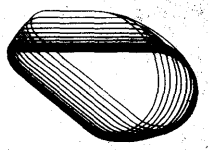


SYSTEM 2400

HARDWARE

PROCESSOR PROGRAMMING IN MACHINE CODE

MOHAWK DATA SCIENCES CORP.

SYSTEM 2400

MOHAWK DATA
SCIENCES CORP.

PROCESSOR PROGRAMMING IN MACHINE CODE

FIRST EDITION

This document contains pertinent descriptive and instructional data on 2400 Processor programming in machine code. Data is based on the utilization of the instruction repertorie and the programmable parameters of individual device controllers to implement data processing/communications oriented programming and functions.

A complete operating manual for an installation should contain a hardware manual for each unit in the installation. The programming information contained herein supplements the programming data supplied with the system.

This publication supersedes the hardware manual *2405 Processor Programming in Machine Code,* Form No. M-1949 and includes:

- Major revisions of the data presented in M-1949 resulting from technical changes/modifications and user comments;
- Specific instructions and data which document the programmable parameters of the SYSTEM 2400 Processors (Models 2404, 2405, and 2408); and
- Device Controller data required to program SYSTEM 2400 peripheral equipment. Change bars indicate additions and corrections; a change bar and an asterisk indicate a deletion.

PALISADE STREET · HERKIMER, NEW YORK 13350

Form No. M-2269-1172

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I
INTRODUCTION

HARDWARE ARCHITECTURE

Figure 1-1 depicts pertinent segments of the 2400 Processors and associated hardware documentation.

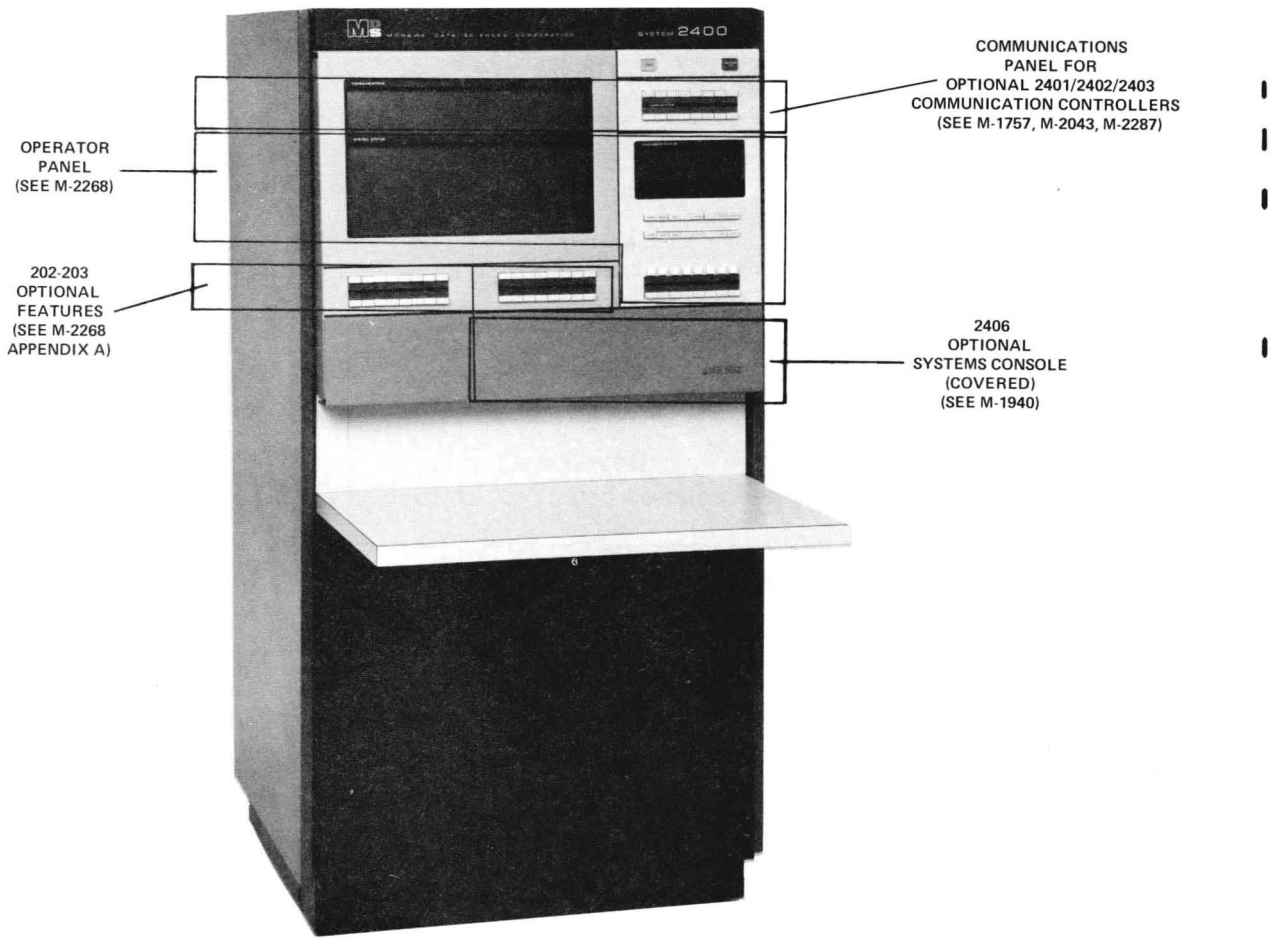


Figure 1. SYSTEM 2400 Processor, Front View

The unique architecture of SYSTEM 2400 hardware provides for:

- concurrent data input and data output operations under hardware control

  with the execution of

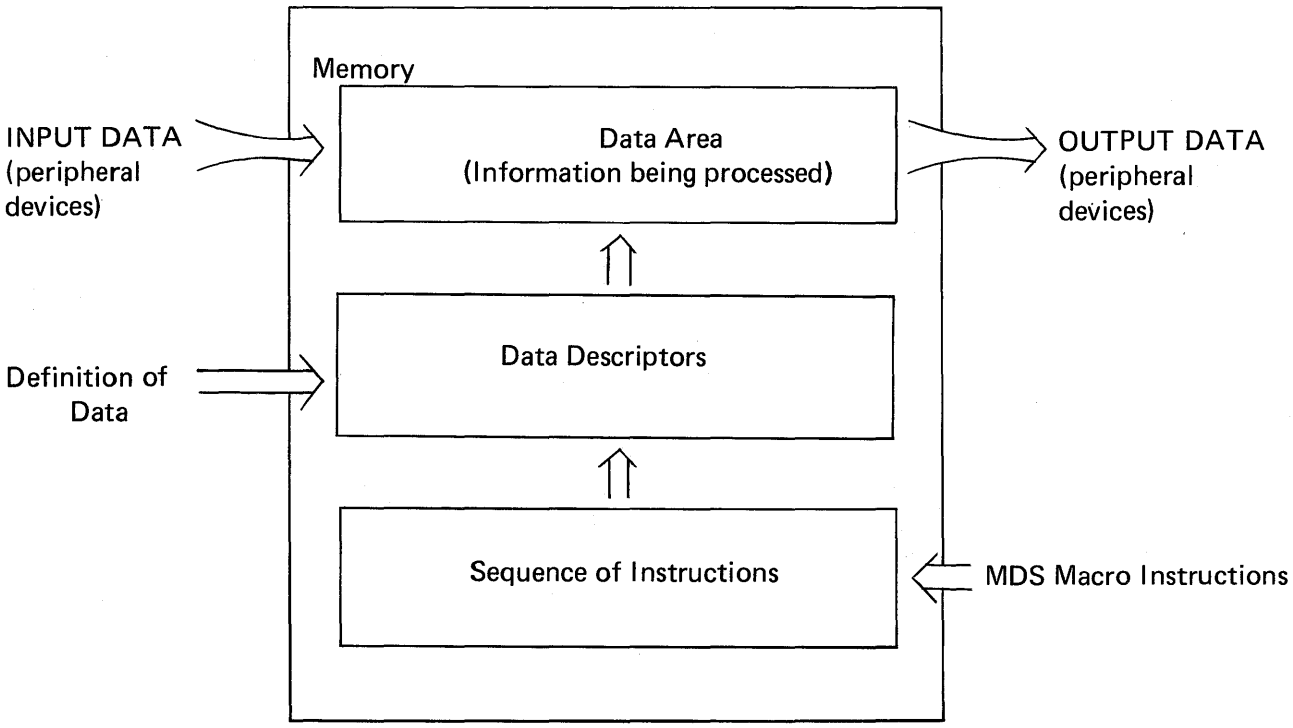
- high-level, variable-address macro-instructions

  which manipulate

- variable-length data units

  which are defined by

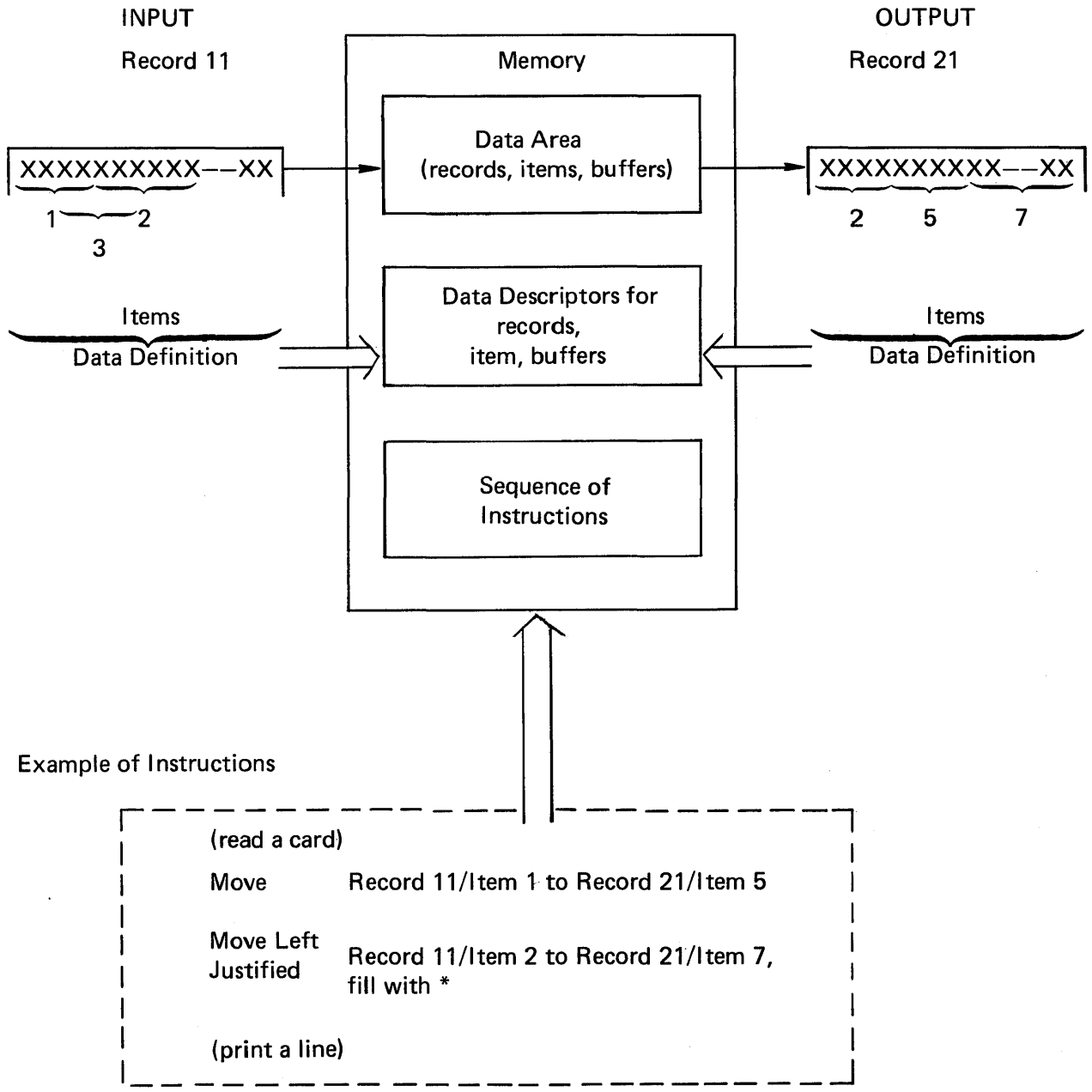
- data descriptors in memory, as shown in the diagram below.



Each instruction addresses data by first referencing the data descriptors. The data descriptors not only give the location of the data in memory, but also give the size and relationship of data units to each other as they appear in memory.

The inclusion of data descriptors in the process of executing macro-instructions provides basic features characteristic of a compiler without the need of a compiler run. The form of data can be modified easily by loading new data descriptors into memory; the instructions which manipulate data thus defined need not be changed.

Structured input and output data is contained in records. Each record is subdivided into items consisting of strings of characters. Records and items are assigned numbers by the programmer, and these numbers are used by the instructions to refer to specific data, as shown in the diagram below. The 2400 Processor can also handle non-structured data.



After the definition of items within each record and the memory positions of all data are loaded into the Data Definition Area, the sequence of instructions is loaded. The process of data input, data manipulation, and data output can then commence.

PRIMARY ELEMENTS

The primary programming elements (see Figure 1-2) that control the processing of data within the 2400 Processor are:

- Program Control Block (PCB)
- Storage Descriptor Area Table (SDAT)
- Records and Buffers
- Three Active Records
- Item Descriptor Tables (IDT)
- Sequence of Instructions

These elements each occupy areas of memory during execution of a program.

PCB – The PCB is fixed in the first 48 locations of memory. The first 16 locations contains the first address (storage descriptor table pointer) of the SDAT and three active registers for program operation in the worker state. The next 4 locations contains the address (program printer) of the sequence of instructions and the real-time clock. The next 12 locations contain interrupt handling instructions, and the last 16 locations of the PCB contains the storage descriptor table pointer and three active registers for program operation in the Exec state.

SDAT – The SDAT can be positioned any place in memory. Its starting address is specified in the PCB. The SDAT is a list of 4-byte entries, each entry corresponding to a record or buffer in memory.

A record descriptor entry, which defines a record, gives the first address of a record (record pointer) and first address of the record's IDT (IDT pointer).

A buffer descriptor entry, which defines a buffer, gives the address limits of the buffer.

Up to 64 records/buffers may be defined in each SDAT.

Records and Buffers – Records and buffers are data areas in memory and contain the data characters manipulated by the Processors. The starting addresses for each record and address limits for each buffer are given in the SDAT.

IDT – For each record in the program, an item descriptor must be defined. The IDT specifies the items within the record. The starting address of each IDT (the IDT pointer) is given in the SDAT.

Active Records – At any given time, three defined records can be accessed by the instructions. The instruction sequence defines which three records are active. Records can be activated and deactivated at will by the program. The descriptors for the three active records are stored in the PCB.

Subsequent sections will delineate the content, purpose, and application of these programming elements and show how they are integrated to form an executable program.

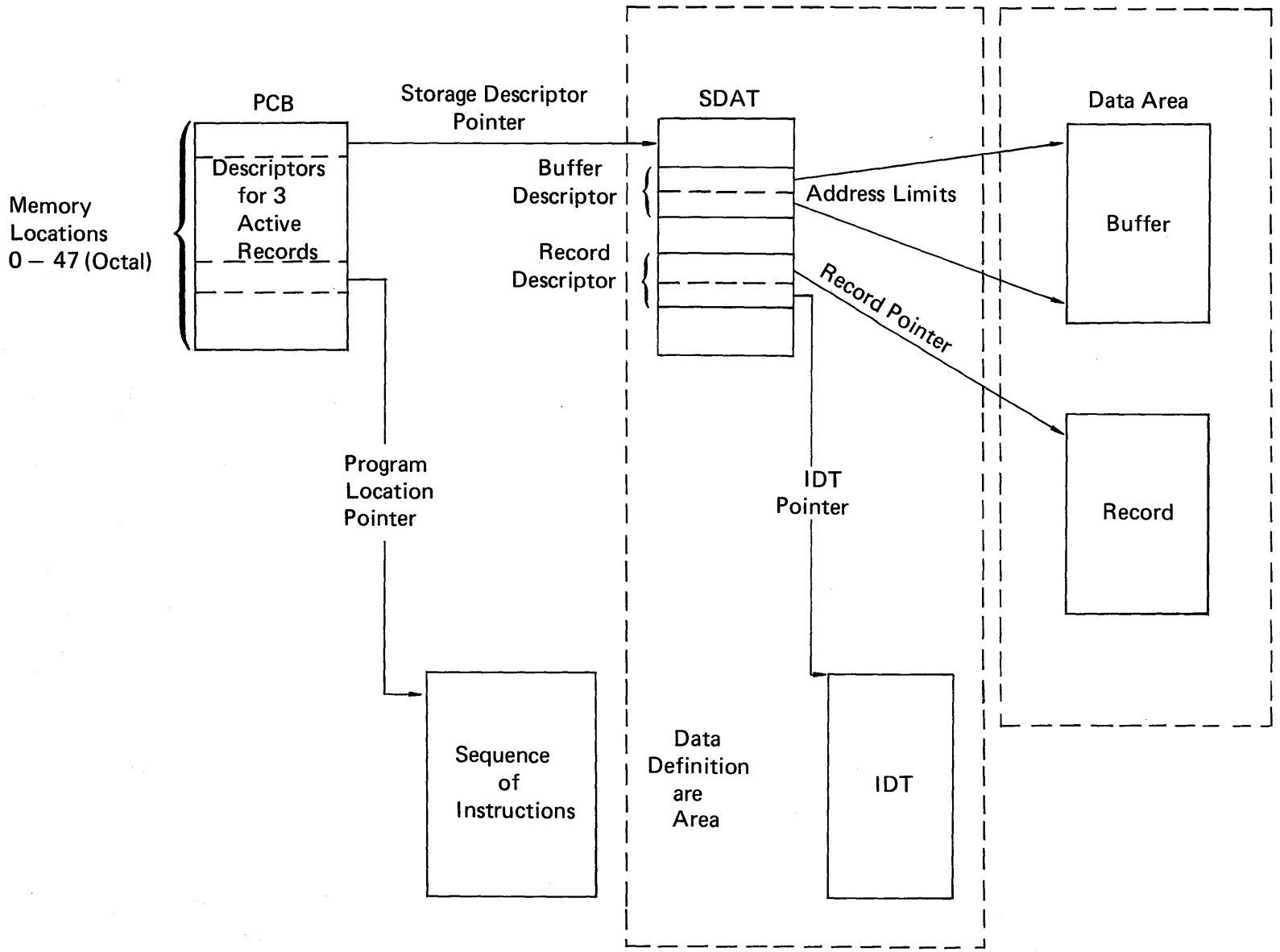


Figure 1-2. Primary Programming Elements of 2400 Processors

MEMORY ADDRESSING

Core memory can range from 8,192 bytes to 65,536 bytes in increments of 4,096 or 8,192 bytes. Each byte is individually accessed in one microsecond with Models 2404 and 2408 and two microseconds with Model 2405 (complete cycle time). Parity (ninth bit used for checking purposes in each byte) and memory power protect (no information is lost if power is lost) are standard features of SYSTEM 2400.

All locations in core memory have a 2-byte address, represented in octal notation, as shown below. In general, the term "location" applies to the area, while the term "address" applies to the numerical identifier or descriptor of a location.

In the above example, the address is represented as $001\text{-}221_8$. Only one address assignment is

- absolute (for the PCB)

while all the others are

- relocatable (such as for the SDAT).

Address notations consist of *two* bytes and are *not* combined to form a total octal number. That is,

- $001\text{-}000_8$ is the 257_{10}th location in core memory
- it is *not* equal to 1000_8
- 1000_8 is equivalent to 512_{10}.

Since the highest octal number that can be represented by an 8-bit configuration is 377_8 (= number 255_{10} and is the 256_{10}th location),

the next address after	$000\text{-}377_8$
is	$001\text{-}000_8$
and the next after	$001\text{-}377_8$
is	$002\text{-}000_8$

and so forth.

Remember that the first memory location starts with zero, so that

address $000\text{-}040_8$ is the 33_{10} rd location
number $000\text{-}040_8$ = number 32_{10}

MACRO-POWER

Each of the instructions has the equivalence in processing power of a subroutine in software, thereby simplifying and condensing the program coding. The instructions are in a variable-length format, with one to four operands, and they perform:

- moving of data
- branching from the program sequence
- comparisons
- tests
- decision functions
- I/O operation and control
- arithmetic calculations
- editing of data
- various miscellaneous functions.

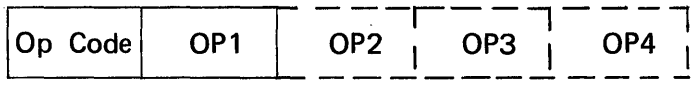
Each instruction is comprised of

- Op Code, for specifying an operation,

  and, except for the Halt instruction,

- 1 to 4 Operands, for specifying a mask or literal and the records, items, buffers, etc., to be operated upon.

The Op Code and Operands are each 1 byte long,



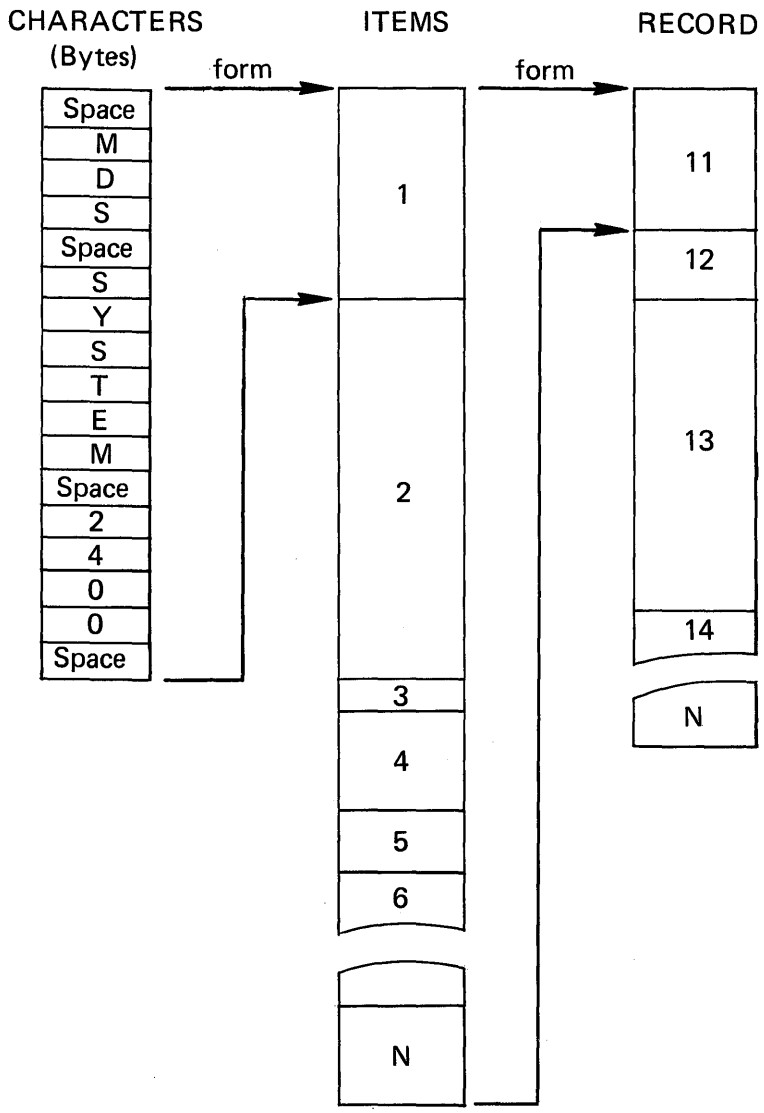
with the Op Code first, followed by the appropriate number of Operands. Op Codes and Operands are expressed in octal notation.

SECTION II

DATA ORGANIZATION

OVERVIEW

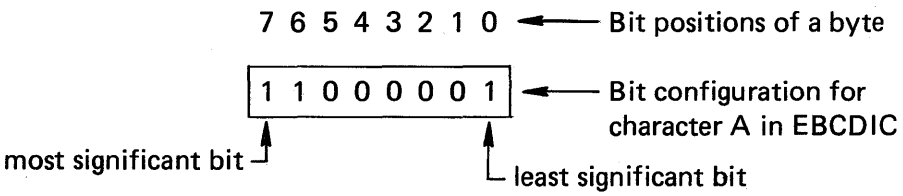
Data organization pertains to the structure and description of information and how it is represented in core memory, as shown in the diagram below. Specifically,



From 1 to 64 RECORDS
    comprise the data to be processed.
From 1 to 64 ITEMS, with a maximum of 256 bytes,
    comprise a RECORD.
From 1 to 256 BYTES
    comprise an ITEM.
8 bits
    comprise a BYTE.

Character and Byte

Eight consecutive bits constitutes a byte, which is the basic unit for representing any one of 256 possible character codes.



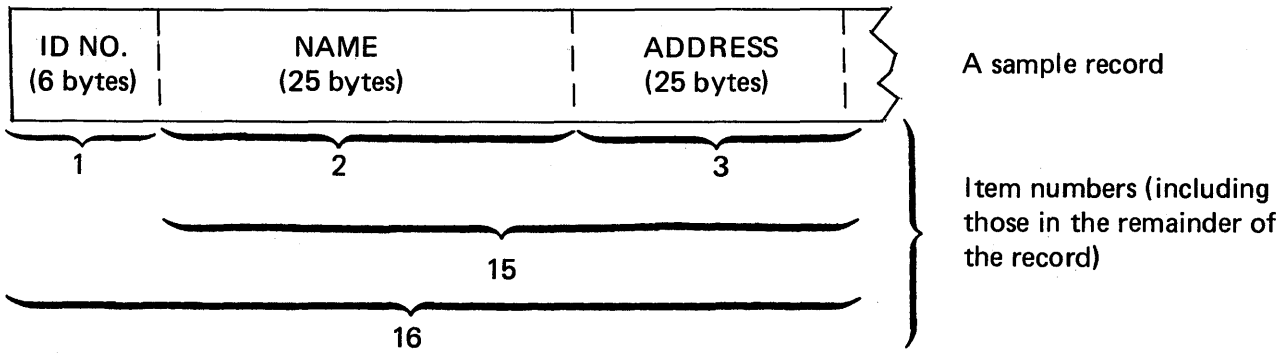
In general, byte and character are synonymous terms; the choice of usage is arbitrary.

Code Representations

Characters may be represented by any of a number of codes, such as EBCDIC or USASCII, or even by codes with less than 8-bit bytes. However, the internal machine code employed by the 2400 Processors is EBCDIC, and any other codes must be converted to EBCDIC (by the Translate Code instruction) if code-sensitive instructions appear in a program. Appendix A contains the code representations for the EBCDIC character set.

Items

Items are related groupings of data strings and are the basic elements comprising a record. Items are variable in length, from 1 to 256 bytes, and may be numbered within positions 000 through 377_8 of the record. Two or more items can also be defined as an item, and a single item can be assigned more than one item number. In other words, items may be defined which overlap other items. For example:



The position (item number) of each item within the record is maintained in the IDT.

Records

Records are comprised of items, varying in number from 1 to 64 items. The maximum record length is 256 bytes. Records contain the actual data to be processed.

Each record is

- delineated on a separate Record Layout or equivalent in terms of its items,

  which serves as the basis for the record to be

- defined in an IDT by the relative positions of items within the record

  and

  assigned a record number in the SDAT.

Thus, each record is uniquely defined for processing by the program.

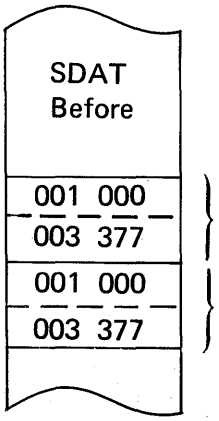
Buffers

A buffer is a temporary storage area in core memory providing transient storage areas for data input/output operations. Areas may also be assigned as working buffers for the temporary storage of data during processing. Data is entered into the input buffer and extracted from the output buffer under the control of the I/O Selector Channels.

Input Buffer – An input buffer is a specified area of core memory that is allocated for the temporary storage of non-structured data from a peripheral device. It can vary in size from one byte up to most of core memory. The input buffer data is not defined by item descriptors, and is therefore considered as non-structured, even though such data may have a specified arrangement of logically related elements in some storage medium outside of the 2400 Processors.

Output Buffer – An output buffer is a specified area of core memory that is allocated for the temporary storage of structured data that is to be transferred to a peripheral device. The output buffer data is received from one or more records in the format (structure) required by the peripheral device. The size of the output buffer is dependent on the characteristics of the peripheral device being used for the output operation.

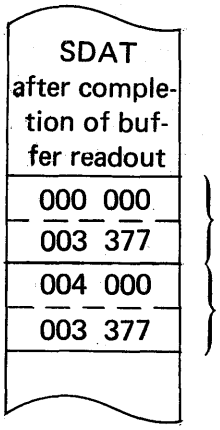
Working Buffer — For blocks of data larger than the maximum record size, such as encountered with high-volume data transfers and when using the Sequential Editing set of instructions, a working buffer serves as a temporary storage area during a processing operation, either for input or output. In general use, an input or an output buffer is used as a working buffer after an input operation or before an output operation. The difference, therefore, is in the SDAT descriptors for a given buffer area. As an example, consider a working buffer used with an input buffer and the SDAT in core with



the input buffer defined by this descriptor (which remains fixed)

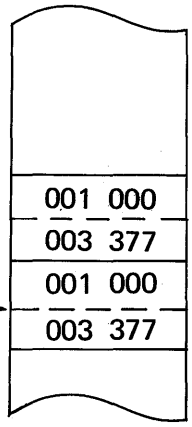
and the working buffer defined by this descriptor.

As data is referenced from the working buffer (using the Sequential Editing instructions), the first half (or buffer first-address pointer) of the working buffer SDAT descriptor serves as a character count register, and



the input buffer descriptor is unchanged

but this descriptor has been changed to the last buffer address + 1. This feature is explained in the Sequential Editing instructions and must be reset (by the Rename instruction) for the next buffer operation to



SECTION III

DATA DESCRIPTION

PERSPECTIVE

To process data within the 2400 Processors, it is necessary to

- Have a precise description of the format in which the data is stored

  and

- know its location in core memory.

This is accomplished by the

- SDAT

  and

- IDT,

which are internally-stored listings of address pointers and item descriptors used by the program to locate designated records and their IDT's, buffers, and items. Entries for each table are prepared on forms and arranged within the overall program sequence.

STORAGE DESCRIPTOR AREA TABLE

The purpose of this table is to define the address pointers for

- each record and its associated IDT

  and

- each buffer.

These are designated in the SDAT as

- record descriptor, consisting of the

  - record pointer
  - IDT pointer

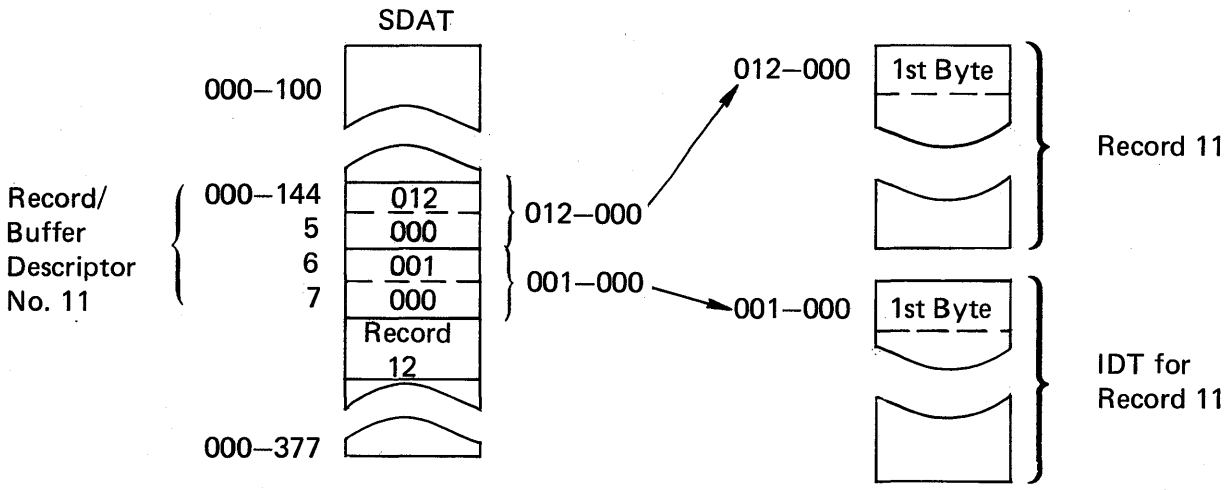
  and

- buffer descriptor, consisting of

  - the first and last addresses of the input and working buffers
  - the first and last + 1 addresses of the output buffer.

Each SDAT descriptor, therefore, consists of four consecutive (address) bytes, and is identified by an SDAT record/buffer number. SDAT record/buffer numbers range from 0 through 77_8, allowing a total of 64 records and/or buffers.

During program execution and when reference is made to an activated record and its items, the program-specified record/buffer descriptor number is correlated with the SDAT so that the address pointers for the record and its IDT can be obtained. This is also true for the buffers, as illustrated in the following diagram.



When the 2400 Processor has obtained the record and IDT pointers from the SDAT descriptor, it can then locate the data item in the given record by means of the item descriptors, as explained below.

ITEM DESCRIPTOR TABLE

The purpose of this table is to define each data item of a record by specifying its relative position within a record. An IDT item descriptor

- is a unique identifier for a record item,

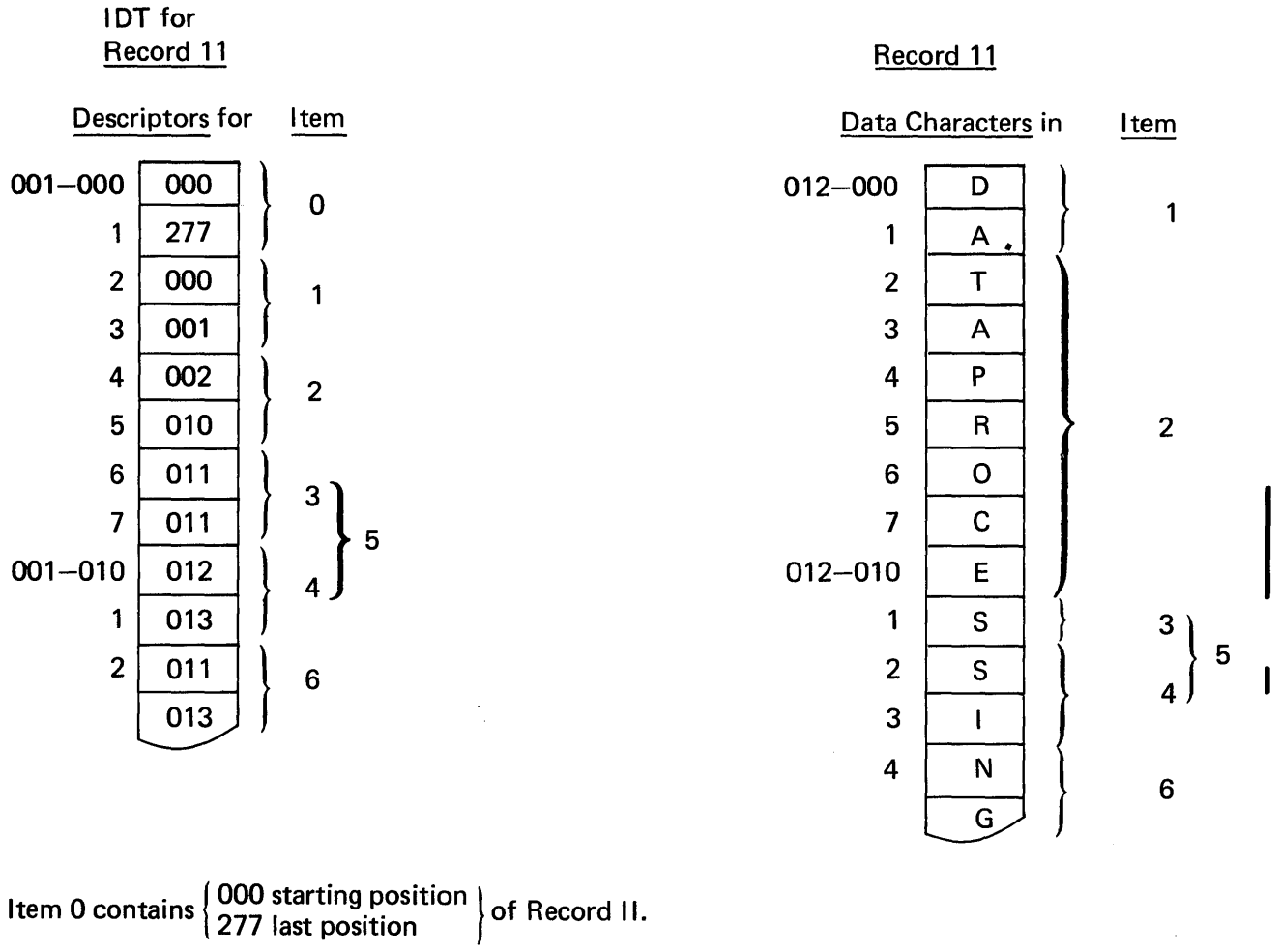
  and consists of

- a 3-digit octal number for specifying the item's starting position

  immediately followed by

- a 3-digit octal number for specifying the item's last position.

Both octal numbers are relative to the 000 starting position of the record. The maximum length of an IDT is 64 item descriptors. IDT descriptor "0" should contain, by convention, the first and last positions of the record.



Item 0 contains { 000 starting position / 277 last position } of Record II.

The above example depicts the core memory layout of a record and its IDT and illustrates their relationship. Locations $001\text{-}000_8$ and $012\text{-}000_8$ were arbitrarily chosen and do not denote any standard locations.

SECTION IV

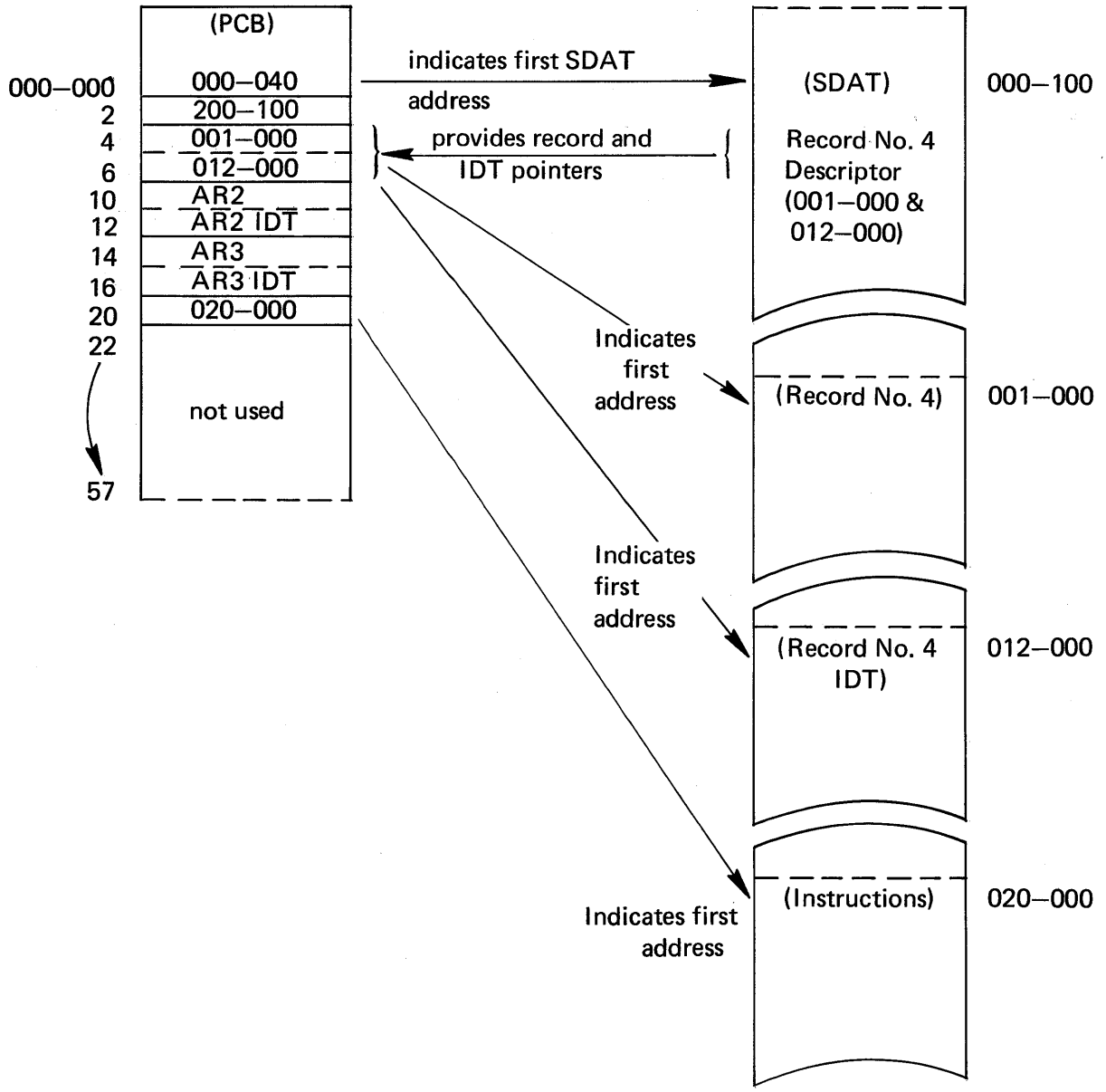
PROGRAM ORGANIZATION

PROGRAM CONTROL BLOCK (PCB)

The PCB provides the linkage between the data to be processed and the set of instructions that direct the processing operations. It consists of the address pointers for

Address (octal)	Content
000,001	SDAT pointer (2405)
002,003	SDAT pointer
004,005	Address of data record, ACTIVE RECORD 1
006,007	Address of item descriptor table, ACTIVE RECORD 1
010,011	Address of data record, ACTIVE RECORD 2
012,013	Address of item descriptor table, ACTIVE RECORD 2
014,015	Address of data record, ACTIVE RECORD 3
016,017	Address of item descriptor table, ACTIVE RECORD 3
020,021	P–BIAS (program start/restart address)
022,023	Real-time clock
024,025	Interrupt
026, 027	Interrupt
028,029	Interrupt
030,031	Interrupt
032,033	Interrupt
034,035	Interrupt
036,037	Interrupt
040,041	SDAT pointer (2405) EXEC STATE
042,043	SDAT pointer EXEC STATE
044,045	ACTIVE RECORD 1
046,047	ACTIVE RECORD 1
050,051	ACTIVE RECORD 2
052,053	ACTIVE RECORD 2
054,055	ACTIVE RECORD 3
056,057	ACTIVE RECORD 3

The PCB is a fixed block of addresses, occupying the first 48 core locations (0 through 57). It is the only core area of absolute addresses used in programming the Processor. This block of address pointers is very important to the program in execution and is accessed frequently.



The SDAT and program location pointers are entered directly when coding and loading the program; however, the active record pointers are loaded from the SDAT during program execution (by the LR1, 2, or 3 instructions) as various records are "activated" (loaded) into the designated PCB area. Optionally, an active record may be loaded from a variable length item of a record rather than from the SDAT. The following instructions are affected: LR1, LR2, LR3, and LSP. Effectively, the instruction is a MOVE, RIGHT-ALIGN, NO-FILL. Note that an IDT only can be loaded by this technique on AR0, AR1, and AR3 at one time. During program execution and when the active record is to be accessed, the Processor uses the AR pointers to locate the designated record and items through the associated IDT. The program location is used primarily by the Branching instructions to determine the absolute branch location.

ACTIVE RECORDS

The AR1, 2, and 3 Record entries refer to the records that can be "activated" by having the corresponding SDAT descriptor entry loaded into the PCB during program execution. They serve as address pointers for the activated records in core memory. For example, an instruction operation that adds an item from record 11 to an item in record 77 and stores the sum in an item in record 14 will require AR1, AR2, and AR3. A data record must be activated before it can be referenced by an instruction.

The AR1, 2, and 3 IDT entries refer to the individual IDT's for the activated records.

PROGRAM LOCATION

The Program Location pointer designates the core-memory address of the first instruction of the program. It is important to note that this pointer is used by *all* of the Branching (GOTO) instructions. Any reference to a "branch to" address must be relative to the address of the first instruction of the program, since the program is relocatable. For example,

- a GOTO instruction refers to a relative address within the program, say $002\text{-}010_8$
- when the program resides in core memory, it is assigned a location starting at, for purposes of illustration, $005\text{-}000_8$
- during execution of the above instruction, the Program Location pointer is added to the GOTO address

$$005\text{-}000_8 + 002\text{-}010_8 = 007\text{-}010_8$$

- giving the absolute branch address in core memory.

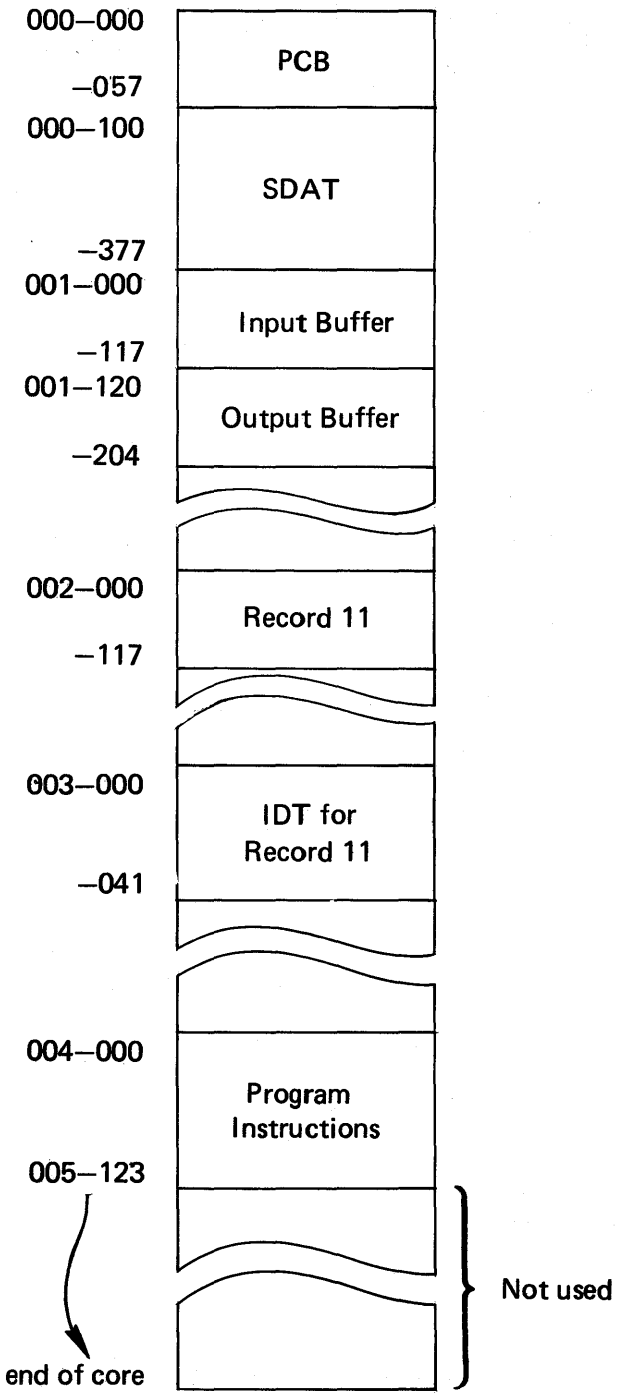
It is also important to note this factor of relocatability when positioning subroutines.

CORE MEMORY

Except for the PCB, core memory assignments are optional. In effect, assignments are made by first-address pointers of blocks of consecutive locations, with

- PCB at absolute locations 000-000 through $000\text{-}057_8$
- SDAT (of which there can be more than one) at any location, assigned as a block of relatively fixed locations
- Records, associated IDT's (as many of both as needed), and buffers at any location, but with each as a block of relatively fixed locations
- Program instructions (there can be more than one program) at any location, assigned as a block of relatively fixed locations.

The following is an example of a core-memory layout of the above program elements.


000–000
–057
PCB
000–100
SDAT
–377
001–000
Input Buffer
–117
001–120
Output Buffer
–204
002–000
Record 11
–117
003–000
IDT for
Record 11
–041
004–000
Program
Instructions
005–123
Not used
end of core

SECTION V

INSTRUCTIONS

FUNCTIONS AND OPERATIONS

Each instruction is classified into one of the following functional categories:

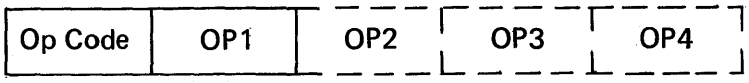
- Data Move – direct transfer of a complete data string from one memory area to another, with or without editing.
- Branching – conditional or unconditional branching from the normal program sequence.
- Compare – comparison between data strings.
- Test – testing for the sign or identity of an item.
- I/O – initiate and control I/O operations.
- General Purpose – perform various operational functions.
- Binary Arithmetic – add and subtract in binary.
- Decimal Arithmetic – add and subtract in decimal coded characters.
- Sequential Editing – manipulate large volumes of non-structured data as they are being transferred between peripheral devices.
- Logical – perform AND, OR, and LRC logical functions by manipulation of OP1, OP2, OP3.
- Interrupt – force the execution of an instruction at fixed memory locations associated with a particular interrupt class.

INSTRUCTION FORMATS

Each instruction is comprised of

- An Op Code, for specifying an operation,
  and, except for the Halt instruction,
- One to four operands, for specifying a mask or literal and the records, items, buffers, etc., to be operated upon.

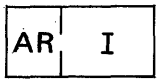
The Op Code and operands are each one byte long



with the Op Code first, followed by the appropriate number of operands.

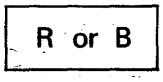
An operand may either refer to an item within a record, a record, or a buffer, or it may specify a literal.

An operand referring to items consists of two parts:

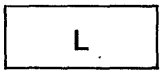


where AR is the active record number (1, 2, or 3) and I is the item number, 00-77.

An operand referring to a record or buffer contains the record or buffer number, 00-57.



A literal is a single character expressed directly in the instruction itself.



The GOTO instruction appears as:

where the address is two bytes.

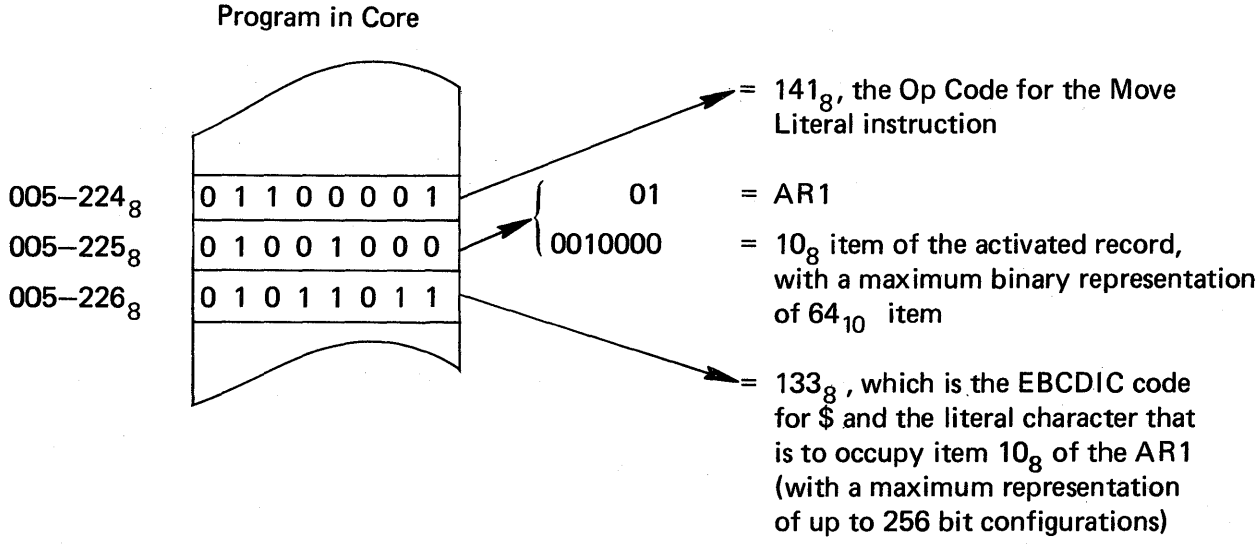
Op Codes and operands are expressed in octal notation. For example, the instruction format

Op Code	OP1	OP2
ML	1/10	$

is

Op Code	OP1	OP2
141	110	133

specifies that the OP2 literal ($) is to be entered into all of the locations of the 10th item of AR1, designated in OP1. It would appear within the program sequence of instructions as three consecutive bytes, expressed as



Data is not referenced directly by the operand. The data descriptors are first accessed to determine the location of the data and its size. For an example, Figure 5-1 presents a summary and overview of declaring a record active and addressing data within that record. The summary proceeds as follows:

(1) Record 11 is declared active by copying its descriptor (Record Pointer and associated IDT Pointer) in the SDAT into AR2 or the PCB, using the | LR2 | 011 | instruction.

(2) An instruction with the operand | 2 | 03 | points to AR2 descriptor, which is in the PCB.

(3) The Record Pointer contains the beginning address of the data, for example – Record 11.

(4) The IDT Pointer contains the beginning address of the IDT for Record 11.

(5) The operand also points to Item 03 in the IDT, which contains the first and last byte positions of Item 3 data within Record 11.

(6) The absolute address limits of the data specified by the operand are calculated by the hardware by adding the first and last position values obtained from the IDT to the Record Pointer address. This data then is the string of characters referenced by the operand.

Note that through the use of the Append or Extract instructions (except Extract Previous Item) a transition of data greater than 256 bytes can be accomplished. The buffer limits to be used are set up (less than, equal to, or greater than 256 bytes) for the instruction. The item is set up for the instruction with the upper limit of the item descriptor set to a value less than the lower limit of the item descriptor. The instruction will terminate upon reaching the end of the buffer, thus allowing a transfer of more than 256 bytes.

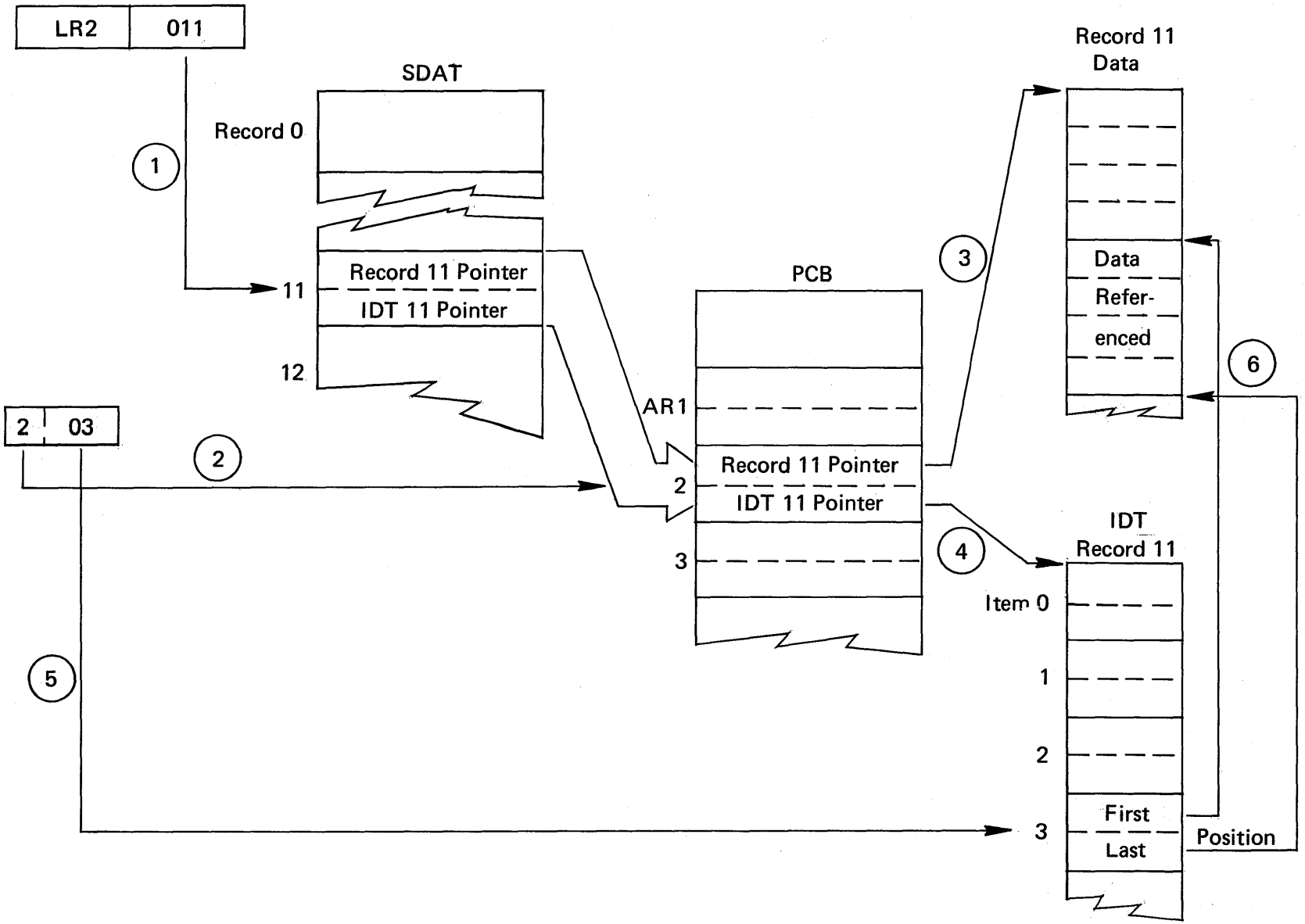


Figure 5-1. Addressing Data in a Record

SECTION VI

INPUT/OUTPUT STRUCTURE

TYPE OF CHANNELS

The SYSTEM 2400 Processors have an extensive input/output structure specifically designed to communicate with a wide variety of peripheral devices. This communication takes place via I/O channels which connect the Processors to the devices. There are two types of channels:

- I/O Selector Channel, the standard channel
- Direct Memory Access Channel, an optional channel.

I/O Selector Channel

Each I/O Selector Channel performs asynchronous, high-speed, byte-oriented data transfers between the Processor core memory and peripheral devices. A Selector Channel transmits data to or from a single I/O device at any given time. A channel can be programmed to be either an input or an output channel. I/O operations on a Selector Channel are overlapped with processing, and all channels can operate concurrently, each at a data rate of (a) 250,000 bytes per second with the 2405 and (b) 500,000 bytes per second with the 2404 and 2408. Data transfer operations are performed under hardware control. Once the I/O process has been initiated by the program on any channel, instruction execution and I/O data transfers are performed concurrently.

One I/O Selector Channel is standard on the basic 2405 Processor with the option of adding up to three more during initial assembly of the unit. However, an assembled unit in the field can be expanded from one to two channels, and expanded from three to four channels. The first two channels are designated 3 and 2, and the last two channels are designated 1 and 0. From one to 16 peripheral devices can be daisy-chained to a channel, at a total remote distance of up to 100 feet.

The 2408 Processor has eight optional I/O Selector Channels. Two are required with a minimal system configuration employing a 2408. Up to six additional channels can be included in the initial assembly or added in the field. The channels are designated as 7, 6, 5, 4, 3, 2, 1, and 0. Off-the-shelf supporting software utilizes a standardized assignment for each I/O Selector Channel and peripheral devices. These are listed in the SYSTEM 2400 Site Survey Report, Form No. M-1737. From one to 16 peripheral devices can be daisy-chained to a channel, at a distance of 100 feet.

The 2404 Processor (KDS system dedicated) has three peripheral attachment points[1] as standard equipage with the option of adding up to five more during initial assembly or as a field expansion. Designation and utilization are the same as cited above for the 2408 with three exceptions: (1) the peripheral devices allowable per attachment point is only one versus 16 for the 2408 and 2405; (2) Key-Display System Keystations (Model 2491) may be remoted from the processor[2] as far as 1000 feet; and (3) only one device controller can be accommodated by each peripheral attachment point.

---

[1] Peripheral attachment point is synonymous with I/O Selector Channel. The former was designated as such to obviate possible confusion over the difference in the allowable number of controllers per I/O Selector channel/ peripheral attachment point.

[2] Keystations are interfaced with the processor via a Model 2428 Multiplexer which is physically located within the Processor cabinetry. The multiplexer accommodates simultaneous operation of multiple keystations by frequency multiplexing input/output data. Refer to Appendix C-19 of this document for programmable parameters of the Model 2428.

DMA CHANNEL

The SYSTEM 2400 Processors encompass Direct Memory Access (DMA) channels to provide a synchronous direct path of communication between an external device and the processor via the memory bus. The external device, by supplying the memory address, controls reading and storing of data into the processor memory.

The number of DMA channels varies between processor models. (Model 2405 has four. Models 2404 and 2408 each have eight.) These are connectable to peripheral devices via an interface module. The module provides both the address of memory and the data to be written or, in the case of a read operation, the address of the data to be accessed. The maximum transfer rate varies between processor models. (Model 2405 has a maximum of 500,000 bytes per second. Models 2404 and 2408 have a maximum of 1 million bytes per second.)

PRIORITIES OF MEMORY ACCESS

In order to preclude competition among the

- I/O Selector Channel, the
- DMA Channel, and

instruction execution control,

for access to core memory, the following priorities govern access during a given memory cycle:

Model 2405 Processor

Input-Output Request	—	I/O Selector Channel 3	Highest Priority
Direct Access Request	—	DMA Channel 3	
Input-Output Request	—	I/O Selector Channel 2	
Direct Access Request	—	DMA Channel 2	
Input-Output Request	—	I/O Selector Channel 1	
Direct Access Request	—	DMA Channel 1	
Input-Output Request	—	I/O Selector Channel 0	
Direct Access Request	—	DMA Channel 0	
Request from Program			Lowest Priority

Models 2404 and 2408 Processors

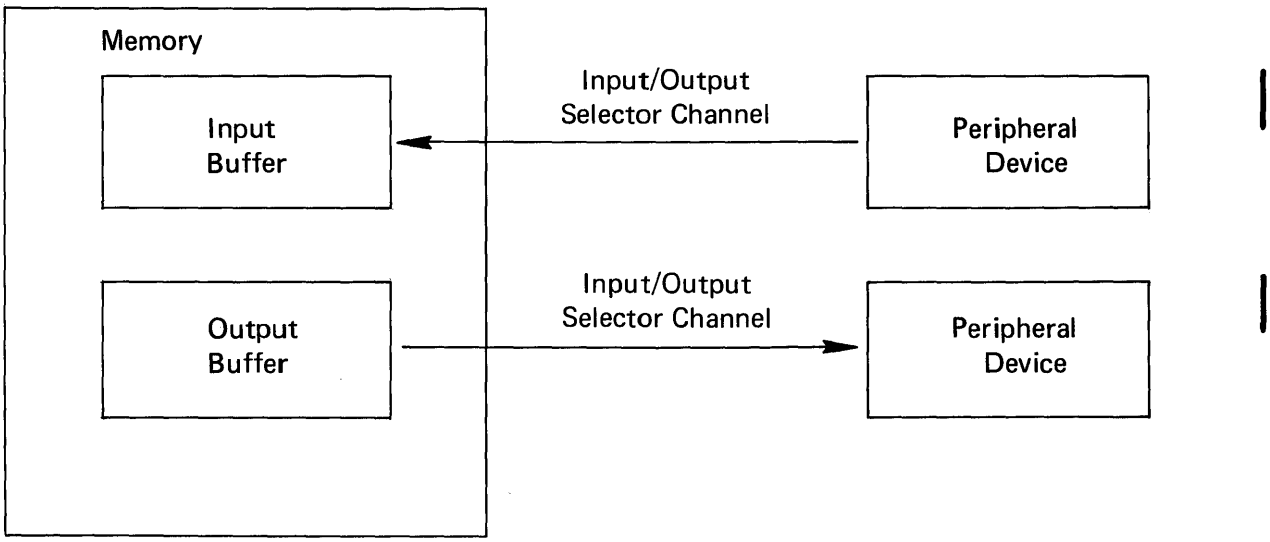
Input-Output Request	–	I/O Selector Channel 7	Highest Priority
Direct Access Request	–	DMA Channel 7	
Input-Output Request	–	I/O Selector Channel 6	
Direct Access Request	–	DMA Channel 6	
Input-Output Request	–	I/O Selector Channel 5	
Direct Access Request.	–	DMA Channel 5	
Input-Output Request	–	I/O Selector Channel 4	
Direct Access Request	–	DMA Channel 4	
Input-Output Request	–	I/O Selector Channel 3	
Direct Access Request	–	DMA Channel 3	
Input-Output Request	–	I/O Selector Channel 2	
Direct Access Request	–	DMA Channel 2	
Input-Output Request	–	I/O Selector Channel 1	
Direct Access Request	–	DMA Channel 1	
Input-Output Request	–	I/O Selector Channel 0	
Direct Access Request	–	DMA Channel 0	
Request from Program	–		Lowest Priority

SECTION VII

PROGRAMMING I/O SELECTOR CHANNELS

I/O INSTRUCTIONS

I/O instructions initiate, control, and direct the I/O operations. These operations take place between the input and output buffers in memory and the I/O channels and their associated peripheral devices. The *SYSTEM 2400 Machine Code and Assembly Language* Manual (Form No. M-1948) gives the detailed descrition of the I/O instructions.



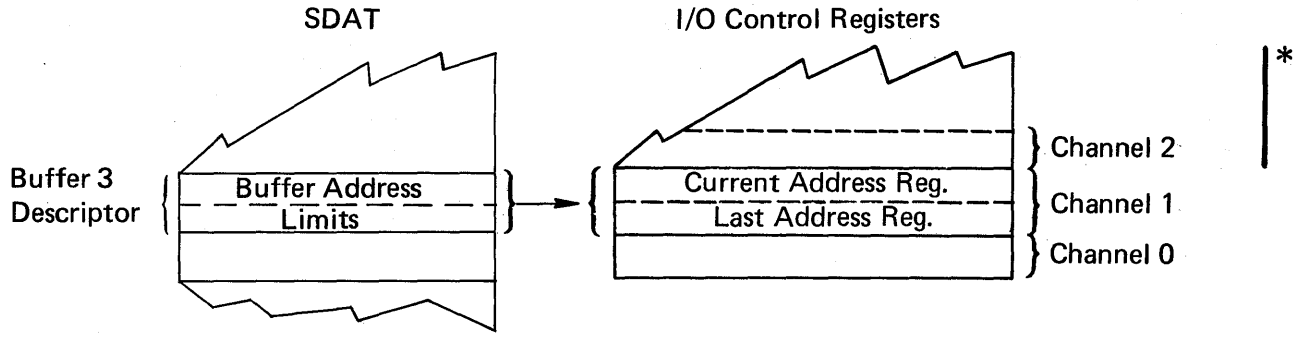
I/O CONTROL REGISTERS

With each I/O Selector Channel, the hardware provides a pair of registers, called I/O Control Registers: The Current Address Register and the Last Address Register. These registers contain memory addresses which maintain the location of the input and output data buffers and control the transfer of data.

The Current Address Register initially contains the first address location of the buffer. Once the I/O operation is underway, it contains the address of the character being transferred.

The I/O Control Registers are loaded by the Initiate Input On Channel and the Initiate Output On Channel instructions. These instructions transfer a buffer descriptor from the SDAT to the specified channel register. Each time a data byte is transferred in or out, the Current Address Register is incremented by one.

For example, an I/O operation, using buffer 3 and channel 1, is initiated by an Initiate I/O instruction (Op Code 114 or 115).



The I/O operation continues concurrently with the execution of succeeding instructions. The Channel Designator is set ACTIVE until all the data is transferred via the peripheral device. The Channel Designator is then set INACTIVE. A special branch instruction is provided to test the status of the Channel Designator.

DATA INPUT

For data input, the I/O Control Registers must contain the first and last address of the input buffer. After each character is stored in the buffer, the Current Address Register is compared to the Last Address Register and then incremented by one. When they are equal, the input operation is terminated and the Channel Designator is set to INACTIVE.

Upon termination, the Current Address Register = last character address + 1, and the Last Address Register = last address.

The contents of the Current Address Register can be stored in memory using the Store Channel Control Register instruction.

The Channel Designator is also set to INACTIVE when a Function Acknowledge signal is received from the peripheral device.

DATA OUTPUT

For data output, the I/O Control Registers must contain the first and last + 1 address of the output buffer. Before each character transfer, the Current Address Register is compared to the Last Address Register and then incremented by one. When they are equal, the operation is terminated and the Channel Designator is set to INACTIVE.

The I/O Control Registers are then

Current Address Register	=	last address + 2
Last Address Register	=	last address + 1.

The Channel Designator is also set to INACTIVE when a Function Acknowledge signal is received from the peripheral device.

The contents of the Current Address Register can be stored in memory using the Store Channel Control Register instruction.

EXTERNAL FUNCTION COMMANDS

Before an I/O instruction is executed, the peripheral device must be conditioned by the program to perform a certain function, such as read, write, rewind, etc. This function command is transmitted to the peripheral device by the External Function On Channel instruction. For every function byte sent to a peripheral device, a status byte is returned to the Processor. The instruction specifies the channel number, the item holding the function command, and the item into which the status byte will be stored. The format of the External Function On Channel instruction is

Op Code	OP1	OP2	OP3
104	AR/1	AR/1	AR/1

OP1 specifies channel number 0, 1, 2, or 3.

OP2 specifies the item that contains the external function code for a peripheral device, with the first byte of the item designating the device address (1 of 16).

OP3 specifies the item that contains the status code from a peripheral device.

The meaning of the status code and the remainder of the external function code are dependent upon the device type of peripheral. If the device status is not received within a specific time period, the status item is filled with null characters and the next instruction is executed.

DIRECT MEMORY ACCESS (DMA) CHANNELS

The DMA Channels are controlled through the use of two special instructions: Special In and Special Out. Each instruction specifies the channel number and an item location which serves as the data input or output area.

APPENDIX A
CHARACTER CODES

This appendix includes tables of various character codes. They are provided herein to give the user quick access to tables normally unavailable without researching through several source documents. Individual tables are provided for —

Table A-1. EBCDIC Character Codes

Char.	7 6 5 4	3 2 1 0	Octal Code	Char.	7 6 5 4	3 2 1 0	Octal Code
A	1 1 0 0	0 0 0 1	301	6	1 1 1 1	0 1 1 0	366
B	1 1 0 0	0 0 1 0	302	7	1 1 1 1	0 1 1 1	367
C	1 1 0 0	0 0 1 1	303	8	1 1 1 1	1 0 0 0	370
D	1 1 0 0	0 1 0 0	304	9	1 1 1 1	1 0 0 1	371
E	1 1 0 0	0 1 0 1	305	Space	0 1 0 0	0 0 0 0	100
F	1 1 0 0	0 1 1 0	306	¢①	0 1 0 0	1 0 1 0	112
G	1 1 0 0	0 1 1 1	307	.	0 1 0 0	1 0 1 1	113
H	1 1 0 0	1 0 0 0	310	<	0 1 0 0	1 1 0 0	114
I	1 1 0 0	1 0 0 1	311	(	0 1 0 0	1 1 0 1	115
J	1 1 0 1	0 0 0 1	321	+	0 1 0 0	1 1 1 0	116
K	1 1 0 1	0 0 1 0	322	|②	0 1 0 0	1 1 1 1	117
L	1 1 0 1	0 0 1 1	323	&	0 1 0 1	0 0 0 0	120
M	1 1 0 1	0 1 0 0	324	!③	0 1 0 1	1 0 1 0	132
N	1 1 0 1	0 1 0 1	325	$	0 1 0 1	1 0 1 1	133
O	1 1 0 1	0 1 1 0	326	*	0 1 0 1	1 1 0 0	134
P	1 1 0 1	0 1 1 1	327	)	0 1 0 1	1 1 0 1	135
Q	1 1 0 1	1 0 0 0	330	;	0 1 0 1	1 1 1 0	136
R	1 1,0 1	1 0 0 1	331	_④	0 1 0 1	1 1 1 1	137
S	1 1 1 0	0 0 1 0	342	-	0 1 1 0	0 0 0 0	140
T	1 1 1 0	0 0 1 1	343	/	0 1 1 0	0 0 0 1	141
U	1 1 1 0	0 1 0 0	344	,	0 1 1 0	1 0 1 1	153
V	1 1 1 0	0 1 0 1	345	%	0 1 1 0	1 1 0 0	154
W	1 1 1 0	0 1 1 0	346	___	0 1 1 0	1 1 0 1	155
X	1 1 1 0	0 1 1 1	347	>	0 1 1 0	1 1 1 0	156
Y	1 1 1 0	1 0 0 0	350	?	0 1 1 0	1 1 1 1	157
Z	1 1 1 0	1 0 0 1	351	:	0 1 1 1	1 0 1 0	172
0	1 1 1 1	0 0 0 0	360	#	0 1 1 1	1 0 1 1	173
1	1 1 1 1	0 0 0 1	361	@	0 1 1 1	1 1 0 0	174
2	1 1 1 1	0 0 1 0	362	'	0 1 1 1	1 1 0 1	175
3	1 1 1 1	0 0 1 1	363	=	0 1 1 1	1 1 1 0	176
4	1 1 1 1	0 1 0 0	364	"	0 1 1 1	1 1 1 1	177
5	1 1 1 1	0 1 0 1	365	‡	1 1 1 0	0 0 0 0	340
				Null	0 0 0 0	0 0 0 0	000
+ Sign	1 1 1 1	X X X X		Substitute Codes			
- Sign	1 1 0 1	X X X X		①] ② ! ③ [④ ^			

Table A-1 (cont'd.). EBCDIC Character Codes

Char.	EBCDIC Code D C B A 8 4 2 1	Octal Code	Char.	EBCDIC Code D C B A 8 4 2 1	Octal Code
A	D C 1	301	6	D C B A 4 2	366
B	D C 2	302	7	D C B A 4 2 1	367
C	D C 2 1	303	8	D C B A 8	370
D	D C 4	304	9	D C B A 8 1	371
E	D C 4 1	305	Space	C	100
F	D C 4 2	306	¢[1]	C 8 2	112
G	D C 4 2 1	307	.	C 8 2 1	113
H	D C 8	310	<	C 8 4	114
I	D C 8 1	311	(	C 8 4 1	115
J	D C A 1	321	+	C 8 4 2	116
K	D C A 2	322	|[2]	C 8 4 2 1	117
L	D C A 2 1	323	&	C A	120
M	D C A 4	324	![3]	C A 8 2	132
N	D C A 4 1	325	$	C A 8 2 1	133
O	D C A 4 2	326	*	C A 8 4	134
P	D C A 4 2 1	327	)	C A 8 4 1	135
Q	D C A 8	330	;	C A 8 4 2	136
R	D C A 8 1	331	¬[4]	C A 8 4 2 1	137
S	D C B 2	342	-	C B	140
T	D C B 2 1	343	/	C B 1	141
U	D C B 4	344	,	C B 8 2 1	153
V	D C B 4 1	345	%	C B 8 4	154
W	D C B 4 2	346	_	C B 8 4 1	155
X	D C B 4 2 1	347	>	C B 8 4 2	156
Y	D C B 8	350	?	C B 8 4 2 1	157
Z	D C B 8 1	351	:	C B A 8 2	172
0	D C B A	360	#	C B A 8 2 1	173
1	D C B A 1	361	@	C B A 8 4	174
2	D C B A 2	362	'	C B A 8 4 1	175
3	D C B A 2 1	363	=	C B A 8 4 2	176
4	D C B A 4	364	"	C B A 8 4 2 1	177
5	D C B A 4 1	365	‡	D C B	340
			Null		000
	Substitute Codes		+ Sign	D C B A X X X X	
	1.] 2. ! 3. [4. ^		- Sign	D C A X X X X	

Table A-2. USASCII Character Codes

Char.	USASCII Code 8765	4321	Octal Code	Char.	USASCII Code 8765	4321	Octal Code
A	0100	0001	101	6	0011	0110	066
B	0100	0010	102	7	0011	0111	067
C	0100	0011	103	8	0011	1000	070
D	0100	0100	104	9	0011	1001	071
E	0100	0101	105	Space	0010	0000	040
F	0100	0110	106	[	0101	1011	133
G	0100	0111	107	.	0010	1110	056
H	0100	1000	110	<	0011	1100	074
I	0100	1001	111	(	0010	1000	050
J	0100	1010	112	+	0010	1011	053
K	0100	1011	113	!	0010	1011	041
L	0100	1100	114	&	0010	0110	046
M	0100	1101	115	]	0101	1101	135
N	0100	1110	116	$	0010	0100	044
O	0100	1111	117	*	0010	1010	052
P	0101	0000	120	)	0010	1001	051
Q	0101	0001	121	;	0011	1011	073
R	0101	0010	122	^	0101	1110	136
S	0101	0011	123	–	0010	1101	055
T	0101	0100	124	/	0010	1111	057
U	0101	0101	125	,	0010	1100	054
V	0101	0110	126	%	0010	0101	045
W	0101	0111	127	_	0101	1111	137
X	0101	1000	130	>	0011	1110	076
Y	0101	1001	131	?	0011	1111	077
Z	0101	1010	132	:	0011	1010	072
0	0011	0000	060	#	0010	0011	043
1	0011	0001	061	@	0100	0000	100
2	0011	0010	062	'	0010	0111	047
3	0011	0011	063	=	0011	0010	075
4	0011	0100	064	"	0010	0010	042
5	0010	0101	065	\	0101	1100	134
				Null	0000	0000	000
				+ Sign	0011	XXXX	
				- Sign	1011	XXXX	

Table A-3. Six-Bit Transcode Character Assignments

CHAR.	SIX-BIT TRANSCODE	CHAR.	SIX-BIT TRANSCODE
SOH	00	-	40
A	01	/	41
B	02	S	42
C	03	T	43
D	04	U	44
E	05	V	45
F	06	W	46
G	07	X	47
H	10	Y	50
I	11	Z	51
STX	12	ESC	52
.	13	,	53
¤	14	%	54
BEL	15	ENQ	55
Sub	16	ETX	56
ETB	17	HT	57
&	20	0	60
J	21	1	61
K	22	2	62
L	23	3	63
M	24	4	64
N	25	5	65
O	26	6	66
P	27	7	67
Q	30	8	70
R	31	9	71
Space	32	SYN	72
$	33	#	73
*	34	@	74
US	35	NAK	75
EOT	36	EM	76
DLE	37	DEL	77

Table A-4. ASR-35 Paper Tape Code

Char.	Binary P 765 4321	Octal Code	Char.	Binary P 765 4321	Octal Code
A	0 100 0001	101	6	0 011 0110	066
B	0 100 0010	102	7	1 011 0111	267
C	1 100 0011	303	8	1 011 1000	270
D	0 100 0100	104	9	0 011 1001	071
E	1 100 0101	305	Space	1 010 0000	240
F	1 100 0110	306	[	1 101 1011	333
G	0 100 0111	107	.	0 010 1110	056
H	0 100 1000	110	<	0 011 1100	074
I	1 100 1001	311	(	0 010 1000	050
J	1 100 1010	312	+	0 010 1011	053
K	0 100 1011	113	!	0 010 0001	041
L	1 100 1100	314	&	1 010 0110	246
M	0 100 1101	115	]	1 101 1101	335
N	0 100 1110	116	$	0 010 0100	044
O	1 100 1111	317	*	1 010 1010	252
P	0 101 0000	120	)	1 010 1001	251
Q	1 101 0001	321	;	1 011 1011	273
R	1 101 0010	322	↑	1 101 1110	336
S	0 101 0011	123	—	0 010 1101	055
T	1 101 0100	324	/	1 010 1111	257
U	0 101 0101	125	,	1 010 1100	254
V	0 101 0110	126	%	1 010 0101	245
W	1 101 0111	327	←	0 101 1111	137
X	1 101 1000	330	>	1 011 1110	276
Y	0 101 1001	131	?	0 011 1111	077
Z	0 101 1010	132	:	0 011 1010	072
0	0 011 0000	060	#	1 010 0011	243
1	1 011 0001	261	@	1 100 0000	300
2	1 011 0010	262	'	0 010 0111	047
3	0 011 0011	063	=	1 011 1101	275
4	1 011 0100	264	"	0 010 0010	042
5	0 011 0101	065	\	0 101 1100	134
			Null	0 000 0000	000
			Line Feed	0 000 1010	012
			Return	1 000 1101	215

Table A-5. Card Hole Pattern

Char.	Row	Char.	Row
A	12 - 1	6	6
B	12 - 2	7	7
C	12 - 3	8	8
D	12 - 4	9	9
E	12 - 5	Space	No Punches
F	12 - 6	¢ [	12 - 8 - 2
G	12 - 7	.	12 - 8 - 3
H	12 - 8	<	12 - 8 - 4
I	12 - 9	(	12 - 8 - 5
J	11 - 1	+	12 - 8 - 6
K	11 - 2	| !	12 - 8 - 7
L	11 - 3	&	12
M	11 - 4	!]	11 - 8 - 2
N	11 - 5	$	11 - 8 - 3
O	11 - 6	*	11 - 8 - 4
P	11 - 7	)	11 - 8 - 5
Q	11 - 8	;	11 - 8 - 6
R	11 - 9	¬ ^	11 - 8 - 7
S	0 - 2	–	11
T	0 - 3	/	0 - 1
U	0 - 4	,	0 - 8 - 3
V	0 - 5	%	0 - 8 - 4
W	0 - 6	–	0 - 8 - 5
X	0 - 7	>	0 - 8 - 6
Y	0 - 8	?	0 - 8 - 7
Z	0 - 9	:	8 - 2
0	0	#	8 - 3
1	1	@	8 - 4
2	2	'	8 - 5
3	3	=	8 -6
4	4	"	8 - 7
5	5	‡ \	0 - 8 - 2
		Null	12 - 0 - 9 - 8 - 1

If two characters, second is a USASCII character.	+ Sign	Same as number
	- Sign	Over Punch With 11

APPENDIX B

GLOSSARY

The following definitions are provided as a convenient, localized reminder of the meanings of some common terms used herein. Their treatment is informal and related specifically to their application in the programming of the Processor.

absolute address – a location designation that is assigned relative to the first location in core memory, such as to the PCB and to the record, IDT, SDAT, etc., when they are assigned core locations.

active record – a record that has had its record pointer and IDT pointer loaded into the PCB.

buffer – core-memory area for temporary storage, such as for I/O or working buffers.

byte – the smallest addressable unit (8 bits of information).

condition designator – an internal circuit condition which can be tested by certain Branching instructions.

core memory or storage – generally synonymous (at least herein).

data string – consecutive characters in core memory.

EBCDIC – Extended Binary Coded Decimal Interchange Code.

GOTO switches – switches on the Operator Panel that are manually set and serve as sense switches for activating a GOTO (Branch) instruction.

IDT – a table of descriptors that define record items.

IDT descriptor – an entry in the IDT which specifies the first and last positions of a given record item.

item – a related group of characters or bytes in a record that occupies consecutive locations in memory. An item may also consist of two or more items. An item may be from one to 256 bytes in length.

item number – One of up to 64 identifiers assigned to the items in a record and corresponding to descriptors in the IDT.

memory (core) map – an overall layout of the contents of core memory, indicating the assigned absolute addresses of the programming segments.

non-structured data – a (usually large) string of consecutive characters that are arranged in blocks, rather than records and items.

PCB – contains addresses that link the program with the data to be processed.

pointer – the address of the first location of a specified storage area, such as a record, IDT, SDAT, etc.

position – a relative location that is assigned with reference to the first (or 000) position of a record layout, IDT, program, etc.

record – a group of related items containing from one to 64 items (or up to 256 bytes).

relocatable – can be assigned to any location in core memory except the first 48 locations.

SDAT – a table of descriptors that defines records and buffers.

SDAT descriptor – an entry in the SDAT that specifies either the record and IDT pointers or the first and last addresses for a buffer.

subroutine – a relatively small sequence of instructions that performs a special or repeated function.

APPENDIX C

I/O CONTROLLERS

This appendix contains pertinent programming information on SYSTEM 2400 peripheral device controllers. Listings and brief narrative descriptions are included for the command and status codes employed by each peripheral device. Variations of these codes may be implemented to accommodate unique features/system functions requested by users.

The data presented herein is based on the assumption that the reader is knowledgeable of and has had experience in programming I/O Controllers and utilizing octal notations. The reader should consult the hardware manuals of the respective peripheral units for in-depth data.

C–2
SYSTEM 2400 DISPLAY CONTROLLER FOR MODEL 2404, 2405, AND 2408 PROCESSORS AND MODEL 2406 SYSTEMS CONSOLE

The Processor control panel contains switches and indicators to enable operation and monitoring of SYSTEM 2400 functions. These controls and indicators are (for the most part) program controlled, although facilities for manual operation are provided for certain functions. The following paragraphs describe the logical divisions of the panel and how to access and utilize them for checkout/operation.

The panel divisions are as depicted in Figure C-2.1. Group and sectional assignments are designated for applicable switches, indicators, and displays. These assignments are made to document all switches and indicators, thereby enabling access and control by programming and manual operation. Note that they are different from the assignments used in the *SYSTEM 2400 Processor Operator Panels* Hardware Manual (Form No. 2268).

- Programming control is accomplished via the basic IOCS (Input Output Control System) handler which enables assembly-language programs to activate or deactivate any available function light, read operator-set control buttons, and display up to nine digits of numeric information.

- Manual operation is at the discretion of the operator, as required, to obtain desired system operation.

channel; 2
select code; 1

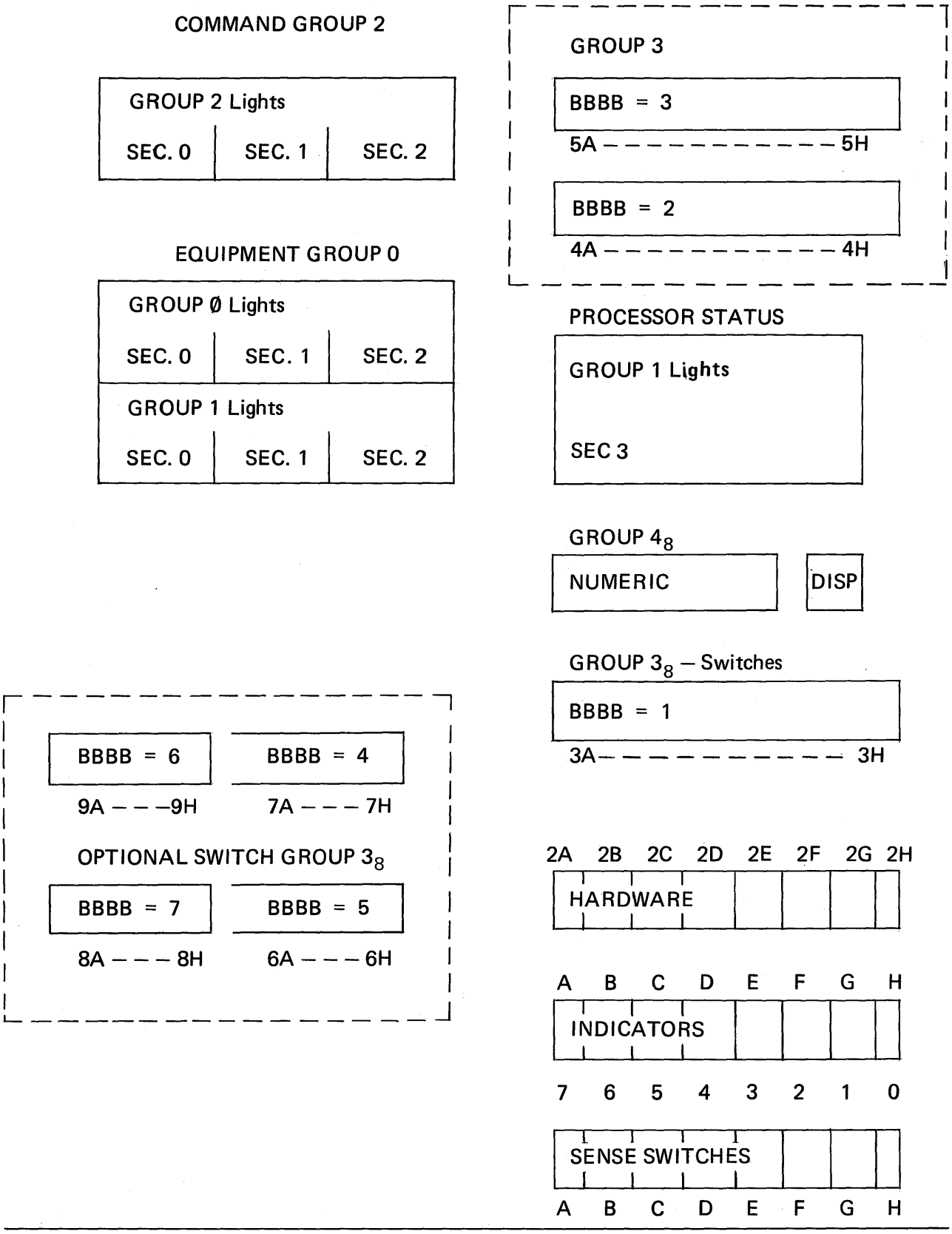


Figure C-2.1. Layout of Control Panel Lights and Switches

Notice Group 1 lights are in two different places on the panel. Group 3 Switches include all the software switches.

CONSOLE PARTITIONING

The divisions of the operator panel are assigned as GROUPS and SECTIONS as follows:

GROUPS – The total panel display is divided into five groups. Some items within groups are optional; therefore, it is the user's responsibility to ascertain which functions were incorporated into his particular system and thus available for use.

Group 0 – Equipment and Status I/O

Group 1 – Operator Messages

Group 2 – Communications

Group 3 – Control Switches

Group 4 – Numeric Display

SECTIONS – Groups are divided into one or more Sections (for indicators) or Switchbars (for switches), as shown in the table below.

GROUP	SECTION/SWITCHBARS
Group 0 Equipment Lights	Sections 0, 1, 2
Group 1 Equipment & Processor Lights	Sections 0, 1, 2, 3, (Section 3 is Processor Status)
Group 2 Command Lights	Sections 0, 1, 2
Group 3 Optional Switches	Switchbars 1, 2, 3, 4, 5, 6, 7 (Switchbars 2 & 3 are Communications dedicated)
Group 4 Numeric Lights	Section 0

The switchbars denoted "HARDWARE" and "SENSE SWITCHES" in Figure C-2.1 contain the same physical set of eight pushbutton switches as the other console switchbars. The HARDWARE switchbars are not programmable and contain fixed assigned hardware functions (as labeled) and operation is as described later within this section.

The Sense Switches are not programmable and *do not* have fixed assigned functions. Their functions are assigned within individual programs and may vary between phases or cycles of the program. Operation is as described in documentation accompanying each program. Examples of their utilization are given in this section.

The bank of lights denoted "INDICATORS" in Figure C-2.1 contains eight indicators which are assigned as the Status Indicators. These indicators are "independent" of the Sense Switches (*not* associated, as the physical location and alignment seem to imply). The primary function of these indicators is to present a visual indication of the response to an initial command requesting a status.

REFERENCE NUMBER/LOCATION ASSIGNMENTS

Some control panel indicator and switch names are variables that may be assigned according to customer requirements. (Blank control panel legend inserts are supplied by MDS with each system.) Therefore, permanent alpha, numeric, or alphanumeric designations are assigned (as depicted in Figure C-2.1) and described below to permit referencing in both software- and hardware-oriented operations.

Group 0 comprises 45 equipment group indicators divided into three sections of fifteen indicators each. These are individually numbered and addressed within a section as shown in the table below:

(1) 0001	(6) 0110	(11) 1011
(2) 0010	(7) 0111	(12) 1100
(3) 0011	(8) 1000	(13) 1101
(4) 0100	(9) 1001	(14) 1110
(5) 0101	(10) 1010	(15) 1111

Group 1 comprises 65 indicators divided into four sections. Sections Ø, 1, and 2 (with 15 indicators each) are equipment group indicators which are individually numbered and addressed as those in Group Ø. Section 3 contains 20 PROCESSOR STATUS indicators of which only two are programmable. The two programmable indicators are "EXECUTE" and "PAUSE" which are addressed as follows:

BBBB	Light
0001	EXECUTE
0010	PAUSE

Group 2 comprises 45 command group indicators divided into three Sections of 15 indicators each. These are individually numbered and addressed as those in Group Ø.

Group 3 comprises 56 switches divided into seven switchbars of eight switches each. These are individually numbered and addressed within a switchbar as shown below:

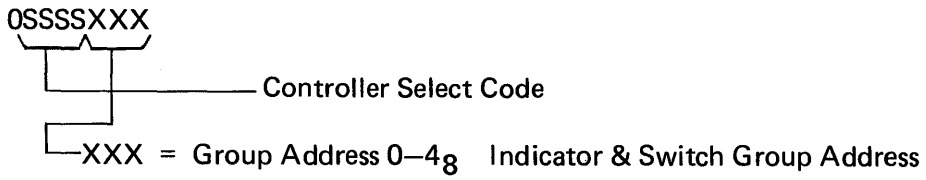
BBBB	
BBBB	0 = 0000
BBBB	1 = 0001
BBBB	2 = 0010
BBBB	3 = 0011
BBBB	4 = 0100
BBBB	5 = 0101
BBBB	6 = 0110
BBBB	7 = 0111

Group 4 comprises a nine-digit display divided into two sets of digits. The leftmost 6 digits indicate the memory location of the next instruction to be executed; the rightmost 3 digits indicate the content of the memory location denoted in the first 6 digits. These are individually numbered and addressed within the display as shown below:

0000 = 0	0101 = 5
0001 = 1	0110 = 6
0010 = 2	0111 = 7
0011 = 3	1000 = 8
0100 = 4	1001 = 9

SELECT & COMMAND CODES

The select and command specifier bytes are supplied in decimal code.
The select code composition is as depicted below (1 byte):



Group Address

0_8 = Equipment & Status I/O

1_8 = Operator Message (including "WAIT, EXECUTE" etc.)

2_8 = Communications

3_8 = Switches

4_8 = Numeric Display (requires 1–9 command bytes following)

Status (1 byte returned in response to Select Command)

The status byte bit notations are –

2_7 = Reserved

2_6 = Selected (main panel available)

2_5 = Communications available

2_4 = Switch option available

2_3 = Command accepted

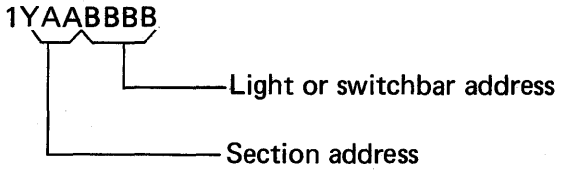
2_2 = Not used

2_1 = Not used

2_0 = Not used

Command Code (1 byte) and all succeeding command bytes

The Command code composition is as depicted below:



AA = Section address of light group to address

BBBB = One of fifteen lights in section to address or one of seven switchbars available to read

Y = 1 Turn ON light

= Ø Clear light or clear switch after reading bar

NOTE:

BBBB = 0000 clears all indicators in the addressed section

AA = 00 when reading switchbars

Section Addressing

Section addressing is as follows:

Section		AA Notation
Ø	=	ØØ
1	=	Ø1
2	=	1Ø
Extra *	=	11

*The addresses for the two programmable lights in the Processor Panel (Group 1, Section 3) are:

BBBB	Light
0001	Execute
0010	Pause

COMMAND SEQUENCE

If rightmost 3 bits of select code = 011_2, the selected switchbar information is returned in place of status for each command byte.

NOTE:

A select code can be followed by 1 or more command bytes allowing the lighting of several lamps in a "Group", or the input of one or all switchbars.

If a parity error occurs on a transfer of information to the Controller, the status bit "CM'D Accepted" will be turned OFF for that command byte and for all following bytes within the External Function command executed. All command bytes having status bit 2^3 (Command Accepted) equal to zero will have been ignored by the Controller and will not change the state of indicators or switches for the remainder of the external function.

If several command codes follow a select code and the status field reserved is not large enough to contain all of the status bytes returned, the trailing bytes are lost. A new select command must be issued to obtain the final status configuration.

Tables C-2.1, C-2.2, and C-2.3 list the programming address of all the Processor indicators and switchbars. They obviate the time-consuming task of individually programming each indicator or switch every time it is utilized.

Table C-2.1. Assignments for Basic and Full IOCS Display Handlers

INDICATOR	BASIC IOCS			FULL IOCS		
	GROUP	SECTOR	LIGHT	GROUP	SECTOR	LIGHT
OVERRUN	0	0	01	0	-	01
PARTIAL XER	0	0	02	0	-	02
MEDIA CK	0	0	03	0	-	03
FEED CK	0	0	04	0	-	04
SEEK CK	0	0	05	0	-	05
EQUIPMENT CHECK	0	0	06	0	-	06
RD DATA CK	0	0	07	0	-	07
WR DATA CK	0	0	08	0	-	08
LRC/CRC CK	0	0	09	0	-	09
VALIDITY CK	0	0	10	0	-	10
INTERV REQD	0	0	11	0	-	11
BUSY	0	0	12	0	-	12
NOT READY	0	0	13	0	-	13
NO SELECT	0	0	14	0	-	14
DVC NOT OP	0	0	15	0	-	15
RUNAWAY	0	1	01	0	-	16
7/9 TRK CK	0	1	02	0	-	17
LOAD POINT	0	1	03	0	-	18
END MEDIA	0	1	04	0	-	19
FILE PROTECT	0	1	05	0	-	20
CHECK 1	0	1	06	0	-	21
CHECK 2	0	1	07	0	-	22
CHECK 3	0	1	08	0	-	23
CHECK 4	0	1	09	0	-	24
CHECK 5	0	1	10	0	-	25
MAG TAPE	0	1	11	0	-	26
CARD READER	0	1	12	0	-	27
CARD PUNCH	0	1	13	0	-	28
LINE PRT	0	1	14	0	-	29
DATA-RECORDER	0	1	15	0	-	30
CARD RD/PCH	0	2	01	0	-	31
PT READER	0	2	02	0	-	32
PT PUNCH	0	2	03	0	-	33
DISK	0	2	04	0	-	34
CHAR PRT	0	2	05	0	-	35
COMM	0	2	07	0	-	37
CASSETTE	0	2	07	0	-	37
CRT	0	2	08	0	-	38
KEYBOARD	0	2	09	0	-	39
MULTIPLEXER	0	2	10	0	-	40
OCR READER	0	2	11	0	-	41
COMPTR CPLR	0	2	12	0	-	42
SPL EQUIP 1	0	2	13	0	-	43
SPL EQUIP 2	0	2	14	0	-	44
CONTROLLER	0	2	15	0	-	45

Table C-2.1 (Cont'd.). Assignments for Basic and Full IOCS Display Handlers

INDICATOR	BASIC IOCS			FULL IOCS		
	GROUP	SECTOR	LIGHT	GROUP	SECTOR	LIGHT
FORMAT CK	1	0	01	1	-	01
VERIFY CK	1	0	02	1	-	02
LENGTH CK	1	0	03	1	-	03
SEQ CK	1	0	04	1	-	04
CK SUM ERR	1	0	05	1	-	05
LABEL CK	1	0	06	1	-	06
RECORD CK	1	0	07	1	-	07
BLOCK CK	1	0	08	1	-	08
BATCH CK	1	0	09	1	-	09
MEM EXCEED	1	0	10	1	-	10
UNEXPRD FIL	1	0	11	1	-	11
REC NOT FND	1	0	12	1	-	12
FIL NOT FND	1	0	13	1	-	13
NO END REC	1	0	14	1	-	14
PRG NOT END	1	0	15	1	-	15
INVLD RESP	1	1	01	1	-	16
BKSP TO LPT	1	1	02	1	-	17
UNDETR ERR	1	1	03	1	-	18
INVLD CHAR	1	1	04	1	-	19
INVLD VFC	1	1	05	1	-	20
INPUT ON	1	1	06	1	-	21
OUTPUT ON	1	1	07	1	-	22
FILE SELECT	1	1	08	1	-	23
FILE FOUND	1	1	09	1	-	24
REWIND	1	1	10	1	-	25
REREAD CARD	1	1	11	1	-	26
LOAD MEDIA	1	1	12	1	-	27
SET PARAM	1	1	13	1	-	28
DUMP MEM	1	1	14	1	-	29
ALARM	1	1	15	1	-	30
END OF FILE	1	2	01	1	-	31
PHASE COMP	1	2	02	1	-	32
END OF JOB	1	2	03	1	-	33
JOB CANCEL	1	2	04	1	-	34
NON RECV ERR	1	2	05	1	-	35
RETRY OPT	1	2	06	1	-	36
IGNORE OPT	1	2	07	1	-	37
EOF OPT	1	2	08	1	-	38
CANCEL OPT	1	2	09	1	-	39
EXEC IF READY	1	2	10	1	-	40
OPER MSG	1	2	11	1	-	41
RECD COUNT	1	2	12	1	-	42
BATCH TOTAL	1	2	13	1	-	43
PHASE NUMBER	1	2	14	1	-	44

Table C-2.1 (Cont'd.). Assignments for Basic and Full IOCS Display Handlers

INDICATOR	BASIC IOCS			FULL IOCS		
	GROUP	SECTOR	LIGHT	GROUP	SECTOR	LIGHT
UNIT NUMBER	1	2	15	1	-	45
EXECUTE	1	3	01	1	-	46
PAUSE	1	3	02	1	-	47
LINE CHECK	2	0	01	2	-	01
PARITY CHECK	2	0	02	2	-	02
SEQ CK	2	0	03	2	-	03
TIMEOUT	2	0	04	2	-	04
ABANDN CALL	2	0	05	2	-	05
NEG REPLY	2	0	06	2	-	06
OVERFLOW	2	0	07	2	-	07
INCOMPLETE	2	0	08	2	-	08
DISCONNECT	2	0	09	2	-	09
HALT	2	0	10	2	-	10
REMOTE ALERT	2	0	11	2	-	11
WAIT	2	0	12	2	-	12
RETRANSMIT	2	0	13	2	-	13
INITIALIZE	2	0	14	2	-	14
DIAL COMPL	2	0	15	2	-	15
PROGRAM 1	2	1	01	2	-	16
PROGRAM 2	2	1	02	2	-	17
PROGRAM 3	2	1	03	2	-	18
RVI	2	1	04	2	-	19
EOT	2	1	05	2	-	20
PRINTER REQ	2	1	06	2	-	21
TAPE REQ	2	1	07	2	-	22
PUNCH REQ	2	1	08	2	-	23
CRT REQ	2	1	09	2	-	24
KEYBOARD ENTRY	2	1	10	2	-	25
POLL	2	1	11	2	-	26
AUTO ANSWER	2	1	12	2	-	27
AUTO CALL	2	1	13	2	-	28
TRANSPARENT	2	1	14	2	-	29
END OF FILE	2	1	15	2	-	30
TERMINATE	2	2	01	2	-	31
SIGNAL REMOTE	2	2	02	2	-	32

Table C-2.2. Numeric Display Assignments for Basic and Full IOCS Display Handlers

DIGIT VALUES	BASIC IOCS			FULL IOCS		
	GROUP	SECTOR	DIGIT	GROUP	SECTOR	DIGIT
1	4	0	01	4	–	1
2	4	0	02	4	–	2
3	4	0	03	4	–	3
4	4	0	04	4	–	4
5	4	0	05	4	–	5
6	4	0	06	4	–	6
7	4	0	07	4	–	7
8	4	0	08	4	–	8
9	4	0	09	4	–	9
0	4	0	00	4	–	0
BLANK	4	0	15	4	–	SPACE CHARACTER

Table C-2.3. Switchbar Assignments for Basic and Full IOCS Display Handlers

SWITCHBAR NUMBER	BASIC IOCS			FULL IOCS		
	GROUP	SECTOR	BAR	GROUP	SECTOR	BAR
3	3	0	03	3	–	3
4	3	0	04	3	–	4
5	3	0	05	3	–	5
6	3	0	06	3	–	6
7	3	0	07	3	–	7
8	3	0	08	3	–	8
9	3	0	09	3	–	9

NOTE: There are eight pushbuttons in a switchbar. These are numbered nA through nH (where n = 3 to 9).

Pushbuttons within switchbar #3 are labeled as follows:

RETRY	3A
IGNORE	3B
EOF	3C
—	3D
CANCEL	3E
—	3F
PAUSE	3G
EXECUTE	3H

C-3
CONTROLLER
FOR MODEL 2453 CARD READER

COMMAND CODES

SELECT

OS	SSS	XXX	–	S	=	Switch Selectable
76	543	210	–	Bit Position		

READ	202
READ BINARY	212
NO-OP	204

STATUS CODES

Bit Position	Meaning
7	End-of-File
6	Selected and Not Busy
5	Busy – if present, all other bits blocked
4	Hopper Empty/Stacker Full
3	Program Check – Buffer aborted due to Pick Fail or Not-Ready Condition
2	Equipment Check – Parity error, Illegal Command, Validity Error
1	Read Trouble – Light and Dark Current errors and motion errors
0	Ready – Command accepted

COMMAND CODES DESCRIPTION

SELECT – The Select command is issued by the Processor to select a particular device on an I/O channel. All devices on that channel must look at the select code and determine if it is for them. The selected device will logically connect to the I/O channel and any other device that may have been connected will disconnect. Once selected, a device will remain selected until it sees a select code not for itself or a channel clear.

After initial selection, the select command is used to input the status of previous operations.

In the Card Reader Controller, there will be a busy period following each data transfer (waiting for the Reader to become ready again). During this period, any status attempt will give only the Busy Bit (bit 5). Attempts to get status on the previous data transfer may be done by a NO-OP command by examining byte 1 of the first command to be answered without a Busy bit.

READ (202) – This command prepares the controller for an Input operation which will take them in Hollerith codes and translate them to EBCDIC before giving them to the processor. Both 64- and 256-character translators are available by simply changing 2 boards. Following either Read command, at least 1 character must be transferred or the controller will remain busy until a channel clear is received.

READ BINARY (212) – If the read binary board is installed, this command prepares the controller for an input operation that will take data from the upper half of the card and place it in the lower 6 bits of the first word: Row 12 in bit 5, row 11 in bit 4, row 0 in bit 3 and row 9 in bit 2, row 8 in bit 1, and row 7 in bit 0.

The lower half of the card will be placed in the lower 6 bits of the second word: Row 6 in bit 5, row 5 in bit 4, row 4 in bit 3, row 3 in bit 2, row 2 in bit 1 and row 1 in bit 0.

Thus, an 80 column card will transfer 160 characters.

NO-OP (204) – This command provides a means of clearing existing status bits after the situation has been corrected, such as Hopper Empty or Pick Fail. It should also be used to initialize an I/O routine because the first status will probably contain a Hopper Empty bit.

COMMAND CODES SUMMARY

Command	Byte 1	Byte 2
Read	Select	202
Read Binary	Select	212
No-Op	Select	204

STATUS CODES DESCRIPTION

7 – END OF FILE. This bit indicates a Hopper Empty condition has occurred with the End-of-File switch depressed.

6 – SELECTED (Not Busy). This bit will be returned in the status of a successful connect and with any other status presented, unless the Busy bit is present.

5 – BUSY. When presented, all other status bits are locked out. This bit indicates that the controller is still working on a previous command. Any existing status conditions are stored until not busy, provided status is taken by a Select function.

4 – HOPPER EMPTY/STACKER FULL. One of these conditions has occurred since the last command was issued.

3 – PROGRAM CHECK. Indicates that the buffer has been terminated and the data transfer did not take place. (Card reader went *into* a not-ready state after command acceptance, but before the data transfer.)

2 – EQUIPMENT CHECK. This bit indicates a parity error on a command byte, a validity error, or an Illegal Command.

1 – READ TROUBLE. Motion errors or light and dark check errors.

0 – READY. Indicates that the command received was a legal command (at least one bit, but no more than one of the lower 3 bits), and the Reader is ready to send data. The Ready bit in the status for a Select command is present from the previous operation and has no meaning for any future commands.

C–4
CONTROLLER
FOR MODEL 2455 CARD READER

FUNCTIONAL DESCRIPTION

The 2455 interfaces to a 2400 I/O selector channel. Operations are initiated by command codes transmitted over the I/O channel.

Command Sequence

Format:

Operations are initiated with an "External Function on Channel" instruction with the following formats defined for the OP2 item.

a) Select
b) Select . . . . . Command

If the select byte is responded to by the 2455, one byte of status is returned to the processor for each byte in the OP2 item.

Select Byte

The select byte is used to address a unique controller out of the 16 possible controllers on an I/O channel.

The select byte is differentiated from the command byte by decoding combinations of I/O channel function lines. (FRQ = Function Request; DRQ = Data Request)

FRQ and DRQ = Select Byte

FRQ and $\overline{\text{DRQ}}$ = Command Byte

$\overline{\text{FRQ}}$ and DRQ = Data Request

Format: XS SSS XXX

S Bits (6 through 3) are a bit pattern unique for each controller on a channel. X bits (7, and 2 through 0) are ignored.

Selection of the 2455 takes place when the bit pattern in the select byte matches the state of a switch-selectable decoder. The decoder has the capability of decoding any of the 16 possible states of the select byte.

After selection is accomplished, the 2455 responds to the following byte as a command.

If the decoder has a negative response to the select byte or a parity error is detected, the 2455 will deselect and ignore all ensuing command or data bytes. Deselection will take place within 2 microseconds after the rise of FRQ.

In acknowledgment of a successful select, the 2455 will respond with a status byte indicating present and/or accumulated conditions no sooner than 3 microseconds after the rise of FRQ.

Command Byte

The command byte is used to initialize the 2455 for a read operation or to input status (in the case of a No-Op).

NO-OP Command

Format: XX XXX 100

No operation is initiated. This command is principally used to ready and/or reset status.

Read Command

Format: XX 000 010

This command prepares the controller for an input operation which will take in Hollerith codes from the card reader and translate them to EBCDIC before giving them to the processor. Both 64-character and 256-character translators are available by simply changing 2 boards. Following the Read command, at least 1 character must be transferred or the controller will remain busy until a channel clear is received.

Read Binary Command

Format: XX 001 010

If the read binary board (Feature 170) is installed, this command prepares the controller for an input operation that will take data from the upper half of the card and place it in the lower 6 bits of the first word: Row 12 in bit 5, row 11 in bit 4, row 0 in bit 3 and row 9 in bit 2, row 8 in bit 1, and row 7 in bit 0.

The lower half of the card will be placed in the lower 6 bits of the second word: Row 6 in bit 5, row 5 in bit 4, row 4 in bit 3, row 3 in bit 2, row 2 in bit 1 and row 1 in bit 0.

This process will repeat such that an 80-column card will transfer 160 characters.

Bits 2^7 and 2^6 will be forced to the zero state for all read binary data transfers.

If Feature 170 is not installed, this command will be flagged as illegal.

Command and Status Tables

Table C-4.1 lists the 2455 Card Reader Commands, while Table C-4.2 lists the Status Bits.

why 200?

Table C-4.1. Model 2455 Card Reader Commands

COMMAND	DESCRIPTION
XX XXX 100	No-Op (204 Clear old (accumulated) status and present new status if not busy. All status will be saved during the busy period.
XX 000 010	Read (202 Initializes read operation of Hollerith codes from cards translated to EBCDIC.
XX 001 010	Read Binary (212) Initializes a Card Image Read operation.

Table C-4.2. Model 2455 Card Reader Status Bits

POSITION	NAME	DESCRIPTION
2^7	EOF	End of File
2^6	Selected	
2^5	Busy	
2^4	HE/SF	Hopper Empty/Stacker Full
2^3	Program Check	Buffer aborted due to Pick Fail or Not-Ready condition
2^2	Equipment Check	Illegal Command/Validity Error
2^1	Read Trouble	Motion/Current Checks
2^0	Ready	Command Accepted

Illegal Commands

The command byte will be checked to insure an executable command has been requested. The following commands will be flagged as illegal:

- Any command with bit 0 active.
- Any command with more than one of the lower 3 bits active.
- Any command with none of the lower 3 bits active.
- A Read Binary Command without Feature 170 installed.
- Any command with a parity error.

An illegal command is *not* executed.

All commands are ignored if "Busy" is a logical "1".

Status Bytes

One byte of status is returned for each byte of the EF (External Function) instruction, if a successful select occurs.

Prior status is returned in response to the select byte. These status bits retain the logical "1" status conditions of new status from the last command and accumulate any "1" bits which may occur (momentarily or permanently) since that time. ("Busy" is an exception.) All eight bits may *not* simultaneously be a logical "0". This state indicates the processor did *not* receive a status response and internally generated a response of zero.

New status is returned in response to the command byte. The command byte clears prior status (except if "Busy" is a logical "1"), so new status bits returned are conditions at the time the command is received.

Both "Busy" and "End of File" are defined such that neither can be a "1" for new status.

Status Bit Format

End of File
Format: 11 010 XX1

This bit indicates that the last card which was read caused the hopper to empty while the End-of-File option was selected (EOF switch lighted).

Selected

Format: X1 0XX XXX

This bit is present when selected, if *not* "Busy".

Busy

Format: 00 100 000

Busy indicates that the controller is presently performing a previously initiated operation. When busy is present, it inhibits gating of any other status bits, but does *not* clear them from their holding logic. Thus, when the busy condition disappears, its bit becomes a logical "0" and the remainder of the bits take on their prior states plus any logical "1" accumulations.

NOTE:

Logical "0" states of any bits other than "busy" are meaningless unless busy is known to be logical "0".

If busy is present in prior status, it will also be present in new status.

Hopper Empty/Stacker Full

Format: X1 01X XXX

This bit indicates the presence of a hopper empty and/or stacker full condition at the reader.

Program Check

Format: X1 0X1 XXX

This bit indicates that the previous input failed to transfer data. This could be due to a pick failure or a *"not* ready" condition at the time an input is initiated, but unless the motion light is on (Read Trouble), the card should still be in the input hopper.

Equipment Check

Format: X1 0XX 1XX

An illegal command was detected, if appearing in new status. A validity error was detected, if appearing in prior status.

Read Trouble

Format: X1 0XX X1X

This bit indicates a motion error, light current error, or dark current error has been detected on the previous card read. If this bit is present, the card in error will be found in the stacker or partially through the read station.

Ready

Format: X1 0XX XX1

This bit indicates command acceptance. This bit is meaningless in select byte status. If present in command byte status, the 2455 is committed to complete whatever operation was requested and will remain in a "busy" state until completion or until a channel clear is received.

Data Transfer

The input data transfer is initiated with an "Initiate Input on Channel" instruction. Card motion is not initiated at command time, but is held up until the first DRQ is received.

If the 2455 is not able to provide the data because (1) a command was not accepted, or (2) the reader went to a *not*-ready state after command acceptance, the 2455 will return a FAKN signal in response to the first DRQ to terminate the buffer.

NOTE:

After Read Command acceptance, at least one data byte must be requested or the 2455 will remain "busy" and wait for the input instruction indefinitely, or until a "channel clear" is received.

If more than a full card of data is requested, the 2455 will return a FAKN signal to a request for the 81st column of data. *No* special status bit will be set in this case.

If a validity error is detected in any column, the Equipment Check bit will be set to a logical "1" only if that column of data was requested (*not* if a short buffer was issued and the error is later in the card).

Program Load

For a program load, the 2455 will receive a normal select byte followed by activation of the "Start Load" control line from the processor. This will be followed by a buffer which will try to continue until the detection of a "9" code in the lower 4 bits.

Select Byte

Response to the select byte will be no different from a normal select.

Start Load

This line will cause the 2455 to go through a dummy command cycle (read bit forced active by the 2455). Data on the channel I/O lines during this signal is meaningless and is not sampled. There will be no response to the processor for this signal (FAKN, Function Acknowledge, is disabled). The normal "clear" performing operation at end of card is disabled, to allow more than one card to be read during a load operation.

Data Transfer

When the Start Load line drops, the processor will initiate an input buffer. Data will be transferred the same as in a normal read operation, except that additional cards will be picked as long as the buffer remains active. Upon detection of a "9" code, the processor will terminate the buffer. This will be detected in the 2455 by sensing the absence of DRQ for more than 15 microseconds, upon which the operation is considered complete.

C–5
CONTROLLER
FOR MODEL 2457 CARD PUNCH

The following is a functional description for the Model 2457 Card Punch controller operating on a selector channel of a 2400 Processor. Table C-5.1 lists the applicable 2457 Command Codes.

Table C-5.1. Model 2457 Card Punch Command Codes

COMMAND	BYTE 1	BYTE 2	BYTE 3	BYTE 4
WRITE	SELECT*	200	200	201
SELECT NORMAL STACKER	SELECT*	200	202	204
SELECT REJECT STACKER	SELECT	200	201	204
NO OP	SELECT*	200	200	204
WRITE BINARY	SELECT*	200	204	201

*SELECT BYTE

76 543 210 Bit Position
XS SSS XXX

S Switch Selectable
X Not Used

For each byte of command sent to the controller from the processor, a status byte will be returned to the processor by the controller. Bytes 1 and 4 are the same format, and bytes 2 and 3 contain the same status information. The first two bytes contain old status and the last two contain current status.

DEFINITIONS

- A byte is eight binary bits
- The capitalized letter "X" signifies a binary digit which may take either the logical "1" or the logical "0" state.

NOTE:

Programmers are cautioned to program any "X" bits in the select or output bytes as logical "0" so that future expansion will not invalidate preceeding software.

- Capitalized letters other than "X" represent command or status bits which may be present in conjunction with the bits defined.
- Byte bit numbering represents the positional power of two represented by the binary digit.
- When bit position is unspecified for a list of bit or bytes, the ordering is from most to least significant.
- Output and input are defined relative to the 2400 Processor.

COMMAND SEQUENCE

Formats

Operations are initiated with an "External Function on Channel" instruction with only the following formats defined for the OP2 item.

Select, dummy, command 1, command 2
Select, dummy,
Select

If the select byte is responded to by the controller, one byte of status is returned for each of the function bytes.

Select byte

The select byte is used to address a unique controller out of the 16 possible controllers on an I/O channel.

Format XS SSS XXX - In the first byte in the command sequence, bits 6 through 3 are a bit pattern unique for each controller on the channel.

The controller will recognize select byte as that in which a function request and data request are presented at the same time.

Controller selection takes place when the bit pattern in the select byte matches the state of a switch selectable decoder. The decoder has the capability of decoding any of the 16 possible states of the select byte.

If the controller has a negative response to a select byte, the controller will deselect and ignore all ensuing commands.

Dummy Command

The dummy command is formatted as 1XXXXXXX and has no effect on the controller. Its function is to allow a 4th status byte of status to be sent to the processor.

Command 1

Command 1 is utilized to generate mode commands to modify the internal mode of the card punch or to generate a binary modifier for the read command for a read binary operation.

Feed Mode Commands

Format: 10 0AB 000

Feed mode commands will be ignored.

Stacker Select Commands

Format: 10 000 CD0

Stacker select commands will be used to change the mode of stacker selection: (a) stacker 1 on normal error free cards, and (b) stacker 2 for cards where skew error occurs or echo check errors occur on punch.

Select Reject Stacker C=0, D=1

The select reject stacker command will cause all the cards fed after acceptance of this command to go to stacker 2.

Select Normal Stacker C=1, D=0

The select normal stacker will cause the card punch to return to its normal mode of stacking cards, which is all cards go to stacker 1 on feed operation except those where a punch error occurs. Those cards go to stacker 2.

Binary Command

Format: 10 000 001

The binary command will result in a read binary operation if the command 2 byte is a read command.

Dummy Command

Format: 10 000 000

A dummy command in byte 3 will have no effect on the controller. Its function is to allow a status byte to be sent to the processor.

Command 2

Command 2 is utilized to initiate a read or punch operation or initiate a mode command. It also depends (in most cases) on the results of the command 1 byte.

No-Op and Card Punch Mode Command

Format: 10 000 100

This command will cause the execution of the command in byte 1 (except for the binary command which requires a read in command 2). If command 1 is a dummy command, this byte will result in a No-Op, a read, and clear status operation.

Punch Command

Format: 10 000 001

This command enables the controller to accept data to be punched from the processor. Up to 80 bytes of data will be accepted. The 81st byte of data will terminate the data transfer.

Read Command

Format: 10 000 010

This command is an illegal command.

Illegal Commands

Illegal commands comprise the following:

- A format of 10 000 111, 10 000 110, 100 000 101, 10 000 011 or 10 000 000 in Command 2 will result in an illegal command.
- A byte in Command 1 where A=B=1 or C=D=1 will result in an illegal command.
- Any parity error on command bytes results in an illegal command.
- An attempted command for a feature not installed and a read command will result in an illegal command.

STATUS BYTES

One status byte is returned for each select, dummy, or command byte executed. If a 4-byte function is executed, status will be returned in the following order (if the control unit is not busy): Prior Status 1, Prior Status 2, New Status 2, and New Status 1.

Prior Status 1

Prior Status 1 is returned in response to the select byte. These status bits retain the logical "1" status conditions of New Status 1 from the last Command 2 executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time. (Busy is an exception.) All eight bits may *not* simultaneously be logical "0". This state indicates the processor did *not* receive a status response and internally generated a response of zero.

End of File (EOF)

Format: 1X0XXXXX

The end of file status bit will be initiated when the hopper goes empty and the last card to be punched enters the read station. It will remain activated as a status bit until the controller assumes not ready.

Selected

Format: X10XXXXX

Logical "1", if selected.

Busy

Format: 00100000

Busy indicates that the controller is presently performing a previously initiated operation. When Busy is present, it inhibits gating of any other status bits (in either status 1 or status 2) but does *not* clear them from their holding logic. Thus, when the Busy condition disappears, its bit becomes logical "0" and the remainder of the bits takes on their prior status plus any logical "1" accumulations.

NOTE:

Logical "0" states of any bits other than Busy are meaningless, unless Busy is known to to logical "0".

Intervention Required

Format: XX01XXXX

Operator assistance is required before the Card Punch can attain an operable state. When in this state, the INOP light on the Card Punch control panel will be illuminated.

Misfeed or Illegal Data

Format: XX0X1XXX

This bit will occur if a pick failure occurs or a data character has been received after the loss of a ready status due to operator intervention between the issuance of a read punch command and the reception of a data byte. Discrimination between illegal data and pick failure can be made by the examination of the second prior status byte. Recovery from the illegal data status condition can be made by re-issuance of the command and data operations.

Equipment Check

Format: XX0XX1XX

An illegal command byte 2 was detected, parity error on command 1 or command 2, or a validity check occurred on the prior read operation.

Data Error

Format: XX0XXX1X

A punch error occurred on the prior data transfer.

Ready

Format: XX0XXXX1

Command bytes 1 and 2 were implemented. If equipment Check of Intervention Required conditions are present, Ready must be logical "0".

Prior Status 2

Prior Status 2 is returned in response to the dummy byte. These status bits retain the logical "1" status conditions of new status 2 from the last command 1 executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time.

Reject Stacker

Format: 1XXXXXXX

Indicates the reject stacker has been selected.

Normal Feed Mode Status

Format: X1XXXXXX

Indicates the Card Punch is in the feed-after-punch mode.

Illegal Data

Format: XX1XXXXX

Indicates the receipt of a data byte when the controller is not Ready.

Jam

Format: XXX1XXXX

Indicates a jam condition in the Card Punch which also results in an Intervention Required status.

Chips

Format: XXXX1XXX

Indicates that the chip bin is full. This bit implies an Intervention Required status.

Stacker Full

Format: XXXXX1XX

Indicates that either stacker one or stacker two is full. This bit implies that an intervention is required.

Hopper Empty

Format: XXXXXX1X

Indicates that the hopper is empty. This bit implies that an intervention is required.

Card Punch Ready

Format: XXXXXXX1

Indicates that the Card Punch is available and ready to process data.

New Status 1

New Status 1 is returned in response to command 2. Command 2 clears the states of prior status 1, so new status 1 bits returned are conditioned at the time command 2 is executed. Status bits are the same as described earlier for Prior Status 1. Both Busy and Data Error are defined so that neither can be a logical "1" for new status 1.

New Status 2

New Status 2 is returned in response to command 1. Command 1 clears tne status of Prior Status 2, so New Status 2 bits returned reflect existing conditions at the time command 1 is executed.

DATA TRANSFER

The output data transfer is initiated with an "Initiate Output on Channel Instruction." If the controller is not prepared for data transmission because a previous command was not previously executed, the controller will return a Function Acknowledge signal in response to the first Data Request signal received.

If the interval between data requests is 15±3 microseconds, the controller will terminate the data transfer operation. A data request appearing after the termination will result in a "function acknowledge" to be returned to the processor.

NOTE:

At least one data byte must be received; the controller will remain busy and await the data transfer indefinitely.

If more than 80 characters are requested by the processor in the normal mode, the controller will return a function acknowledge to the 81st character.

A parity error occuring on a Card Punch data transfer will result in an Equipment Check status.

A master clear due to the depression of the clear switch or the activation of an interlock will terminate the operation.

Table C-5.2. 2457 Card Punch Status Bits Summary

BYTES 1 AND 4		BYTES 2 AND 3	
Bit	Name	Bit	Name
7	EOF	7	REJECT STACKER
6	SELECTED	6	NORMAL FEED MODE STATUS
5	BUSY	5	ILLEGAL DATA
4	INTERVENTION REQUIRED	4	JAM
3	MISFEED OR ILLEGAL DATA	3	CHIPS
2	EQUIPMENT CHECK	2	STACKER FULL
1	DATA ERROR	1	HOPPER EMPTY
0	READY	0	READY

C–6
CONTROLLER
FOR MODEL 2459 CARD PUNCH

Command Sequence

Format

Operations are initiated with an "External Function on Channel" instruction with the following formats defined for the OP2 item.

Select
Select . . . . . Command

If the select byte is responded to by the 2459, one byte of status is returned to the processor for each byte in the OP2 item.

Select Byte

The select byte is used to address a unique controller out of the 16 possible controllers on an I/O channel.

The select byte is differentiated from the command byte by decoding combinations of I/O channel function lines.

FRQ and $\overline{\text{DRQ}}$ = Select Byte
$\overline{\text{FRQ}}$ and $\overline{\text{DRQ}}$ = Command Byte
$\overline{\text{FRQ}}$ and DRQ = Data Request

Format: XS SSS XXX

S bits (6 through 3) are a bit pattern unique for each controller on a channel. X bits (7, and 2 through 0) are ignored.

Selection of the 2459 takes place when the bit pattern in the select byte matches the state of a switch-selectable decoder. The decoder has the capability of decoding any of the 16 possible states of the select byte.

After selection is accomplished, the 2459 responds to the following byte as a command.

If the decoder has a negative response to the select byte or a parity error is detected, the 2459 will deselect and ignore all ensuing command or data bytes. Deselection will take place within 2 microseconds after the rise of FRQ.

In acknowledgment of a successful select, the 2459 will respond with a status byte indicating present and/or accumulated conditions, no sooner than 3 microseconds after the initialization of FRQ.

Table C–6.1. Model 2459 Card Punch Commands

COMMAND	DESCRIPTION
XX 0XX 100	No-Op (204) Clear old (accumulated) status and present new status if not busy. All status will be saved during the Busy period.
XX 0XX 001	WRITE (201) Initializes Write operation of EBCDIC codes from cards translated to HOLLERITH.
XX 1XX 001	WRITE BINARY (241) Initializes a Card Image Write operation.

Table C–6.2. Model 2459 Card Punch Status Bits

POSITION	NAME
2^7	Card Jam
2^6	Selected
2^5	Busy
2^4	Intervention Required
2^3	Pick Failure
2^2	Equipment Check
2^1	Data Error
2^0	Ready

Command Byte

The command byte is used to initialize the 2459 for a write operation or to input status (in the case of a NO-0P).

NO-0P Command

Format: XX 0XX 100

No operation is initiated. This command is principally used to read and/or reset status.

Write Command

Format: XX 0X0 001

This command prepares the controller for a data transfer operation which will take in Hollerith codes from the processor and translate them to EBCDIC before giving them to the card punch. Both 64-character and 256-character translators are available by simply changing 2 boards. Following the write command, at least 1 character must be transferred or the controller will remain Busy until a Channel Clear is received.

Write Binary Command

Format: XX 1X0 001

If the write binary board (Feature 171) is installed, this command prepares the controller for a data transfer operation that will take the first byte of data from the processor, truncate the upper two bits (bit 7 and 6), and place the remaining six bits in the upper half of the card (bit 5 in row 12, bit 4 in row 11, bit 3 in row 0, bit 2 in row 9, bit 1 in row 8, and bit 0 in row 7).

The second byte of data will likewise be truncated and placed in the lower half of the card (bit 5 in row 6, bit 4 in row 5, bit 3 in row 4, bit 2 in row 3, bit 1 in row 2 and bit 0 in row 1).

This process will repeat such that 160 bytes of data will be required to punch an 80-column card.

If this command is used when Feature 171 is not installed, it will be flagged as illegal.

Offset

Format: XX *X1 ***

*The pattern of these bits must constitute a legal command. The offset modifier bit will cause an ejected card to be offset approximately 1/2 inch in the punched card stacker.

NOTE:

All cards will be offset if a data error exists.

Illegal Commands

The command byte will be checked to insure an executable command has been requested. The following commands will be flagged as illegal:

- Any command with bit 1 active.
- Any command with more than one of the lower 3 bits active.
- Any command with none of the lower 3 bits active.
- A Write Binary Command without Feature 171 installed.
- Bit 5 active with bit 2 active.
- Any command with a parity error.

An illegal command is not executed.

All commands are ignored, if "Busy" is a logical "1".

Status Bytes

If a successful select occurs, one byte of status is returned for each byte of the EF instruction.

Prior status is returned in response to the select byte. These status bits retain the logical "1" status conditions of New Status from the last command and accumulate any "1" bits which may occur (momentarily or permanently) since that time. (Busy is an exception. All eight bits may not simultaneously be a logical "0". This state indicates the processor did not receive a status response and internally generated a response of zero.)

New status is returned in response to the command byte. The command byte clears prior status (except if "Busy" is a logical "1"), so new status bits returned are conditions at the time the command is received.

Status Bit Format

Card Jam

Format: 11 01X XXX

Card Jam indicates that there has been an eject failure or a stack failure. Possible card damage may have occurred.

Selected

Format: X1 0XX XXX

If not Busy, this bit is present when the 2459 is selected.

Busy

Format: 00 100 000

Busy indicates that the controller is presently performing a previously initiated operation. When Busy is present, it inhibits gating of any other status bits, but does *not* clear them from their holding logic. Thus, when the Busy condition disappears, its bit becomes a logical "0" and the remainder of the bits take on their prior status plus any logical "1" accumulations.

NOTE:

Logical "0" states of any bits other than Busy are meaningless, unless Busy is known to be logical "0".

If Busy is present in prior status, it will also be present in new status.

Intervention Required

Format: X1 01X XXX

Intervention Required indicates that there is a problem that may be cleared by operator intervention. Any of the following problems can give an "Intervention Required" indication: hopper empty, stacker full, stack failure, pick failure, eject failure, and chip box full.

Pick Failure

Format: X1 011 XXX

Pick Failure indicates that the 2459 initiated a pick card operation, and the card did not appear in the registration gate in a given amount of time.

Equipment Check

Format: X1 0XX 1XX

An illegal command was detected if appearing in new status. A validity error was detected if appearing in prior status.

Data Error

Format: X1 0XX X1X

Data Error indicates that the data received and/or punched by the 2459 was erroneous. The following problems can cause a Data Error indication: (1) a parity error detected on data transfer from processor to punch; (2) proper punch pins were not engaged; and (3) improper character received by the 2459 (64-character translator *must* be installed).

Ready

Format: X1 0XX XX1

This bit indicates command acceptance. This bit is meaningless in select byte status. If present in command byte status, the 2459 is committed to complete whatever operation was requested and will remain in a Busy state until completion or until a "Channel Clear" is received.

Data Transfer

The input Data Transfer is initiated with an "Initiate Output on Channel " instruction. Card motion is *not* initiated at command time, but is held up until the first DRQ is received.

If the 2459 is *not* able to accept the data either because a command was not accepted or because the punch went to a not-ready state after command acceptance, the 2459 will terminate the buffer by returning a FAKN signal in response to the first DRQ.

NOTE:

After Write command acceptance, at least one data byte *must* be sent or the 2459 will remain "Busy" and wait for the input instruction indefinitely, or until a "Channel Clear" is received.

If more than a full card of data is transmitted, the 2459 will return a FAKN signal to a Request for the 81st column of data. No special status bit will be set.

If a validity error is detected in any column, the Equipment Check bit will be set to a logical "1".

C–7
CONTROLLER
FOR MODEL 2458 CARD READER/PUNCH

The following is a functional description for a Model 2458 Card Reader/Punch controller operating on a selector channel of a 2400 processor. Table C-7.1 lists the applicable 2458 command codes.

Table C-7.1. Model 2458 Card Reader/Punch Commands

COMMAND	BYTE 1	BYTE 2	BYTE 3	BYTE 4
READ	SELECT*	200	200	202
READ BINARY	SELECT*	200	204	202
WRITE	SELECT*	200	200	201
SELECT NON FEED MODE	SELECT*	200	210	204
SELECT NORMAL MODE (CHAR. NON-FEED MODE)	SELECT*	200	220	204
SELECT NORMAL STACKER	SELECT	200	202	204
SELECT REJECT STACKER	SELECT*	200	201	204
NO OP	SELECT*	200	200	204

*SELECT BYTE

76	543	210	Bit Position
OS	SSS	XXX	

S Switch Selectable
X (Not Used)

For each byte of command sent to the controller from the processor, a status byte will be returned to the processor by the controller. Bytes 1 and 4 are the same format, and bytes 2 and 3 contain the same status information. The first two bytes contain old status and the last two contain current status.

DEFINITIONS

- A byte is eight binary bits
- The capitalized letter "X" signifies a binary digit which may take either the logical "1" or the logical "0" state.

NOTE:

Programmers are cautioned to program any "X" bits in the select or output bytes as logical "0" so that future expansion will not invalidate preceeding software.

- Capitalized letters other than "X" represent command or status bits which may be present in conjunction with the bits defined.
- Byte bit numbering represents the positional power of two represented by the binary digit.
- When bit position is unspecified for a list of bits or bytes, the ordering is from most to least significant.
- Output and input are defined relative to the 2400 processor.

COMMAND SEQUENCE

Formats

Operations are initiated with an "External Function on Channel" instruction with only the following formats defined for the OP2 item.

Select, dummy, command 1, command 2
Select, dummy,
Select

If the select byte is responded to by the controller, one byte of status is returned for each of the function bytes.

Select Byte

The select byte is used to address a unique controller out of the 16 possible controllers on an I/O channel.

Format 0SSSSXXX - The logical "0" at bit 7 identifies the byte as select. Bits 6 through 3 are a bit pattern unique for each controller on the channel.

Controller selection takes place when the bit pattern in the select byte matches the state of a switch selectable decoder. The decoder has the capability of decoding any of the 16 possible states of the select byte.

After selection is accomplished, the controller will respond to following bytes with bit 7 a logical "1" as commands.

If the controller has a negative response to a select byte, the controller will deselect and ignore all ensuing non-select commands.

Dummy Command

The dummy command is formatted as 1XXXXXXX and has no effect on the controller. Its function is to allow a 4th status byte of status to be sent to the processor.

Command 1

Command one is utilized to generate mode commands to modify the internal mode of the Card Reader/Punch or to generate a binary modifier for the read command in a read binary operation.

Feed Mode Commands

Format: 10 0AB 000

Feed mode commands will be used to change the feed mode condition of the Card Reader/Punch.

Select Non-Feed Mode A=0, B=1

Acceptance of the select non-feed mode command will cause the Card Reader/Punch *not* to feed a card after completion of a punch command operation.

Select Normal Mode A=1, B=0

Acceptance of the select normal mode command will cause the Card Reader/Punch to revert to the normal mode where a card will automatically feed after completion of a punch operation.

Stacker Select Command

Format: 10 000 CD0

Stacker select commands will be used to change the mode of stacker selection: (a) stacker 1 on normal error free cards, and (b) stacker 2 for cards where skew error occurs on read or echo check error occurs on punch.

Select Reject Stacker C=0, D=1

The select reject stacker command will cause all the cards fed after acceptance of this command to go to stacker 2.

Select Normal Stacker C=1, D=0

The select normal stacker will cause the Card Reader/Punch to return to its normal mode of stacking cards, which is all cards go to stacker 1 on a feed operation except those where a reader/punch error occurs. Those cards go to stacker 2.

Binary Command

Format: 10 000 001

The binary command will result in a read binary operation if the command 2 byte is a read command.

Dummy Command

Format: 10 000 000

A dummy command in byte 3 will have no effect on the controller. Its function is to allow a status byte to be sent to the processor.

Command 2

Command 2 is utilized to initiate a read or punch operation or initiate a mode command. It also depends (in most cases) on the results of the command 1 byte.

NO-OP & Card Reader/Punch Mode Command

Format: 10 000 100

This command will cause the execution of the command in byte 1 (except for the binary command which requires a read in command 2). If command 1 is a dummy command, this byte will result in NO-OP, a read, and clear status operation.

Punch Command

Format: 10 000 001

This command enables the controller to accept data to be punched from the processor. Up to 80 bytes of data will be accepted. Any data in excess of 80 characters will be ignored in the non-feed mode. In the normal mode, the 81st byte of data will terminate the data transfer.

Read Command

Format: 10 000 010

This command enables the controller to accept data from the Card Reader/Punch and send it to the processor. Up to 80 bytes of data will be accepted. The 81st character will result in a termination of the operation.

Illegal Commands

Illegal commands are of the following types:

- A format of 10 000 111, 10 000 110, 100 000 101, 10 000 011 or 10 000 000 in command 2 will result in an illegal command.
- A byte in command 1 where A=B=1 or C=D=1 will result in an illegal command.
- Any parity error on command bytes results in an illegal command.

STATUS BYTES

One status byte is returned for each select, dummy, or command byte executed. If a 4-byte function is executed, status will be returned in the following order (if the control unit is not busy): Prior Status 1, Prior Status 2, New Status 2, and New Status 1

Prior Status 1

Prior Status 1 returned in response to the select byte. These status bits retain the logical "1" status conditions of New Status 1 from the last Command 2 executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time. (Busy is an exception.) All eight bits may not simultaneously be logical "0". This state indicates the processor did not receive a status response and internally generated a response of zero.

End of File (EOF)

Format: 1X0XXXXX

The end-of-file status bit will be indicated when the hopper goes empty and the last card to be read enters the read station. It will remain activated as a status bit until the controller assumes not ready.

Selected

Format: X10XXXXX

Logical "1", if selected.

Busy

Format: 00100000

Busy indicates that the controller is presently performing a previously initiated operation. When Busy is present, it inhibits gating of any other status bits (in either status 1 or status 2) but does *not* clear them from their holding logic. Thus, when the Busy condition disappears, its bit becomes a logical "0" and the remainder of the bits takes on their prior states plus any logical "1" accumulations.

NOTE:

Logical "0" status of any bits other than Busy are meaningless, unless Busy is known to be logical "0".

Intervention Required

Format: XX01XXXX

Operator assistance is required before the Card Reader/Punch can attain an operable state. When in this state, the INOP light on the Card Reader/Punch will be illuminated.

Misfeed or Illegal Data

Format: XX0X1XXX

This bit will occur if a pick failure occurs or a data byte has been received after the loss of Ready status due to operator intervention between the issuance of a read or punch command and the reception of a data byte. Discrimination between illegal data and pick failure can be made by the examination of the second prior status byte. Recovery from the illegal data status condition can be made by re-issuance of the command and data operations.

Equipment Check

Format: XX0XX1XX

An illegal command byte 2 was detected, parity error on command 1 or command 2, or a validity check occurred on the prior read operation.

Data Error

Format: XX0XXX1X

A reader/punch error occurred on the prior data transfer.

Ready

Format: XX0XXXX1

Command bytes 1 and 2 were implemented. If Equipment Check or Intervention Required conditions are present, Ready must be logical "0".

Prior Status 2

Prior Status 2 returned in response to the dummy byte. These status bits retain the logical "1" status conditions of new status 2 from the last command 1 executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time.

Stacker Normal

Format: 01XXXXXX

Indicates the stacker mode of the Card Reader/Punch has been selected.

Normal Feed Mode Status

Format: X1XXXXXX

Indicates the Card Reader/Punch is in the feed-after-punch mode.

Illegal Data

Format: XX1XXXXX

Indicates the receipt of a data byte when the controller is not ready.

Jam

Format: XXX1XXXX

Indicates a jam condition in the Card Reader/Punch which also results in an Intervention Required status.

Chips

Format: XXXX1XXX

Indicates that the chip bin is full. This bit implies an Intervention Required status.

Stacker Full

Format: XXXXX1XX

Indicates that either stacker one or stacker two is full. This bit implies operator intervention is required.

Hopper Empty

Format: XXXXXX1X

Indicates that the hopper is empty. This bit implies that an intervention is required.

Card Reader/Punch Ready

Format: XXXXXXX1

Indicates that the Card Reader/Punch is available and ready to process data.

New Status 1

New Status 1 is returned in response to command 2. Command 2 clears the states of prior status 1, so that new status 1 bits returned are conditions at the time command 2 is executed. Status bits are the same as described earlier for Prior Status 1. Both Busy and Data Error are defined so that neither can be a logical "1" for new status 1

New Status 2

New Status 2 is returned in response to command 1. Command 1 clears the status of Prior Status 2, so New Status 2 bits returned reflect existing conditions at the time command 1 is executed.

DATA TRANSFER

The output data transfer is initiated with an "Initiate Output on Channel Instruction." If the controller is not prepared for data transmission because a previous command was not previously executed, the controller will return a Function Acknowledge signal in response to the first Data Request signal received.

If the interval between data requests is 15±3 microseconds, the controller will terminate the data transfer operation. A data request appearing after the termination will result in a "function acknowledge" to be returned to the processor.

NOTE:

At least one data byte must be received; the controller will remain Busy and await the data transfer indefinitely.

If more than 80 characters are requested by the processor in the normal mode, the controller will return a function acknowledge to the 81st character. In a non-feed mode punch operation, the characters in excess of the 80th column are ignored. No special status bit will be set to indicate this truncation of buffer conditions.

A parity error occuring on a card punch data transfer will result in an Equipment Check Status.

A master clear due to the depression of the clear switch or the activation of an interlock will terminate the operation.

Status Summary

Table C-7.2 gives a summary listing of the bits in Status Bytes 1, 2, 3, and 4.

Table C-7.2. Model 2458 Card Reader/Punch Status Bytes and Bits

BYTES 1 & 4		BYTES 2 & 3	
2^7	EOF	2^7	STACKER NORMAL
2^6	SELECTED	2^6	NORMAL FEED MODE
2^5	BUSY	2^5	ILLEGAL DATA
2^4	INTERV REQD.	2^4	JAM
2^3	MISFEED	2^3	CHIPS
2^2	EQUIP CK	2^2	STACKER FULL
2^1	DATA ERROR	2^1	HOPPER EMPTY
2^0	READY	2^0	ON LINE

C–8
CONTROLLER
FOR MODEL 2467 PAPER TAPE READER

PROGRAMMING INSTRUCTIONS

Programming of 2467 Paper Tape Reader is similar to all other 2400 peripherals by using the corresponding commands and status bytes. Select code is determined by the positions of the four select switches and the select functions by consecutive FRQ and DRQ from the SYSTEM 2400 Processor.

Commands

COMMAND	BYTE 1	BYTE 2	
NO-OP	SEL	204	
READ NORMAL FORWARD	SEL	202	
READ NORMAL BACK	SEL	302	
READ FAST FORWARD	SEL	212	
READ FAST BACK	SEL	312	
MOTION NORMAL FORWARD	SEL	222	(EOF)
MOTION NORMAL BACK	SEL	322	(EOF)
MOTION FAST FORWARD	SEL	232	(EOF)
MOTION FAST BACK	SEL	332	(EOF)

Command Byte

The command byte consists of 8 bits, numbered from 0 to 7.

Bit 7 Not used.

Bit 6 0 = Forward direction of paper tape movement.
1 = Backward direction of paper tape movement.

Bit 5 0 = No Interlock (unit remains remote)
1 = Interlock (unit goes local after execution of instruction and paper tape is positioned after last data character read).

Bit 4 0 = Data Transfer to CPU
1 = No Data Transfer (termination of function by ALARM, NO SPROCKET, or 8 consecutive BLANKS)

Bit 3 0 = Normal Speed (750 ch/s)
1 = High Speed (approx. 2000 ch/s)

Bit 2 1 = No-OP

Bit 1 1 = Tape motion

Bit Ø 1 = Not Used

Status Byte

Bit 7 EOF (End of File)
This status bit is placed when more than 8 consecutive blank characters are read after a non-blank character.

Bit 6 Selected and *not* busy
Status for indication of proper selection. No command is in execution.

Bit 5 Busy
Command in execution

Bit 4 Operator intervention required
Status occurs when ALARM is indicated or unit is LOCAL.

Bit 3 Not 8 tracks selected
Status for indication of tape guide position

Bit 2 Channel error
Status indicates channel parity error or illegal command

Bit 1 Data Error (Dark check, sprocket sequence check)
Status active when ERROR is lit.

Bit 0 Ready
Status for indication of Paper Tape Reader ready.

C–9
CONTROLLER FOR
MODEL 2468 PAPER TAPE PUNCH

PROGRAMMING INSTRUCTIONS

Programming of the 2468 Paper Tape Punch is similar to all other 2400 peripherals by using the corresponding commands and status bytes.

Commands

201 Write
204 No Op
241 Write Binary

Command Byte

The command byte consists of 8 bits, numbered for 0 to 7.

Bit 7	Not used
Bit 6	Not used
Bit 5	0 = No interlock 1 = Interlock (unit goes LOCAL after execution of instruction)
Bit 4	Not used
Bit 3	Not used
Bit 2	NO-0P
Bit 1	Not used (gives invalid operation)
Bit 0	Transport and punch paper tape

Select code is determined by the positions of the 4 select switches; selection function is determined by FRQ and DRQ.

Status Byte

Bit 7	Not used
Bit 6	Selected and not busy Status for indication of proper selection. No command is in execution.
Bit 5	Busy Command in execution
Bit 4	Operator intervention required Status occurs when ALARM is indicated or unit in LOCAL.
Bit 3	Not 8 tracks selected Status for indication of tape guide position

Bit 2	Channel error Status indicates channel parity error or illegal command
Bit 1	Status active when PAPER LOW is lit
Bit Ø	Ready Status for indication of Paper Tape Punch ready.

C—10
CONTROLLER (FEATURE 180)
FOR MODEL 2411, 2413, 6401, AND 6403 DATA-RECORDERS

COMMAND CODES

SELECT

OS SSS XXX — S = Switch Selectable (Controller Address)

X — Data Recorder Unit Select Code

76 543 210 — Bit position

COMMAND	BYTE 1	BYTE 2
Read	Select	202
Read Retry	Select	212
Write	Select	201
Write Retry	Select	211
Write Tape Mark	Select	301
Write Tape Mark Retry	Select	311
No Op	Select	204

STATUS CODES (Bytes 1, 2, 3, and 4)

Bit Position	Meaning
7	Tape Mark Detected
6	Selected and Not Busy
5	Busy — if present, all other bits blocked
4	Intervention Required — Off Line, or Command incompatible with state of Entry or Verify
3	Short Transfer
2	Equipment Check — Parity Error, Illegal Command, DR memory error
1	Data Error — Tape Error
0	Ready

COMMAND CODES DESCRIPTION

SELECT – The Select command is issued by the Processor to select a particular device on an I/O channel. All devices on that channel must look at the select code and determine if it is for them. The selected device will logically connect to the I/O channel, and any other device that may have been connected will disconnect. Once selected, a device will remain selected until it sees a select code not for itself or a channel clear.

After initial selection, the select command is used to input the status of previous operations.

In the DATA-RECORDER Controller, there will be a busy period following each data transfer, waiting for the DATA-RECORDER to become ready again. During this period, any status attempt will give only the Busy Bit (bit 5). All attempts to get status on the previous data transfer must be done by a select command because an operational command will clear out any existing bits and give status on that command only.

When a data transfer has been initiated with one of the DATA-RECORDERS on a controller handling multiple DATA-RECORDERS, status must be taken with a Select command that has the same unit select code as the one that initiated the transfer. This must be done before addressing other DATA-RECORDERS on the same controller.

WRITE (201) – This command prepares the controller for an output operation. Following a write command, at least 1 character must be transferred or the controller will remain busy until a channel clear is received.

If the buffer length is shorter than the thumbwheel switch setting, the record will be space-filled to the end. If the buffer length is longer, the buffer will be terminated at the switch set position and a short transfer status will be returned.

READ (202) – This command prepares the controller for an input operation. Following a Read command, at least 1 character must be transferred or the controller will remain busy until a channel clear is received.

If the buffer length is shorter than the thumbwheel switch setting, the DATA-RECORDER will cycle out to the end of record and a short transfer status will be returned. If the buffer length is longer, the buffer will be terminated at the point of the switch setting. No short transfer status will be returned.

READ RETRY, WRITE RETRY, WRITE TAPE MARK RETRY (212, 211, 311) – These commands are similar to Read and Write, except tape is backspaced one record before reading or writing.

WRITE TAPE MARK (301) – This command causes a tape mark to be written on tape; no data transfer is required.

NO-OP (204) – This command provides a means of clearing existing status bits after the situation has been corrected, such as Intervention Required. It should also be used to initialize an I/O routine because the first status will probably contain an Intervention Required bit.

STATUS CODES DESCRIPTION

7 – Tape Mark Detected. A tape mark was sensed on the previous read operation.

6 – Selected. This bit will be returned in the status of a successful connect and with any other status presented, unless the Busy bit is present.

5 – Busy. When presented, all other status bits are locked out. This bit indicates that the controller is still working on a previous command. Any existing status conditions are stored until not busy, provided that status is taken by a select function.

4 – Intervention Required. This bit indicates one or more of the following conditions, all of which will require manual intervention:

- OFF-LINE,
- a command incompatible with the state of Entry or Verify,
- Read command without the presence of Read Ready.

3 – Short Transfer. A write buffer longer (or a read buffer shorter) than the thumbwheel switch setting that was attempted.

2 – Equipment Check. This bit indicates a parity error on a command byte or data transfer, or an Illegal Command, or a memory error from the DATA-RECORDER.

1 – Data Error. This bit indicates a tape error detected by the DATA-RECORDER.

0 – Ready. Indicates that the command received was a legal command (at least one bit, but no more than one of the lower 3 bits) and the DATA-RECORDER is ready to accept this command. The Ready bit in the status for a select command is present from the previous operation and has no meaning for any future commands.

C–11
CONTROLLER
FOR MODEL 2436, 2436-I, 2437, 2437-I, 2438, 2438-I, 2439, and 2439-I MAGNETIC TAPE UNITS

CHANNEL COMMUNICATION

Channel communication with a particular device is organized into three periods: initial selection, data transfer, and terminating status.

Initial selection and terminating status are accomplished by use of an External Function command in the program. Data transfer is initiated by an input/output instruction which sets up buffer control words describing the area of memory which the selector channel inputs or outputs data.

INITIAL SELECTION

The External Function, which enables initial selection of a device, describes three fields of information. The first operand defines the channel to operate on. The second operand defines the area from which the select code and command bytes are output from. The third operand defines the receiving field for storage of status, which is returned in response to the select and command bytes defined by the second operand. One byte of status is returned for each byte defined by the second operand. The input status field should never be longer than required by the second operand.

When an External Function command is issued by the program, all devices on the selected channel decode the first byte of information transmitted via the channel as a select code. If a device was selected and sees a select code for another device, the selected device deselects or disconnects from the channel and the device being addressed by the select code connects to the channel.

If the select code sent does not match any of the devices attached to the channel, no devices respond and the Processor terminates the external function instruction by storing a status of all zero's in the first byte of the status field.

Upon decoding a correct select code, the selected device returns a byte of status to the Processor and prepares to receive the necessary command bytes for initiating an operation. The number of command bytes necessary is dependent upon the design of the Controller and the function desired. The tabular presentation below indicates the acceptable formats for tape transport Models 2436, 2436-I, 2437, 2437-I, 2438, 2438-I, 2439, 2439-I.

Format 1

(4 byte)	**External Function**	**Initiating an Operation**
	Second operand data	
		Select 1, dummy command 2, modifier command 3, command 4
	Third operand data	
		Prior status 1, prior status 2, new status 2, new status 1

Format 2

(2 byte)	External Function	Seeking Termination Status
	Second operand data	
		Select, dummy command
	Third operand data	
		Prior status 1, prior status 2

All operations on the tape transport are initiated via format 1. The operations break down into two types: Read and Write commands which enable a data transfer, and immediate commands which cause tape motion like tape erase, search tape mark, write tape mark, Rewind, or Backspace. An immediate command which does *not* cause tape motion but allows updating of status is the No-Op command.

In Table C-11.1, the available operations are listed. Modifier command byte and the select byte are broken down into a binary pattern to facilitate defining the available combinations, whereas command byte 4 and the dummy command are given in tri-octal notation.

Table C-11.1. Tape Drive Controller Commands

Format =	Select 0X XXX 000_2	Dummy Command 200_8	Modifier Command 3 IB CDE FGH	Command 4
Modifier Command 3	**Command 4**	**Description**	**Available Models**	**Available Modifier Bit Combinations**
1B 000 G00	201_8	Write to tape	2436, 2436-I, 2437, 2437-I, 2439, 2439-I,	G-may be active "1" only for 7-track tape drives (2436, 2436-I)
1B 000 000	301_8	Write tape mark	2436, 2436-I, 2437, 2437-I, 2439, 2439-I	
1B 000 000	221_8	Erase tape	2436, 2436-I, 2437, 2437-I, 2439, 2439-I	
1B CD0 0GH	202_8	Read from tape	2436, 2436-I, 2437, 2437-I, 2438, 2438-I, 2439, 2439-I	D, G, and H should be inactive "(0)" if B is active.
1B CD0 00H	212_8	Read backward from tape	Optional 2437, 2437-I, 2439, 2439-I	
1B 000 000	214_8	Backspace 1 record	2436, 2436-I, 2437, 2437-I, 2438, 2438-I, 2439, 2439-I	
1B 000 F00	244_8	Rewind tape	2436, 2436-I, 2437, 2437-I, 2438, 2438-I, 2439, 2439-I	
10 000 000	204_8	No-Op	2436, 2436-I, 2437, 2437-I, 2438, 2438-I, 2439, 2439-I	

Modifier Bit Definition for Command 3

B = 2^6 — Interrupt upon completion. Initiation of a command other than No-Op with this bit active will enable an interrupt to be returned upon completion of the command. This capability is presently only available with the Model 2439, 2439-I Tape Drives.

C = 2^5 — Search Tape Mark. Initiation of a Read command with this bit active modifies the Read command to cause it to scan the tape in the selected direction for a tape mark. When searching tape marks on an odd-parity, seven-track tape, parity errors will be detected since tape marks are written in even parity and data is written in odd parity; therefore, the status of tape mark active and data check is a valid status and indicates a tape mark has been found. This status may also occur when bad data is detected in a normal record prior to finding a tape mark.

D = 2^4 — Parity Identification. Initiation of a Read command with this bit active will enable the tape controller to insert an octal 237_8 code into the data stream any time a frame of data from tape contains incorrect parity. This enables the user to pin point the area of a record that is causing read data checks (vertical parity errors).

F = 2^2 — Lockout. This modifier is only available with the Rewind Command and when active causes the tape transport (upon completion of a rewind) to terminate in the LOCAL mode, necessitating operator intervention for further use of the drive.

G = 2^1 — Even/Odd Parity. This modifier bit is available only for use with seven-track tape drives (Models 2436, 2436-I, 2438, and 2438-I. When active with a Read or Write command, all data is checked for even parity. With this bit inactive, all data is checked for odd parity.

H = 2^0 — Low Gain. The low-gain modifier bit enables the programmer to alter the clipping level of the read amplifiers in the tape transport. It is suggested that this bit be alternately activated on each re-read of a failing record. This bit should *not* be active under a normal read; it should only be used during error recovery.

COMMAND DESCRIPTION

Write to Tape — This command prepares the tape transport logic for an output operation. Following initiation of a Write command, the control logic of the tape transport will remain busy until receipt of at least one Data Transfer or a Channel Clear command. Tape motion is initiated upon receipt of the first data character. All data to be included in a record on tape must be transferred under one output instruction from the Processor. Each output instruction must be preceded by an External Function instruction initiating a write, and each output instruction should be followed by a terminating status check to insure correct writing of the information.

It should be noted that when writing on 7-track tape drives that certain precautions must be observed. When writing 7-track even parity, the output buffer must not contain a byte of zeros. Also, the upper two bits (2^6 and 2^7) of each byte of the output buffer must be equal (both zeros or both ones).

Write Tape Mark – This command causes an even-parity code 017_8 to be written on seven-track tapes or as odd-parity code 023_8 to be written on NRZ1 nine-track tape. A gap of 3.5 inches is erased preceding the tape mark, and a normal inter-record gap exists following the tape mark. Tape motion is initiated upon receipt of the command and no data is transferred between Processor and that tape transport logic. The control logic of the tape transport will remain busy until the completion of the operation.

Erase Tape – This command, when initiated, will cause the tape transport to erase 3.5 inches of tape. Tape motion will commence upon receipt of the command and the controller will remain busy until completion of the command.

Read From Tape – This command prepares the tape transport logic for an input operation. Following initiation of a Read command, the control logic will remain busy until receipt of a Data Request or a Channel Clear command. Tape motion is initiated upon receipt of the first data request initiated by an input command. The input field reserved should include sufficient space to store all desired information from one tape record. Each input instruction must be preceded by an External Function initiating a Read command. If modifier bit C is active, the input instruction is *not* necessary and should *not* be executed following the External Function instruction.

Backspace 1 Record – This command causes the tape transport to position tape one record back (toward load point). Tape motion occurs upon command initiation and *no* data transfer is necessary. Read errors detected upon termination are meaningless and should be ignored, since the function of the command is only to reposition tape. Detection of the beginning-of-tape sensor or runaway status terminates the command and indicates no data was found.

Rewind Tape – The Rewind Tape command enables the tape transport to do a high-speed rewind back to load point. Tape motion is initiated upon receipt of the command. The control logic indicates a rewinding status and *not* a busy status during operation of the command. A good termination status will contain the following status bits: Selected, Rewinding, LOCAL, and Beginning-Of-Tape. It may also contain Interrupt and Ready.

NO-OP – This command provides a means of clearing existing status bits, after the situation has been corrected, such as LOCAL mode. It should also be used to initialize an I/O routine because the first status will probably contain a local bit (set while the tape drive was in LOCAL mode). New status can be checked on bytes 3 and 4.

Status – Status information is returned in response to an External Function (EF) instruction. During an operation, status is accumulated and held until cleared by initiation of a new operation. The tape control logic contains 16 bits of unique status returned (in two bytes) in response to an EF instruction.

When an operation is initiated, the first two bytes of status reflect the accumulation of data from the initiation of the last command until the present command being initiated. The second two bytes of status reflect the state of the tape transport upon receipt of the present command.

If the control logic honors the command, a status of "selected" and "ready" will be returned in byte four. If the command cannot be initiated, a status of "selected" and "equipment check" will be returned.

The busy status bit (when active) is returned by itself. This bit is active until the completion of the operation requested. If the last operation was a Read, then busy remains active until after all check characters have been received from tape and checked (insuring the collection of all available status prior to initiation of another command).

STATUS CODES DESCRIPTION (BYTES 1 and 4)

7 – INTERRUPT. Reserved for future use.

6 – SELECTED. This bit will be returned in the status of a successful connect and with any other status presented, unless the Busy bit is present.

5 – BUSY. When presented, all other status bits are locked out. This bit indicates that the controller is still working on a previous command. Any existing status conditions are stored until *not* Busy. When Busy, this bit will be present in all status bytes.

4 – TAPE MARK DETECTED. The previous read operation found a tape mark. An equipment check can also be expected in the status, when reading a 7-channel, odd-parity tape mark.

3 – DATA CHECK. The previous operation found one of the following errors: LRC or CRC error, write-check error detected by the tape drive, or parity error on data read from the tape during Read or Read After Write Check command.

2 – EQUIPMENT CHECK. This bit indicates a parity error on a command byte of an illegal command. It can also initiate a parity error on data read from tape during a Read operation or on data received from the Processor for transfer to tape during a Write operation.

1 – OVERRUN/RUNAWAY. Overrun indicates that the Processor failed to present a DRQ in time for one of the Read or Write clock pulses. Therefore, data may have been lost. Runaway indicates that the previous operation caused tape motion in excess of approximately 8 feet.

0 – READY. This bit indicates the command received was a legal command (at least one bit, but no more than one of the lower three bits) checked on byte 4 only. It also says the LOCAL/REMOTE switch is in REMOTE. This bit will be blocked, if a Write command is issued when no file protect ring is installed.

STATUS CODES DESCRIPTION (BYTES 2 and 3)

7 – SHORT TRANSFER. The previous Read operation did *not* take all the data that was available in the record on tape.

6 – 7 CHANNEL. This bit indicates the tape drive that is attached is a 7-track drive (Model 2436 or 2436-I) or is in 7-track mode if 7-9 channel switchable drive (Model 2438 or 2438-I) is being used.

5 – WRITE ENABLED. This bit indicates that a file protect ring is on the mounted tape.

4 – RUNAWAY. The previous operation caused tape motion in excess of approximately 8 feet, indicating lack of data on tape. This bit is presented so that Runaway can be distinguished from Overrun.

3 – REWINDING. A Rewind operation has been initiated. During a Rewind, the tape drive goes to a LOCAL condition and this bit is to determine why the LOCAL status is present. These bits will *not* clear until another command is issued.

2 – END OF TAPE. This bit indicates an end-of-tape reflective marker has been sensed. This status is set in a forward direction.

1 – BEGINNING OF TAPE. This bit indicates a load point marker has been sensed.

NOTE:

This bit will also be present in the terminating status of the Write or Read of the first data block from load point.

0 – LOCAL. LOCAL/REMOTE switch is in the LOCAL position or a rewind operation is in process.

Table C-11.2. Tape Drive Commands and Status Bits

COMMAND	BYTE 1	BYTE 2	BYTE 3	BYTE 4
READ	SELECT	200	200	202
READ LO GAIN	SELECT	200	201	202
READ EV. PAR.	SELECT	200	202	202
READ EV. PAR. LO GAIN	SELECT	200	203	202
READ PE.	SELECT	200	220	202
READ LO GAIN PE.	SELECT	200	221	202
READ EV. PAR. PE.	SELECT	200	222	202
READ EV. PAR. LO. GAIN PE.	SELECT	200	223	202
READ BKWD.	SELECT	200	200	212
READ BKWD. LO GAIN	SELECT	200	201	212
READ BKWD. EV. PAR.	SELECT	200	202	212
READ BKWD. EV. PAR. LO GAIN	SELECT	200	203	212
READ BKWD. LO GAIN PE.	SELECT	200	221	212
READ BKWD. EV. PAR. LO GAIN PE.	SELECT	200	223	212
SEARCH TAPE MARK	SELECT	200	240	202
SEARCH TAPE MARK EV. PAR.	SELECT	200	242	202
SEARCH TAPE MARK BKWD.	SELECT	200	240	212
SEARCH TM. BKWD. EV. PAR.	SELECT	200	242	212
WRITE	SELECT	200	200	201
WRITE EV. PAR.	SELECT	200	202	201
WRITE TM.	SELECT	200	200	301
REVERSE ONE BLOCK	SELECT	200	200	214
REWIND	SELECT	200	200	244
REWIND & LOCKOUT	SELECT	200	204	244
ERASE TAPE	SELECT	200	200	221
NO OP	SELECT	200	200	204

STATUS	
BYTES 1 & 4	BYTES 2 & 3
2^7 INTERRUPT	2^7 SHORT TRANSFER
2^6 SELECTED NOT BUSY	2^6 7 CHANNEL
2^5 BUSY	2^5 WRITE ENABLED
2^4 TM DETECT	2^4 RUNAWAY
2^3 DATA CK	2^3 REWINDING
2^2 EQUIP CK	2^2 EOT
2^1 OVERRUN/RUNAWAY	2^1 BOT
2^0 READY	2^0 LOCAL

C–12
CONTROLLER
FOR MODEL 2471 AND 2473 DISC STORAGE DRIVES

SELECTION AND DESELECTION

Disc Controller – If the Controller receives a control word from the SYSTEM 2400 Processor which (according to the input/output specifications) represents a select/deselect command, the Controller will respond if the select code is recognized as being assigned to the Controller and contains odd parity. In the event the select command contains even parity or does not contain the select code for the Controller, no response is issued from the Controller to the Processor.

Disc(s) – Select information (specifying which disc unit of a possible four is being addressed) is contained in the same command byte that the Controller recognizes to be a select byte. A change from one disc unit to another should be made only when the Controller is in the Ready state.

FUNCTIONS

There are eight functions which the Disc Controller will perform upon request from the Processor, and one function which is initiated manually. The first eight are via the disc commands listed in Table C-12.1 and discussed in following subparagraphs. The latter "Load Mode" is a manually initiated function causing the Controller to perform a normal Read operation at Sector 0, Head 0, Track 0, Disc 0. This allows the capability to load one track of data on a boot (5,120 bytes).

Table C-12.1. Disc Commands

COMMAND	BYTE 1	2	3	4
WRITE PROTECT	05X	220	- - -	- - -
MASTER CLEAR	05X	210	- - -	- - -
SEND STATUS	05X	200	- - -	- - -
RESTORE	05X	214	- - -	- - -
SEEK	05X	204	2TU	2TL
WRITE	05X	201	20H	2SS
READ	05X	202	20H	2SS
COMPARE	05X	212	20H	2SS
VERIFY	05X	222	20H	2SS

NOTE:

TU – Track Address (upper 4 Bits).
TL – Track Address (lower 4 Bits).
H – Head Select (0 = upper, 1 = lower).
SS – Starting Sector. ($0\text{-}47_8$).
X – Disc Unit (0-3)

LEGAL TRACK ADDRESS
$000 - 202_{10}$

WRITE PROTECT $(220)_8$ – This prevents writing on disc until the operator manually switches off the write-protect.

MASTER CLEAR $(210)_8$ – The Controller is forced to the Ready State.

SEND STATUS $(200)_8$ – The Controller transmits its status to the Processor.

RESTORE $(214)_8$ – The Controller causes the moveable head disc specified by the Select Command to return its head boom to track zero. This command clears any "seek incomplete" condition that may be present in the selected disc.

WRITE $(201)_8$ – The Controller is conditioned to accept data (byte serially) upon request from the Processor and to store this data onto the disc (bit serially) starting at the surface and sector specified by bytes three and four of the Write Command Sequence and ending when Data Requests from the Processor cease or the end of the track is reached.

READ $(202)_8$ – The Controller is conditioned to: recover data from the disc starting at the surface and sector specified by bytes three and four of the Read Command sequence, and to transmit upon request this data (byte serially) to the Processor until requests cease or the end of the track is reached.

An odd parity bit is generated for each byte transmitted to the Processor.

SEEK $(204)_8$ – The Controller causes the head boom to be moved to a position specified by bytes three and four of the seek command sequence.

VERIFY $(222)_8$ – The Controller is conditioned to accept data (byte serially) from the Processor, to compare each byte in turn with data bytes from the disc, to condition a bit in the Controller Status Word as a result of the compare. The compare bit is set by the Verify Command Sequence and continues until the Processor ceases requesting comparisons or until the end of the track is reached.

COMPARE $(212)_8$ – The Controller is conditioned to accept data (byte serially) from the Processor, to compare each byte in turn with data bytes from the disc, and to condition a bit in the Controller Status Word as a result of the compare. A binary zero byte (000_8) presented by the Processor forces the equal condition to exist at the time of the test, thereby, masking status information from the disc on that byte. The compare bit is set during Compare Command Sequence and remains set as long as the compare test indicates equal. Comparisons begin at the surface and sector specified by bytes three and four of the Compare Command Sequence and continue until the Processor ceases requesting compares or until the end of the track is reached.

Status Codes Description

The Disc Controller will set the appropriate bits in the status word for conditions that are described below. The status word contains current status information about the Controller and the disc(s). Figure C-12.2 illustrates the status word.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	VALID STATUS 1=VALID	COMPARE EQUAL 1=EQUAL	ILLEGAL COMMAND 1=ERROR	OVERRUN 1=ERROR	EQUIPMENT ERROR 1=ERROR	PARITY ERROR 1=ERROR	READY 1=READY

Figure C-12.2. Disc Controller Status Word Format

VALID STATUS WORD (1=Valid Status) — This bit is always set in the status word. It provides the Processor with the capability of differentiating between a valid status word transmitted from the Controller in response to a Processor request and an artificail status word of all zeros (000_8), which is generated by the Processor when it does not receive a response from a peripheral device it has interrogated. The valid status bit occupies the 2 position within the status word.

COMPARE EQUAL (1 = Equal Compare) — This bit is set during the fourth byte of either a Verify or Compare command sequence and remains set unless cleared by a *not* equal condition at the time of a test. The compare bit occupies the 2^5 position in the status word.

ILLEGAL COMMAND (1 = Illegal Command) — This bit in the status word is set if the Controller receives a command other than the six valid commands, or if the command byte contains even parity. The condition remains until cleared by receipt of a valid command from the Processor. No operation involving the Disc(s) is initiated while this condition is active. This bit occupies the 2^4 position within the status word.

OVERRUN (1 = Overflow) — This bit in the status word indicates a request from the Processor for operation in a sector *not* available for an operation. The availability of a sector is determined by two criteria: the sector address requested must *not* exceed 47_8, and the sector of operation must *not* be sector zero if the operation is multi-sectored and operation in sector 47_8 has just been completed. The overrun condition remains active until a valid sector of operation is requested by the Processor. The overflow bit occupies the 2^3 position within the status word.

EQUIPMENT ERROR (1 = Error) — This bit in the status word indicates the presence of a Seek Incomplete Condition or a Logical Address Interlock Condition (both are generated in the disc). The Seek Incomplete Condition indicates that, due to some malfunction, a seek operation was *not* completed. This condition will be maintained until a Restore Command 214_8 is issued by the Processor. The Logical Address Interlock Condition indicates that a command to move the heads to a track position greater than 202_{10} has been received, and the command is therefore, *not* executable. The Seek Command is *not* executed. This condition exists until the status word has been sampled. The equipment error bit occupies the 2^2 position in the status word.

PARITY ERROR (1 = Error) – This bit of the status word is set if: during the transmission of data from the Processor to the Controller, a Data Byte is found to contain even parity, if during a Read, Verify, or Compare Operation an error is found in the data transmitted from the disc to the Controller. The error condition exists until the fourth byte of another command sequence is received by the Controller. Parity Error Status is transmitted as bit 2^1 of the status word.

READY (1 = Ready) – This bit position of the status word indicates that the Controller and the selected disc are able to execute an instruction. Ready is disabled during the fourth byte of a Seek, Write, Read, Verify, or Compare instruction and on the second byte of a Restore Instruction. The file ready line from only the selected disc is included in the ready bit. This bit occupies the 2^0 position in the status word.

BUFFER TERMINATE RESPONSE – If the Illegal Command or Overrun condition is active, a buffer terminate command is issued by the Controller in response to a data request from the Processor. No status is transferred with the terminate command, but the indicated status conditions are available to the Processor upon request.

Address

Table C-12.3 lists the 2471 track address codes utilized in bytes three and four of the disc commands.

Table C-12.3. Disc Track Address Table

BYTE 3	BYTE 4	TRACK
200	200	000
200	217	015
201	200	016
201	217	031
202	200	032
202	217	047
203	200	048
203	217	063
204	200	064
204	217	079
205	200	080
205	217	095
206	200	096
206	217	111

BYTE 3	BYTE 4	TRACK
207	200	112
207	217	127
210	200	128
210	217	143
211	200	144
211	217	159
212	200	160
212	217	175
213	200	176
213	217	191
214	200	192
214	212*	202

*MAXIMUM ADDRESS

C-13
CONTROLLER
FOR MODEL 2443 AND 2444 CHAIN PRINTERS

COMMAND CODES

SELECT

OS SSS XXX — S = Switch Selectable

76 543 210 — Bit Position

WRITE	201
NO-OP	204

STATUS CODES

Bit Position	Meaning
7	Not used
6	Selected and not busy
5	Busy — If present, all other bits blocked
4	Intervention Required — Off-line, hood open, form check, blown fuse, print check, or carriage open
3	Not used
2	Equipment Check — Parity Error, Illegal Command
1	Data Error — Printer Parity Error — optional
0	Ready

COMMAND CODES DESCRIPTION

SELECT — The Select command is issued by the Processor to select a particular device on an I/O channel. All devices on that channel must look at the select code and determine if it is for them. The selected device will logically connect to the I/O channel and any other device that may have been connected will disconnect. Once selected, a device will remain selected until it sees a select code not for itself or a channel clear.

After initial selection, the select command is used to input the status of previous operations.

In the Printer Controller, there will be a busy period following each data transfer, waiting for the Printer to become ready again. During this period, any status attempt will give only the Busy bit (bit 5). All attempts to get status on the previous data transfer must be done by a Select command because an operational command will clear out any existing bits and give status on that command only.

WRITE (201) – This command prepares the controller for an output operation. Following a write command, at least 1 character must be transferred or the Controller will remain busy until a channel clear is received. With the Model 2443/2444 the first character for each line will be interpreted as a line advance code.

Line advance codes are as follows:

040 – no space

041 – space 1 line

042 – space 2 lines

043 – space 3 lines

etc.

000 – skip to channel 1

001 – skip to channel 2

002 – skip to channel 3

etc.

With the Model 7520 (Feature 130) the standard MDS Edit ECC codes are applicable when the Printer is in Edit mode.

NO-OP (204) – This command provides a means of clearing existing status bits after the situation has been corrected, such as Intervention Required. It should also be used to initialize an I/O routine, because the first status will probably contain an Intervention Required bit. If a data error has been detected, a No-Op command must be sent in order to clear the error condition in the printer.

STATUS CODES DESCRIPTION

7 – Not Used.

6 – Selected. This bit will be returned in the status of a successful connect and with any other status presented, unless the busy bit is present.

5 – Busy. When presented, all other status bits are locked out. This bit indicates that the controller is still working on a previous command. Any existing status conditions are stored until *not* Busy, provided status is taken by a Select function.

4 – Intervention Required. This bit indicates one or more of the following conditions, all of which will require manual intervention:

Off-line, hood open, form check, blown fuse, print check or carriage open.

3 – Not Used.

2 – Equipment Check. This bit indicates a parity error on a command byte or data transfer, or an illegal command.

1 – Data Error (Used in Model 2443/2444 only). This bit indicates a parity error detected by the printer. A No-Op command must be issued in order to clear this condition.

0 – Ready. Indicates that the command received was a legal command (at least one bit, but no more than one of the lower 3 bits), and the Printer is ready to receive data. The ready bit in the status for a Select command is present from the previous operation and has no meaning for any future commands.

Summary

Table C-13.1 gives a summary listing of applicable command and status codes.

Table C-13.1. Model 2443 and 2444 Printer Commands

COMMAND	BYTE 1	BYTE 2
PRINT NO OP	SELECT SELECT	201 204
STATUS		
BYTES 1, 2, 3, 4		
2^7 NOT USED 2^6 SELECTED, NOT BUSY 2^5 BUSY 2^4 INTERVENTION REQUIRED	2^3 NOT USED 2^2 EQUIP. CK. 2^1 DATA ERROR 2^0 READY	

C-14
CONTROLLER
FOR MODEL 2445 AND 2446 PRINTERS

DEFINITIONS

The capitalized letter X signifies a binary digit which may take either the logical "1" or "0" state.

NOTE:

> Programmers are cautioned to program any X bits in select, command, or output bytes as logical "0" so that future logic expansion will not invalidate preceding software.

Capitalized letters other than X represent command or status bits which may be present in conjunction with the bit(s) defined.

Byte bit numbering represents the positional power of two represented by the binary digit (e.g., bit 4 is 000 10000).

VFU is the abbreviation for the perforated tape-controlled vertical format unit used to control the movement of printer forms.

When the position is unspecified for a list bit or bytes, the ordering is from most to least significant (e.g., if the list ABC is defined to occupy bit positions 5 through 3; A is positioned at bit 5, B at 4, and C at 3).

Output and input are defined relative to the processor (e.g., output refers to a transfer from processor to printer).

COMMAND SEQUENCE

Operations are initiated with an "External Function or Channel" instruction with only the following formats defined for the OP2 item:

Select . . . dummy . . . command 1 . . . command 2.

Select . . . dummy.

Select.

If the select byte is responded to by the controller, one byte of status is returned to the processor for each of the function bytes in the OP2 item.

Select Byte

The select byte is used to address a unique controller out of the 16 possible controllers on an I/O channel.

Format: 0SSSSXXX

The logical "0" at bit 7 identifies the byte as select. Bits 6 through 3 are a bit pattern unique for each controller on a channel.

Controller selection takes place when the bit pattern in the select byte matches the state of a switch-selectable decoder. The decoder has the capability to decode any of the 16 possible states of the select byte.

After controller selection is accomplished, the controller responds to following bytes with bit 7 a logical "1" as commands.

If the decoder has a negative response to the select byte, the controller deselects and ignores all ensuing non-select commands.

If a parity error is detected on the Select byte, the controller deselects and ignores all ensuing command bytes.

Dummy Command

The dummy command is formatted as 1XXXXXXX and has no effect on the controller. Its sole functions are to facilitate checking of any following command bytes and gate a status byte to the processor.

Command 1 is utilized to control paper spacing (see Tables C-14.1 and C-14.2). In conjunction with commands selected in Command 2, Command 1 controls the VFU channel to be skipped to or the number of lines to be spaced.

Format: 1ZABCCCC

Skip to Channel A—0

If the command executed is skip to channel, CCCC patterns 1100 through 1111 are undefined and will not be executed. However, if executed, the undefined commands will not result in paper runaway. The value of B will not affect the operation.

Space N Lines A—1

If the command executed is space N lines, all BCCCC patterns are defined. Patterns 00000 through 11111 select, respectively, no spacing through spacing 31 lines.

Z = 1 for Before
Contradicts Hardware manual in use of 2^7 & 2^6
12/74 jcc

Table C-14.1. Printer Controller Forms Spacing Codes

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	DESCRIPTION
1	Z	0	X	0	0	0	0	Feed to VFU Ch 1
1	Z	0	X	0	0	0	1	Feed to VFU Ch 2
1	Z	0	X	0	0	1	0	Feed to VFU Ch 3
1	Z	0	X	0	0	1	1	Feed to VFU Ch 4
1	Z	0	X	0	1	0	0	Feed to VFU Ch 5
1	Z	0	X	0	1	0	1	Feed to VFU Ch 6
1	Z	0	X	0	1	1	0	Feed to VFU Ch 7
1	Z	0	X	0	1	1	1	Feed to VFU Ch 8
1	Z	0	X	1	0	0	0	Feed to VFU Ch 9
1	Z	0	X	1	0	0	1	Feed to VFU Ch 10
1	Z	0	X	1	0	1	0	Feed to VFU Ch 11
1	Z	0	X	1	0	1	1	Feed to VFU Ch 12
1	Z	1	0	0	0	0	0	Suppress paper feed (no line feed)
1	Z	1	0	0	0	0	1	Advance one line
1	Z	1	0	0	0	1	0	Advance two lines
1	Z	1	0	0	0	1	1	Advance three lines
1	Z	1	0	0	1	0	0	Advance four lines
								.
		Through						.
1	Z	1	1	1	1	1	1	Advance 31 lines

Feed Before or After Print: Modifier Bit Z

Bit 2^6 of the VFU byte defines when forms motion occurs. When 2^6 is equal to a logical "Ø", forms motion occurs after the print cycle. If bit 2^6 is set to a logical "1", forms motion occurs after the loading of data and before the print cycle.

Command 2

Command 2 is utilized to initiate printing and/or paper movement. This extent of paper movement, if initiated, is dependent upon the information in Command 1, as summarized in Table C-14.2.

No-Op Command

Format: 1XEX0000

No printing or paper spacing will take place. This command is used principally to read and/or reset status.

Skip or Space Immediate

Format: 10EXG100

Paper Spacing, as defined by command byte 1, will occur immediately. The controller will remain busy until the operation is completed.

Read Aux Panel

Format: 11EXX010

This command enables the controller to input information from a remote operator panel. The 2^6 bit defines the read command as a read of Aux panel. Information from the auxiliary panel is obtained via an input data function. The number of bytes and type of information read is dependent upon the implementation of the panel. (The operator panel referred to here is not a standard feature of printers supplied in 2400 systems.)

Print Command

Format: 10EFG001

Prepare to receive data bytes for printing via the output instruction. After execution of the output instruction, forms motion and printing will occur as described in preceding paragraphs.

Write Aux Panel

Format: 1DXXX001

The D bit active enables the controller to output information (via an output function) to an auxiliary panel, enabling the driving of a remote operator display panel (via the program).

With the D bit active, all information is inhibited from reaching the printer line buffer and enabled to the operator display logic. The number of bytes to be sent and the function of individual bits is defined by the panel utilized.

Modifier Bits

Format: 1XEFGXXX

Interrupt Control: Bit E

Bit E, when active, will enable the controller to interrupt upon completion of a skip or space immediate or print commands. Activation of a switch on the auxiliary panel will cause an interrupt to the processor. If the auxiliary panel switch is depressed during a print or forms motion operation, one interrupt will be sent upon completion of the requested operation.

Bit E, when inactive, disables all interrupts.

Print Illegal Code: Bit F

Bit F, when active, will cause the controller to present a signal to the printer, allowing all illegal characters detected to be printed as a predefined error code.

Bit F, when inactive, will cause the printer to suppress further printer and forms motion. Reference to the appropriate printer documentation is necessary for determining proper error-recovery procedures.

Skip to TOF: Bit G

Bit G, when active, will cause an automatic skip to top of form (TOF) when Spacing N Lines and bottom of form (BOF) is detected.

Table C-14.2. Summary of Model 2445 and 2446 Printer Commands

COMMAND 1	COMMAND 2	DESCRIPTION
1XXXXXXX	1XEX0 000	No-Operation. Clear old status and present new status.
1ZABCCCC	10EXG 100	Skip to Channel or Space N Lines. The command initiates an immediate forms movement without a print cycle.
1ZABCCCC	10EFG 001	Print and Skip to Channel or Print and Space N Lines. This command executes a print cycle and a forms movement in the order defined by bit Z.
1XXXXXXX	11EXX 010	Read Aux Panel. Input the information available at the auxiliary operator's panel.
1XXXXXXX	11EXX 001	Write Aux Panel. This command enables driving indicators at the auxiliary operator's panel.
If the optional auxiliary operator panel is not available, the Read Aux Panel and Write Aux Panel commands will be answered with an Equipment Check status response.		

Illegal Commands

Format: 1XXXXUOV

Command byte 2 will be checked to insure an executable command has been requested. An executable command will not contain simultaneous active bits in positions U and V.

An illegal command is not executed.

Any parity error on a command function (command 1, or command 2) results in an illegal command.

All 4-byte functions are ignored if busy is a logical "1".

Status Bytes

One status byte is returned for each select, dummy, or command byte executed. If a 4-byte function is executed, status will be returned in the following order if the control unit is not busy: Prior Status 1, Prior Status 2, New Status 2, and New Status 1.

Prior Status 1 is returned in response to the select byte. These status bits retain the logical "1" status conditions of New Status 1 from the last Command 2 executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time. (Busy is an exception.) All eight bits may not simultaneously be logical "0". This state indicates the processor did not receive a status response and therefore internally generated a response of zero.

Completion Interrupt

Format: 1X0XXXXX

The interrupt bit will go active upon completion of the last requested operation if selected via command byte 2. The bit indicates a completion interrupt has been sent via the controller to the processor.

Selected

Format: X10XXXXX

Logical "1" if selected.

Busy

Format: 00100000

Busy indicates that the controller is presently performing a previously initiated operation. When busy is present, it inhibits gating of any other status bits (in either status 1 or status 2), but does not clear them from their holding logic. Thus, when the busy condition disappears, its bit becomes a logical "0" and the remainder of the bits takes on their prior status, plus any logical "1" accumulations.

NOTE:

Logical "0" states of any bits other than busy are meaningless unless busy is known to be logical "0".

Illegal Data

Format: XX01XXXX

An undefined data byte has been detected by the printer. As implemented by the printer, detection of this illegal data character may occur in either the load cycle or print cycle of a print operation.

If the status of Illegal Data is returned in response to a print operation, the programmer may repeat the command and data operation to attempt a recovery.

If the error has occurred in the load cycle (indicated by the fact that "Print Error" status is not set), no forms motion or printing has occurred.

If the error has occurred in the Print cycle, "Print Error" active, a partial print operation has occurred and forms motion may or may not have occurred, dependent on whether pre- or post-spacing had been requested.

Intervention Required

Format: XX0X1XXX

Operator assistance is required to set the printer to an operable state. This bit will signify that the yoke is open or that the printer is off-line.

Equipment Check

Format: XX0XX1XX

An illegal Command 2 byte was detected or parity was incorrect on either Command 1 or Command 2.

Print Error

Format: XX0XXX1X

A Print Error condition indicates that the printer, during a print cycle, has detected either an invalid code in its line buffer or a hardware malfunction as defined by "Print Check 2". Recovery is available to the programmer by reloading the buffer and overprinting the line. If forms motion was requested prior to printing, a VFU code of no-lines-space should be requested when overprinting the line.

Ready

Format: XX0XXXX1

Command bytes 1 and 2 were honored. If Equipment Check or Intervention Required conditions are present, Ready must be logical "0".

Prior Status 2 is returned in response to the dummy byte. These status bits retain the logical "1" status condition of new status 2 from the last Command 1 executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time.

Panel Active

Format: 1XXXXXXX

A Panel Active condition indicates that a switch on the optional printer operator panel has been pressed. If the previous command to the controller had enabled interrupts via modifier bit E in Command byte 2, an interrupt would have been sent at the end of the last operation if the switch was pressed during the last operation. If a switch on the panel was pressed while an operation was not in progress, an interrupt would have been sent and the status of panel interrupt would be posted in the controller without bit 2^7 in Status Byte 1 being set.

The status of panel interrupt is cleared after a read of the optional operator panel.

Print Check 1

Format: X1XXXXXX

A Print Check 1 condition indicates a line recoverable hardware malfunction has occurred while in the print cycle. Further printing and forms motion is suppressed. Errors such as data memory parity error, code disc parity error, or invalid code in memory due to hardware malfunction are grouped under this bit.

Print Check 2

Format: XX1XXXXX

A Print Check 2 condition indicates that a hardware malfunction has occurred in the print cycle and the printed line may be destroyed. Further printing and forms motion is suppressed.

Errors such as timing errors or hammer check may cause a Print Check 2 status. See applicable printer documentation for specific print check 2 conditions.

Fault

Format: XXX1XXXX

The Fault status is active any time a condition is sensed rendering the print mechanism inoperative.

Typical fault conditions are: mechanical interlocks, loss of power and blown fuses. See appropriate printer documentation for specific fault conditions.

VFU Check

Format: XXXX1XXX

The VFU Check Condition indicates a forms runaway or a paper break above the print line.

Paper Low

Format: XXXXX1XX

A Paper Low condition indicates the absence of paper below the print line. Upon first detection of this condition, the printer will go off-line. The operator then has the option of placing the printer on-line, allowing the completion of the last form, or of changing forms immediately and returning the printer to a ready state.

If the operator elects to complete the printing of the last form, a Paper Low condition will be returned for each line printed until a channel 1 is sensed from the VFU, at which time a paper low will be returned and the printer will go off-line. The operator will be unable to return it to the on-line state until new forms are placed in the printer.

Auxiliary VFU Channel 9

Format: XXXXXX1X

Channel 9 was detected during the last forms movement.

Bottom-of-Form Detected

Format: XXXXXXX1

The bottom-of-form channel was detected. The channel to be used for bottom-of-form will be determined by the buffer logic of the printer the controller is connected to.

New Status 1 is returned in response to Command 2. Command 2 clears the status of prior status 1, so new status 1 bits returned are conditions at the time Command 2 is executed. Status bits are the same as described in preceding paragraphs on status bytes following a Select. Both Busy and Data Error are defined so that neither can be a logical "1" for New Status 1. Refer to Table C-14.3.

New Status 2 is returned in response to Command 1. Command 1 clears the status of Prior Status 2, so New Status 2 bits returned reflect existing conditions at the time Command 1 is executed. Refer to Table C-14.3.

Table C-14.3. Summary of Model 2445 and 2446 Printer Status Bits

POSITION	NAME STATUS 1	DESCRIPTION
2^7	Interrupt	An interrupt upon completion has been sent
2^6	Selected	
2^5	Busy	
2^4	Illegal Data	Illegal load character, illegal print character
2^3	Intervention Required	Fault VFU Check Off-Line
2^2	Equipment Check	Illegal Command, Command Parity Error, Character Parity Error
2^1	Print Error	Print Check 1, Print Check 2, Illegal Print Code
2^0	Ready	

POSITION	NAME STATUS 2	DESCRIPTION
2^7	Panel Active	Switch pressed on Aux Panel
2^6	Print Check 1	
2^5	Print Check 2	
2^4	Fault	
2^3	VFU Check	Paper break or runaway
2^2	Paper Low	Paper absence detected below print line
2^1	Aux VFU Ch 9	
2^0	Bottom-of-Form	VFU Channel 12

Data Transfer

The output data transfer is initiated with an "Initiate Output on Channel instruction. If the controller is not prepared for data transmission because a print command was not previously executed, the controller will return a Function Acknowledge signal in response to the first Data Request signal received.

If the interval between Data Requests is 15 ± 3 microseconds, any time before a full print line of data is received the controller will assume a short data transfer and initiate a print command. A "Data Request" signal appearing after the print command will cause a "Function Acknowledge" to be returned to the processor.

NOTE:

At least one data byte must be received; the controller will remain busy and await the output instruction indefinitely after execution of a print command.

If more than a full print line of data is received, the controller will return a Function Acknowledge signal to the first Data Request signal received which exceeds the print line size. The full line of data received will be printed, and no special status bit will be set to indicate truncation of the output buffer.

If a data parity error or illegal data byte is detected, the appropriate status bit will be set to a logical "1", busy set to a logical "0", and the line will not be printed.

Detection of a Fault or Master Clear condition will terminate the output operation.

C-15
CONTROLLER
FOR MODEL 2401 AND 2402 COMMUNICATIONS CONTROLLERS

COMMAND CODES

Select Command

OS SSS XXX — S = switch selectable

76 543 210 — Bit position (high-order bit = 7)

Command Sequence

Command Byte 01

Transmit (Send)	201
Receive	202
Terminate	240
Drop Data Terminal Ready	204
Terminate & Disconnect	244

COMMAND SEQUENCES

The following external function command (EF code) sequences are available for conditioning the communications controllers for operation:

SEQUENCE	OP CODE	OUTPUT FIELDS	STATUS BYTES
1	EF Code	Select	1
2	EF Code	Select, dummy	1, 2
3	EF Code	Select, dummy, Cmd 1	1, 2, 3
4	EF Code	Select, dummy, Cmd 1, Cmd 2	1, 2, 3, 4
5	EF Code	Select, dummy, Cmd 1, Syn Code	1, 2, 3, 4

Sequences 1 and 2 allow the user to gain prior status information from the controller without affecting the operation taking place.

Sequence 3 is used to enable a new operation within the controller. (See the description below of Command Byte 1 for the available command codes.)

Sequences 4 and 5 are used to initialize the controller by storing the proper synchronization code, enabling the proper modem interface and selecting the correct line code. These commands should be executed only when the controller is in an idle state, because any changes initiated during an operating mode can disrupt an active data transfer. (See the descriptions below of Command Bytes 1 and 2 for the correct Command Byte 1 sequences.

The descriptions for Command Bytes 1 and 2 define the status bytes which may be returned in response to the command sequences.

Select Byte

The select byte is used to address a single unique controller out of the 16 possible controllers on an I/O channel.

Format: OSSSSXXX

The high-order logical "0" at bit position 7 identifies the byte as a select byte. Bits 6 through 3 "SSSS" are a bit pattern unique for each controller on a channel.

Controller selection takes place when the bit pattern in the select byte[1] matches the state of a switch-selectable decoder. The decoder has the capability to decode any of the 16 possible states of the select byte.

After controller selection is accomplished, the controller will respond to following bytes having a logical "1" in bit position 7) as commands.[2]

If the decode has a negative response to the select byte or detects a parity error, the controller will deselect and ignore all ensuing non-select commands.

Dummy Command

The dummy command is formatted as 1XXXXXXX and has no effect on the controller. Its functions are to facilitate checking of any following command bytes and to gate a status byte to the Processor from the communications controller.

Command Byte 1

Command Byte 1 is described below and summarized in Table C-15.1.

SEND STATUS COMMAND

Format: 10000000

A Send Status command conditions the controller under EF sequence 3 to reply with 3 bytes of accumulated status and one byte of new status without effecting the operation in progress at the time.

If a Send Status command is followed by Command Byte 2, as in EF sequence 4, the controller is enabled for receipt of a new set of parameters, as described under LOAD below.

SYNC LOAD COMMAND

Format: 11000000

A Sync Load command conditions the controller under EF sequence 5 to receive a SYN byte. Upon receipt of the SYN byte, the lower leftmost seven bits are placed in the sync register; the eighth bit loaded is always a "0" (zero) bit.

Prior to initial contact on the communication facility the controller will be preset with a SYN byte via the Sync Load command.

---

[1] Bit 7 is variable for 2404 operation.

[2] In the 2404 Processor, a select code need not have a bit 7 set to logical 0.

DISCONNECT COMMAND

Format: 10000100

A Disconnect command causes the communications line to be disconnected from the user equipment by dropping the Data Terminal Ready (DTR) signal. This command will be preceded by or accompany a Terminate command.

If a Terminate command does not precede the Disconnect command, an Illegal Command status signal will be returned and *no* operation will take place.

If a Terminated command accompanies a Disconnect command (244), the Data Terminal Ready signal is dropped momentarily and then returned to its true state. This mode of operation will be used when *not* in an auto-answer environment.

If the Disconnect command follows a Terminate command, the Data Terminal Ready signal will drop and not return until a Transmit or Receive command is set.

TERMINATE COMMAND

Format: 10100000

A Terminate command conditions the controller for completing the operation in progress.

If the Terminate command is issued while in the Receive mode, the controller will abandon byte synchronization and clear the receive mode immediately. The controller is then able to accept a new command, e.g., Transmit or Disconnect.

If the Terminate command is issued while in the transmit mode, the controller will drop the Request-To-Send signal upon completing the transmission of the last data byte transferred.

A user should issue no command except a Send Status command until the "busy" status condition has been dropped, thereby indicating the completion of the transmit function. When the "busy" status condition drops, the controller is available for a Receive or Disconnect function.

Transmit Command

Format: 100X0001

This command conditions the controller to transmit by raising the modem interface Data Terminal Ready and Request to Send signals. The controller will indicate a "busy" status until the modem has acknowledged the Request to Send signal with a Clear to Send signal. When the Busy signal falls, the controller is available for a data transfer. Data may be transferred a block at a time or a byte at a time.

SYNC FILL MODIFIER

Format: 10010001

If bit X is equal to a logical "1", the controller will add SYN bytes to the data stream whenever the channel has *not* presented the next data byte in time to transmit it in its proper time slot.

RECEIVE COMMAND

Format: 100XX010

This command conditions the controller to receive data by raising the Data Terminal Ready signal, if it is *not* already active and searching for byte synchronization.

A Busy signal, will be returned in Status Byte 1 until synchronization can be achieved.

The ready condition indicates the detection of the Data Set Ready signal from the modem.

SYNC DELETE MODIFIER

Format: 10010010

The sync delete modifier bit conditions the controller to delete all leading sync bytes it detects in the receive data stream. If a receive command is issued with the sync delete bit active, the Busy signal will remain active until after byte synchronization has been achieved and a non-sync byte has been detected.

RESET CHARACTER SYNC MODIFIER

Format: 100X1010

The reset character sync modifier bit should be set to logical "0" (zero) upon initially establishing the receive mode.

When the receive mode is established and the Reset Character Sync command is issued, the controller drops byte synchronization and initiates a new search for byte synchronization. The byte remaining in the parallel register will be cleared upon detection of the command.

AUTO CALL COMMAND

Format: 10001001

The Auto Call command conditions the controller to receive data for dialing a remote location on a switched network.

The user should re-issue the command until a "ready" status is returned, indicating that the auto dialer has acknowledged the request to dial with the Data-Line-Occupied signal.

Upon receiving the "ready" status, the user can present the controller with the data for dialing.

Status bit "Data Set Ready" in status byte 2 will be returned to indicate that the modem has completed the line connection.

If the call cannot be completed, the status and an Abandon Call signal will be returned and the user must issue a Disconnect command and try again.

Command Byte 2

Command Byte 2 is an initialization byte for the control unit and conditions it to the particular code set and line configuration desired.

Command Byte 2 must be presented to the controller prior to enabling the line connection.

The function of each bit is defined in the following paragraphs, starting with the least significant rightmost bit, and the format of Command Byte 2 is summarized in Table C-15.1.

DATA CLOCK SPEED SELECT HIGH/LOW

Format: 1XXXXXXY

An active Y bit causes the controller to select the high data rate of the Data Set currently selected. (If the Data Set is *not* equipped for rate selection, this command is ignored.)

DATA SET SELECT 1 OR 2

Format: 1XXXXXYX

The Y bit enables the controller to select one of two possible Data Set connections. An attempt to select a connection when the Data Set is *not* physically installed will result in an illegal command status. (The Data Set is normally installed in connection # 1.) An active Y bit causes the controller to select connection # 2.

LINE CODE SIZE SELECT

Format: 1XYYYXXX

The Y bits define the byte size to be employed for information exchange. If bit 5 is zero, the controller defaults to an 8-bit line code, as used in binary synchronous transmission of EBCDIC code. Other available values of Y are:

100 = 4-bit line code
101 = 5-bit line code
110 = 6-bit line code
111 = 7-bit line code
000 = 8-bit line code

GENERATE/CHECK ODD/EVEN PARITY

Format: 1YXXXZXX

Bit Y active conditions the controller to generate parity in the transmit mode and to check line parity in the receive mode.

In the receive mode, byte parity is checked and the "line error" status is set if an error is detected. The parity received is passed on to the Processor undisturbed.

In the transmit mode, byte parity is generated for the low-order data bits (code length -1) and passed to the line following transmission of the low-order bits.

Bit Z determines whether the parity generated and checked is to be odd or even. If bit Z is a logical "1," even parity will be utilized.

ILLEGAL COMMANDS

Illegal commands are not executed.

Command Byte 1 will be checked to ensure that an executable command has been requested. An executable command should contain no more than one logical "1" in bit positions 0, 1, and 2.

Whenever Command Byte 1 requests a disconnect (see Command Byte 1 description above) and the controller is in a send or receive mode ready condition, the command is flagged as illegal.

Whenever Command Byte 3 selects a non-existent modem (Data Set), the command is flagged as illegal (see "Data Set Select 1 or 2" paragraph under Command Byte 2 description above).

Table C-15.1. Summary of Command Bytes 1 and 2

FORMAT		
Command Byte 1	Command Byte 2	DESCRIPTION
10000000	1XXXXXXX	Send Status Clear prior status and present new status.
11000000	1XXXXXXX	Sync Load Load Sync register with the byte to be used for character synchronization (byte 4).
10000100	1XXXXXXX	Disconnect Disconnect the communications line and lock out succeeding calls.
10100100	1XXXXXXX	Terminate Disconnect Disconnect one communications line and allow succeeding calls.
10100000	1XXXXXXX	Terminate Come to an orderly completion of the current receive or transmit operation.
100X0001	1XXXXXXX	Transmit Set up conditions to send data.
100XX010	1XXXXXXX	Receive Set up conditions to acquire data.
10001001	1XXXXXXX	Auto Call Set up conditions to initiate a call on a switched network.
10XXXXXX	1XXXXXXY	Select High or Low Data Clock Speed.
10XXXXXX	1XXXXXYX	Select Data Set 1 or 2.
10XXXXXX	1XYYYXXX	Select size of line code.
10XXXXXX	11XXXZXX	Generate/Check Odd or Even Parity. (Z=1 for even parity) (Z=0 for odd parity)

STATUS BYTES

One status byte is returned for each select, dummy, or command byte executed. If a 4-byte function is executed, status will be returned in the following order (if the control unit is not busy): prior status 1, prior status 2, new status 1, new status 2.

Status Byte 1

Status Byte 1 is returned in response to the select byte. These status bits retain the logical "1" status conditions of New Status 1 from the last command executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time. All eight bits may *not* simultaneously be logical "0". This state indicates that the Processor did *not* receive a status response, thus internally generating a response of zero ("Ø").

Bit 2^7 (Interrupt) – This bit is reserved for the interrupt function.

Bit 2^6 (Selected) – This bit will be returned in the status of a successful selection and with any other status presented.

Bit 2^5 (Busy) – This bit will be present whenever the controller's parallel register is *not* available for the operation with the Processor or whenever the terminate (or terminate disconnect) sequence is being performed. Busy status is *not* accumulated as are the other status bits; it reflects the actual busy condition at the time status is taken.

Bit 2^4 (Illegal Command) – This bit indicates that the command received was interpreted as illegal by the controller and not acted upon (see "Illegal Commands" paragraph in Command Byte 2 description above).

Bit 2^3 (Lost Carrier) – This bit indicates that the communications line carrier was lost during byte synchronization in the receive mode.

Bit 2^2 (Channel Parity Error) – This bit indicates a parity error on a command byte or data byte transfer.

Bit 2^1 (Overrun) – This bit indicates that the Processor has failed to "keep up" with the data communications line, unless the Sync Fill option has been selected.

Bit 2^0 (Ready) – This bit indicates that the controller is in the send or receive mode and that a Data Set Ready signal is present from the modem.

STATUS BYTE 2

Status Byte 2 is returned in response to the dummy byte, if executed. These status bits accumulate any "1" bits which occur, momentarily or permanently, since the last command executed.

Bit 2^7 (Data Line Occupied) – This bit indicates that the handset has been uncradled. When this bit goes active, the auto dialer is available to accept data for dialing. If the controller does *not* have auto-call capability, this bit will remain inactive.

Bit 2^6 (Ring Indicator) – This bit is utilized to indicate (in an auto-answer environment) that a ring has been detected. Upon receipt of the ring indicator, the user should condition the controller to present a Data Terminal Ready signal to the modem.

When the controller is being used in an auto-call environment, this bit active indicates that the auto-call command has been unable to complete the connection.

Bit 2^5 (Abandon Call/Retry) – This bit is utilized when operating in auto-call mode to indicate that a call has failed and a recall is required. This bit remains inactive in a controller without auto-call capability.

Bit 2^4 (SYN Fill/Delete) – This bit indicates that a Sync Fill/Delete operation has been performed.

Bit 2^3 (Character Synchronization) – This bit is set in the receive mode to indicate that the controller has achieved byte synchronization with the communications line.

Bit 2^2 (Line Error) – This bit indicates that a parity error was detected during a check parity operation.

Bit 2^1 (Data Set Ready) – This bit indicates the presence of a Data Set connected and selected by the controller.

Bit 2^0 (Clear to Send) – This bit indicates that the Data Set has acknowledged the Request to Send signal from the controller and transmission of data may begin.

STATUS BYTES 3 AND 4

Status Byte 3 and Status Byte 4 are returned in response to Command Byte 1 and Command Byte 2, respectively.

The bits returned in both Status Byte 3 and Status Byte 4 have the same meaning as those returned in Status Byte 1, except that the status returned reflects the conditions present only at the time the command is executed. It does *not* indicate status bits accumulated since a prior command sequence.

Table C-15.2 summarizes the status bytes.

Table C-15.2. Summary of Status Bytes

STATUS BYTES 1, 3, and 4	
BIT POSITION	NAME
2^7	Interrupt
2^6	Selected
2^5	Busy
2^4	Illegal Command
2^3	Lost Carrier
2^2	Channel Parity Error
2^1	Overrun
2^0	Ready

STATUS BYTE 2	
BIT POSITION	NAME
2^7	Data Line Occupied
2^6	Ring Indicator
2^5	Abandon Call/Retry
2^4	Sync Fill/Delete
2^3	Character Synchronization
2^2	Line Error
2^1	Data Set Ready
2^0	Clear to Send

C-16
CONTROLLER
FOR MODEL 2403 COMMUNICATIONS CONTROLLER

COMMAND SEQUENCES

The following external function command (EF code) sequences are available to condition the controller for operation:

SEQUENCE	OP CODE	OUTPUT FIELDS	STATUS BYTES
1	EF	Select	1
2	EF	Select, Dummy Byte	1, 2
3	EF	Select, Dummy Byte, Cmd 1	1, 2, 3
4	EF	Select, Dummy Byte, Cmd 1, 2	1, 2, 3, 4
5	EF	Select, Dummy Byte, Cmd 1, Cmd 3	1, 2, 3, 4

Sequences 1 and 2 allow the user to gain prior status information from the controller without affecting the current operations. Sequence 3 is used to enable a new operation within the controller. Sequences 4 and 5 are utilized to initialize the controller for:

- Enabling the proper modem interface,
- Selecting the transmission speed (bps),
- Selecting internal or external clocking,
- Selecting echoplex mode of transmission,
- Setting the number of stop bits to be transmitted,
- Selecting the correct line intelligence (byte length), and
- Establishing parity generation and checking.

Channel Select Byte 1

DEVICE SELECT

The select byte is used to address the desired device controller (one of a possible sixteen).

Format: OSSSSMMM

The logical "0" at bit position 7 identifies the byte as a device select byte.[1] Bits 6 through 3 are a bit pattern that is unique for each device controller on a channel. Controller selection takes place when the bit pattern in the select byte matches that of a switch-selectable decoder. The decoder has the capability to decode any of the 16 possible states of the select byte. A select code is defined as the presence of an FRQ and a DRQ. Bits 2 through 0 are described under "Select Modifier" below.

After device controller selection is accomplished, the 2403 responds to following bytes that have a logical "1" in bit position 7 as commands for Processor operation. (Bit 7 is variable for 2404 Processor operation.)

If the decoder has a negative response to the select byte, the controller deselects and ignores all ensuing non-select commands. If a parity error is detected on the select byte, the controller will *not* be selected and all ensuing non-select commands are ignored.

SELECT MODIFIER

The three low-order bits of the select byte are treated as modifier bits for the 2403 controller. These bits (if at logical "1") modify the instruction sequences for the controller.

Modifier 1 (0SSSSMM1) – The low-order bit, if a logical 1, will condition the controller to treat a DRQ as if it were an input data request; if logical 0, it will condition the controller as if it were an output data request.

Modifier 2 (0SSSSM1M) – The next-to-low-order bit, if a logical 1, will condition the controller to clear accumulative interrupts on the trailing edge of sequence 3.

Modifier 3 (0SSSS1MM) – The second-to-low-order bit is reserved for future implementation.

Dummy Byte, Byte 2

Format: 1XXXXXXX

The dummy byte has no effect on the controller. Its function is to acquire prior secondary status previous to the issue of a command function.

Command 1, Byte 3

COMMAND FLAG

Format: 1MXXXXXX

Bit 6 of byte 3 functions as a command pointer for byte 4. If bit 6 is a logical "0", byte 4 will be interpreted as command 2. If bit 6 is a logical "1", byte 4 will be interpreted as command 3.

---

[1] In the 2404 Processor, a select code need *not* have bit position 7 set to logical zero (0).

COMMAND 1 MODIFIER

Format: 1XXXMXXX

Bit 3 of byte 3 (if a logical "1") functions as a modifier for the structure of command 1.

Call Request Command

Format: 1XXX1XX1

Bit 0 and bit 3 of byte 3 (if a logical "1") are interpreted as a call request command, conditioning the controller to activate the call request signal to the auto-calling unit (ACU) and to receive data from the Processor for dialing a remote terminal on a switched network. The user should cycle on the command until the "DLO" status is returned, indicating that the auto-dialing unit has acknowledged the request to dial with the data-line-occupied signal posted as status. Upon receiving DLO status, the user can present the controller with the data for dialing. The posted data-set-ready status (MODEM OK) indicates that the modem has completed the line connection, and, if the interrupt for call-request-complete has been enabled, the controller will create a hardware interrupt upon receipt of the data-set-ready signal. If the call cannot be completed, the abandon-call-retry status will be posted and the user must issue a terminate-call-request command and try again.

Break Command

Format: 1XXX1X10

Bit 1 and bit 3 having a logical "1" in command 1 will be interpreted by the controller as a break command, causing the controller to introduce a fixed line state of spaces for a period preset to the modem interface (see Appendix A) after the current data byte has been transferred to the communications line. If another byte is present in the parallel register, the data will be shifted to the communications line after the break command has been performed.

DISCONNECT COMMAND

Format: 1XXXX1XX

Bit 2 of byte 3 (if a logical "1") conditions the 2403 to disconnect the communications line from the user equipment by dropping the data-terminal-ready signal. The command (if issued in transmit or receive mode) will be preceded by or accompany a terminate command. If the terminate command is *not* issued in this case, the controller will post "illegal status" and lock out the controller from accepting this byte and adjacent bytes in the command sequence.

If a terminate command accompanies a disconnect command, the data-terminal-ready signal is dropped for 500 milliseconds and then returned to its true state. This mode of operation will be used when *not* in an auto-answer environment.

If the disconnect command follows a terminate command, the data-terminate-ready signal will drop and *not* return until either the controller is initialized or the transmit/receive function is performed.

TERMINATE COMMAND (BIT 5, BYTE 3)

Terminate Transmit

Format: 1X1XXXX1

A terminate transmit command will cause the 2403 (if in transmit mode) to drop the request-to-send signal upon completing the transmission of the last stop bit to the modem interface.

During the terminate function, the controller will post the busy status. When the Busy status drops, the 2403 is available for any command sequence. Completion of the terminate command will clear the transmit mode.

Terminate Receive

Format: 1X1XXX1X

A terminate receive command will cause the 2403 (if in receive mode) to clear the receive mode immediately.

Terminate Auto Call

Format: 1X1XXXXX

A terminate auto-call command will clear a call request unconditionally, if a call request command has been issued.

RECEIVE COMMAND (BIT 1, BYTE 3)

Format: 1XXXX01X

This command conditions the 2403 to receive data by raising the data-terminal-ready signal (if it is *not* already active) and searching the line data for a start bit. The receive-buffer-available status will *not* be posted until the start bit is detected and a byte is assembled and ready for transfer. When available, the buffer-ready status is posted and will (if enabled) cause a program interrupt.

TRANSMIT COMMAND (BIT 0, BYTE 3)

Format: 1XXX00X1

This command conditions the controller to transmit by raising the modem interface signals data-terminal-ready and request-to-send. Until the modem has acknowledged the request-to-send signal with a clear-to-send signal, the transmit buffer ready status will *not* be posted. When this has been posted, the controller can now enter data transfer mode. Data may be transferred a byte at a time or by block bursts.

ENABLE INTERRUPTS

Format: 1XX1XXXX

Bit 4 of command 1 (if a logical "1") enables interrupts for the following interrupt conditions:

- Call Request Complete
- Stop Bad
- Transmit Buffer Ready
- Receive Buffer Register Ready
- Break Detection
- Ring Indicator
- Open Line

Command Two, Byte 4

Command byte 2 is used to select internal baud rate, line intelligence, and the number of stop bits to be inserted after a byte has been transmitted.

INTERNAL CLOCK RATE (BYTE 4, BITS 3, 4, 5, and 6)

Format: 1SSSSXXX

The four high-order bits (6, 5, 4, and 3) of byte 4 will be decoded as one of ten selectable baud rates; they are coded as follows:

CLOCK SELECT	BIT DECODE 6543	BAUD RATE (bps)
1	0000	1,200
2	0001	600
3	0010	300
4	0011	200
5	0100	150
6	0101	110
7	0110	100
8	0111	75
9	1000	50
10	1001	2,400

LINE CHARACTER LENGTH (BYTE 4, BIT 2 AND 1)

Format: 1XXXXSSX

Bits 2 and 1 of byte 4 will be decoded as one of four selectable line intelligences. Line intelligence is herein defined as: "the meaningful data that resides within stop and start bits of a transmitted or received byte." The decoding of bits 2 and 1 are as follows:

LINE INTELLIGENCE	BIT DECODE	
LENGTH	BIT 2	1
8	0	0
7	0	1
6	1	0
5	1	1

ONE/TWO STOP BIT INSERTION (BYTE 4, BIT 0)

Format: 1XXXXXX1

Bit 0 of byte 4 (if a logical "1") will cause the 2403 to insert two stop bits after every transmitted byte. If set to a logical "0", the 2403 will insert one stop bit after every transmitted byte.

Command 3, Byte 4

This command byte is utilized for modem selection, line parity, internal clock, originate terminal, self test, and echoplex modes of operation.

GENERATE/CHECK PARITY (BYTE 4, BIT 6 AND 5)

Odd/Even Parity

Format: 1YƵXXXXX

Bit Y active (i.e., logical "1") will cause the 2403 to generate parity in the transmit mode and to check parity in the receive mode. In the receive mode, character parity is checked for validity and, if in error, the line error status is posted. In the transmit mode, character parity is generated for the low-order data bits (line intelligence 1) and passed to the modem interface following transmission of the low-order bit.

Bit Ƶ determines whether parity is to be odd or even. If bit Ƶ is a logical "1", even parity will be used.

ECHOPLEX MODE (BYTE 4, BIT 4)

Format: 1XX1XXXX

If bit 4 is a logical "1", the 2403 will be placed in echoplex mode. Echoplex mode will clock the received data back onto the transmit data line.

ALTERNATE MODEM SELECT (BYTE 4, BIT 2)

Format: 1XXXXYXX

The Y bit enables the 2403 to select one of two possible Data Set connections. An attempt to select a connection when the Data Set is not physically installed will post the illegal command status bit. (The Data Set is normally installed in connection # 1.) The Y bit (when a logical "1" causes the 2403 to select connection # 2.

INTERNAL CLOCK (BYTE 4, BIT 3)

Format: 1XXX1XXX

Bit 3 (when a logical "1") will cause the 2403 to select its own internal clock. When it is a logical "0", the modem clock (external) is selected. This command is used in conjunction with command 2, byte 4, bits 4, 5, and 6.

ORIGINATE TERMINAL

Format: 1XXXXX1X

Bit 1 of byte 4 (when a logical "1") will cause the controller to present an originate-terminal signal to the modem interface.

In originate mode, the controller is the transmitting terminal. If bit 1 is a logical "0", the controller will be the "called" terminal, i.e., it will be in the receive mode.

SELF TEST

Format: 1XXXXXX1

Bit Ø of byte 4(when a logical "1") will cause the controller to be placed in a self-test mode of operation, whereby the transmit data, request-to-send, carrier signal, etc., are looped back within the controller, enabling the controller to functionally bypass the modem.

ILLEGAL COMMANDS

Illegal commands are *not* executed by the 2403 controller, however; the command will be acknowledged by the controller by setting the illegal status bit active. The following list of illegal commands is sensed by the 2403:

Command One

- A call request, when in transmit mode.
- A disconnect without a terminate in either transmit or receive mode.
- A call request with ACU power off.

Command Two

- No valid condition; storage only.

Command Three

- A modem selection that is *not* available.

SUMMARY OF COMMANDS

Table C-16.1 summarizes the sequences of the 2403 controller.

Table C-16.1. 2403 Controller Command Sequences

COMMAND SEQUENCE				FUNCTION
SELECT	COMMAND 1	COMMAND 2	COMMAND 3	
0SSSSMMM	10000000			Send Status: Clear prior status and present new status.
0SSSSM1M	1XXXXXXX			Clear Interrupts: Clears all accumulative interrupts.
0XXXXMMM	11000000[1]		1ABCDEFG	Load Command 3: A. Generate Parity B. Odd/Even C. Echoplex mode D. Internal clock E. Alternate modem F. Originate G. Self test
0SSSSMMM	1000000[1]	1AAAABBC		Load Command 2: A. Bps rate B. Line intelligence C. Stop bit 1 or 2
0SSSMMM	10010000			Enable Interrupt: Interrupts: Open line Ring indicator Call request complete Break RCVE buff ready XMIT buff ready Stop bad
0SSSSMM1	10000000			Input Data Request: Transfer data from 2403 to Processor
0SSSSMM0	10000000			Output Data Request: Transfer data from Processor to 2403.
0SSSS000	1X00X0X1			Transmit: Set up conditions to send data.
0SSSS000	1X00X01X			Receive: Set up conditions to receive data.
0SSSS000	1X1000X1			Terminate Transmit: Come to an orderly completion of the current transmit operation.
0SSSS000	1X10001X			Terminate Receive: Come to an orderly completion of the current receive operation.

Table C-16.1 (Cont'd.). 2403 Controller Command Sequences

COMMAND SEQUENCE				FUNCTION
SELECT	COMMAND 1	COMMAND 2	COMMAND 3	
0SSSS000	1X1000XX			Terminate Call Request: Terminate call request if active.
0SSSS000	1X000100			Disconnect: Disconnect one communication line and lockout succeeding calls.
0SSSS000	1X1001XX			Disconnect Terminate: Disconnect one communication line and allow succeeding calls.
0SSSS000	1X001001			Call Request: Set up conditions to initiate a call on switched network.
0SSSS000	1X001010			Break: Insert a transmit "break" function for a preset period of time (see Appendix A).

[1]This is a part of a controller "initialization" routine that need be performed only in the beginning of a program sequence.

SUMMARY OF STATUS BYTES

STATUS BYTE 1

BIT POSITION	NAME
2^7	Interrupt
2^6	Selected
2^5	Performing Operation
2^4	Transmit Buffer Ready
2^3	Receive Buffer Ready
2^2	Accumulated Error Flag
2^1	Break Detected or Ring Indicator
2^0	Modem Ready (Controller Ready)

STATUS BYTE 2

BIT POSITION	NAME
2^7	Data Line Occupied
2^6	Lost Carrier or Abandon Call Retry
2^5	Line Error
2^4	Stop Bit Bad
2^3	Invalid or Parity Error
2^2	Overrun
2^1	Open Line
2^0	Reserved

Primary Status Rank One

Primary status is returned in response to the select byte. These status bits retain the logical "1" status conditions of new status 1 from the last command executed and accumulate any "1" bits which may occur (momentarily or permanently) since that time. All eight bits may *not* simultaneously be logical "0". This state indicates that the Processor did *not* receive a status response and internally generated a response of zero.

Bit 2^7 (Interrupt) – This bit will be logical "1" whenever an interrupt function has been generated by the 2403 controller.

Bit 2^6 (Selected) – This bit will be returned in the status of a successful selection and with any other status presented.

Bit 2^5 (Performing Operation) – This bit indicates that the controller is busy performing an operation and *cannot* accept a command sequence.

Bit 2^4 (Transmit Buffer Ready) – This bit will be returned whenever the transmit buffer is ready for a byte transfer tc the Processor.

Bit 2^3 (Receive Buffer Ready) – This bit will be returned whenever the receive buffer has a byte for transfer to the Processor.

Bit 2^2 (Accumulated Error Flag) – This bit will be the accumulative sum of any error, as posted in the secondary rank two status.

Bit 2^1 (Break Detected or Ring Indicator) – This bit represents the "or" state of several mutually exclusive status bits that occur only at a given time during a program sequence. *Break Detected:* This bit will be active whenever the controller has detected a "break". *Ring Indicator:* This bit is utilized in an auto-answer environment that a ring has been detected. Upon receipt of the ring status, the user should condition the controller to present a data-terminal-ready signal to the modem.

Bit 2^0 (Modem Ready) – This bit indicates that the modem is available for transmit or receive mode data transfers (data-set-ready signal).

Secondary Status Rank Two

Secondary status is returned in response to the DUMMY byte, if executed. These status bits accumulate any "1" bits which occur, momentarily or permanently, since the last command issued.

Bit 2^7 (Dial Line Occupied) – Active, this bit indicates that the handset has been taken "Off Hook." When this bit goes active, the auto dialer is available to accept data for dialing. If the controller does *not* have Auto Call capability, this bit will remain inactive.

Bit 2^6 (Lost Carrier or Abandon Call Retry) – This bit is accumulated in primary status bit 2^2 and represents the "OR" of several status bits that are mutually exclusive and occur only at a given time during a program sequence. *Lost Carrier:* This bit will be active whenever the controller is in the receive mode and has lost the carrier during the assembly of a data byte. This indication would cause a FAKN to an input data transfer . *Abandon Call/Retry:* When the controller is being used in an auto-call environment, this bit active indicates that the auto-call command has been unable to complete the connection.

Bit 2^5 (Line Error) – This bit is accumulated in primary status bit 2^2 and indicates that a parity error was detected during a check parity operation or that the receive data was invalid due to line distortion.

Bit 2^4 (Stop Bit Bad) – This bit is accumulated in primary status bit 2^2 and indicates that the data byte assembled in the receive mode did *not* have a stop bit. A bad stop will cause the controller to FAKN and input data transfer.

Bit 2^3 (Invalid or Parity Error) – This bit is accumulated in primary status bit 2^2 and indicates the "or" of two status conditions. *Invalid:* This bit indicates that a command sequence was detected as invalid and the controller could *not* perform the operation. *Channel Parity Error:* This bit indicates a parity error on a command byte or an output data byte transferred from the Processor. A parity error on an output data transfer will cause the controller to FAKN the transfer.

Bit 2^2 (Overrun) – This bit is accumulated in primary status bit 2^2 and indicates that the processor has failed to "keep up" with the receive data communications line.

Bit 2^1 (Open Line) – This bit is accumulated in primary status bit 2^2 and indicates that a period of three seconds of a fixed-line state of spaces has been detected on the receive data communications line by the controller. An open line will cause the controller to FAKN an input data transfer.

Bit 2^0 – This bit is reserved.

Cumulative Status

Cumulative status is that which is stored in the controller until a command 1 sequence has been initiated.

Current Status

Status is updated (cleared) on a command 1 sequence. Any status returned to this sequence will then be current status.

Dynamic Status

Dynamic status is a status indication that need *not* be updated to current status by a command 1 sequence. These status indications are:

- Transmit Buffer Ready
- Receive Buffer Ready

Status Bytes

Byte 1 — Primary accumulative and dynamic status is returned.

Byte 2 — Secondary accumulative status is returned.

Byte 3 — Current primary status is returned.

Byte 4 — Primary accumulative and dynamic status is returned.

C–17
CONTROLLER FOR
MODEL 2441 MATRIX PRINTER

CODE

The printer operates according to the 1967 USA Standard Code of Information Interchange (ASCII).

Codes not recognized by the printer will be ignored. Optionally, a *diamond* illegal character will be printed in place of the non-recognizable code.

FUNCTION CONTROL CODES

The following function control codes are recognized by the printer and cause it to act as described below. These codes can be received by the printer via the input interface.

SPACE (SP)

This code moves the print head one character space to the right without printing.

BACKSPACE (BS)

This code moves the print head one character space to the left without printing.

Carriage Return (CR)

This code causes the print head to be moved to the left until the left margin is reached.

Line Feed (LF)

This code advances the paper one vertical position without moving the print head.

New Line Option (NL)

When this option is installed, the printer will perform a carriage return operation and also a line feed operation upon receiving a line feed code.

Horizontal Tabulation Option (HT)

This code causes the print head to move continuously to the right (without printing) until a tab set position or right margin is reached. When this option is not installed, this code is treated as an illegal character.

Form Feed Option (FF)

This code causes the printer to skip to channel 1 when the VFU is installed. When the option is not installed, this code is treated as an illegal character.

Vertical Tabulation Option (VT)

This code causes the printer to skip to channel 2 when the VFU option is installed. When this option is not installed, this code is treated as an illegal character.

Escape Option (ESC)

This code is always used in conjuntion with another graphic code to perform other functions not specified by the ASCII control characters.

This code followed by the graphic characters in Table C-17.1 will perform the functions described in the table, if the VFU option is installed. When this option is not installed, this code is treated as an illegal character.

Table C-17.1. Escape Option Supplementary Codes

Graphic Code	Function
A	Skip to Channel 1
B	Skip to Channel 2
C	Skip to Channels 1 and 2
D	Skip to Channel 3
E	Skip to Channels 1 and 3
F	Skip to Channels 2 and 3
G	Skip to Channels 1, 2 and 3
H	Skip to Channel 4
I	Skip to Channels 1 and 4
J	Skip to Channels 2 and 4
K	Skip to Channels 1, 2 and 4
L	Skip to Channels 3 and 4

Additional Function Control Codes

The printer will be designed with flexibility to provide for the detection and recognition of other control codes for future expansion or customer requirements.

C–18
CONTROLLER FOR MODEL 2493 TELETYPEWRITER

The 2493 Teletypewriter interfaces to the 2400 processor via one of the SYSTEM 2400 Communication Controllers (2401, 2402, or 2403). Programming parameters of the controllers are applicable to the 2493. Hence, for in-depth coverage on the teletypewriter programmable characteristics, refer to the appropriate subsection herein that documents the communication controller implemented.

Table C-18.1 gives a summary of 2493 command and status codes.

Table C-18.2 lists 2493 Teletype codes.

Table C-18.1. Model 2493 Teletypewriter Commands and Status Codes

2493 TELETYPE COMMANDS

COMMAND	BYTE 1	BYTE 2	BYTE 3	BYTE 4
NO OP	SELECT	200	200	200
TRANSMIT	SELECT	200	201	304
RECEIVE	SELECT	200	202	304
TERMINATE	SELECT	200	240	304

2401/2402 COMMUNICATIONS CONTROLLERS AND 2493 TELETYPE STATUS

BYTES 1, 3, 4		BYTE 2	
2^7	INTERRUPT	2^7	DATA LINE OCCUPIED
2^6	SELECTED	2^6	RING INDICATOR
2^5	BUSY	2^5	ABANDON CALL/RETRY
2^4	ILLEGAL COMMAND	2^4	SYNC FILL/DELETE
2^3	LOST CARRIER (Reverse Channel)	2^3	IN SYNC
2^2	CHANNEL PARITY	2^2	LINE ERROR
2^1	OVERRUN	2^1	DATA SET READY
2^0	READY	2^0	CLEAR TO SEND

Table C-18.1 (Cont'd.). Model 2493 Teletypewriter Commands and Status Codes

2403 COMMUNICATION CONTROLLER AND 2493 TELETYPE STATUS

STATUS BYTE 1	
Bit	Name
7	Interrupt
6	Selected
5	Performing operation (Busy)
4	Transmit Buffer Ready
3	Receive Buffer Ready
2	Accumulated Error Flag
1	Break Detected or Ring Indicator

STATUS BYTE 2	
Bit	Name
7	Data Line Occupied
6	Lost Carrier or Abandon Call Retry
5	Line Error
4	Stop Bit Bad
3	Invalid or Parity Error
2	Overrun
1	Open Line
0	Reserved

Table C-18.2. 2493 Teletype Codes (USASCII Code)

Bits b_4 b_3 b_2 b_1	b_7 0 b_6 0 b_5 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
0 0 0 0	NUL	DLE	SP	0	\	P	@	p
0 0 0 1	SOH	DC1	!	1	A	Q	a	q
0 0 1 0	STX	DC2	"	2	B	R	b	r
0 0 1 1	ETX	DC3	#	3	C	S	c	s
0 1 0 0	EOT	DC4	$	4	D	T	d	t
0 1 0 1	ENQ	NAK	%	5	E	U	e	u
0 1 1 0	ACK	SYN	&	6	F	V	f	v
0 1 1 1	BEL	ETB	'	7	G	W	g	w
1 0 0 0	BS	CAN	(	8	H	X	h	x
1 0 0 1	HT	EM	)	9	I	Y	i	y
1 0 1 0	LF	SS	*	:	J	Z	j	z
1 0 1 1	VT	ESC	+	;	K	[	k	{
1 1 0 0	FF	FS	,	<	L	~	l	|
1 1 0 1	CR	GS	-	=	M	]	m	}
1 1 1 0	SO	RS	.	>	N	^	n	/
1 1 1 1	SI	US	/	?	O	_	o	DEL

C–19
CONTROLLER FOR
MODEL 2428 MULTIPLEXER

The 2428 Multiplexer is interfaced to the 2400 Processor through the standard Input/Output selector channel. The multiplexer provides a means of:

- Data transfer from 1 to 32 terminals,
- Sending variable length data transmission to any of the 32 terminals, and
- Polling the terminals until some action is required on a terminal. When action is required, the Multiplexer will send a Service Request to the Processor.

INTERFACE BETWEEN MULTIPLEXER AND TERMINAL

The interface is designed to function over 1000 feet of total cable length. Provisions are made to allow additional buses, each with its own driver/receiver in the Multiplexer so up to four 1000 foot buses can be run in parallel.

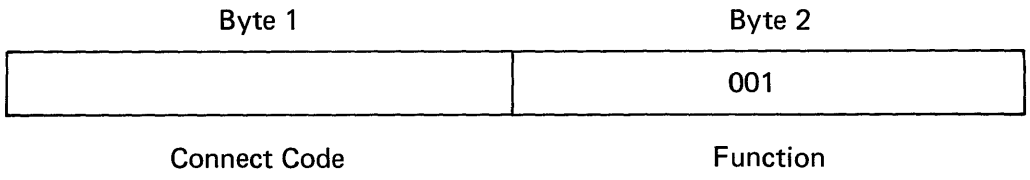
OUTPUT TRANSFER

The output transfer is initiated with an External Function instruction as follows:

EF Item 1, Item 2, Item 3

Item 1 contains the channel number

Item 2 is a 2-byte item with the following meaning:



The connect code is a variable and will be strapped in the Multiplexer.
The function 001 will initiate an output.

Item 3 is a 1-byte item with the following meaning:

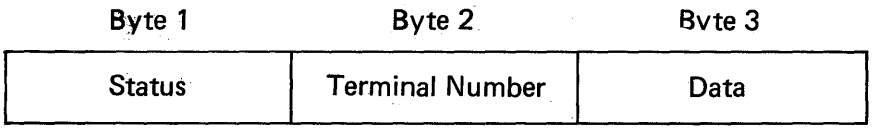
Bit 0 = Multiplexer is ready
Bit 1 = Parity Error on last output.

After the output has been initiated, the Multiplexer is ready for the data transfer. An Initiate output instruction will start the transfer from the Multiplexer to the terminal, and this transfer will continue as long as the channel continues to bring up Data Requests.

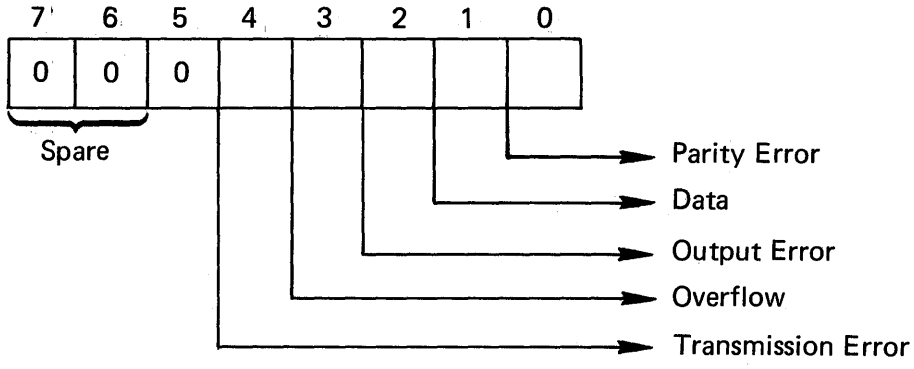
INPUT TRANSFER

If there is no output transfer, the Multiplexer is continually polling the terminals. When the Multiplexer finds a terminal that needs service, it stops the polling and raises the service request. In the interrupt routine of the processor, an Initiate Input Instruction is executed and three bytes are inputted to the Processor. When the Multiplexer has sent the last byte, it drops the Service Request and continues polling the terminals.

The Initiate Input Instruction inputs three bytes with the following meanings:



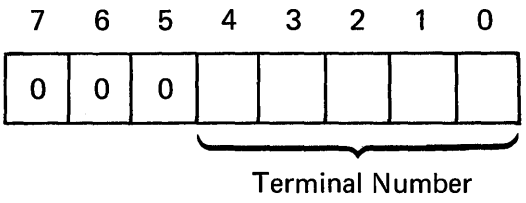
Byte 1 Bit Assignments



Byte 1 shows the state of the terminal that need servicing. The meaning of these bits are:

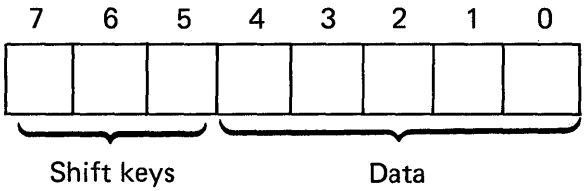
Bit Position	Meaning
0	Parity Error (PE) - If this bit is a "1", the parity check was in error from the keyboard to the multiplexer.
1	Data (D) - If this bit is a "1", there is a data byte in byte 3.
2	Output Error (OE) - If this bit is a "1", there was a transmission error on the last output transfer to the terminal.
3.	Overflow (O) - If this bit is a "1", the buffer in the keyboard is full of data and another key was depressed and lost.
4	Transmission Error (TE) - If this bit is a "1", there was a transmission error on the poll from the terminal. This condition is not established until three unsuccessful attempts have been made to input the data to the multiplexer.

Byte 2 Bit Assignments



Bits 0 - 4 from the terminal address of this input. Bits 5, 6, and 7 are spares. Terminal numbers are from 00_8 to 37_8 .

Byte 3 Data



Bits 0 - 4 contain the data. Bits 5, 6, and 7 contain the encoded shift information.

NOTE:

Transmission error, output error, and/or data error will cause a service request.

READER'S COMMENT FORM

SYSTEM 2400 Processor Programming in Machine Code, Form No. M-2269-1172

Please restrict remarks to the publication itself. Comments pertaining to hardware/software difficulties and improvement requests should be submitted in the established manner. Please give specific page and line references with your comments when appropriate.

ERRORS NOTED:

SUGGESTIONS FOR IMPROVEMENT:

HOW DO YOU USE THIS DOCUMENT?

☐ AS AN OPERATOR'S REFERENCE MANUAL ☐ AS AN INTRODUCTION TO THE SUBJECT
☐ AS AN INSTRUCTOR IN A CLASS ☐ AS A STUDENT ☐ FOR ADVANCED KNOWLEDGE OF SUBJECT

YOUR NAME _____ DATE _____

TITLE _____

COMPANY _____

ADDRESS _____
STREET CITY STATE ZIP CODE

Form No. M-1459-70 Printed in USA

Fold

Fold

Postage Will Be Paid by Addressee

No Postage Stamp Necessary If Mailed in the United States

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT No. 73 HERKIMER, NEW YORK

MOHAWK DATA SCIENCES CORP.

PALISADE STREET

HERKIMER, NEW YORK 13350

ATTN: TECHNICAL AND EDUCATIONAL DOCUMENTATION

Cut Along Line

Fold

Fold

THE THROUGHPUT SPECIALISTS

MOHAWK DATA
SCIENCES CORP.

PALISADE STREET • HERKIMER, NEW YORK 13350