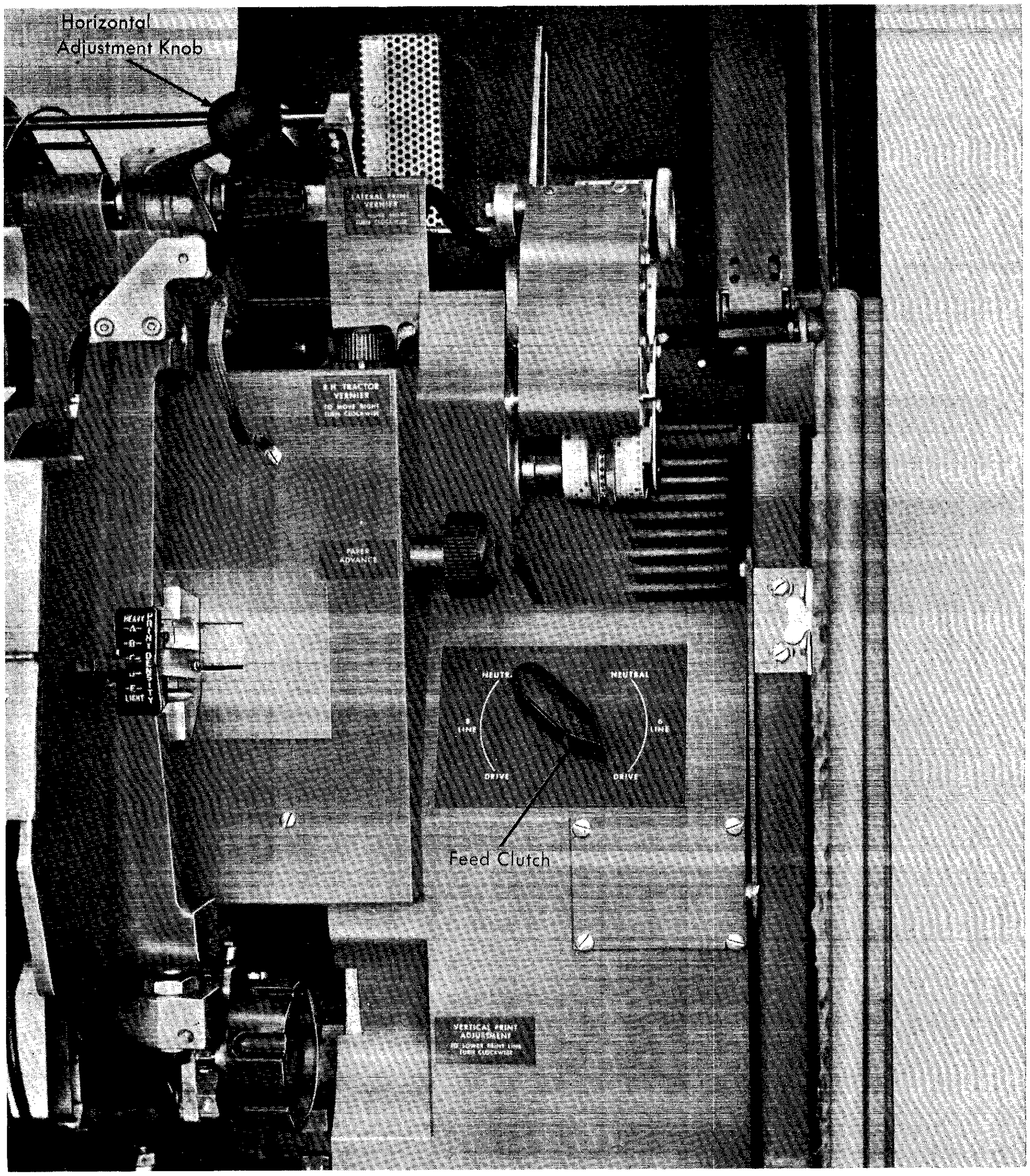


Figure 81. Carriage Controls

checked (by rotating the dial slowly in each direction from the normal setting) to determine the limits of good print quality.

PRINT-UNIT RELEASE LEVER

This lever permits access to the form transport area (see Figure 82).

PRINT-LINE INDICATOR AND RIBBON SHIELD

The lower ribbon shield is also used as a print-line indicator. It pivots along with the ribbon mechanism. The front side of this shield is marked to show print position location (Figure 84).

When used as a print-line indicator, the shield indicates where the lower edge of characters will print.

When the printer frame is open, the indicator pivots against the forms so that the print line may be set with respect to the forms.

HORIZONTAL ADJUSTMENT

This device (see Figure 81) positions the printing mechanism horizontally. When the lever is raised, the print mechanism unlocks, and can be positioned horizontally within its 2.4-inch travel.

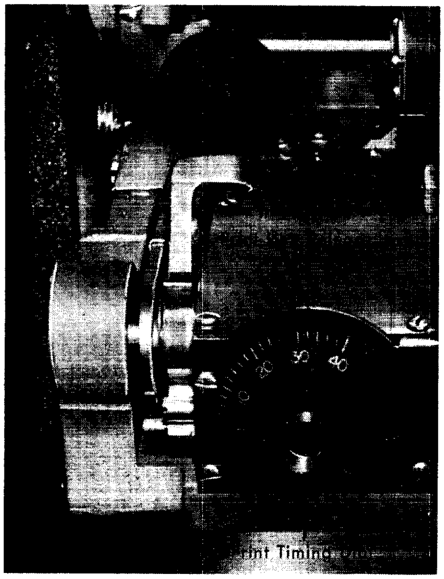
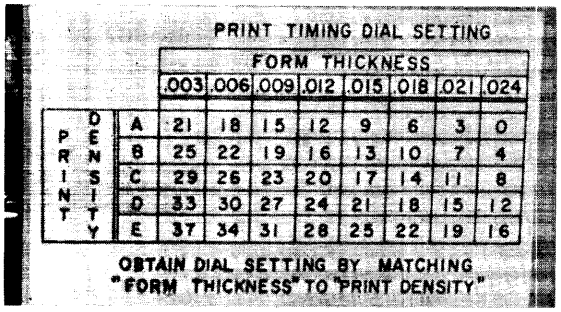


Figure 82. Print-Timing Dial and Print Unit Release Lever



PRINT TIMING DIAL SETTING

PRINT DENSITY	FORM THICKNESS							
	.003	.006	.009	.012	.015	.018	.021	.024
A	21	18	15	12	9	6	3	0
B	25	22	19	16	13	10	7	4
C	29	26	23	20	17	14	11	8
D	33	30	27	24	21	18	15	12
E	37	34	31	28	25	22	19	16

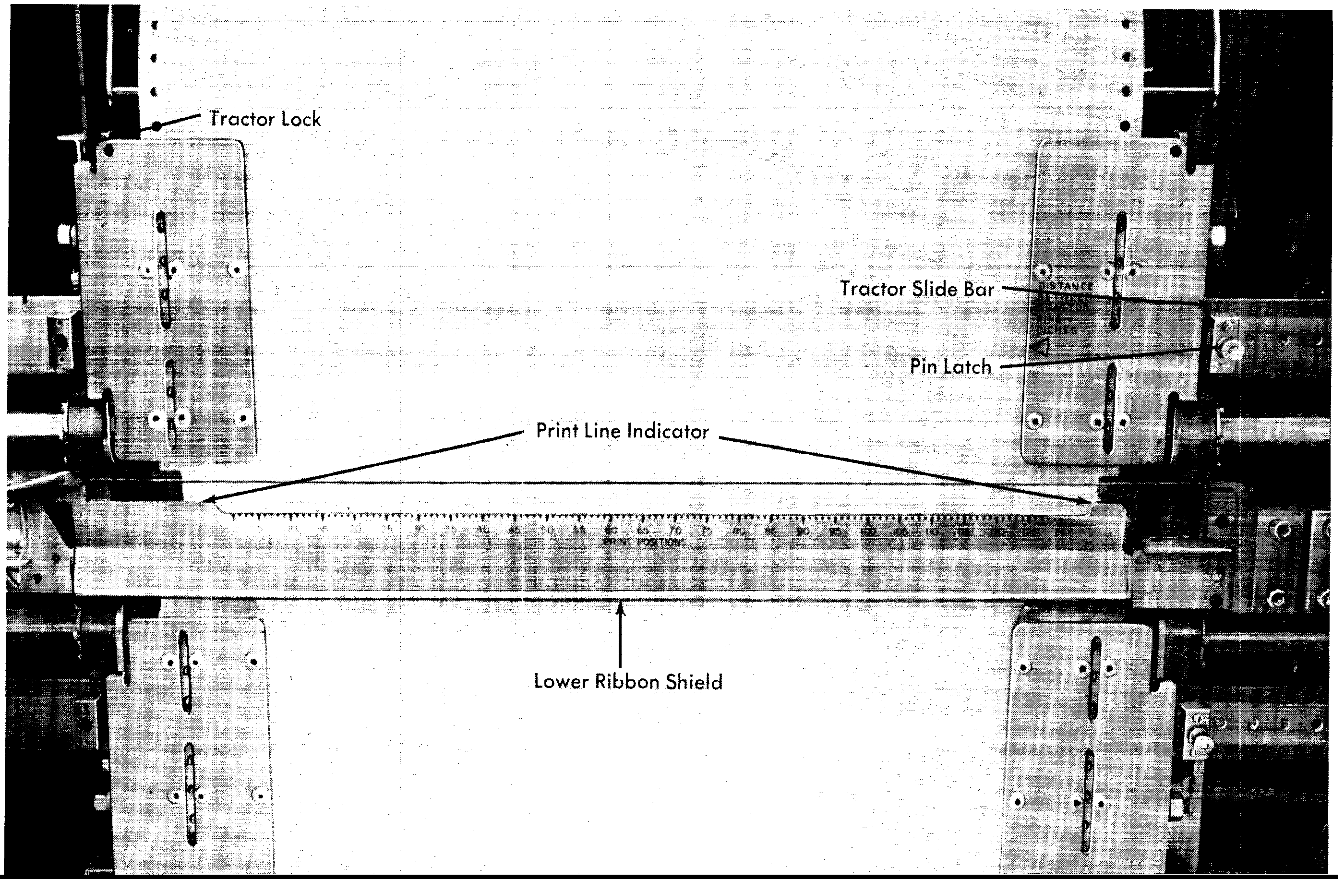
OBTAIN DIAL SETTING BY MATCHING "FORM THICKNESS" TO "PRINT DENSITY"

Figure 83. Print-Timing Dial, Chart

R. H. TRACTOR VERNIER

This knob (see Figure 81) allows for fine adjustments in paper tension. It can be used for adjustments of up to one-half inch.

Tractor Slide Bar. There are two tractor slide bars, (see Figure 84) upper and lower. The forms tractors are mounted on these bars. The forms tractors are movable, and to facilitate this movement there are notches in the tractor slide bar. A procedure for proper adjustment of these notches, according to the



form being used, is given for the upper tractor slide bar. The description would be the same for the lower slide bar.

The left tractor is locked in place by a spring-loaded latch in one of the nine notches located one inch apart on the tractor slide bar. The third notch from the left end is the normal location for most applications.

The first notch is used for forms from 5½ to 18¾ inches wide. When this notch is used, the print unit's lateral movement is limited to .4 inch.

The second notch is used for forms from 4½ to 17¾ inches in width. When this notch is used, the print unit's lateral movement is limited to 1.4 inch.

The third notch is used for forms from 3½ to 16¾ inches wide. When this notch or notches 4 through 9 are used, full lateral print unit movement (2.4 inches) is possible.

The ninth (last) notch can be used for forms from 3½ to 10¾ inches wide. When this notch is used, the first usable print position is 38.

The right-hand tractor is locked in place by spring-loaded pins snapped into any one of 27 holes, located one-half inch apart on the tractor slide bar.

The movement of the tractor slide bar, in which the holes are located, is controlled by the right-hand tractor vernier. Movement of up to ½ inch can be made by using the vernier knob.

Indicator Panel Lights

GATE INTERLOCK

This light turns on when the print unit is not locked in position (Figure 85).

BRUSH INTERLOCK

This light is on if the carriage tape brushes are not latched in position for operation.

SHIFT INTERLOCK

This light turns on to indicate that the manual feed clutch is not properly positioned.

THERMAL INTERLOCK

This light indicates that a temperature above the operating limit has been sensed in the hammer unit or chain-drive unit; the light remains on until the temperature drops to an acceptable level. The 1403 is interlocked during this time.

HIGH SPEED START

This light turns on when a high-speed skip has been initiated.

LOW SPEED START

This light turns on when a low-speed skip or line spacing has been initiated.

HIGH SPEED STOP

This light turns on to indicate that high-speed skipping is to be stopped.

LOW SPEED STOP

This light turns on to indicate that a low-speed skip stop has been initiated. It is on when the carriage is not in motion.

Tape-Controlled Carriage

The tape-controlled carriage (Figure 86) controls high-speed feeding and spacing of continuous forms. The carriage is controlled by punched holes in a paper tape

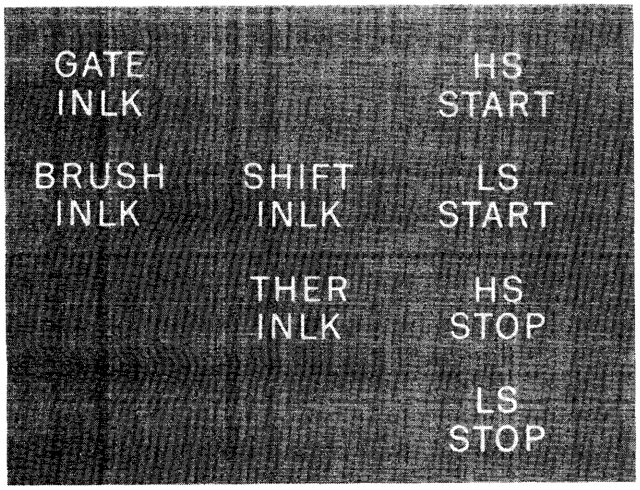


Figure 85. Printer Indicator Panel

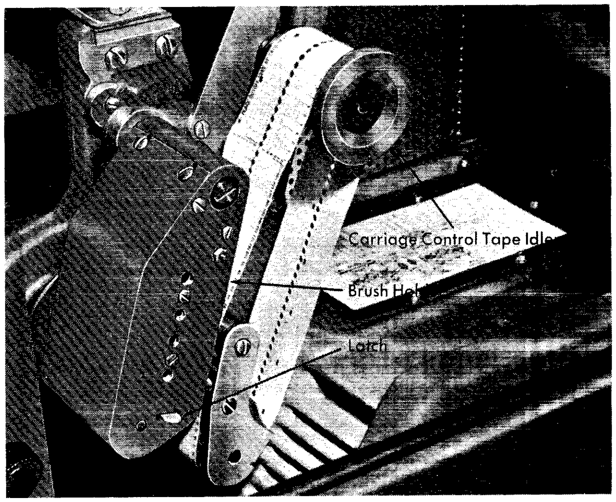


Figure 86. Tape-Controlled Carriage

that corresponds in length to the length of one or more forms. Holes punched in the tape stop the form when it reaches any predetermined position.

Carriage skip channels 1-12 are standard. The tape circuits initiate special signals that are sent to the CPU when channels 9-12 are sensed. Program testing of carriage channels 9 and 12 is standard.

Vertical spacing and skipping are initiated by the stored program. Horizontal spacing is 10 characters to the inch. Vertical spacing of either six or eight lines to the inch can be manually selected by the operator.

Forms skip at the rate of 33 inches per second. With the dual-speed carriage, distances of less than eight lines are skipped at 33 inches per second, and those of more than eight lines at 75 inches per second. The last eight spaces skipped in a high-speed skip are skipped at 33 inches per second.

The carriage accommodates continuous forms, up to a maximum of 22 inches in length (at 6 lines per inch) or 16½ inches (at 8 lines per inch). The minimum length is 1 inch. For efficient stacking of forms, the recommended maximum forms length is 17 inches. The width of the form can vary from a recommended minimum of 3½ inches to a maximum of 18¾ inches, including punched margins.

Forms can be designed to permit printing in practically any desired arrangement. Skipping to different sections of the form can be controlled by the program and by holes punched in the carriage tape.

CONTROL TAPE

The control tape (see Figure 86) has 12 columnar positions indicated by vertical lines. These positions are called channels. Holes can be punched in each channel throughout the length of the tape. A maximum of 132 lines can be used to control a form, although for convenience, the tape blanks are slightly longer. Horizontal lines are spaced six to the inch for the entire length of the tape. Round holes in the center of the tape are pre-punched for the pin-feed drive that advances the tape in synchronism with the movement of a printed form through the carriage. The effect is exactly the same as though the control holes were punched along the edge of each form.

Punching the Tape. A small, compact punch (Figure 87) is provided for punching the tape. The tape is first marked in the channels in which the holes are to be punched. This can be done easily by laying the tape beside the left edge of the form it is to control, with the top line (immediately under the *glue* portion) even with the top edge of the form. A mark is then made in the first channel, on the line that corresponds to the first printing line of the form. Additional marks are made in the appropriate channels for each of the other skip stops, and for the overflow signal required for the form.

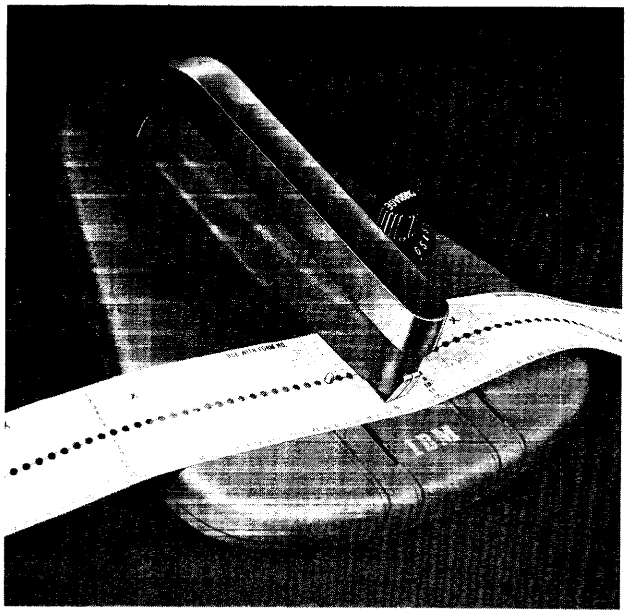


Figure 87. Tape Punch

The marking for one form should be repeated as many times as the usable length of the tape (22 inches) allows. With the tape thus controlling several forms in one revolution through the sensing mechanism, the life to the tape is increased. Finally, the line corresponding to the bottom edge of the last form should be marked for cutting after the tape is punched.

The tape is inserted in the punch by placing the line to be punched over a guide line on the base of the punch and placing the center feed holes of the tape over the pins projecting from the base. The dial is then turned until the arrow points at the number of the channel to be punched. Pressing on the top of the punch, toward the back, cuts a rectangular hole at the intersection of a vertical and horizontal line in the required channel of the tape. The tape should never be punched in more than one channel on the same line. Holes in the same channel should not be spaced closer than 8 lines apart. After the tape is punched, it is cut and looped into a belt. The bottom end is glued to the top section, marked *glue,* with the bottom line coinciding with the first line. Before the tape is glued, the glaze on the tape should be removed by an ink eraser; if this is not done, the tape ends may come apart. The center feed holes should coincide when the two ends of the tape are glued together.

The last hole punched in the tape should be at least four lines from the cut edge, because approximately the last half inch of the tape overlaps the *glue* section when the two ends are spliced. If it is necessary to punch a hole lower than four lines from the bottom of the form, the tape should be placed with the top line (immediately under the *glue* portion) four lines lower than the top edge of the form, before marking the channels. To compensate for the loss, the tape should then be cut four lines lower than the bottom edge of the form.

8-LINES-PER-INCH SPACING

The control tape for 8-lines-per-inch spacing is punched as it would be for normal 6-lines-per-inch spacing. Each line on the tape always equals one line on the form, regardless of whether the latter be 6 or 8 lines-per-inch. In measuring a control tape for a document printed 8 lines to the inch, every ⅛ inch on the form represents one line on the tape.

CARRIAGE TAPE BRUSHES

Two sets of reading brushes (Figure 88), mounted on the same frame, are used to sense holes in the carriage control tape. A small contact roll is used for each set of brushes. One set is called the *slow brushes.* The other set is called the *stop brushes.* Seven spaces, as measured by the control tape, separate the brush sets. The slow brushes are positioned ahead of the stop brushes.

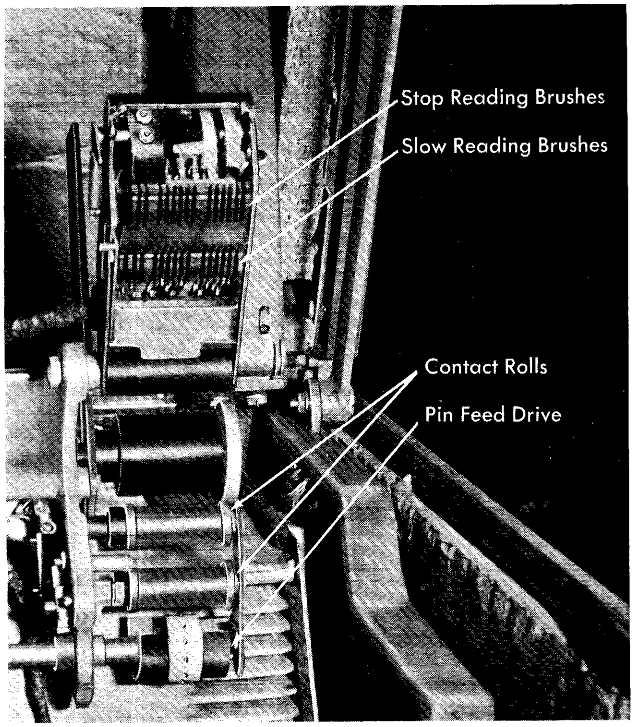


Figure 88. Carriage-Tape Brushes

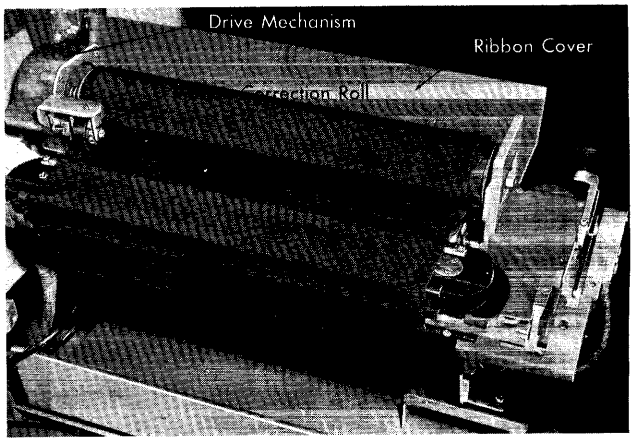


Figure 89. Ribbon Mechanism

The slow brushes are used to control high-speed skipping. They regulate the speed of the last eight spaces of a high-speed skip.

All carriage tape brushes can function to stop a carriage skip under control of the stored program.

INSERTING CONTROL TAPE IN CARRIAGE

1. Raise the counter-balanced cover of the printer to gain access to the tape-reading mechanism.
2. Turn the feed clutch to a disengaged (neutral) position (see Figure 81).
3. Raise the brushes by moving to the left the latch located on the side of the brush holder.
4. Place one end of the tape loop, held so that the printed captions can be read, over the pin-feed drive wheel so that the pins engage the center drive holes.
5. Place the opposite end of the loop around the adjustable carriage control tape idler.
6. Remove the excess slack from the tape by loosening the locking knob on the idler and moving the idler in its track. Tighten the knob when the desired tension is reached. The tape should be just tight enough so that it gives slightly when the top and bottom portions of the loop are pressed together (see Figure 86). If it fits too tightly, damage occurs to the pin-feed holes.
7. Press the brushes down until they latch, and close the printer cover, when the tape is in position.
8. Press the carriage restore key to bring the tape to its home position, and turn the feed clutch knob back to the engaged position. The carriage is ready to operate.

RIBBON CHANGING

To change the ribbon (Figure 89) on the IBM 1403 Printer:

1. Turn off the power in the printer.
2. Lift up the printer cover.

3. Pull back and unlock the print unit release lever. Swing the print unit out.
4. Open the top ribbon cover.
5. Unlatch the print-line indicator ribbon shield and swing it against the form.
6. Push the top ribbon roll to the right (hinged side of print unit), lift out the left end of the ribbon roll, and remove roll from the drive end of mechanism.
7. Slip the ribbon out from under the ribbon correction roll.
8. To remove the bottom roll, press the ribbon roll to the right, and lower the left end of the ribbon roll and remove it from the drive end of the mechanism.

When replacing the ribbon in the machine, hand-tighten the ribbon to remove slack from in front of the printing mechanism. Ribbons are available in widths of 5, 8, and 11 inches in addition to the standard 14 inches. The ribbon width lever (Figure 90) can adjust the ribbon-feed mechanism to accommodate the various ribbon widths.

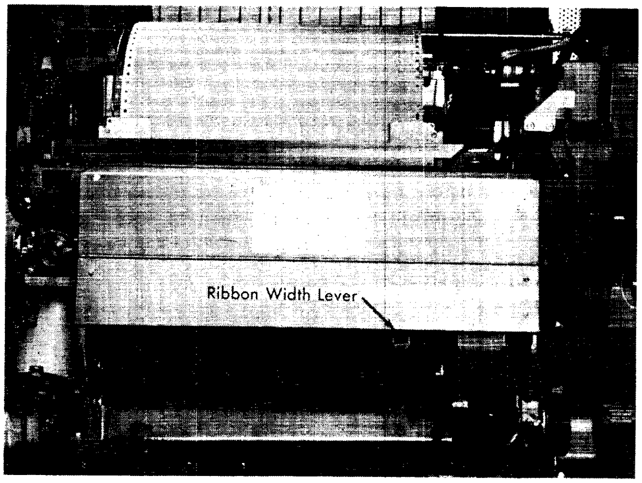


Figure 90. Front Cover, Open

FORMS INSERTION

1. Raise the counterbalanced cover of the printer to gain access to the print and forms area.
2. Turn the feed clutch knob to a neutral position.
3. Unlock and swing back the print unit by using the print unit release lever.
4. Unlock the paper guide bars by pulling out on the raised handles (upper and lower).
5. Open the upper and lower forms tractors (Figure 91).
6. Set the left forms tractors slightly to the left of the first unit position by pulling up or down in the tractor lock (upper and lower tractor). See Figure 84.
7. Insert form on pins and close tractor cover.

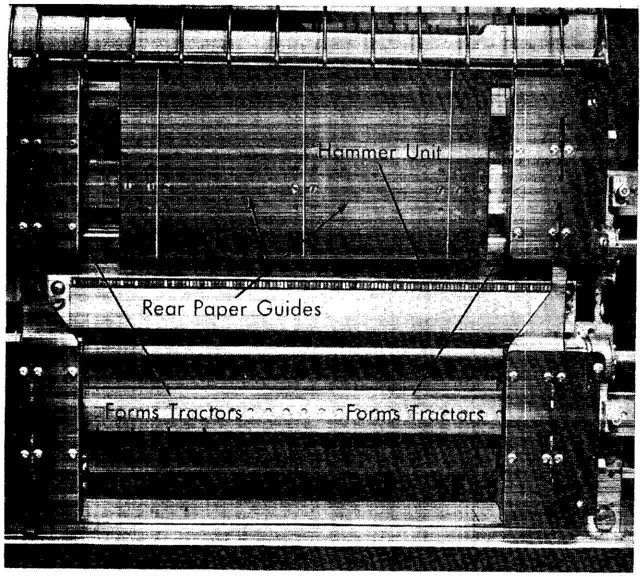


Figure 91. Forms Tractor

8. Pull out on right tractor pin and move tractor to desired location to line up the right side of form. The pin should latch in one of the recessions in the tractor slide bars. See Figure 84.
9. Insert form on pins and close tractor covers.
10. Use the tractor vernier knob to tighten the tension on the form. This knob is used for adjustments of up to one-half inch.
11. Check the position and line where printing will occur, by swinging the ribbon shield against the form (it is marked with each print position). If the horizontal alignment is not correct, it can be adjusted by using the horizontal adjustment knob and/or the lateral print vernier knob for slight adjustments. The vertical adjustment can be made by using the paper advance knob and/or vertical print adjustment knob.

*12. Return the upper and lower paper guide bars to the closed positions (Figure 92).

13. Return the print unit to its normal position and lock it in place.
14. Restore the carriage tape to the first printing position by pressing the carriage restore button.
15. Return the feed-clutch knob to a drive position at either six or eight lines-per-inch, depending on the form to be printed.
16. Close the outside cover of the printer.

*Some 1403 printers have the tractor-mounted jam detection device which, together with elimination of front "clip on" paper guides, eliminates the need for the upper and lower paper guide bars. The forms insertion procedure for a 1403 with the tractor mounted jam detection device instead of the upper and lower tape guides is the same except that steps 4 and 12 are skipped.

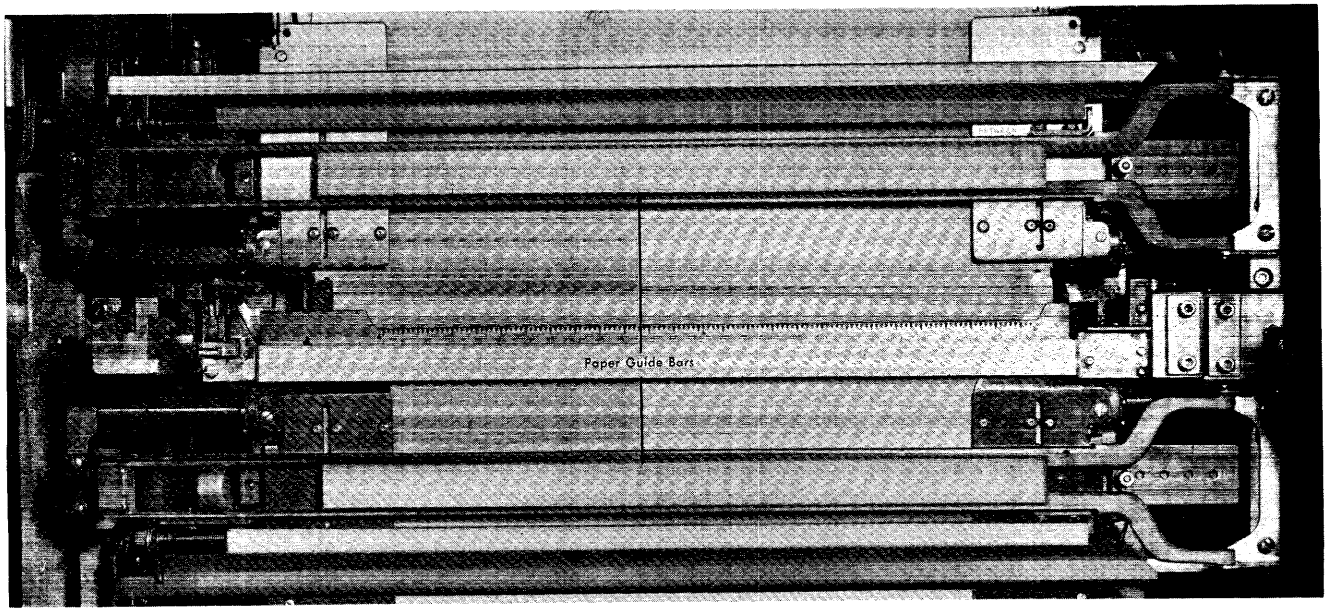


Figure 92. Paper Guide Bars

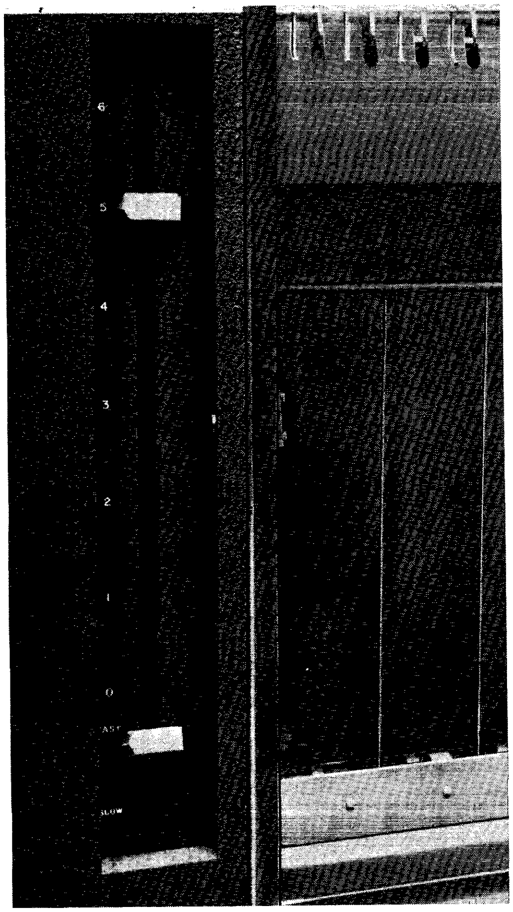


Figure 93. Paper-Stacker Controls

PAPER STACKER

The paper stacker provides a manual control for optimum stacking of paper at the rear of the printer. Two controls (Figure 93) permit the operator to set up the paper stacker for each individual run.

The upper lever controls the position of the paper guide at the stacker. This lever is indexed (0-6) so that the setup position can be recorded for reference in the operator's procedures.

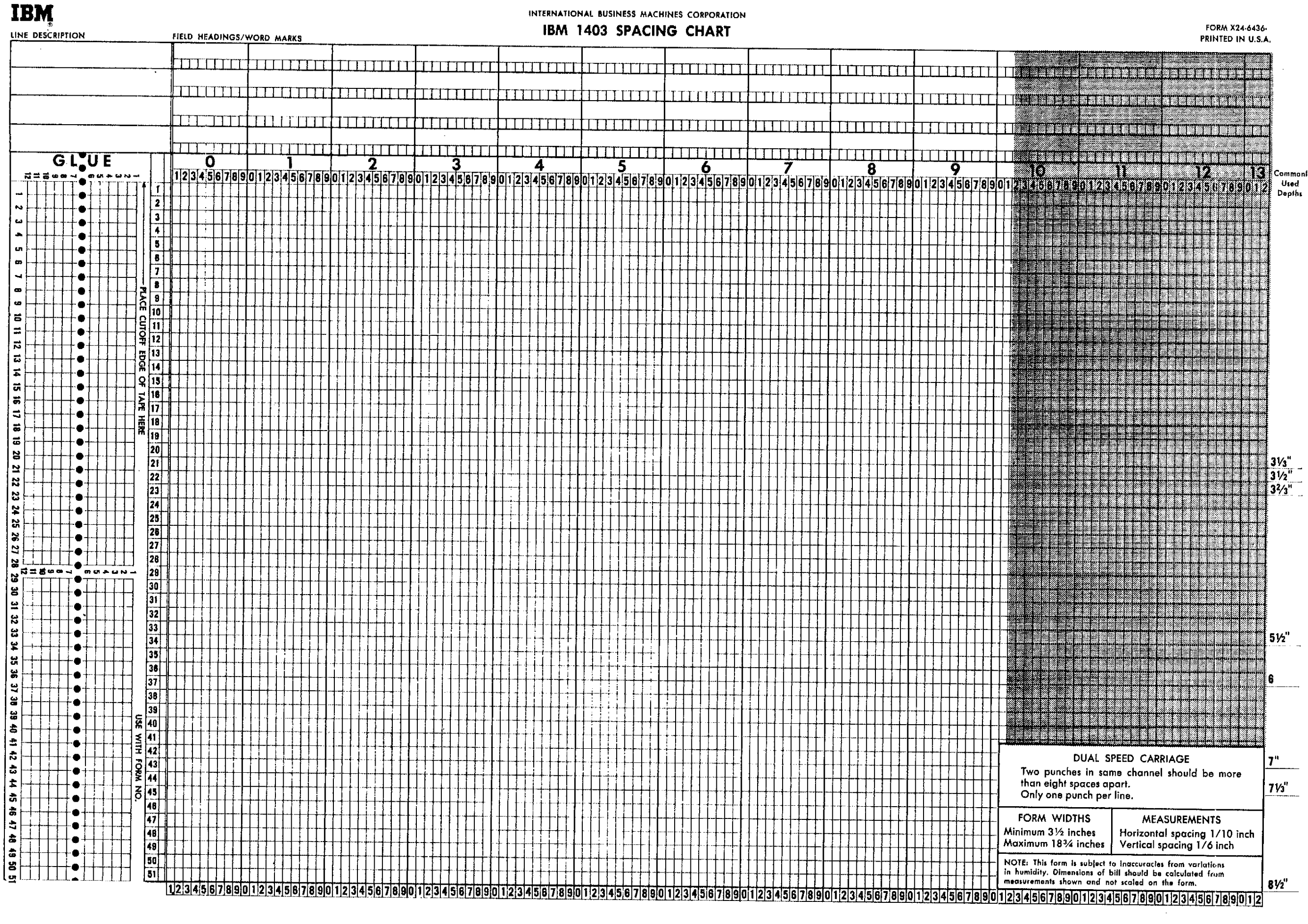
The lower lever is a speed control that is set to keep light tension on the paper form feeding into the stacker. The speed control has five settings. The setting of this control is selected according to the carriage operation being used. For example, if the job is a listing operation with no long skips, the slow position is selected. However, this must also be conditioned by the kind of forms being used because of varying weight of the paper.

Form Design

Some of the customary rules for designing forms should be reconsidered in the light of the many new features introduced by the IBM 1403 Printer.

1. The print unit contains 100 print positions in a 10.0-inch width or a maximum of 132 print positions (special feature) in a 13.2-inch width. Each print position can print any character.
2. Editing, high-speed skipping and other features are included in the system.

One of the basic tools used in designing forms is the spacing chart shown in Figure 94. The numbers across the top from 0 to 13 represent the tens and hundreds positions of the print-position number, and the numbers directly beneath represent the units position


IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

IBM 1403 SPACING CHART

FORM X24-6436-
PRINTED IN U.S.A.

LINE DESCRIPTION

FIELD HEADINGS/WORD MARKS

GLUE

PLACE CUTOFF EDGE OF TAPE HERE

USE WITH FORM NO.

Commonly Used Depths

DUAL SPEED CARRIAGE
Two punches in same channel should be more than eight spaces apart.
Only one punch per line.

FORM WIDTHS	MEASUREMENTS
Minimum 3½ inches Maximum 18¾ inches	Horizontal spacing 1/10 inch Vertical spacing 1/6 inch

NOTE: This form is subject to inaccuracies from variations in humidity. Dimensions of bill should be calculated from measurements shown and not scaled on the form.

Figure 94. Forms-Spacing Chart

of the print-position number. Print-position 42 can be located by referring first to the 4 column and then to the digit 2 within the 4 column. Print-position 9 can be located by referring to the 0 column and then to the digit 9 within that column.

A facsimile of the carriage-control tape is shown at the left (in Figure 94) for marking the control punching for a specific form. Notations have been included relative to standard form-widths and form-depths, lateral movement of the carriage, and instructions to forms manufacturers.

The IBM 1403 Printer carriage is designed to feed marginally-punched continuous forms satisfactorily under the conditions and specifications outlined in Figure 95. These specifications, if followed, give maximum operating efficiency when the 1403 carriage is used. They are not intended to be restrictive, but rather they are intended to permit customers to purchase their continuous forms from the manufacturer of their choice.

FORM DESIGN AS AFFECTED BY THE PRINT UNIT

In view of the 100 or 132 print positions and the 13.2-inch print unit, these factors should be considered when designing forms to be used on the IBM 1403 Printer:

1. The maximum form width is 18¾ inches, and the minimum is 3½ inches (see Figure 95).
2. The maximum form length is 22 inches at six-lines-per-inch spacing, or 16½ inches at 8 lines per inch. For efficient stacking of forms, the recommended maximum forms length is 17 inches.
3. Because all print positions can print all characters, form depth can be reduced, and carbon paper eliminated, by the use of side-by-side printing. For example, *sold to* and *ship to* names can be printed on the same line, one on the left side of the form and the other on the right.
4. Forms can be designed for printing six or eight lines to the inch. Single-space, eight-lines-per-inch printing is not recommended when the registration between lines is critical.
5. Forms can be designed for variable line spacing within a form by use of single-, double-, or selective-space control.
6. It is possible to dispense with many vertical lines, because the system can be programmed to print

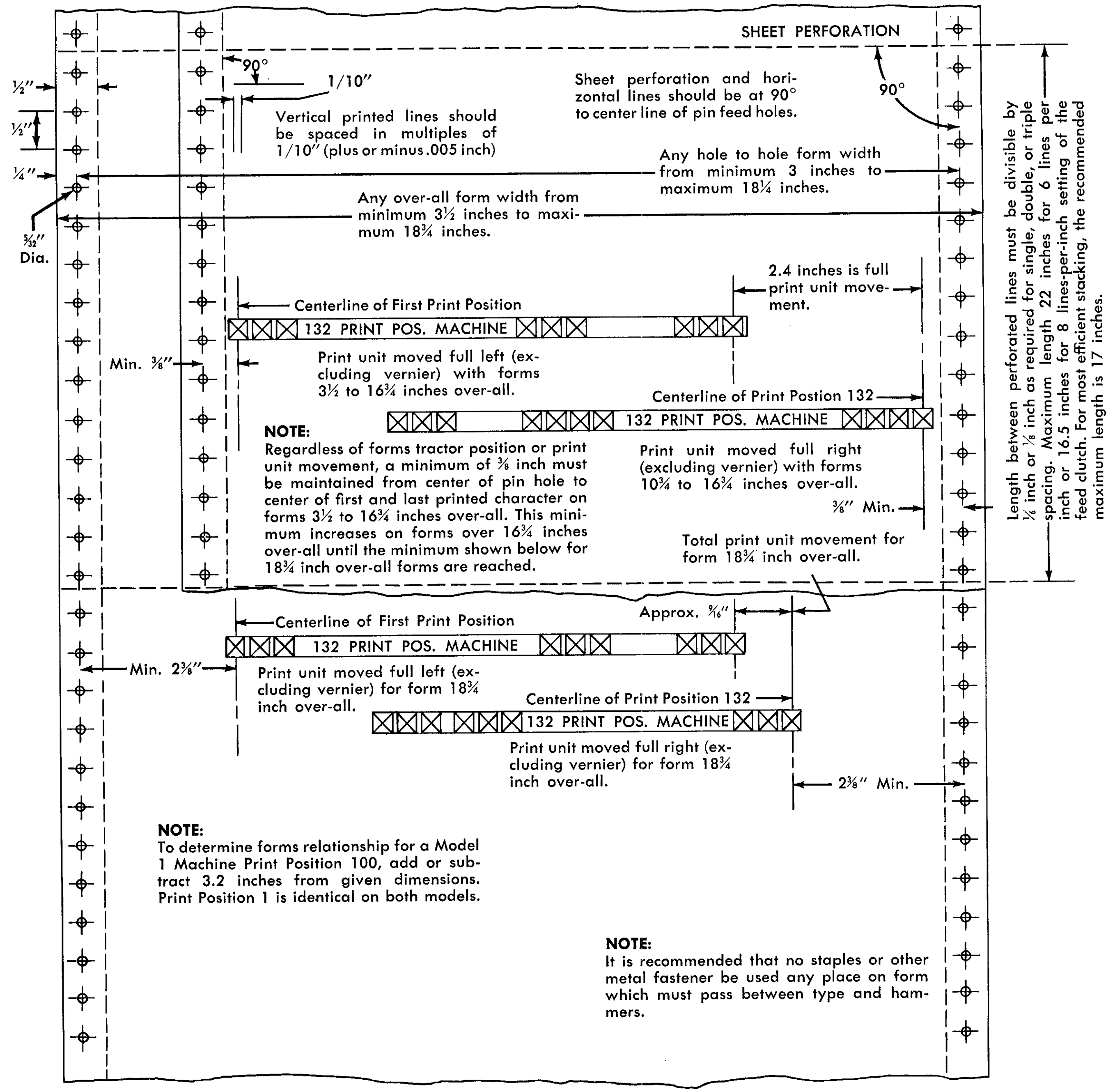


Figure 95. Form Specifications

commas, decimals, oblique lines, dashes, and other symbols.

7. A vertical line should not be printed between two adjacent printing positions because there is an over-all maximum tolerance of only .013 inch between adjacent characters.
8. The number of legible copies that can be produced depends on the weight of the paper used for each form, and on the carbon coating.

   Because the striking force of the print hammers is not adjustable, paper and carbon should be tested in conjunction with the print-density control lever and the print timing dial.
9. The CR (credit symbol) prints from two print positions and the minus sign prints from one. For this reason the minus sign is recommended as a credit symbol instead of the CR symbol.
10. The dollar symbol does not have to be preprinted on a check form, because this symbol can be programmed to print immediately to the left of significant digits.

Paper Characteristics. The paper used for continuous forms must be of sufficient weight and strength to prevent the holes from tearing out during feeding or ejecting of the form. This is particularly important when single-part forms are being used.

The paper must not be so stiff as to cause improper feeding or excessive bulging, particularly at the outfold.

Paper must be as free from paper dust or lint as possible.

Weight. The number of legible copies required is a factor in determining the weight of the paper to be used in a multiple-part set.

Best results on multiple-copy forms require a lightweight paper of 13 pounds or less, except for the last copy. Again, the number of copies, as well as the distance of the form away from the hammers (this distance can be varied by use of the print-density control-lever), affects the determination of paper weight.

Feeding and legibility performance can best be determined by making test runs of sample sets of forms.

Friction. During the feeding operation, friction on marginally-punched continuous forms should be eliminated by the following means:

1. Place the pack of forms directly beneath the front of the printer on the forms stand, in a position that eliminates any abnormal *drag* on the forms.
2. Allow sufficient clearance between the hammers and the print chain, to permit the forms to be fed by the pins freely, and without interference. This can be accomplished by properly setting the print-density control-lever.

Perforated Lines. The perforations between forms should be sufficiently deep to permit easy separation, but not so deep as to tear in ordinary handling or feeding through the machine.

The perforated lines at the end of the form should always be located at 90 degrees to a vertical center line through the marginal holes.

Cut and uncut portions should be uniformly accurate in length and spacing to insure proper and efficient tearing.

Vertical perforations, at the margin for removal of the marginally punched strip, can vary, depending upon requirements. The distance from the edge of the form to the marginal perforations is usually ½ inch.

Marginal Holes. Continuous forms should have holes in both right and left margins, $\frac{5}{32}$ inch in diameter, spaced vertically ½ inch apart from center to center, the full length of the form. The holes should be located this way on all copies of all sets throughout each pack of forms.

It is possible, however, to use holes of any size, shape, and spacing that accomplish the equivalent feeding conditions.

Vertical lines passing through the two vertical rows of pin holes must be parallel. It is recommended that the edges of the form be ¼ inch from the vertical center lines through the holes.

A horizontal line passing through the center of any two marginal holes on the same line should be at a 90-degree angle to either vertical center lines through the marginal holes.

Spacing between holes, center-to-center, must be such that the pins in the forms tractor, ⅛ inch in diameter and spaced ½ inch apart, enter and leave the holes in the paper, freely without tearing the paper.

Width of Forms. Although forms of any width within the extremes of those shown in Figure 95 can be used, it is recommended that form widths be confined to the standard sizes shown in Figure 96.

Length of Forms Between Perforated Lines. The 1403 accommodates marginally-punched continuous forms up to a maximum length of 22 inches, at 6-lines-per-inch. It is recommended, however, that form lengths be confined to regular lengths, such as 3, 3⅓, 3½, 3⅔, 4, 4¼, 5, 5½, 6, 7, 8, 8½, 10, 11, 12, 14, 16, and 17 inches.

OVER-ALL WIDTH (INCHES)	HOLE-TO-HOLE (INCHES)
4¾	4¼
5¾	5¼
6½	6
8	7½
8½	8
9½	9
10⅝	10⅛
11	10½
11¾	11¼
12	11½
$12\frac{27}{32}$	$12\frac{11}{32}$
13⅝	13⅛
14⅞	14⅜
15½	15
16	15½
16¾	16¼
$17\frac{25}{32}$	$17\frac{9}{32}$

Figure 96. Standard-Size Forms

Line Spacing. The forms tractor of the IBM 1403 can be set by the operator for single-space printing, 6- or 8-lines-per-inch. For 6-lines-to-the-inch spacing, the length of the form must be evenly divisible by ⅙ inch for single spacing, by ⅓ inch for double spacing, and by ½ inch for triple spacing. Similarly, 8-lines-to-the-inch spacing requires that the length of the form be evenly divisible by ⅛ inch for single spacing, by ¼ inch for double spacing, and by ⅜ inch for triple spacing.

Single-space, 8-lines-per-inch printing on the 1403 is not recommended when the registration between lines is critical.

Multiple Copies. Multiple-copy forms consisting of more than four parts, and forms with the first part made of paper of more than 13-pound weight, should be tested under operating conditions to determine the suitability of feeding and legibility.

If multiple-copy forms are not fastened together, the carbon paper must be kept in line with the form by an acceptable method. One such method is center carbon without pin holes, glued to the set, or full-width carbon paper punched with substantially larger marginal holes that are approximately centered with the corresponding holes in the form. Marginal holes in the carbon that are substantially larger than the corresponding holes in the forms make allowance for carbon shrinkage and provide the processing tolerance necessary for some of the commonly used form structures.

One-time carbon paper or carbon-backed paper can be used. The carbon paper or coating should produce the required number of legible copies without excessive smudging. This can be determined best by making test runs with sample sets of forms containing different qualities of carbon papers.

Fastening of Multiple-Copy Forms. The width, length, and number of copies of the form determine the fastening requirements for satisfactory feeding through the forms tractor. For most efficient stacking, however, it is recommended that a suitable fastening method always be used with multiple copy forms.

If the construction of the form is such that the parts are of different widths, the necessity for, and the method of, fastening the form should be determined by the width of the parts, the depth of the form (shown in Figure 97), and weight of paper.

Forms of fanfold construction can be used on the IBM 1403 Printer.

When card-tag or rag-content paper stock is used, a test of sample sets of forms should be made to determine the exact fastening requirements. The fastening may consist of any satisfactory method, such as stitching or gluing, that prevents the copies from shifting. It is essential, however, that whatever fastening medium is used should not impair the feeding or printing alignment of the form.

FORM DEPTH (Inches)	MAXIMUM DISTANCE BETWEEN FASTENINGS (Inches)
1 to 5	5
5-1/5 to 11	11
11 to 14	7
14 to 17	8½

Figure 97. Fastening Requirements for Multiple-Copy Forms

Registration of Forms. The assembly of multiple-copy forms should insure that the punching and printing of all copies of the form are in absolute registration with the material printed by the 1403. The following tolerances should be maintained.

1. *Vertical Lines:* Vertical columns of print positions are spaced 1/10 inch apart. There are 50 printing spaces in 5 inches. Vertical rules printed on a form should be spaced in multiples of 1/10 inch.

The center line of any one character, with reference to any other character on the same line, may have a plus or minus tolerance of .0065 inch, or a maximum over-all tolerance of .013 inch. From a forms viewpoint, it is practically impossible to guarantee that the cumulative tolerance of printing-plate shrinkage, paper shrinkage, and marginal-hole perforations does not exceed .0065 inch. This precludes the possibility of retaining satisfactory registration if vertical rules are spaced to split between print positions.

Where vertical lines are required, such rules should split the respective print position, thereby assigning that particular position for the columnar field (dollars and cents, for example) separation. However, in view of the fact that the 1403 can print special characters such as period and comma in every print position, the use of these symbols as decimal points, etc., avoids the need for vertical lines for such separations.

Vertical printed lines should parallel a vertical center line passing through the marginal holes.

2. *Horizontal Lines:* Horizontal printed lines on the form should be at a 90-degree angle to the vertical center line passing through the paper-feed pin holes.

The spacing should conform to the setting of the 1403 forms tractor—6- or 8-lines-to-the-inch.

3. *Margins:* It is recommended that no staples or other metal fasteners be used with multiple-copy forms. If unavoidable, it is important that either the left or right margin (whichever has the staples) be set outside the print hammer area, so that staples or other metal fasteners do not pass between the chain and hammer unit.

IBM 1403 Operation Codes

Refer to Figure 98 for 1403 operation codes.

Write a Line

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
W	$\overset{\vee}{\mathrm{M}}$	% 20	xxxxx	W

Function. This instruction transfers 100 (or 132) characters in core storage to the print synchronizer. The hundreds position of the x-control field specifies the channel (% is channel 1). The tens position of the x-control fields specifies the input/output unit used (2 is printer). The units position of the x-control field specifies the operation (0 specifies a write operation without word marks). The B-address specifies the high-order (starting) position of the data record (B-field) in storage. Data records are transferred from high-order to low-order position (left to right). The operation is stopped by the first group-mark with word-mark sensed in core storage. At the end of a correct data transfer, the printer is started and prints a line with the data just transferred. If the $\overset{\vee}{\mathrm{L}}$ op code is used, word marks are translated to word-separator characters during the transfer to the print synchronizer. There is no character on the chain for a word-separator character and a blank space results during the printing. When operation is in the LOAD mode, a blank space (corresponding to the position that contained the word-separator character) appears ahead of its associated character. This *increases* the result field length in the print synchronizer and may cause the loss of some of the field.

Word Marks. A group-mark with word-mark must appear in the core-storage position to the immediate right of the data record.

Timing. T = 49.5 + I/O.

(See IBM 1403 *Printer Timing Considerations.*)

Note. A BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction, $\overset{\vee}{\mathrm{R}}$ (I) ⋕, must be given before the next channel 1 I/O unit operation, to be sure that the data was properly transferred.

(A specific $\overset{\vee}{\mathrm{R}}$ (I) d instruction may be substituted for the $\overset{\vee}{\mathrm{R}}$ (I) ⋕ instruction. This will not cause a system interlock as long as a branch to the I-address, specified in the $\overset{\vee}{\mathrm{R}}$ (I) d instruction, occurs.)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	B + LB + 1

Write Word Marks

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
WM	$\overset{\vee}{\mathrm{M}}$	% 21	xxxxx	W

Function. This instruction transfers all word marks in the data field to the print synchronizer as digit ones. The one in the units position of the x-control field specifies the write word mark operation. Positions without word marks are transferred to the print synchronizer as blanks. At the end of the data transfer, the printer is started and prints a line with the word mark data just transferred. If the $\overset{\vee}{\mathrm{L}}$ op code is used, the effect on printing is the same as for WRITE A LINE.

Word Marks. A group-mark with a word-mark must appear in the core-storage position to the immediate right of the data record.

I/O UNIT	OP CODE	X-CTRL FIELD	DESCRIPTION	MNEMONIC	d-CHARACTER		OPERATION	NOTES	
					CHARACTER	CONTROL			
PRINTER	$\overset{\vee}{\mathrm{M}}$ or $\overset{\vee}{\mathrm{L}}$	%20	Write a line	W or WW	W	. . . .	Transfer 100 or 132 characters from storage to print synchronizer and print a line.	Word marks in storage area are undisturbed.	If $\overset{\vee}{\mathrm{L}}$ op code is used word marks are translated to word separator characters for printing.
		%21	Write word marks	WM	W	Print word marks as a digit 1	Transfer word marks from storage to print synchronizer and print word marks.	Positions that do not contain word marks are blanks.	
CARRIAGE	$\overset{\vee}{\mathrm{F}}$		Carriage control	CC	See Figure 99 for list on d-characters and their associated operations.				

Figure 98. IBM 1403, Op Codes

Timing. $T = 49.5 + I/O$.

(See IBM 1403 *Printer Timing Considerations.*)

Note. A BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction, $\overset{\vee}{R}$ (I) ⧧, must be given before the next channel 1 I/O unit operation, to be sure that the data was properly transferred.

(A specific $\overset{\vee}{R}$ (I) d instruction may be substituted for the $\overset{\vee}{R}$ (I) ⧧ instruction. This will not cause a system interlock as long as a branch to the I-address, specified in the $\overset{\vee}{R}$ (I) d instruction, occurs.)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	B + LB + 1

Control Carriage

Instruction Format.

Mnemonic	*Op Code*	*d-character*
CC	$\overset{\vee}{F}$	x

Function. The tape-controlled carriage is instructed to skip to the channel specified by the d-character. The numerical portion of the d-character specifies the number of spaces to be taken, or the tape channel hole that terminates the skip. The d-characters and the operations they initiate are shown in Figure 99.

An automatic single-space is initiated at the completion of a successful data transfer from the CPU, *only* if the forms have not been moved since the last print line. To prevent the automatic space, forms may be moved by either a forms operation or by pressing the space or restore keys on the 1403.

Word Marks. Word marks are not affected.

d	IMMEDIATE SKIP TO	d	SKIP AFTER PRINT TO	d	IMMEDIATE SPACE
1	Channel 1	A	Channel 1	J	1 Space
2	Channel 2	B	Channel 2	K	2 Spaces
3	Channel 3	C	Channel 3	L	3 Spaces
4	Channel 4	D	Channel 4		
5	Channel 5	E	Channel 5		
6	Channel 6	F	Channel 6	d	SPACE AFTER PRINT
7	Channel 7	G	Channel 7		
8	Channel 8	H	Channel 8	/	1 Space
9	Channel 9	I	Channel 9	S	2 Spaces
0	Channel 10	?	Channel 10	T	3 Spaces
#	Channel 11	•	Channel 11		
@	Channel 12	⌑	Channel 12		

Figure 99. d-Character for Control-Carriage Instruction

Timing. $T = 13.5$.

(See IBM 1403 *Printer Timing Considerations.*)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	Bp

Figures 100 and 101 are the I/O channel-status indicators set during IBM 1403 operations.

INDICATOR	d-CHARACTER BIT	CONDITION
Not Ready	1	Printer not ready, Printer not on line, Printer power off, Printer out of forms
Busy	2	Previous line still being printed
Data Check	4	Print synchronizer detects parity error (line is not printed)
Condition	8	Print synchronizer detects timing error, Print synchronizer detects hammer fire check (line is not printed)
Wrong Length Record	B	Wrong length record (line is not printed)
No Transfer	A	Never set

Figure 100. I/O Channel-Status Indicators Set During Write a Line and Write Word Marks Operation

INDICATOR	d-CHARACTER BIT	CONDITION
Not Ready	1	Printer not ready, Printer not on line, Printer power off, Printer out of forms
Busy	2	Forms in motion forms instruction waiting to be executed
Data Check	4	Never set
Condition	8	
Wrong Length Record	B	
No Transfer	A	

Figure 101. I/O Channel-Status Indicators Set During IBM 1403 Carriage Operation

IBM 1403 Printer, Timing Considerations

The transfer of data from the print area of core storage to the print synchronizer requires 1,100 microseconds for 100 print positions, and 1,452 microseconds for 132 print positions. The printer is not busy at this time; BUSY comes ON at the successful completion of the transfer. It remains ON for a minimum of 82,420 microseconds if there is not an automatic space, or a minimum of 103,820 microseconds if there is an automatic space. In case of an unsuccessful transfer, the printer may be readdressed immediately by the CPU; however, the second data transfer will not actually start until 1,463 microseconds after the initiation of the first transfer.

Special Features

Numerical Print Feature

The numerical print feature for the IBM 1403 Printer has been designed for those businesses having certain 1410 applications that require no alphabetic printing. For example, banks, insurance companies, and utilities prepare many reports with only numerical printing. With this feature, the time required to produce these reports can be reduced by as much as 50 per cent. The manufacturing, wholesaling, and retailing levels of other industries can also use this feature for the many applications in which reports are (or can be) numerically coded.

With this feature, the systems user can switch from the alphamerical to the numerical mode, simply by changing the chain cartridge in the 1403. The numerical chain is composed of 15 character sets, with 16 characters (digits 0 through 9 $. , * - ⌑) in each set. In the numerical mode, the 1403 can print 1285 lines per minute — more than twice as fast as in the alphamerical mode.

To change from one mode to another, an operator, with no special tools, removes one chain and replaces it with the other. Before locking the new cartridge in place, it is only necessary to move the chain enough to permit the chain drive to engage. When a chain cartridge is placed in the 1403, the corresponding mode is selected automatically. If the printer is in the numerical mode, characters other than the 16 specified for numerical printing cause a print check error.

Interchangeable Chain Cartridge Adapter

Many scientific and commercial applications require distinctive type styles for particular printing jobs. This special feature for the IBM 1403 Printer allows chain cartridges to be interchanged.

With this feature, an operator can insert an interchangeable chain cartridge with a different type font, type style, or special character arrangement.

The procedure for changing a cartridge is:

1. Turn off system power.
2. Lift up the printer cover.
3. Pull back and unlock the print unit release lever.
4. Unlatch the ribbon shield and swing it against the paper.
5. Open the ribbon cover and remove the lower ribbon spool. Slide ribbon from under the skew roll and store the lower ribbon spool on the ribbon cover.
6. Grasp the cartridge handles and raise them to a vertical position. (This unlocks the cartridge from the T-casting.)
7. Lift straight up on the handles and raise the cartridge until it clears its locating pins. At this point it is free from the machine. Place the cartridge on a surface that will tolerate oil and ink. (A container is provided for storing the cartridge that is not in use.)
8. Grasp the handles of the second interchangeable cartridge and, raising them to a vertical position, lift the cartridge into position over the locating pin. (Check for foreign matter clinging to underside of cartridge.)
9. Lower the cartridge gently into position over its guide pins and release the handles *(do not force either handle down at this point)*. The 132-hammer end of the cartridge should settle fully down to the base. The 1-hammer end will not be down in position at this time.
10. Rotate the chain in the normal printing direction (counterclockwise, as viewed from the top). The chain can be rotated by pressing your finger against a character on the chain. At the same time, apply pressure to the button (located between the print-timing dial and the cartridge) on the top cover. Rotate the chain slowly until the drive key drops into the drive slot. The chain will stop and the cartridge will settle correctly into position on the 1-hammer end.
11. Lower the cartridge handles to their horizontal position. *Do not force.* If force is required, the cartridge is not fully seated; repeat steps 8 to 10.
12. Replace the ribbons; latch the ribbon shield into place; close the T-casting and the top cover; apply power to the system and resume printing.

IBM 1410 System — Magnetic Tape

The three system components (1411, 1414, 1415) or a card-oriented IBM 1410 system can be expanded to form a tape-oriented IBM 1410 system. Up to 10 IBM 729 II, IBM 729 IV (Figure 102), or IBM 7330 Magnetic Tape Units (Figure 103) can be attached to the system's data transmission channel (1 additional channel available). The 729 II and 729 IV tape units may be intermixed on the same data transmission system. The 7330 may be intermixed on the same channel with 729 II and IV units if the Tape Intermix Feature is installed. The 729 and 7330 operating principles and functions of the keys and lights are identical. Figure 104 shows significant operating characteristics of the 729 II and IV and the 7330.

Magnetic-Tape Operating Principles

Magnetic tape is a special plastic tape, coated on one side with a layer of magnetic oxide material. Data are recorded on the magnetic oxide of the tape in the form of magnetized spots or bits. Information written on tape remains there indefinitely, or until the tape is used in a new write operation. When the recorded information is no longer needed, the tape can be used to record new data. The write operation automatically erases old information. Reflective spots, manually placed on the tape, are photoelectrically sensed to indicate the beginning (or load point) and the physical end of the useful portion of the tape. The load-point reflective spot is about 12 feet from the front end of the tape, and the reflective spot designating the end of the usable tape is 18 feet from the physical end. Tape is wound on plastic reels 10½ inches in diameter. A full reel contains about 2,400 feet of usable tape, but lengths as short as 50 feet can be used.

During reading or writing, tape is moved from the file reel, through the left vacuum column across the read-write head, through the right vacuum column, to the machine reel.

Reading or writing on a tape takes place while the tape moves across the read-write head. The vacuum columns control separate drive motors and permit the read-write mechanism and each one of the two tape reels to move tape independently of the other two units. The read-write mechanism feeds tape according

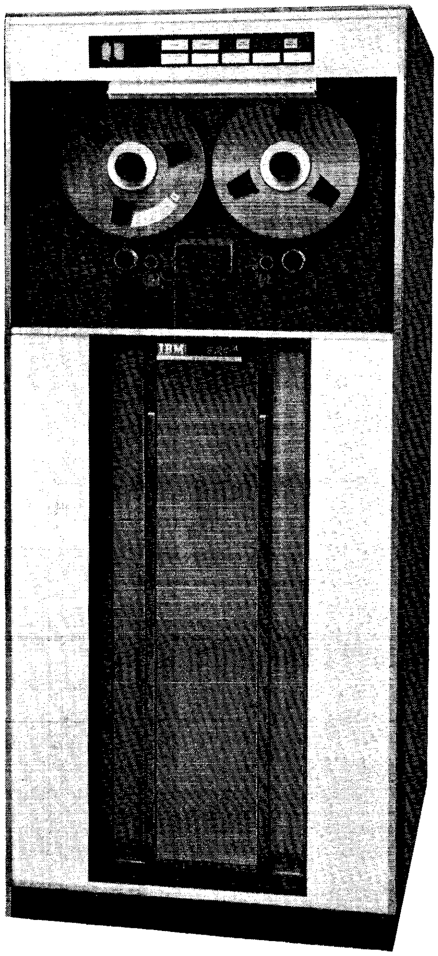


Figure 102. IBM 729 Magnetic Tape Unit, Model II

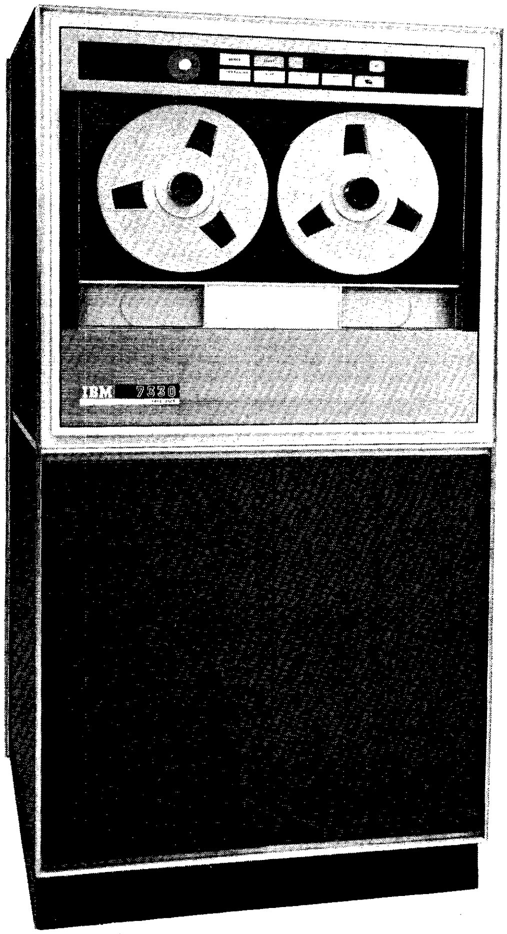


Figure 103. IBM 7330 Magnetic Tape Unit

OPERATING CHARACTERISTICS	729-II	729-IV	7330
Density, Characters Per Inch	200 or 556	200 or 556	200 or 556
Tape Speed, Inches Per Second	75	112.5	36
Inter-Record Gap Size, Inches	3/4	3/4	3/4
Character Rate, Characters Per Second	15,000 or 41,667	22,500 or 62,500	7,200 or 20,016
High Speed Rewind, Minutes	1.2	.9	2.2
Regular Rewind, Inches Per Second	75	112.5	36

Figure 104. Tape Unit Characteristics

to instructions for the stored program. The file reel feeds tape when the tape reaches a minimum slack point in the vacuum column, and the machine reel winds tape when the slack tape reaches a maximum low point in the vacuum column.

The head assembly, located between the vacuum columns, is built in two sections. The lower section is stationary, but the upper section can be moved up or down under control of the tape-unit keys. When the upper section is up, the operator can thread tape. When down, it places the read-write head in close contact with the tape for reading or writing. The tape reels and head are accessible when the reel door is open.

REFECTIVE SPOTS

Reflective spots, also referred to as photosensing markers, are placed on the tape to enable the tape unit to sense where reading and writing are to begin and to stop. The markers are small pieces of plastic, one inch by 3/16 inch, coated with vaporized aluminum on one side and with adhesive on the other. They are fastened to the base (uncoated) side of the tape. The photoelectric cells sense them as either the load-point marker where reading or writing is to begin on tape, or as the end-of-reel marker where reading or writing is to stop.

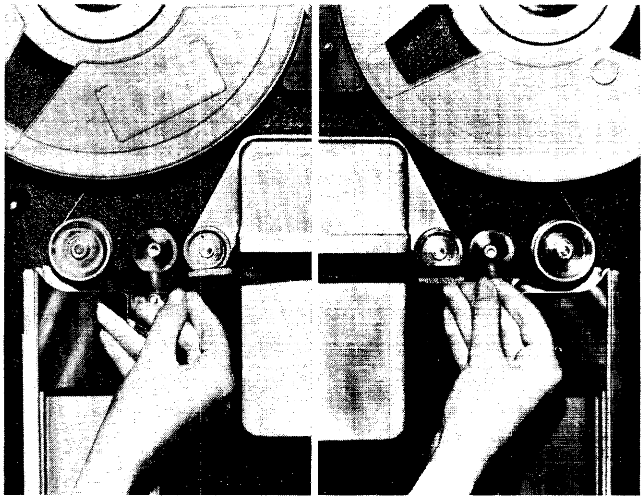


Figure 105. Reflective Spots on Tape

There must be about 12 feet of tape between the beginning of the reel and the load-point marker. This footage is used to thread the tape over the feed rolls and the read-write head. Information must not be stored in this space. To indicate the load point, the one-inch dimension of the marker must be parallel to, but not more than 1/32 inch from, the channel 1 edge of the tape – the edge nearer the operator when the reel is mounted (Figure 105).

About 18 feet of tape should be reserved between the end-of-reel marker and the physical end of the tape attached to the hub of the machine reel. To indicate end-of-reel, the marker must be placed parallel to, but no more than 1/32 inch from, the C-track edge of the edge of the tape (the edge nearer the tape unit when the reel is mounted. See Figure 105).

Place load-point and end-of-reel markers on tape with care. They should be properly aligned and pressed tightly onto the tape with back of the fingernail. To reduce the collection of dust on the unrolled tape, place markers while the tape is loaded on a unit. If this is done away from the unit, keep the unrolled end of tape off the floor and away from dusty areas.

FILE-PROTECTION RING

The back of the tape reel (machine side) has a circular groove that can hold a plastic ring (see Figure 106). This ring (called the *file-protection ring),* when inserted, permits writing on tape. A tape can be read whether or not the ring is inserted. The file-protection

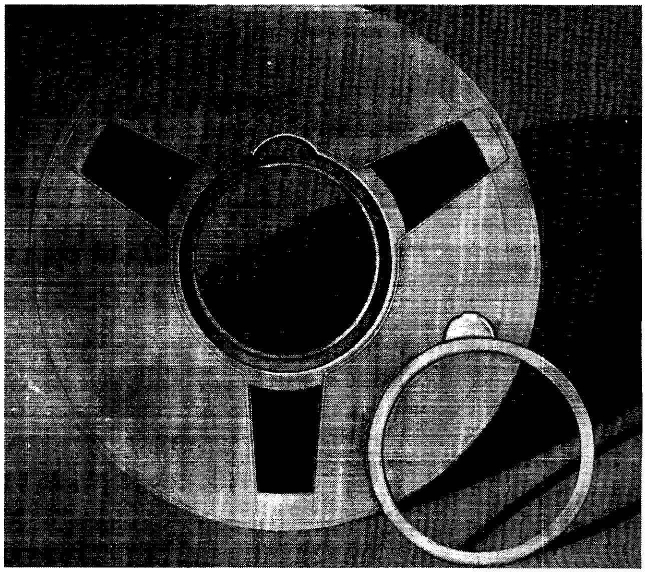


Figure 106. File-Protection Device

ring should be removed from the tape reel after writing on tape is completed thereby preventing accidental writing and a resultant loss of valuable tape records. Never remove the file-protection ring while tape is loaded in the vacuum columns of the tape unit, because this could cause a broken or damaged tape.

Tape Characteristics

Data are recorded in a 7-bit code, in seven parallel channels along the tape. Figure 107 shows tape characters and their corresponding codes.

Records are separated from each other by about ¾ inch of blank (unrecorded) tape, called an *inter-record gap.*

Each tape character is composed of an even number of magnetic bits. A check bit (labeled C in Figure 107) is written if the number of bits in the other six positions is odd. An even-parity check on each character insures accuracy for tape-read and tape-write operations.

Besides this vertical parity check, a horizontal check (HC in Figure 108) is made on each record. The bits in each horizontal row are automatically counted when the record is written, and a bit (similar in function to the vertical-check bit) is written at the end of each odd-count row. The vertical combination of these horizontal-check bits makes up the horizontal-check character. Thus, the coding of this character can change from record to record. When the tape is read, the same automatic count is made; but now each row in the complete record should have an even number of bits, or an error exists. The horizontal-check character is used for checking only and is never read into core storage. Odd-bit redundancy tapes (binary tapes) can be accepted by and written by a 1410 system.

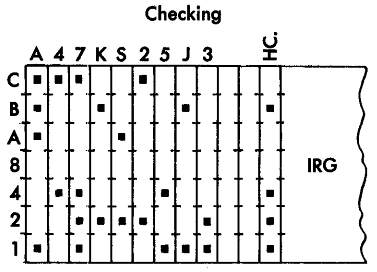


Figure 108. Vertical and Horizontal Check

Tape Checking

The IBM 729 and 7330 tape units have increased reliability through two features: the two-gap head, and dual-level sensing. The two-gap head makes it possible to automatically verify the validity of recorded information at the time it is written. The relative positions of the read and write gaps (Figure 109) are such that a character recorded by the write gap passes the corresponding read gap; and thus, when each character of a record is written, it is read, and a parity check is applied. If an error is detected, the stored program receives a signal, and corrective action can be taken. The ability of the two-gap head to read tape in both reading and writing operations makes it possible to check these operations by dual-level sensing. The read head reads the tape at two levels of pulse strength:

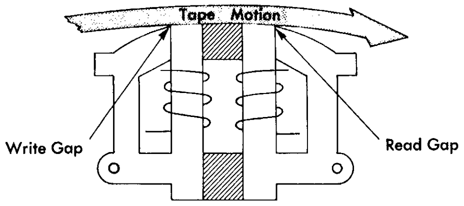


Figure 109. Read and Write Gap

*Whenever the C and A bit combination on even parity tape is read into core storage, it comes in as a C bit only. Since the A bit is a valid 1410 d-character modifier (move instructions), whenever programs are to be written on tape, the odd parity mode should be used. At all other times, it is recommended that even parity tape be used. In odd parity mode, the blank character is represented on tape by a C bit and in 1410 core storage by a C bit. The ¢ character is represented on tape by an A bit and in 1410 core storage by an A bit.

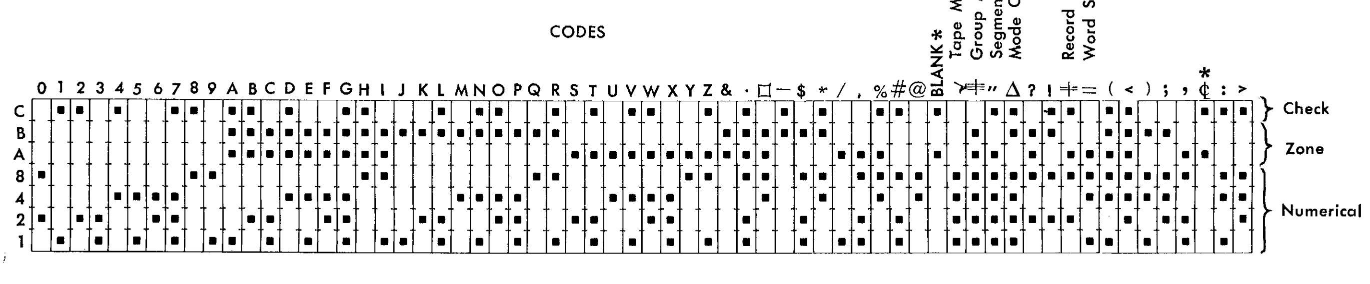


Figure 107. Magnetic-Tape 7-Bit Coding

high and low. There is a 7-position high register and a 7-position low register (one position for each tape channel). In a tape-read operation, the high-sensitivity level register is checked for even parity. If there is an odd number of bits, the contents of the low-sensitivity level register are sent to the read-write register.

The contents of the read-write register are sent to core storage. Detection of a validity check sets on a validity-check indicator which in turn sets on the data check I/O channel status indicator. The data check I/O channel status indicator can be interrogated by use of the d-character 4 in the instruction, BRANCH IF DATA CHECK I/O CHANNEL STATUS INDICATOR ON. Thus, a bit that results in a weak but valid signal can be read from tape. If the character is still invalid, a validity-check signal is given.

In checking tape-write operations, the unit becomes harder to satisfy by automatically making the high register less sensitive than it is for tape read. (It still has a higher sensitivity level than the low register, however.) Each tape character written is read back and must be valid in both registers. The contents of the high register are validity-checked and then are matched, bit for bit, with the contents of the low register.

If the validity check in the high register detects an odd number of bits, or if the bit-by-bit match between registers is unequal, a validity-check signal is given.

Figure 110 shows the sensitivity levels and the relative strength of pulses that are acceptable or not acceptable in read and in write conditions.

If a tape error is suspected, the tape unit can be back-spaced by programming, and the record can be re-read. If the error persists, the operator can intervene, or the program can branch to an error routine.

Dust or damage to the magnetic tape are the most frequently detected errors that occur during write operations. Such imperfections are usually isolated. In order to skip the defective section, the 1410 has an instruction that causes the tape to space forward about 3.5 inches when the next write operation is initiated. During the space operation, this 3.5 inch area is erased so that extraneous data are not sensed during succeeding read operations. The tape-write operation continues after the skip is completed.

When writing from load point, a space of 3.5 inches occurs prior to writing the record and start time is increased by about 27 milliseconds.

IBM 729 and 7330 Magnetic Tape Units Operating Keys and Lights

The operating keys and lights of the IBM 729 and 7330 magnetic tape units are located at the top of the unit, above the tape reels (Figure 111). The lights are all on the upper row, and the keys are on the lower row. The address selection dial is at the left.

DENSITY SWITCH

This selects high or low density operation, depending on the tape operating mode desired.

ADDRESS SELECTION DIAL

This dial assigns a number, from 0 to 9 and blank, to the tape unit, to identify it to the stored program. The setting should not be changed when a tape operation is in progress.

SELECT LIGHT

The select light turns ON automatically, when the address selection dial is properly positioned and the unit is addressed by the computer whether the computer is ready or not.

READY LIGHT

This light, when ON, indicates that the tape unit is ready for operation. See *Start Key* for method of turning this light on. The reel door should never be opened when the ready light is ON.

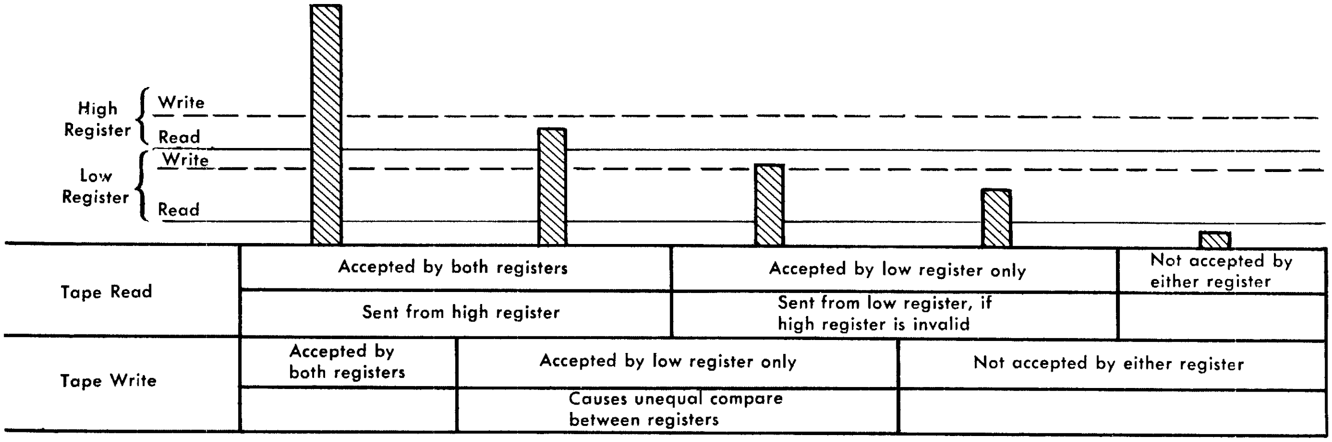


Figure 110. Relative Sensitivity Levels

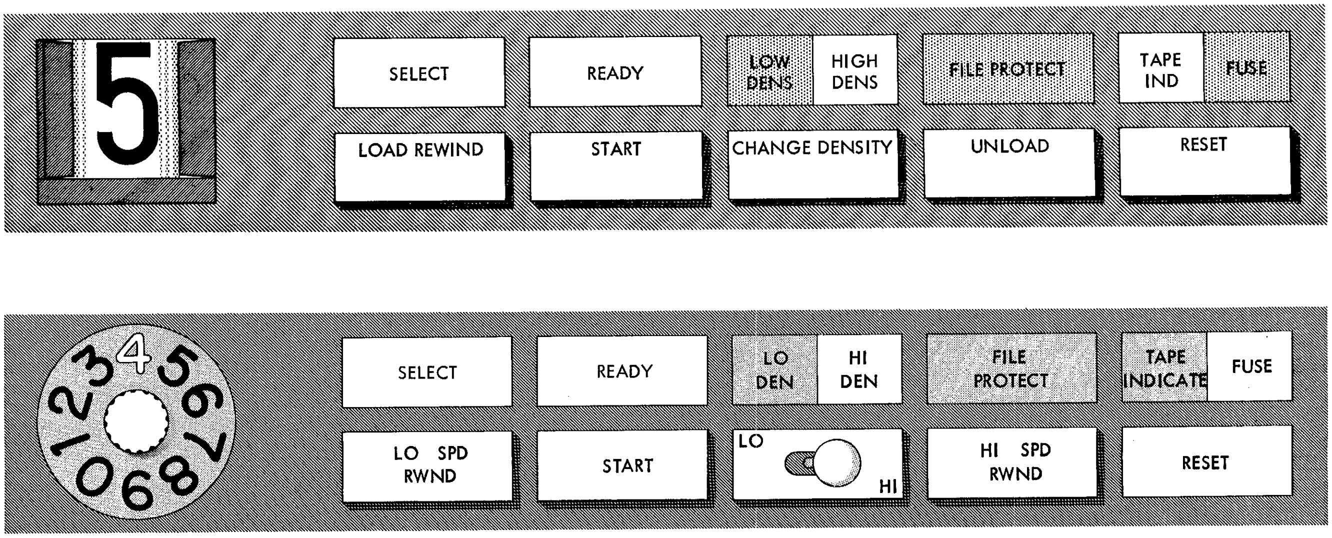


Figure 111. IBM 729 and 7330 Magnetic Tape Units, Operating Keys and Lights

TAPE INDICATE ON LIGHT

This indicator is turned ON by:

1. sensing the end-of-reel marker while writing on tape
2. sensing the tape mark while reading tape.

The indicator can be turned off by:

1. pressing the unload key on the tape unit
2. executing a rewind tape and unload operation or executing a BRANCH IF CONDITION I/O CHANNEL STATUS INDICATOR ON (end of file) instruction in the stored program. (Because of the rapid internal processing speed in executing the instructions that turn the light off, it may appear as if the light was never turned on.)

FILE-PROTECTION LIGHT

This light automatically turns on if the unit is loaded with a reel that does not have the file-protection ring inserted in the back of the reel. The tape cannot be written as long as the file-protection light is ON. This light is ON whenever the tape unit is not in ready status.

FUSE LIGHT

This light turns on automatically whenever a fuse in the unit has blown.

TAPE DENSITY LIGHTS

These two lights (high and low) indicate the density in which the tape unit is operating. They are controlled by the setting of the density switch on 7330's, and by operating a button on the 729's.

LOAD REWIND KEY

This key is operative only when the reel door is closed and the ready light is OFF. Use of this key, after tape has been properly mounted in the magnetic tape unit, lowers tape into the columns, lowers the head assembly, and moves tape in the rewind direction until the load point reflective spot is sensed. If the reflective spot is not to the right of the read-write head when this key is pressed, the tape will unwind from the machine reel.

Caution: Do not open the reel door during rewind or load point searching.

START KEY

Use of this key places the tape unit in ready status and turns on the ready light, provided that:

1. the reel door is closed
2. tape has been loaded into the columns
3. the tape unit is not in the process of finding the load point (rewind or load-point operation).

UNLOAD KEY

This key is operative only when the ready light is OFF, tape is in the vacuum columns, and the reel door is closed. Use of this key raises the head assembly, and removes the tape from the columns, regardless of the distribution of tape on the two reels. If the tape is not at load point when the operator wishes to change tape reels, a load-point search should be initiated first by pressing the load-rewind key. Pressing the unload key also turns off the tape-indicate-on light.

RESET KEY

On a 729 II or IV, this key turns OFF the ready light. It also stops any tape operation except load and unload. If this key is pressed during a high-speed rewind, the operation stops, and then continues as a slow-speed rewind. If the reset key is pressed during a slow-speed rewind, the operation stops.

On a 7330, this key turns OFF the ready light. It also stops any operation being performed.

REEL DOOR INTERLOCK

When the door is open, the interlock contact prevents any normal operation of the tape unit. The reel door should never be opened when the ready light is ON, or during any load-rewind operation.

REEL RELEASE KEY

When this key is pressed, both reels may be turned manually for threading tape or removing the file reel. To operate the reel release key, open the reel door.

Note: Steps required to place a 7330 tape unit in ready status after a high-speed rewind are:

1. Open the reel door.
2. Press and hold the reel release button through step 5.
3. Manually rotate the take-up reel a few times until the load point is on the reel.
4. Move the read-write head lever to a vertical position. This will lower the head.
5. Rotate each reel, as necessary, to move the tape into the vacuum columns properly.
6. Close the reel door.
7. Press the load rewind and start keys.

Operating Pointers

Consider the following points whenever a tape unit is in operation:

1. Do not change the address of a tape unit by operating the address selector switch during the execution of a program that uses other tape units. This applies whether the unit is in ready status or not.
2. Never set two tape units to the same address.
3. Do not open the door of a tape unit unless the tape is out of the vacuum columns and the read-write head is raised.
4. In the event of a power failure with tape units in ready status, have an IBM customer engineer remove the tape from the read-write head and the vacuum columns of every unit in ready status before power is restored.
5. Do not operate the DC OFF key on the 1410 with the tape units in ready status, because extraneous noise may be recorded on the tape when DC is turned on.
6. Rules that *must be followed* for tape unit operations are:

IBM *7330 Tape Units.* When a tape unit is in write status, the tape unit *cannot* be switched into read status unless preceded by a backspace.

IBM *7330 Tape Units.* If a tape unit is in read status, and a write operation is called for, the system must be programmed to: backspace the last record read, re-write this record, and then continue in write status. In a write, backspace, read, write sequence, the backspace causes an unchecked erase forward (occurs only on first backspace after write). This may be used with discretion.

Magnetic Tape Control Instructions

These instructions control reading and writing magnetic tape (with or without word marks), backspacing tape, writing tape marks, rewinding tape reels, and skipping over defective areas. There are three types of magnetic tape control instructions — UNIT CONTROL, READ AND WRITE TAPE, and READ AND WRITE TAPE WITH WORD MARKS. They are shown in detail in Figure 112.

Unit Control

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*d-character*
(See Figure 112)	$\overset{\text{v}}{\text{U}}$	% Un	x

Function. The tape unit, specified by the unit-position n-character in the x-control field, performs the operation indicated by the d-character. The d-characters and the operations they initiate are:

Operation	*d-character*
Backspace tape record	B
Skip and blank tape	E
Write tape mark	M
Rewind	R
Rewind and unload	U

Word Marks. Word marks are not affected.

Timing. $T = .0045\ (L + 1) + T_m$ ms.

(See *Tape Timing Considerations.*)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	Bp

Read or Write Tape

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
RT or WT	$\overset{\text{v}}{\text{M}}$	% Un	xxxxx	R or W

Function. The n-character in the x-control field specifies the tape unit that performs the operation. The B-address specifies the high-order position of the tape record core-storage area. If the d-character is R, the tape record is read into core storage from the tape. If the d-character is W, the tape record in core storage is written on the tape. Either an inter-record gap in the tape record or a group-mark with a word-mark in core storage (whichever is sensed first) stops a read tape operation. If a group-mark with a word-mark is sensed first, the data transfer stops,

	OPERATION CODE	MNEMONIC	d-CHARACTER	OPERATION	NOTES
$\overset{\vee}{U}$	Backspace tape	BSP	B	Tape unit backspaces over one complete tape record.	A tape mark (8421) is considered a tape record. The 1410 is not interlocked during the operation.
	Skip and blank tape	SKP	E	Erases about 3.5 inches of tape before next tape-write operation. A tape-read operation cancels the operation.	The next instruction should be a tape-write operation for the same tape unit.
	Write tape mark	WTM	M	A tape mark (8421) is written on tape as a single-character record.	The 1410 is interlocked during the operation.
	Rewind	RWD	R	Tape unit rewinds, loads its tape, and positions itself at load point.	At the completion of the operation, the tape unit is in a ready status at load point. (Always low speed on 7330.)
	Rewind and unload	RWU	U	Tape unit rewinds and unloads its tape.	At the completion of the operation, the tape unit is effectively disconnected. (High speed on 7330 — requires manual reloading.)
$\overset{\vee}{M}$	Read tape	RT	R	A record is transferred from magnetic tape to core storage.	
	Write tape	WT	W	A record is transferred from core storage to magnetic tape.	
$\overset{\vee}{L}$	Read tape with word marks	RTW	R	A record with word marks is transferred from magnetic tape to core storage.	
	Write tape with word marks	WTW	W	A record with word marks is transferred from core storage to magnetic tape.	
$\overset{\vee}{L}$	Read tape	RTG	$	Read from magnetic tape to core-storage. Group-marks with word-marks in core-storage have no effect on operation.	Stop transfer when inter-record gap is sensed or last core-storage position is encountered.
$\overset{\vee}{L}$	Write tape	WTE	X	Contents of core-storage are written on tape. Group-marks with word-marks in core-storage have no effect on operation.	Stop transfer when last core-storage position is encountered.

Figure 112. Magnetic-Tape Control Instructions

but the tape movement continues to the next inter-record gap. A group-mark with a work-mark in core storage stops the write operation and causes an inter-record gap on the tape.

Word Marks. Word marks in the tape record do not affect either operation.

Timing. $T = .0045 (L + 1) + T_m$ ms.
(See *Tape Timing Considerations.*)

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	B + LB + 1

Note. If the x-control field contains %Bn, the magnetic-tape operation is performed in an odd-parity mode. A tape mark is always even parity. If a tape mark is encountered during an odd-parity operation, a data check and an end-of-file indication result. If ASTERISK CONTROL is on, an asterisk is placed in core storage. Add a B-character to the RT or WT mnemonic to form a binary mnemonic. In magnetic tape operation, the character in the hundreds position of the x-control field indicates the channel and overlap mode: %, Ch 1, non-overlap; @, Ch 1, overlap; ⌑, Ch 2, non-overlap; *, Ch 2, overlap.

Read or Write Tape with Word Marks

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
RTW or WTW	$\overset{\vee}{L}$	% Un	xxxxx	R or W
RTG or WTE	$\overset{\vee}{L}$	% Un	xxxxx	$ or X

Function. The first set of instructions has the same function as the read tape and write tape instructions

1410 core-storage location	A	B	C	
1410 core-storage code	C82	41W	4	
1410 meaning	0	$\overset{v}{5}$	4	
Tape positions	A	B	C	D
Tape code	82	A841	41	C4

Figure 113. Word Mark Translation for Write Tape with Word Marks

Tape positions	A	B	C	D
Tape code	82	A841	41	C4
1410 meaning	0	$\overset{v}{5}$	4	
1410 core-storage location	A	B	C	
1410 core-storage code	C82	41W	4	

Figure 114. Word Mark Translation for Read Tape with Word Marks

except that word marks are written on tape as word-separator characters (A841), and word-separator characters are read into storage as word marks. In a write operation, a word-separator character is written on tape, one position ahead of the associated character (Figure 113). In a tape read operation, word marks are associated with the next character read from tape (Figure 114).

Word Marks. Each word-separator character requires a one-tape position.

Timing. $T = .0045 (L + 1) + T_m$ ms.

(See *Tape Timing Considerations.*)

Note. This instruction is used whenever word marks must be indicated in the tape record. If a tape record is written with word marks, it must be read with word marks when the data are required for a subsequent operation. This assures proper translation between the tape and core storage.

These instructions can be done in an odd-parity mode. The x-control field and mnemonics are handled as described in read or write tape.

Address Registers after Operation.

I-Add. Reg.	*A-Add. Reg.*	*B-Add. Reg.*
NSI	Ap	B + LB + 2

The second set of instructions operates in the same manner as previously described for the first set of instructions. An X d-character initiates a tape write operation. Starting at the core-storage location specified, the contents of core storage are written on tape until the last core-storage position is encountered. A $ d-character initiates a tape read operation. Characters are read from tape and stored in core storage until an inter-record gap on the tape is sensed or the last core-storage position is encountered.

Note. Instructions using the $ or X d-character cannot be overlapped.

Tape Unit Operation Status Indicators

A BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction, $\overset{v}{R}$ (I) ╪, must be given before the next I/O unit operation, on the channel being used, to be sure that the data was properly transferred.

(A specific $\overset{v}{R}$ (I) d instruction may be substituted for the $\overset{v}{R}$ (I) ╪ instruction. This will not cause a system interlock as long as a branch to the I-address, specified in the $\overset{v}{R}$ (I) d instruction, occurs.) Figure 115 is the I/O channel-status indicators set during tape operations.

INDICATOR	d-CHARACTER BIT	CONDITION
READ(R),WRITE(W), CONTROL UNIT(U)		
Not Ready (R-W-U)	1	Tape unit not ready No such tape unit selected Tape adapter unit not on line Tape adapter unit power off
Busy (R-W-U)	2	Tape unit rewinding Tape adapter unit busy (backspace or 7330 read-write not finished)
Data Check (R)	4	Processing unit received wrong parity character Tape adapter unit sent wrong parity character Tape mark read in odd parity mode
(W)		Tape adapter unit received wrong parity character Tape adapter unit detects rbc parity error
(U)		Set if write tape mark in odd parity
Condition (R)	8	1st character of record was tape mark
(W)		Foil strip detected
(U)		Never set (unless tape mark read)
Wrong Length Record (R)	B	Wrong length record (always set when d-character is $)
(W-U)		Never set
No Transfer (R-W-U)	A	Never set

Figure 115. I/O Channel-Status Indicators Set During Tape Operations

Tape Timing Considerations

All tape units on a given channel in a 1410 system are under control of a tape adapter unit (TAU). TAU can control the operations of only one tape unit at a time. If one tape unit is busy, no other tape unit can be used until all operations on the busy one have been completed (except rewinding). The execute times of IBM 1410 tape instructions vary according to the type and model of tape units used in the system (Figure 116).

C is the character rate in milliseconds based on the setting of the tape density switch.

N is the number of characters in the record.

CN is record time (number of characters in the record, times the character rate).

Start time is the time necessary for the tape unit to accelerate to operating speed.

Stop time is the time necessary for the tape unit to decelerate and stop.

Record check time is the time it takes to read or write the check character. This time is based on the read-write head gap (the distance that separates the read and write heads) and the time it takes a single character written on tape to travel from the write head to the read head.

T_m — Tape movement can be determined from the following:
1 = Number of Characters
C = Character Rate
729 II at 200 cpi = .067 ms
at 556 cpi = .024 ms
729 IV at 200 cpi = .044 ms
at 556 cpi = .016 ms
7330 at 200 cpi = .139 ms
at 556 cpi = .050 ms

729 Model II, Read 10.7 + CN ms = TAU and Processing interlocked
Write 11.7 + CN ms = TAU and Processing interlocked

729 Model IV, Read 6.8 + CN ms = TAU and Processing interlocked
Write 7.8 + CN ms = TAU and Processing interlocked

7330 Read 20.5 + CN ms = TAU interlocked
8.1 + CN ms = Processing interlocked
Write 20.3 + CN ms = TAU interlocked
13.8 + CN ms = Processing interlocked

Rewind
729 Model II = 1.2 minutes/reel
729 Model IV = .9 minutes/reel
7330 (High Speed) = 2.2 minutes/reel

Skip and Blank Tape
(add to subsequent write time)
729 Model II = 40.5 ms
729 Model IV = 27 ms
7330 = 103 ms

Backspace (after Read)
729 Model II = 46 + CN ms
729 Model IV = 33 + CN ms
7330 = 428 + CN ms

Backspace (after Write)
729 Model II = 52 + CN ms
729 Model IV = 37 + CN ms
7330 = 435 + CN ms

Figure 116. Tape Movement Specifications

IBM 729 II Tape Timings

During a 729 II *read* operation, the tape adapter unit and the processing unit are interlocked for 10.7 + CN ms (Figure 117). This includes:

10.5 ms — start time
.2 ms — record check time for high-density tape (.6 ms for low-density tape)
CN ms — record time

Therefore, in a tape read operation, processing can take place during 1.9 ms of stop time.

During a 729 II *write* operation, the tape adapter unit and the processing unit are interlocked for 11.7 + CN ms (Figure 117). This includes:

7.5 ms — start time
4.2 ms — record check time for high-density tape (4.6 ms for low-density tape)
CN ms — record time

Therefore, in a tape write operation, processing can take place during .9 ms of stop time. The difference between the .2 ms record check time of read tape operations and 4.2 ms record check time of write tape operations is due to the read-write head gap time (4.0 ms).

For job timing estimates of tape read and write operations, the nominal formula 10.8 + CN ms can be used.

IBM 729 IV Tape Timings

During a 729 IV *read* operation, the tape adapter unit and the processing unit are interlocked for 6.8 + CN ms (Figure 118). This includes:

6.7 ms — start time
.1 ms — record check time for high-density tape (.4 ms for low-density tape)
CN ms — record time

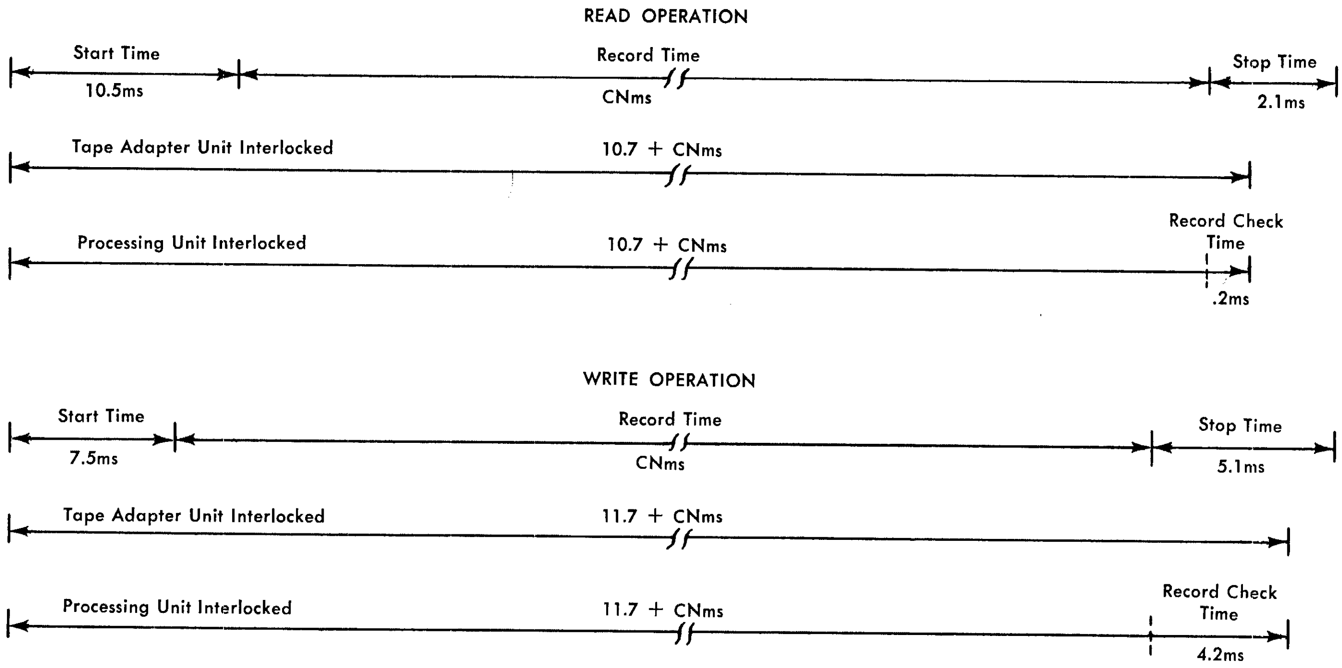
Therefore, in a tape read operation, processing can take place during 2.0 ms of stop time.

During a 729 IV *write* operation, the tape adapter unit and the processing unit are interlocked for 7.8 + CN ms (Figure 118). This includes:

5.0 ms — start time
2.8 ms — record check time for high-density tape (3.0 ms for low-density tape)
CN ms — record time

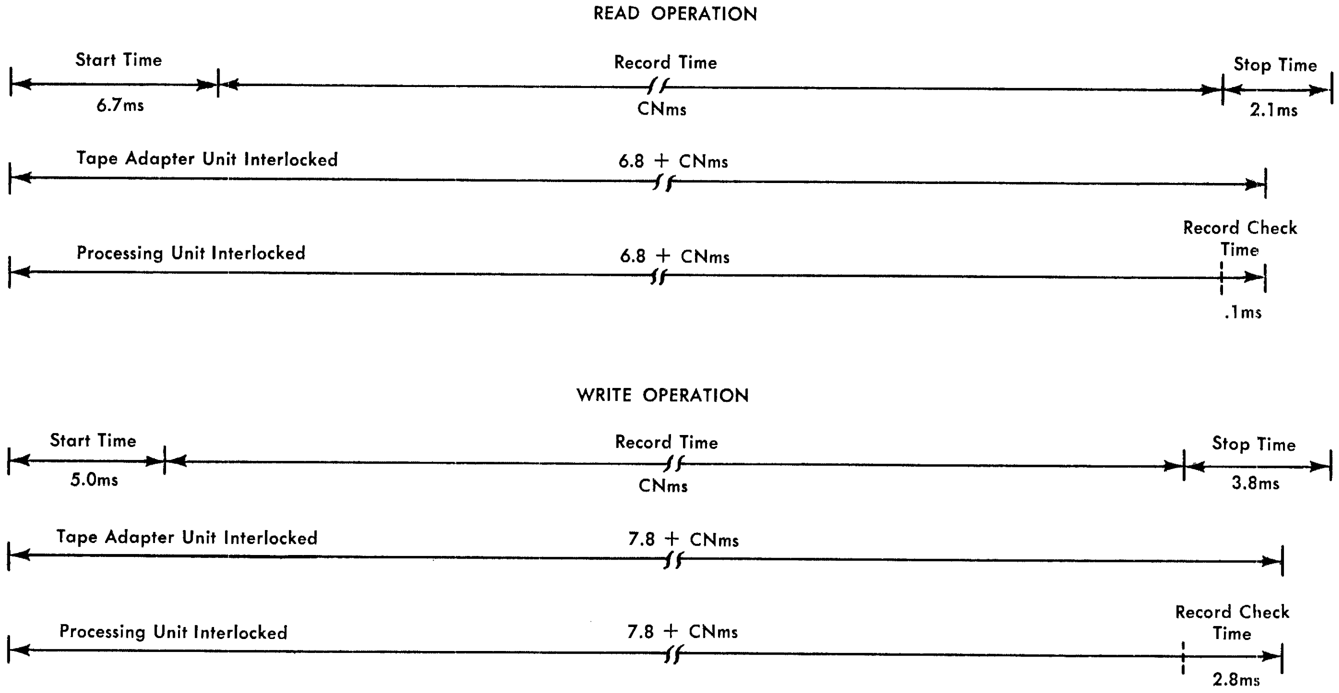
Therefore, in a tape write operation, processing can take place during 1.0 ms of stop time. The difference between the .1 ms record check time of read tape operations and the 2.8 ms record check time of write tape operations is due to the read-write head gap time (2.7 ms).

For job timing estimates of tape read and write operations the nominal formula 7.3 + CN ms can be used.



Note: Add 26.7ms to start time if tape unit is on load point when writing.

Figure 117. IBM 729, Model II Read-Write Tape Timing



Note: Add 26.7ms to start time if tape unit is on load point when writing.

Figure 118. IBM 729, Model IV Read-Write Tape Timing

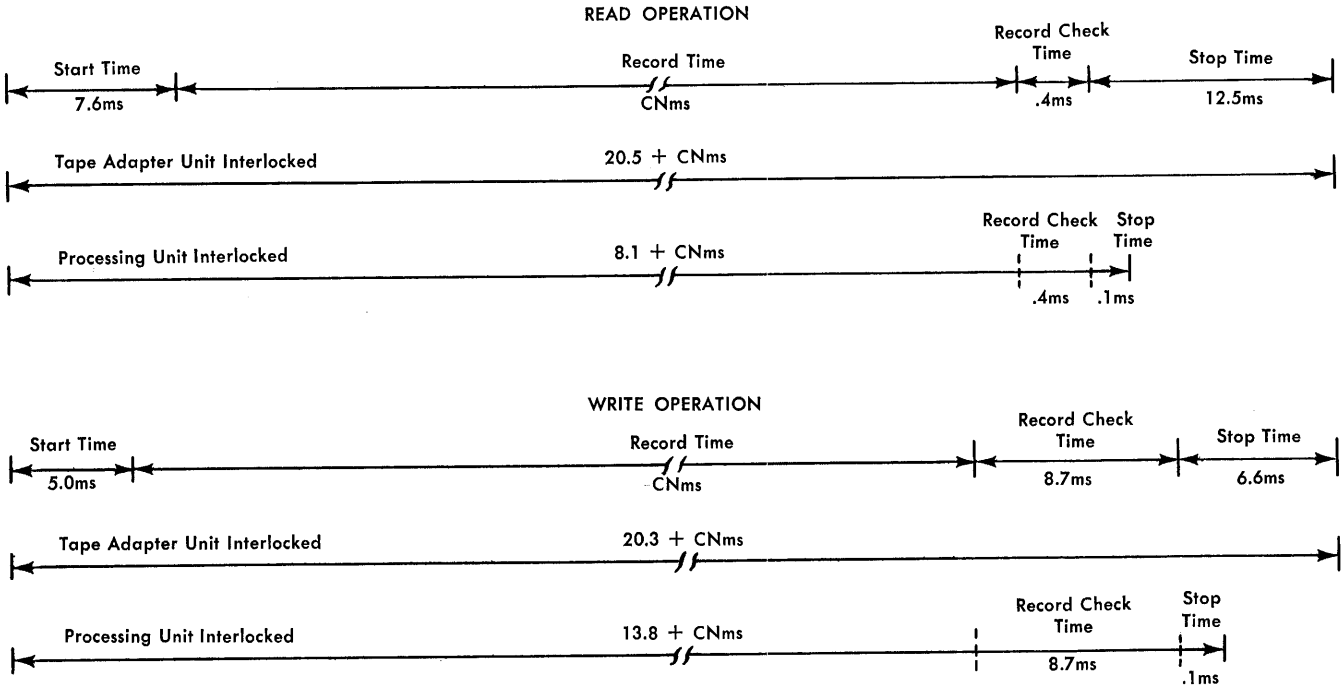


Figure 119. IBM 7330 Read-Write Tape Timing

IBM 7330 Tape Timings

During a 7330 *read* operation, the tape adapter unit is interlocked for 20.5 + CN ms (Figure 119). This includes:

7.6 ms — start time
12.5 ms — stop time
.4 ms — record check time for high-density tape (1.0 ms for low-density tape)
CN ms — record time

During the same read operation, the processing unit is interlocked for 8.1 + CN ms. This includes:

7.6 ms — start time
.1 ms — stop time
.4 ms — record check time for high-density tape (1.0 ms for low-density tape)
CN ms — record time

Therefore, in a tape read operation, processing can take place during 12.4 ms of stop time.

During a 7330 *write* operation, the tape adapter unit is interlocked for 20.3 + CN ms (Figure 119). This includes:

5.0 ms — start time
6.6 ms — stop time
8.7 ms — record check time for high-density tape (9.3 ms for low-density tape)
CN ms — record time

During the same write operation, the processing unit is interlocked for 13.8 + CN ms. This includes:

5.0 ms — start time
.1 ms — stop time
8.7 ms — record check time for high-density tape (9.3 ms for low-density tape)
CN ms — record time

Therefore, in a tape write operation, processing can take place during 6.5 ms of stop time. The difference between the .4 ms record check time of read tape operations and the 8.7 ms record check time of write tape operations is due to the read-write head gap time (8.3 ms).

For job timing estimates of read operations, use the formula 20.1 + C (N + 7) ms, where the factor C (7) is the record check time.

For job timing estimates of write operations, use the formula 19.9 + C (N + 7) ms, where the factor C (7) is the record check time and 8.3 ms of the 19.9 ms is the read-write head gap time.

RAMAC—Oriented IBM 1410 System

IBM 1405 Disk Storage

The IBM 1405 Disk Storage unit (Figure 120) combines the data processing capabilities of the IBM 1410 Data Processing System with the advantages and facility of large-capacity random access storage. The combination of the 1410 and 1405 provides an efficient and economical in-line data processing system.

The in-line method of data processing continually maintains the records of a business in an up-to-date status. Any transaction affecting a business can be processed when it occurs, and all records and accounts affected are updated immediately. The executives of an organization have available, at any time, information representing the status of any account at that moment.

Records in the IBM 1405 Disk Storage unit are stored on the faces of magnetic disks. The 1405 Model 1 has a storage capacity of 10 million alphamerical characters of information on 25 disks. Model 2 has a storage capacity of 20 million alphamerical characters on 50 disks.

The Model 2 disk-storage unit is divided into two modules. Each module contains 50,000 records on 25 disks. Each disk face has 200 tracks which are subdivided into 5 sectors. Each sector can contain a 200-character record (Figure 121).

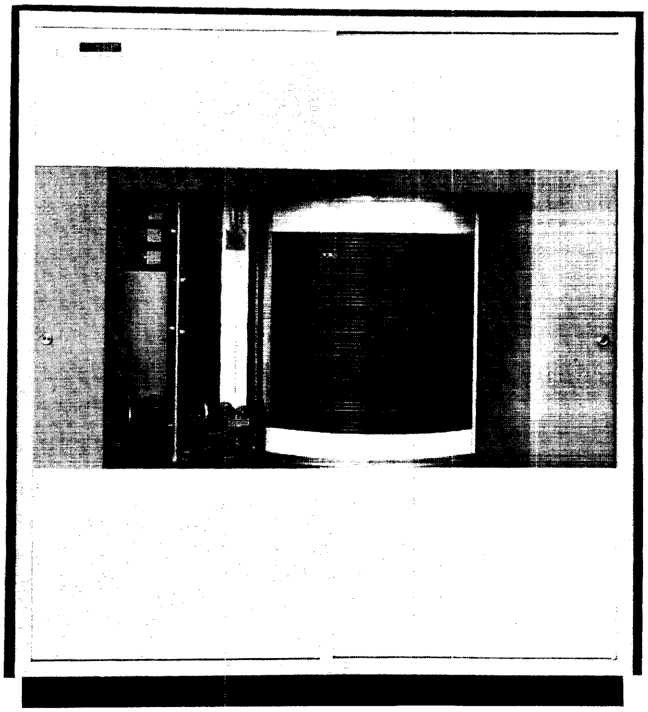


Figure 120. IBM 1405 Disk Storage Unit

Access Mechanism

The disk-storage unit can contain as many as three access mechanisms. One is standard, and the other two are available as special features. Each fork-shaped access mechanism has two read-write heads that are used to read and record data in the unit. One read-write head is for the top disk surface; the other is for the bottom disk surface. During a seek operation, the access mechanism moves vertically to seek a disk, and horizontally to seek a record (Figure 122).

To execute a READ, WRITE, or WRITE DISK CHECK instruction, the access mechanism must have been directed to the proper track location by a previously-executed SEEK instruction.

Speed

The disks rotate on a vertical shaft at the rate of 1,200 revolutions per minute. Data are read or recorded at the rate of approximately 25,000 characters per second. *Access time* is 100 milliseconds, minimum, and 800 milliseconds, maximum. Access time is the time required to locate a particular disk-unit record. Read, write, and write-disk check operations can be performed on a disk record without having to reseek, if no other seek operation intervenes.

Coding

The magnetic-disk recording code is the same binary-coded decimal used in the IBM 1411 processing unit.

Data are recorded in 7-bit or 8-bit code on the disk. The disk is coated with a magnetic oxide material. To insure the accuracy of incoming and outgoing data, a parity check is made when data are transferred to and from the disk unit, and a programmed write check is made to compare data written on the disk unit against the data in core storage.

Indicator Lights

POWER ON

When all the necessary power is being supplied to the disk-storage unit, the power-on light is turned on (Figure 123).

READY

This light is ON when the 1405 is available for use in the 1410 system. When the light is OFF, it indicates low air pressure, a power supply failure, or improper functioning of the reading or writing circuits. If the light is OFF, the not ready I/O channel-status indicator is set ON at the end the first data transfer attempted (the data transfer operation is bypassed and the indicator is set).

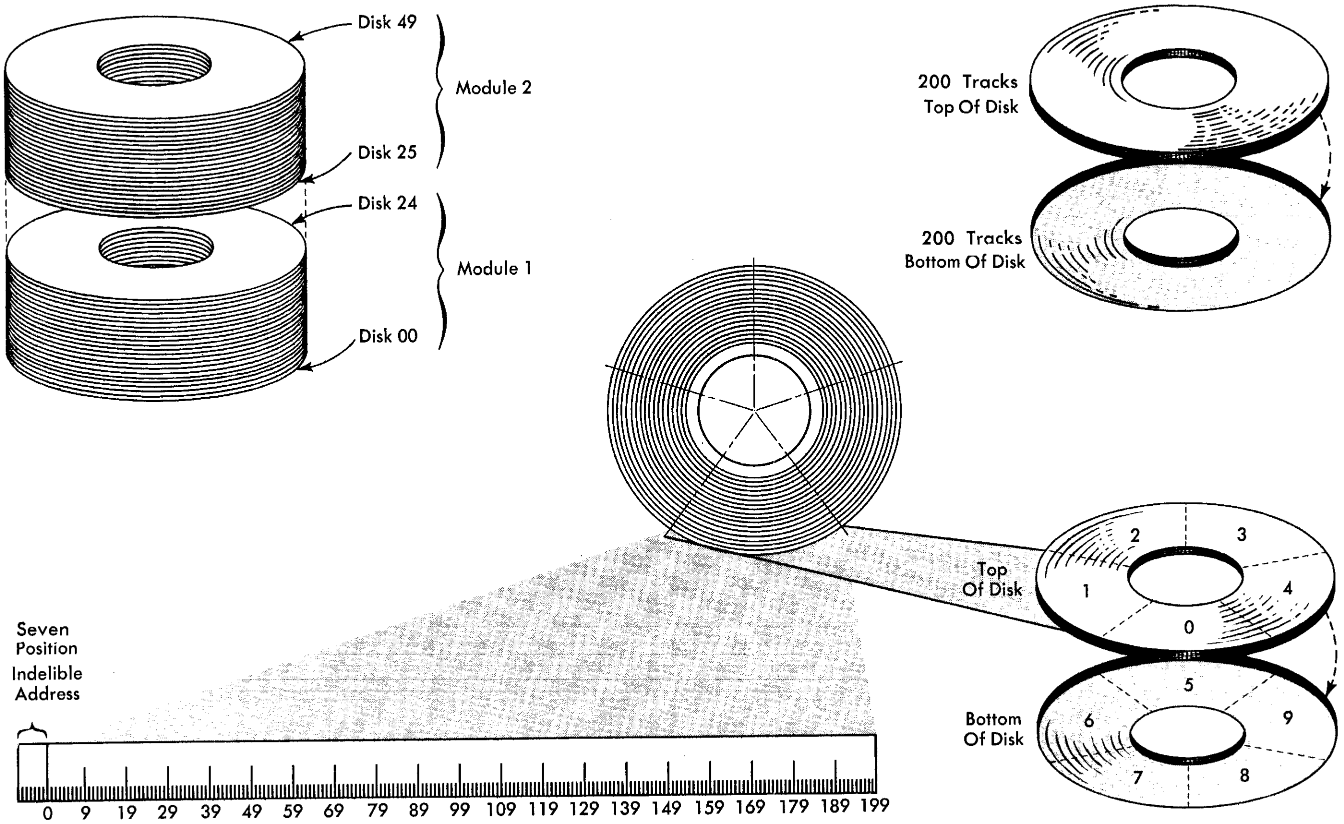


Figure 121. Disk Storage Unit Organization

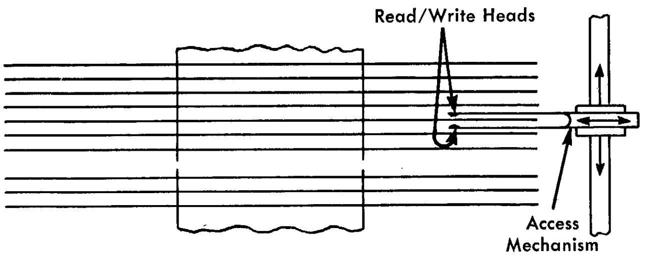


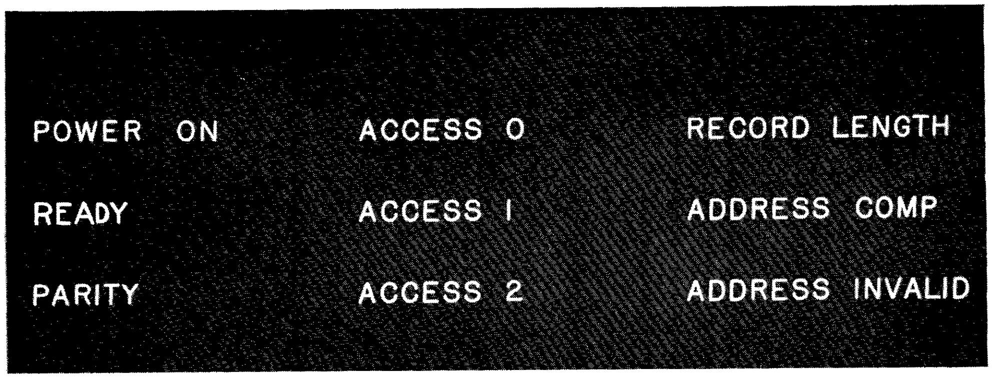
Figure 122. Access Mechanism

PARITY

This light comes ON when a parity check occurs as a character is read from, or written on a disk. The light is turned OFF by a computer-reset, error-reset, or write disk-check interlock reset operation, or by the next disk-storage unit operation.

ACCESS 0, 1, 2

These access lights correspond to the addresses of the access mechanisms in the 1405. Each light indicates that the corresponding access mechanism has been placed in an inoperable condition by either a logic



POWER ON	ACCESS 0	RECORD LENGTH
READY	ACCESS 1	ADDRESS COMP
PARITY	ACCESS 2	ADDRESS INVALID

Figure 123. IBM 1405 Indicator Lights

safety circuit or by a customer engineer. (This inoperable condition also sets the not-ready I/O channel-status indicator ON when that access mechanism is addressed.)

RECORD LENGTH

This light comes ON when an incorrect record length is encountered as a record is read from, or written on a disk. (This condition also sets the wrong-length record I/O channel-status indicator ON.) The light is turned OFF by a computer-reset, error-reset, or write-disk interlock reset operation, or by the next disk-storage unit operation.

ADDRESS COMP

This light comes ON when the record address in core storage does not agree with the record address in disk storage. (This condition also sets the condition and the no-transfer I/O channel-status indicators ON.)

ADDRESS INVALID

This light comes ON when an instruction addresses an access mechanism or a disk-storage unit that is not in the system. (This condition also sets the not-ready I/O channel-status indicator ON.)

Disk-Storage Addressing

Each track within a sector has an indelible 7-digit record address preceding the 200-character record area (Figure 124). The disk records are arranged sequentially in ascending order from bottom to top of the disk storage unit. The record address of the first record in the outside track of the bottom disk is x0000000 (Figure 125). The address of the last record in the inside track of the top disk is x0999990. (The "x" in the record address refers to the access arm to be used and is not part of the 7-digit indelible address). The designation of any 1405 unit connected to a 1410 data transmission channel must start with 0. For example, if all five 1405 units are connected to channel 1, the designation of these units would be 0, 1, 2, 3, and 4. If two 1405 units are connected to channel 1 and three to channel 2, the two units connected to channel 1 would be designated as 0 and 1 and the other three units would be designated as 0, 1, and 2.

IN 1410 CORE STORAGE						
Access Arm	Unit	Disk Face	Track	Sector	Constant	GM WM
X 0-2	X 0-4	XX 00-99	XX 00-99	X 0-9	X 0	≢ v
	ON 1405 DISK STORAGE					

Figure 124. Record Address Format

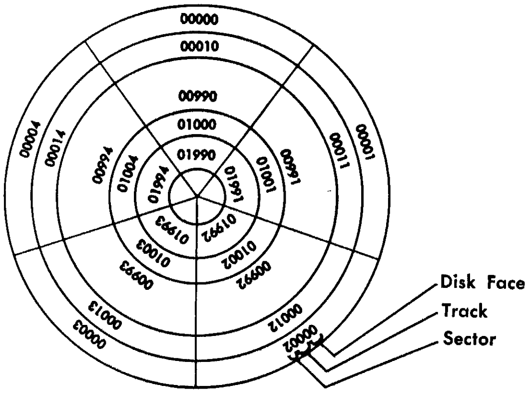


Figure 125. Record Layout

Disk-Storage Control Instruction Format

All disk-unit operations are initiated by a disk-storage unit instruction. This instruction, by using specific characters in certain locations, can initiate the reading or writing of single records or full tracks and other disk-unit operations. The format for the disk-unit instruction is shown in Figure 126.

The various parts of a disk-unit instruction and their uses are:

1. General mode of operation. This part identifies the operation as a move operation or a load operation. A move designation records data on, or reads data

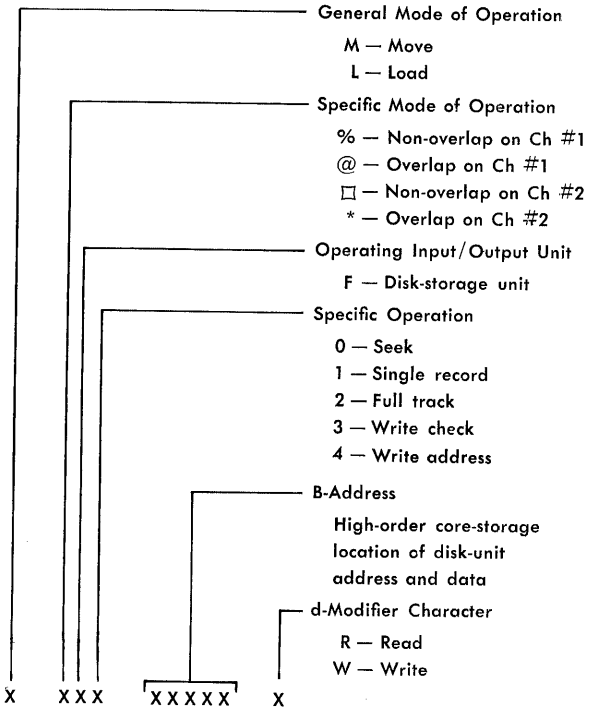


Figure 126. Disk Storage Unit Instruction Format

from the disk unit in a 7-bit coding. The recording sequence for each character is: C B A 8 4 2 1. A load designation records data on or reads data from the disk unit in an 8-bit coding. The recording sequence for each character is: M C B A 8 4 2 1.

Note: Data written in one of these two codings must be read in the same coding.

2. Specific mode operation. This part defines which data transmission channel is to be used and whether the operation is performed with or without overlapping. For more detail on overlap, refer to *Overlap Feature* in the *System Features and Considerations* section.
3. Operating input-output unit. The character F specifies the disk-storage unit as the active input-output device for this operation.
4. Specific operation. This part specifies the operation as a seek operation (0), a single-record operation (1), a full-track operation (2), a write-check operation (3), or a write-address operation (4). Specification of the operation as a read or write operation is done by the d-modifier character.
5. B-address. This part specifies the high-order core-storage position of the field containing the 8-character record address and the record data. Refer to Figure 124 for the core-storage format of the record address.

   Following the record address in core-storage is the input or output record area.
6. d-Modifier character. This area specifies the operation as a read operation (R) or a write operation (W).

Disk-Storage Control Instructions

These operations make it possible to seek a disk record, read and write a single record or a full track (with or without word marks), check for accurate write operations, and write-record addresses.

Seek Disk Record

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
SD	$\overset{\vee}{M}$ or $\overset{\vee}{L}$	x F 0	xxxxx	R or W

Function. This instruction causes the positioning of the specified access arm to a particular track location. The units position of the x-control field specifies a seek operation (0). The B-address specifies the high-order position of the record address in core storage. This record address specifies the access arm, module, disk, track, and record number.

The selected access mechanism seeks the record specified by the record address. Processing can continue while the access mechanism is moving.

Word Marks. Word marks are not affected.

Timing. T = .144 + access time (600 ms-avg) ms.

NOTE: If the access mechanism is already at the track to be used in the next operation, a SEEK instruction need not be given.

Example. Seek record number 210090 with access mechanism 1. Storage locations 00590-00597 should contain 10210090.

Address Registers after Operation.

I-Add. Reg.	*B-Add. Reg.*
NSI	B

Write Single Record

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
WD	$\overset{\vee}{M}$	x F 1	xxxxx	W

Function. A single record is written on the disk-storage unit from core storage. The units position of the x-control field specifies a single-record operation (1). The B-address specifies the high-order position of the record address. This address specifies the access arm, module, disk, track, and record number. Following the record address in core storage is the output record area. The d-modifier character specifies a write operation.

The record should fill the entire disk-unit sector area. If a group-mark with a word-mark is encountered in core storage prior to the end of the sector, the wrong-length record I/O channel-status indicator and its associated indicator light are turned on. (The remainder of the record is filled with invalid blanks.)

Word Marks. A group-mark with a word-mark must appear at the position to the immediate right of the last character of the record address. This is needed for the record-address length check.

A group-mark with a word-mark must appear at the position to immediate right of the last character of each core-storage record. The end of the sector, or a group-mark with a word-mark to the right of the core-storage records, stops the writing operation. The group-mark with a word-mark is not written.

Timing. T = .144 + rotational delay (0-50) + 9.5 ms.

NOTES:

1. This instruction must be followed by a WRITE DISK CHECK instruction before disk-storage unit is used again.
2. If the operation code is $\overset{\vee}{L}$, the word marks in the

specified core-storage area are transferred and written on the disk unit along with the rest of the data. The coding on the disk unit is now an 8-bit coding (MCBA 8421), and each sector has the capacity for 176 8-bit characters and an indelible address.

Address Registers after Operation.

I-Add. Reg.	*B-Add. Reg.*
NSI	B + LB + 1

Write Full Track

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
WDT	$\overset{\vee}{\text{M}}$	xF2	xxxxx	W

Function. This operation is the same as WRITE SINGLE RECORD except that the units position of the x-control field specifies a full-track operation (2). A full track (5 sectors – 200 characters each) is written. The write operation will start with sector zero or sector 5, regardless of which sectors, other than zero or 5, may be specified by the record address in core storage.

On the disk file, a physical full track consists of ten sectors (0-9) with sectors 0-4 on the top of a disk and 5-9 on the bottom of the disk. In this manual, "full track" actually means one-half of a physical full track. The instructions READ and WRITE FULL TRACK refer to five sectors, 0-4 or 5-9.

This instruction must be followed by a WRITE DISK CHECK instruction before the disk unit is used again.

If the operation code is $\overset{\vee}{\text{L}}$, the word marks in the specified core-storage area are transferred and written on the disk unit along with the rest of the data. The coding on the disk unit is now an 8-bit coding (MCBA 8421).

Timing. T = .144 + rotational delay (0-50) + 49.5 ms.

Write Disk Check

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
WDC	$\overset{\vee}{\text{M}}$	xF3	xxxxx	W

Function. Data recorded on the disk-storage unit during a preceding write operation are read from the unit and compared, character by character, with the data from core storage.

The units position of the x-control field specifies the write disk check operation (3). The B-address specifies the high-order position of the disk-unit address followed by the record data.

Word Marks. Same as write operation.

Timing. T = .144 + rotational delay (0-50) +9.5 ms.

Address Registers after Operation.

I-Add. Reg.	*B-Add. Reg.*
NSI	B + LB + 1

NOTE: A LOAD WRITE operation (specified by Op-code $\overset{\vee}{\text{L}}$) must be followed by a load write-disk check operation.

Read Single Record

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
RD	$\overset{\vee}{\text{M}}$	xF1	xxxxx	R

Function. A single record is read from the disk-storage unit into core storage. The units position of the x-control field specifies a single-record operation (1). The B-address specifies the high-order position of the record address. This address specifies the access arm, module, disk, track, and record number. The input record area follows the record address in core storage. The d-modifier character specifies a read operation.

Reading from the disk-storage unit proceeds until the end of the sector or until a group-mark with a word-mark is sensed in core storage, whichever occurs first.

Word Marks. A group-mark with a word-mark sensed in core storage ends the operation.

Timing. T = .144 + rotational delay (0-50) + 9.5 ms.

Address Registers after Operation.

I-Add. Reg.	*B-Add. Reg.*
NSI	B + LB + 1

NOTE: If the operation code is $\overset{\vee}{\text{L}}$ the word marks written in the disk-unit record area are transferred and recorded in core storage with the rest of the data.

Read Full Track

Instruction Format.

Mnemonic	*Op Code*	*X-control field*	*B-address*	*d-character*
RDT	$\overset{\vee}{\text{M}}$	xF2	xxxxx	R

Function. This operation is the same as READ SINGLE RECORD except that the units position of the x-control field specifies a full-track operation (2). A full track (5 sectors – 200 characters each) is written. The read operation will start with sector 0 or 5, regardless of which sectors, other than 0 or 5, may be specified by the record address in core storage.

If a group-mark with a word-mark is sensed in core storage before the end of the last sector, or if the last sector is reached before sensing the group-mark with a word-mark, the reading operation is ended, and the wrong-length record I/O channel-status indicator and its associated indicator light are turned on.

If the operation code is $\overset{\text{v}}{\text{L}}$, the word marks written in the disk-unit record area are transferred and recorded in core storage with the rest of the data.

Timing. T = .144 + rotational delay (0-50) + 49.5 ms.

Write Address

Instruction Format.

Op Code	*X-control field*	*B-address*	*d-character*
$\overset{\text{v}}{\text{M}}$	xF4	xxxxx	W

Function. A single indelible address for a single specified sector is written on the disk unit. The WRITE ADDRESS KEY, located on the disk-storage control unit CE panel, must be operated before the instruction is executed. Data recorded in the record portion of the specified sector must be rewritten as a separate operation.

The units portion of the x-control field specifies the write address operation (4).

Disk-Storage Unit Operation Status Indicators

A disk-storage unit operation can set any one of the I/O channel-status indicators ON (the conditions that set the various indicators are shown in Figure 127). These I/O channel status indicators must be tested before any new I/O operation can start. If one of the indicators is set from a previous I/O operation and a new I/O operation, on the same channel, is called for, the interlock light on the IBM 1415 Console signifies a violation of the programming rules.

IBM 1405 Disk Storage, Timing Considerations

Disk-Storage Access Time

To calculate timing for magnetic disk operations, it is necessary to estimate the average time it takes to seek the records needed for a particular application. If input to the operation is in sequence, the average access time is less than if the input data is unsorted. This can be explained by the fact that the duration of

INDICATORS & ASSOCIATED BITS	OPERATIONS				
	SEEK	ALL DISK OPERATIONS	SINGLE OR FULL TRACK READ	WRITE DISK CHECK	WRITE SINGLE OR FULL TRACK OR WRITE ADDRESS
NOT READY (1)	Access inoperative. No such access or module. Disk storage control unit power off or operating off-line.	Access inoperative. No such access or module. Disk storage control unit power off or operating off-line.	Access inoperative. No such access or module. Disk storage control unit power off or operating off-line.	Access inoperative. No such access or module. Disk storage control unit power off or operating off-line.	Access inoperative. No such access or module. Disk storage control unit power off or operating off-line.
BUSY (2)	Access mechanism still in motion from previous operation.	Access mechanism still in motion from previous operation.	Access mechanism still in motion from previous operation.	Access mechanism still in motion from previous operation.	Access mechanism still in motion from previous operation.
CHECK (4) DATA	Disk storage control unit detects an invalid address character.	Disk storage control unit detects an invalid data character. Processing unit detects an invalid data character (input only)	Disk storage control unit detects an invalid data character. Processing unit detects an invalid data character (input only)	Disk storage control unit detects an invalid data character.	Disk storage control unit detects an invalid processing unit data character.
CONDITION (8)	Not applicable	No address compare.	No address compare.	Data in core storage does not compare with data in disc storage.	Data in core storage does not compare with data in disc storage.
WRONG LENGTH RECORD (A)	Wrong length address detected.	Wrong length record detected.	Wrong length record detected.	Wrong length record detected.	Wrong length record detected.
NO TRANSFER (B)	Disk storage control unit detects an invalid address character. Wrong length address.	No address compare.	No address compare.	No address compare.	No address compare.

Figure 127. I/O Channel-Status Indicators Set During IBM 1405 Operation

the seek depends on how far the access arm must travel.

To seek a track on another disk, the access arm moves horizontally, vertically, and horizontally again. The minimum time to move from the outside track of one disk to the outside track of an adjacent disk is 415 milliseconds. The maximum length of a seek operation is from the inside track of the top disk to the inside track of the bottom disk and takes 800 milliseconds. Figure 128 shows track-to-track access times.

To seek a different-track on the same disk (top or bottom face), the arm moves horizontally only. In this case, minimum seek time is 90 milliseconds and maximum seek time is 250 milliseconds (Figure 129).

Disk-to-disk access time ranges from 100 to 315 milliseconds. Figure 130 shows timing for these operations.

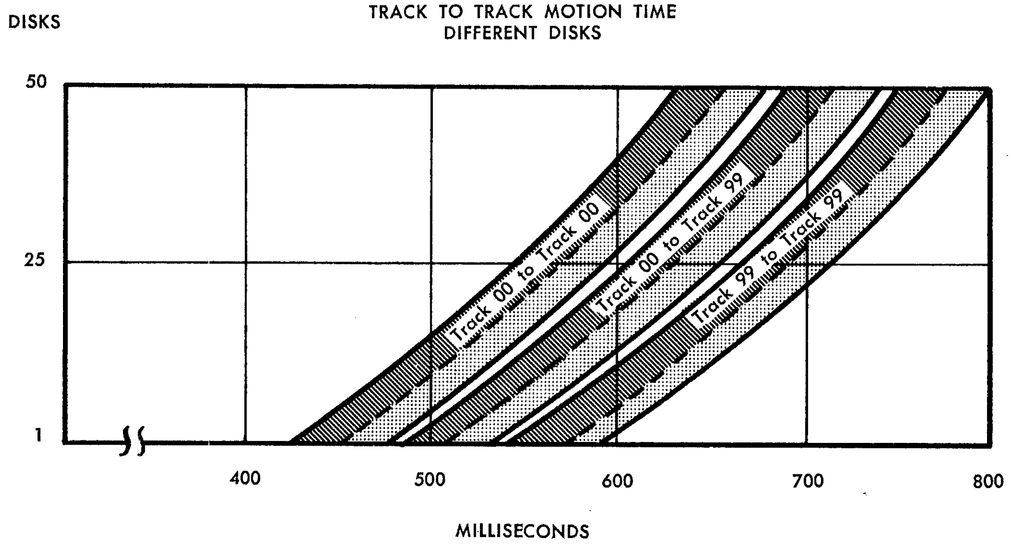


Figure 128. Track-to-Track Motion Time, Different Disks

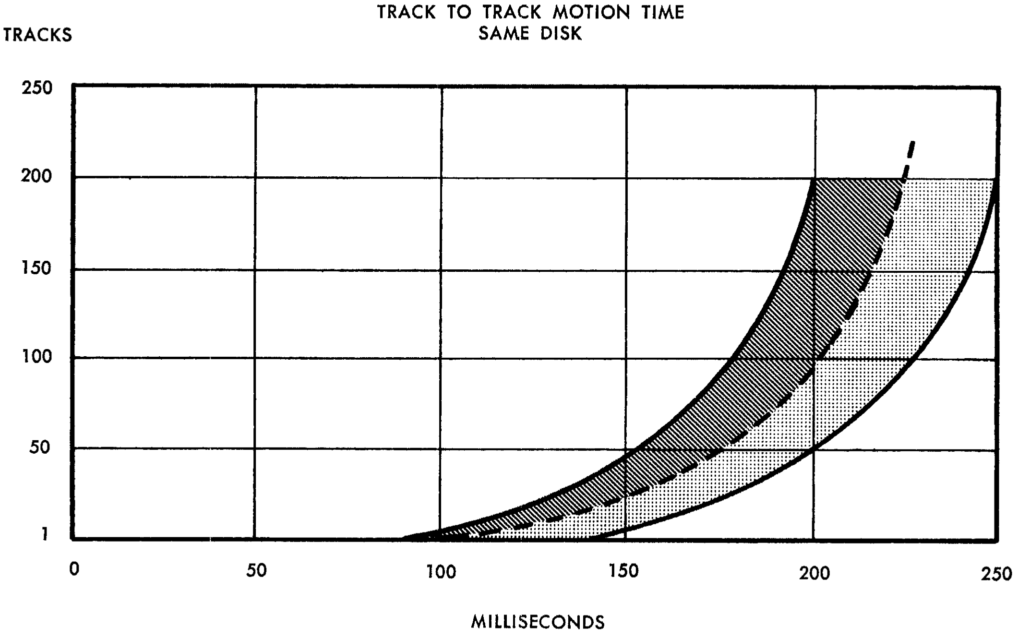


Figure 129. Track-to-Track Motion Time, Same Disk

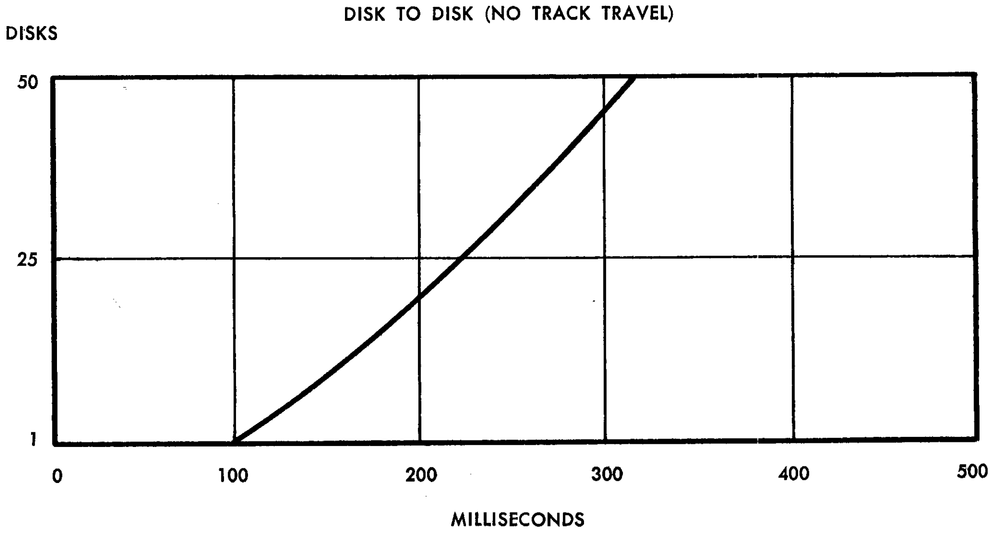


Figure 130. Disk-to-Disk Travel (No Track Travel)

System Features and Considerations

Data Transmission Channel 2

The IBM 1410 Data Processing System has one standard data-transmission channel (channel 1). The input-output units or features served by channel 1 are:

1. Up to 10 magnetic tape units (can be 729 II's and IV's – intermix standard, or 7330's)
2. One IBM 1009 Data Transmission Unit
3. One IBM 1011 Paper Tape Reader
4. Up to 20 IBM 1014 Remote Inquiry Units
5. Up to 5 IBM 1301 units (cannot be intermixed)
6. One IBM 1402 Card Read-Punch, Model 2
7. One IBM 1403 Printer
8. Up to 5 IBM 1405 units (cannot be intermixed)
9. One IBM 1412 Magnetic Character Reader
10. One IBM 1415 Console
11. The Telegraph Input-Output Feature
12. One IBM 1419 Magnetic Character Reader

If the hundreds position of an input-output instruction's x-control field contains a per cent symbol (%), it specifies that channel 1 will be used.

An additional data-transmission channel (channel 2) is available, and these units can be served by channel 2:

1. Up to 10 magnetic tape units (can be 729 II's and IV's – intermix standard, or 7330's)
2. Up to 5 IBM 1301 Disk Storage units (total disk storage units for both channels cannot exceed 5)
3. Up to 5 IBM 1405 Disk Storage units (total disk storage units for both channels cannot exceed 5)
4. One IBM 1412 Magnetic Character Reader
5. One IBM 1419 Magnetic Character Reader

If the hundreds position of an input-output instruction's x-control field contains a lozenge (⌑), it specifies that channel 2 will be used.

Processing Overlap Feature

The basic IBM 1410 Data Processing System is interlocked whenever an input-output operation is being performed. No processing can occur because the core-storage area is being used during the operation. An IBM 1410, with the processing overlap feature installed, allows computing to occur in the system while part of the input-output operation is being performed. While the input-output unit is preparing to send or receive data, the system continues computing. The computing is interrupted as each individual character is stored in, or sent from, core storage.

Feature Components

The overlap feature necessitates the addition of some transmission and controlling circuitry for each affected data-transmission channel. Each channel has two single-character synchronizer registers associated with it. These registers are used as an intermediate storage area during the character-by-character data transfers between the system and the input-output units. The channel-1 registers are called the E1 and E2 registers. The channel-2 registers are called the F1 and F2 registers.

Each channel also has a 5-character address register associated with it. This address register specifies the core-storage location of the character being transferred. The channel-1 register is called the E-address register, and the channel-2 register is called the F-address register.

The six I/O channel-status indicators (previously discussed in the *Logic Operations* section) also indicate conditions resulting from overlapped input-output operations, as well as non-overlapped operations. The six indicators on channel 1 are tested with an Ř (I) d instruction, and the six indicators on channel 2 are tested with an X̌ (I) d instruction.

Each channel has its own overlap-in-process indicator, which can be tested with a TEST AND BRANCH instruction – J̌ (I) 1 or 2. The indicator is turned ON at the beginning of an overlapped operation and is automatically turned OFF when the operation is completed. When the light is ON, it indicates that the presently specified I/O unit is *not busy* completing a previous operation, and has started performing the current operation. The J̌ (I) 1 or 2 instruction is normally executed immediately following the overlapped I/O unit instruction.

Input-Output Operation

The operation of any I/O unit is initiated by a specific input-output instruction, consisting of an operation code, an x-control field, a B-address, and a d-modifier character. For read or write operations, the operation code is either M̌ (move) or Ľ (load). This specifies a data transfer with or without word-mark transfer.

The x-control field is a three-character field, with each character specifying one or more things. The hundreds position of the x-control field specifies which channel will be used and whether the operation will be executed in an overlapped mode. The symbols, and the channel and operation, they indicate are:

Symbol	*Channel*	*Operation*
%	Channel 1	non-overlap mode
@	Channel 1	overlap mode
□	Channel 2	non-overlap mode
*	Channel 2	overlap mode

The two remaining positions of the x-control field, the B-address, and the d-modifier character have the same meaning, regardless of the operation mode.

NON-OVERLAP OPERATION

A non-overlapped operation is initiated by using the specified symbol in the hundreds position of the x-control field (% for channel 1; □ for channel 2) associated with the input-output instruction.

If a non-overlapped tape-read operation was being executed on channel 1, the instruction would be $\check{M}$ or $\check{L}$ (%U1) (01500) R, and would be executed as shown in Figure 131.

Executing the instruction stops processing completely. Reading a 100-character record from an IBM 729-II high-density tape takes 13.2 ms (10.8 ms tape start time + 100 characters times a 24μs character rate).

If the processing of the record takes 15 ms, then the tape is inactive until the record processing is complete. Only one operation can take place at any one time.

OVERLAP OPERATION

An overlap operation is initiated by using the specified symbol in the hundreds position of the x-control field (@ for channel 1; * for channel 2) associated with the input-output instruction.

Once the execution of the overlapped I/O unit instruction is begun, the program is immediately restarted and the program advances to the next instruction in sequence. No other I/O unit can be addressed on that channel until the operating I/O unit has completed its operation. This is necessary because there is only one set of I/O channel-status indicators for each channel, and they must be tested before the next I/O operation begins. This *status test* must be completed before the next I/O unit operation on that channel is encountered and is insured by the system's circuitry. The status test is considered complete if either:

1. a BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction, $\check{R}$ (I) ⧧, is executed before encountering the next I/O unit instruction on the same channel; or
2. a specific $\check{R}$ (I) d instruction (see Figure 31) is executed and *results in a branch* before encountering the next I/O unit instruction on the same channel.

Tape Read Operation. If an overlapped tape-read operation was being executed on channel 1, the instruction would be $\check{M}$ or $\check{L}$ (@U1) (01500) R, and the operation (Figure 132) would take place as follows:

1. The B-address of 01500 is read into the E-address register and the tape movement starts. (The A- and B-address registers continue being used in the processing that overlaps the tape-read operation.) Processing continues during tape start time.

2. As soon as the E2 register receives a character from tape, the character is transferred to the E1 register.

3. At the end of the next processing cycle, processing is suspended for 5.25 μs, while the character is transferred from the E1 register to the core-storage position specified by the E-address register. (With the

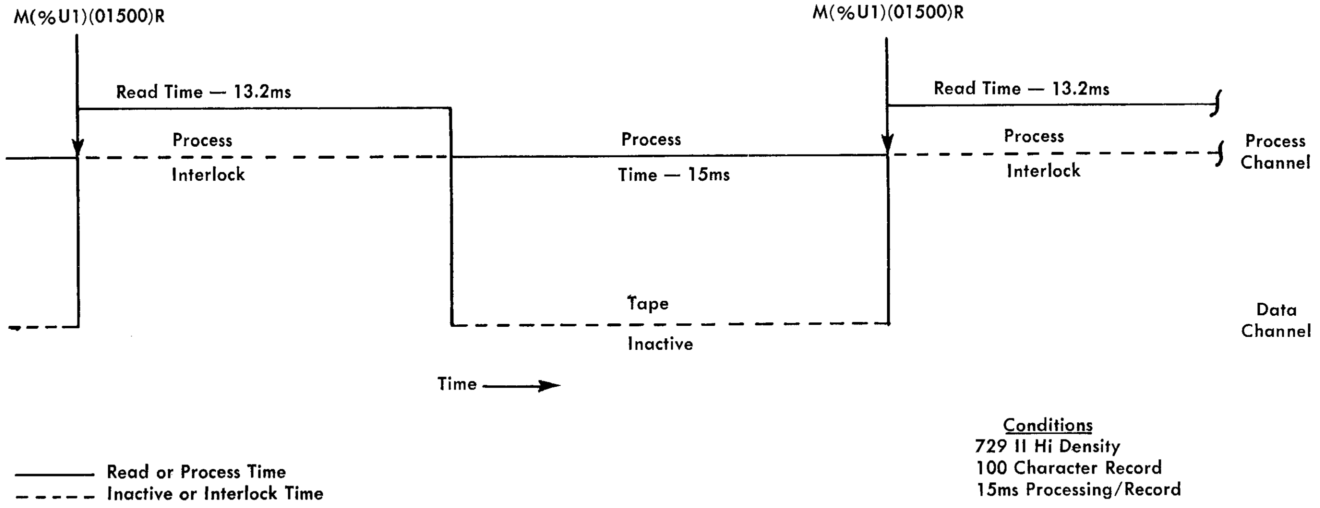


Figure 131. Non-overlap Tape Read Operation

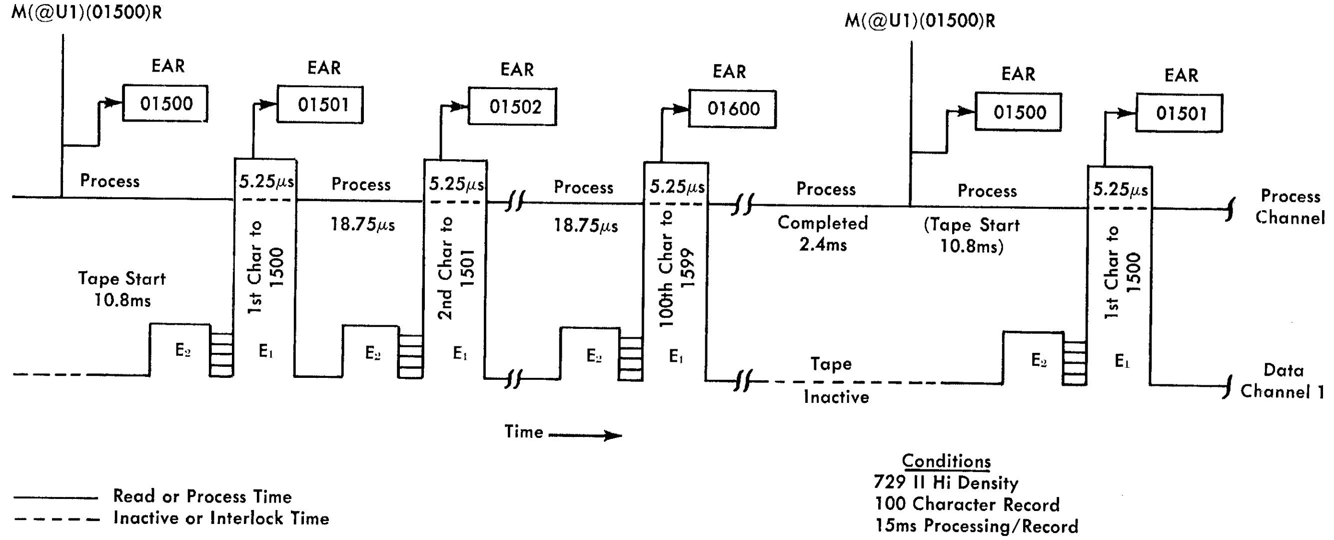


Figure 132. Overlap Tape Read Operation

24 μs character rate of IBM 729 II's there is an *average* of 18.75 μs of processing time available between tape-character transfers.)

4. The E-address register is increased by one during the 5.25 μs suspension of processing. After the 5.25 μs suspension, processing continues.

During the reading of this tape record, it is possible to process a previously-read record. If 15 ms of processing time is required, 10.8 ms of processing time can be utilized during the tape start time. The remaining 4.2 ms of required processing time is overlapped by 5.25 μs of tape-character transfer time. (Each character read into core storage requires 5.25 μs cycle. Therefore, the available processing time during an overlap operations is *reduced* by N × 5.25 μs, when N is the number of characters transferred. A 100-character record × 5.25 μs equals 525 μs.)

At the end of the tape-record transfer, 2.4 ms of process is still needed to finish processing the previous record. No tape activity occurs during this 2.4-ms period. At the completion of the overlap operation, the E-address register contains the address that is two positions to the right of the last core-storage position read into.

Tape Write Operation. Overlapping a tape-write operation is similar to a tape-read operation. Processing is suspended 5.25 μs every time a character is transferred from core storage to the E register. (The E-address register contains the core-storage location of the character transferred.) The character is transferred from the E1 register to the E2 register and then to tape during processing.

Each character read out of core storage required a 5.25 μs cycle. Therefore, the available processing time during an overlap operation is *reduced* by N × 5.25 μs, when N is the number of characters transferred.

At the end of the overlap operation, the E-address register contains the address that is two positions to the right of the last core-storage position read out of.

Tape Read and Write Operation. A combination tape-read and write operation (Figure 133) makes use of both channels, and permits the overlapping of reading, writing, and processing. Each channel operates independently of the other channel, until both channels require a suspension of processing at the same time. At such a time, a priority is set up which depends on the type of the last completed cycle. If the last completed cycle was a process or channel-2 cycle, the priority order is channel 1, channel 2, process. If the last cycle involved channel 1, the priority order is channel 2, channel 1, process.

NOTE: The available processing time between each character transfer should be considered as utilized 100%, even though the time is not an exact multiple of a core-storage cycle. For example, an overlapped channel 1 tape-read operation (Figure 132) had a process time between character transfers that averaged 18.75 μs. Because the average characters for IBM 729-II high-density tapes vary from 21-27 μs, the available overlapped process time then varies from 15.75—21.75 μs. A basic process-cycle point is .75 μs, and a process cycle (4.5 μs in length — 6 cycle points × .75μs) can begin immediately after a character transfer is completed. If the available process time is 15 μs, three complete process cycles

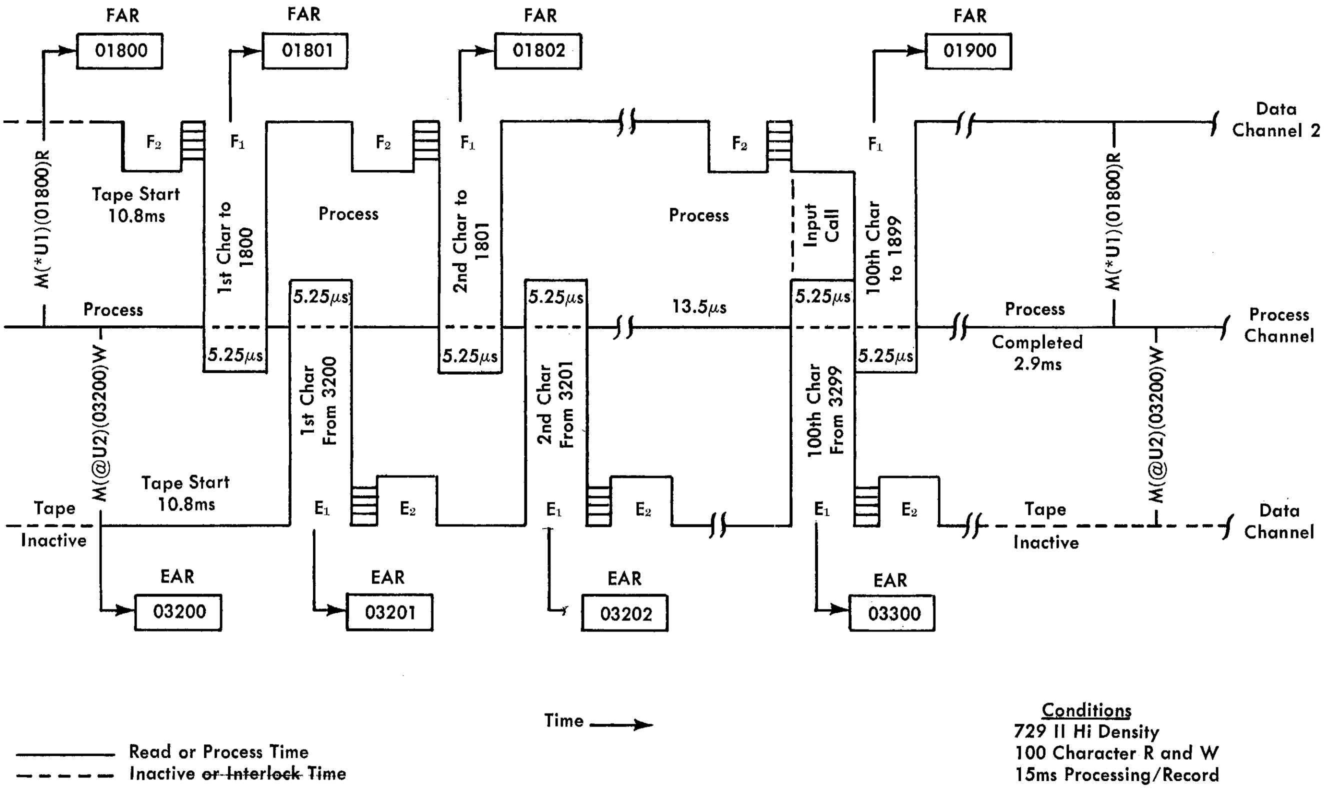


Figure 133. Overlap Tape Read and Write Operation

are completed in 13.5 μs. Because the system has no indication that the next tape character will be available in 1.5 μs, another process cycle is executed. This delays the next tape character transfer by 3 μs; however, since each tape character passes serially through the E1 and E2 registers, it gives the channel the facility to temporarily get a character behind, during the data transfer. This 3 μs delay is regained by picking up 1.5 μs on the next two tape-character transfer cycles. In this way, the available process time is fully utilized.

I/O Synchronizer Operation. All of the card-type input-output units that use the I/O synchronizer are serviced by channel 1. Because of this, transfers in and out of the synchronizers cannot be overlapped with each other. However, processing can overlap this type of data transfer.

Overlap Error or Stop Conditions. The error or stop conditions that can occur during an overlap operation are:

1. An I/O instruction, encountered before completing a previous I/O operation on the same channel, causes an immediate system stop. This condition requires that reprogramming must be done.
2. A processing unit error occurs during an overlapped tape operation. The system stops and the normal restart procedures must be used.
3. A programmed stop or a console stop occurring during an overlapped input-output unit operation. When this occurs, processing stops, but the I/O unit continues until the data transfer is completed. Any restart procedure that is used *must* include a BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction so that the status test is satisfied before the next I/O instruction is encountered. The status test can also be satisfied by a $\overset{v}{R}$ (I) d instruction that results in a branch.

Overlap Operational Considerations. Operational considerations during an overlapped operation are:

1. An overlapped I/O instruction, executed on an IBM 1410 Data Processing System without the overlap feature, causes the system to stop.

2. The B-field of both an overlapped and a non-overlapped I/O instruction can be indexed.

3. On a 1410 system equipped with both channels and the overlap feature, read/write/processing, read/read/processing, or write/write/processing is possible with one exception.

The exception is with I/O instructions containing d-modifier characters of $ or X. These operations are always executed in a non-overlapped mode, even if they are programmed to operate in the overlap mode.

4. Unit control operations of the forms, such as $\check{U}$ (@ m m) d or $\check{U}$ (* m m) d, can be executed. (The same restrictions apply to the use of these instructions as apply to the $\check{M}$ or $\check{L}$ (X) (B) d instructions.) Tape-backspace operations are not overlapped; because there is no data transfer involved. The 1410 system is free to continue processing during the entire operation, and the control unit initiating the operation, are released as soon as the backspace operation is under way.

5. If the console I/O printer is operated in the *inquiry* mode on channel 1, the operation is the same as in the un-overlapped mode except that the program continues overlapped with the inquiry operation. The channel 1 *overlap in process* latch remains ON until the operator presses either the cancel or the release key.

6. As soon as a BRANCH IF ANY I/O CHANNEL STATUS INDICATOR ON instruction is encountered during an overlap operation, processing is stopped until the I/O operation on that channel is completed. If this processing was not stopped, any I/O channel-status indicator set ON after the $\check{R}$ (I) d or $\check{X}$ (I) d instruction was executed would not be detected before it was turned OFF by the next I/O operation.

An exception to this occurs when the I/O unit signals *not ready* or *busy*. A not ready or busy signal indicates that the I/O operation never began, and there is no need to stop processing.

IBM 1401-1410 Compatibility

The basic design of the IBM 1410 Data Processing System permits running many programs as originally written for IBM 1401 Data Processing System. By using the compatibility switch on the CE panel of the IBM 1415 Console, it is possible for a 1410 system to provide a major increase in total system's performance and still maintain the ability to utilize programs written for all models of the 1401 (A, B, C, D, E, F). Any standard configuration of 1401 system components (1401, 1402, 1403, 1405, 729, 7330) and many 1401 system special features are fully utilized by a 1410 system operating in the 1401 mode. These features are:

1. Additional Storage (10K 1410 system operates as an 8K 1401 system; all other 1410 systems operate as a 16K 1401 system)
2. Multiply-Divide
3. Expanded Print Edit
4. Advanced Programming (includes Indexing, Store Address Register, and Move Record)
5. Print Storage
6. High-Low-Equal Compare
7. Read-Punch Release (See *Note*)
8. Sense Switches

Note: Because the 1410 system makes use of an intermediate storage area on data transfers between core storage and the reader and/or punch, these instructions do not cause actual card-feed motion.

The 1410 is not compatible with a 1401 program that makes use of the following special features:

1. The Column Binary feature, including the operations:
   a. Read Column Binary
   b. Punch Column Binary
   c. Move Binary Code
   d. Move and Binary Decode

Note: The bit-test operation, available on 1401 systems as part of the column binary feature, is a standard feature on the 1410 system and operates in the 1401 mode.

2. The Compressed Tape feature, including the operations:
   a. Read Compressed Tape
   b. Move and Insert Zeros
3. The Punch Feed Read feature, including:
   a. Read-Punch Feed
   b. Read-Punch Feed and Branch
   c. Write-Read Punch Feed
   d. Write-Read Punch Feed and Branch
4. The Serial I/O Adapter, including provisions for attaching the following units:
   a. IBM 1009 Data Transmission Unit
   b. IBM 1011 Paper Tape Reader
   c. IBM 1412 Magnetic Character Reader
   d. IBM 1418 Optical Character Reader
   e. IBM 1419 Magnetic Character Reader

Note: The 1412 can be attached to a 1410 system, but will not operate in the 1401 mode.

Most 1401 programs written in IBM 1401 *Autocoder* or SPS-1 or 2 can be expected to run on a 1410 operating in the 1401 mode with little, if any, reprogramming. Some reprogramming may be needed on other IBM 1401 programs.

Index

A24-1407-2

IBM 1410 Printed in U. S. A. A24-1407-2

IBM
International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York