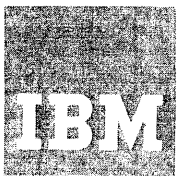
File No. 1130-01
Form A26-5881-3



Systems Reference Library

IBM 1130 Functional Characteristics

This manual provides basic programming and operating information for the 1130 Computing System. The functional aspects of the System are explained in detail, and the operational characteristics are described in terms of program instructions, input/output operations, and Central Processing Unit console displays and functions. Intended as a reference manual, the material presented assumes some prior knowledge of stored program computers.

PREFACE

Intended primarily as a reference tool, this manual presents information on a level that requires a minimum of prior knowledge of stored-program computers. Some of the terms used in the following pages, however, may be unfamiliar to the inexperienced. To avoid lessening the value of the book as a reference tool, explanations of terms are confined to the context of their use.

The IBM publication, Introduction to IBM Data Processing Systems (Form F22-6517) provides an excellent introduction to the stored-program computer. The reader interested in communications should also be familiar with the IBM publication, General Information – Binary Synchronous Communication (Form A27-3004).

Fourth Edition

This manual is a major revision of the previous edition (A26-5881-2) and makes it obsolete.

Specifications contained herein are subject to change from time to time. Any such change will be reported in subsequent revisions or Technical Newsletters.

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

Copies of this and other IBM publications can be obtained through IBM Branch Offices.

A form is provided at the back of this publication for your comments.

This manual was prepared by the IBM Systems Development Division, Product Publications, Dept. 455, Bldg. 014, San Jose, California 95114. Send comments concerning the contents of this manual to this address.

CONTENTS

The IBM 1130 Computing System provides the capacity and versatility to accomplish the engineering and scientific computations that formerly were possible only with large computer systems. The 1130 fulfills the "general purpose" requirements of these areas with computing power well above previous systems in the same cost range. The 1130 system can also handle supporting commercial data processing applications.

The design of the 1130 system is oriented to the operator. Only a minimum of training and experience with computing systems is necessary to make the 1130 usable by engineering and research personnel for solutions to problems in individual projects. In addition, programs and programming systems, supplied by IBM, relieve the user of detailed programming and provides for the statement of problems in a familiar language.

The compact, easily-operated 1130 system features the IBM 1131 Central Processing Unit (CPU) in three separate models with various core storage capacities and speeds. The 1131 Model 1 has a core storage capacity of 4,096 or 8,192 sixteen-bit words. The 1131 Model 2 has a core storage capacity of 4,096, 8,192, 16,384, or 32,768 sixteen-bit words. An additional 512,000 words of storage is available on-line with the Single Disk Storage feature of the 1131 Model 2. Single Disk Storage provides random or sequential access to data. The interchangeable disk cartridge places the required information at the disposal of the system and allows virtually unlimited off-line storage capacity. The 1131 Model 1 and the 1131 Model 2 have a core storage cycle time of 3.6 microseconds. The 1131 Model 3 has storage capacities of 8,192, 16,384, or 32,768 sixteen-bit words with a cycle time of 2.2 microseconds. The Model 3 also has the Single Disk Storage feature.

The CPU also includes a console with data displays and switches for operator control, a keyboard for data entry, and a console printer.

The basic 1130 system consists of an 1131 CPU Model 1, (with or without disk storage) and either card input/output or paper tape input/output. Card I/O can use the IBM 1442 Card Read Punch Model 6 or 7 or the IBM 2501 Card Reader Model A1 or A2 with the IBM 1442 Card Punch Model 5; the 2501 can also be used with the IBM 1442 Card Read Punch Model 6 or 7. Paper tape I/O uses the IBM 1134 Paper Tape Reader and the IBM 1055 Paper Tape Punch. To either configuration can be added: one or two 2310 Disk Storage units Model B1 or B2, additional core storage, an IBM 1132 Printer, an IBM 1403 Printer, or an IBM 1627 Plotter. The IBM 1231 Optical Mark Page Reader can be used in a 1442 card or a paper tape I/O system. Any system can include both paper tape I/O and card I/O and both line printers. The Synchronous Communications Adapter special feature provides communication with remote devices over telephone lines, and the Storage Access Channel provides the ability to attach additional devices and non-IBM devices to the system.

The IBM 1130 Computing System applications are varied and include some aspect of every industry, financial, and governmental operation. In the aerospace, construction, engineering, fabrication and assembly industries, the 1130 can be used for complex mathematical problems, operating analysis and scheduling, estimating, equipment and machine design, simulation, and job cost analysis.

In the processing industries, blending formulas, material balance, material evaluation, forecasting, and unit operations are a few of the applications suitable for the 1130.

Also, in many areas of the transportation, marketing, financial, insurance, utilities, and distribution fields, the 1130 system provides capability not previously available in a system of its size.

FUNCTIONAL CHARACTERISTICS

The ability of the IBM 1131 Central Processing Unit (Figure 1 and 2) to ask for and accept input data, perform the calculations required, and produce the output results desired is due to the many functional elements of the machine. Each of these elements is explained in this section, and from these descriptions the CPU emerges as the sum of its parts — the nerve center of the computing system.

The descriptions that follow concern the storage of data and program instructions, the formats in which data and instructions are stored and used, functions of CPU registers, the fundamental arithmetic operations and how they are performed, and the aspects of addressing core storage and attached input/output (I/O) units.

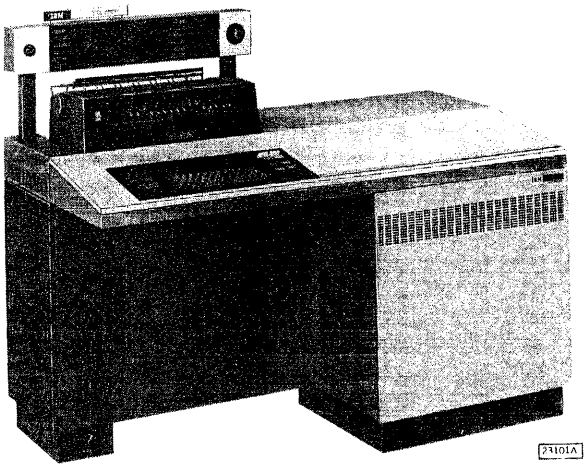


Figure 1. IBM Central Processing Unit (Model 1A, 1B, 2A, or 2B)

CORE STORAGE

The 1131 main storage uses magnetic cores for data and program instruction storage. Basic storage

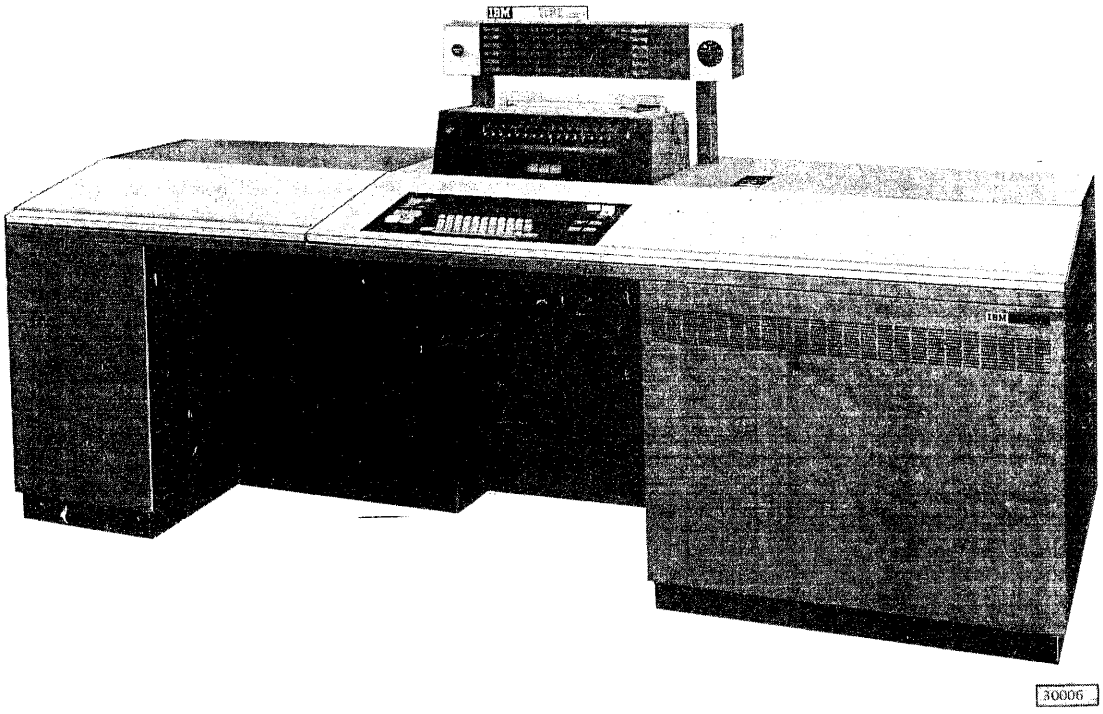


Figure 2. IBM 1131 Central Processing Unit (Model 2C, 2D, 3B, 3C, or 3D)

capacity is 4096 16-bit words; the core storage may be expanded up to 32,768 words. Each 16-bit word has two additional bits, called parity bits, which are used for internal data checking only.

The main storage memory cycle (the time required to place a word in core storage or to retrieve it from core storage) is 3.6 microseconds (μsec) in the Model 1 or Model 2 and 2.2 μsec in the Model 3.

Addressing

Each 16-bit word in core storage is locatable through an address that specifies the position of the word. Addresses range from 00000 to 32,767. The high-order address is contiguous with the low-order address, which provides for "wraparound" addressing. This means that in sequential processing of addresses the highest position of core storage (4095, or 8191, or 16383, or 32767) is followed by 00000 without further specification by the CPU.

Reserved Storage Locations

The following are core storage decimal addresses reserved for specific purposes and not available for general data storage.

Tag Bits	Core Storage Address	Description
00	--	Displacement
01	0001	Index Register 1
10	0002	Index Register 2
11	0003	Index Register 3
--	0008 - 0013	Interrupt Addresses
--	0032 - 0039	Printer Scan Field

20246

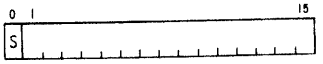
The use of each of the foregoing addresses is described in the appropriate subsequent section of this manual.

DATA FORMAT

Data in the 1131 CPU is in fixed-point binary form. Each number is treated as a signed integer: positive numbers are in true binary with a sign of 0, and negative numbers must be stored and operated upon in 2s complement form with a sign bit of 1. Complementing is done by inverting each bit of the number (including the sign bit) and adding 1 to the low-order bit. The following example illustrates this procedure.

Positive number	0001101001001100
Inverted	1110010110110011
Add 1	1
Resulting negative number	1110010110110100

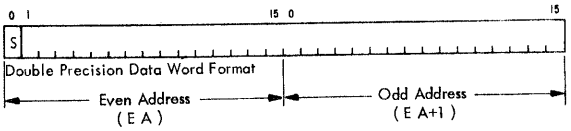
Data is stored as either a single precision word or a double precision word. A single precision data word comprises 16 bits; bit positions are numbered 0 to 15 from left to right. The high-order bit (0) is the sign position.



Single Precision Data Word Format

20070

The largest base-10 (decimal) values of single precision words are +32,767 and -32,768. A double precision data word contains 32 bits, and is composed of two sequential single precision words. The high-order bit (0) is the sign position.



20071A

A double precision data word is addressed by the leftmost word, which must have an even address.

The highest base-10 values of double precision data words are +2,147,483,647 and -2,147,483,648. The largest positive number ($2^{31}-1$) is one less than the largest negative number (2^{31}) because the sign (0 for plus, 1 for minus) is, arithmetically, part of the number.

All CPU storage is in binary form, and internal addressing and console displays are in 16-bit binary notation. Because of the ease of operation with 16-bit words in the hexadecimal number system (base 16), all programming systems for the IBM 1130 Computing System use this notation. Figures 3 and 4 show comparable values of decimal and hexadecimal number systems. Space obviously does not permit a complete listing; therefore, the value of each power of 2 through $2^{15}-1$ and $2^{31}-1$ are shown. The binary and hexadecimal number systems are described in Appendix A.

Positive Binary Values	Powers of 2	Absolute Values		Negative Binary Values
Bit Positions 11 1111 0123 4567 8901 2345		Decimal Notation Base-10	Hexa-decimal Notation Base-16	Bit Positions 11 1111 0123 4567 8901 2345
0000 0000 0000 0000	-	0	0	No negative zero
0000 0000 0000 0001	0	1	1	1111 1111 1111 1111
0000 0000 0000 0010	1	2	2	1111 1111 1111 1110
0000 0000 0000 0100	2	4	4	1111 1111 1111 1100
0000 0000 0000 1000	3	8	8	1111 1111 1111 1000
0000 0000 0001 0000	4	16	10	1111 1111 1111 0000
0000 0000 0010 0000	5	32	20	1111 1111 1110 0000
0000 0000 0100 0000	6	64	40	1111 1111 1100 0000
0000 0000 1000 0000	7	128	80	1111 1111 1000 0000
0000 0001 0000 0000	8	256	100	1111 1111 0000 0000
0000 0010 0000 0000	9	512	200	1111 1110 0000 0000
0000 0100 0000 0000	10	1,024	400	1111 1100 0000 0000
0000 1000 0000 0000	11	2,048	800	1111 1000 0000 0000
0001 0000 0000 0000	12	4,096	1,000	1111 0000 0000 0000
0010 0000 0000 0000	13	8,192	2,000	1110 0000 0000 0000
0100 0000 0000 0000	14	16,384	4,000	1100 0000 0000 0000
0111 1111 1111 1111	-	32,767	7,FFF	1000 0000 0000 0001
No positive equivalent	15	32,768	8,000	1000 0000 0000 0000

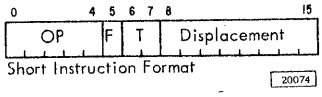
20072A

Figure 3. Value Ranges, Single Precision

INSTRUCTION FORMATS

Program instructions in the 1130 system are in either short or long format.

Short Instruction Format



Short Instruction Format 20074

OP (Operation) Code: These five bits specify the operation performed. Specific operations are described in the Operation section of this manual.

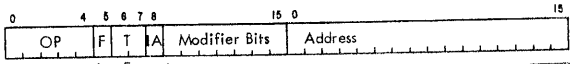
F (Format): The F bit controls the instruction format. A 0 indicates a short instruction format; a 1 designates a long instruction format.

T (Tag): These two bits specify the instruction address register or index register (XR) to be used for effective address generation.

Displacement: The data contained in these eight bits is added to the data in the instruction counter or index register specified by the tag bits to form the effective address (EA). (The Effective Address Generation section of this manual describes this and other aspects of address modification.) If the displacement amount is negative it is in 2s complement, and the sign, in bit position 8, is maintained in the resulting high-order position when the displacement is expanded to 16 bits for address modification.

NOTE: Displacement bits have other uses; for example, bits 8 and 9 are used as shift modifiers, etc.

Long Instruction Format



Long Instruction Format 20075A

The first eight bit positions of the long instruction are the same as the short format. The remaining bit positions of this double precision word are used as follows.

IA (Indirect Address): A 0 indicates a direct address (contained in the second word). A 1 bit in this position designates an indirect address, which is described in the Effective Address Generation section.

Modifier Bits: Bit positions 9 through 15 have various uses as modifiers and are described under the applicable instructions.

Address: These 16 bits contain the address which may be used in its current form or modified by indirect addressing and/or EA modification.

REGISTERS

The CPU has auxiliary storage areas, called registers, that are used to store data during the performance of operations directed by the stored program. Each register has a distinct purpose and is concerned with a specific type of data. Closely interrelated, they provide the CPU with the necessary functions to provide the results required.

Index Registers

Index registers are located in core storage and are used to contain data added to an instruction to provide an effective address. In a short instruction,

Positive Binary Values	Powers of 2	Absolute Values		Negative Binary Values
Bit Positions 11 1111 1111 2222 2222 2233 0123 4567 8901 2345 6789 0123 4567 8901		Decimal Notation Base - 10	Hexidecimal Notation Base - 16	Bit Positions 11 1111 1111 2222 2222 2233 0123 4567 8901 2345 6789 0123 4567 8901
0000 0000 0000 0000 0000 0000 0000 0000	-	0	0	No negative zero
0000 0000 0000 0000 0000 0000 0000 0001	0	1	1	1111 1111 1111 1111 1111 1111 1111 1111
0000 0000 0000 0000 0000 0000 0000 0010	1	2	2	1111 1111 1111 1111 1111 1111 1111 1110
0000 0000 0000 0000 0000 0000 0000 0100	2	4	4	1111 1111 1111 1111 1111 1111 1111 1100
0000 0000 0000 0000 0000 0000 0000 1000	3	8	8	1111 1111 1111 1111 1111 1111 1111 1000
0000 0000 0000 0000 0000 0000 0001 0000	4	16	10	1111 1111 1111 1111 1111 1111 1111 0000
0000 0000 0000 0000 0000 0000 0010 0000	5	32	20	1111 1111 1111 1111 1111 1111 1110 0000
0000 0000 0000 0000 0000 0000 0100 0000	6	64	40	1111 1111 1111 1111 1111 1111 1100 0000
0000 0000 0000 0000 0000 0000 1000 0000	7	128	80	1111 1111 1111 1111 1111 1111 1000 0000
0000 0000 0000 0000 0000 0001 0000 0000	8	256	100	1111 1111 1111 1111 1111 1111 0000 0000
0000 0000 0000 0000 0000 0010 0000 0000	9	512	200	1111 1111 1111 1111 1111 1110 0000 0000
0000 0000 0000 0000 0000 0100 0000 0000	10	1,024	400	1111 1111 1111 1111 1111 1100 0000 0000
0000 0000 0000 0000 0000 1000 0000 0000	11	2,048	800	1111 1111 1111 1111 1111 1000 0000 0000
0000 0000 0000 0000 0001 0000 0000 0000	12	4,096	1,000	1111 1111 1111 1111 1111 0000 0000 0000
0000 0000 0000 0000 0010 0000 0000 0000	13	8,192	2,000	1111 1111 1111 1111 1110 0000 0000 0000
0000 0000 0000 0000 0100 0000 0000 0000	14	16,384	4,000	1111 1111 1111 1111 1100 0000 0000 0000
0000 0000 0000 0000 1000 0000 0000 0000	15	32,768	8,000	1111 1111 1111 1111 1000 0000 0000 0000
0000 0000 0000 0001 0000 0000 0000 0000	16	65,536	10,000	1111 1111 1111 1111 0000 0000 0000 0000
0000 0000 0000 0010 0000 0000 0000 0000	17	131,072	20,000	1111 1111 1111 1110 0000 0000 0000 0000
0000 0000 0000 0100 0000 0000 0000 0000	18	262,144	40,000	1111 1111 1111 1100 0000 0000 0000 0000
0000 0000 0000 1000 0000 0000 0000 0000	19	524,288	80,000	1111 1111 1111 1000 0000 0000 0000 0000
0000 0000 0001 0000 0000 0000 0000 0000	20	1,048,576	100,000	1111 1111 1111 0000 0000 0000 0000 0000
0000 0000 0010 0000 0000 0000 0000 0000	21	2,097,152	200,000	1111 1111 1110 0000 0000 0000 0000 0000
0000 0000 0100 0000 0000 0000 0000 0000	22	4,194,304	400,000	1111 1111 1100 0000 0000 0000 0000 0000
0000 0000 1000 0000 0000 0000 0000 0000	23	8,388,608	800,000	1111 1111 1000 0000 0000 0000 0000 0000
0000 0001 0000 0000 0000 0000 0000 0000	24	16,777,216	1,000,000	1111 1111 0000 0000 0000 0000 0000 0000
0000 0010 0000 0000 0000 0000 0000 0000	25	33,554,432	2,000,000	1111 1110 0000 0000 0000 0000 0000 0000
0000 0100 0000 0000 0000 0000 0000 0000	26	67,108,864	4,000,000	1111 1100 0000 0000 0000 0000 0000 0000
0000 1000 0000 0000 0000 0000 0000 0000	27	134,217,728	8,000,000	1111 1000 0000 0000 0000 0000 0000 0000
0001 0000 0000 0000 0000 0000 0000 0000	28	268,435,456	10,000,000	1111 0000 0000 0000 0000 0000 0000 0000
0010 0000 0000 0000 0000 0000 0000 0000	29	536,870,912	20,000,000	1110 0000 0000 0000 0000 0000 0000 0000
0100 0000 0000 0000 0000 0000 0000 0000	30	1,073,741,824	40,000,000	1100 0000 0000 0000 0000 0000 0000 0000
0111 1111 1111 1111 1111 1111 1111 1111 No positive equivalent	- 31	2,147,483,647 2,147,483,648	7F,FFF,FFF 80,000,000	1000 0000 0000 0000 0000 0000 0000 0001 1000 0000 0000 0000 0000 0000 0000 0000

(2005-1)

Figure 4. Value Ranges, Double Precision Word

the amount in the displacement field of the instruction is added to the amount in the index register specified by the tag bits (6 and 7). The result becomes the effective address used by the instruction in the operation specified by the operation code. These operations and the functions of the EA are explained more fully in the Operation section.

Register Number	Instruction Code in Bits 6 and 7	Core Storage Location
1	01	0001
2	10	0002
3	11	0003

Machine Registers

The ten registers in the CPU are basic to the system and are functional elements of the CPU. Each register operates as necessary to enable the CPU to provide the results specified by the program. The abbreviation for each register name is the designation by which it is usually identified.

ACC (Accumulator): This 16-bit register contains the result of an arithmetic operation. It can be loaded from or stored in core storage, shifted right or left, and otherwise manipulated by specific arithmetic and logical instructions.

EXT (Accumulator Extension): This 16-bit register is the low-order extension of the ACC. It is used during multiply and divide operations, shifting of the ACC and EXT, and double-word arithmetic.

TAR (Temporary Accumulator): This 16-bit register is the image of the ACC and is used to store the contents of the ACC during effective address computation.

AFR (Arithmetic Factor Register): This 16-bit register holds one operand during arithmetic and logical operations. (The other operand is provided by the ACC.)

SBR (Storage Buffer Register): This 16-bit register is the buffer between the CPU and core storage, and every word of data transferred into or out of core storage passes through the SBR.

SAR (Storage Address Register): This 14-bit register contains the address pertaining to each reference to a core storage word.

IAR (Instruction Address Register): This 14-bit register holds the address of the next sequential instruction.

OP (Operation Register): This five-bit register holds the OP code of the instruction being performed.

TAG (Operation Tag Register): This three-bit register contains the F and T bits of the instruction. It controls the instruction length and selects the index register.

CCC (Cycle Control Counter): This six-bit register is used primarily to count CPU cycles and control shift operations.

ARITHMETIC FUNCTIONS

The arithmetic functions of the 1131 CPU are addition, subtraction, multiplication, and division. Results of arithmetic operations are algebraic. The correct sign is maintained as part of each operation.

Data is stored in the CPU in binary form; positive quantities have a plus sign (0) in the high-order position of the word and negative quantities have a minus sign (1) in the high-order position. Negative numbers are stored and operated upon in 2s complement form.

Addition and subtraction can be done in either single or double precision. Multiplication operates with single precision multiplier and multiplicand and provides a double precision product. In division, the dividend is double precision and the divisor and quotient are single precision.

Each arithmetic operation is described in detail under the specific program instruction in the Operation section.

Indicators

The two indicators associated with the ACC are the Overflow and Carry indicators. Each can be turned on irrespective of the other. The conditions of the indicators are explained under each instruction.

Carry Indicator: The Carry indicator is ON if the last position shifted out of the high-order position of the ACC contains a 1-bit. This indicator is reset for each add, subtract, or shift-left operation; it facilitates multiple precision (beyond double word) arithmetic.

Overflow Indicator: This indicator is turned on by an add, subtract, or divide operation when the result exceeds the capacity of the ACC or by a Load Status instruction in which the word at the EA has a 1 in bit position 15. The Overflow indicator can be turned off only by program test, a Store Status instruction, or a Load Status instruction in which the word at the EA has a 0 in bit position 15.

EFFECTIVE ADDRESS GENERATION

As has been noted previously, the location of a 16-bit single precision word or a 32-bit double precision word is denoted by a binary address. The range of addresses, expressed in decimal numbers, is 00000 through 32767. Most of the program instructions, which are explained in the Operation section, instruct the CPU to obtain the data at a specified location and perform a certain operation on it. For example, an Add instruction could say, in effect, add the amount stored at location 1904 to the amount in the accumulator and leave the result in the accumulator.

The location 1904 is the effective address of the data referred to by the instruction.

It is part of the versatility of the 1131 CPU that the address in the instruction being executed can be modified as a specific occasion requires. As the

result of a particular computation, for example, one of several courses may be indicated. Computation of the effective address of the location of the next instruction or of the next data worked on allows the CPU to proceed according to the predetermined course of action. The factors involved in computing the effective address are described in the following paragraphs.

Short Instruction

The short instruction displacement field contains the amount that is added to a specified figure to achieve the EA (effective address). The tag bits of the instruction designate the other factor. See Figure 5.

	F = 0 (Direct Addressing)	F = 1, IA = 0 (Direct Addressing)	F = 1, IA = 1 (Indirect Addressing)
T = 00	EA = Disp + IAR	EA = Add	EA = C/Add
T = 01	EA = Disp + XR1	EA = Add + XR1	EA = C/Add + XR1
T = 10	EA = Disp + XR2	EA = Add + XR2	EA = C/Add + XR2
T = 11	EA = Disp + XR3	EA = Add + XR3	EA = C/Add + XR3

Disp = Contents of Displacement field of instuction.
Add = Contents of Address field of instruction.
C = Contents of Location specified by Add or Add + XR.

20111 8

Figure 5. Determination Effective Address

IAR: Tag bits of 00 indicate that the displacement is added to the IAR to form the effective address. the IAR contains the address of the next or immediately following instruction.

Index Registers: The three index registers can also be used to modify the displacement to form the effective address. Tag bits of 01, 10, or 11 designate registers 1, 2, or 3. Again, the contents of the specified register, added to the displacement, form the effective address.

Long Instruction

Long instructions are modified in much the same way as short instructions with the added versatility of indirect addressing. See Figure 5.

Direct Addressing: In the long instruction, a direct address is indicated by a 0 in the IA field. The effective address is governed by the contents of the tag field. Tag bits of 00 indicate that the address field of the instruction contains the effective address, which requires no modification. Tag bits of 01, 10, or 11 specify that the contents of the Address field are added to index register 1, 2, or 3, respectively, to form the EA.

Indirect Addressing: A 1-bit in the IA field of the instruction signifies that addressing is indirect, i.e., the address field of the instruction contains the address of the location in memory that contains the EA significant to the accomplishment of the instruction. The indirect address can be the contents of the address field by the instruction (T = 00), or it can be modified by being added to an index register (T = 01, 10, or 11). As an example (Figure 6), the indirect address is 0914. The CPU goes to that address and finds the contents of the location to be 2719. The EA, then, is 2719. This provides one more level of modification of a given address and provides more versatility in programming for variations to the main line program.

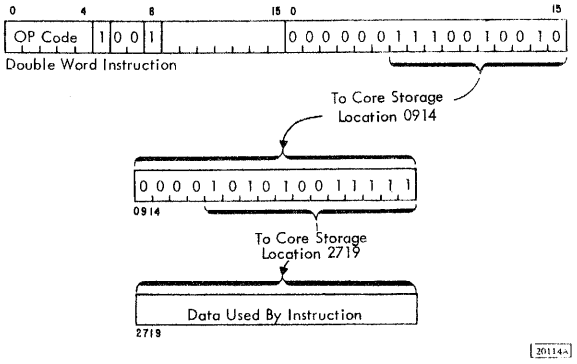


Figure 6. Indirect Addressing

OPERATION

The IBM 1130 instruction set (Figure 7) comprises individual instructions divided into five classes. Modifications of these instructions enable additional operations. In the descriptions that follow, the name of the instruction is followed by the mnemonic symbols, the binary representation of the operation code, and execution times.

LOAD AND STORE INSTRUCTIONS

Load ACC (LD-11000)

The contents of the memory location specified by the EA replace the contents of the ACC. The contents of the memory location are unchanged.

The Carry and Overflow indicators are not affected.

Load Double (LDD-11001)

The contents of the memory locations specified by the EA and EA + 1 are loaded into the ACC and EXT, respectively. This instruction provides a double-word load for use with double-word arithmetic. The EA must be even for correct operation. If the EA is odd, the contents of that location are loaded into both the ACC and EXT. The contents of the memory location are not changed.

Carry and Overflow indicators are not affected.

Store Accumulator (STO-11010)

The contents of the ACC replace the contents of the memory location specified by the EA. The contents of the ACC are not changed.

The Carry and Overflow indicators are not affected.

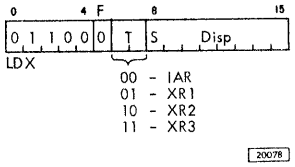
Store Double (STD-11011)

The contents of the ACC and EXT replace the contents of the memory locations specified by the EA and EA + 1. This instruction provides a double-word store for use with double-word arithmetic. The EA must be even for correct operation. If the EA is odd, the contents of the ACC are stored at the EA and the contents of the EXT are not stored. The contents of the ACC and EXT are not changed.

The Carry and Overflow indicators are not affected.

Load Index (LDX-01100)

The contents of the register specified by the tag bits of the instruction are replaced by the data specified. In the short instruction (F = 0), the register is loaded with the displacement. In the long instruction (F = 1) the register is loaded with the address (IA = 0) or the contents of the memory location specified by the address (IA = 1).



A tag of 00 results in an unconditional branch to the address loaded into the IAR.

The Carry and Overflow indicators are not affected.

Store Index (STX - 01101)

The contents of the register specified by the tag bits are stored in the memory location specified by the EA. (See the table under Load Index for tag bit codes.) The contents of the register are not changed.

The Carry and Overflow indicators are not affected.

Store Status (STS - 00101)

The status of the Carry and Overflow indicators are stored in bits 14 and 15, respectively, of the word at the EA. Bits 0-7 of the storage word remain unchanged; bits 8-13 are reset to zeros. The status of each indicator is reflected by storing a 1 bit if the indicator is on and a 0 if the indicator is off.

The Carry and Overflow indicators are reset as a result of the operation.

NOTE: The word in memory in which the status of the indicators is stored is normally the next Load Status instruction, the description of which follows.

Load Status (LDS-00100)

This instruction is always in the short format (F = 0). The Carry and Overflow indicators are set to the

Instruction	Mnemonic	Binary OP Code	Execution Times (in microseconds) for 3.6 µsec Core Only**							
			Single Word (F = 0)				Double Word (F = 1)			
			T = 00		T = 01, 10, or 11		T = 00		T = 01, 10, or 11	
			Avg.	Max.	Avg.	Max.	Avg. ①	Max. ①	Avg. ①	Max. ①
Load and Store										
Load ACC	LD	11000	7.6	-	11.2	-	10.8	-	14.8	-
Load Double	LDD	11001	11.2	-	14.9	-	14.4	-	18.0	-
Store ACC	STO	11010	7.6	-	11.2	-	10.8	-	14.8	-
Store Double	STD	11011	11.2	-	14.9	-	14.4	-	18.0	-
Load Index	LDX	01100	4.5	-	7.2	-	7.2	-	11.8	-
Store Index	STX	01101	7.6	-	11.2	-	11.8	-	15.4	-
Load Status*	LDS ⑦	00100	3.6	-	3.6	-	-	-	-	-
Store Status	STS	00101	7.6	-	11.2	-	10.8	-	14.8	-
Arithmetic										
Add	A	10000	8.0	13.0	11.7	16.6	11.2	16.2	15.3	20.3
Add Double	AD	10001	12.2	22.0	15.8	25.6	15.3	25.2	19.3	29.5
Subtract	S	10010	8.0	13.0	11.7	16.6	11.2	16.2	15.3	20.3
Subtract Double	SD	10011	12.2	22.0	15.8	25.6	15.3	25.2	19.3	29.5
Multiply	M	10100	25.7	40.0	29.3	43.6	29.3	43.6	32.9	47.2
Divide	D	10101	76.0	150.8	79.6	154.4	79.6	154.4	83.2	150.0
And	AND	11100	7.6	-	11.2	-	10.8	-	14.8	-
Or	OR	11101	7.6	-	11.2	-	10.8	-	14.8	-
Exclusive Or	EOR	11110	7.6	-	11.2	-	10.8	-	14.8	-
Shift Left* Modifier Bits 8 & 9:										
Shift Left ACC 00	SLA ⑦	00010	③	-	④	-	-	-	-	-
Shift Left ACC and EXT 10	SLT ⑦	00010								
Shift Left and Count ACC 01	⑧ SLCA ⑦	00010								
Shift Left and Count ACC and EXT 11	⑧ SLC ⑦	00010								
Shift Right* Modifier Bits 8 & 9:										
Shift Right ACC 00 or 01	SRA ⑦	00011								
Shift Right ACC and EXT 10	SRT ⑦	00011								
Rotate Right 11	RTE ⑦	00011	⑤		⑥					
Branch										
Branch and Store IAR	BSI	01000	7.6	-	11.2	-	10.8 ②	-	14.8	-
Branch or Skip on Condition	BSC	01001	3.6	-	3.6	-	7.2 ②	-	11.2	-
Modify Index and Skip	MDX	01110	4.5	9.9	11.2	16.2	18.5	23.4	18.5	23.4
Wait*	WAIT ⑦	00110 ⑨	3.6	-	3.6	-	-	-	-	-
Input/Output										
Execute I/O	XIO ⑩	00001	11.2	-	14.8	-	14.8	-	18.4	-

* Valid in short format only

** For 2.2 µsec core, multiply figures shown by 61%.

NOTES:

1 Indirect addressing, where applicable, adds 3.6 µsec to execution time
2 If branch is taken
3 3.6 + .45 (N-4)
4 7.2 + .45 (N-4)
5 N ≥ 16: 3.6 + .45 (N-19)
N = 16: 3.6
N < 16: 3.6 + .45 (N-4)
6 N > 16: 7.2 + .45 (N-19)
N = 16: 7.2
N < 16: 7.2 + .45 (N-4)
where N = number of positions shifted
7 Indirect addressing not allowed
8 If T = 00, functions as SLA or SLT
9 All unassigned OP codes are defined as Wait operations
10 If XIO Read or Write, add 3.6 µsec

20243B

Figure 7. 1130 Instruction Set and Execution Times

conditions indicated by bits 14 and 15, respectively, of the instruction. A 1 sets the indicator to the ON condition; a 0 sets it to the OFF condition.

The Carry and Overflow indicators are set according to the bits in positions 14 and 15.

NOTE: The Load Status instruction is the word in memory in which the status of the indicators is stored by the previous (Store Status) instruction.

ARITHMETIC INSTRUCTIONS

Add (A-10000)

The contents of the memory location specified by the instruction are added algebraically to the contents of the ACC. Negative data is in 2s complement form. The sum replaces the contents of the ACC. The contents of the memory location remain unchanged.

The Overflow indicator is turned on if the sum is greater than the capacity of the ACC, 2^{15} -1 or -2^{15}. If the indicator is ON when the overflow occurs, it is not changed.

The Carry indicator is set by a carry out of the high-order bit position of the ACC. The Carry indicator is dynamic and is conditioned for each Add instruction.

Add Double (AD-10001)

The contents of the memory locations at EA and EA +1 are added algebraically to the contents of the ACC and EXT. Negative data is in 2s complement form. This instruction provides double-word addition in which the ACC and EXT are considered as one 32-bit accumulator. The sum replaces the contents of the ACC and EXT; the contents of the memory locations are not changed. The EA must be even for correct operation. If the EA is odd, the contents of that location are added to both the ACC and EXT.

The Carry and Overflow indicators are affected as in the Add instruction (for the two-word result, of course).

Subtract (S-10010)

The contents of the memory location specified by the instruction are directly subtracted from the contents of the ACC. The result replaces the contents of the ACC. The contents of the memory location are not changed.

The Carry and Overflow indicators are affected as in the Add instruction. The Carry indicator, if on, reflects a borrow condition.

Subtract Double (SD-10011)

The contents of the memory locations at EA and EA +1 are subtracted from the contents of the ACC and EXT. This instruction provides double-word subtraction in which the ACC and EXT are considered one 32-bit accumulator. The difference replaces the contents of the ACC and EXT; the contents of the memory location are not changed. The EA must be even for correct operation. If the EA is odd, the contents of that location are subtracted from both the ACC and EXT.

The Carry and Overflow indicators are affected in the same way as in the Subtract instruction.

Multiply (M-10100)

The contents of the memory location specified by the instruction (the multiplicand) is multiplied algebraically by the contents of the ACC (multiplier). The 32-bit product replaces the contents of the ACC and EXT. Bit 15 of the EXT is the low-order bit, and bit 0 of the ACC is the high-order bit. Contents of the memory location are unchanged.

The Carry and Overflow indicators are not affected.

NOTE: The largest product that can be developed is 2^{30}, which results from multiplier and multiplicand of -2^{15}.

Divide (D-10101)

The contents of the ACC and EXT are considered a 32-bit, double word dividend, divided by the contents of the memory location specified by the instruction. The quotient replaces the contents of the ACC and the remainder, which carries the sign of the dividend, is placed in the EXT.

The Overflow indicator is turned on by an attempt to divide by zero or by a quotient overflow, which occurs when the quotient exceeds the range of -2^{15} to $2^{15}-1$. An overflow causes the ACC and EXT to be left in an undefined state.

Logical AND (AND-11100)

The contents of the memory location specified by the instruction are ANDed, bit-by-bit, with the contents of the ACC; the results replace the contents of the ACC. The contents of the memory location remain unchanged.

The AND operation compares each bit position of two words (fields) and places a 1 bit in the result field (ACC) position if both fields contain a 1 bit in that position. The table that follows illustrates the

four possible bit combinations in the same bit position of two ANDed words.

Memory	ACC	Result
0	0	0
0	1	0
1	0	0
1	1	1

The Carry and Overflow indicators are not affected.

Logical OR (OR-11101)

The contents of the memory location specified by instruction are ORed, bit by bit, with the contents of the ACC. The results replace the contents of the ACC; the contents of the memory location are unchanged.

The OR operation compares each bit position of two words (fields) and places a 1-bit in that position of the result field (ACC) if either field contains a 1-bit in that position. The table that follows illustrates the four possible bit combinations in the same bit position of the two ORed fields.

Memory	ACC	Result
0	0	0
0	1	1
1	0	1
1	1	1

The Carry and Overflow indicators are not affected.

Logical Exclusive OR (EOR-11110)

The contents of the memory location specified by the instruction are exclusive-ORed, bit by bit, with the contents of the ACC. The result replaces the contents of the ACC; the contents of the memory location are unchanged.

The exclusive-OR operation compares each bit position of two words (fields) and places a 1 bit in that position of the result field (ACC) if either field, but not both, contains a 1 bit in that position. The table that follows illustrates the four possible bit combinations in the same bit position of the two exclusive ORed fields.

Memory	ACC	Result
0	0	0
0	1	1
1	0	1
1	1	0

The Carry and Overflow indicators are not affected.

SHIFT INSTRUCTIONS

All shift operations are in the short format (F = 0) only. Each of the three Shift Right and four Shift Left instructions is defined by bits 8 and 9 of the basic Shift Right and Shift Left instructions. Except for the Shift Left and Count instructions, the number of positions shifted is controlled by the field specified by the tag bits, as shown by the table that follows. (XR is the abbreviation for index register.)

Tag Bits	Shift Controlled By Low-Order Six Bits
00	Displacement
01	XR1
10	XR2
11	XR3

If the shift count is zero in the control field addressed, the instruction performs as a No-Op, and the Carry indicator is not affected.

Shift Left ACC (SLA-00010)

Bits 8 & 9 = 00

The ACC is shifted left the number of positions specified by the shift count, and vacated (low-order) bit positions are set to 0. The EXT is not affected.

The condition of the Carry indicator is determined by the contents of the last bit position shifted out of the ACC. The Carry indicator is turned on by a 1 bit in the last position shifted out of the high-order position of the ACC; it is turned off by a 0. The Overflow indicator is not affected.

Shift Left ACC and EXT (SLT-00010)

Bits 8 & 9 = 10

The ACC and EXT are shifted left (as a 32-bit double word) the number of positions specified

by the shift count, and vacated bit positions are set to 0.

The Carry and Overflow indicators are affected as in the Shift Left ACC instruction.

Shift Left and Count ACC (SLCA-00010)

Bits 8 & 9 = 01

Tag bits of 00 cause this instruction to be executed the same as a Shift Left ACC instruction. Tag bits of 01, 10, or 11 cause the six low-order bits of the designated register to be transferred to the CCC (Cycle Control Counter) as a shift count. The count is decremented by one for each position the ACC is shifted to the left. The shift is terminated by a 1 bit being shifted to the high-order position of the ACC or the CCC being decremented to zero. The decremented count is loaded into the six low-order positions of the index register. Bit positions 0-7 of the index register are not affected.

The Carry indicator is turned ON if the shift is terminated by a 1 bit in the high-order of the ACC. It is turned off if the shift is terminated by the CCC being decremented to zero. If a 1 bit in the high-order position of the ACC coincides with the CCC being decremented to zero, the Carry indicator is turned off.

The Overflow indicator is not affected.

NOTE: If the count (n) is decremented to zero, a shift left n positions has occurred. If the count is initially zero or if the sign bit is initially a 1 bit, the instruction performs as a No-Op.

Shift Left and Count ACC and EXT (SLC-00010)

Bits 8 & 9 = 11

This instruction is the same as the Shift Left and Count ACC instruction, except that both the ACC and EXT are shifted. The high-order bits of the EXT are shifted into the low-order positions of the ACC, and the vacated low-order positions of the EXT are set to zero.

The Carry and Overflow indicators are the same as for the Shift Left and Count ACC instruction.

Shift Right ACC (SRA-00011)

Bits 8 & 9 = 00 or 01

The ACC is shifted right logically the number of positions specified by the shift count. Low-order bits shifted out are lost; high-order positions vacated are set to zeros. The EXT is not affected.

The Carry and Overflow indicators are not affected.

Shift Right ACC and EXT (SRT-00011)

Bits 8 & 9 = 10

The ACC and EXT are shifted right arithmetically (as a 32-bit double word) the number of positions specified by the shift count. The value of the sign-bit (position 0 of the ACC) is entered in all vacated positions. Low-order bits of the EXT are shifted out and lost.

The Carry and Overflow indicators are not affected.

Rotate Right ACC and EXT (RTE-00011)

Bits 8 & 9 = 11

The ACC and EXT are shifted right (as a 32-bit double word) the number of positions specified by the shift count. In effect, a continuous loop is formed, so that the high-order positions of the ACC pick up the bits shifted out of the low-order positions of the EXT. For example, if the shift count is three, all positions of the ACC and EXT shift three positions to the right, and the values of EXT bit positions 13, 14, and 15 are put in ACC bit positions 0, 1, and 2.

The Carry and Overflow indicators are not affected.

BRANCH INSTRUCTIONS

Branch instructions provide the means for departing from a sequential series of instructions, by testing to determine if a stated condition or combination of conditions exists, and returning to the point from which the departure was made.

Branch or Skip on Condition (BSC-01001)

Six separate conditions of the ACC can be tested by placing a 1 bit in the appropriate bit position of the instruction. The bit positions and corresponding conditions tested for are contained in the table that follows.

Bit Position	Condition
15	Overflow indicator OFF
14	Carry indicator OFF
13	ACC contents even
12	ACC positive, not zero
11	ACC negative
10	ACC zero

The contents of the ACC are not changed by testing.

Short Instruction Format (F = 0)

If any one of the conditions specified by the instruction is true, the program skips over the next word in memory and goes to the second word in sequence. This means that a BSC instruction in the short format must always be followed by a short-format instruction. If an instruction in the long format were to follow, a skip would send the program to the second word of the instruction, and a programming error would result.

If none of the conditions is true, the next sequential instruction is executed.

If bit positions 10 through 15 contain zeros (no condition tested), the instruction performs as a No-Op.

Long Instruction Format (F = 1)

When none of the conditions specified is true, the program branches to the EA. If any one of the conditions is true, the next sequential instruction is executed.

If no condition is specified, the program branches to the EA. This allows the long format of the BSC instruction to be used as an unconditional branch. An explanation of the computation of the EA is contained in the section Effective Address Generation. When this instruction specifies an indirect address (IA = 1), it enables a return to the program routine from which the CPU departed to accomplish a subroutine. This is accomplished by making the indirect address in this instruction identical to the EA of the branch and Store IAR instruction that effected the branch.

Programming Note: When an interrupt request has been detected by a priority level, the program is directed to service the request by interrupting. During the servicing, all interrupt requests of equal or lower status are effectively constrained from interrupting while the servicing of the higher priority is in progress. However, if a request is detected for a higher priority level than is presently in progress, the program is immediately interrupted again. This is frequently called nesting of interrupts.

At the completion of servicing any level of interrupt, it is necessary to signal the priority hardware to reset the priority status of the highest level that is on. This reset permits lower priority requests, including those that may have been temporarily constrained but recorded, to be accepted once again by the CPU. This is effected by making bit 9 = 1 in this instruction. This programmed recognition of waiting interrupts should not be confused with a normal subroutine linkage back to a mainline program, in which case bit 9 should be set to 0.

The BSC is a conditional instruction, and when bit 9 = 1, the interrupt level is reset only when the branch or skip occurs.

Indicators

The overflow indicator is reset when tested by the BSC instruction; the Carry indicator is not reset by testing.

Branch and Store IAR (BSI-01000)

The BSI instruction can be used in either the short or long format.

Short Instruction Format (F = 0)

The contents of the IAR (the location of the next sequential instruction) are stored at the EA of the BSI instruction. The IAR is then set to the EA + 1, which becomes the location of the next instruction executed. For example, assume that the BSI instruction is at memory location 0500 and that the EA generated is 0600. Execution of the BSI instruction stores 0501 at memory location 0600 and branches to 0601, which is the location of the next instruction.

Long Instruction Format (F = 1)

In the long format, the BSI instruction branches conditionally under the same circumstances as the BSC instruction. The conditions to be tested are designated by bit placement in bits 10-15, as shown by the table in the description of the BSC instruction. If none of the conditions is true, the contents of the IAR are stored at the EA and execution of the instruction proceeds as described for the short format. If one or more of the conditions is true, the next sequential instruction is executed.

Indicators

In the short format, the Carry and Overflow indicators are not affected; in the long format, the Overflow indicator is reset when tested. The Carry indicator is not reset.

Modify Index and Skip (MDX-01110)

This instruction can be used to modify an index register, the IAR, or the contents of a word in memory. Except as noted, a skip occurs only if the index register or memory word being modified changes sign or is zero after modification.

A skip causes the program to skip over the next word in memory and go to the second word in sequence. This means that an MDX instruction should normally be followed by an instruction in the short format. If a long-format instruction were to follow, a skip would send the program to the second word of the instruction, and a programming error would result.

Short Instruction Format (F = 0)

The expanded displacement is added to the register specified by the tag bits of the instruction, according to the table that follows. The displacement is expanded to 16 bits by duplicating the sign bit eight positions to the left in the resulting high-order position.

Tag Bits	Operation
00	Displacement added to IAR
01	Displacement added to XR1
10	Displacement added to XR2
11	Displacement added to XR3

When the Tag bits of the instruction are 00, the MDX instruction becomes a branch, a skip, or a No-Op. Since the IAR contains the address of the next instruction, a displacement value of zero merely sends the CPU to the next instruction; a positive value of one results in a skip; and any other value results in a branch to the modified address in the IAR. (The displacement can also be negative.)

Long Instruction Format (F = 1)

Modification is accomplished according to the contents of the Tag and IA fields of the instruction. If the tag is 00, independent of the IA bit, the expanded Displacement (bits 8 through 15 of the first word of the instruction) is added to the contents of the memory location specified by the address field of the instruction. The displacement is expanded to 16 bits by duplicating the sign bit eight positions to the left in the resulting high-order position. If the tag bits are not 00, the IA bit becomes the controlling factor, as shown below.

IA Bit = 0: The contents of the Address field of the instruction are added to the index register (XR) specified by the Tag bits:

T = 01 XR1
T = 10 XR2
T = 11 XR3

IA Bit = 1: The contents of the memory location specified by the address are added to the designated index register, according to the tag bit values noted above.

Indicators

The Carry and Overflow indicators are not affected.

Wait (WAIT-00110 and Undefined Op Codes)

This instruction is in the short format only. The operation of the CPU stops in a wait condition and can be restarted manually or by the detection of an interrupt. A manual restart causes resumption of the program with the next sequential instruction; an interrupt causes resumption at a point determined by the interrupt branch operation. Cycle stealing operations continue in the wait condition.

The Carry and Overflow indicators are not affected.

INPUT/OUTPUT OPERATIONS

The IBM 1130 Computing System offers a variety of I/O devices. The keyboard for input and the console printer for output are standard on the IBM 1131 Central Processing Unit (CPU) Models 1, 2 and 3. In addition, the 1131 Models 2 and 3 provide the large-capacity storage and random-access availability of data inherent in the Single Disk Storage. The following attached units offer a wide diversity of I/O media:

IBM 1134 Paper Tape Reader
IBM 1055 Paper Tape Punch
IBM 1442 Card Read Punch
IBM 1442 Card Punch
IBM 2501 Card Reader
IBM 1132 Printer
IBM 1403 Printer
IBM 2310 Disk Storage
IBM 1627 Plotter
IBM 1231 Optical Mark Page Reader

In addition the Storage Access Channel (SAC) provides the ability to attach special customer devices to the 1130 and the Synchronous Communications Adapter (SCA) provides the ability to communicate with remote devices and systems over common-carrier transmission lines.

The programmed operation of each of these units and features is described in succeeding sections. The operating procedures related to the mechanical functioning of the devices are explained in the publication "IBM 1130 Computing System, Input/Output Units" (Form A26-5890).

EXECUTE I/O (XIO-00001)

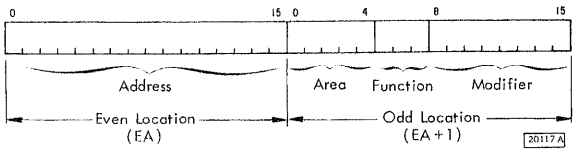
This instruction can be in either the short or long format, and operation is the same, except for the inherent differences in the manner in which the EA is generated, and the fact that the long format can have either a direct or indirect address.

The effective address is the memory location of the first word of the I/O control command (IOCC); EA + 1 is the location of the second word of the IOCC.

The contents of the ACC, if significant, must be stored prior to execution of the XIO instruction because the ACC is used in the analysis of the IOCC.

Input/Output Control Command

The format of the IOCC follows.



Address

The use of this 16-bit field depends on the function and the device specified.

Device

This 5-bit field (Area) identifies the I/O device.

1130 Device Codes (five-bit field of the IOCC)

Area Decimal	Code Binary	Device
1	00001	Console Keyboard & Printer
2	00010	1442 Card Read Punch
3	00011	1134 Paper Tape Reader & 1055 Paper Tape Punch
4	00100	Internal Single Disk Storage
5	00101	1627 Plotter
6	00110	1132 Printer
7	00111	Console Entry Switches
8	01000	1231 Optical Mark Page Reader
9	01001	2501 Card Reader
10	01010	Synchronous Communications Adapter
17	10001	2310 Disk Storage Drive 1
18	10010	2310 Disk Storage Drive 2
19	10011	2310 Disk Storage Drive 3
20	10100	2310 Disk Storage Drive 4
21	10101	1403 Printer

Function

The primary I/O functions are specified by the 3-bit function code:

000 – Not used

001 – Write
This code is used to transfer a single word from storage to an I/O unit. The address of the storage location is provided by the address field of the IOCC.

010 – Read
This code is used to transfer a single word from an I/O unit to storage. The address of the storage location is provided by the address field of the I/O Control Command.

011 – Sense Interrupt
This code is used to load the ACC with the interrupt level status word (ILSW) for the level being serviced.

100 – Control
This code causes the selected device to interpret the modifier field as a specific control action.

101 – Initiate Write
This code provides the ability to initiate a write operation on a device or unit which will subsequently make data transfers from storage via a data channel.

110 – Initiate Read
This code provides the ability to initiate a read operation from a device or unit which will subsequently make data transfers to storage via a data channel.

111 – Sense Device
This code loads the ACC with the DSW (device status word) for the device specified in the IOCC. The status indicators are reset by specifying modifier bits as follows: bit 15 for the highest level to which the device is connected, bit 14 for the next highest level, and so on.

Modifier

This portion of the command provides additional information for the device and function specified.

INTERRUPT

The interrupt facility provides an automatic branch from the normal program sequence, based upon an external condition. A maximum of six interrupt levels are available with the 1130 Computing System. They are assigned as follows:

Level	Device
0	1442 Card Read Punch (column read, punch)
1	1132 Printer, Synchronous Communications Adapter
2	Disk Storage, Storage Access Channel (SAC)
3	1627 Plotter, SAC
4	1442 (operation complete); Keyboard/ Console Printer; 1134 Paper Tape Reader; 1055 Paper Tape Punch, 2501 Card Reader, 1403 Printer, 1231 Optical Mark Page Reader, SAC
5	Console (Program Stop switch, and Interrupt Run), SAC

Interrupt Philosophy

Because of the number of types of interrupt requests, it is not always possible to cause a branch to a unique address for each interrupt condition. For the same reason, it is frequently not desirable to cause one branch for all interrupt requests and require the program to determine the individual request(s) requiring service. Therefore, it is expedient to group the many individual request lines into a lesser number of priority levels. This grouping accomplishes two very important functions: First, it allows all interrupt requests common to a specific device to have the privilege of interrupting immediately if the only requests waiting or being serviced are of a lower priority level. Service is returned to the initial request only after all higher level requests have been serviced. Second, since a unique branch can be defined for each interrupt priority level, it is possible to combine many interrupt requests on a common priority level and therefore use a common interrupt subroutine to service many requests.

There are two important operating characteristics of the 1130 interrupt system: (1) when more than one request line is connected to any priority level, it is necessary, by programming means, to identify the individual request(s) causing the priority level to be energized; (2) the first request that causes an interrupt prevents future requests on the same or lower priority levels from interrupting until the completion of servicing the first interrupt is signaled by a "branch out" operation. (See Branch or Skip on Condition-BSC.)

Interrupts that occur on the same level for which an interrupt is being serviced can be detected and acknowledged before the branch out operation is executed.

Program Operation

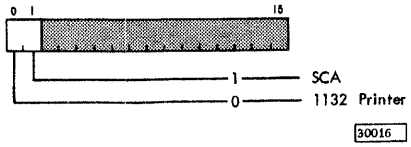
An interrupt may be recognized by the CPU at the completion of any program instruction. It is initiated by the basic interrupt control, which forces execution of a CPU-generated Branch and Store IAR (BSI) instruction. The indirect address of the generated BSI instruction is in location 8-13, corresponding to the level of interrupt. This location should contain the address of the location in the interrupt routine where the IAR is to be stored.

As defined by the BSI instruction description, the IAR is stored at the EA (effective address) and program execution is resumed with the branch to the EA+1. It is the responsibility of the interrupt subroutine to store all data and/or index registers that are used by the routine, and to restore the same registers prior to departing from the subroutine. (See the description of BSC.)

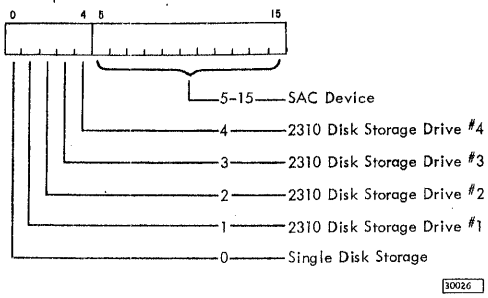
Several devices can request an interrupt on most levels. It thus becomes necessary for the program to determine the requesting device. This is accomplished by issuing an XIO instruction with a function of Sense Interrupt

The Sense Interrupt function is decoded and sent to all I/O devices, along with the current interrupt level being serviced. Each device is requesting service on the current level will have a bit appear in the ILSW that is loaded into the accumulator.

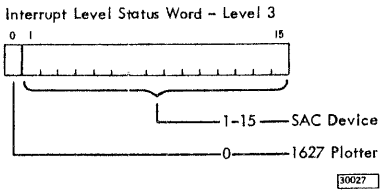
Interrupt Level Status Word - Level 1



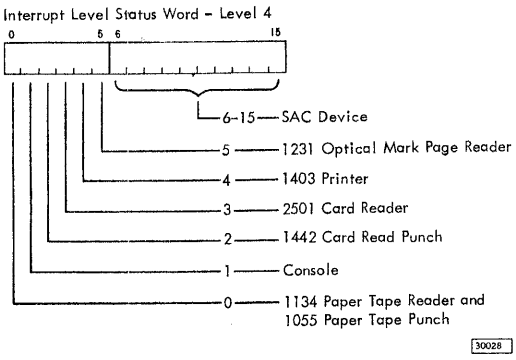
Interrupt Level Status Word - Level 2



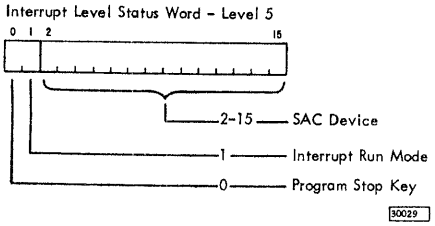
Interrupt Level Status Word - Level 3



Interrupt Level Status Word - Level 4



Interrupt Level Status Word - Level 5



Each device is given a particular bit position in its ILSW to indicate its interrupt request status, a 1 bit if on and 0 if off. The status indicator(s) in the device(s) is not affected by the sensing of the ILSW. It is possible for a device to contain several conditions which may cause an interrupt on the same interrupt level. When this condition exists, the interrupt conditions are logically ORed to become a single interrupt. The identification of the interrupting condition within the device is accomplished by sensing the device status word (DSW) as discussed in subsequent paragraphs.

Interrupt Identification

Following loading of the ILSW in the ACC (accomplished by an XIO Sense Interrupt instruction), the Shift Left and Count instruction is used to facilitate examination of the ILSW. First, an index register is loaded with a quantity which corresponds to the number of request signals connected to a particular interrupt level, followed by the Shift Left and Count instruction (SLC). The resulting count in the index register is unique and corresponds to the first non-zero bit of the ILSW in the Accumulator. (It is also possible to execute a Shift Left and Count of both the ACC and EXT. Refer to the SLC Instruction Description.) The SLC is followed by a Branch or Skip on Condition instruction (BSC) utilizing the F = 1 format with IA = 1, indexed with the result of the SLC. This provides, in conjunction with a branch table, a unique branch for each non-zero bit of the ILSW.

After the device causing an interrupt has been identified from data in the ILSW, it is necessary to determine the indicator(s) within the particular device causing the interrupt. This is accomplished by issuing a subsequent XIO Sense Device instruction with an area assignment corresponding to that of the device being interrogated. The status indicators are reset after the information has been loaded in the ACC if a bit is present in position 15 of the modifier. If a device can initiate interrupts on more than one interrupt level, the indicators are reset by specifying modifier bits as follows: bit 15 for the highest level to which the device is connected, bit 14 for the next highest level, and so on.

The data in the ACC is now referred to as the DSW (device status word).

Device Status Word

The DSW contains one bit of information for each indicator within the device. These usually fall into three categories, (1) error or exception interrupt conditions, (2) normal data or service-required interrupts, and (3) routine status conditions.

Programming Note: When an interrupt request has been detected by a priority level, the program is directed to service the request by interrupting. During the servicing, all interrupt requests of equal or lower status are effectively constrained from interrupting while the servicing of the higher priority is in progress. However, if a request is detected for a higher priority level than is presently in progress, the program is immediately interrupted again. This is a frequently called nesting of interrupts.

At the completion of servicing any level of interrupt, it is necessary to signal the priority hardware to reset the priority-status of the highest level that is on. This reset permits lower priority requests (including those that may have been temporarily constrained, but recorded) to be accepted once again by the CPU. The reset is accomplished by a BSC Instruction with bit 9 = 1. This programmed recognition of waiting interrupts should not be confused with a normal subroutine linkage back to a mainline program, in which base bit 0 should be set to zero.

The BSC is a conditional instruction, and when bit 9 = 1, the interrupt level is reset only when the branch or skip occurs.

Figure 8 shows a typical procedure for recognizing interrupts.

1131 SINGLE DISK STORAGE

Disk storage provides the IBM 1130 Computing System with low-cost random or sequential access data storage. On-line data capacity is 512,000 words (may be expanded up to 2,560,000; see IBM 2310 Disk Storage); off-line capacity is virtually unlimited because the interchangeable disk cartridge is easily removed and replaced with another. Thus, the large storage capacity, comparable to that of magnetic tape, coupled with the unique advantage of random access, affords the 1130 Computing System great flexibility in the handling of engineering, scientific, industrial, and commercial programs.

System programs, object programs, subroutines, and often-used table data can be stored in the same removable disk cartridge and a specific computation accomplished by mounting the cartridge and feeding the variable data into the 1131 CPU. This simplifies problem solving and increases the throughput of the system.

DESCRIPTION

Single disk storage for the 1130 System is contained in the CPU cabinet and is connected to the CPU by a high-speed data channel. It is composed of two components: the disk and drive assembly and the access mechanism.

Disk Assembly

The disk assembly (Figure 9) is a single disk drive, completely enclosed in a protective housing, or cartridge. The recording medium is an oxide-coated disk that provides two surfaces for the magnetic recording of data. When mounted in the CPU enclosure, the disk drive rotates at the rate of 1500 revolutions per minute.

Access Mechanism

The disk storage access mechanism has two horizontal arms. Each arm has a magnetic read/write head, and each head is positioned to read or write on the corresponding disk surface as the access arms straddle the disk in the manner of a large tuning fork. The entire assembly moves horizontally forward and backward, so that the heads have access to the entire recording area.

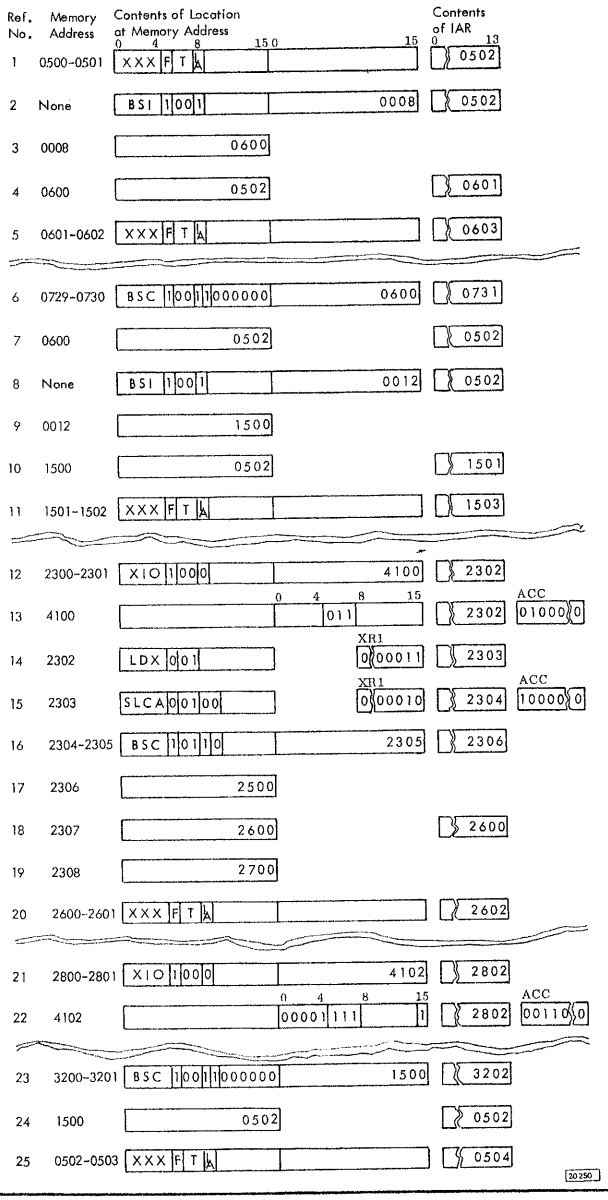
The access mechanism is positioned automatically, at the home position (outside cylinder) when the disk cartridge is inserted.

Disk Organization and Capacity

The access mechanism is moved back and forth by program instructions and can be placed in any one of 200 positions, from a point near the periphery of the disk to a point near the center of the disk. At each position, the heads can read or write in a circular pattern on both surfaces of the disk, as it revolves. These circular patterns of data are called tracks. The track on the upper surface of the disk and the corresponding track on the lower surface, both of which can be read or written while the access mechanism is in the same position, are called a cylinder. Figure 10 shows the innermost and outermost cylinders of two tracks each. To complete the picture, the 198 intermediate cylinders, or pairs of tracks, should be visualized; they were omitted for the sake of clarity of the diagram.

For convenience in transferring data between the CPU core storage and disk storage, each track is divided into four equal segments called sectors. Sectors are numbered by the cylinder, from 0 through 7, as shown in Figure 11. Sectors 0 - 3 divide the upper surface track, and sectors 4 - 7, the lower. A sector contains 321 data words and is the largest segment of data that can be read or written with a single instruction.

Sample Interrupt Recognition Procedure



The notes that follow are numbered to correspond to the reference numbers in the procedure. Each reference number cited in text is circled, e.g., ①, to avoid confusion with numbers necessary to the procedure, such as memory addresses.

In the registers, instructions, and data words, only the necessary 0-bits and 1-bits are shown. Op codes are shown in alphabetic symbols, and decimal numbers are used to identify memory locations. Binary notation, where used, is obvious. IAR, ACC, and XR1 are shown only where needed for understanding of the operation.

1. Mainline program instruction. During execution of this instruction, the 1442 initiates a card read interrupt.

2. At the conclusion of the ① instruction, the CPU blocks the next program instruction and interposes a CPU-generated BSI to start the Level 0 Interrupt procedure.

3. IA of the Level 0 BSI is 0008; the EA at 0008 is 0600.

4. The IAR (0502) is stored at the EA (0600); the IAR is then loaded with the EA + 1 (0601).

5. First instruction of the Level 0 interrupt subroutine. The subroutine must store the status of each register, all data, etc., that could be altered by execution of the subroutine. Before leaving the subroutine, the program must restore all registers, data, etc., to the condition that existed when the interrupt occurred.

6. The last instruction of the interrupt subroutine is a BSC instruction with a Bit-9 value of 1, which resets the priority status so that interrupts of equal or lower priority can be recognized. If no interrupt is waiting, return to the interrupted program is effected by the IA(0600) of the BSC being equal to the EA of the CPU-generated BSI that initiated the interrupt routine, ②. The BSC is shown as an unconditional branch (Bits 10-15 = 0); the branch could have been conditional, i.e., the branch executed only if conditions specified by Bits 10-15 were true.

NOTE: The term interrupted program is used to designate either the main program being executed or an interrupt subroutine of lower priority. For example, the ① instruc-

Figure 8. Sample Interrupt Recognition Procedure, Part I

tion could be in a routine to service a console printer-keyboard, Level 4 interrupt. Thus, the mainline program can be thought of as a routine with no priority, to which the CPU returns when no interrupts are waiting.

7. The EA (0502) is the location of the next mainline program instruction and is loaded into the IAR.

8. To illustrate an interrupt with a low priority occurring while a higher priority interrupt is being serviced, we assume that the console printer-keyboard initiated a Level 4 interrupt while the card read interrupt was being serviced. We assume that no Level 0, 1, 2, or 3 interrupts are waiting and that the Level 4, CPU-generated BSI can now be interposed, as in (2) for Level 0.

9. The IA (0012) of the BSI is the memory location assigned to Level 4 interrupts and contains the EA (1500).

10. The IAR is stored at the EA (1500) and then loaded with EA+1 (1501).

11. First instruction of Level 4 subroutine. See (5) . Last housekeeping instruction takes subroutine to 12 .

12. The XIO instruction EA (4100) is the memory location of the IOCC. The IAR contents remain at 2302 because the IOCC controls an I/O device and is not a sequential program instruction.

13. The IOCC function code of 011 (Sense Interrupt) causes the ILSW for Level 4 to be loaded into the ACC.

14. XR1 is loaded with a quantity equal to the number of response signals connected to the ILSW.

15. SLCA instruction is terminated when the 1-bit associated with the console printer-keyboard interrupt is shifted into the high order position of the ACC. XR1 is reduced by one.

16. BSC instruction address is modified by XR1 (+2) to form the IA (2307). A bit in the 0-position of the ILSW (paper tape reader and paper tape punch) results in an XR1 of 3 and an IA of 2308. A bit in the 2-bit position (card read-punch) results in an XR1 of 1 and an IA of 2306.

17. An IA of 2306 has the EA 2500, which is the memory location of the first instruction of the Card Read-Punch interrupt subroutine.

18. An IA of 2307 has the EA 2600, which is the memory location of the first instruction of the Console Printer-Keyboard interrupt subroutine.

19. An IA of 2308 has the EA 2700, which is the memory location of the first instruction of the Paper Tape Reader and Paper Tape Punch interrupt subroutine.

20. First instruction of housekeeping sequence for Console Printer-Keyboard subroutine.

21. The XIO instruction EA (4102) is the memory location of the console printer-keyboard IOCC.

22. The IOCC Sense Device function code (111) causes the DSW of the console printer-keyboard (00001) to be loaded into the ACC. The 1-bit in position 15 causes the response to be reset. The example shows 1-bits in positions 2 and 3 of the ACC (DSW), which indicate that the operator initiated an interrupt request on the keyboard and that the console entry switches are to be read. A programmed subroutine determines the cause of the interrupt (Bit 2) and the interrupting device (Bit 3). A routine then follows that reads the data into memory, accomplishes any housekeeping required, and releases the CPU, as shown in (23) .

23. Procedure is the same in (6) and (7) . The IA (1500) of the BSC instruction is equal to the EA of the CPU-generated BSI instruction that initiated the interrupt; see (8) , (9) , and (10) .

24. The EA (0502), located at 1500, is loaded into IAR.

25. The instruction at 0502 is the next one to be executed in the mainline program. See (1) for previous mainline instruction.

Figure 8. Sample Interrupt Recognition Procedure, Part II

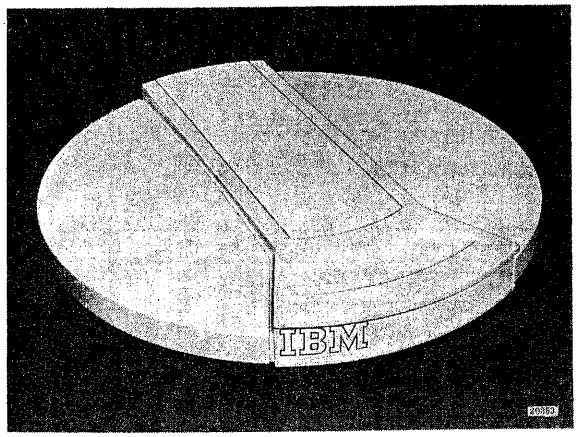


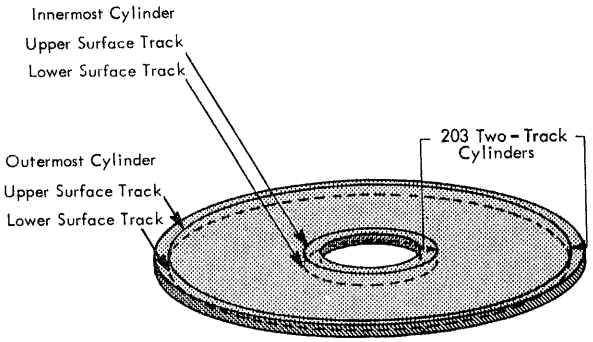
Figure 9. Disk Assembly Cartridge

In the programs and programming systems provided by IBM, e.g., the monitor system and its programs, the first word of a 321-word sector is used for cylinder sector number.

Therefore, the first word of the sector can not be used by the programmer if the assembler program or other components of the monitor system are to be used.

A disk storage word comprises 16 data bits and four check and space bits.

Table 1 shows the organizational components of disk storage. Note that capacities are based on the 320-word sector.



NOTE: The thickness of the disk has been greatly exaggerated in order to show the relative positions of the upper and lower surface tracks.

20254A

Figure 10. Disk Storage Cylinder Schematic

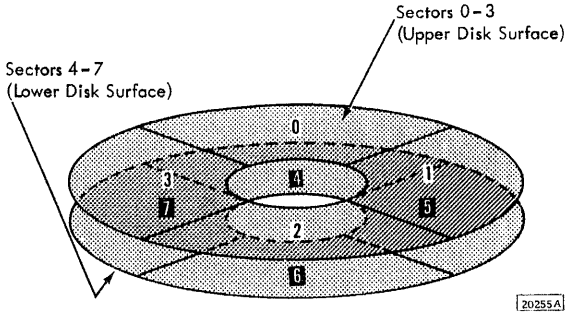


Figure 11. Disk Storage Sector Numbers

Timing

Timing considerations of disk storage operation involve three elements: access time, reading and writing data, and the time during which the CPU is tied up.

Access: The access mechanism moves in increments of two cylinders at the rate of 15 ms per increment. Thus, in the formula that follows, the number of cylinders (N) must be even. (The next higher even number is used if an odd number of cylinders is specified.) During the 20 ms to 25 ms stabilization period that follows the last incremental movement, a Read or Write instruction can be given and will be started at the end of the stabilization period.

$$\text{Access time (ms)} = 7.5(N) + \tau$$
$$\text{where} \quad 20 \text{ ms} \leq \tau \leq 25 \text{ ms}$$

Read/Write: Reading or writing of data in disk storage is at the rate of 27.8μsec per word. Average rotational delay time is 20 ms, based on 1500 rpm, or 40 ms per revolution. Thus, a sector can be read or written in an average of 30 ms. Although there are no timing considerations for head switching, there are programming considerations in consecutive sector operations because there is an interval

Table 1. Disk Storage Data Organization

No of. \ Per →	Word	Sector	Track	Cylinder	Disk
Bits	16	5,120	20,480	40,960	8,192,000
Data Words		320	1,280	2,560	512,000
Sectors			4	8	1,600
Tracks				2	400
Cylinders					200

of over 420 μsec between sectors; the interval is increased by 27.8 μsec for each word less than 321 read or written.

A full cylinder of eight 321-word sectors can be read or written in 100 ms because the rotational delay is required for only the first sector.

CPU Time: An interrupt in a disk storage operation occurs only at the end of the Seek or Read/Write operation. This means that once the instruction is initiated, disk storage operation is virtually independent of the CPU. As data is being read or written, a cycle is literally "stolen" from the CPU operation in progress every 27.8 μsec for the transmission of the next word. Thus, except for the normal instruction times, the CPU is busy only 14 ms of the 100 ms required to read or write a full cylinder. The remaining 86 ms are available for other program operations.

Data Checking

Data is checked on each transmission between core storage and disk storage. The number of bits of each word is divided by four as the word leaves core storage, and the number of bits necessary to make the division even (modulo 4) is added to the end of the word. The modulo 4 test is performed again each time a word is written in disk storage or read from it. A word that is not modulo 4 causes the data check bit to be set in the disk storage DSW.

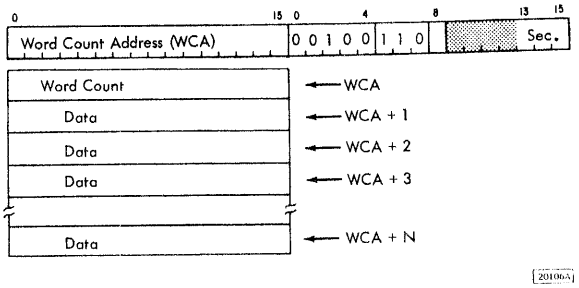
Procedure for Changing the Disk Storage Cartridge

1. Turn off the disk drive motor with the power switch adjacent to the disk mechanism.
2. Open the hinged cover of the disk storage drive enclosure.
3. Pull down on the release/lock handle and remove the cartridge. An interlock prevents removal of the cartridge until the disk has stopped spinning.
4. Another cartridge can be installed by simply inserting it into the aperture. The access mechanism read/write heads are automatically positioned as the cartridge is inserted.
5. Raise the access release/lock handle to lock the cartridge in place.
6. Close the cover of the disk storage drive enclosure and start the disk drive motor. An interlock prevents the motor from starting unless the cartridge is correctly inserted and the disk is in place. The disk reaches ready status in approximately 90 seconds.

PROGRAMMING SINGLE DISK STORAGE

I/O Control Command (IOCC)

Initiate Read (110)



This instruction causes the number of words specified by the word count to be read from the disk storage sector identified by modifier bits 13-15. The address word of the instruction contains the WCA (word count address), and modifier bit 8 determines whether the command is a Read instruction (0) or a Read-Check instruction (1).

A full sector, 321 words, is the maximum transmission with one instruction. Succeeding sectors, or parts of sectors, require an Initiate Read instruction for each one.

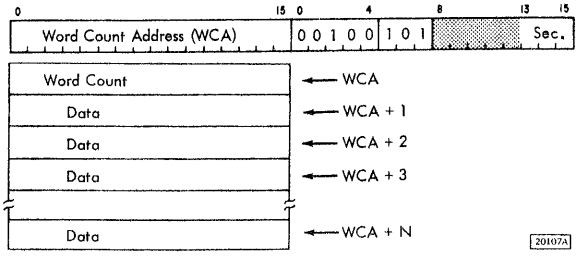
An operation-complete interrupt occurs when the number of words in the word count has been transmitted.

Read Instruction (Bit 8 = 0): Beginning with the first word of the indicated sector, data is read into core storage location WCA + 1 and ascending addresses. The word count, which is stored at the location specified by the WCA, controls the number of words transmitted and, consequently, the number of core storage locations occupied by the disk storage data. For example, assume that a word count of 152 is stored at WCA 1000. The 152 words read from disk storage would be stored at addresses 1001 through 1152.

The programmer must be aware of the core storage locations required for incoming disk storage data so that useful data is not written over and lost.

Read-Check Instruction (Bit 8 = 1): Data is read from disk storage, as in the Read instruction, and the number of bits of each word is checked for modulo 4. If the division for any word is not even, the Data Check indicator bit is set in the disk storage DSW. Neither disk storage nor core storage is affected by the Read-Check instruction.

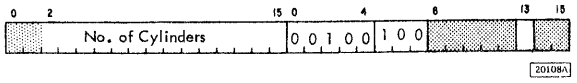
Initiate Write (101)



This instruction causes the number of words specified by the word count to be written in disk storage, beginning at the first word of the sector indicated by modifier bits 13-15. The address word of the instruction contains the address of the Word Count (WCA). The data is transmitted from core storage location WCA + 1 and ascending addresses. A full sector, 321 words, is the maximum transmission with one instruction. Succeeding sectors, or parts of sectors, require an Initiate Write instruction for each one.

An operation-complete interrupt occurs when the number of words in the word count has been transmitted.

Control (100)



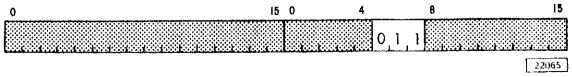
This instruction causes the access mechanism to move in increments of two cylinders for the number of cylinders specified by the Address word of the instruction. If the number of cylinders is odd, the first increment consists of one cylinder.

Modifier bit 13 controls the direction of movement: a 0 moves the access mechanism forward (toward the center of the disk); a 1 moves it backward.

When the access mechanism has moved the number of cylinders specified, an operation-complete interrupt occurs.

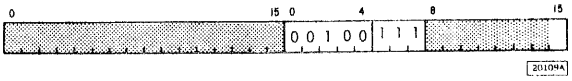
NOTE: Cylinders do not carry an identifying number. It is the responsibility of the program, therefore, to maintain the necessary information relative to the position of the access mechanism. A Control command which specifies an access motion of zero cylinders is treated as a No-Op and does not result in an operation-complete interrupt.

Sense Interrupt (011)



This command causes the accumulator to be loaded with the interrupt level status word (ILSW) for the highest interrupt level in progress. The Single Disk Storage is assigned to interrupt level 2. If the Single Disk Storage is the interrupting device, the ILSW contains a 1 bit in bit position 0 (see IBM 2310 Disk Storage for assignment of other disk storage drives).

Sense Device (111)



This instruction causes the device status word (Figure 12) of disk storage to be read into the ACC. Operation complete and data error (except select and unsafe) indicators are reset if modifier bit 15 is a 1.

Interrupt

Operation Complete: This is the only interrupt associated with disk storage, and is turned on at the end of a read, read-check, write, or control (access) operation. It also occurs if the disk storage is in a read, read-check, or write operation at the leading edge of a sector pulse.

Indicators

Data Error: This indicator is turned on when:

1. A modulo-4 error is detected during a read, read-check, or write operation.

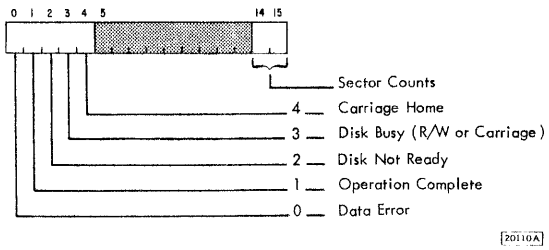


Figure 12. Disk Storage Device Status Word

2. The disk storage is in a read or write mode at the leading edge of a sector pulse.
3. A write select error has occurred in the disk storage drive.
4. The power unsafe latch is set in the attachment.

Conditions (1) and (2) are turned off by a Sense Device instruction with modifier bit 15 set to one. Conditions (3) and (4) are reset by turning off the Single Disk Storage, allowing for the cartridge unlock indicator to light, turning on the Single Disk Storage and waiting until the disk ready indicator (heads loaded for the 2310) to light. After this sequence of events disk operation can resume.

Operation Complete: This indicator is turned on at the end of:

1. A read, read-check, or write operation when the word count is reduced to zero.
2. A read, read-check, or write operation at the leading edge of the following sector pulse if the word count is not reduced to zero by that time.
3. A carriage seek operation.

Disk Not Ready: This indicator is turned on with disk not ready or busy or disabled or off-line or power unsafe latch set. Also included in the disk not ready is the write select error, which can be a result of power unsafe or write select (bit 0 and bit 2 will be turned on).

Disk Busy (R/W or Carriage): This indicator is on during execution of a disk storage instruction. It turns off when the operation is completed.

Carriage Home: This indicator is on when the access mechanism is at the home position (cylinder 000).

Sector Count: These bits represent the sector number of the next available sector to be used for reading or writing.

Programming Considerations

Disk Organization

It is important in planning a routine for loading disk storage that the cylinder concept be taken into consideration. Related data should be grouped in the same cylinder, when possible, to eliminate unnecessary seek operations. Therefore, when disk addresses are assigned to a group of related data, the disk locations made available should be limited to the number required, plus an expansion factor. The most frequently used data should be stored in the low-numbered cylinders to minimize seek time.

Customer Error Correction Routines

In the event that an error is detected by the CPU circuitry, it is recommended that the following procedure be executed:

1. Re-seek the cylinder upon which the error was detected.
2. Re-execute the operation in which the error occurred.

This procedure should be executed from three to ten times prior to establishing the occurrence of a disk error.

CONSOLE

The Console (Figure 13) is an integral part of the IBM 1131 Central Processing Unit and consists of the input keyboard, console printer, display panel, function switches and lights and Console Entry switches.

While the keyboard and console printer are usually considered as one unit, control of each of them by the operator and by the stored program is discrete. For this reason, the functional description and programmed operation of each unit is considered separately in the sections that follow.

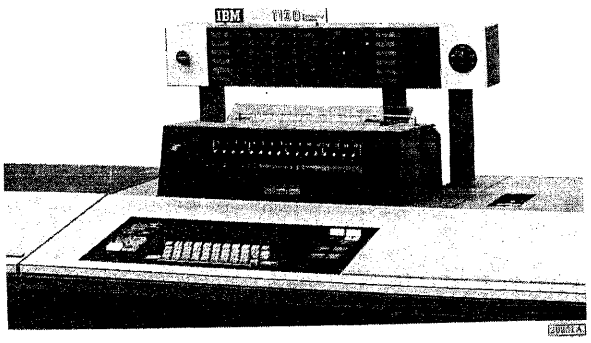


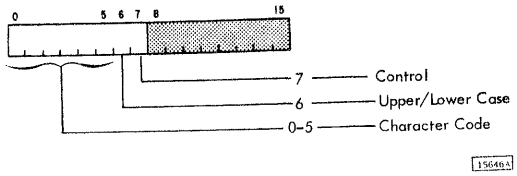
Figure 13. 1131 Console

CONSOLE PRINTER FUNCTIONAL DESCRIPTION

The console printer provides output at a maximum rate of 15.5 characters per second. Data to be printed is transferred from core storage to the console printer by direct program control.

Data characters and control characters (space, tabulate, etc.), are sent to the console printer by means of the Write command. Because control characters and data characters are sent in the same manner, the message to be printed contains a mixture of data characters and control characters in the sequence necessary to give the desired formatted output.

The character format within a core storage word to be transmitted to the console printer is:



Each word transmitted to the console printer contains one data character or one control character.

Data Coding

Data to be printed is coded by the program into the console printer code. Figure 14 shows the characters which can be printed by the standard print element.

The data-character code also contains (in B6) the information as to whether the character is an uppercase (UC) shift or lower-case (LC) shift character. The printer shifts automatically as required for each data character.

A printer Write command is modified by the B7 position of the output character word. If B7 equals one, the Write command to the printer is interpreted as a control function. If B7 equals zero, the Write command is interpreted as a print function.

The codes for console printer control functions are shown in Figure 15.

PRINTER PROGRAMMING

The console printer operates in the 1130 System under direct program control.

Character Code Bits						U/L Case		Ctrl
B0	B1	B2	B3	B4	B5	B6=0 LC	B6=1 UC	B7
0	0	1	1	1	1	A	A	0
0	0	0	1	1	0	B	B	0
0	0	0	1	1	1	C	C	0
0	0	1	1	0	0	D	D	0
0	0	1	1	0	1	E	E	0
0	0	0	1	0	0	F	F	0
0	0	0	1	0	1	G	G	0
0	0	1	0	0	1	H	H	0
0	0	1	0	0	0	I	I	0
0	1	1	1	1	1	J	J	0
0	1	0	1	1	0	K	K	0
0	1	0	1	1	1	L	L	0
0	1	1	1	0	0	M	M	0
0	1	1	1	0	1	N	N	0
0	1	0	1	0	0	O	O	0
0	1	0	1	0	1	P	P	0
0	1	1	0	0	1	Q	Q	0
0	1	1	0	0	0	R	R	0
1	0	0	1	1	0	S	S	0
1	0	0	1	1	1	T	T	0
1	0	1	1	0	0	U	U	0
1	0	1	1	0	1	V	V	0
1	0	0	1	0	0	W	W	0
1	0	0	1	0	1	X	X	0
1	0	1	0	0	1	Y	Y	0
1	0	1	0	0	0	Z	Z	0
1	1	1	1	1	1	1	(	0
1	1	0	1	1	0	2	+	0
1	1	0	1	1	1	3	<	0
1	1	1	1	0	0	4	¬	0
1	1	1	1	0	1	5	)	0
1	1	0	1	0	0	6	;	0
1	1	0	1	0	1	7	*	0
1	1	1	0	0	1	8	'	0
1	1	1	0	0	0	9	"	0
1	1	0	0	0	1	0	|	0
1	1	0	0	0	0	#	=	0
1	0	1	1	1	1	/	-	0
1	0	0	0	0	1	-	?	0
1	0	0	0	0	0	,	:	0
0	1	0	0	0	1	&	>	0
0	1	0	0	0	0	$	.	0
0	0	0	0	0	1	@	%	0
0	0	0	0	0	0	.	¢	0

20085

Figure 14. Console Printer Character Coding

I/O Control Commands (IOCC)

The console printer is addressed by a five-bit device code in the IOCC, 00001.

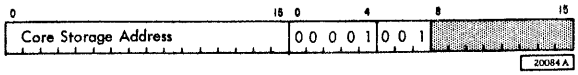
Function	0 1 2 3 4 5 6 7 8 15
Carrier Return	1 0 0 0 0 0 0 1
Tabulate	0 1 0 0 0 0 0 1
Space	0 0 1 0 0 0 0 1
Backspace	0 0 0 1 0 0 0 1
Shift to Red*	0 0 0 0 1 0 0 1
Shift to Black*	0 0 0 0 0 1 0 1
Line Feed	0 0 0 0 0 0 1 1

* May be done concurrently with any other function.

20090A

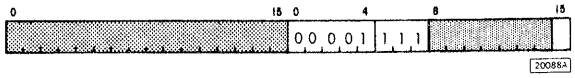
Figure 15. Console Printer Control Functions

Write (001)



This command causes bits 0-7 of the word at the core storage location specified by the address to be sent to the printer for printing or control.

Sense Device (111)



This command causes the keyboard/console printer device status word to be placed in the ACC. Figure 16 shows only those bits which relate to the console printer. Modifier bit 15 on specifies that all keyboard/console printer responses are to be reset.

Interrupts

There is only one interrupt (ILSW-4, bit 1) associated with the console printer attachment.

Service Response: This interrupt occurs each time the console printer has completed printing the data and/or the control operation required by the last word transmitted by the Write command.

Indicators

The following indicators are entered into the CPU by a Sense Device command.

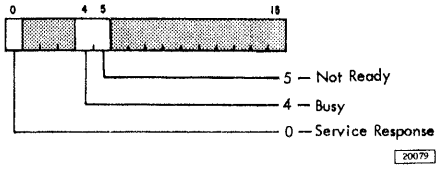


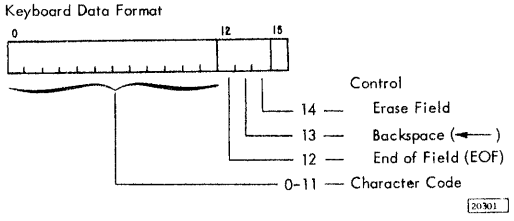
Figure 16. Console Printer Device Status Word

Not Ready: When off, this indicates that the printer is properly loaded with forms, has dc power, and is not busy. It is necessary that the program always determine that the not ready indicator is off before a Write command is given. If a Write command is given while not ready is on, loss of information will probably occur. No indication is given of this loss.

Busy: When on, this indicates that the console printer is in the process of typing a character or executing a control and therefore should not be given a Write command. The busyline is active from the time data is sent to the printer until the printer has completed the action required.

KEYBOARD FUNCTIONAL DESCRIPTION

The input speed of the keyboard (Figure 17) is limited only by the speed of the operator. Keyboard entries are not automatically printed unless the CPU is programmed to provide an output of the entry on the printer. The keyboard emits a coded character for each key struck by the operator. These characters are related to IBM card coding. (See Appendix B. Character Code Chart). Striking the A character key places bits in positions 0 and 3 of the CPU word; striking the I character key places bits in positions 0 and 11 of the word; striking a 9 character places a bit in position 11 of the word; etc.



The two-position Console/Keyboard switch indicates to the program the desired source of the console input data, either the keyboard or the console entry switches.

Keyboard Function Keys

Interrupt Request: This key initiates a keyboard restore and causes an interrupt in the CPU.

End of Field (EOF): When the CPU reads in response to this key, a word containing a 12 bit only is placed in memory. Analysis of this word allows the program to determine that no further characters are to be sent in this message.

Backspace (←): When the CPU reads in response to this key, a word containing a 13 bit only is placed in memory. Analysis of this word allows the program to determine that the last character received is to be replaced by the next character to be entered.

Erase Field: When the CPU reads in response to this key, a word containing a 14 bit only is placed in memory. Analysis of this word allows the program to determine that the message being entered is to be deleted and replaced by a corrected message.

Mode: There are two mode keys: Numeric (upper case shift) and Alphabetic (lower case shift). These keys place the keyboard in the indicated mode. The keyboard remains in the selected mode until changed. (If the keyboard does not have an alpha key in the lower right hand corner, the numeric key must be held down continously while entering numeric data.)

Restore: This key allows the operator to restore the keys if they should become locked.

Keyboard Light

Keyboard Select: This light comes on when the CPU has performed a Control command. This light goes off when a Read command is performed.

Operating Procedure

The following procedure describes a typical use of the keyboard (manual start).

1. The operator presses the Interrupt Request key, which initiates a request interrupt and places the keyboard in a restore status.
2. The CPU honors the request interrupt and determines that the keyboard is the device that caused the interrupt.
3. The CPU issues a control command to select the keyboard. When keyboard is selected, the Select light is turned on to signal the operator that a character can be entered.
4. When a character key is pressed, the keyboard initiates a service interrupt to the CPU.
5. In response to the service interrupt, the CPU performs a Read command, which enters the character into core storage and removes the keyboard from the selected status.
6. Before another character can be entered, the CPU stored program must issue another control command to select the keyboard.

NOTE: When the request is initiated by the stored program, the operation is the same, beginning at Step 3.

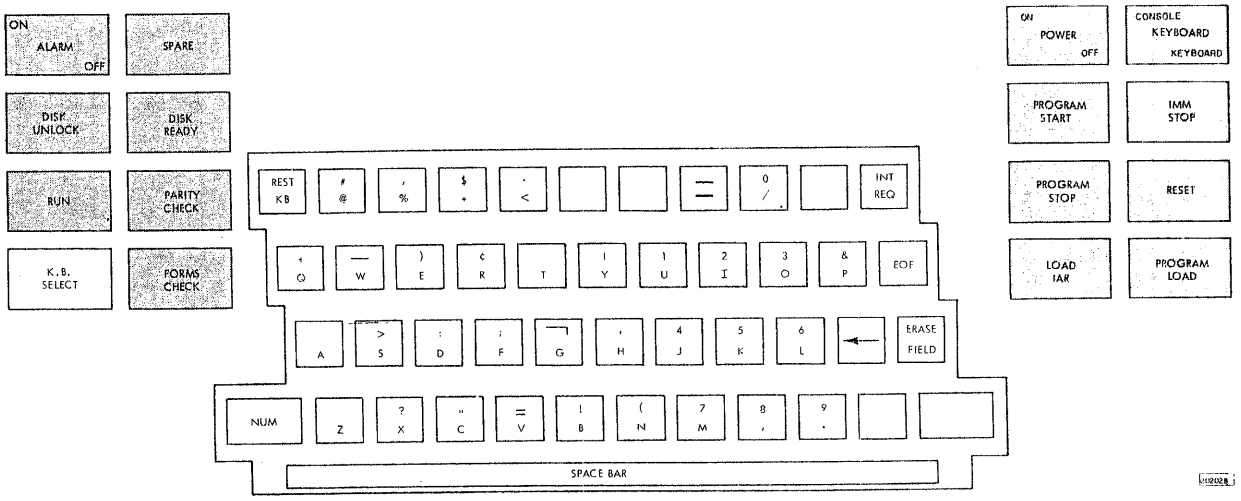


Figure 17. 1131 Console Keyboard

If the CPU performs a Read command when the keyboard is not selected, no bits are entered.

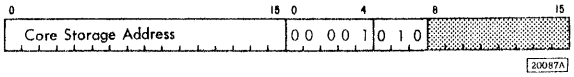
KEYBOARD PROGRAMMING

The keyboard operates under direct program control of the 1130 Computing System.

I/O Control Commands (IOCC)

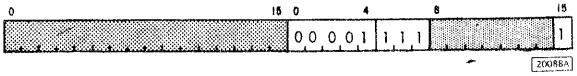
The keyboard is addressed by the same device code used by the console printer, 00001.

Read (010)



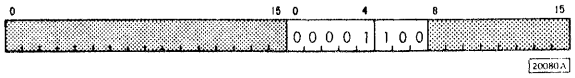
This command enters a single input character from the keyboard into the core storage location specified by the address of the IOCC.

Sense Device (111)



This command reads the keyboard/console printer device status word into the ACC. Figure 18 shows only those bits associated with the keyboard. Modifier bit 15 on specifies that all keyboard/console printer responses are to be reset.

Control (100)



This command places the keyboard in a select status so that a character can be entered.

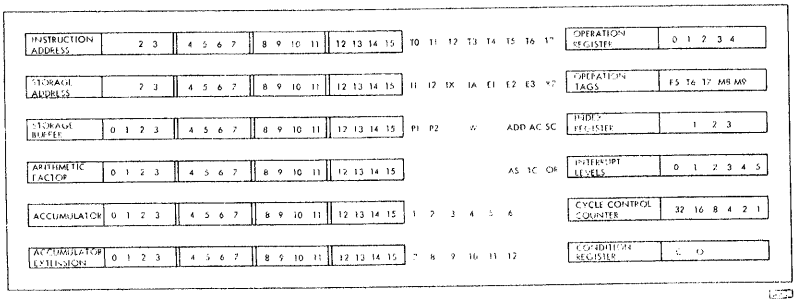


Figure 19. Console Panel

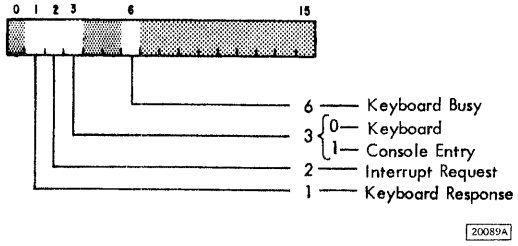


Figure 18. Keyboard Device Status Word

Interrupts

The two interrupts associated with the keyboard are assigned to the same level of priority.

Interrupt Request: This interrupt is initiated by the Request key located on the keyboard.

Keyboard Response: This interrupt signals that a character key has been pressed and that a character is ready to be entered into core storage.

CONSOLE DISPLAY PANEL

The contents of the registers within the computer are displayed on the console panel (Figure 19) by means of small incandescent lights. Each bit in each register position is represented by a light. The light is on when the bit which it represents is present in the word displayed. The Mode switch selects the operating mode of the system.

Indicator Displays

Instruction Address Register (IAR): The instruction address register is one row of 14 indicator lamps. Each lamp displays the status of one bit position in the IAR.

Storage Address Register (SAR): The storage address register is one row of 14 indicator lamps. Each lamp displays the status of one bit position in the SAR.

Storage Buffer Register (SBR): One row of 16 indicator lamps. Each lamp indicates the status of one bit position in the SBR. At the end of each machine cycle the SBR reflects the bits that were read into core storage on the cycle just completed.

Arithmetic Factor Register (AFR): One row of 16 indicator lamps. Each lamp indicates the status of one bit position in the AFR.

Accumulator Register (ACC): One row of 16 indicator lamps. Each lamp indicates the status of one bit position in the ACC.

Accumulator Extension Register (EXT): One row of 16 indicator lamps. Each lamp indicates the status of one bit position in the EXT.

Clock Timer (T): One row of eight indicator lamps. These lamps indicate the last clock step completed.

Machine Cycle: One row of seven indicator lamps. These lamps indicate the type of machine cycle in process when in single step mode. They indicate the machine cycle just completed when in any other mode.

Control Functions: Two rows of three indicator lamps. These lamps indicate the status of the following functions: add, arithmetic control, shift control, accumulator sign, accumulator carry, and zero remainder.

CE Lights: Two rows of six indicator lamps. Each lamp can be wired by a CE to give a visual indication of any status condition in the machine.

Operation (OP) Register: One row of five indicator lamps. These lamps indicate the operation in process when in single step mode or single machine cycle mode. They indicate the operation just completed when in any other mode.

Operation Tags: One row of five indicator lamps. These lamps indicate the status of the format, tag, and modifier bits of the instruction shown in the operation register.

Interrupt in Process: One row of six indicator lamps. These lamps indicate the interrupt level being serviced.

Cycle Control Counter: One row of six indicator lamps. These lamps indicate the binary value contained in the shift counter.

Condition Register: One row of two indicator lamps. These lamps indicate the status of the carry indicator and the overflow indicator.

Parity (2): These two indicators reveal which half of the word contains the parity error when one occurs.

Wait Op: This indicator is on when the CPU is in a wait condition.

X7: This indicator is off when "cycle steal" is operating.

Mode Switch

The Mode switch (Figure 20) selects one of seven operating modes.

Single Step (SS): With the Mode switch set to SS, each depression and release of the Start key causes the 1131 clock to advance one step, e.g., from T1 to T2.

Single Memory Cycle (SMC): With the mode switch set to SMC, each depression of the Start Key causes the 1131 to advance one machine cycle (for example from I-1 to I-2).

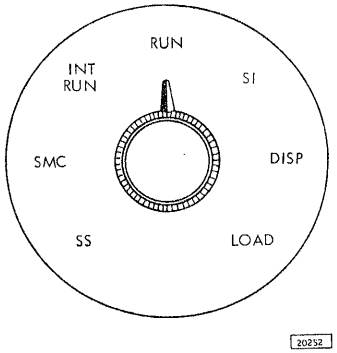


Figure 20. Console Mode Switch

Interrupt Run (INT RUN): With the Mode switch set to INT RUN, a level 5 interrupt occurs after each mainline program instruction is completed. This is a convenient device for program trace routines.

Program Run (RUN): With the mode switch set to its normal position, RUN, pressing the Start Key causes the 1131 to advance through its stored program.

Single Instruction (SI): With the mode switch set to SI, each depression of the Start key causes the 1131 to interpret and execute a single instruction.

Display Core Storage (DISP): With the mode switch set to DISP, pressing the Start key will display (in the SBR) the core storage word at the location specified by the address in the instruction address register (IAR), and advances the IAR.

Load Core Storage (LOAD): With the mode switch set to LOAD, pressing the Start key will load the data from the Console Entry switches into core storage at the location specified by the address in the IAR, and advances the IAR.

CONSOLE ENTRY SWITCHES

These 16 toggle switches are used to set up data or instructions to be entered into core storage. Each switch represents a bit position in a 16-bit word. The procedures that follow provide for entering the information from the Console Entry switches (CES) by means of manual control, keyboard interrupt, or XIO instruction.

Manual Entry: This procedure causes the bits set in the CES to be loaded into the word at the core storage address in the IAR.

1. Set the Mode switch to LOAD.
2. Set the CES to the core storage address where the data is to be stored.
3. Press the Load IAR switch.
4. Set the data word in the CES.
5. Press the Start key.

Keyboard Interrupt: This procedure requires an interrupt subroutine to service a level 4 interrupt.

1. Set the Console/Keyboard switch to CONSOLE.
2. Press the keyboard Interrupt Request key.
3. A level 4 interrupt occurs, and the keyboard DSW is loaded into the ACC by a Sense Device instruction.
4. DSW bit 3 was set to 1 by the Console/Keyboard switch. This indicates to the interrupt subroutine that the CES should be read by the XIO Read instruction.
5. Return to the main line program is by the regular method of a BSC instruction with modifier bit 9 set to one.

XIO Read Instruction: The settings of the CES can be read by an XIO Read instruction at any time during a stored program routine. The device code of the instruction is set to 00111.

CONSOLE FUNCTION SWITCHES AND LIGHTS

These switches and lights are located on both sides of the keyboard (Figure 21).

Function Lights

Forms Check: This indicator is turned on when the last form has been detected by the console printer forms contact.

Keyboard Select: This indicator is turned on by a programmed instruction (XIO Control) that requests input data from the keyboard.

Parity Check: This indicator is turned on when a parity error (even number of bits) is detected in either half of the word read out of storage.

Alphabetic: This lamp indicates that the keyboard is in alphabetic (lower case) shift. The letters and symbols which appear in the bottom portion of the keyboard character keys can be entered when the keyboard is in alphabetic shift. (Not present on some systems.)

Numeric: This lamp indicates that the keyboard is in numeric (upper case) shift. The letters and symbols which appear on the top portion of the keyboard character keys can be entered only when the keyboard is in numeric shift. (Not present on some systems.)

Disk Ready (File Ready): This indicator is on when disk storage is available for reading or writing.

Disk Unlock: This indicator is on when the disk cartridge may be removed from the drive.

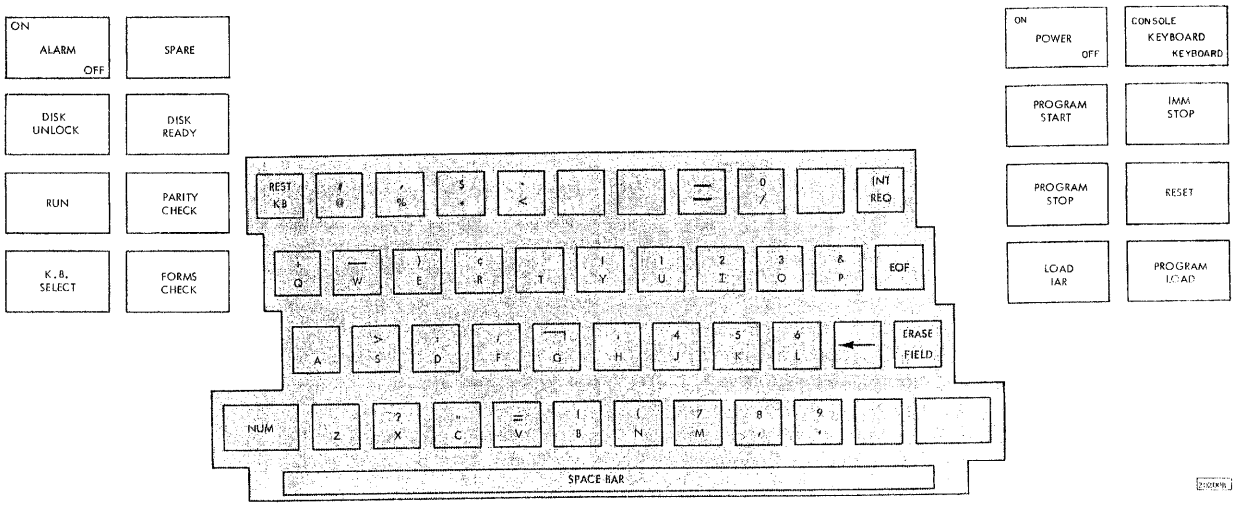


Figure 21. Console Function Lights and Switches

Run: This indicator is on when the CPU is in operation and the meter is running.

Function Switches

Console/Keyboard: This two-position toggle switch sets bit position 3 in the keyboard DSW, which indicates to the program the desired source of the console input data (either the keyboard or the console entry switches). See the preceding Console Entry switches and the DSW for the Sense Device (111) command (Figure 18).

Program Start: Pressing this pushbutton switch causes the machine to take one clock step or machine cycle and continue to take additional cycles if required by the setting of the mode switch.

IMM Stop: Pressing this pushbutton switch causes an immediate stop of the processor interrupt, although the I/O devices will finish their present cycle. Data from the cycle steal devices will be lost if they are operating at the time the IMM Stop key is pressed. A complete program restart is normally required.

Program Stop: Pressing this switch causes a level 5 interrupt. After the program has satisfied all interrupts for levels 0 through 4 (all I/O devices except the console) it enters a routine for level 5. Because the console is the only unit on level 5 (unless SAC device on level 5), the DSW for the console can be sensed without interrogating the interrupt level status word. The program stop bit in the DSW should be used to branch the program to a wait loop that causes the CPU and I/O units to cycle down to a stop and blocks all mainline operations until the console operator intervenes. This indicator is reset by pushing the program start key.

Reset: Pressing this pushbutton switch (effective only when out of run mode or stopped) resets all I/O and machine registers, cycle and control triggers, and status indicators.

Load IR: Pressing this pushbutton switch places the status of the 16 Console Entry switches in the IAR. The console Mode switch must be set to the Load position.

Program Load: Pressing this pushbutton switch loads the first card or paper tape record into core storage, beginning at 00000.

IBM 1442 CARD READ PUNCH

The IBM 1442 Card Read Punch (Figure 22), Model 6 or Model 7, provides card input/output for the IBM 1130 Computing System.

Card read punch operations are under direct program control

FUNCTIONAL DESCRIPTION

The IBM 1442 Card Read Punch is a single unit that processes cards serially, column by column, from a single supply hopper. All cards first pass the read station, then the punch station. This permits each card to be read, punched, or read and punched. Reading and punching cannot occur simultaneously, however, because of the difference in operating speeds.

Maximum machine speeds are:

Card Reading:	Model 6, 300 cards per minute
	Model 7, 400 cards per minute
Card Punching:	Model 6, 80 columns per second
	Model 7, 160 columns per second

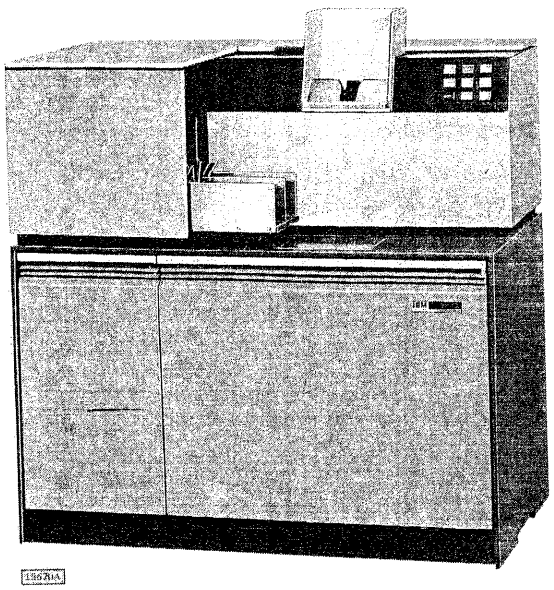


Figure 22. IBM 1442 Card Punch

Maximum reading rates are attained only when successive Start-Read commands arrive early enough to re-energize the read clutch before the clutch latch point is reached. To accomplish this, successive Start-Read commands must arrive within 35 milliseconds (25 ms Model 7) after the operation complete interrupt is given by the card read punch. If a Start-Read command does not arrive within this time, the maximum reading rate becomes 285 cards per minute (cpm) for Model 6 and 375 cpm for Model 7.

Punching rates depend on the position of the card when the last column is punched. The punching speed ranges are:

Model 6 – 49 cpm to 262 cpm
Model 7 – 91 cpm to 355 cpm

The approximate time required to process a single card is:

Model 6 – 216 ms + 12.5 ms per card column spaced or punched.
Model 7 – 163 ms + 6.25 ms per card column spaced or punched.

Data Coding

The card read punch reads and punches IBM card image only. Any code translation required must be done by the stored program. As shown in Figure 23 the twelve rows (12-9) in a card column correspond to the 0-11 bits, respectively of a core storage word. A 1 bit represents a punched hole; a 0 bit represents a card position not punched. Thus, the word in Figure 23 contains 1 bits in bit positions 0 and 3 to represent the "A" read from the card. For output, a 1 bit results in a hole punched in the related position of the card read column.

A special load mode is initiated by pressing the Program Load key on the 1130 console. In the load mode, data is split (Figure 24), as it enters core storage, to form the load program.

OPERATING PROCEDURES

Before any operation can begin, the card read punch must be placed in the ready condition. With power on and cards in the hopper, the Start key is pressed. This feeds the first card into position at the read station (Figure 25).

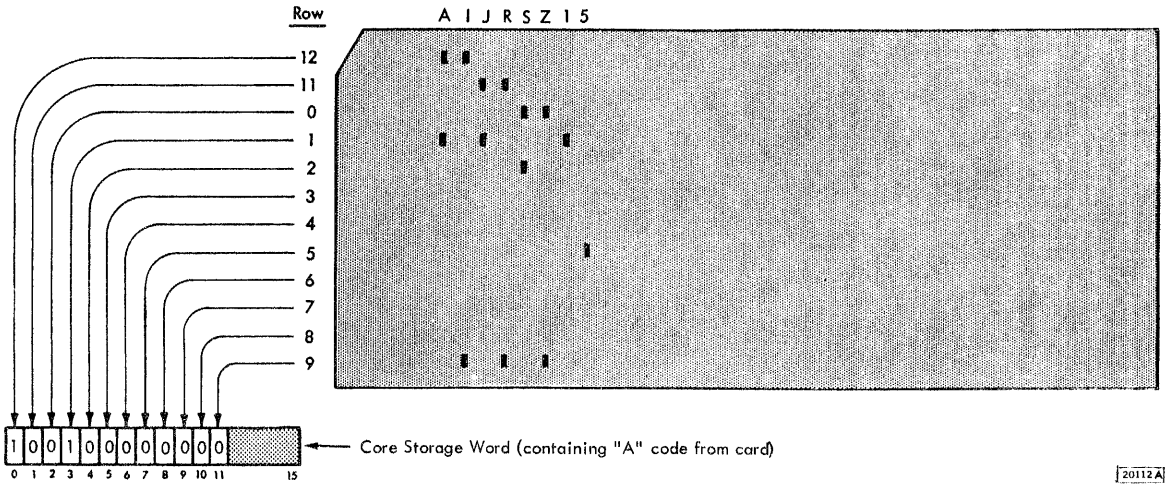


Figure 23. Normal Mode Read

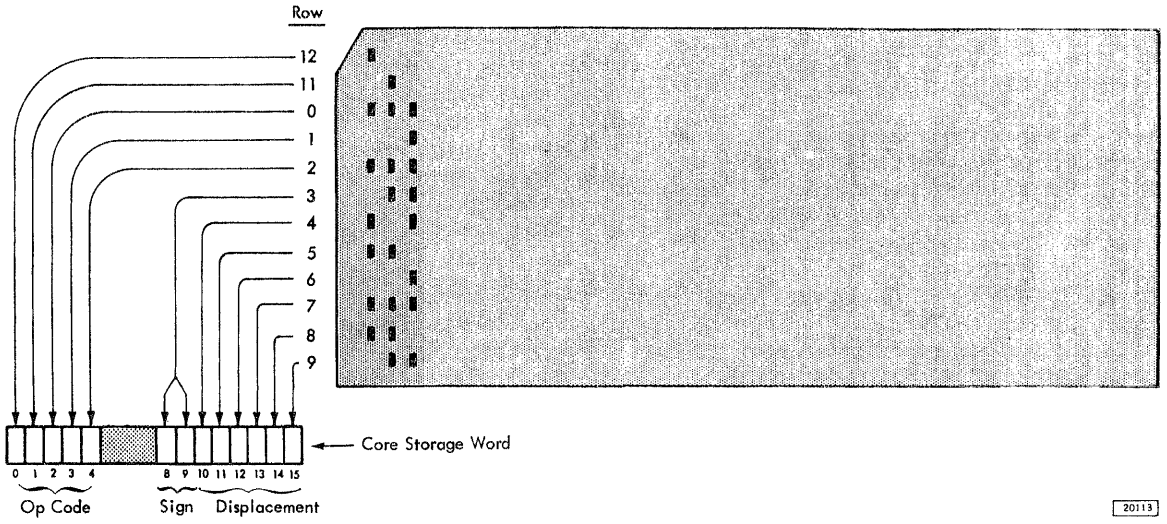


Figure 24. Load Mode Read

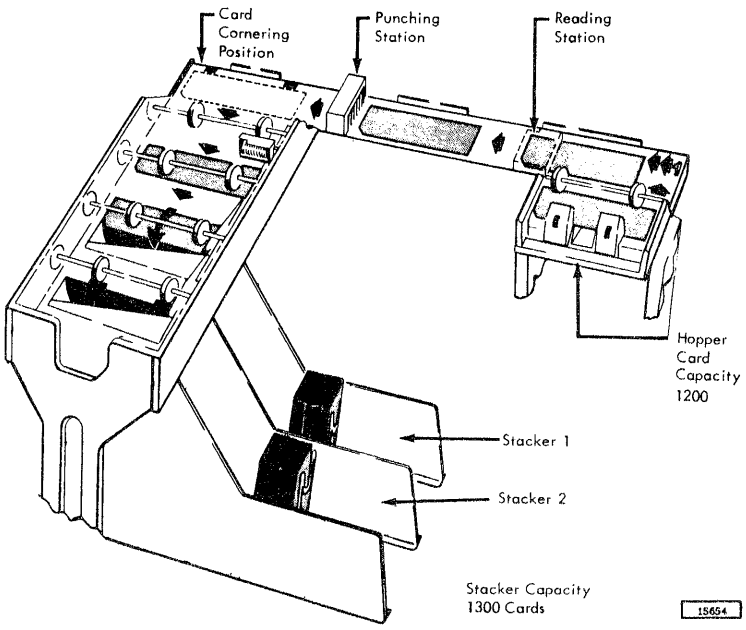


Figure 25. 1442 Card Path Schematic

Card Feeding

Card reading or punching may begin after the initial feed cycle (run in).

A constant-speed drive moves the cards through the serial path during a feed cycle. A feed cycle is initiated by a control command of Feed Cycle or Start-Read or Punch (if no card is positioned at the punch station). The feed cycle does three things.

1. It moves a card from the punch station to the stacker.
2. It moves a card through the read station and places it in the punch station with column 1 under the punches.
3. It moves a card from the hopper to the read station.

An incremental drive moves the card through the punch station for punching.

When the hopper is emptied, the operator can either reload the hopper and continue operations or he can initiate a last-card sequence.

Program Load

Program load can be initiated by pressing the Program Load key on the 1130 console after a system reset and the "run in" cycle of a load card. This load mode causes the load-card data to be placed in 80 consecutive memory positions beginning at memory position 00000, then causes the CPU to go to memory position 00000 for its next instruction.

Card Reading

A control (Start-Read) command initiates card reading. This command causes columns 1-80 of the card to be read in one continuous motion of the card. Each column of data is read, checked, and placed in a buffer register. A read response interrupt is given for each column read. Checking is accomplished automatically by reading each column twice and comparing the results bit by bit. This read-check-interrupt process continues until all 80 columns have been read. An operation complete interrupt is given after all 80 columns have been read. The Last Card indicator will be on if the card read is the last card in the deck.

Card Punching

A control (Start-Punch) command initiates card punching. As each column passes the punch station a punch response interrupt is given.

Automatic checking is accomplished by comparing the punch check echo data with the single-character punch buffer, which contains the character from the CPU. Each column punched is checked at the same time that the punch response interrupt is given for the data of the next column to be punched.

The card motion and punching process continues until the punch data word contains a one in the 12-bit position (punch data is in bits 0-11). When this end-punch bit is detected, the card read punch punches that column, moves to the next column, and gives an operation complete interrupt. No more punch response interrupts are given. All punching on the card must be completed at one time.

A feed cycle is necessary to eject a punched card to the stacker, and can be initiated by a control command.

Programming Note. A control command specifying Start Punch results in a feed cycle if it has not been preceded by a control command specifying Feed Cycle or Read Card.

Last Card Sequence

When the hopper becomes empty during a feed cycle, the card read punch is taken out of ready status. The operator may continue processing cards by loading more cards into the hopper and pressing the Start key or he may initiate a last-card sequence by pressing the Start key without loading more cards in the hopper.

If the last-card sequence is to be entered, the program determines this by testing the last card indicator in the device status word. This indicator is turned on when the last card passes the read station.

When the Start key is pressed without cards in the hopper, the 1442 is placed in the ready condition and allows two more feed cycles to be taken.

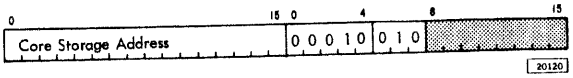
PROGRAMMING

The IBM 1442 Card Read Punch operates under the direct program control of the CPU.

I/O Control Commands

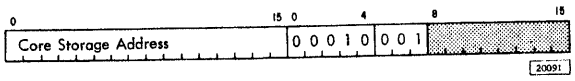
The card read punch is addressed by the 5-bit Device code, 00010.

Read (010)



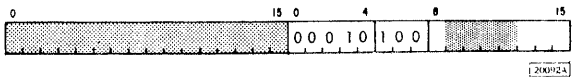
This command causes a card column image to be entered from the card read punch into the core storage location specified by the address.

Write (001)



This command causes the data in the memory location specified by the address of the IOCC to be transmitted and punched as a card column image in the card.

Control (100)



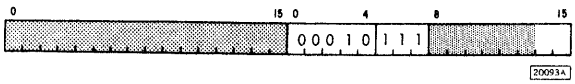
This command causes the Card Read Punch to accomplish the function specified by the Modifier.

Modifier bits that have significance are:

- Bit 14 Feed Cycle – causes all cards in the feed path to advance one station. There are no read column response interrupts.
- Bit 13 Start Read – causes the card to move through the read station. As each column is read and checked, the card read punch initiates a read column response interrupt.
- Bit 15 Start Punch – starts the punching operation and initiates a punch response interrupt. If a card is not at the punch station, a card will feed past the read station without data entering the system.
- Bit 8 Stacker Select – causes the card leaving the punch area to enter the alternate stacker. This control applies only to the next card leaving the punch station.

Modifier bits B14, B13, and B15 of a control command should not be used in combination with each other.

Sense Device (111)



This command directs the card read punch to place its device status word (Figure 26) into the CPU ACC. Modifier bit 15 on resets responses for level 0; modifier bit 14 on resets responses for level 4.

Interrupts

The three interrupts associated with the card read punch are divided into two groups.

Level 0 Interrupt

Read Response: This interrupt signals that a column of data is ready to be entered into main storage. This interrupt request must be serviced within 800 μsec for the 1442 Model 6 and 700 μsec for the 1442 Model 7. Time from Start Read to first read column request interrupt is 28.4 msec for the Model 6 and 23.8 msec for the Model 7.

Punch Response: This interrupt signals that a column of data must be transmitted from the CPU within 1000 μsec for the 1442 Model 6 and 300 μsec for the Model 7. Time from the Start Punch instruction to the first punch column response interrupt varies from 1.22 ms to 12.5 ms on the Model 6 and 1.56 to 6.25 ms on the Model 7.

Level 4 Interrupt

Operation Complete: This interrupt occurs after a card has been read. It indicates that column 80 of the card has passed the read station. This interrupt occurs 20.6 ms after column 80 for the Model 6 and 15.4 ms after column 80 for the Model 7.

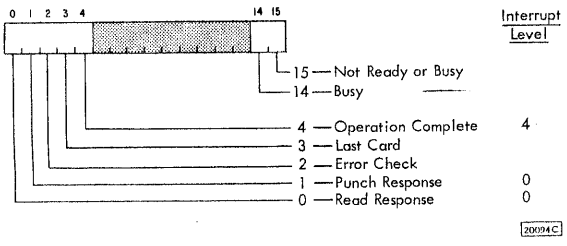


Figure 26. 1442 Device Status Word

This interrupt also occurs after the last column to be punched has been punched and checked with the punch drive stopped. This will occur 12.5 ms after the terminating Write function for the Model 6 and 6.25 ms after the terminating Write function for the Model 7.

This interrupt is forced if a hopper check, feed check in the punch station, transport error, or feed clutch error occurs while the 1442 is busy. This interrupt is also forced if a read registration check or punch check occurs. No subsequent reading or punching can be done in the card which caused the error regardless of the column in which the error occurred.

There is no time limit on the request for service of this interrupt.

Indicators

Not Ready: This indicator shows that the 1442 is either busy or is not in a ready condition. When the 1442 is not ready, manual intervention is required to ensure that the following conditions are met.

1. Power on.
2. Card registered at read station (initially).
3. Cards are in hopper or last-card sequence is in progress.
4. Stacker not full.
5. Feed-Check light off (no card jam or feed failure conditions).
6. If the stop key has been pressed, the Start key must have been subsequently pressed.
7. Chip box not full or removed.

Busy: The Busy indicator indicates that a command cannot be initiated because an operation is already in progress.

Last Card: This indicator shows that column 80 of the last card has passed the read station and the hopper is empty, and will be on when the operation complete interrupt occurs.

Error Check: Indicates that any of seven error conditions exists in the 1442. Any of the seven error conditions remove the 1442 from the ready condition; the 1442 can be reset only by depressing the Non-Process Runout key while the hopper is empty.

The error conditions are:

1. Read Registration Check. Indicates that a read error has occurred. This can result from incorrect registration of the card or

failure of the first and second reading of the column to compare equal. When this error is detected, an operation complete interrupt is forced and column interrupts are terminated.

2. Punch Check. Indicates that a punching error has been detected. When this error is detected, an operation complete interrupt is forced and column interrupts are terminated.
3. Hopper. Indicates card failed to pass properly from the hopper to the read station. See Programming Note below.
4. Transport. Indicates a jam in the stacker. See Programming Note below.
5. Feed Check – Read Station. Indicates a card failed to eject from the read station.
6. Feed Check – Punch Station. Indicates a card improperly positioned at the punch station. See Programming Note below.
7. Feed Clutch. Indicates the 1442 took a feed cycle that was not called for. See Programming Note below.

Programming Note: Error indicator is not turned on until after the operation complete interrupt is given. An exception to this is an XIO start punch operation requiring an automatic feed cycle. If another operation is initiated before the error indicator is turned on, these errors force an operation complete interrupt although no reading or writing has taken place. An XIO start punch requiring an automatic feed cycle is treated as two operations: (1) feed cycle (2) punch operation.

IBM 1442 CARD PUNCH

The IBM 1442 Card Punch Model 5 provides card output for the IBM 1130 Computing System. The 1442 and the 2501 Card Reader provide a separate card path for punched card input and output.

FUNCTIONAL DESCRIPTION

The IBM 1442 Card Punch is a single unit that punches cards serially, column by column. The punching speed is 160 columns per second. The characteristics of the 1442 Model 5 are the same as the punching characteristics of the Model 7. (See IBM 1442 Card Read Punch.) The 1442 Model 5 is attached in the place of the Model 6 or Model 7, thus precluding the attachment of two 1442s to an 1130 System.

IBM 2501 CARD READER

The IBM 2501 Card Reader (Figure 27), Model A1 and Model A2, provides card input for the IBM 1130 Computing System. Card reading is under direct program control.

FUNCTIONAL DESCRIPTION

The IBM 2501 Model A1 reads cards at a maximum rate of 600 cards per minute (cpm); the Model A2 reads at a maximum rate of 1000 cpm.

Cards are read serially, that is, column by column beginning with column one. Each column is read twice and the two readings are compared to check reading accuracy. Thus, off-punched and mispositioned cards are detected.

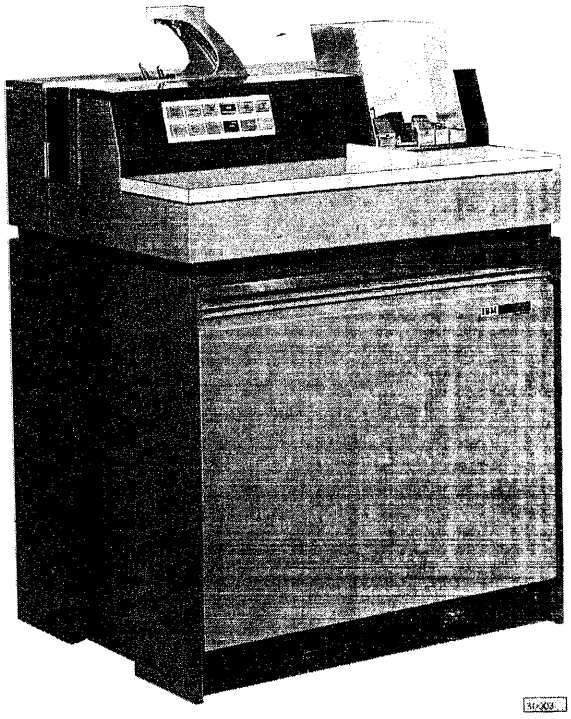


Figure 27. IBM 2501 Card Reader

Data Coding

The 2501 reads punched cards in card image only. Any code translation required must be done by the stored program in the CPU. As shown in Figure 23, the twelve rows (12-9) in a card column correspond to the 0-11 bits, respectively, of a core storage word.

A special load mode is initiated by pressing the Program Load key on the 1130 console. In the load mode, data is split as it enters core storage to form the load program. Refer to Figure 24.

Operating Procedures

Before any card reader operation can begin, the 2501 must be placed in the ready condition. With power on and cards in the hopper, press Start key. This feeds the first card into position at the read station and turns on the ready condition.

Card Feeding

After the initial feed cycle (run in) card reading may begin. Card feeding is initiated by an Initiate Read command. This command causes the card to begin moving. If the data is to be ignored (as in card feeding), the word count must be zero.

Card movement is as follows:

1. The card at the read station moves through the read station to the stacker.
2. A card moves from the hopper to the read station.

When the hopper is emptied, the 2501 leaves the ready condition. The operator may reload the hopper and press the start key to continue processing or the operator may press the start key without reloading the hopper to initiate the last card sequence.

Program Load

Program load may be initiated by pressing the program load key on the 1131 console. This causes the load card data to be loaded into the first 80 memory locations. After the card has been loaded the instruction address register is reset to 00000 and the CPU goes to this address for its next instruction.

Card Reading

An Initiate Read (110) command causes the card to be read in one continuous motion. The number of columns actually transferred to core storage depends upon the word count in the first word of the data table.

The data is read into a buffer register where it is checked. Then a cycle steal request is given for each column to be transferred. The checking is accomplished by reading the data a second time and comparing it to the data previously read into the buffer. After the last column (column 80) has been read an operation complete interrupt is requested.

Last Card Sequence

When the hopper becomes empty during a feed cycle, the 2501 is taken out of the ready status. Intervention by the operator is required to continue processing cards. The operator may reload the hopper and press the 2501 Start key to continue processing or the operator may initiate the last card sequence by pressing the 2501 Start key with the hopper empty.

The last card sequence places the 2501 in the ready condition for one more feed cycle and turns on the Last Card indicator. An operation complete interrupt is given at this time. The last card indicator remains on until a Sense Device Status Word command (111) is issued with bit 15 on.

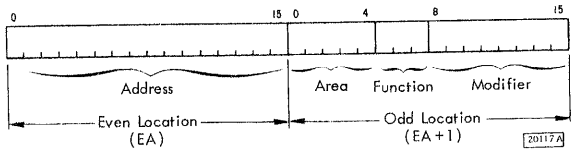
PROGRAMMING

The IBM 2501 Card Reader operates under the control of the stored program in the CPU.

I/O Control Commands

The 2501 is addressed by the five-bit device code 01001.

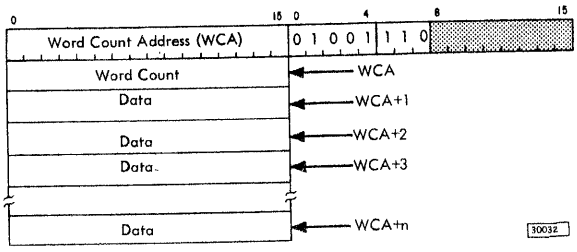
The format of the IOCC follows:



The address word specifies the location in core storage of the data table. The first word of the data table designates the number of words to be read. This word is called the word count. The word count

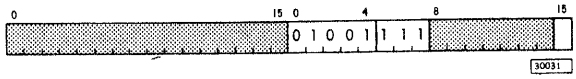
is located in bit positions 9 through 15 and should never exceed 80 (50 hexadecimal). If the word count exceeds 80, information in succeeding core locations is destroyed.

Initiate Read (110)



This code provides the ability to start a read operation, which subsequently makes data transfers to core storage via a data channel by means of cycle stealing.

Sense Device (111)



This command sets the accumulator with the device status word (DSW) of the 2501 (Figure 28). The DSW bits 3 and 4 (Last Card and Operation Complete) are reset if bit 15 is on when this command is executed.

Interrupts

The operation complete is the only interrupt associated with the 2501. This interrupt occurs after column 80 has passed the read station and feed checking has been completed. The operation complete interrupt is independent of the word count and terminates further cycle steal requests. The number of characters actually transferred to the CPU depends upon the word count. The 2501 is assigned to interrupt level status word 4. Bit 3 is turned on if the 2501 caused the interrupt. The Sense Interrupt (011) command causes ILSW 4 to be loaded into the accumulator if level 4 is being serviced.

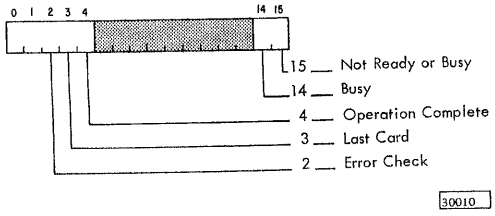


Figure 28. 2501 Device Status Word

Indicators

Not Ready or Busy: This indicator indicates that the 2501 is not in a ready condition, or that it has received an instruction and is the process of executing it.

Busy: This indicator indicates that a card read is in progress and therefore another read card cannot be initiated. This indicator turns off when the operation complete interrupt occurs.

Operation Complete: This indicates that a card read operation has been completed.

Last Card: This indicates that the last card fed from the hopper, and the operator initiated a last-card sequence. The indicator may be turned off by a Sense Device command, with reset (bit 15) on.

Error Check: This indicates a feed check or a read check.

Reader and System Timing

There are two basic timing considerations of importance to the user of a 2501 Card Reader attached to an IBM 1130 Computing System:

1. Card throughput in cards per minute (cpm).
2. Time available for other system operation.

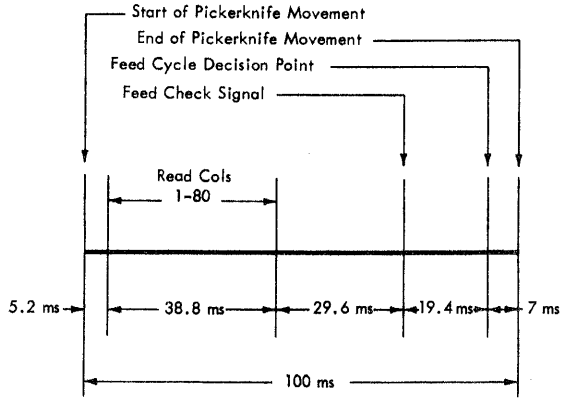
System Operations

The 1131 is capable of performing operations (such as reading, processing, and punching) simultaneously. After an operation is initiated, the CPU is busy for only 288 microseconds. The remainder of the card read cycle is available for other use.

Card Throughput

The 2501 Model A1 has a 100-ms card feed cycle; the Model A2 has a 60-ms card feed cycle. To maintain the rated speed, an Initiate Read command must occur every 100 ms for the Model A1 and every 60 ms for the Model A2. A basic timing consideration of importance to the user of the 2501 Card Reader to an IBM 1130 System is the time between the Feed Check

Model A1 - 600 CPM (All times shown are nominal at rated thruput.)



Model A2 - 1,000 CPM (All times shown are nominal at rated thruput.)

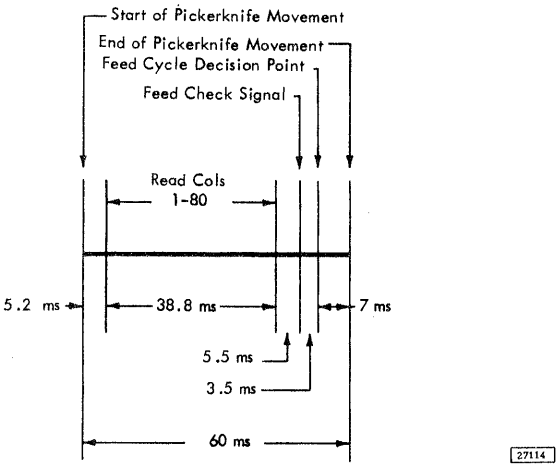


Figure 29. 2501 Timing Schematic

Signal and the Feed Cycle Decision point. This timing is shown in Figure 29. In order to maintain rated throughput, the read instruction must be received within 18.3 ms (A1) and 3.0 ms (A2) following an interrupt (Feed Check Signal).

If an Initiate Read command misses the feed cycle decision point, the card reader waits until the feed cycle decision point of the next cycle before starting to execute the command. The result in this case is the throughput is about one-half of the maximum.

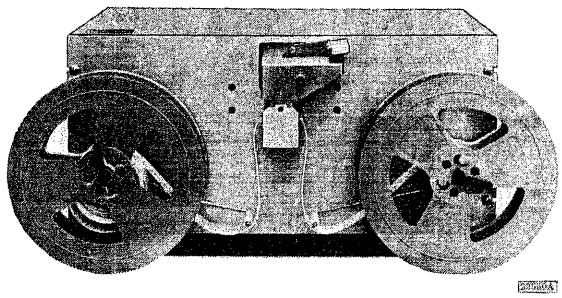


Figure 30. IBM 1134 Paper Tape Reader

IBM 1134 PAPER TAPE READER AND IBM 1055 PAPER TAPE PUNCH

The IBM 1134 Paper Tape Reader and the IBM 1055 Paper Tape Punch (Figures 30 and 31) provide paper tape input/output for the IBM 1130 Computing System.

FUNCTIONAL DESCRIPTION

The 1134 and 1055 operate under direct program control.

The 1134 Paper Tape Reader reads one-inch eight-channel paper tape at a maximum rate of 60 columns per second (cps).

The 1055 Paper Tape Punch punches one-inch eight-channel paper tape at a maximum punching rate of 14.8 cps.

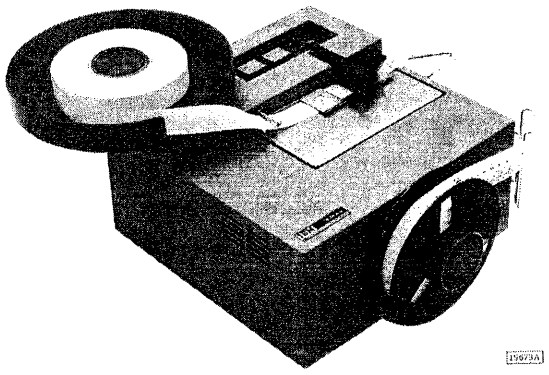


Figure 31. IBM 1055 Paper Tape Punch

Character Code

The 1134 Paper Tape Reader reads input data into the core storage as an image of the holes in the tape. One paper tape character is read into each addressed core storage location. Any code translation must be made by programming. (See Appendix B.)

Figure 32 indicates which bits of the word correspond to the respective holes in the paper tape read by the 1134.

The 1055 Paper Tape Punch punches data as an image of the data contained in the core storage word on a character-to-character basis as shown in Figure 32.

Special data-character and control-character (feed code, etc.) coding and recognition must be handled by the stored program.

Program Load

An 1130 system that does not have Card I/O will have the Program Load feature added to the paper tape reader. This feature operates by means of design logic rather than program control. Four-bit paper tape characters are automatically assembled into four-character groups to form 16-bit data words. The Program Load feature then loads these words into core storage beginning at location 00000.

Only tape channels 4, 3, 2, and 1 are used. When a channel 5 punch is encountered in other than a delete character, program loading stops; the IAR is reset to zero; and program control begins at 00000.

Characters →	First	Second	Third	Fourth
Tape Channels →	4 3 2 1	4 3 2 1	4 3 2 1	4 3 2 1
Bits →	0 1 2 3	4 5 6 7	8 9 10 11	12 13 14 15

DESCRIPTION OF OPERATION

Paper Tape Reader

A control command initiates the reading of data from the 1134. This command moves the paper tape to the character position and loads the character which was at the read station into an input buffer. At the time the buffer has been loaded with data, an interrupt is initiated signaling the program that information is available for reading into the core storage position specified by the address word of a subsequent Read (Paper Tape) command.

The elapsed time from the execution of the control (Read Paper Tape) command until the interrupt is initiated is approximately 500 μs. To maintain the 60 cps operating speed of the 1134, the Read command must be given within 15 ms after the interrupt so that another control (Read Paper Tape) command can be executed to energize the reader clutch preparatory to reading the next character.

Paper Tape Punch

Execution of a command initiates the punching of data onto the 1055. The execution of the Write command

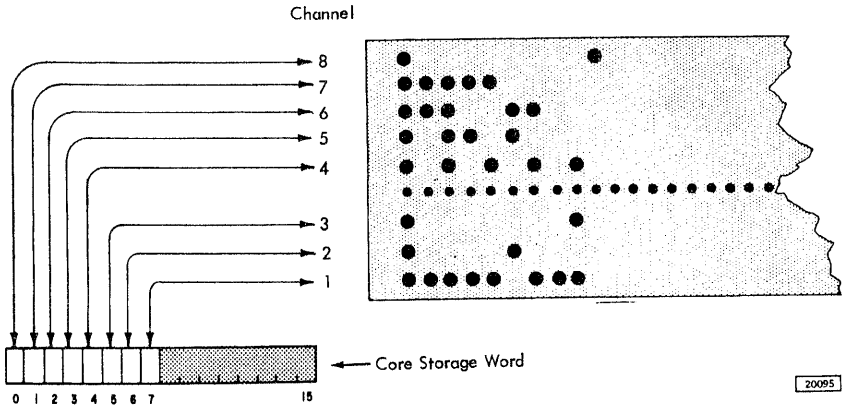


Figure 32. Paper Tape/Core Storage Format

starts the punch, and the data in the core storage position specified by the address word is punched into the tape. Each core storage word contains one paper tape character to be punched in the tape.

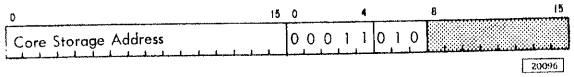
PROGRAMMING

The IBM 1134 Paper Tape Reader and the IBM 1055 Paper Tape Punch operate under direct program control with the exception of the paper tape Program Load feature.

I/O Control Commands (IOCC)

The 1134 and 1055 are addressed by the same five-bit device code, 00011.

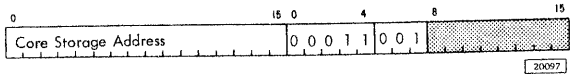
Read (010)



This command reads one character from paper tape into core storage.

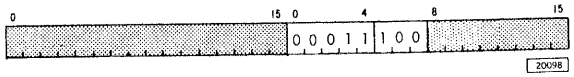
The address word specifies the location in core storage where the tape character is to be stored.

Write (001)



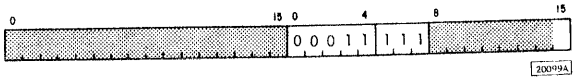
This command writes one character from core storage to the paper tape punch. The address word specifies the location in core storage where the tape character is stored.

Control (100)



This command must be given prior to each character to be read from the 1134. Execution of this command causes: (1) one character to enter the paper tape reader buffer, and (2) the tape to be advanced one column. A reader service response interrupt is initiated to indicate that a character from paper tape can be read into the core storage location specified by a subsequent Read (Paper Tape) command.

Sense Device (111)



This command is used to enter the device status word (Figure 33) into the ACC. Modifier bit 15 on, indicates that the responses are to be reset.

Interrupts

Reader Response: This interrupt occurs when the reader has completed the execution of a control command. This interrupt indicates to the CPU that a character is available to be entered into core storage.

Punch Response: This interrupt occurs when the punch has completed punching as directed by the execution of a Write command. It indicates that the punch can accept the next command.

Indicators

The following indicators can be entered into the ACC by a Sense Device command.

Punch Not Ready: This indicator is on when tape is not feeding freely from the tape spool, when the tape pressure roll holder is not down and holding the tape against the feed wheel, or when tape is not present. Manual intervention is required to clear these conditions. The indicator is also on if the punch is busy. (See Punch Busy indicator.)

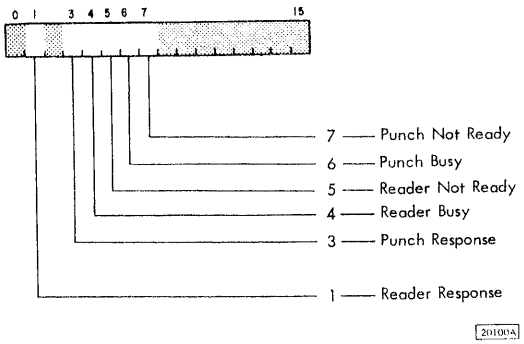


Figure 33. Paper Tape Device Status Word

This indicator should always be tested by the program before a Write command is given. If a Write command is given while this indicator is on, loss of information will probably occur. No indication is given of this loss.

Reader Not Ready: This indicator is on when the tape tension switch is open. This condition exists when the paper tape is broken or not feeding freely. Manual intervention is required to clear these conditions. This indicator is also on if the reader is busy. (See Reader Busy indicator.)

The program should test this indicator before a Read command is given. If a Read command is given while this indicator is on, erroneous data can be read into core storage. No valid indication can be given as to whether the data read is correct or incorrect.

Punch Busy: This indicator is on for the total time the punch is mechanically engaged and punching a character (68 ms). During this time the punch should not be sent another Write command.

Reader Busy: This indicator is on from the time a Control command (Start Paper Tape Reader) is given until data is available. A reader response interrupt signals that data is available.

IBM 1627 PLOTTER

The IBM 1627 Plotter (Figure 34) provides an exceptionally versatile, reliable, and easy-to-operate plotting system for the IBM 1130 Computing System. The plotter converts tabulated digital information into graphic form. Bar charts, flow charts, organization charts, engineering drawings, and maps are among the many graphic forms of data which can be plotted on the 1627 Plotter.

Two models of the 1627 are available and the major characteristics are as follows.

Model 1 – Plotting area: 11 inches by 120 feet, 1/100-inch incremental-step size, 18,000 steps/minute.

Model 2 – Plotting area: 29-1/2 inches by 120 feet, 1/100-inch incremental-step size, 12,000 steps/minute.

See Figure 35 for more information.

OPERATION

Data from core storage is transferred serially to the 1627 under direct program control, where it is translated into 1627 actuating signals. These signals are then converted into drawing movements by the 1627 Plotter.

The actual recording is produced by incremental movement of the pen on the paper surface (y-axis) and/or the paper under the pen (x-axis). The pen is mounted in a carriage that travels horizontally across the paper, as viewed from the front of the plotter. The vertical plotting motion is achieved by rotation of the pin feed drum, which also acts as a platen (Figure 36).

The drum and the pen carriage are bidirectional; that is, the paper moves up or down, and the pen moves right or left. Control is also provided to raise or lower the pen from or to the paper surface. The pen remains in the raised or lowered position until directed to change to the opposite status.

The drum and pen-carriage movements and the pen status are controlled by digits transferred to the 1627. Each output word is decoded into a directional signal which causes a 1/100-inch incremental move-

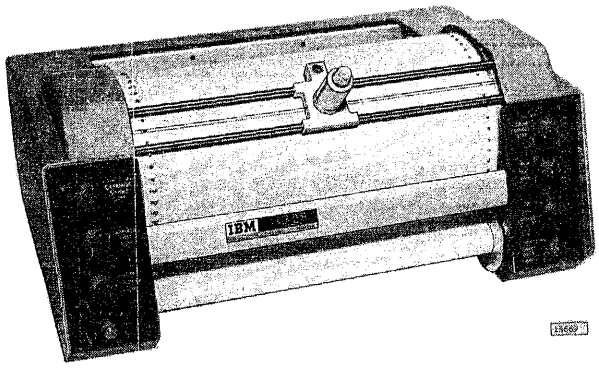


Figure 34. IBM 1627 Plotter

		Model 1	Model 2
Speed	X, Y Increments	18,000 Steps/Min	12,000 Steps/Min
	Pen Status Change	600 Operations/Min	600 Operations/Min
Increment Size		1/100 Inch	1/100 Inch
Chart Paper	Width	12 Inches	31 Inches
	Plotting Width	11 Inches	29 1/2 Inches
	Length	120 Feet	120 Feet
	Sprocket Hole Dimensions	.130 Inch Dia on 3/8 Inch Centers	.188 Inch Dia on 1 Inch Centers

Figure 35. 1627 Operating Characteristics

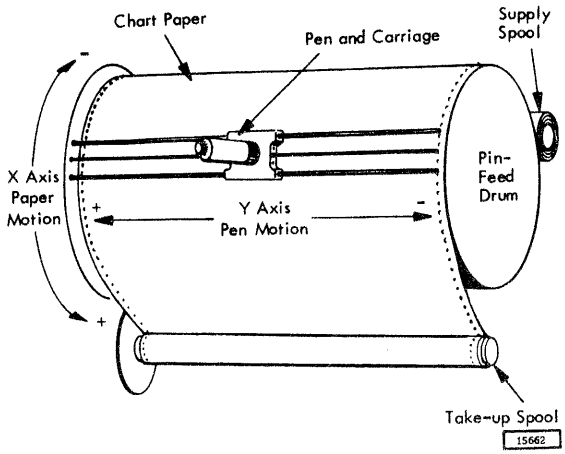


Figure 36. Plotter Paper and Pen Movements

ment of the pen carriage (Figure 37) and/or paper, or a raise-pen or lower-pen movement. The motion or action resulting from each word in the output record is shown in Figure 38.

The time required for execution of raise-pen and lower-pen commands is 100 ms. The time to plot a point is approximately 5 ms (3.3 ms for 300 steps/sec).

PROGRAMMING

The IBM 1627 Plotter operates under direct program control of the IBM 1130 Computing System.

I/O Control Commands (IOCC)

The 1627 is addressed by the five-bit device code of the IOCC.

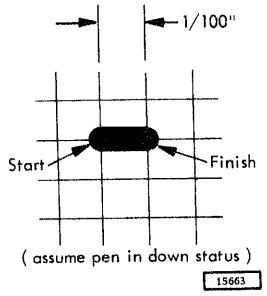


Figure 37. Result of One Horizontal (y-axis) Movement

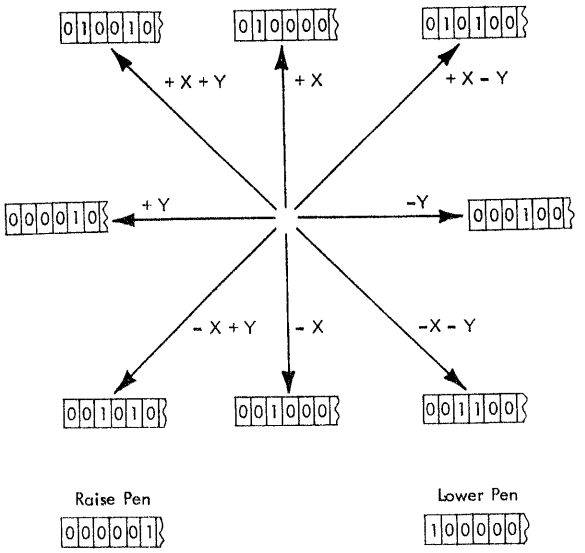
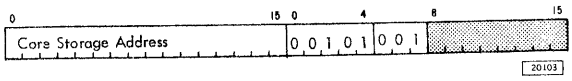


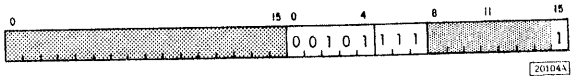
Figure 38. Plotter Command Codes

Write (001)



This command causes bit positions 0 through 5 of the word in the core storage location specified by the Address to be sent to the 1627 to control the movement of the pen or drum.

Sense Device (111)



This command causes the 1627 device status word (Figure 39) to be placed in the accumulator. Modifier bit 15 on specifies reset for the plotter response.

Interrupt

There is only one interrupt associated with the 1627 attachment: plotter response. This interrupt occurs

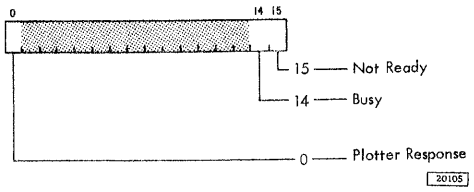


Figure 39. 1627 Device Status Word

when the 1627 has completed the action specified by the last character transmitted by the Write command.

Indicators

Not Ready: When this indicator is off, it indicates that the 1627 has power on and can accept information.

Busy: This indicates that the 1627 is in a busy status and cannot accept a character. After the first Write command the program should wait for succeeding plotter interrupts to initiate Write commands. If a Write command is given while busy is on, loss of information will probably occur. No indication is given of this loss.

IBM 1132 PRINTER

The 1132 Printer (Figure 40) provides printed output for the 1130 Computing System at maximum rates of 80 lpm (lines per minute) for alphamerical printing and 110 lpm for numerical printing. The print line is 120 print positions long; horizontal spacing is 10 characters per inch. Vertical spacing of six or eight lines per inch can be selected by the operator.

FUNCTIONAL DESCRIPTION

The 1132 contains a printwheel with 48 alphabetic, numeric, and special characters for each of the 120 printing positions. Special (FORTRAN) characters are as follows:

& - / . $, * ( ) ' + =

Each wheel rotates continuously and moves forward to print when the data in the output record specifies that the character to be printed is in the print position. Thus, all similar characters for the entire line are printed on the same cycle. Forty-eight cycles are required to print the complete line.

Forms control is provided through a tape-controlled carriage that uses the standard IBM carriage tape. Channels 1 through 6, 9, and 12 are available to the stored program.

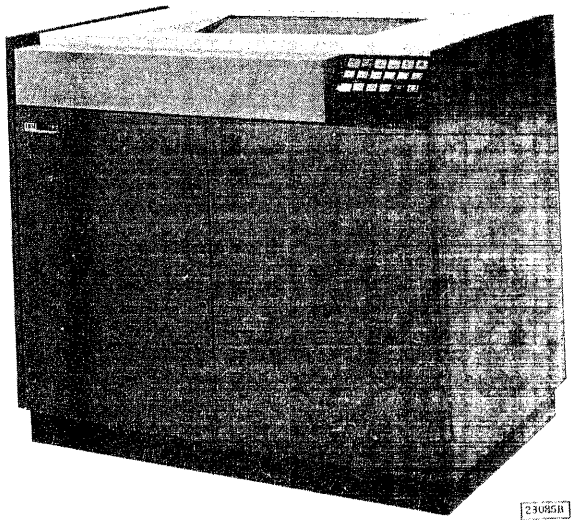


Figure 40. IBM 1132 Printer

The 1132 uses interrupt circuitry and responds on level 1. The core storage address related to the interrupt level is 0009; the device code of the printer is 00110. When an interrupt occurs, the DSW (Figure 41) for the printer can be sensed directly unless the communications feature is installed.

Operation

The data to be printed is assembled in core storage in the same order, including spaces, as the line that is to be printed. During each of the 48 cycles necessary to print all 48 characters, the character next in position to print is read from the character emitter and is compared with each character of the

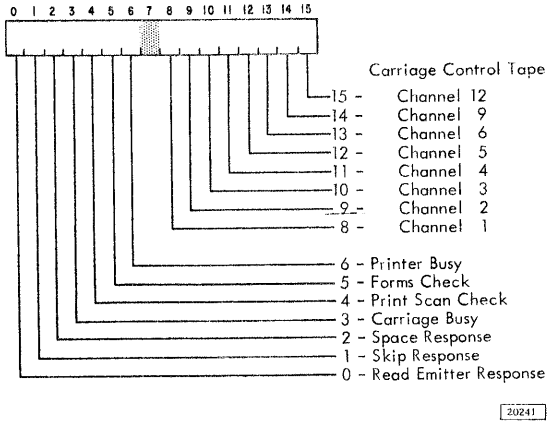


Figure 41. 1132 Device Status Word

output record. For each equal comparison, a 1 bit is put in the printer scan field in the position corresponding to the printwheel to be fired. The printer scans the field in a cycle-steal mode and fires each printwheel whose position contains a 1 bit. The printer scan field is located in core storage locations 0032 through 0039. The 16 bits of each of the first seven words and bits 0 through 7 of the eighth word represent the 120 printwheels.

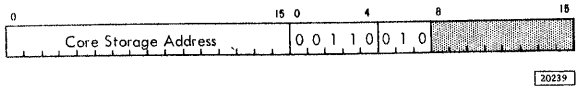
PROGRAMMING

The IBM 1132 Printer operates under direct program control of the CPU.

Printer Control Instructions

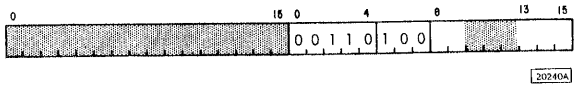
The 1132 Printer is addressed by the binary device code of 00110.

Read Emitter (010)



This instruction causes the eight-bit code of the next character to be printed, emitted by the printer, to be read into the core storage location specified.

Control (100)



This command causes the execution of the function specified by the modifier bit. A 1 bit in the position indicated in parentheses after each instruction causes the operation described.

Start Printer (Bit 8): This causes the printer to start taking the printer scan field information. The printer continues to take print scan cycles at 11.2-ms intervals until it receives a Stop Printer command. Each position that contains a 1-bit causes the corresponding printwheel to print the character in position on that cycle. After the field of eight words has been scanned, a 1 bit is placed in bit position 0 of the 1132 device status word. (See Figure 41.) This causes an interrupt when level 1 is the highest level waiting.

Stop Printer (Bit 9): This instruction causes the printer to be put in a ready (not-busy) state and inhibits subsequent printer interrupts. The Stop Printer instruction should not be given until all of the following conditions are met:

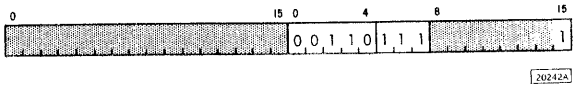
- 18 scan cycles have been completed after the command to print the last character.
- The carriage has stopped after a skip operation.
- The interrupt response from the last space command has occurred.

Start Carriage (Bit 13): This command initiates a skip operation, which is halted by a Stop Carriage instruction.

Stop Carriage (Bit 14): This command stops the carriage at the end of a skip operation. A punch in carriage control tape channel 1, 2, 3, 4, 5, 6, 9, or 12 initiates an interrupt request, identified by bit 1 of the DSW. When the desired tape channel bit in the DSW is on, a Stop Carriage command should be given.

Space (Bit 15): This command is given to space the carriage one line after a line is printed. After the space operation, an interrupt is initiated and a 1 bit is put in bit position 2 of the DSW to indicate spacing is completed. Another space can now be initiated.

Sense Device (111)



This instruction causes the DSW of the 1132 Printer to be placed in the ACC. The functions of the bit positions of the DSW are shown in Figure 37.

Programming Notes: Prior to initiating a Start Printer instruction, the program should set the printer scan field (0032-0039) to zeros to ensure that no erroneous bits are sensed on the first scan.

Before the first line of a record is printed, the status of the 1132 indicators should be checked. This is done by bringing the printer DSW into the ACC with a Sense Device instruction. The modifier bits of the Sense Device instruction should be set to zeros to prevent reset of the DSW responses and indicators. Bits 3, 5, and 6 (see Figure 41) of the DSW are tested. If all three positions are set to zero, the printer is ready to print the first line.

After the code of the next character has been emitted by the printer and read into core storage by a read emitter instruction, 11.2 ms (milliseconds) are available to scan the output record and set up the 1 bits in the printer scan field. At the end of 11.2 ms,

the printer begins its scan and fires each printwheel represented by a 1 bit. If the program has been interrupted for a considerable period by level 0, the programmed scan may not have been completed. To ensure that the program is aware of this condition, the first steps of the scan program for each character should clear the printer scan field to zeros and, upon completion of the programmed scan, place a bit in position 15 of the eight word (0039). When the printer scans the field it checks this position. If it is zero, the scan incomplete indicator (bit 4 of the DSW) is turned on. The program can test this indicator and branch to an error routine that provides for 47 idle scan cycles and a resumption of programmed scanning with the character not completed.

After the final scan cycle for a line of printing, 16 idle scan cycles must be taken before spacing or skipping can be started. If the operation is a single or double space, the next scan cycle can be started two scan cycles after the space operation is initiated. If the spacing operation is for more than three spaces, scan cycles for the next print line can be started after the last space command is given.

After each printer scan cycle, a 1 bit is placed in bit position 0 of the 1132 DSW, causing an interrupt when level 1 is the highest pending level. During an idle scan cycle the printer scan field should be set to zeros, except for bit 15 of the eight word. This prevents the incomplete scan indicator from being turned on.

IBM 1403 PRINTER

The IBM 1403 Printer (Figure 42) greatly increases the output capabilities of the IBM 1130 Computing System while reducing the time that the CPU is required to print thus leaving more time for other functions. The 1403 is available in two models for attachment to the 1130:

1. The Model 6 has a maximum printing speed of 340 lines per minute (lpm).
2. The Model 7 has a maximum printing speed of 600 lpm.

Each printer can print 48 different characters in 120 positions. There are 26 alphabetical, 10 numerical and 12 special characters.

Vertical spacing and skipping are initiated by the stored program. Horizontal spacing is 10 characters per inch. Standard vertical spacing is six and eight lines per inch, controlled manually by the operator. Skipping is about 33 inches per second.

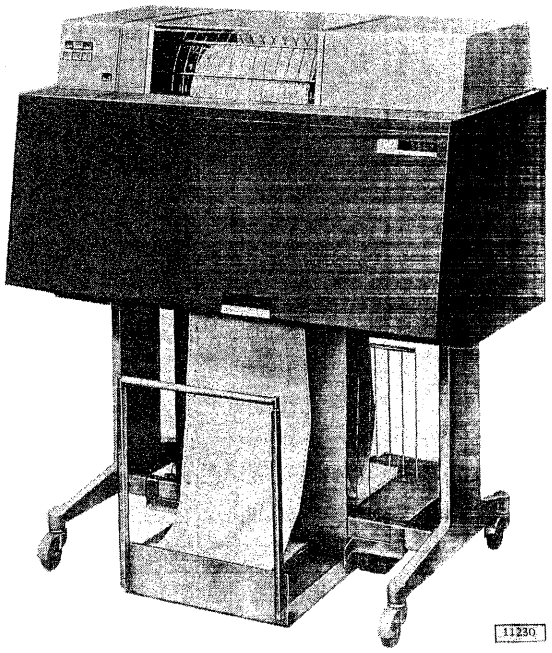


Figure 42. IBM 1403 Printer

FUNCTIONAL CHARACTERISTICS

Printing

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

Data to be printed must first be edited, translated to the 1403 Binary Code (Figure 43), and arranged in core storage in exactly the form that it is to be printed. The data format in core storage is two seven-bit characters per word (Figure 44).

Spacing and Skipping

Spacing is always performed one line at a time under control of the stored program in the CPU.

Carriage skipping is controlled by prepunched holes in a paper or plastic tape 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.

Character	Hex	Bits							
		0	1	2	3	4	5	6	7
		8	9	10	11	12	13	14	15
A	64	0	1	1	0	0	1	0	0
B	25	0	0	1	0	0	1	0	1
C	26	0	0	1	0	0	1	1	0
D	67	0	1	1	0	0	1	1	1
E	68	0	1	1	0	1	0	0	0
F	29	0	0	1	0	1	0	0	1
G	2A	0	0	1	0	1	0	1	0
H	6B	0	1	1	0	1	0	1	1
I	2C	0	0	1	0	1	1	0	0
J	58	0	1	0	1	1	0	0	0
K	19	0	0	0	1	1	0	0	1
L	1A	0	0	0	1	1	0	1	0
M	5B	0	1	0	1	1	0	1	1
N	1C	0	0	0	1	1	1	0	0
O	5D	0	1	0	1	1	1	0	1
P	5E	0	1	0	1	1	1	1	0
Q	1F	0	0	0	1	1	1	1	1
R	20	0	0	1	0	0	0	0	0
S	0D	0	0	0	0	1	1	0	1
T	0E	0	0	0	0	1	1	1	0
U	4F	0	1	0	0	1	1	1	1
V	10	0	0	0	1	0	0	0	0
W	51	0	1	0	1	0	0	0	1
X	52	0	1	0	1	0	0	1	0
Y	13	0	0	0	1	0	0	1	1
Z	54	0	1	0	1	0	1	0	0
0	49	0	1	0	0	1	0	0	1
1	40	0	1	0	0	0	0	0	0
2	01	0	0	0	0	0	0	0	1
3	02	0	0	0	0	0	0	1	0
4	43	0	1	0	0	0	0	1	1
5	04	0	0	0	0	0	1	0	0
6	45	0	1	0	0	0	1	0	1
7	46	0	1	0	0	0	1	1	0
8	07	0	0	0	0	0	1	1	1
9	08	0	0	0	0	1	0	0	0
=	4A	0	1	0	0	1	0	1	0
$	62	0	1	1	0	0	0	1	0
.	6E	0	1	1	0	1	1	1	0
'	0B	0	0	0	0	1	0	1	1
,	16	0	0	0	1	0	1	1	0
(	57	0	1	0	1	0	1	1	1
-	61	0	1	1	0	0	0	0	1
)	2F	0	0	1	0	1	1	1	1
+	6D	0	1	1	0	1	1	0	1
/	4C	0	1	0	0	1	1	0	0
*	23	0	0	1	0	0	0	1	1
&	15	0	0	0	1	0	1	0	1

30015A

Figure 43. 1403 Binary Code

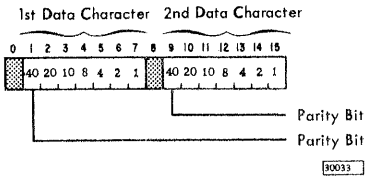


Figure 44. 1403 Data Format - Hexadecimal Bits

Control Tape

The control tape has 12 columns, 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 forms, although for convenience, the tape blanks are slightly longer. Horizontal lines are spaced six to the inch for the entire length of tape. Round holes in the center of the tape are prepunched for the pin-feed drive that advances the tape in synchronization 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.

Keys and Lights

The keys and lights (Figure 45) provide the operator control of the printer during setup and program interruptions that require operator attention.

Start Key: This key puts the machine in a ready status. A duplicate Start key is located in the rear of the printer for operator convenience.

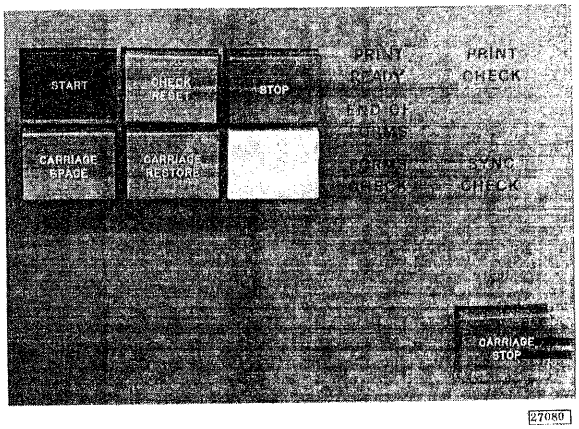


Figure 45. 1403 Operator Control Panel

Check-Reset Key: This key is used to reset a printer error indication. Then the Start key is used to restart the operation.

Stop Key: This key is used to remove the printer from the ready status. A duplicate Stop key is located at the rear of the machine for operator convenience. The Stop key also turns off the Ready light.

Carriage Restore: The Carriage Restore key causes the carriage to be positioned at channel 1 when the printer is not ready. If the carriage feed clutch is disengaged, the form does not move. If it is engaged, the form moves with the control tape.

Carriage Stop: This key stops the carriage operation, and turns on the forms check light.

Carriage Space: This key causes the carriage to advance the form one space if the clutch is engaged if the printer is not ready.

End-of-Form Light: This light shows an end-of-form condition and the machine stops. If an end-of-form occurs during a skip or while spacing within the last form in the printer, printing continues automatically until the skip to a new form occurs.

Forms Check Light: This light indicates form-feed trouble in the forms tractors. This light turns off when corrective action is taken and the Check Reset key is pressed.

Ready Light: This light indicates that the printer is ready to print.

Print Check Light: This light indicates a print error.

Sync Check Light: This light indicates that the chain is not in synchronization with the compare counter for the printer. The timing is automatically corrected and the light is extinguished by pressing the check reset key.

A synchronization check may result if the forms cart is not in contact with the base of the machine through the grounding straps.

Manual Controls

Feed Clutch: The feed clutch controls the carriage-tape drive and form-feeding mechanism and selects six- or eight-lines-per-inch spacing. When set to neutral, the clutch is disengaged and automatic form feeding cannot take place.

Paper Advance Knob: This knob positions the forms vertically. It is used only when the feed clutch is disengaged.

Vertical Print Adjustment: This knob controls fine spacing adjustment of forms at the print line. The carriage tape is not affected by this knob.

Print-Unit Release Lever: This lever permits access to the form transport area.

Print-Line Indicator and Ribbon Shield: The lower ribbon shield is also used as a print-line guide. It pivots with the ribbon mechanism when the print unit is opened but may be unlatched from the print unit and pivoted independently. The front side of this plastic shield is marked to show print-position locations.

When the ribbon shield is used as a print-line indicator, the ends of the shield indicate where the lower edge of the character prints.

Horizontal Adjustment: This device 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 limit.

Right-Hand Tractor Vernier: This knob controls fine adjustment in paper tension. It can be used for adjustments of up to 1/2 inch.

Tractor Slide Bar: The forms tractors are mounted on two tractor slide bars, upper and lower. To facilitate positioning forms tractors, notches are provided in the tractor slide bar. These notched settings provide for form widths from 3-1/2 inches to 18-3/4 inches.

Print-Density Control Lever: The print hammer unit is designed to accommodate different thicknesses of forms. To provide a vernier control for print impressions, a print-density control lever is used. Print density varies from A (darkest) to E (lightest).

Print-Timing Dial: This dial is set to a fixed indicator and is used for fine adjustment of print quality. The proper setting of the dial is obtained from the print-timing dial chart located on the ribbon cover.

Indicator Panel Lights

In addition to the lights on the exterior of the 1403, an indicator panel, located below the feed clutch (inside the front cover), is a diagnostic aid.

Gate Interlock: This light indicates the print unit is not locked into position.

Brush Interlock: This light indicates the carriage tape brushes are not latched in position.

Shift Interlock: This light indicates that the manual feed clutch is not properly positioned.

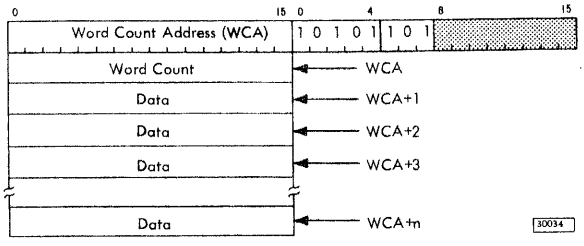
Thermal Interlock: This light indicates that a thermal unit has caused a fuse to burn out. An IBM Customer Engineer should be notified.

PROGRAMMING

All operations of the IBM 1403 Printer are under control of the stored program in the CPU.

Data to be printed must be edited, translated to the 1403 Binary Code, and arranged in core storage in exactly the form that it is to be printed.

Initiate Write (101)



An Initiate Write command transfers data from core storage to the print buffer using the cycle steal method.

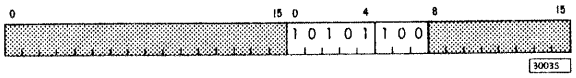
During data transfer each core location to be printed is addressed twice. During the first cycle, bits 1-7 are transferred to an even address of the print buffer. During the second cycle, bits 9-15 are transferred to the next higher odd address of the print buffer.

The total time demand on the processor is dependent on the core storage cycle time. Approximately 432 μsec is required for the 3.6 μsec core storage, and approximately 264 μsec is required for the 2.2 μsec core storage.

The printer does not interrupt the CPU until after the 120-position buffer is filled. It then initiates a transfer complete interrupt on level 4.

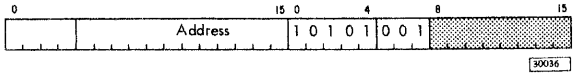
After completion of printing the line a level 4 interrupt is initiated to signal print complete.

Control (100)



An XIO Control command initiates a single line space.

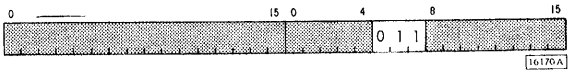
Write (001)



An XIO Write command controls carriage skipping. This command causes the carriage to skip even if it is at the specified channel. It skips until that channel is detected again. The carriage may be controlled to skip to any channel 1-12 by placing a 1-bit in the address 4-15, respectively.

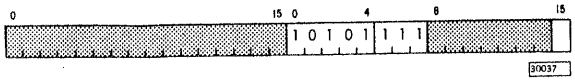
A carriage control command given prior to loading of the print buffer causes immediate execution of the command. If the command is given during loading of the buffer, the command is not executed until after the line is printed. The programmer must check to insure that the carriage is not busy when the command is given.

Sense Interrupt (011)



This command causes a bit 4 to be placed in the ILSW (level 4) if the 1403 is the interrupting device.

Sense Device (111)



An XIO Sense Device command causes the 1403 DSW (Figure 46) to be placed into the accumulator. If bit 15 is on when the command is executed, the DSW interrupt and channel 9 and 12 indications are reset.

Device Status Word Indicators

Bit 0 – Parity: This bit in the DSW indicates an even bit count.

Bit 1 – Transfer Complete Interrupt: This bit indicates that the 1403 buffer is full.

Bit 2 – Print Complete Interrupt: This bit indicates the 1403 has completed printing a line.

Bit 3 – Carriage Interrupt: This bit indicates the carriage has completed a space or skip operation.

Bit 5 – Ring Check: This bit indicates an error occurred in modification of the Buffer Address Register.

Bit 6 – Sync Check: This bit indicates that the print chain is not in synchronization with the compare counter.

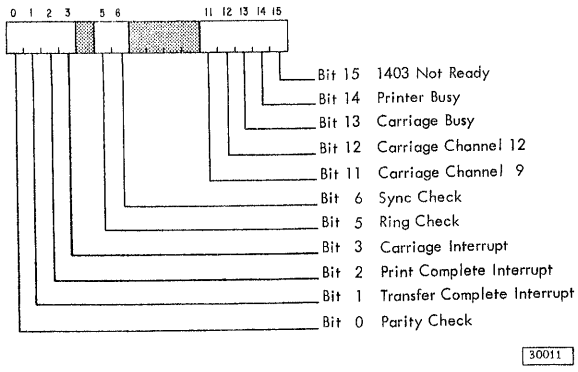


Figure 46. 1403 Device Status Word

Bit 11 – Carriage Channel 9: This bit indicates the carriage passed a channel 9 punch in the carriage control tape.

Bit 12 – Carriage Channel 12: This bit indicates the carriage passed a channel 12 punch (normally used for the last printing line on a form) in the carriage control tape.

Bit 13 – Carriage Busy: This bit indicates that the carriage is performing a space or skip operation. The bit goes off when bit 3 comes on to signify completion.

Bit 14 – Printer Busy: This bit indicates that the 1403 buffer is being loaded or a line is being printed.

Bit 15 – 1403 Not Ready: This bit indicates the 1403 is not ready. Printing, spacing, or skipping under program control cannot occur until the 1403 is ready.

IBM 1231 OPTICAL MARK PAGE READER

The IBM 1231 Optical Mark Page Reader (OMPR) represents a breakthrough in the area of source recording and data entry. The OMPR provides a facility for recording the data at its source, in a form that can be read directly into the IBM 1130 Computing System.

The 1231 (Figure 47) reads positional marks made by an ordinary lead pencil on paper documents. The positional marks are read directly into the 1130 core storage.

Documents are read at a maximum rate of 2,000 sheets per hour.

FUNCTIONAL CHARACTERISTICS

The 1231 uses sonic delay lines for storing controls and data. Controls are marked on the regular data sheet and entered into delay line storage during the program load cycle. This data sheet is referred to as a program control sheet. The program control sheet is automatically placed in the select stacker during the load cycle.

As data sheets are read, data is stored in the delay lines according to instructions from the program control sheet. Each word to be stored on the delay line must be programmed by the program control sheet.

When a data sheet passes under the photoelectric read head, each word is tested for conditions, such as no-mark, multi-mark, or other-than-one. Switches

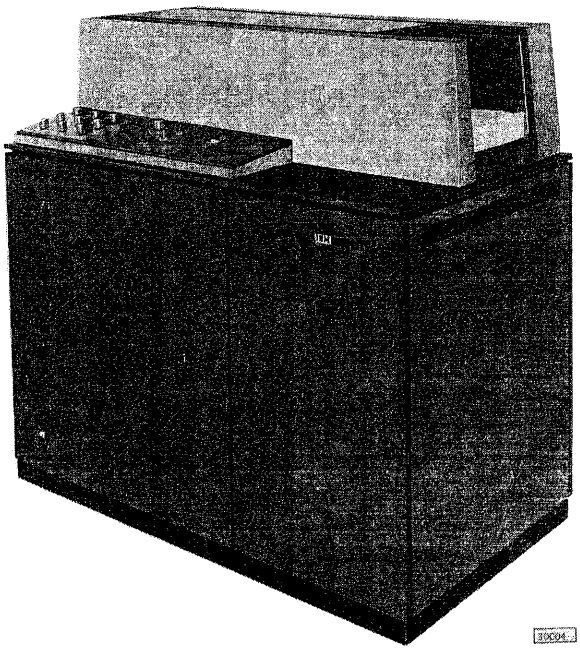


Figure 47. IBM 1231 Optical Mark Page Reader

on the 1231 control panel in conjunction with the program control sheet control the test of these conditions. Any word that does not pass the requirements of the switch settings causes the data sheet to be routed to the select stacker.

Document Path

The data sheet begins its movement through the Optical Mark Page Reader when it is fed from the hopper by CPU program control. The document then passes under a read head and is next transported through the transport area, past a selection station, and on into one of the two gravity stackers.

IBM 1231 Message Format

Each word transferred from the 1231 to the 1130 reads into a single position of core storage. Words are transferred one segment at a time to the A buffer and the B buffer in the attachment; all odd segments (A buffer) enter positions 0-4 and 14, and all even segments (B buffer) enter positions 5-9 and 15 (Figure 48). If the 1231 is programmed for only one segment, all segments enter positions 0-4 and 14. Words with marks in positions 0, 1, 2, 3 or 4 transfer

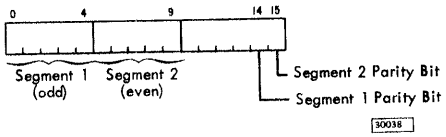


Figure 48. 1231 Data Format

to core storage as an odd segment and marks in positions 5, 6, 7, 8, or 9 transfer to the 1130 as an even segment. Combinations of the bits make up a valid character which must be translated by the 1130 stored program. Any or all of the marking positions on the data sheet may contain marks.

Data is read by the 1231 from left to right, top to bottom, a row at a time. Information from a data sheet is stored in the following sequence:

1. Segment one of the first word programmed to read.
2. Segment two of the first word.
3. Segment one of the second word programmed to read.
4. Segment two of the second word.

If only one segment of any word is programmed to read, then each segment goes into a separate core storage word.

Keys and Switches

Start Key: A depression of the Start key establishes the 1231 in a ready condition after the program sheet has been loaded.

Stop Key: A depression of the Stop key halts document feeding and lowers the hopper to allow for the loading of more data sheets.

Reset Key: A depression of the Reset key raises the hopper to the feed position and resets the electronic circuitry. Check or error conditions should be analyzed before pressing the reset key. The Start key must then be pressed to place the 1231 in a ready condition.

Program Load Key: A depression of the Program Load key clears the delay line storage of previously stored data, and conditions the machine for program loading. This key is lighted during the program load cycle. The Start key is then pressed to load the program sheet.

Master Mark Switch: The Master Mark switch is active only on machines equipped with the Master Mark special feature. This switch controls the capability of the 1231 to recognize a master mark on the right edge of the data sheet. When this switch is on, the recognition of a master mark causes new master data to be accepted. This data is placed in ten core storage locations, controlled by the CPU Program, which are not a part of the area for storing detail data. The master data will remain unchanged by the 1231 in the 1130 processor until another sheet with a master timing mark is recognized by the 1231 and by the 1130 program. The 1130 is alerted by means of a master mark indicator in the device status word.

Feed Mode Switch: The feed mode switch has two settings: CONTINUOUS and ON-DEMAND. This switch must always be set to ON-DEMAND for 1130 operations.

Check Length Switch: Three check length switches are located on the operator's panel, one for each of three sets of switches associated with fields. These switches have two settings: SEGMENT and WORD. The settings define the length of the item as it is to be checked for each field. The SEGMENT setting checks for a one-segment word; the WORD setting checks for a two-segment word.

Select Condition Switches: Each of the three select switches has four settings: OFF, NO MARK, MULTI-MARK, and OTHER-THAN-ONE. Each switch is associated with a check length switch and one of the three fields. The settings represent the conditions in a given field under which a document is directed to the select stacker.

Read Mode Switches: These three switches, each associated with a set of field checking switches, determine the conditions of mark discrimination. Each read mode switch has four settings: single response, multiple response, single response select uncertainties, and multiple response select uncertainties. These settings allow the 1231 to discriminate between good marks (heavy mark) and poor marks (light mark or poor erasure).

Control Timing Mark Switch: This switch enables the 1231 to eliminate the 90-ms delay associated with the timing mark checking feature indicator. The switch has two settings: YES and NO. YES is used when the documents to be processed have 106 timing marks. NO is used for normal documents and eliminates the 90-ms delay.

Timing Mark Check Switch: This is an 11-position rotary switch with settings numbered 0 through 9 and OFF. The switch is preset by the operator to match the units position count of timing marks on the data sheets to be processed. For example, if there are 106 timing marks on a document to be processed, the switch is set to 6.

Lights

Start Key Light: The start key light being off, indicates that the machine is in a ready state. The light goes off when the start key is pressed, and the light remains off until the machine is conditioned to the not-ready state.

Feed Check Light: This light indicates a sheet jam, a misfeed, a double-sheet feed, a full stacker, or an empty hopper. These conditions cause the machine to stop, and the condition must be corrected before the light can be turned off by pressing the reset key.

Process Check Light: This light indicates the following conditions:

1. A parity error in storage logic.
2. The count of data-sheet timing marks is not in agreement with the setting of the timing mark switch.
3. Failure of processing unit to take data from the B register before the A register loaded new data into it.
4. A logic or delay line failure when:
   a. No control bits are loaded into the master line during the reading of a program control sheet.
   b. No data bits are loaded during the reading of a data sheet.
5. Timing marks on the detail data sheet does not equal at least the number of words programmed to read by the program control sheet.

Read Light: This light indicates that the read head lamp is burned out or weak. If a depression of the Reset key does not turn off the read light indicator, an IBM Customer Engineer should be called.

System Stopped Light: This light is turned on whenever the 1130 is stopped while connected to the 1231.

Refeed Sel Doc Light: This light comes on whenever one or more of the following conditions occurs (the last document in the select stacker must be reprocessed);

1. A multi-mark is detected during the reading of the master mark document.
2. An uncertainty is detected during the reading of the program control sheet.
3. An uncertainty, without an accompanying dark mark, is detected during the reading of the master mark sheet.

PROGRAMMING

Programming for the IBM 1231 Optical Mark Page Reader depends upon two sources of control: controls stored within the 1231 and controls received from the 1130. Controls stored within the reader are entered into storage from a program control sheet.

Program Control Sheet

A program control sheet is a data sheet with certain operational controls marked in the data areas. Each word from the data sheet consists of ten positions. Each word in the delay line storage consists of 16 positions: ten for the positions on the data sheet, and six for storing operational and internal controls generated by circuitry of the 1231.

Every word that is to be retained for transferral to the 1130 must have an operational control marked in that word on the program control sheet. When operational control information is entered into storage, the controls go into some of the six control positions associated with each word.

During the program load cycle, the mark positions used as control positions are:

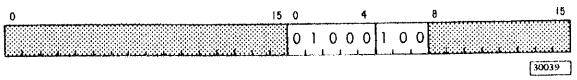
1. A mark in position 8. This designates that a word is to be stored. This word is available later for readout to the 1130 system. The number of timing marks on the detail data sheet must at least equal the number of words programmed to read by the program control sheet.
2. A mark in position 0 and a mark in position 8. This stores data from segment 1 only. (Bits 0-4 or 10-14 of the data sheet.)
3. A mark in position 5 and a mark in position 8. This stores data from segment 2 only. (Bits 5-9 or 15-19 of the data sheet.)
4. A mark in position 6. This indicates the start of field checking according to the settings of Field I switches.
5. A mark in position 7. This indicates the start of field checking according to the settings of Field II switches.
6. A mark in position 6 and a mark in position 7. This indicates the start of field checking according to the settings of Field III switches.

System Programming

There are four I/O commands used with the 1231: Control (100), Read (010), Sense Interrupt (011), and Sense Device (111). In addition the Control command uses three modifier bits to expand the number of commands.

The 1231 is addressed by the five-digit device code (01000=area code 8).

Control (100)



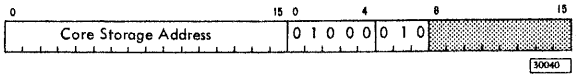
The control command uses three modifier bits:

Read Start (bit 13): Causes the document to move through the read station. Data is collected and placed on the delay line controlled by the control sheet and switch settings. As soon as the first word programmed for output is placed on the delay line, it is made available to the processor.

I/O Disconnect (bit 14): Terminates the read operation from the document and clears the delay line storage by signaling the 1231 that no more data is desired. This command should be given to prevent a read (overrun) error on the next document if all data from the previous document is not cleared from the delay line storage.

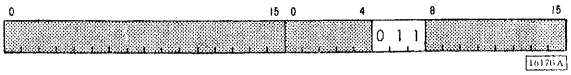
Select Stacker (bit 8): Causes the document just read to enter the select stacker. This command can be given within 50 ms after the last word is placed on the delay line. Indicator bit 5 in the device status word is on if it is permissible to select the document. This bit should be tested prior to issuing the Select command. If the bit is off and a command is issued the command will be ignored.

Read (010)



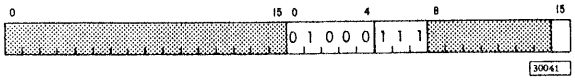
This command causes the next word in the 1231 attachment buffer to be loaded into the core storage location specified by the address.

Sense Interrupt (011)



The Sense Interrupt command causes a bit 5 to be placed in the accumulator if the 1231 caused the interrupt or wishes to be serviced.

Sense Device (111)



The Sense device command causes the 1231 DSW (Figure 49) to be placed in the accumulator. Modifier bit 15 resets the responses.

Interrupts

There are four interrupts associated with the 1231. All are on level 4.

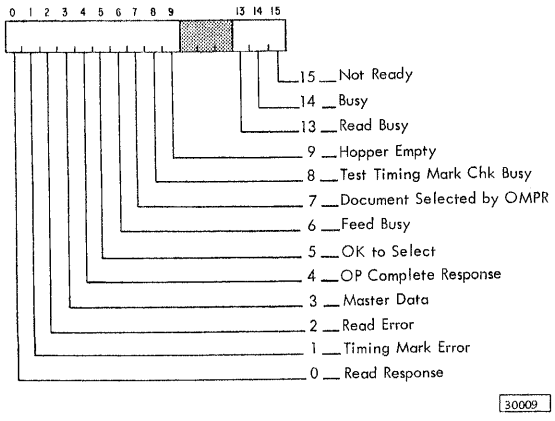


Figure 49. 1231 Device Status Word

Read Request Interrupt: Signals that a word (one or two segments) has been loaded in the 1231 attachment buffer and can be accepted by the CPU. An XIO Sense DSW command with bit 15 turns off the read request and loads the 1231 into the accumulator. An XIO Read command is needed to turn off the read busy indicator and transfer the word to core storage.

Operation Complete Interrupt: Occurs after the last word has been accepted by the CPU. End of transmission from the 1231 causes this interrupt. Timing mark error and read error can also cause this interrupt. See Operation Complete.

It is possible to read the last character of a document and receive the end of transmission signal from the 1231 before the timing mark check is made if the timing mark switch is set to yes. This would turn on the operation complete interrupt, which would then be serviced by the processor. If the Timing mark switch were set to YES, a timing mark error could occur after the operation complete routine. Were this to occur, the Timing Mark Error indicator would be turned on. This indicator would remain on until the 1231 is placed in the ready state. The operator would be aware of this error either by program analysis or by the visual indication of the 1231 control panel.

Timing Mark Error Indicator: When there is a timing mark error. Timing mark error turns on the operation complete interrupt. The error is dependent upon the settings of two switches -- Timing Mark Check switch and Control Timing Mark switch. The timing mark check switch is an 11-position rotary switch. It has an off position and ten positions labeled 0 through 9. In the off position no checking is performed. If checking is desired, the switch is placed on the digit corresponding to the units value of the number of timing marks on the documents. The control timing mark switch has two positions labeled YES and NO. The no position is used for documents having the normal number of timing marks (100 or less); the yes position is used for documents having 106 timing marks.

Read Error Indicator: On if an even count (parity), overrun, or no bits error is detected. The error turns on operation complete interrupt to alert the processor and prevent further transfer of data from the attachment.

Device Status Word Indicators

Not Ready: If this indicator is off the 1231 is ready to accept instructions from the CPU program. The following conditions are necessary for the 1231 to remain in a ready condition:

1. Power on.
2. Off line-on line switch set to ON LINE.
3. Control sheet loaded.
4. Hopper loaded.
5. No read or feed errors.
6. Start key is depressed after the program load light on the 1231 goes off after loading the control sheet.

Busy: This indicator comes on after a Read Start command has been issued and remains on until an operation complete interrupt is received or an XIO Disconnect command is issued. This indicator being on indicates that a document is being read.

Read Busy: This indicates that the 1231 attachment buffer is full. It is turned off when the XIO Read command for the 1231 is given.

Feed Busy: This indicator comes on when an XIO Control command with modifier bit 13 is executed. It remains on until the first 1231 interrupt is turned off. Another start feed command should not be given while this indicator is on.

Read Response: This indicator comes on each time the attachment buffer is loaded. It is reset with the XIO Sense Device command with bit 15 on.

Operation Complete: This indicator is turned on by the end of transmission and signifies that the last word has been read by the CPU. It is also turned on by a timing mark error or read character error. It is turned off by an XIO Sense Device with bit 15 on.

Okay to Select: This indicator comes on when a document read is initiated and remains on for 50 ms after the document has been read.

Test Timing Mark Check: This indicator comes on just before the first character interrupt from the 1231 attachment and remains on for 90 ms after the last word has been placed on the delay line. If the control timing mark switch is set to "YES", the timing mark check is not made until the end of the 90 ms period. If this indicator is to be tested, data processing should not take place until the indicator goes off.

Master Mark: This indicates that data to be transferred is master data. A master data subroutine should place master data in a reserved area. The indicator does not come on if the master mark switch if off.

Read Error: This indicates that a parity error (even count) occurred, an overrun condition occurred, or that no bits were entered onto one of the delay lines with either a data sheet or a control sheet. The not ready and the operation complete interrupt is turned on. The dealy line storage is cleared so that it is not necessary to give an I/O disconnect.

Document Selected: This is caused by either a mark count reject or data uncertainity, according to the setting of the field checking switches and the program control sheet. It is turned off by an XIO Sense Device with bit 15 on. Document feeding is not inhibited. When this indicator is turned on the delay line storage is cleared of data and the transfer to the processor is terminated. The operator must be flagged by the program that this has happened. If the next Feed command has been issued it will be necessary to refeed two documents. If the document selected indicator is turned on and serviced before the next feed command, only the top sheet in the select stacker should be processed. Data should not be processed if this indicator is on.

Hopper Empty: This indicates that the 1231 hopper is empty and turns on not ready.

IBM 2310 DISK STORAGE

The IBM 2310 Disk Storage (Figure 50) provides additional random-access storage capabilities for the IBM 1130 Computing System. Up to five disk storage drives may be attached to the 1130 including the Single Disk Storage located within the 1131 CPU.

The 2310 Model B1 consists of a housing that contains one Disk Storage Drive and space for an additional drive. The 2310 Model B2 contains two Disk Storage Drives. A maximum of two 2310s may be attached to the 1130 via the channel multiplexer in the 1133.

FUNCTIONAL DESCRIPTION

The 2310 Disk Storage is a small, compact device that has both random and sequential access capabilities and provides low-cost bulk storage with relatively fast access. With a full-capacity system,

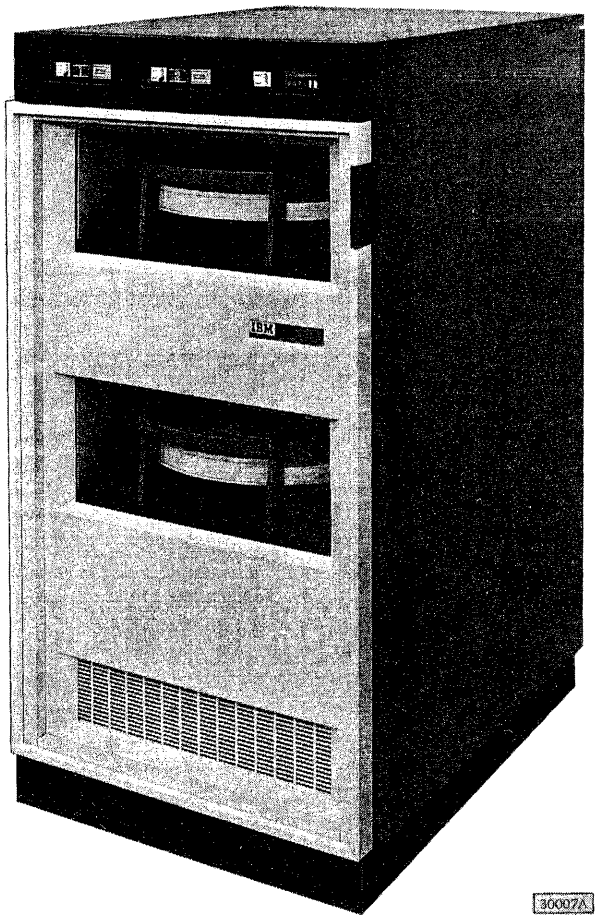


Figure 50. IBM 2310 Disk Storage Model B2

there are 2,560,000 words on-line in addition to core storage.

The disk storage recording medium is an oxide-coated disk in an interchangeable 2315 Disk Cartridge. A magnetic head, provided for each surface, performs reading and writing functions. A stepping actuator positions the heads over any one of 203 two-track cylinders; 200 are available for customer use and three are reserved for system use.

Incremental cylinder-to-cylinder motion is used for all head movement. A magnetic actuator moves the carriage, with the heads attached, either one or two cylinders at a time. Each movement requires 15 milliseconds; therefore, a movement of 10 cylinders requires 75 ms. An additional 20 ms to 25 ms are required for the heads to stabilize before reading or writing can begin. This delay is provided automatically by the drive.

Data Organization

The 2315 Disk Cartridge is divided into 200 cylinders of two tracks each: one on the upper surface and one on the lower surface. A head is provided for each of these surfaces.

Each track is further divided into four sectors for ease of block handling. The capacity of each disk available to the customer is 512,000 words.

Data Transfer Checking

Data that is transferred between the disk storage attachment and disk storage is checked for parity to ensure accuracy. The number of bits of each word is divided by four as the word is transferred, and the number of bits necessary to make the division even is added to the end of the word. This method of checking is known as modulo 4.

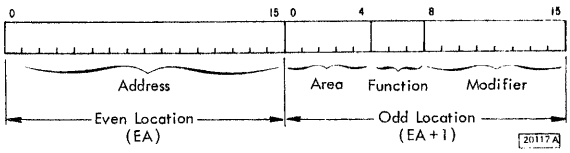
Changing the 2315 Disk Cartridge

Use the procedure described in the Single Disk Storage section of this manual to safely change the 2315 Disk Cartridge.

PROGRAMMING

Each Disk Storage Drive, including the drive located within the CPU, is assigned an area code.

I/O Control Command (IOCC)



Address

The address word is used in two different ways:

1. When used with Initiate Write and Initiate Read, the address word specifies the location in core storage of the data table. The first word in the data table contains the number of words to be

operated on in that sector. For example, if the command address contains 1000 and the sector word count found at the core storage location is 50, then the table of data occupies locations 1001 through 1050.

2. When used in conjunction with the control function, the address word specifies the number of cylinders to be moved.

Device (Area)

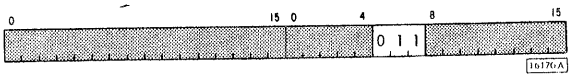
This five-digit field identifies the specific Disk Storage Drive:

Drive No.	Device Code
1	17 (10001)
2	18 (10010)
3	19 (10011)
4	20 (10100)

Function

The primary I/O functions are specified by the three-bit function code.

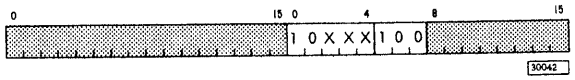
Sense Interrupt (011)



This command causes the accumulator to be loaded with the interrupt level status word (ILSW) for the highest interrupt level in progress. Since all disk storage drives are assigned to interrupt level 2, ILSW bits are assigned to each drive:

Drive No.	ILSW Bit
1	1
2	2
3	3
4	4

Control (100)

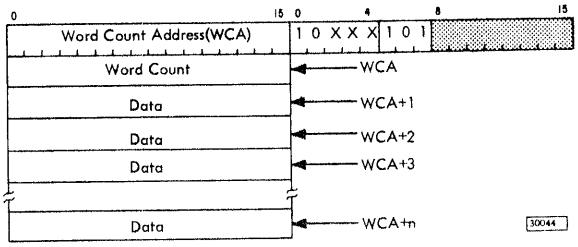


This command causes the disk storage to seek a specified number of cylinders. The direction of the seek operation is specified by modifier bit 13. Bit 13 set to zero moves the actuator in a forward motion (toward the center of the disk). Bit 13 set to one moves the actuator toward the home position (outside edge of disk).

The address portion of the IOCC is used with the Control command to specify the number of cylinders to be moved.

A Control command with a zero word count performs a No-Op, and no interrupt occurs.

Initiate Write (101)

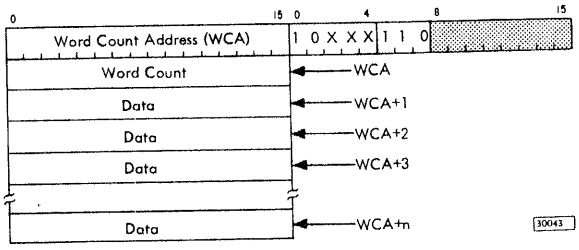


This command writes a specific number of words (up to 321) from the CPU core storage to disk storage. The exact number of words is specified by the word count. To write more than one sector, a separate Initiate Write command must be given for each sector or portion of a sector to be written. Modifier bits contain the binary address of the disk sector to be written.

The disk storage should not be instructed to write without first checking the ready-not busy status.

A Write command with a zero word count should not be issued.

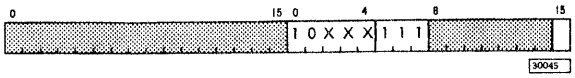
Initiate Read (110)



This command reads a specific number of words (up to 321) from the disk storage to core storage. The exact number of words is specified by the word count. To read more than one sector, a separate Initiate Read command must be given for each sector to be read. Modifier bits contain the binary address of the disk sector to be read.

An Initiate Read command with a zero word count should not be issued.

Sense Device (111)



This command causes the device status word (DSW) to be read into the accumulator. All indicators associated with the 2310 DSW are reset when modifier bit 15 is on (set to 1) except a power unsafe or write select condition. These conditions are only reset by stopping the disk motor and then restarting it.

The device status word is shown in Figure 51. Refer to the Indicators section under Single Disk Storage in this manual for a description of the DSW indicators.

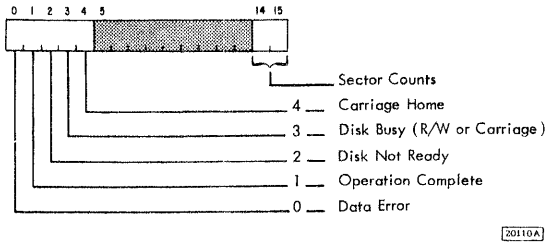


Figure 51. Disk Storage Device Status Word

Modifier

Bits 8 through 15 of the IOCC are defined in the modifier field. These bits are used in a variety of ways in conjunction with the function code.

Bit 8: When used with Initiate Read, bit 8 is decoded to determine the specific read operation. If bit 8 is on, a read-check operation is performed. The difference between read-into-core storage and read-check is that read-into-core storage causes the data to be stored in the specific table area in core storage, whereas read-check passes the data through the disk storage adapter data register and checks it for modulo-4 errors. No cycle steals are taken during a read-check operation, so the processor is free for other use once the read-check is initiated.

Bit 13: This bit is used in conjunction with two functions:

1. When used with the Control function, bit 13 set to 1 specifies movement of the actuator toward the outer edge of the disk (backward). Bit 13 set to 0 causes the actuator to move toward the center of the disk (forward). The number of cylinders moved is specified by the address portion of the IOCC.
2. Bit 13 is also used with bit 14 and bit 15 as the sector address.

Bit 13, Bit 14, Bit 15: When used with Initiate Write or Initiate Read, these three bits contain the three-bit binary address (0-7) of the disk sector to be read or written.

Bit 15: Bit 15 is used with two functions;
1. When used with Sense Device, bit 15 set to 1 resets the operation complete and data error indicators. Unless the data error bit 0 is caused by the power unsafe latch or write select error.
2. When used with Initiate Read or Initiate Write, bit 15 (together with bit 13 and bit 14) is used for the sector address.

STORAGE ACCESS CHANNEL

The Storage Access Channel (SAC) provides the 1130 with additional I/O capability. If the 1403 Printer or 2310 Disk Storage is included in the system, then it is necessary to attach the 1133 Multiplex Control Enclosure to the SAC. However, an additional channel (SAC II) is provided by the 1133 as a special feature.

Through the facilities of SAC or SAC II, the user may attach his own device. The customer device may interrupt on any level from 2 through 5. Any bit within ILSWs 2 through 5 that has not been previously assigned may be used. This is also true for the assignment of area codes for the customer device. The customer device may be assigned any area code that has not been previously assigned.

FUNCTIONAL DESCRIPTION

The storage access channel feature allows external devices or systems to communicate directly with the 1131 core storage unit. The transfer of data to or

from core storage and the SAC takes place in one of two modes.

1. Cycle Steal Mode. An XIO instruction, Initiate Read or Initiate Write, gives control of the data transfers to the SAC. When the SAC transfers a word or words to or from core storage, CPU cycles are stopped and a cycle steal cycle or cycles are taken. The CPU program has no control of or awareness of the cycle steal cycles.
2. Interrupt Mode. The external device can cause an interrupt of the CPU program by bringing up an interrupt-request-level 2, 3, 4, or 5 line, which is serviced by the CPU in the same manner as the basic interrupts.

Because of the SAC's ability to interrupt on levels 2, 3 and 5, interrupt level status words for these levels, as well as for level 4, are provided so that the CPU program may determine which device caused the interrupt.

When an interrupt is caused by a basic device, the CPU program must give an XIO Sense Interrupt command. The attachment for the device places the ILSW bit for that device on the I/O input bus, and reads the bit into the B register and transfers to the Accumulator. If a device on the SAC causes an interrupt, the CPU program must give an XIO Sense Interrupt command, and the device must decode the command and place its ILSW bit on the I/O input bus to be read into the B register.

When an XIO Sense Device command is given to the SAC, the device must decode the command and set the status bits on the I/O input bus.

The customer must provide his own interrupt routines and controlling programs.

The customer may assign to devices on the SAC, any area codes that are not already assigned to a basic device on his system. The decoding of the area codes is done in the devices on the channel.

The customer may assign any bit in the ILSW to a device on the SAC that is not assigned to a basic device on his system.

No change is made to the 1131 or the SAC attachment in the assignment of area codes, interrupt levels or ILSW bits.

Cycle Steal Priority

There are four cycle steal priority levels. The Disk Storage devices are on level 0; SAC is on level 1; the 1132 is on level 2; and the 2501 is on level 3. There is no polling of cycle steal requests. That means the SAC, by keeping its request active, may completely block the 1132 printer and other lower priority devices.

PROGRAMMING

The storage access channel (SAC) operates on the IBM 1130 system under direct program control or cycle steal control.

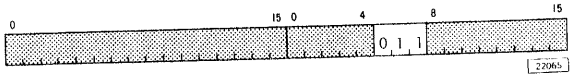
An XIO instruction addresses an I/O control command (IOCC) word which is placed on the I/O output bus.

The devices or systems on the SAC must decode the IOCC area code to select one device or system for the operation.

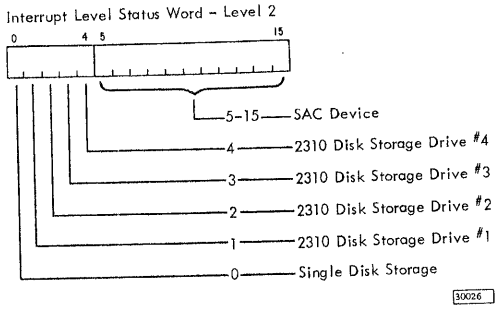
The device or system selected must decode the function field and control the transfer of data to or from core storage.

I/O Control Commands

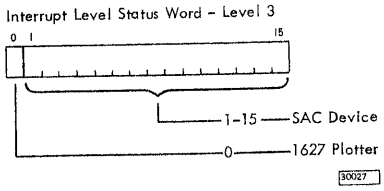
Sense Interrupt (011)



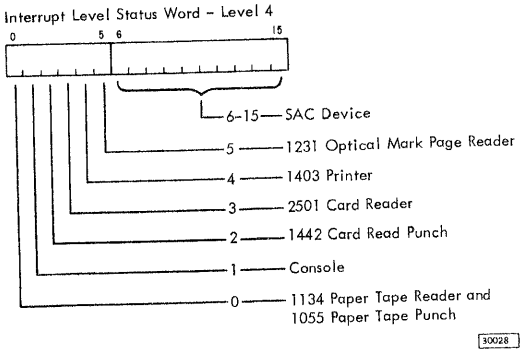
The Sense Interrupt IOCC is placed on the I/O output bus, and the interrupt level being serviced is sent to SAC. The device then sets its assigned bit on the I/O input bus. The CPU program then analyzes the ILSW and branches to the subroutine for the device.



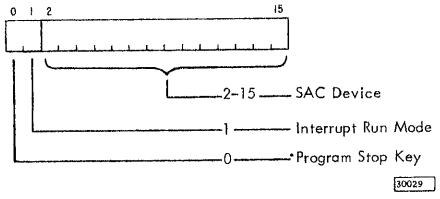
Interrupt Level Status Word - Level 3



Interrupt Level Status Word - Level 4

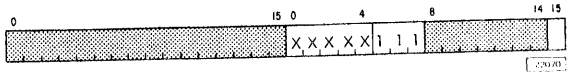


Interrupt Level Status Word - Level 5



The customer assigns interrupt status bits for the devices on the channel in his programs. The devices may bring up an interrupt status bit assigned by the customer. The interrupt status bits may be any bits not used by a basic device.

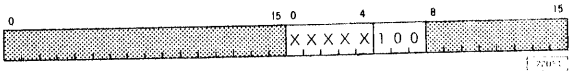
Sense Device (111)



This command sets the IOCC on the I/O output bus. The devices decode the area code and the selected device decodes the sense device function and sets the device status word (DSW) bits on the I/O input bus to read into the accumulator.

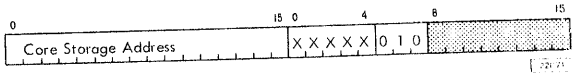
The conditions causing the interrupt are turned off by setting the modifier bit 15 to 1. If the device interrupts on more than one level, the conditions are turned off by modifier bit 15 for the highest level, bit 14 for the next highest level, etc.

Control (100)



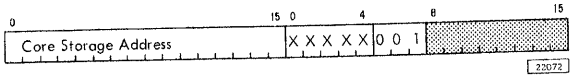
This command sets the IOCC on the I/O output bus. The devices decode the area code. The selected device decodes the control function and sets controls in the device to perform the action specified by the modifier bits (8-15) of EA + 1 or EA (address word). The device and the customer provided programs control the function to be performed.

Read (010)



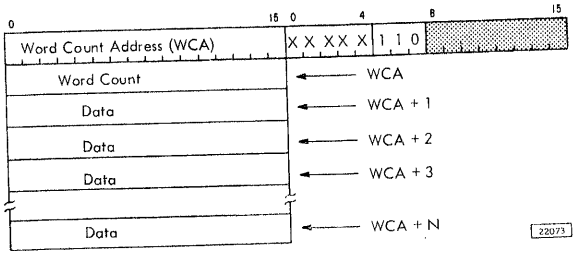
This command sets the IOCC on the I/O output bus. The devices decode the area code. The selected device decodes the read function and sets a single word on the I/O input bus on the E-3 cycle.

Write (001)



This command sets the IOCC on the I/O output bus. The devices decode the area code. The selected device decodes the write function and on the E-3 cycle takes the word from the I/O output bus.

Initiate Read (110)

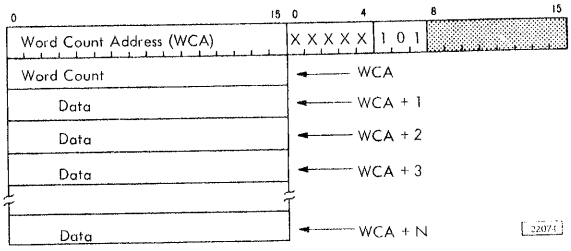


This command sets the IOCC on the I/O output bus. The devices decode the area code. The selected device decodes the initiate read function and sets the controls in the device for cycle steal operation.

The word count address (WCA) is sent to the device.

The first cycle steal cycle is taken and the word count is transferred to the device. The device then controls the transfer of data to the CPU core storage by cycle steal level 1 cycles until the number of words specified by the word count has been transferred.

Initiate Write (101)



This command sets the IOCC on the I/O output bus. The devices decode the area code. The selected device decodes the initiate write function and sets the controls in the device for cycle steal operation.

The word count address (WCA) is sent to the device.

The first cycle steal cycle is taken and the word count is transferred to the device. The device then controls the transfer of data from the CPU core storage to the device by cycle steal level 1 cycles until the number of words specified by the word count has been transferred.

For additional information regarding SAC, refer to "IBM 1131 Central Processing Unit Original Equipment Manufacturers' Information", (A26-3645).

The Synchronous Communications Adapter special feature enables the IBM 1130 Computing System to function as a point-to-point or multipoint data transmission terminal, using either private or commercial common-carrier (switched or non-switched) line transmission facilities. The adapter sends data to or receives data from the line transmission facilities under control of the stored program in the 1130. It operates on an interrupt request basis similar to that used by other input/output devices in the IBM 1130 Computing System.

The Synchronous Communications Adapter (SCA) provides data interchange between remote locations and a central data-processing location. The mode of communication may be either binary synchronous or synchronous transmit-receive and requires its own program. The mode is switch selected by the operator. IBM supplies subroutines to support both modes.

The term "synchronous transmission" is used to describe continuous bit-stream transmission, without start-of-character identification. Thus, synchronous transmission is more efficient than start/stop transmission because fewer control bits are transmitted.

Binary Synchronous Communications (BSC)

The binary synchronous mode of data transmission provides for point-to-point and multipoint operation. The 1130 may be the master station in a communication network or it may serve as a "slave" station to a larger computing system. IBM Programming Systems provides "master" and "slave" station support for point-to-point operation and "slave" station support for multipoint operation.

The capability of BSC mode to operate with any six, seven, or eight-bit level code provides the 1130 with the ability to communicate with a greater variety of devices. It is no longer necessary for a device to adhere to an eight-bit level code in order to communicate with the 1130 system.

Certain factors should be considered in selecting a character set if the user does not use the IBM-supported character sets. The six- and seven-bit codes provide a faster and more efficient type of communication because the data sets are rated in bits per second. Thus, the fewer number of bits to make a character, the more character that may be transmitted to any given segment of time. However, the number of separate characters that can be contained in a code is decreased, proportionately, as the number of bits used to make a character is decreased.

IBM programming systems support for the SCA in the BSC mode includes a subroutine for point-to-point operation and a subroutine for multipoint operation of a slave station. Using these programs, text may be transmitted in either normal text (Extended Binary-Coded-Decimal Interchange Code, System/360 and 1130 internal code) or full-transparent text. Full-transparent text uses EBCDIC communication control characters. In normal text, data may not have the same bit configuration as any control character. In full-transparent text, control character recognition is handled by special procedure, thus making it possible to have data with the same configuration as control characters. Full-transparent text permits unrestricted coding of data within messages, and is useful in transmitting binary data, decimal data, and other data configurations.

A 2701 or 2703 Data Transmission Unit with the Binary Synchronous feature (SDA-2) must be attached to System/360 Models 30, 40, 50, 65 and 75, for communication in the BSC mode.

Synchronous Transmit-Receive (STR)

All synchronous transmit-receive (STR) devices use the four-of-eight line transmission code shown in Figure 52. The STR mode provides only point-to-point communication. It is used to communicate with the IBM 1009 Data Transmission Unit, the IBM 7701 and 7702 Magnetic Tape Transmission Terminals, the IBM 1013 Card Transmission Terminal, the IBM 7710 and 7711 Data Communication Units, and other STR devices.

The SCA provides the 1130 System with the ability to communicate with System/360 Model 20's Communication Adapter (#2073) and other System/360s which have the IBM 2701 Data Transmission Unit attached. Communication with System/360 (other than Model 20) in the STR mode requires a 2701 (with the SDA-1 feature) attached to System/360.

Graphic	4 of 8 Code NXOR 8 4 2 1	Graphic	4 of 8 Code NXOR 8 4 2 1
blank	1 1 1 1 0 0 0 0*	F	0 1 1 0 0 1 1 0
¢	0 1 1 0 1 0 1 0	G	1 0 0 0 0 1 1 1
.	1 0 0 0 1 0 1 1	H	0 1 1 1 1 0 0 0
<	0 1 1 0 1 1 0 0	I	0 1 1 0 1 0 0 1
(	0 1 0 1 0 1 1 0	J	1 1 0 1 0 0 0 1
+	0 0 1 1 0 1 1 0	K	1 1 0 1 0 0 1 0
!**	1 0 0 0 1 1 0 1	L	1 1 0 0 0 0 1 1
&	1 0 0 0 1 1 1 0	M	1 1 0 1 0 1 0 0
!	1 1 0 0 1 0 1 0	N	1 1 0 0 0 1 0 1
$	0 1 0 0 1 0 1 1	O	1 1 0 0 0 1 1 0
*	1 1 0 0 1 1 0 0	P	0 1 0 0 0 1 1 1
)	0 1 0 1 1 1 0 0	Q	1 1 0 1 1 0 0 0
;	0 0 1 1 1 1 0 0	R	1 1 0 0 1 0 0 1
¬	0 1 0 0 1 1 0 1	none#	1 0 1 0 1 0 1 0
-	0 1 0 0 1 1 1 0	S	1 0 1 1 0 0 1 0
/	1 0 1 1 0 0 0 1	T	1 0 1 0 0 0 1 1
,	0 0 1 0 1 0 1 1	U	1 0 1 1 0 1 0 0
%	1 0 1 0 1 1 0 0	V	1 0 1 0 0 1 0 1
_	0 1 0 1 1 0 1 0	W	1 0 1 0 0 1 1 0
>	0 0 1 1 1 0 1 0	X	0 0 1 0 0 1 1 1
?	0 0 1 0 1 1 0 1	Y	1 0 1 1 1 0 0 0
:	0 0 1 0 1 1 1 0*	Z	1 0 1 0 1 0 0 1
#	0 0 0 1 1 0 1 1	0	1 0 0 1 1 0 1 0
@	1 0 0 1 1 1 0 0	1	1 1 1 0 0 0 0 1
'	0 0 0 0 1 1 1 1	2	1 1 1 0 0 0 1 0
=	0 0 0 1 1 1 1 0	3	1 0 0 1 0 0 1 1
"	0 0 0 1 1 1 0 1	4	1 1 1 0 0 1 0 0
A	0 1 1 1 0 0 0 1	5	1 0 0 1 0 1 0 1
B	0 1 1 1 0 0 1 0	6	1 0 0 1 0 1 1 0
C	0 1 1 0 0 0 1 1	7	0 0 0 1 0 1 1 1
D	0 1 1 1 0 1 0 0	8	1 1 1 0 1 0 0 0
E	0 1 1 0 0 1 0 1	9	1 0 0 1 1 0 0 1

*This is correct for System/360 Programs, but is not consistent with certain other STR devices. See the specific device manual.

**Group Mark

#Record Mark

114708

Figure 52. STR 4-of-8 Line Transmission Code

LINE ATTACHMENT

The Synchronous Communications Adapter is attached to either private or commercial line transmission facilities through a common-carrier data set. In the United States the interface for this data set is defined by EIA (Electronic Industries Association) Standard RS-232-B and requires a Western Electric data set model 201A4, 201B2, 202C1, 202D1, or an IBM-approved equivalent. Outside the United States the data set is defined by the CCITT (Consultive Committee on International Telephone and Telegraph) Standard and requires an IBM 3977 Modem or an IBM-approved equivalent.

The SCA can operate in half-duplex mode using either two-wire or four-wire line transmission facilities. Data rates, selected by the machine operator, are 600, 1200, 2000, or 2400 baud (bits per second).

Half-Duplex Operation

Half duplex is a mode of operation wherein either terminal can transmit or receive in conjunction with the remote terminal, but neither terminal can transmit and receive data simultaneously. In effect, the operation is quite similar to a normal telephone conversation; that is, one party talks while the other party listens. During the course of the conversation, each party may alternate between talking and listening as often as necessary.

Two-Wire Operation

Synchronous transmit-receive or binary synchronous operation with a two-wire half-duplex transmission system requires a delay of approximately 200 milliseconds when the adapter switches from receiving to transmitting data. This turnaround delay allows the data set and the communication lines to reverse the direction of transmission and line echo to settle. The amount of delay is therefore related to the character of the line and data set. Line turnaround time is controlled by the data set and when completed, the data set signals the adapter. The adapter does not transmit until the data set signals the completion of line turnaround.

Four-Wire Operation

When the adapter is connected to a four-wire half-duplex transmission system, line turnaround time is eliminated in the STR mode of operation. The adapter can remain synchronized by transmitting idles (See Figure 54) on one pair of wires while receiving data on the second pair of wires.

In binary synchronous mode of transmission, the four-wire system eliminates line turnaround time, but resynchronization (which requires only the synchronous idle sequence, Figure 57) is always necessary before each transmission. Synchronization is always controlled by the sending device because the receiving device may have last been synchronized with another device and may not be in synchronization with the device that now wishes to communicate.

FUNCTIONAL DESCRIPTION

The entire Synchronous Communications Adapter is contained within the 1131 Central Processing Unit. The adapter functions as an input/output control unit between the 1130 system and the transmission line. All data transfer is character-synchronous. This

means that once an initial synchronous idle character is recognized, each subsequent character is recognized as a group of incoming data bits timed by an internal electronic clock. Continuous regulation of the receiver's clock is provided.

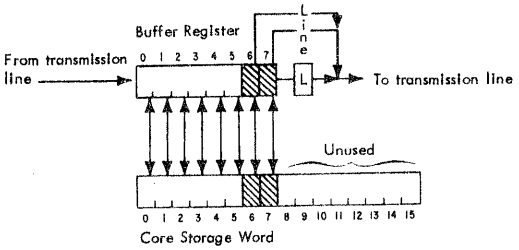
Incoming data from the transmission line is serial by bit and serial by character. As the data comes in, it is stored, one bit at a time, in the receive deserializer. When a complete character has been assembled, the character is transferred into the buffer register. Then the adapter initiates an interrupt request to notify the CPU that a character is ready to be read into core storage. When the interrupt request is serviced, the character is read in parallel into the high-order eight positions of a 16-bit word in core storage.

Outgoing data, from core storage to the transmission line, is taken in parallel from the high-order eight positions of the address location in core storage. The adapter initiates an interrupt request to notify the CPU that the adapter is ready to accept a character from core storage. When the interrupt request is serviced, the character is transferred in parallel to the adapter buffer register. Data from the register is subsequently sent to the transmission line one bit at a time.

Data transfer to or from the transmission line begins with the low order position. Each eight-bit character is located in bit positions 0-7 of a 16-bit core storage location as follows:

bit transfer sequence	bit position in core storage
first	7
second	6
third	5
fourth	4
fifth	3
sixth	2
seventh	1
eighth	0

The seventh and eighth (bit 6 and 7) bits are ignored when using a six-bit level code. The eighth bit (bit 7) is ignored when using a seven-bit level code (Figure 53).



Shaded areas show unused bit positions 6 and 7 for 6-bit and 7-bit codes respectively.

30023A

Figure 53. Communication Data Flow

Timers

There are three electronic timers in the SCA. Each timer is adjustable between 0.28 seconds and three seconds. One timer is set for three seconds and another is set for 1.25 seconds. The third timer is available for other use.

In the STR mode the three-second timer is designated as the receive timer and causes an interrupt and turns on DSW bit 3 when in the receive (monitor) mode to signal the end of the listening period while establishing synchronization. This interrupt also occurs in the transmit mode if a clear to send is not received from the data set within a three-second period. Clear to send is a signal from the data set when it is ready.

The 1.25-second timer is used in the synchronize mode to signal the end of the transmission of idle characters for synchronization in the STR mode. It also causes an interrupt with DSW bit 3 on. This timeout is always coincident with a write response.

The third timer is inhibited in STR operation.

An XIO Control (100) command with bit 10 on inhibits the 1.25- and three-second timers when it is first issued. Issuing the command a second time removes the inhibited status, leaving the timers free to run. This command reverses the status of the timers each time it is issued.

The timers may be restarted at any time by issuing a Sense Device (111) command with bit 14 on if they are not inhibited.

In the BSC mode, the timers are set the same as for STR, but they have a different function. The receive timer (three sec) starts to run when the program enters the receive mode. The program should restart this timer when it detects the synchronous idle sequence (Figure 57). The sending station must transmit this sequence every 1.25 seconds. The three-second timer also interrupts in the transmit mode if a clear to send is not received from the data set within three seconds. In either case DSW bit 3 is turned on.

The 1.25-second timer is used in the synchronize mode to signal the program that it is time to transmit the synchronous idle sequence.

The third timer (designated the program timer) will interrupt in any mode if it is allowed to run.

An XIO Control (100) command with bit 10 on inhibits the 1.25- and three-second timers and starts the program timer. If the program timer is allowed to time out it removes the inhibit condition from the other timers. Issuing another Control command with bit 10 on has the same affect as the timeout. A Sense Device (111) command with bit 14 on will restart any timer that is running.

Synchronous Transmit-Receive (STR) Operation

In order to communicate with a STR device, the STR/BSC switch (see Indicators and Switches) must be placed in the STR position, and the 1130 must contain a program to control the communication. The program must use the four-of-eight code and must use STR line-control conventions. IBM provides a subroutine to control STR communication. This program is described in the manual "IBM 1130 SCA Subroutines" (C26-3706).

STR line-control conventions are described below. Most of the operations described are performed automatically when the IBM subroutine is used. These operations are described here for the user that wishes to write his own routines, and to provide a general understanding of STR communication.

IBM programming systems support for the SCA in the STR mode of operation uses the four-of-eight code. Two types of characters are used:

1. Control characters are used to control line functions; i.e., to acknowledge receipt of a message, to acknowledge synchronization, to signal start of a message or the end of a transmission. The four-of-eight code, used by STR devices, contains special characters used to control line functions.
2. Data characters contain the information to be transferred to or from the adapter. The four-of-eight code contains 64 valid data characters; however, some STR devices do not utilize all of the 64 data characters. The 1130 system can recognize any or all of the 64 data characters as directed by the stored program, but the programmer should determine the character set recognized by the remote STR to avoid sending invalid characters.

Control Operations - STR

The four-of-eight code contains special characters which are reserved for control functions. These control characters and their bit structures are shown in Figure 54. Control sequences are initiated by the 1130 program and are transmitted to the remote terminal as data. The remote terminal then has the responsibility of recognizing the control sequence and responding appropriately.

Control Characters	4 of 8 Code							
	N	X	O	R	8	4	2	1
Idle	0	0	1	1	1	0	0	1
Start of Record 1 or Acknowledge 1 (SOR 1 or ACK 1)	0	1	0	1	0	0	1	1
Start of Record 2 or Acknowledge 2 (SOR 2 or ACK 2)	0	0	1	1	0	0	1	1
Transmit Leader (TL)	0	0	1	1	0	1	0	1
Control Leader (CL)	0	1	0	1	0	1	0	1
End of Transmission (EOT)*	0	1	0	1	1	0	1	0
Inquiry or Error (INQ or ERR)	0	1	0	1	1	0	0	1
Telephone*	0	1	0	1	1	1	0	0
Group Mark	1	0	0	0	1	1	0	1
Longitudinal Redundancy Check (LRC)**	-	-	-	-	-	-	-	-

* Also used as a data character

** This character has a 0 bit in each bit position that contained an even number of 1 bits for that bit position in the data record. If that bit position in the record had an odd number of 1 bits the LRC character ranges from all 0s to all 1s and thus, is not in the 4 of 8 code.

16162A

Figure 54. STR Control Characters

All operations of the adapter are controlled by the 1130 program. The program places the adapter in either the synchronize, transmit, or receive mode. In addition the program must initially store the idle character in the Sync/Idle register and, must generate the longitudinal redundancy check (LRC) character, which is transmitted at the end of each record.

The idle character is a special character which the adapter transmits automatically to the receiving terminal when no other data or control characters have been transferred to the adapter for transmission. This condition occurs during the synchronization mode at the start of each transmission, and when the program responds too slowly to the adapters request for data. The idle character is not included in the LRC character. At least one idle character must be transmitted before each block of records. The adapter does this automatically on line turnaround.

Control characters are used generally in two-character sequences (Figure 55). Each sequence is made up of a leader character and a trailer character. Two of the control characters can be used as leaders of a control sequence. These are the transmit leader (TL) character and the control leader (CL) character. The special characters used as trailers each have two possible meanings depending on whether the TL or the CL character precedes them. For example, the INQ/ERR character is interpreted as an INQ character when preceded by the TL leader and is interpreted as an ERR character when preceded by the CL leader. The end-of-transmission sequence and the telephone sequence consist of one control character followed by one of two data characters. These data characters are interpreted as being part of a control sequence only when they are preceded by the CL character. When not preceded by the CL character, they are interpreted as data.

The inquiry control sequence is used by a terminal when it wishes to transmit a message. The terminal that is in control status may at any time send the inquiry control sequence, which notifies the other terminal of the desire to transmit and asks for permission to do so. If the other terminal is able to receive a message, it acknowledges the inquiry control sequence with an acknowledge sequence.

Control Sequence	Control Character Sequence	
	Leader Character	Trailer Character
End of IDLE (EOI)*	CL	1 IDLE
Inquiry (Synchronized ?)*	TL	INQ
Acknowledge (Synchronized)	CL	ACK 2
Telephone Sequence *	CL	TEL
Acknowledge Telephone *	CL	TEL
Start of Record 1 (SOR 1) 1st or odd numbered record	TL	SOR 1
Start of Record 2 (SOR 2) 2nd or even numbered record	TL	SOR 2
End of Transmittal Record (EOTR)	TL	LRC
Acknowledge Record 1	CL	ACK 1
Acknowledge Record 2	CL	ACK 2
Repeat Last Record (ERROR)	CL	ERR
Intermediate LRC**	GM	LRC
End of Transmission (EOT)*	CL	EOT
Acknowledge EOT *	CL	EOT

*These sequences are always preceded by a 1.25 second transmission of IDLE characters.
** This sequence may be required on some terminals i.e. 1013, 7701, 7702

Figure 55. Control Sequences 16163c

The start-of-record control sequence is transmitted immediately before each block of data. The start-of-record 1 (SOR 1) control sequence is transmitted before the first, third, fifth, etc., record of each message, while the start-of-record-2 (SOR 2) control sequence is transmitted before the second, fourth, sixth, etc., record of each message. This odd-even labeling of each record is used to ensure that no records of a message are lost or duplicated.

The end-of-transmittal record control sequence is sent immediately after each record of a message. The end-of-transmittal record control sequence contains the LRC character, which is used to check the validity of the transmission.

One of the acknowledge control sequences is sent by the receiving terminal after it correctly receives each block of data. This control sequence indicates to the transmitting terminal that it may proceed to send another record. The acknowledge record 1 control sequence should be sent after a record that began with the start-of-record-1 control sequence is received, while the acknowledge record 2 control sequence should be sent after a record that began with the start-of-record-2 control sequence is received. This assures the sender that the receiver has not lost a record. The last acknowledgement is always sent in response.

The repeat last record (error) control sequence is sent by a receiving terminal if it receives a block of data that is in error. This sequence notifies the transmitting terminal that it should repeat the transmission of the last record.

The end-of-transmission control sequence is sent by the transmitting terminal after it has sent the last record of a message. This indicates that the message has been sent completely. A receiving terminal answers the end-of-transmission control sequence by sending back an end-of-transmission control sequence, thereby notifying the transmitting terminal that the receiving terminal has received the full message. After the transmission of these two end-of-transmission control sequences, the two terminals return to synchronize mode of operation and exchange idle characters.

The telephone control sequence can be sent by either terminal and indicates that the terminal operator desires voice communication, via the handset, with the other terminal operator.

Synchronize Mode - STR

The synchronize mode provides a means of synchronizing the transmitting and receiving terminals to ensure the proper recognition of data bits and characters as they are transmitted between terminals. The synchronize mode, sometimes referred to as handshaking, consists of the transmission of a series of characters for 1.25 seconds, followed by a control sequence and then turning around and listening for a similar series of characters from the other terminal for three seconds. The time intervals for transmit (1.25 seconds) and receive (three seconds) are controlled by timers in the adapter. The timers are under control of the program. The character used in the synchronization sequence is called an Idle character.

At the end of the 1.25-second transmission time, the transmitting terminal sends an end-of-idle control sequence – a control leader (CL) followed by an idle character – (Figure 55). This control sequence signals the receiving (remote) terminal to change from receive mode to transmit mode. When the turn-around is completed (200 ms for two-wire-half-duplex) the remote terminal transmits the idle character for 1.25 seconds. At the end of this time the remote terminal sends the end-of-idle sequence. If neither terminal has a message to transmit, the synchronization sequence continues.

Transmit Mode - STR

When a terminal has a message to transmit, that terminal sends 1.25 seconds of idle characters followed by the inquiry sequence. This sequence informs the remote terminal that a message is about to be transmitted. The remote terminal, if it is in synchronization and is ready to receive, sends an acknowledge control sequence. On receipt of the acknowledge sequence, the transmitting terminal transmits its message.

The first two characters of a message are the start-of-record-1 sequence. This sequence is preceded by one or more idle characters. This sequence is followed by the message data characters for this record. Some terminals may use or require an intermediate block check. (This sequence is GM-LRC.) At the end of the record, the end-of-transmittal-record (EOTR) sequence is sent. This sequence consists of a TL character and a longitudinal redundancy check (LRC) character. Two functions are performed by this sequence: it indicates the end of the record, and provides (via the LRC character) the receiving terminal with a method of checking for a complete message. The receiving terminal acknowledges the EOTR by sending the acknowledge 1 or 2 sequence (if LRC compares) or by the error sequence (if LRC does not compare).

Messages which contain more than one record indicate the start of the second record by sending a start-of-record-2 sequence. The start-of-record-1 sequence is used each time an odd-numbered record is transmitted, and the start-of-record-2 sequence is used each time an even numbered record is transmitted. The use of the two different start of record sequences enables detection of lost or duplicated data records from a terminal.

When the receiving terminal has acknowledged the correct receipt of the last record of a message, the transmitting terminal sends the end of transmission sequence. This sequence consists of a CL character and an end-of-transmission (EOT) character. The receiving terminal acknowledges the EOT sequence by returning the same sequence. The terminals, if so programmed, return to the synchronize mode.

Receive Mode - STR

In the receive mode, the adapter accepts data from other line devices and transfers it to the 1130 core storage. Prior to the transfer of data, the transmitting and receiving terminals must be synchronized.

In the receive mode, the adapter compares the incoming data to the character in the idle register. After at least one Idle character has been recognized, the first non-idle character detected and all subsequent characters including idles are transferred into core storage. Idle characters and control sequences are not included in the LRC. When the transmitting terminal signals the end of a record, the 1130 program checks the transmitted LRC character with the one compiled from the received record. If the two LRC characters are the same, the 1130 program generates the appropriate acknowledgement which is then sent from the adapter to the transmitting terminal. If the LRC characters are not the same, the 1130 program responds with an error sequence which is then sent from the adapter to the transmitting terminal. Normally, the 1130 program requests that the previous record be transmitted again. The number of transmission attempts is controlled by the programmer and may vary.

Special Programming

Special programming techniques are required in STR when an 1130 is used to communicate with a hardware device such as a 1013, 1009, or 7702. The special technique is required when either a 201 Data Set or an IBM 3977 Modem is used in a two-wire operation. These data sets do not allow data or control characters to modulate the carrier prior to the clear-to-send signal from the data set. Therefore, no Idles are received from these devices before the Control Leader (CL) or Transmit Leader (TL). Since the 1130 SCA requires at least one recognizable character before interrupting the CPU, the following special technique should be used:

1. If the 1130 is the slave, it will be receiving records. After writing the acknowledgment character (ACK 1, ACK 2, or ERR), the program should load the sync/idle register with the TL. Since the TL is now the recognizable character, it is not loaded into the buffer for the CPU to read. The first character which interrupts the CPU is the trailer. The program must indicate to itself that the TL has already been received. This should be done when the first read interrupt occurs. If the 1130 times out, the remote station may send a message beginning with a TL or it may begin "handshaking" beginning with an idle character. To cope with either possibility, after a time out, the 1130 program loads the sync register with a TL, and if another timeout occurs, the sync register is then loaded with an idle character. Once character phase is reestablished, the alternating of TL or Idle characters ceases.
2. If the 1130 is the master, it is sending records. After writing an INQ, the LRC character of an EOTR, or the last character of an abort sequence (idle), the program should load the sync/idle register with the CL. Since the CL is now the recognizable character, it is not loaded into the buffer for the CPU to read. The first character which interrupts the CPU is again the trailer. The program must indicate to itself that the CL has already been received. This can be done either at the time the sync/idle register is loaded with the CL or when the first read interrupt occurs.

   For both cases (1 and2), the sync/idle register should be reloaded with the idle character prior to each transmission. An idle character should also remain in the sync/idle register after the program writes the idle of an end-of-idle sequence, or the TEL character, or the EOT character.

   Since the above technique works for all data sets and STR devices, it is recommended that it be followed wherever a mixture of data sets and/or STR devices are used.

Binary Synchronous Communications (BSC) Operation

In binary synchronous operation the receiving terminal's ability to understand or interpret the data it receives is the prime consideration in selecting the code to use for communication.

A variety of codes for communication are available. The user may select any code of six, seven, or eight levels. IBM programming systems for the 1130 use the Extended Binary-Coded-Decimal Interchange Code (EBCDIC) communication control characters for all BSC operations. Figure 56 shows the control characters and Figure 57 shows the sequences in which they are used. In full-transparent text, control character recognition should be handled by a special procedure, thus making it possible to have data with the same configuration as control characters. All characters are transferred to core storage in the CPU for program interpretation.

In the selection of a code, care must be taken in selecting the proper SYN character (see Figure 56).

Character	Bit Configuration 0 1 2 3 4 5 6 7	Meaning
SYN	0 0 1 1 0 0 1 0	Synchronous Idle
DLE	0 0 0 1 0 0 0 0	Data Link Escape
ENQ	0 0 1 0 1 1 0 1	Enquiry
SOH	0 0 0 0 0 0 0 1	Start of Heading
STX	0 0 0 0 0 0 1 0	Start of Text
ETB	0 0 1 0 0 1 1 0	End of Transmission Block
ETX	0 0 0 0 0 0 1 1	End of Text
EOT	0 0 1 1 0 1 1 1	End of Transmission
NAK	0 0 1 1 1 1 0 1	Negative Acknowledgement
*ACK 0	0 1 1 1 0 0 0 0	Positive Acknowledgement (even record)
*ACK 1	0 1 1 0 0 0 0 1	Positive Acknowledgement (odd record)
* Control characters when preceded by DLE		

30018

Figure 56. Binary Synchronous EBCDIC Control Characters

Characters	Meaning
ENQ	Enquiry
SOH	Start of Heading
STX	Start of Text
DLE STX	Start of Transparent Text
ETB CRC-16*	End of Block
DLE ETB CRC-16	End of Transparent Block
ETX CRC-16	End of Text
DLE ETX CRC-16	End of Transparent Text
DLE ACK 1	Acknowledgement of Odd Record
DLE ACK 0	Acknowledgement of Even Record
NAK	Negative Acknowledgement
EOT	End of Transmission
DLE EOT	Disconnect Signal
SYN SYN	Synchronous Idle (Normal)
DLE SYN	Synchronous Idle (Transparent Text)

* CRC-16 is a 16-bit cyclic check character accumulated from text and heading data.

30019

Figure 57. Binary Synchronous Control Sequences

If only two characters are used for synchronization, the first bit of the SYN character must be 0. In all cases the bit configuration must be non-repeating; that is, if several SYN characters are in a continuous flow, the bit configuration must be in such a manner that the beginning of each character can be recognized.

Control Operation — Binary Synchronous

The binary synchronous communications control procedures are generally independent of the transmission code. Any code having a fixed number of bits (six, seven, or eight) per character may be used if the ten control characters are set aside and a proper choice is made for the synchronous idle character. The EBCDIC control sequences are presented in this manual (Figure 57).

The control sequences are initiated by the 1130 program and transmitted to the remote terminal as data. The remote terminal then has the responsibility of recognizing the control sequences and responding appropriately.

All operations of the adapter are controlled by the 1130 program. The program places the adapter in either the synchronize (transmit) or the receive mode. In addition the program must initially store the synchronous idle (SYN) character in the sync/idle register. The program also accumulates the block check character (CRC-16), which is transmitted at the end of each record.

In BSC, data may be transmitted in two modes: normal (EBCDIC) text and full-transparent text. In normal text mode, data may not have the same bit configuration as any control character. In full-transparent text, data may contain any bit configuration since control character recognition is handled by a special procedure. Full-transparent text is quite useful in transmitting machine language and other codes that may contain control characters.

In full-transparent mode, the DLE STX sequence is a special sequence that is transmitted prior to transmitting full-transparent text. When a receiving terminal receives this sequence it will stop checking for control characters and treat all subsequent characters as transparent text. The only control character that is recognized is another DLE character. The detection of another DLE character switches the mode back to normal text mode, and the receiving terminal will start checking for control characters. If the next character is DLE or SYN, the receiving program will treat the character as data or a synchronous idle and will return to the transparent mode. Therefore, in full-transparent text mode, all control characters, including SYN, must be preceded by the DLE character to be recognized by the receiving terminal. In full-transparent mode the program must store the DLE character in the sync/idle register. The SYN character must be stored after leaving full-transparent text mode.

Line Turn-Around

When a terminal wishes to transmit, it sends two SYN characters followed by the ENQ character. Then the terminal goes to the receive mode and waits for an acknowledgment from the receiving terminal. The receiving terminal detects the ENQ character as a request from the transmitting terminal and goes to the transmit mode and replies with a positive acknowledgment (ACK0) if it is ready to receive. When the transmitting terminal receives the positive acknowledgment, it may start to transmit its record. If the receiving terminal is not ready to receive, it should respond with the NAK character (negative acknowledgment). If the terminal is unable to respond, the transmitting terminal will time out in three seconds.

Several of the control characters when detected by the program should cause line turnaround; that is, the transmitting terminal switches to the receive mode and the receiving terminal switches to the synchronize (transmit) mode.

The end-of-block and the end-of-text (ETB and ETX) characters, (if not followed by the block check character (CRC-16)), also cause line turnaround. IBM subroutines always use the block check character. The acknowledgments alternate: ACK1 for the first record and all succeeding odd records, and ACK0 for the second record and all succeeding even records. If the block check character is used, the line turnaround follows it. When a station is through transmitting, it may relinquish its right to transmit by sending the end-of-transmission (EOT) character. The EOT character does not require an acknowledgment. The right to transmit reverts back to the master station or to contention if a master station is not designated.

Multi-Point Operation

In multi-point, centralized operation, IBM programming systems include subroutines that permit the 1130 to operate only as a slave station. Programs to support non-centralized operation must be supplied by the user. A slave station is one that may respond to a call from the control (master) station, but cannot initiate the call. Initialization is performed when the control station sends polling or selection addresses. A particular polling address gives a unique station on the line an opportunity to transmit to the control station. The polled station responds with a positive response (data transmission) or a negative response (EOT). Selection addresses are used to request a particular station to receive data transmission. A selected station responds with its status, ready to receive (ACK0) or not ready to receive (NAK).

In non-centralized operation, the operation is similar to centralized operation except the selected station (after being polled) must respond with its address and the address of the station to which it wishes to transmit. The selected station must reply with its address and a positive acknowledgment if it is ready to receive or a negative acknowledgment if it cannot receive.

Receive Mode-Binary Synchronous

In the receive mode, the adapter accepts data from the transmission line and transfers it to the 1130 core storage. Prior to the transfer of data, the adapter must be synchronized with the transmitting terminal. An Initiate Read command (110) with all modifier bits (8-15) set to zero places the adapter in the receive mode.

In the receive mode, the adapter compares the incoming data to the character in the sync/idle register. After at least two SYN characters have been recognized, the first non-SYN character detected and all subsequent characters including SYN characters are transferred to core storage. The receive mode is terminated by the program when it detects a valid turnaround sequence. If a receive timeout occurs, an End Operation command should be used to reset the clock. The slave should issue an Initiate Read command. The next Initiate Read or Initiate Write command to the adapter should follow immediately. A master should issue an Initiate Write command to send ENQ.

Data Transmission-Binary Synchronous

All messages are preceded by a minimum of two SYN characters. The two SYN characters are required when the 1130 is the receiving terminal.

When a terminal has a message to transmit, that terminal sends the Synchronous Idle sequence followed by the Enquiry control character. Some stations may require up to six SYN characters for initial synchronization if data set clocking is not used. The enquiry character informs the remote terminal that a message is about to be transmitted. The remote terminal, if it is ready, synchronizes on the SYN characters and acknowledges by sending the acknowledge control sequence (DLE ACK0). Upon receipt of the acknowledge control sequence, the transmitting terminal transmits its message. The entire message including control characters and check characters are generated, and transmitted from core storage, under control of the stored program in the CPU.

Synchronize Mode-Binary Synchronous

The synchronize mode in binary synchronous communication is a transmit mode which allows a timeout to occur if the transmission is longer than 1.25 seconds. The program must insert the synchronous idle sequence after this timeout to ensure that the receiving terminal remains synchronized. Data transmission may continue after the synchronous idle sequence. The receiving terminal will time out if it does not receive the synchronous idle sequence within three seconds. A Control command (100) with bit 11 set to 1 places the adapter in the synchronize mode.

Transmit Mode-Binary Synchronous

The transmit mode may be used in binary synchronous operations in lieu of the synchronize mode where a timeout is not required or desired. An Initiate Write command (101) with bit 9 set to 0 places the adapter in the transmit mode.

PROGRAMMING

All adapter operations are programmed using the 1130 XIO instruction (see Execute I/O description in this manual). The effective address position of the XIO instruction specifies the address of the two-word IOCC which is required for the desired operation.

The adapter interrupts the 1130 system program on interrupt level 1. Bit position 1 of this interrupt level status word (ILSW) indicates that the interrupting device is the adapter. The program then senses the device status word (DSW).

The DSW is generated by the adapter to indicate the cause of the interrupt (Figure 58).

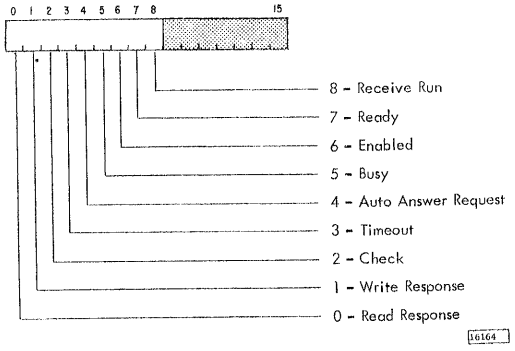


Figure 58. Device Status Word

The DSW bit positions indicate the following conditions:

Bit 0 - The adapter is in receive mode (or diagnostic mode) and the buffer register in the adapter contains a data character which should be transferred into the 1130 core storage.

Bit 1 - The adapter is in transmit mode (or diagnostic mode) and requires a data character from the 1130 core storage for transmission.

Bit 2 - This bit indicates an error condition, data overrun, or character gap.

Data overrun indicates that a character was still in the buffer when another character came, either from the transmission line (receive mode) or from core storage (transmit mode). This condition results in a loss of data. In the transmit mode, data overrun is the result of a program sending another character to the adapter without an interrupt request from the adapter. In the receive mode, this condition is the result of a program operating too slow; that is, a character is received from the transmission line before the preceding character has been transferred to core storage.

Character gap indicates that the data characters are being received by the buffer too slowly for correct adapter operation. In the transmit mode, the program is operating too slow. (Note: The adapter automatically inserts the character in the sync/idle register.) In the receive mode, the program requested another character from the adapter without an interrupt request from the adapter.

Bit 3 - In STR this bit indicates the end of the 1.25-second timeout for transmission of idle characters, or the end of the three second listening time for synchronization. In BSC this bit indicates it is time to insert the synchronous idle sequence in synchronize mode, or a receive timeout occurred in the receive mode.

Bit 4 - This bit indicates that the data-set phone is ringing.

Bit 5 - This bit indicates that the adapter is in either the receive or transmit mode.

Bit 6 - This bit indicates that the adapter has been enabled for an auto answer request interrupt. (See Bit 4.)

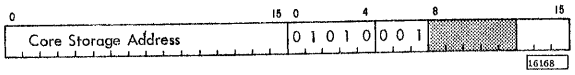
Bit 7 - This bit indicates that the data set is connected and ready to receive, synchronize, or transmit data.

Bit 8 - This bit is used with two-wire half-duplex STR systems only. In indicates that the adapter is in the "slave" mode. In slave mode the adapter transmits the normal acknowledge responses but does not transmit data records. Receive and transmit clocks are tied together.

I/O CONTROL COMMANDS (IOCC)

The adapter is addressed by the five-bit (bits 0 through 4) device (area) code in the IOCC. This code is ten (decimal).

Write (001)



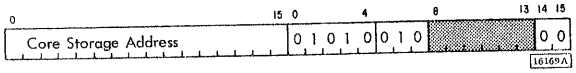
A Write command without a modifier bit instructs the 1130 to transfer the contents of the specified core storage address to the adapter buffer. The adapter then serializes the contents of the buffer register onto the transmission line.

If modifier bit 13 is set, this command is used to set the sync/idle register. The 1130 transfers the data to the sync/idle register in the adapter. The idle character is transmitted during the synchronize mode or when the adapter is in transmit mode and has not received a data character for transmission.

Modifier bit 14 turns off audible alarm trigger in the adapter.

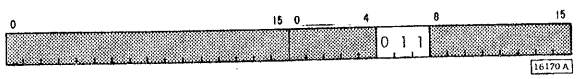
Modifier bit 15 turns on the audible alarm trigger.

Read (010)



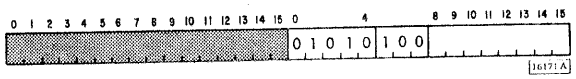
The Read command instructs the 1130 to transfer the contents of the adapter buffer to the core storage location specified in the address portion of the command. Modifier bits 14 and 15 must be 0s in application programs. When on, they are used for reading diagnostic words.

Sense Interrupt (011)



The Sense Interrupt command causes the 1130 to transfer the interrupt level status word (ILSW) of the interrupt being serviced to the accumulator to determine the device causing the interrupt. Bit position 1 of the ILSW for interrupt level 1 indicates the adapter was the interrupting device. The device status word is then examined to determine the cause of the interrupt in the program (Figure 58).

Control (100)



The Control command is always used with a modifier bit. This command causes the adapter to accomplish the functions specified by the modifier.

Modifier bit 8, when set to 1, enables the adapter for Auto Answer operation. Auto Answer allows the adapter to interrupt the 1130 program in response to a telephone ring from the remote terminal.

Modifier bit 9, when set to 1, disables the Auto Answer operation and does not allow a telephone ring from the remote terminal to interrupt the 1130 program.

Modifier bit 10 reverses the status of the timers from run to inhibit or from inhibit to run.

Modifier bit 11 sets the adapter to the synchronize mode. This is used to establish and maintain synchronization in the STR mode without program interruption. Idles are transmitted without the program being interrupted until transmit timeout occures. In binary synchronous mode, modifier bit 11 allows the adapter to transmit in the synchronize mode. Write responses occur normally. A transmission longer than 1.25 seconds causes a timeout interrupt. The program must transmit the synchronous idle sequence before continuing to transmit data. The synchronous idle sequence is the only synchronization necessary in BSC. This usually consists of two sync characters.

The on condition of modifier bit 12 places the adapter in a diagnostic condition. Bit 12 should be off for all application programs. Because of the short time between interrupts in this condition, the diagnostic program should be run alone.

Modifier bit 13 is the End Operation command. This command resets the adapter regardless of the mode of operation. If the adapter is in the transmit mode, the resets is delayed until a character gap of one character is detected. This allows the last character to get through the data set before the adapter is reset. It then resets the adapter and also resets the timers used in the synchronization

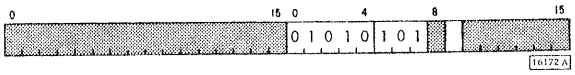
mode and disconnects the adapter from the communication line if a switched network is used. In binary synchronous mode, this command should be issued after a receive timeout to reset the clock.

Modifier bit 14 is used to set the adapter for a six-bit character frame. Setting bit 14 automatically resets the seven-bit character mode.

Modifier bit 15 is used to set the adapter for a seven-bit character frame. Setting bit 15 automatically resets the seven-bit character mode.

Both frame size modes are reset when the adapter leaves both the receive and transmit mode. Thus it becomes necessary to re-enter the proper mode after each line turnaround. Attempting to set both bit 14 and bit 15 with the same instruction is ambiguous and may result in an error.

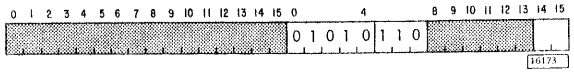
Initiate Write (101)



The Initiate Write command places the adapter in the transmit mode of operation.

Initiate Write with modifier bit 9 on resets all conditions in the adapter.

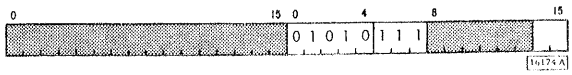
Initiate Read (110)



The Initiate Read command places the adapter in the receive mode of operation.

An Initiate Read command with modifier bit 14 on sets the send-receive run trigger and places the adapter in slave mode operation for STR operation. This mode of operation is used with two-wire half-duplex systems. In the slave mode the adapter should not be programmed to transmit data records to the master. The only transmissions that the adapter will make are the normal responses to the inquiry from the master and the normal acknowledgments. The Start Read command and a modifier bit 15 clears the send-receive run trigger and removes the adapter from slave mode operations. This places the adapter in the master mode, which is used for the transmission of data.

Sense Device (111)



The Sense Device command instructs the 1130 to sense the device status word (DSW). The DSW is generated by the adapter to indicate the cause of the interrupt. The DSW for the adapter is shown in Figure 58.

Sense DSW with modifier bit 14 on will restart the timers. If the synchronous idle sequence is received while in BSC receive mode, the program should restart the timer. If the timer is not reset within three seconds, the adapter will cause a time-out interrupt. Sense DSW with modifier bit 15 on resets the device status word responses.

INDICATORS AND SWITCHES

The console panel for the 1131 CPU with the adapter installed is shown in Figure 59. The non-shaded portion of the figure shows the indicators which relate to the adapter.

(RDY) ready: This indicator lights when the data set is ready.

(ABL) Enabled: This indicator lights when the 1130 program has enabled the adapter to respond to a ring indicator signal from the data set.

(REC) Receive: This indicator lights when the adapter receive trigger is on.

(TSM) Transmit Mode: This indicator lights when the adapter is in the transmit mode.

(BFR) Buffer Loaded: This indicator lights when the buffer contains data.

(CLK) Clock Running: This indicator lights when the receive clock is running.

(DI) Data IN: This indicator lights when the receive data line from the data set is at a zero or space level.

(CP) Character Phase: This indicator lights when the adapter is operating in character phase.

The addition of the adapter to the 1130 system also modifies the switches and lights on the console keyboard and the switch panel below the disk drive as shown in Figures 60 and 61. The shaded portion

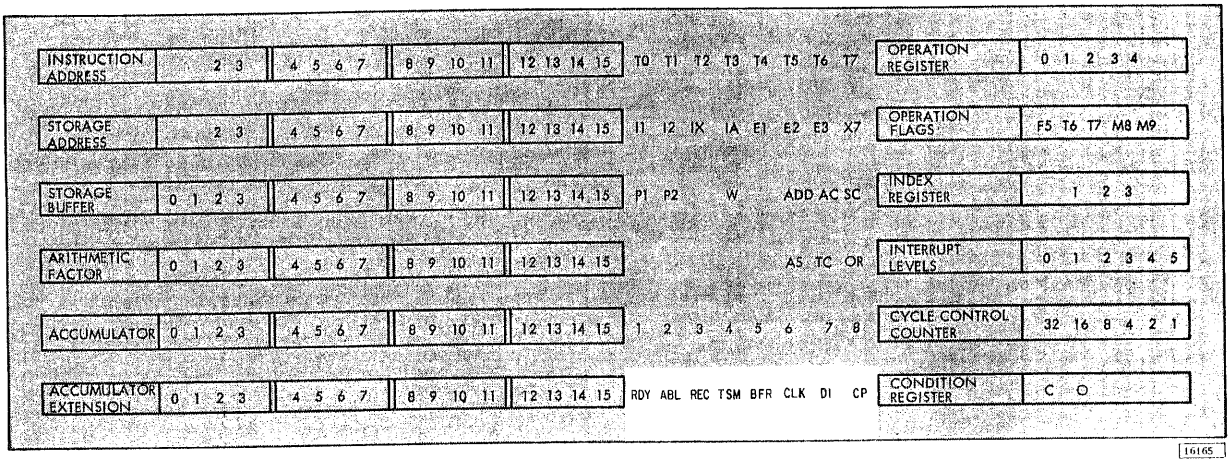


Figure 59. Console Panel

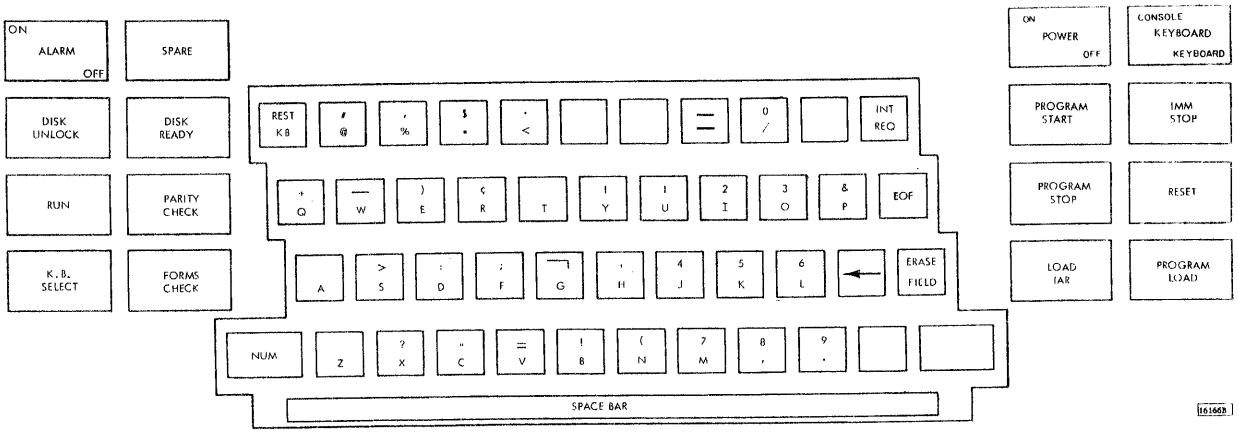


Figure 60. Console Keyboard

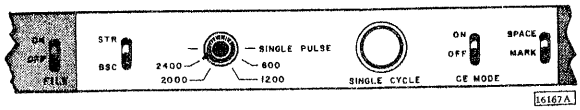


Figure 61. Operator Panel

of Figure 61 are unchanged. The addition to Figure 60 is the Alarm switch. The deletions to the console and keyboard are the Alpha and Numeric lights and the Alpha key. On machines with the keyboard shown in Figure 60, keyboard input will always be alpha (lower case) unless the Numeric key is held depressed (upper case).

STR/BSC: This toggle switch is set to STR or to BSC to select the mode of communication.
Speed Selection: This rotary switch is set to establish the number of bits per second which may be transmitted or received. In actual practice the switch setting is determined by the data set.
Single Cycle: This pushbutton switch is used by the CE to aid in maintaining the adapter.
CE Mode: This switch is used by the CE in maintaining the adapter. This switch must be turned off for normal adapter operations.
Space/Mark: This switch is effective only when the CE mode switch is on. It controls the flow of data to the read deserializer.
Audible Alarm: This switch is located on the console keyboard and provides a means of turning off the adapter alarm in the CPU, should the program not do so for any reason.

TIMING FOR SCA PROGRAMMING

In order to prevent an overrun on receive, a character must be sent from the SCA buffer following a read response interrupt within the period shown in Figure 62. Also to prevent a character gap on transmission, a character must be written to the SCA buffer following a write response within the period shown in Figure 62.

Time Between Characters

Baud Speed \ Char. Size	6 Bit	7 Bit	8 Bit
600	10.0 ms	11.6 ms	13.3 ms
1200	5.0 ms	5.8 ms	6.6 ms
2000	3.0 ms	3.5 ms	4.0 ms
2400	2.5 ms	2.9 ms	3.3 ms

Character Rate

Baud Speed \ Char. Size	6 Bit	7 Bit	8 Bit
600	100 cps	85.7 cps	75 cps
1200	200 cps	171 cps	150 cps
2000	333.3 cps	286 cps	250 cps
2400	400 cps	343 cps	300 cps

30024A

Figure 62. Transmission Timing

NUMBER CONCEPT

The concept of assigning a symbol to represent a value or a quantity has been important to man since the earliest attempts to communicate. As life became more complex, the need for symbols to represent more than one or two and be more precise than "many" became evident. Early counting methods, based on the ten fingers on both hands, evolved into the decimal system, which is in most common use today.

The decimal system is built around the base ten and used the 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9 symbols. Combining these symbols and the place system for their arrangement, any number can be expressed, no matter how large or how small. The value depends on its place in a row of symbols. For example, the symbol 1, by itself, has a place value of 1. Combined with another symbol, as in 21, the 1 symbol still has a place value of 1. Reverse the symbols, however, (12) and the 1 symbol now has a place value of 10.

The placement of symbols is often called "positional notation". The rules of positional notation are generally applicable to all number systems, regardless of the base used.

Number systems most frequently encountered in the use of computers, other than the decimal system, are the binary and hexadecimal number systems.

Binary Number System

Computers function in the binary mode because this is the most simple means of expressing value. There are only two possible states: on or off. Therefore, the binary mode system may also be called a base 2 system. In some computers, the values associated with the binary notation are related directly to the binary number system. This system is not used in all computers, but the method of representing values using this numbering system is useful in learning the general concept of data representation.

The common decimal number system uses ten symbols or digits to represent all quantities, and the place value of the digits signifies units, tens, hundreds, thousands, and so on. The value increases by a factor of (base) 10. Likewise, the binary or base 2 number system uses only two symbols or digits: 0 and 1. The position value of the bit symbols (0 or 1) is based on the progression of powers of 2; units, twos, fours, eights, sixteens, and so on.

Hexadecimal Number System

Binary numbers require about three times as many positions as decimal numbers to express the equivalent number. This is not much of a problem to a computer; however, in talking and writing or in communication with the computer, these binary numbers are bulky. A long string of 1's and 0's cannot be effectively transmitted from one individual to another. Some shorthand method is necessary. The hexadecimal number system fills this need.

Because of the simple relationship of hexadecimal to binary, numbers can be converted from one system to another by inspection. The base of the hexadecimal system is 16. This means there are 16 symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. The letters A, B, C, D, E, and F represent 10, 11, 12, 13, 14, and 15, respectively.

Four binary positions are equivalent to one hexadecimal position because a maximum of 15 can be represented by each hexadecimal position. (F hexadecimal = 1111 binary.)

Refer to the IBM publication "Number Systems" (Form C20-1618) for a more detailed description of binary and hexadecimal numbers and their use in arithmetic.

APPENDIX B. CHARACTER CODES

Ref No.	EBCDIC Binary 0123	EBCDIC Binary 4567	EBCDIC Hex	IBM Card Code Rows 12	11	0	9	8	7-1	IBM Card Code Hex	Graphics and Control Names
0	0000	0000	00	12		0	9	8	1	B030	NUL
1		0001	01	12			9		1	9010	SOH
2		0010	02	12			9		2	8810	STX
3		0011	03	12			9		3	8410	ETX
4		0100	04	12			9		4	8210	PF Punch Off
5		0101	05	12			9		5	8110	HT Horiz.Tab
6		0110	06	12			9		6	8090	LC Lower Case
7		0111	07	12			9		7	8050	DEL Delete
8		1000	08	12			9	8		8030	
9		1001	09	12			9	8	1	9030	
10		1010	0A	12			9	8	2	8830	SMM
11		1011	0B	12			9	8	3	8430	VT
12		1100	0C	12			9	8	4	8230	FF
13		1101	0D	12			9	8	5	8130	CR
14		1110	0E	12			9	8	6	80B0	SO
15		1111	0F	12			9	8	7	8070	SI
16	0001	0000	10	12	11		9	8	1	D030	DLE
17		0001	11		11		9		1	5010	DC1
18		0010	12		11		9		2	4810	DC2
19		0011	13		11		9		3	4410	DC3
20		0100	14		11		9		4	4210	RES Restore
21		0101	15		11		9		5	4110	NL New Line
22		0110	16		11		9		6	4090	BS Backspace
23		0111	17		11		9		7	4050	IDL Idle
24		1000	18		11		9	8		4030	CAN
25		1001	19		11		9	8	1	5030	EM
26		1010	1A		11		9	8	2	4830	CC
27		1011	1B		11		9	8	3	4430	CU1
28		1100	1C		11		9	8	4	4230	FLS
29		1101	1D		11		9	8	5	4130	GS
30		1110	1E		11		9	8	6	40B0	RDS
31		1111	1F		11		9	8	7	4070	US
32	0010	0000	20		11	0	9	8	1	7030	DS
33		0001	21			0	9		1	3010	SOS
34		0010	22			0	9		2	2810	FS
35		0011	23			0	9		3	2410	
36		0100	24			0	9		4	2210	BYP Bypass
37		0101	25			0	9		5	2110	LF Line Feed
38		0110	26			0	9		6	2090	EOB End of Block
39		0111	27			0	9		7	2050	PRE Prefix
40		1000	28			0	9	8		2030	
41		1001	29			0	9	8	1	3030	
42		1010	2A			0	9	8	2	2830	SM
43		1011	2B			0	9	8	3	2430	CU2
44		1100	2C			0	9	8	4	2230	
45		1101	2D			0	9	8	5	2130	ENQ
46		1110	2E			0	9	8	6	20B0	ACK
47		1111	2F			0	9	8	7	2070	BEL
48	0011	0000	30	12	11	0	9	8	1	F030	
49		0001	31				9		1	1010	
50		0010	32				9		2	0810	SYN
51		0011	33				9		3	0410	
52		0100	34				9		4	0210	PN Punch On
53		0101	35				9		5	0110	RS Reader Stop
54		0110	36				9		6	0090	UC Upper Case
55		0111	37				9		7	0050	EOT End of Trans.
56		1000	38				9	8		0030	
57		1001	39				9	8	1	1030	
58		1010	3A				9	8	2	0830	
59		1011	3B				9	8	3	0430	CU3
60		1100	3C				9	8	4	0230	DCA
61		1101	3D				9	8	5	0130	NAK
62		1110	3E				9	8	6	00B0	
63		1111	3F				9	8	7	0070	SUB

Ref No.	EBCDIC Binary 0123	EBCDIC Binary 4567	EBCDIC Hex	IBM Card Code Rows 12	11	0	9	8	7-1	IBM Card Code Hex	Graphics and Control Names
64	0100	0000	40	no punches						0000	(space)
65		0001	41	12			9		1	B010	
66		0010	42	12			9		2	A810	
67		0011	43	12			9		3	A410	
68		0100	44	12			9		4	A210	
69		0101	45	12			9		5	A110	
70		0110	46	12			9		6	A090	
71		0111	47	12			9		7	A050	
72		1000	48	12			9	8		A030	
73		1001	49	12				8	1	9020	
74		1010	4A	12				8	2	8B20	¢
75		1011	4B	12				8	3	8420	. (period)
76		1100	4C	12				8	4	8220	<
77		1101	4D	12				8	5	8120	(
78		1110	4E	12				8	6	80A0	+
79		1111	4F	12				8	7	8060	| (logical OR)
80	0101	0000	50	12						8000	&
81		0001	51	12	11		9		1	D010	
82		0010	52	12	11		9		2	C810	
83		0011	53	12	11		9		3	C410	
84		0100	54	12	11		9		4	C210	
85		0101	55	12	11		9		5	C110	
86		0110	56	12	11		9		6	C090	
87		0111	57	12	11		9		7	C050	
88		1000	58	12	11		9	8		C030	
89		1001	59		11			8	1	5020	
90		1010	5A		11			8	2	4820	!
91		1011	5B		11			8	3	4420	$
92		1100	5C		11			8	4	4220	*
93		1101	5D		11			8	5	4120	)
94		1110	5E		11			8	6	40A0	;
95		1111	5F		11			8	7	4060	¬ (logical NOT)
96	0110	0000	60		11					4000	- (dash)
97		0001	61			0			1	3000	/
98		0010	62		11	0	9		2	6810	
99		0011	63		11	0	9		3	6410	
100		0100	64		11	0	9		4	6210	
101		0101	65		11	0	9		5	6110	
102		0110	66		11	0	9		6	6090	
103		0111	67		11	0	9		7	6050	
104		1000	68		11	0	9	8		6030	
105		1001	69			0		8	1	3020	
106		1010	6A	12	11					C000	
107		1011	6B			0		8	3	2420	, (comma)
108		1100	6C			0		8	4	2220	%
109		1101	6D			0		8	5	2120	_ (underscore)
110		1110	6E			0		8	6	20A0	>
111		1111	6F			0		8	7	2060	?
112	0111	0000	70	12	11	0				E000	
113		0001	71	12	11	0	9		1	F010	
114		0010	72	12	11	0	9		2	E810	
115		0011	73	12	11	0	9		3	E410	
116		0100	74	12	11	0	9		4	E210	
117		0101	75	12	11	0	9		5	E110	
118		0110	76	12	11	0	9		6	E090	
119		0111	77	12	11	0	9		7	E050	
120		1000	78	12	11	0	9	8		E030	
121		1001	79					8	1	1020	
122		1010	7A					8	2	0820	:
123		1011	7B					8	3	0420	#
124		1100	7C					8	4	0220	@
125		1101	7D					8	5	0120	' (apostrophe)
126		1110	7E					8	6	00A0	=
127		1111	7F					8	7	0060	"

Ref No.	EBCDIC Binary 0123	4567	EBCDIC Hex	IBM Card Code Rows 12	11	0	9	8	7-1	IBM Card Code Hex	Graphics and Control Names
128	1000	0000	80	12		0		8	1	B020	
129		0001	81	12		0			1	B000	a
130		0010	82	12		0			2	A800	b
131		0011	83	12		0			3	A400	c
132		0100	84	12		0			4	A200	d
133		0101	85	12		0			5	A100	e
134		0110	86	12		0			6	A080	f
135		0111	87	12		0			7	A040	g
136		1000	88	12		0		8		A020	h
137		1001	89	12		0	9			A010	i
138		1010	8A	12		0		8	2	A820	
139		1011	8B	12		0		8	3	A420	{
140		1100	8C	12		0		8	4	A220	
141		1101	8D	12		0		8	5	A120	
142		1110	8E	12		0		8	6	A0A0	
143		1111	8F	12		0		8	7	A060	
144	1001	0000	90	12	11			8	1	D020	
145		0001	91	12	11				1	D000	j
146		0010	92	12	11				2	C800	k
147		0011	93	12	11				3	C400	l
148		0100	94	12	11				4	C200	m
149		0101	95	12	11				5	C100	n
150		0110	96	12	11				6	C080	o
151		0111	97	12	11				7	C040	p
152		1000	98	12	11			8		C020	q
153		1001	99	12	11		9			C010	r
154		1010	9A	12	11			8	2	C820	
155		1011	9B	12	11			8	3	C420	}
156		1100	9C	12	11			8	4	C220	
157		1101	9D	12	11			8	5	C120	
158		1110	9E	12	11			8	6	C0A0	
159		1111	9F	12	11			8	7	C050	
160	1010	0000	A0		11	0		8	1	7020	
161		0001	A1		11	0			1	7000	
162		0010	A2		11	0			2	6800	s
163		0011	A3		11	0			3	6400	t
164		0100	A4		11	0			4	6200	u
165		0101	A5		11	0			5	6100	v
166		0110	A6		11	0			6	6080	w
167		0111	A7		11	0			7	6040	x
168		1000	A8		11	0		8		6020	y
169		1001	A9		11	0	9			6010	z
170		1010	AA		11	0		8	2	6820	
171		1011	AB		11	0		8	3	6420	
172		1100	AC		11	0		8	4	6220	
173		1101	AD		11	0		8	5	6120	[
174		1110	AE		11	0		8	6	60A0	
175		1111	AF		11	0		8	7	6060	
176	1011	0000	B0	12	11	0		8	1	F020	
177		0001	B1	12	11	0			1	F000	
178		0010	B2	12	11	0			2	E800	
179		0011	B3	12	11	0			3	E400	
180		0100	B4	12	11	0			4	E200	
181		0101	B5	12	11	0			5	E100	
182		0110	B6	12	11	0			6	E080	
183		0111	B7	12	11	0			7	E040	
184		1000	B8	12	11	0		8		E020	
185		1001	B9	12	11	0	9			E010	
186		1010	BA	12	11	0		8	2	E820	
187		1011	BB	12	11	0		8	3	E420	
188		1100	BC	12	11	0		8	4	E220	
189		1101	BD	12	11	0		8	5	E120	]
190		1110	BE	12	11	0		8	6	E0A0	
191		1111	BF	12	11	0		8	7	E060	

Ref No.	EBCDIC Binary 0123	4567	EBCDIC Hex	IBM Card Code Rows 12	11	0	9	8	7-1	IBM Card Code Hex	Graphics and Control Names
192	1100	0000	C0	12		0				A000	(+ zero)
193		0001	C1	12					1	9000	A
194		0010	C2	12					2	8800	B
195		0011	C3	12					3	8400	C
196		0100	C4	12					4	8200	D
197		0101	C5	12					5	8100	E
198		0110	C6	12					6	8080	F
199		0111	C7	12					7	8040	G
200		1000	C8	12				8		8020	H
201		1001	C9	12			9			8010	I
202		1010	CA	12		0	9	8	2	A830	
203		1011	CB	12		0	9	8	3	A430	
204		1100	CC	12		0	9	8	4	A230	
205		1101	CD	12		0	9	8	5	A130	
206		1110	CE	12		0	9	8	6	A0B0	
207		1111	CF	12		0	9	8	7	A070	
208	1101	0000	D0		11	0				6000	(- zero)
209		0001	D1		11				1	5000	J
210		0010	D2		11				2	4800	K
211		0011	D3		11				3	4400	L
212		0100	D4		11				4	4200	M
213		0101	D5		11				5	4100	N
214		0110	D6		11				6	4080	O
215		0111	D7		11				7	4040	P
216		1000	D8		11			8		4020	Q
217		1001	D9		11		9			4010	R
218		1010	DA	12	11		9	8	2	C830	
219		1011	DB	12	11		9	8	3	C430	
220		1100	DC	12	11		9	8	4	C230	
221		1101	DD	12	11		9	8	5	C130	
222		1110	DE	12	11		9	8	6	C0B0	
223		1111	DF	12	11		9	8	7	C070	
224	1110	0000	E0			0		8	2	2820	
225		0001	E1		11	0	9		1	7010	
226		0010	E2			0			2	2800	S
227		0011	E3			0			3	2400	T
228		0100	E4			0			4	2200	U
229		0101	E5			0			5	2100	V
230		0110	E6			0			6	2080	W
231		0111	E7			0			7	2040	X
232		1000	E8			0		8		2020	Y
233		1001	E9			0	9			2010	Z
234		1010	EA		11	0	9	8	2	6830	
235		1011	EB		11	0	9	8	3	6430	
236		1100	EC		11	0	9	8	4	6230	
237		1101	ED		11	0	9	8	5	6130	
238		1110	EE		11	0	9	8	6	60B0	
239		1111	EF		11	0	9	8	7	6070	
240	1111	0000	F0			0				2000	0
241		0001	F1						1	1000	1
242		0010	F2						2	0800	2
243		0011	F3						3	0400	3
244		0100	F4						4	0200	4
245		0101	F5						5	0100	5
246		0110	F6						6	0080	6
247		0111	F7						7	0040	7
248		1000	F8					8		0020	8
249		1001	F9				9			0010	9
250		1010	FA	12	11	0	9	8	2	E830	
251		1011	FB	12	11	0	9	8	3	E430	
252		1100	FC	12	11	0	9	8	4	E230	
253		1101	FD	12	11	0	9	8	5	E130	
254		1110	FE	12	11	0	9	8	6	E0B0	
255		1111	FF	12	11	0	9	8	7	E070	

INDEX

READER'S COMMENT FORM

IBM 1130 Functional Characteristics — Form A26-5881-3

- Your comments, accompanied by answers to the following questions, help us produce better publications for your use. If your answer to a question is "No" or requires qualification, please explain in the space provided below. Comments and suggestions become the property of IBM.

	Yes	No
• Does this publication meet your needs?	☐	☐
• Did you find the material:		
Easy to read and understand?	☐	☐
Organized for convenient use?	☐	☐
Complete?	☐	☐
Well illustrated?	☐	☐
Written for your technical level?	☐	☐

- What is your occupation? _____
- How do you use this publication?

As an introduction to the subject?	☐	As an instructor in a class?	☐
For advanced knowledge of the subject?	☐	As a student in a class?	☐
For information about operating procedures?	☐	As a reference manual?	☐

Other _____

- Please give specific page and line references with your comments when appropriate.

COMMENTS

- Thank you for your cooperation. No postage necessary if mailed in the U.S.A.

fold

fold

FIRST CLASS
PERMIT NO. 2078
SAN JOSE, CALIF.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

POSTAGE WILL BE PAID BY . . .

IBM Corporation
Monterey & Cottle Rds.
San Jose, California
95114

Attention: Product Publications, Dept. 455

fold

fold

Printed in U.S.A. A26-5881-3

IBM®

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]

Printed in U.S.A. A26-5881-3

IBM®

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]