PRELIMINARY EDITION

Burroughs

B 1700 SYSTEMS

REFERENCE MANUAL

Burroughs Corporation
Detroit, Michigan 48232

$4.00

Printed in U.S. America

11-73

1057155

Burroughs believes that the information described in this manual is accurate and reliable, and much care has been taken in its preparation. However, no responsibility, financial or otherwise, is accepted for any consequences arising out of the use of this material. The information contained herein is subject to change. Revisions may be issued to advise of such changes and/or additions.

This reprint also includes the information released under the following:

PCN 1057155-001 (9/8/72)
PCN 1057155-002 (9/28/73)

Correspondence regarding this document should be addressed directly to Systems Documentation, Technical Information Organization, TIO-Central, Burroughs Corporation, Burroughs Place, Detroit, Michigan 48232.

Table of Contents

Table of Contents (cont)

Table of Contents (cont)

Table of Contents (cont)

Table of Contents (cont)

Table of Contents (cont)

Table of Contents (cdnt)

List of Illustrations

List of Illustrations (cont)

List of Illustrations (cont)

List of Illustrations (cont)

List of Tables

INTRODUCTION

The B 1700 series of computers are small-to-medium-scale systems which utilize the latest data processing concepts, such as defined fields, delayed binding, and interpretive structure. The technique of micro-programming is used to accomplish most data manipulation. This concept brings to Burroughs Corporation users a new level of computer responsiveness for business and scientific problems, as well as in data communications. This responsiveness is available to even the smallest organization that has a requirement for electronic data processing. Speed and flexibility are the key words which describe this system. To support this, Burroughs Corporation offers:

a. Bit addressability. Main memory is addressable to the individual bit. It has no preferred word or byte boundaries which are visible to the rest of the system.

b. Memory modularity is provided, from a minimum of 16K bytes to 40K bytes of main memory on the B 1712 and 64K bytes on the B 1714. The B 1726 systems contain two types of memory: main memory and control memory. Control memory is up to four times faster than main memory in either 2K bytes standard or 4K bytes optionally. Main memory size is provided from 24K bytes minimum to 98K bytes maximum.

c. Higher-level programming languages available are: COBOL, FORTRAN, BASIC, RPG, and a Master Control Program (MCP).

d. Flexibility of the system lies in its ability to be micro-programed. Micro-instructions operate on a set of hardware registers available to the micro-programmer. Sets of micro-instructions called interpreters are supplied by Burroughs Corporation.

SECTION 1

SYSTEM DESCRIPTION

GENERAL.

This section contains an overall description of the B 1700 series of data processing systems without explaining the specifics of the hardware. The systems are designed for modularity utilizing monolithic circuitry. The basic B 1710 System consists of a processing unit, main memory, peripheral I/O controllers, console control panel, power supplies, and cabinets. The basic B 1720 System consists of a processing unit, port interchange, main memory, control memory, peripheral I/O controllers, maintenance console, power supplies, and cabinets.

SYSTEM CONFIGURATION.

The equipment is modular and can utilize any peripheral units within the scope of the I/O controllers described in this manual. System configurations and comparisons are given in figures 1-1 and 1-2, and in tables 1-1 and 1-2.

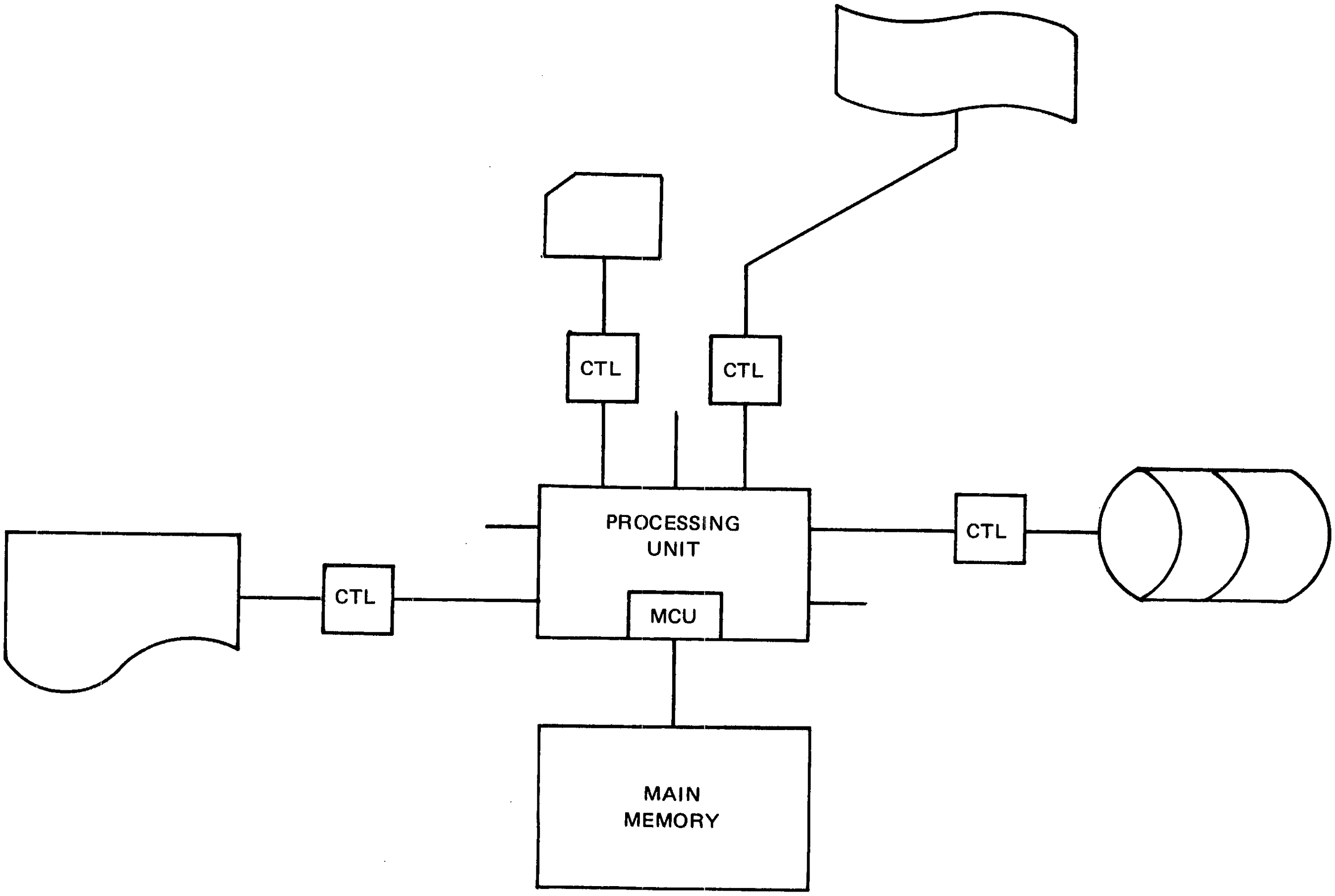


Figure 1-1. B 1712/B 1714 Systems Configuration

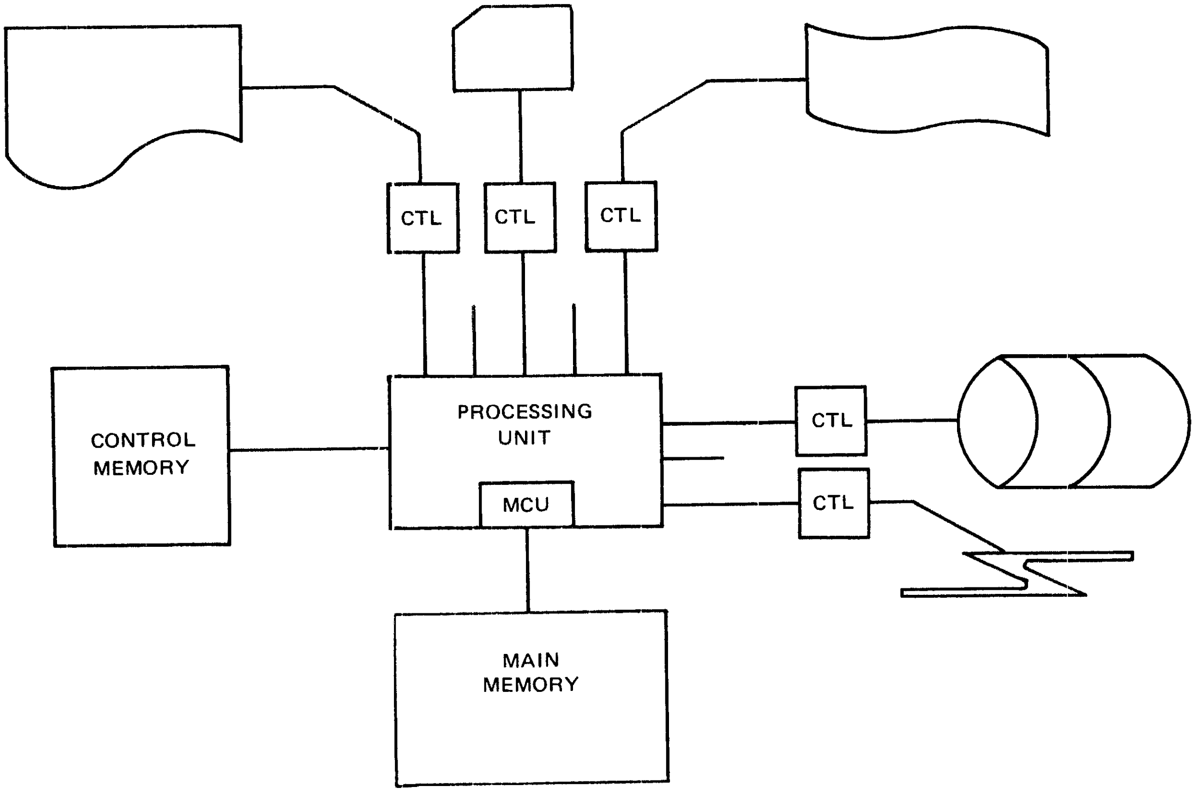


Figure 1-2. B 1726 System Configuration

Table 1-1

System Comparison

Characteristic	B 1712/B 1714	B 1726
Processors	1	1
Clock rate	2 MHz/4 MHz	6 MHz
Main memory	Yes	Yes
Minimum size	16,384 bytes	24,576 bytes
Maximum size	40,960/65,536 bytes	98,304 bytes
Control Memory	No	Yes
Size	Not applicable	2048 bytes standard an additional 2048 bytes optional
Port interchange	No	Yes
Maximum number of ports	Not applicable	2

Table 1-1 (cont)

System Comparison

Characteristic	B 1712/B 1714	B 1726
I/O controllers Minimum number Maximum number	Yes 1 8	Yes 1 8

Table 1-2

System Configuration Chart

Unit Number	Description	Capabilities	Remarks	B 1712	B 1714	B 1726
1351	Single Line Control				x	x
	Std. Asynch.				x	x
	Std. Synch.				x	x
9111	Card Reader	80 col. 800 CPM				x
9112	Card Reader	80 col. 1400 CPM				x
9115	Card Reader	80 col. 300 CPM		x	x	x
9116	Card Reader	80 col. 600 CPM			x	x
9119-1	Card Reader	96 col. 300 CPM		x	x	x
9131-1	Reader Sorter	1000 DPM	13 pocket			x
9134-1	Reader Sorter	1625 DPM	4/8/12/16/pocket			x
9135-2	Reader Sorter	900 DPM	8 pocket		x	x
9135-3	Reader Sorter	900 DPM	12 pocket		x	x
9136-5	Reader Sorter	600 DPM	8 pocket		x	x
9136-6	Reader Sorter	600 DPM	12 pocket		x	x
9210	Card Punch	80 col. 100 CPM		x	x	x
9240-1	Line Printer	475 LPM	132 prt. pos.		x	x
9240-2	Line Printer	700 LPM	132 prt. pos.		x	x
9240-3	Line Printer	1040 LPM	132 prt. pos.			x
9245-16	Line Printer	300 LPM	132 prt. pos.	x	x	
9245-19	Line Printer	400 LPM	132 prt. pos.		x	
9247-3	Line Printer	750 LPM	132 prt. pos.		x	x
9249-1	Line Printer	90 LPM	132 prt. pos.	x	x	
9249-2	Line Printer	180 LPM	132 prt. pos.	x	x	
9319-2	Card Reader	96 col.		x	x	x
	Punch	300/60 CPM				
9319-4	High Speed	96 col.			x	x
	Reader Punch	500/120 CPM				
9340	Console Printer	10 CPS		x	x	x
9371-7	H-P-T DF Memory Bank	20ms avg access time; 231KB avg trsf rate.	7MB storage cap			x
9371-14	H-P-T DF Memory Bank	40ms avg access	14MB storage cap			x

Table 1-2 (cont)

System Configuration Chart

Unit Number	Description	Capabilities	Remarks	B 1712	B 1714	B 1726
9319-2	Card Reader Punch	96 col. 300/60 CPM		x	x	x
9319-4	High Speed Reader Punch	96 col. 500/120 CPM			x	x
9340	Console Printer	10 CPS		x	x	x
9371-7	H-P-T DF Memory Bank	20ms avg access time; 231KB avg trsf rate.	7MB storage cap.			x
9371-14	H-P-T DF Memory Bank	40ms avg access time; 238KB avg trsf rate.	14MB storage cap.			x
9374-10	Add-on increment (for 9371-14)	40ms avg access time; 238KB avg trsf rate.	14MB storage cap.			x
9374-14	Add-on increment (for 9371-7)	20ms avg access time; 231KB avg trsf rate.	7MB storage cap.			x
9381-12	Magnetic Tape Unit	9 ch. 800 BPI NRZ	18KB - 2 st. cluster		x	x
9381-13	Magnetic Tape Unit	9 ch. 800 BPI NRZ	18KB - 3 st. cluster		x	x
9381-14	Magnetic Tape Unit	9 ch. 800 BPI NRZ	18KB - 4 st. cluster		x	x
9381-22	Magnetic Tape Unit	9 ch. 800 BPI NRZ	36KB - 2 st. Cluster		x	x
9381-23	Magnetic Tape Unit	9 ch. 800 BPI NRZ	36KB - 3 st. cluster		x	x
9381-24	Magnetic Tape Unit	9 ch. 800 BPI NRZ	36KB - 4 st. cluster		x	x
9390-3	Magnetic Tape Unit	7-channel tape	18/50 KC			x

Table 1-2 (cont)

System Configuration Chart

Unit Number	Description	Capabilities	Remarks	B 1712	B 1714	B 1726
9419-2	Card Reader Punch/ Data Recorder	96 col. 300/60 CPM		x	x	x
9419-6	Multi-purpose Card Unit	96 col. 300/300/60 CPM	6 stackers	x	x	x
9480-1	Disk Storage	2.3 megabyte	Single	x	x	x
9480-2	Disk Storage	4.6 megabyte	Dual	x	x	x
9481-1	Disk Storage	4.6 megabyte	Single	x	x	x
9481-2	Disk Storage	9.2 megabyte	Dual	x	x	x
9491-2	Magnetic Tape Unit	10KB Drive 9 channel		x	x	x
9486-2	Disk Storage	95.5 megabyte	Dual			x

SYSTEM CONCEPT.

This series of computers can be described as systems that execute their programs under control of "micro" instructions. It is impractical for the programmers to write micro-programs; therefore, various higher level languages have been developed for customer use on the system. It is not the intent of this section to describe these higher-level languages but to describe the relationship that exists between the micro-program and the higher-level languages.

S (secondary) language instructions are intermediate instructions which are equivalent to the machine language of a conventional system. For each S-instruction there exists a string of micro-instructions which interpretively execute the functions specified by that S-instruction.

It must be remembered that the S-language instruction does not directly cause the hardware to perform a function.

Because S-instructions are software defined by the sequence of micro-instructions, they may be as complex as the language designer requires. In most cases, S-instructions contain data addresses or base relative addresses, length of data fields, units of data, and an operation to be performed upon the data.

S-instructions may completely specify a compiler-level language. Burroughs Corporation has defined the S-instructions and has written the necessary interpreters and compiler programs for several compiler languages: COBOL, FORTRAN, RPG (Report Program Generator), and BASIC. These compiler programs generate the necessary S-language instructions to perform the various operations specified by the higher level language.

INTERPRETERS.

In addition to the S-language program (compiled user program), another program which is referred to as an interpreter is utilized. An interpreter has been developed for each of the high-level S-languages: COBOL, FORTRAN, RPG, and BASIC. It is the function of the interpreter to fetch the S-language instruction(s) from main memory, and interpret or execute them. The S-instructions are decoded and a series of micro-instructions are executed to cause the hardware to perform the function specified by the S-language instruction. On completion of the execution of a series of micro-instructions (representing an S-instruction), the interpreter fetches the next S-language instruction and the operation continues in this manner. The series of micro-instructions to perform each S-language instruction may be stored in control memory, main memory, or be called from disk.

The S-language programs and required interpreters are located and loaded from disk under control of the Master Control Program (MCP). On completion of the load, the MCP then passes control to the S-language program. The MCP is written in either SDL or micro-code. The SDL version (MCP II) requires the SDL interpreter or a portion of it to always be present in the system. For COBOL programs a COBOL interpreter is called and is in memory with the SDL interpreter, when required. The memory resident portion of MCP I is micro-coded; therefore, only the interpreter required for the user program need be present.

INTERRUPT SYSTEM CONCEPT.
The interrupt system is described as a "soft" interrupt system. That is, any interrupt that occurs does not cause any specific hardware function and is only recognized by the software. On recognition of the interrupt by the software, there are no specific rules for handling the interrupt and it is left solely to the discretion of the programmer. The handling of these interrupts will be discussed in the MCP Reference Manual (not yet available).

DEFINED FIELD CONCEPT.
An integral part of the micro-programming technique is the defined field concept (bit addressability). This assumes that storage is described in units of bits. It is a generalized scheme for handling field lengths ranging from one bit to a large number of bits through the use of a bit address and bit length, regardless of how fields may cross hardware "byte" boundaries. Special hardware has been implemented to achieve bit addressability and variable length data fields within main memory. The hardware contains logic to adjust the address and length values in the addressing registers. Micro-instructions in turn are capable of setting the processor control circuits to operate on different units of data and iterating the micro-instruction sequence until the field in main memory is exhausted. Data units may vary from one bit to 15 bits. Field length may vary from one to 65,535 bits.

Example: Assume a Write command to main memory from bit address 2014 (hexadecimal) with a field length of 12 bits. the data to be written is three decimal (4-bit) digits with a value of 345. A Write forward results in the "3" beginning at bit address 2014 (see figure 1-3).

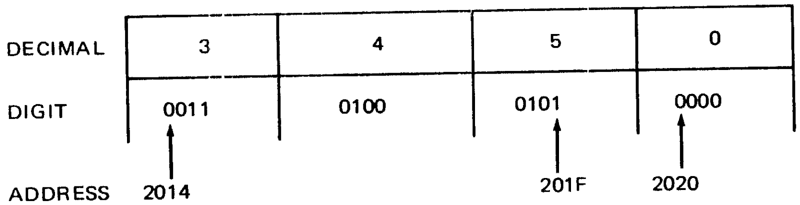


Figure 1-3. Write Forward

However, a Write reverse results in a "3" beginning at address 2008 (see figure 1-4).

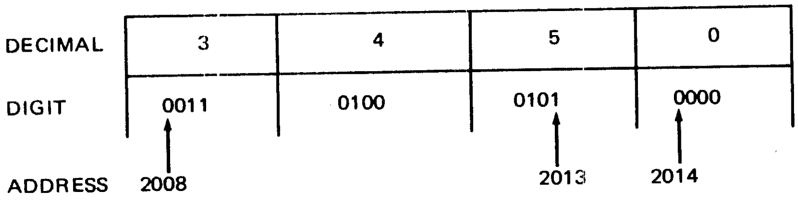


Figure 1-4. Write Reverse

Also note that a Write forward from bit address 2014 is equivalent to a Write reverse from bit address 2020.

DELAYED BINDING.
The noncommitment of resources until the actual need for these resources arises may be referred to as delayed binding. The proper utilization of resources enables the user to get the maximum system throughput per dollar investment.

Main memory is completely managed by the B 1700 Master Control Program. User code that is called into memory by the MCP may or may not be in one contiguous area. This allows for utilization of even small areas of memory.

Data space is within the area specified by the base and limit registers and is divided into two portions: overlayable and nonoverlayable. Program code is not in the area specified by the base and limit registers.

Compiler-generated code is completely independent of absolute memory locations and consequently is not restricted to particular areas of memory. Re-entrant code may exist; that is, one COBOL Compiler may be resident in memory for two or three compiles that are being accomplished, with each compile having a unique data area assigned to it.

Virtually infinite memory, proper control of input/output resources, bit addressability, and a flexible system architecture are other features which provide for proper utilization of resources on the B 1700 family of computers.

FUNCTIONS OF MAJOR UNITS.

PROCESSING UNIT.
The processing unit provides the control and combinatorial portion of the system along with various addressing and data manipulation registers.

I/O INTERFACE.
The I/O interface is connected to the processor and handles all communications between it and the I/O controllers on the B 1700 Systems. The I/O interface is capable of handling up to eight I/O controllers.

PORT INTERCHANGE.
The port interchange, if applicable, is the access path to main memory.

MEMORY CONTROL UNIT.
Main memory is connected to the memory control unit, which handles all memory access requests. The memory control unit resolves the addressing conflict between the bit-oriented data accesses and the byte-oriented parity checking. On the B 1710 Systems, the memory control unit is connected directly to the processor; on the B 1726, the memory control unit is connected to the port interchange which is connected to the processing unit.

CONTROL MEMORY.
Control memory, if present, is used to store the sequences of micro-instructions which perform macro-operations (COBOL and FORTRAN). Control memory is four times faster than main memory and is available in two sizes: 1024 or 2048 16-bit words. If sufficient control memory is not available, portions of main memory are utilized for storage of micro-instructions.

I/O CONTROLLER(S).
Each I/O controller controls the operation of the peripheral(s) connected to it. Each controller is a special-purpose device to interface with one type of peripheral. In most cases an I/O controller interfaces with a single device; however, in some cases a single I/O controller interfaces with two or more devices on a timed-shared basis (magnetic tape unit, disk files).

CENTRAL SYSTEM CABINET.
Connectable cabinets are used to contain all central system modules: memory, power supplies, and circulation fans.

CONSOLE CONTROL PANEL.
The console control panel is an integral part of the processor, consisting of a display panel plus various register- and data-manipulation switches. It is located on the front of the cabinet containing the processor. (See figure 1-5.)

CONSOLE DISPLAY LIGHTS.

REGISTER DISPLAY INDICATORS. These consist of a row of 24 lights which monitor the status of the 24-bit main bus of the processor. In the HALT state the register (or function) designated by the register select switches is displayed.

POWER INDICATOR. This switch/indicator is used to turn power on or off and to light the appropriate ON/OFF indicator.

RUN INDICATOR. This light indicates the processor is in the execute or fetch state, as opposed to the HALT or Read/Write main memory state.

STATE INDICATOR. This light, which is software-controlled, indicates that either control (MCP) or normal state operations are presently being executed. It is lit when the system is in control state.

BOT INDICATOR. This light indicates that the tape cassette is at the Beginning Of Tape position. The light is under hardware control.

PARITY INDICATOR. This light indicates that an irrecoverable parity error has occurred while reading a tape cassette. The light is under hardware control.

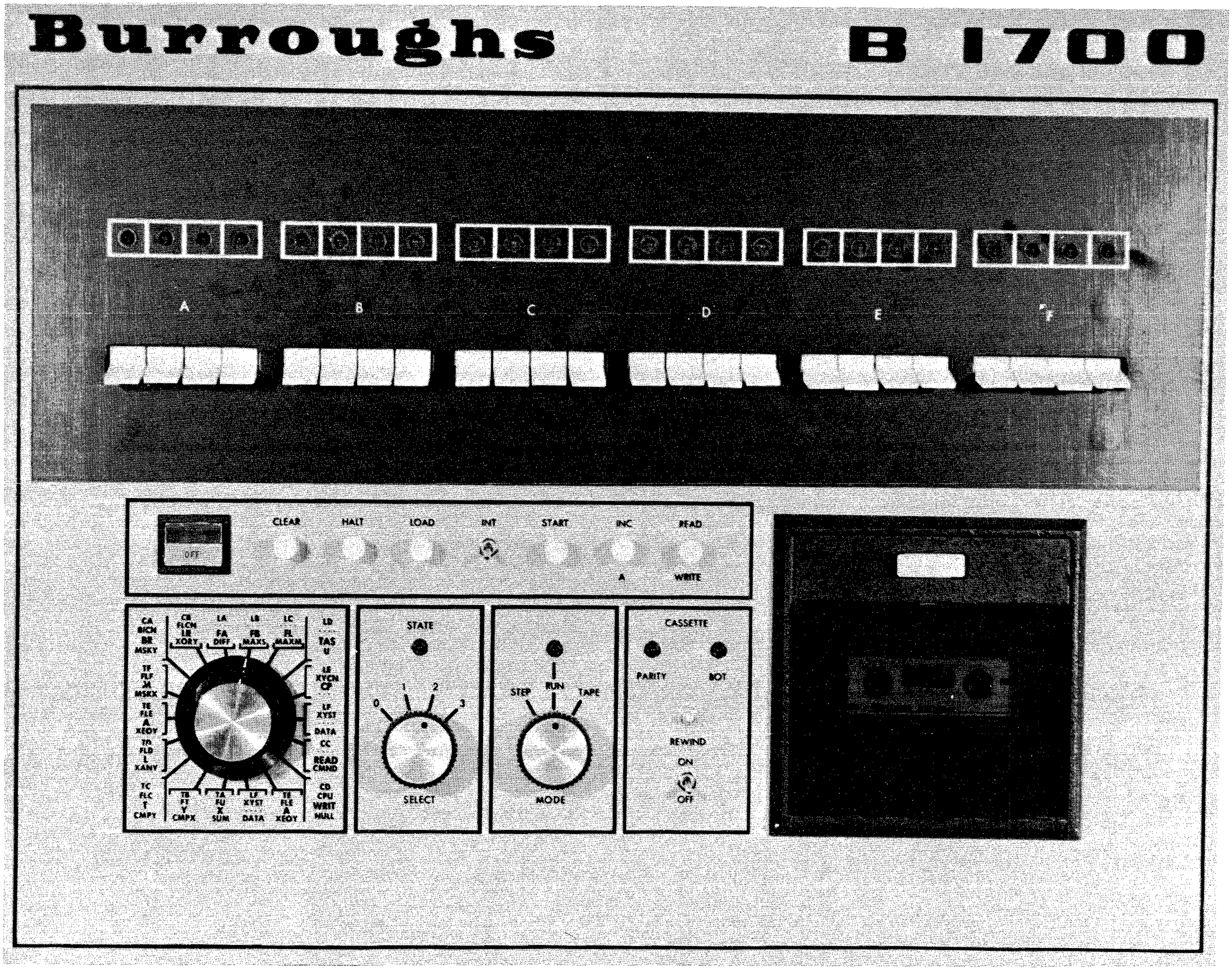


Figure 1-5 Console Control Panel

CONSOLE CONTROLS.

CLEAR SWITCH. This is a pushbutton switch that clears the registers in the processor. On the B 1712 and B 1714, the switch clears the MAR(A), M, C, and U registers. On the B 1726, the switch clears all interrupts in the port interchange, memory control unit, and I/O controllers.

HALT SWITCH. This is a pushbutton switch which causes the processor to halt execution at the end of the present micro-instruction. The next micro-instruction to be executed is fetched and stored in the micro-instruction register (M register). Which register the register select switch is pointing to is displayed on the 24 register display lights.

LOAD SWITCH. This is a pushbutton switch that causes the data displayed on the register display lights to be entered in the register pointed to by the register select switches.

DATA ENTRY SWITCHES. These consist of a row of 24 toggle switches which are operative only when the processor is in the HALT state. The desired bit configuration may be entered into the register selected by the register select switches by using the appropriate toggle switches. The data entered is displayed on the register display lights. (Refer to READ/WRITE switch.)

INT (INTERRUPT) SWITCH. This is a toggle switch that causes a console interrupt condition to be set in the processor control register (C register). This interrupt is testable by the MCP, and the MCP decides what the appropriate system reaction is to be.

START SWITCH. This is a pushbutton switch that initializes various functions within the system. It is used in conjunction with the console MODE switch, the CLEAR pushbutton, and the READ/WRITE pushbutton.

INC-A SWITCH. This is a pushbutton switch which increments the control memory address register (A register) by one micro-instruction word address when displaying or writing control memory from the console. The register select switch must be set to MSM, and the console MODE switch must be set to TAPE.

READ/WRITE SWITCH. This is a pushbutton switch that enables manual writing or reading of data into memory. The processor must be in the halt state and the desired data address must be loaded into the A register. To read data the register select switches must be set up for (pointing to READ) a read operation. Depressing of the READ/WRITE switch will cause the data at the address in the A register to be displayed in the register display lights. For a write operation the desired destination address must be loaded in the A register. The Register Column and Register Row Select Switches must point to the WRIT position and the desired data to be written must be entered into the system via the 24 Data Entry Switches. Depression of the Read/Write Switch completes the Write operation

REGISTER COLUMN AND REGISTER ROW SELECT SWITCHES. The register COLUMN SELECT switch is a 4-position rotary switch which selects one of four columns of registers (or functions). (Refer to table 1-3.)

Table 1-3

Register Select Switch
Column-Row Matrix

Row	Column 0	Column 1	Column 2	Column 3
0	TA	FU	X	SUM
1	TB	FT	Y	CMPX
2	TC	FLC	T	CMPY
3	TD	FLD	L	XANY
4	TE	FLE	A (MAR)	XEOY
5	TF	FLF	M	MSKX
6	CA	BICN	BR	MSKY
7	CB	FLCN	LR	XORY
8	LA	TOPM(*1)	FA	DIFF
9	LB	Reserved	FB	MAXS
A	LC	Reserved	FL	MAXM
B	LD	Reserved	TAS	U
C	LE	XYCN	CP	MBR(*1)
D	LF	XYST	MSM(*1)	DATA
E	CC	INCN(*1)	Console READ	CMND
F	CD	CPU	Console WRIT	NULL

The register ROW SELECT switch is an 18-position rotary switch which selects one of 16 rows of registers (or functions). Two rows are repeated for convenience; they are rows 4 and 13 (D) in table 1-3. These two repeated rows are the most utilized.

The various registers are addressed within the micro-instructions or from the front console by a series of co-ordinates. The first co-ordinate is to select one of 16 rows of registers and the second co-ordinate then selects one of four columns. The particular register (or function) selected by the intersection of the column and row is displayed when the processor is in the HALT state.

CONTROL MODE SWITCH. The control MODE switch is a rotary switch with the following positions: STEP, RUN, and TAPE.

*1 Not available on B 1710 Systems.

When the switch is in the STEP position, pressing the START switch causes the micro-instructions in the M register to be executed and the next micro-instruction to be fetched.

When in the RUN position, pressing the START switch causes the continuous fetch and execution of micro-instructions from the address specified in the micro-instruction address register.

When in the TAPE position, pressing the START switch causes the tape cassette reader to start reading the mounted TAPE cassette. The processor then enters a fetch-execution phase wherein micro-instructions are accepted and executed directly from the tape cassette. The tape cassette may contain load routines, confidence checks, or maintenance test routines (MTR). Initial micro-instructions on the cassette tape can cause subsequent micro-instructions to be loaded into memory from any I/O device.

CASSETTE REWIND. This is a pushbutton switch which causes the tape cassette to rewind to the BOT (beginning of tape) mark.

CASSETTE ON/OFF. This is a toggle switch which controls power to the tape cassette unit.

CONSOLE PRINTER/KEYBOARD.
To operate with the Burroughs Master Control Program, a console printer/keyboard is required for communications between the Master Control Program and the system operator.

MAIN MEMORY.
Main memory is addressable to the individual bit and has variable operation lengths from 0 to 24 bits in parallel. The length, direction, and address of operation is under micro-program control, either up or down within the address continuum. Main memory has no preferred word or byte boundaries which are visible to the rest of the system.

Main memory is a logical part of the processor since most of its control logic is part of that unit. The processor is the only unit to which main memory interfaces. The actual storage medium is, however, physically separate from the processor and is modular in design to allow for expandable capacity.

ADDRESSING.
Main memory is addressed by a 24-bit absolute address, a 1-bit field direction indicator, and a 5-bit field length value. The address is modified by the field direction bit. The field length value specifies the number of data bits in the direction indicated to be accessed, and varies from zero (no operation) to 24. For field length values greater than 24 bits, address values may be incremented and micro-instruction sequences are re-executed until the entire field has been accessed.

PARITY CHECKING.
Reliability of data within main memory is maintained with a hardware parity checking technique. Main memory is physically byte oriented

with a parity bit for every eight data bits. The main memory control logic resolves addressing, parity checking, data rotation, and compaction as described in the following paragraphs.

The high-order 21 bits of an address point to a byte within main memory where the operation is to begin. This "key byte" and three more bytes above or below, depending on the field direction bit, are read into the read memory information register (R-MIR) in the memory control unit. Parity is always checked on all 36 bits within this register.

Next, the low-order three bits of the address along with the field length value and the field direction bit (both contained in the micro-instruction) determine what is passed to or from main memory.

A parity error on any operation sets a bit in the processor C register (CD bit 3 set to 1), and appropriate action is taken by the MCP.

MEMORY DATA TRANSFER

Data is transferred, right-adjusted, 24 bits (three bytes) is parallel to and from main memory. Any operation calling for less than 24 bits will have leading (left) zeros supplied by the memory control unit. The field length controls the main memory operation and leading zeros are ignored or generated as needed within the memory control unit.

READ OPERATION.

A 24-bit address, a 1-bit field direction sign, and a 5-bit field length value are received with the Read request by the memory control unit from the processor. The Read is non-destructive of the data in memory and parity is checked during the operation. The data is then stripped of parity, right-adjusted, and masked with leading zeros as it is moved into the processor.

FIELD LENGTH.

The 5-bit field length value indicates the length of the data, right-adjusted, to be read from or written into memory. The normal field length value range is from zero to 24 bits.

The value of zero leaves the data in main memory unchanged. A value of 25 or 26 implies a field length of 24 and forces correct parity or incorrect parity on data written during the Write or Swap operation. Field lengths greater than 24 are reserved.

Information formerly on this page
was deleted by PCN 1057155-001.

SECTION 2

B 1710 PROCESSORS

GENERAL.

This section discusses the B 1710 processing unit(s), which consist of the control portion and the combinatorial portion of the system along with the various addressing and data manipulation registers.

PROCESSING UNIT.

The processor is the general-purpose data manipulator of the system. It performs the basic micro-functions necessary for a variety of commercial, scientific, and data communications applications. These micro-functions are performed by a set of elementary operators called micro-instructions. Micro-instructions operate on strings of bits within registers that were formerly the responsibility of the hardware logic.

By stringing a series of micro-instructions together, a macro-operation comparable to that defined by an assembler or compiler-level instruction may be implemented. This micro-instruction string is an interpreter for a macro "S-instruction" operation. Micro-instruction strings are executed in a "read-only" mode and are stored in main memory.

PROCESSOR REGISTERS.

MICRO-INSTRUCTION (M) REGISTER.

The M register (micro-instruction register) is a 16-bit register used to hold and decode the current micro-instruction which has been obtained from main memory or the TAPE cassette. The M register is addressable as a 16-bit source or destination register. The right-most 16-bits, if more than 16 are transferred, are loaded into the M register. Data is always transferred right-adjusted and zero-filled to the left. Each micro-instruction to be loaded from memory is ORed into the M register. Thus, moves to the M register allow modification of the next micro-instruction without altering the original micro-instruction in main memory. The state of this register is decoded by the processor to enable various control levels to perform the operation called for by the current micro-instructions.

MICRO-INSTRUCTION ADDRESS REGISTER (MAR).

The MAR (micro-instruction address register) is a 19-bit register capable of addressing micro-instructions located in main memory. It addresses 16-bit micro-operators in main memory which are assumed to be Located at bit addresses exactly divisible by 16. When used in this manner the low-order four bits of MAR are ignored. Thus, only the left-most 15 bits of MAR are actually utilized in addressing.

The MAR is capable of having binary increments from 0 through 4095 added to or subtracted from it, with a high-speed carry adder to facilitate micro-program branching. The MAR is automatically incremented by one word (16 bits) as each micro-instruction is fetched. Memory wraparound can occur and is permitted.

The MAR is addressable as a source or destination register. When addressed as a source, the low-order four bits are set to zero. When used as a destination, the low-order four bits of the source are stored in the low-order four bits of the MAR. They are, however, not significant. Direct addressing of 40,960 bytes on the B 1712 and 65,536 bytes on the B 1714 is possible through the MAR.

The MAR also serves to address main memory for the Read/Write memory micro-instruction. The address of the next micro-instruction is temporarily stored in a scratchpad holding register. The main memory address to be accessed is obtained from the FA register.

ADDRESS (A) STACK.

The A stack is a 16-element deep, 24-bit wide memory which operates as a push down stack, i.e., a last-in-first-out (LIFO) structure. Address wraparound occurs when more than 16 entries to the stack are made. Data is not destroyed on a removal from the stack. The micro-instructions Call and Move to TAS (top of A stack) result in a "put into" (push) stack request, whereas Exit and Move from TAS result in a "take out of" (pop) stack request.

Using this stack, the micro-routines operate in a normal call-return mode. This allows for highly shared micro-routines. The A stack is not intended to be used exclusively as an address stack; it has been made 24 bits wide to allow for operand storage. (See figure 2-1.)

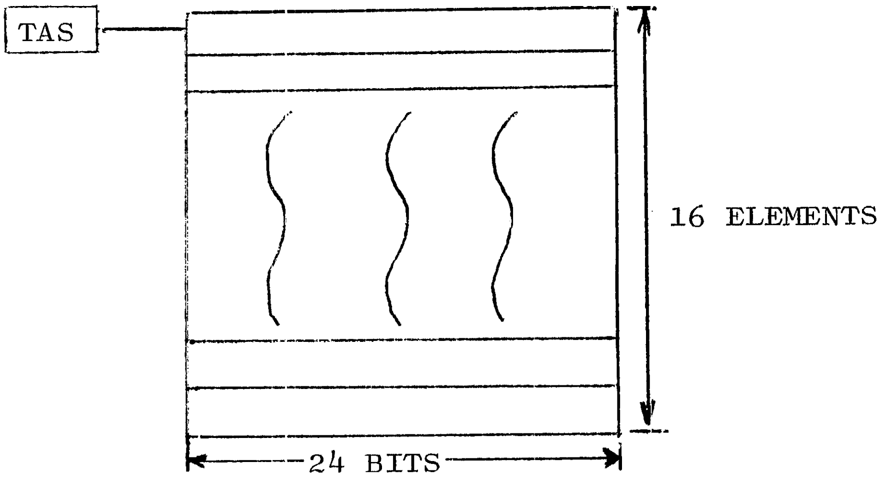


Figure 2-1. A Stack

TOP OF THE ADDRESS STACK (TAS) REGISTER.

The TAS register (top of the A stack) is the 24-bit register which is currently on the top of the A stack. Moves to or from TAS result in data in the A stack being automatically moved to or from the A stack on a last-in-first-out basis.

GENERAL PURPOSE REGISTERS (L, T, X, Y).

L REGISTER. The L register (figure 2-2) is a 24-bit general-purpose register. It is addressable as a source or as a destination register, either 24 bits in parallel or in 4-bit groups. Since the L register is addressable in 4-bit groups, its contents is available for analysis and alteration via the 4-bit function box. Manipulate, Skip, and Bit-Test-Branch micro-instructions may operate on the data in the L register. The L register is one of four registers (X, Y, L, T) capable of Read/Write operations with main memory.

LA	LB	LC	LD	LE	LF
3 0	3 0	3 0	3 0	3 0	3 0
23 ←					→ 0

Figure 2-2. L Register

T REGISTER. The T register (figure 2-3) is a 24-bit general-purpose register. It is addressable as a source or as a destination register, either 24 bits in parallel or in 4-bit groups. Since the T register is addressable in 4-bit groups, its contents is available for analysis and alteration via the 4-bit function box. Manipulate, Skip, and Bit-Test-Branch instructions can operate on the data in the T register, being one of the four registers (X, Y, T, L) capable of Read/Write operations with main memory. The T register is also capable of Shift/Rotate and Extract operations.

TA	TB	TC	TD	TE	TF
3 0	3 0	3 0	3 0	3 0	3 0
23 ←					→ 0

Figure 2-3. T Register

X AND Y REGISTERS. The X register and the Y register are 24-bit general purpose registers. They are used primarily to hold and act as sources for two of the three operands of the combinatorial section (the 24-bit function box) of the processor. The other operand is the CYF register (carry flip-flop). The X and Y registers are both addressable as source and destination registers.

Both registers, along with the L register and the T register, are capable of Read/Write operations with main memory. Both registers are capable of the Shift/Rotate operation. The X register is affected by the Normalize operation.

FIELD (F) DEFINITION REGISTER.
The F register (field definition) specifies the address and lengths of data fields in main memory (see figure 2-4). It is a 48-bit register which is functionally divided into two portions: a 24-bit FA (field address) register and a 24-bit FB register which is divided into a 4-bit FU (field unit) register, a 4-bit FT (field type) register, and a 16-bit FL (field length) register.

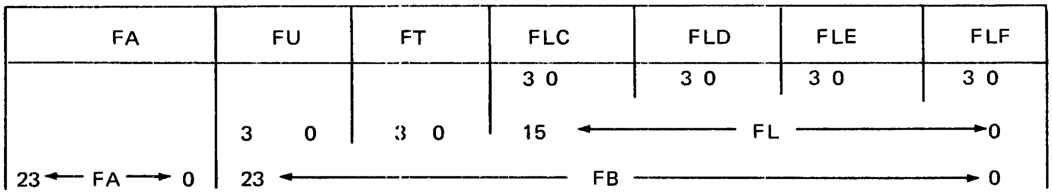


Figure 2-4. F Register

FA PORTION OF THE F REGISTER. The FA register holds the absolute main memory address of a data field and has the capability of addressing 524,488 bits (64KB) without regard to physical boundaries. FA may be addressed as either a source or destination register.

FB PORTION OF THE F REGISTER. FB may be considered as one 24-bit register or as various subregisters. The FU register holds descriptive information regarding the units of data which make up the addressed field in main memory (binary or 4-bit groups). The FT register holds additional descriptive information. The FU and FT values do not affect the main memory operation. The FU and FT values are usually supplied by S-instructions. The FU may also be used with the BIAS micro-instruction to set the CPU and CPL registers. The FL register holds the total length in binary of the field being operated on in main memory, and is capable of describing fields up to 65,536 bits in length.

FB may be addressed as either a source or destination register or addressed in part as FU (4 bits), FT (4 bits), and FL (16 bits). FL may also be addressed in 4-bit subfields (FLC, FLD, FLE, or FLF).

SCRATCHPAD.
A scratchpad of 16 words, each 48 bits wide, is provided to hold field descriptors of operands (see figure 2-5). Scratchpad may be addressed as 16 48-bit words (S00-S15) or 32 24-bit words (S00A-S15A, S00B-S15B). Any word may be a source or a destination and may be used to hold pointers, values, or any other information.

The second half of the first 48-bit word (S00B), along with FU and FL, is always entered as input to the 4-bit function box and used in certain decision-making logic. The field length value in the FL register is continually being compared against a corresponding portion (SFL) of the S00B word to determine the relationship of their contents: high, low, equal, or zero. This relationship may then be used in the Bit-Test-Branch micro-instruction (refer to FLCN register). The FU and FL portions of the F register and the like portions (SFU and SFL) of S00B

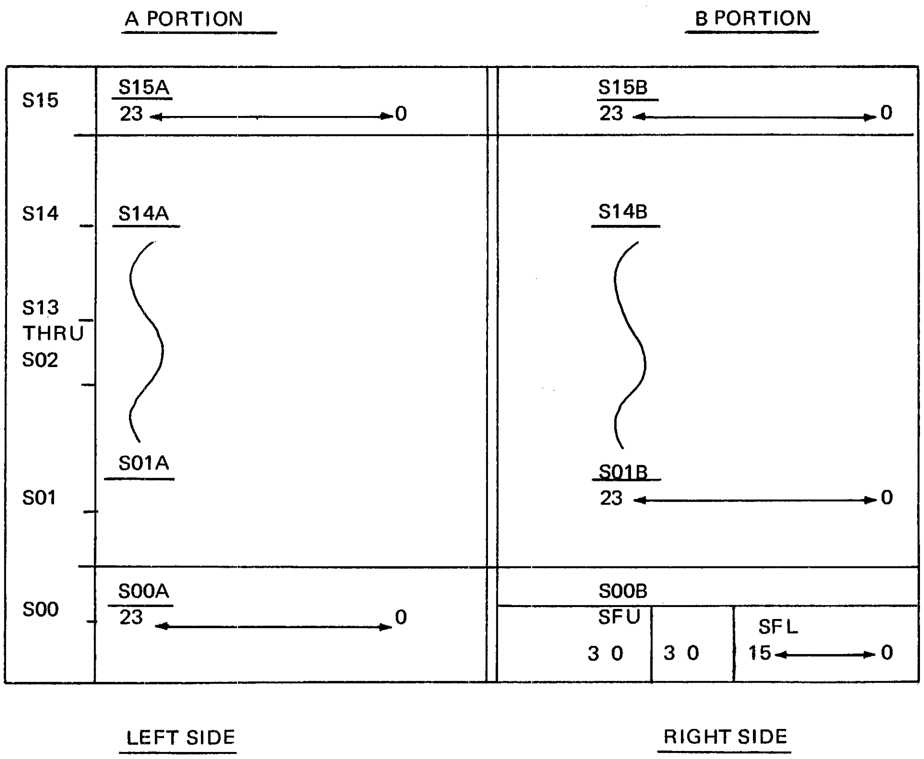


Figure .2-5. Scratchpad

may also be used in the combinatorial section to set the value in the CP portion of the C register.

FIELD LENGTH CONDITION (FLCN) REGISTER.
The FLCN register (figure .2-6) is a 4-bit register which contains the result of a comparison between the FL portion of the F register and the corresponding portion of the first scratchpad word (right-most 16 bits of S00B) which is called SFL. It has the following interpretation:

FL≠0 . - FL not equal to 0
FL<SFL - FL less than SFL
FL>SFL - FL greater than SFL
FL=SFL - FL equal to SFL

FL=SFL	FL>SFL	FL<SFL	FL≠0
3	2	1	0

Figure .2-6. · FLCN Register

The relevant bits of FLCN are SET by comparing FL and SFL as positive 16-bit quantities. A one bit in a position of the FLCN register indicates that the respective condition is true.

BASE AND LIMIT REGISTERS (BR, LR).

The base register (BR) and limit register (LR) may be used as source or destination registers. They are used for memory protection and for base relative addressing. Memory protection is provided for in the MCP by checking the main memory address in all memory Write operations with the BR and LR allowing the operation to take place only within these limits.

CONTROL (C) REGISTER.

The C register (control) is a collection of independent registers which are utilized by the interrupt system of the processor and the combinatorial section (see figure 2-7). The C register is 24 bits wide and is divided into three 8-bit functional parts.

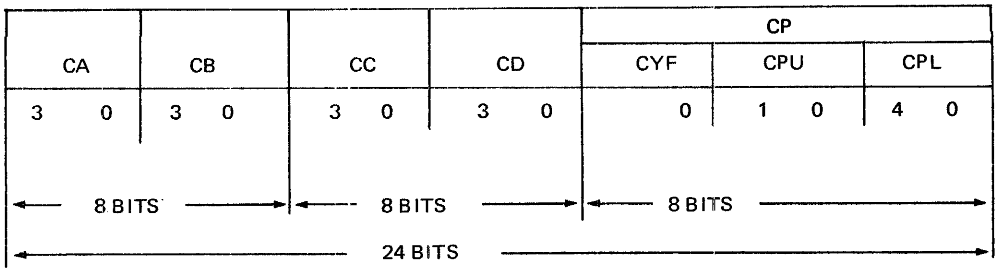


Figure 2-7. C Register

CA AND CB REGISTERS. This portion of the C register is addressable in 4-bit units as CA and CB. These portions are available as general-purpose source and destination registers.

CC AND CD REGISTERS. This portion of the C register is addressable in two 4-bit units defined as CC and CD. CC and CD are used for storage of the processor states and for processor interrupts as shown in the following paragraphs.

<u>CC Register.</u>

The bit meanings for the CC register are as follows:

a. Bit 0 - console interrupt.
b. Bit 1 - I/O service request.
c. Bit 2 - timer interrupt.
d. Bit 3 - console state.

The console interrupt bit is set from the console INT (interrupt) switch and allows the software to come to an orderly stop. The I/O service request interrupt bit is set by an I/o controller when it issues a service request to the processor. The micro-program must determine what caused the interrupt and transfers or receives data from the device. The timer interrupt bit is set by the hardware timer every 100 milliseconds. Once the timer interrupt bit is set, it stays set until the MCP resets it.

CD Register.

The bit meanings for the CD register are as follows:

a. Bit 0 - reserved.
b. Bit 1 - reserved.
c. Bit 2 - reserved.
d. Bit 3 - memory parity error interrupt.

Any parity error detected during a main memory operation sets the memory parity error interrupt bit (CD bit 3).

All interrupts in the processor are soft and no reaction occurs as a result of an interrupt bit being set until the micro-program tests for such settings.

CYF, CPU, AND CPL REGISTERS. The least-significant bits of the C register are composed of the arithmetic unit carry flip-flop (CYF), the 2-bit arithmetic unit type (CPU), and the 5-bit combinatorial data length control (CPL). Moves from CP to a 24-bit register automatically inserts 16 leading zeros. Moves to CP from any register use only the least-significant eight bits of the field being moved.

The Carry micro-instruction is supplied for manipulating CYF, thereby remembering the status of a carry or borrow beyond the length of an operand. For SUM and DIFF results, CYF is always an input to the combinatorial section along with the X and Y registers. CPU controls the arithmetic unit type of the combinatorial section and is addressable only as a destination register. The possible settings and their meanings are listed below:

CPU	Data Type
00	Binary
01	4-bit binary
10	Undefined
11	Undefined

CPL controls the length of the operands being entered as input to the combinatorial section and should be an integral multiple of the data type specified by CPU for valid arithmetic results. The maximum allowable value of CPL is 24, which is an integral multiple of both data types. Larger values are reserved.

MAXIMUM SIZE OF MAIN MEMORY (MAXS) REGISTER.

The MAXS register is a 24-bit pseudo register which is set by a field engineer to indicate the maximum size of the installed main memory. The MAXS register is addressable as a source register only.

MAXIMUM SIZE OF CONTROL MEMORY (MAXM) REGISTER.
The MAXM register is a 24-bit pseudo register which is set by a field engineer to indicate the maximum size of the installed control (micro-instruction) memory. For this processor MAXM always contains a value of zero. The MAXM register is addressable as a source register only.

TAPE CASSETTE INPUT (U) REGISTER.
The U register is a 16-bit register used primarily to accumulate input (usually micro-instructions) from the console TAPE cassette. The U register is addressable only as a 16-bit source.

Only the micro-instruction Register Move may access the U register. In TAPE mode the contents of this register can be moved by a micro-program on the tape to the M register for execution.

DATA REGISTER.
The DATA register is a 24-bit pseudo register which can act as a source or destination. It is used to transfer data between the I/O devices and the processor. When used as a source it accepts 24 bits of data from the I/O controller. When used as a destination register, data is placed into the DATA register and is sent to the I/O controller.

COMMAND (CMND) REGISTER.
The CMND register (command) is a 24-bit pseudo register which can act as a destination only and is used to transfer I/O commands to the I/O devices.

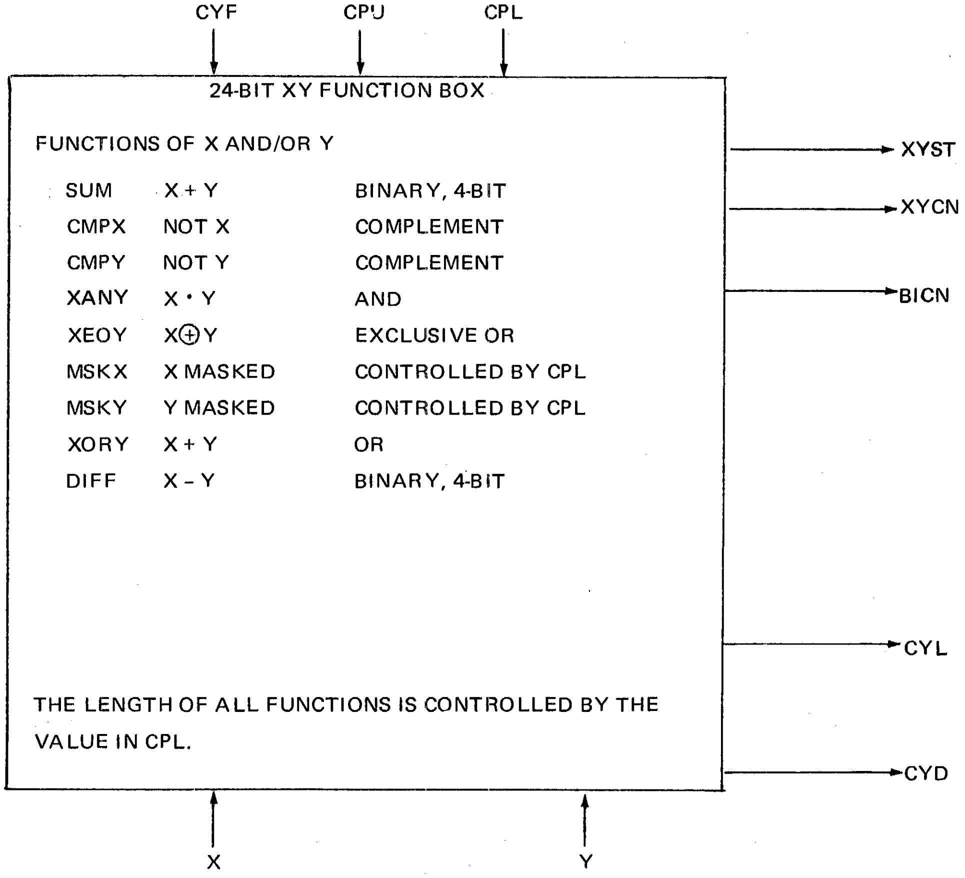
NULL REGISTER.
The NULL register is a 24-bit pseudo register which contains zeros. Moves from NULL may be used for clearing various registers to zeros.

THE ARITHMETIC AND COMBINATORIAL SECTION (24-BIT FUNCTION BOX).
The 24-bit arithmetic and combinatorial section (figure 2-8) is composed of a 24-bit arithmetic unit and a 24-bit combinatorial unit. It has as data inputs the contents of the X and Y registers and the carry indicator (CYF). It also uses the CPU and CPL portions of the C register.

All results from the combinatorial section are generated immediately and are continuously available to the micro-programmer. A move to one of the input registers or an alteration of a value in the CYF, CPU, or CPL registers immediately generates a new result. The results are accessed by moving the contents of a result register to a destination register or by testing one of the 4-bit condition registers.

The results are most of the commonly used functions between two operands. These include the AND, OR, EXCLUSIVE-OR, sum, carry out, difference, and borrow functions, and the set of equal to, greater than, and less than relationals. The results of the unary operations of complementation and masking are also available.

The results of the arithmetic unit are under control of the CPU and the CPL registers as follows.



FUNCTIONS OF X AND/OR Y

SUM	X + Y	BINARY, 4-BIT
CMPX	NOT X	COMPLEMENT
CMPY	NOT Y	COMPLEMENT
XANY	X • Y	AND
XEOY	X⊕Y	EXCLUSIVE OR
MSKX	X MASKED	CONTROLLED BY CPL
MSKY	Y MASKED	CONTROLLED BY CPL
XORY	X + Y	OR
DIFF	X - Y	BINARY, 4-BIT

THE LENGTH OF ALL FUNCTIONS IS CONTROLLED BY THE VALUE IN CPL.

Figure 2-8. Arithmetic and Combinatorial Section (24-Bit Function Box)

CPU	Unit Type	Possible CPY Values	Data Type
00	1-bit operands	1 to 24	Binary
01	4-bit operands	4, 8, 12, 16, 20, or 24	4-bit binary
10	Undefined		
11	Undefined		

For valid arithmetic operations, the operand length must be an exact multiple of the length of the unit specified by CPU.

Each of the following register contents are generated immediately and are available to the micro-programmer upon request.

SUM RESULT REGISTER.
The sum of the X, Y, and CYF registers (X, Y, and CYF are inputs to the 24-bit function box) is produced and placed in the 24-bit pseudo sum

register. Zeros in the more-significant bit positions (left-most) of the 24-bit result are produced when the length given by CPL is less than 24. CYF should be set to zero at the start of a sum generation.

The added logic performs differently for the four possible values in the CPU register as follows:

a. IF CPU = 00, the binary sum is produced.

b. If CPU = 01, the decimal sum is produced by considering the input to be comprised of up to six 4-bit units. The 4-bit units are added decimally and a carry is propagated from one 4-bit unit to the next more significant 4-bit unit whenever the binary sum of two 4-bit units exceeds 9.

c. If CPU = 10 or 11, the sum is not defined.

DIFFERENCE (DIFF) RESULT REGISTER.
The difference of the X, Y, and CYF registers (X, Y, and CYF are inputs to the 24-bit function box) is produced in the 24-bit pseudo DIFF register. Zeros in the more significant bit positions (left-most) of the 24-bit result are produced when the length determined by CPL is less than 24. Difference results are generated by adding the contents of X to the one's complement of both Y and CYF. The complement values are hardware generated and do not alter the values of Y or CYF.

The absolute value of (Y + CYF) may be greater than the value in X and produces a negative result. Such negative results are in complement form and are indicated by CYD (carry difference) = 1. If CYD = 0, the difference result value is a positive number. CYD is generated from all 24 bits of X and Y. It is not controlled by the value of CPL. CYD may be used to alter the value of CYF. Intermediate underflow may thus be remembered by CYF through iterations of a field.

The difference logic produces results under control of the CPU register as follows:

a. If CPU = 00, the binary difference is produced. Negative results are expressed in 2's complement form when CYD = 1.

b. If CPU = 01, the decimal difference is produced by considering the input to be comprised of up to six 4-bit units. The 4-bit units are subtracted and any borrow is propagated from one 4-bit unit to the next as in decimal subtraction. Negative results are expressed in 10's complement form when CYD = 1.

c. If CPU = 10 or 11, the difference is not defined.

AND/OR/EXCLUSIVE-OR (XANY, XEOY, XORY) RESULT REGISTERS.
The results of the appropriate logical function, AND/OR/EXCLUSIVE-OR, of the X and Y registers (X and Y are inputs to the 24-bit function box) are produced and placed in the appropriate 24-bit pseudo register (XANY, XEOY, XORY, respectively). Zeros in the more significant bit positions are produced when the length determined by CPL is less than

24.

COMPLEMENT X/COMPLEMENT Y (CMPX, CMPY) RESULT REGISTERS.
The one's complement of the appropriate register X or Y (X or Y and CPL are inputs to the 24-bit function box) is produced and placed in the appropriate 24-bit pseudo register, CMPX or CMPY, respectively. Zeros in the more significant bit positions of the 24-bit result are produced when the length given by CPL is less than 24.

MASKED X/MASKED Y (MSKX, MSKY) RESULT REGISTERS.
The mask of the contents of the appropriate register X or Y (X or Y and CPL are inputs to the 24-bit function box) is produced and placed in the appropriate 24-bit pseudo register, MSKX or MSKY, respectively (see figure .2-9). The value of CPL determines the number of bits placed in MSKX or MSKY. Zeros in the more significant bit positions of the 24-bit result are produced when the length given by CPL is less than 24.

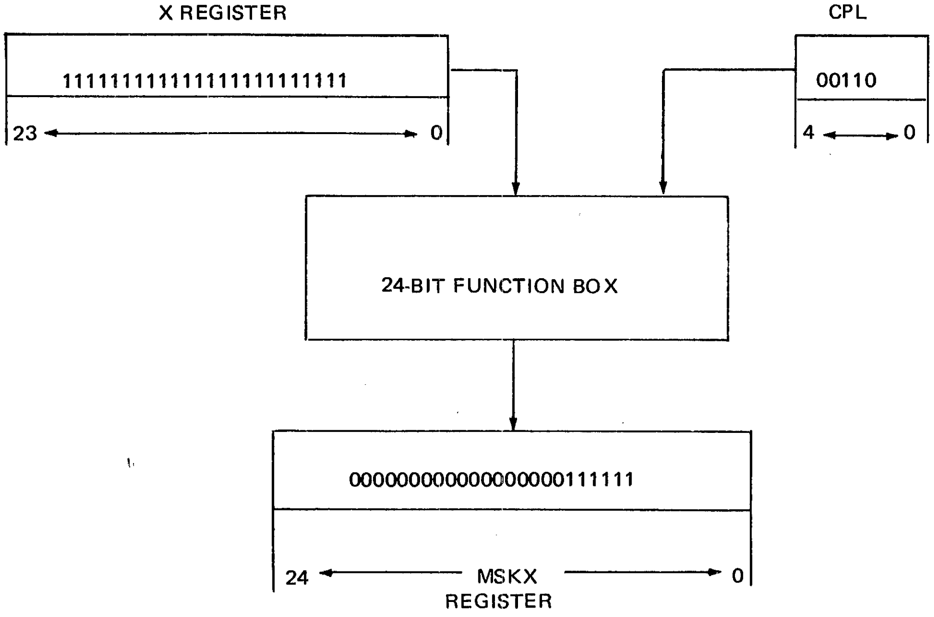


Figure .2-9. MSKX Result Register

BINARY CONDITIONS (BICN) REGISTER.
The BICN register (figure .2-10) is a 4-bit pseudo register which may act as a source only and indicates the following binary conditions: CYL (carry out level), CYD (borrow out level), CYF (carry flag), and LSUY (least significant unit of Y).

LSUY	CYF	CYD	CYL
3	2	1	0

Figure .2-10. BICN Register

The LSUY is true if the least significant unit of the Y register is equal to 1 and CPU equals 00, or the least significant unit of the Y register is equal to 1001 (nine) and CPU is equal to 01. If CPU equals 10 or 11, LSUY is not defined.

The carry out and borrow out levels are a function of the inputs (X, Y, and CYF) and CPL. CYF may be manipulated by a special micro-instruction or as data in the high-order position of the CP register.

X/Y CONDITION (XYCN) REGISTER.
The XYCN register (figure 2-11) is a 4-bit pseudo register which may act as a source only and contains the following relational conditions: X > Y (X is greater than Y), X < Y (X is less than Y), X = Y (X equal Y), and MSBX (the most significant bit of X).

MSBX	X = Y	X<Y	X>Y
3	2	1	0

Figure 2-11. XYCN Register

The relevant bits of XYCN are SET by comparing X and Y as positive 24-bit quantities. The MSBX is true if the most-significant bit of the X register, as designated by CPL, is a 1.

X/Y STATE (XYST) REGISTER.
The XYST register (figure 2-12) is a 4-bit pseudo register which may act as source only and contains the following relational conditions: X ≠ 0 (X is not equal to zero), Y ≠ 0 (Y is not equal to zero), Interrupt (INT is true if any of the following conditions contained by CC and CD are true: timer interrupt, I/O service request interrupt, console interrupt, and memory parity error interrupt), and LSUX (the least significant unit of X).

LSUX	INT	Y≠0	X≠0
3	2	1	0

Figure 2-12. XYST Register

LSUX is true when the least significant unit of the X register is a 1 and CPU equals 00, or when the least significant unit of the X register equals 1001 (nine) and CPU equals 01. If CPU equals 10 or 11, LSUX is not defined.

The relational conditions are based on the binary value of all 24 bits of X or Y.

FOUR-BIT ARITHMETIC AND COMBINATORIAL SECTION (FOUR-BIT FUNCTION BOX).
The 4-bit arithmetic and combinatorial section of the processor is used to generate most of the normally used functions between two 4-bit oper-

ands and can accept as input the contents of any one of the following 4-bit registers.

TA	TB	TC	TD	TE	TF
LA	LB	LC	LD	LE	LF
FU	FT	FLC	FLD	FLE	FLF
CA	CB	CC	CD		
BICN	XYCN	XYST	FLCN		

A second input is obtained from the Four-Bit Manipulate micro-instruction itself. It has as possible results the commonly used functions between two operands. these include: SET, AND, OR, EXCLUSIVE-OR, Binary Modulo 16 Sum, and Binary Modulo 16 Difference functions. The results may only be directed to the same register which acted as the source. The sum and difference result can be tested for overflow and underflow, respectively.

The BICN, XYCN, XYST, and FLCN registers are available as source registers only.

The 4-bit combinatorial section also provides for the selective testing of any of the bits within the 4-bit registers and relative branching based on the results of the test (Bit-Test-Branch, micro-instructions). The Skip When micro-instruction also tests any combination of bits, up to four, and branches on the result.

Listed in table 2-1 is a summary of the various 4-bit conditions which are available to the micro-programmer. (Refer to table 1-3 for the registers and functions in the column-row matrix.)

Table 2-1

4-Bit Conditions

Register	Bit 3	Bit 2	Bit 1	Bit 0
BICN	LSUY	CYF	CYD	CYL
XYCN	MSBX	X = Y	X < Y	X > Y
XYST	LSUX	INT	Y ≠ 0	X ≠ 0
FLCN	FL = SFL	FL > SFL	FL < SFL	FL ≠ 0
CC	Console state lamp	Timer interrupt	I/O service interrupt request	Console interrupt
CD	Memory parity interrupt	Reserved	Reserved	Reserved

SECTION 3

B 1726 PROCESSOR

GENERAL.

This section discusses the B 1726 processing unit which consists of the combinatorial portion of the system along with the various addressing and data manipulation registers.

PROCESSING UNIT.

The processor is the general-purpose data manipulator of the system. It performs the basic micro-functions necessary for a variety of commercial, scientific, and data communications applications. These micro-functions are performed by a set of elementary operators called micro-instructions. Micro-instructions operate on strings of bits within registers that were formerly the responsibility of the hardware logic.

By stringing a series of micro-instructions together, a macro-operation comparable to that defined by an assembler or compiler-level instruction may be implemented. This micro-instruction string is an interpreter for a macro ".S-instruction" operation (refer to S-Instructions). Micro-instruction strings are executed in a "read-only" mode and are usually stored in the high-speed control memory of the system. They may, however, be executed directly from the main memory of the system. Control memory may also be overlaid with micro-instructions from main memory.

CONTROL MEMORY.

This Read/Write memory resides within the processing unit and is used only to hold micro-instructions. If the set of micro-instructions in use exceeds available control memory, the excess micro-instructions are stored in, and can be executed from, main memory. The processor properly addresses the micro-instructions in either location (refer to Micro-Instruction Addressing). Micro-instructions which are being executed from main memory, however, result in a decrease in system throughput. This decrease occurs because control memory is four times faster than main memory. Micro-instruction sequences that are executed many times are located in control memory. Micro-instructions may be overlaid from main memory into control memory. The larger the amount of control memory the greater the system throughput.

Control memory size is 1024 or 2048 words. Each word is 16 bits wide to accommodate one micro-instruction. The micro-programmer should locate the sequences of code according to their expected frequency of usage. The greater the frequency of usage, the lower their address in control memory. This is done to give the highest execution speed possible for the system, regardless of the installed size of control memory. Micro-instructions which normally would have been stored in the missing portion of control memory are located in available main memory and addressed directly. Control memory operations are overlapped with main memory operations except for the Read, Write, Dispatch, and Overlay micro-instructions. Control memory operates in a "read only" mode, with the two exceptions of the Overlay micro-instruction and the console control panel in the halt/display mode.

MICRO-INSTRUCTION ADDRESSING.

To facilitate the fetch and execution of micro-instructions which are located in control memory and/or main memory, three hardware registers and certain decision-making logic have been implemented. The three registers are the A register (14-bit address), the TOPM register (top of control memory), and the MBR register (micro-instruction base). The addressing logic operates in the following manner.

As each micro-instruction is fetched, the A register is automatically incremented by 1. The micro-instruction just fetched is executed. Before fetching the next micro-instruction, the value in the A register is compared to (TOPM x 512). The TOPM register normally contains a value equal to the number of bytes (MOD 1000) of control memory present on the system, so that multiplying TOPM by 512 yields the number of the first micro-instruction outside control memory.

If the value in the A register is less than (TOPM x 512), the next micro-instruction is fetched from control memory at the address given in the A register. The process then returns to the initial micro-instruction fetched.

If the value in the A register is equal to or greater than (TOPM x 512), the next micro-instruction is fetched from main memory. The bit address of that instruction is obtained from the following formula:

(A x 16) + MBR

The MBR register contains the base address above which micro-instructions in main memory are stored. The process then returns to the micro-instruction fetched. If the address generated by the above formula is equal to or greater than the value in the MAXS register, a memory-read-address-out-of-bounds interrupt occurs when the fetch is attempted.

Figure 3-1 illustrates the above sequence of operations.

For example, suppose there are 4K bytes (2048 words) of control memory and 64K bytes (524,288 bits) of main memory installed on the system. In this case, the TOPM register should be set to 4 by the MCP to indicate the size of control memory. If the MBR register is set by the MCP to bit address 426,984, there is sufficient space for 4096 micro-instructions to be stored in main memory. Micro-instructions are fetched from control memory until the A register is equal to 2048 (TOPM x 512), indicating that the size of control memory has been exceeded. At this time, the processor begins to fetch micro-instructions from main-memory, starting at bit address 459,752 ((A x 16) +MBR). This process continues until the A register is reset by a micro-instruction or until the end of main memory is reached (((A x 16) + MBR) > MAXS). In the first case, a branch is executed to a different area of control or main memory; in the second case, a memory-read-address-out-of-bounds interrupt occurs.

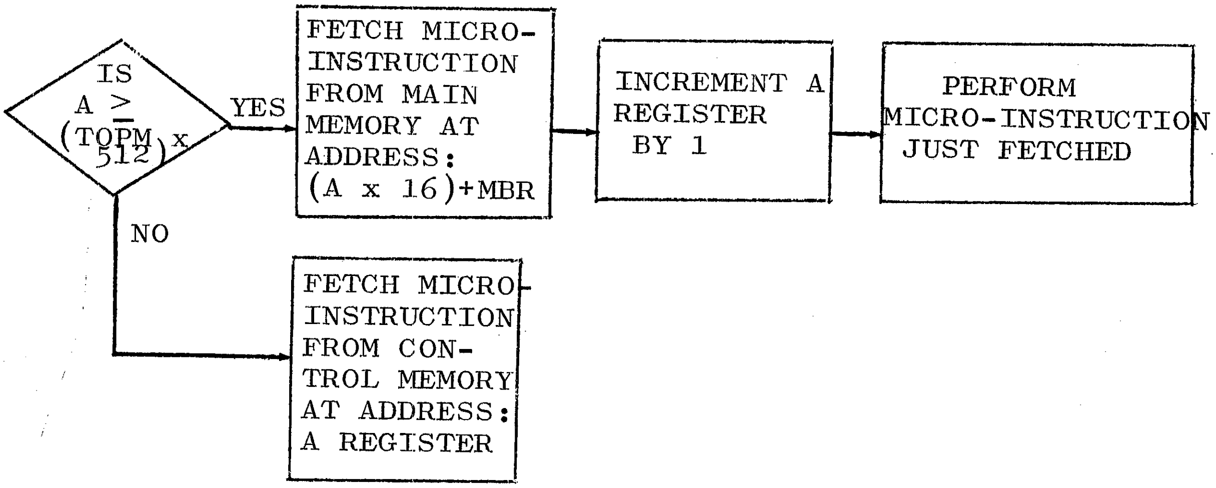

IS A ≥ (TOPM x 512)
YES
NO
FETCH MICRO-INSTRUCTION FROM MAIN MEMORY AT ADDRESS: (A x 16)+MBR
INCREMENT A REGISTER BY 1
PERFORM MICRO-INSTRUCTION JUST FETCHED
FETCH MICRO-INSTRUCTION FROM CONTROL MEMORY AT ADDRESS: A REGISTER

Figure 3-1. Micro-Instruction Fetch and Execution

Note the apparent discrepancy between the value in the MBR register and the address where the first micro-instruction is fetched from main memory. The area between the two addresses is exactly the size of the installed control memory (as indicated by (TOPM x 512)), and is available for data or code storage by S-instruction programs.

PROCESSOR REGISTERS.

MICRO-INSTRUCTION (M) REGISTER.

The M register (micro-instruction) is a 16-bit register used to hold and decode the current micro-instruction which has been obtained from control memory or main memory.

The M register is addressable as a 16-bit source or destination register. The right-most 16 bits, if more than 16 are transferred, are loaded. Each micro-instruction to be loaded from control memory is ORed with whatever may be in the M register. Thus, moves to the M register allow modification of the next micro-instruction without altering the original micro-instruction. The state of this register is decoded by the processor to enable various control levels to perform the operation called for by the micro-instructions.

CONTROL/MICRO-STRING MEMORY (MSM) REGISTER.

The MSM register is a 16-bit pseudo register which is addressable as a source when the console control panel is in a halt/display mode or as a destination register when the console control panel is in the tape mode. As a source register MSM contains the micro-instruction which is pointed to by the A register. It is addressable as a destination register by the Move 24-Bit Literal micro-instruction and by the Register Move micro-instruction in the tape mode.

ADDRESS (A) REGISTER.

The A register (address) is a 14-bit register capable of addressing up to 16,384 micro-instructions (each 16 bits in length) located in main memory and/or control memory (see figure 3-2). The A register is capable of having binary increments from 0 to 4095 added to or subtracted from it, with a high-speed carry adder to facilitate micro-program branching. The A register is automatically incremented by one as each micro-instruction is fetched.

BINARY ADDRESS	0 0 0 0
13 0 17 4	3 0

Figure 3-2. A Register

The A register is addressable as a source or a destination. When addressed as a source, its contents are automatically multiplied by 16. When used as a destination, the right-most four bits of the source are lost. These functions are accomplished by the presence of four non-functioning bits appended to the low-order end of the register which are always zero. When used as a source, the A register is actually transferred as an 18-bit field, thus automatically accomplishing the multiplication by 16. As a destination, the source is moved right-justified into the 18-bit destination field, with the four low-order bits masked to zeros. These extra four bits cannot be addressed and are completely transparent to the micro-programmer.

TOP OF CONTROL MEMORY (TOPM) REGISTER.

The TOPM register (top of control memory) is a 4-bit register which may act as a source or a destination (see figure 3-3). This register is automatically multiplied by 512 and compared with the address in the A register to determine whether the next micro-instruction should be fetched from main memory or control memory. This automatic multiplication is accomplished by the presence of nine nonfunctioning bits which are always zero, appended to the low-order end. These extra nine bits cannot be addressed and are completely transparent to the micro-programmer. For each 1K bytes of control memory the TOPM register is incremented by 1.

0 1 0 0	0 0 0 0 0 0 0 0 0
3 0 TOPM	

Figure 3-3. TOPM Register

MICRO-INSTRUCTION BASE REGISTER (MBR).

The MBR (micro-instruction base register) is a 24-bit register which may act as a source or a destination. It is used to contain the base address (modulo 16) above which all micro-instructions stored in main memory are located. The micro-instructions do not begin exactly at the address in the MBR register.

The address in main memory of the next micro-instruction to be fetched is equal to the contents of the A register times 16 added to the contents of MBR, or (A x 16) + MBR.

ADDRESS (A) STACK.

The A Stack (figure 3-4) is a 32-element deep, 24-bit wide memory which operates as a push-down stack, i.e., a last in, first out (LIFO) structure. Address wraparound occurs when more than 32 entries to the stack are made. Data is not destroyed on a removal from the stack. The micro-instructions Call or Move to TAS (top of A stack) result in a "put into" (push) stack request, whereas Exit and Move from TAS result in a "take out of" (pop) stack request.

Using this stack, the micro-routines operate in a call-return mode which allows for highly shared micro-routines and reduces the control memory requirements. The A stack is not intended to be used extensively as an operand stack; it has been made 24 bits wide to allow for operand storage.

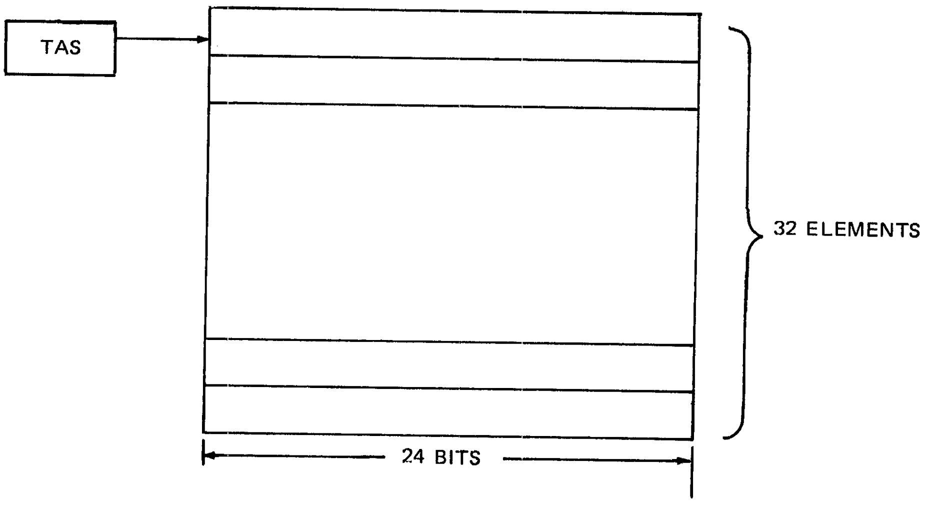


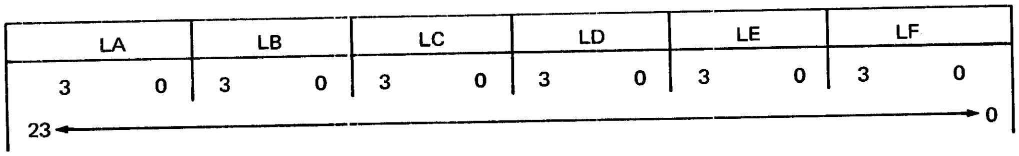
Figure 3-4. A Stack

TOP OF ADDRESS STACK (TAS) REGISTER.

The TAS register (top of the A stack) is the 24-bit register which is currently the top of the A stack. Moves to or from the TAS register result in data in the A stack being automatically moved to or from the A stack on a last in, first out basis.

GENERAL PURPOSE REGISTERS (L, T, X, Y).

L REGISTER. The L register (figure 3-5) is a 24-bit, general-purpose register. It is addressable as a source or as a destination register, either 24 bits at a time or in 4-bit groups. Since the L register is addressable in 4-bit groups, its contents are available for analysis and alteration via the 4-bit function box. Manipulate, Skip, and Bit-Test-Branch micro-instructions may operate on the data in the L register. The L register is one of four registers (X, Y, L, T) capable of Read/Write/Swap operations with main memory.

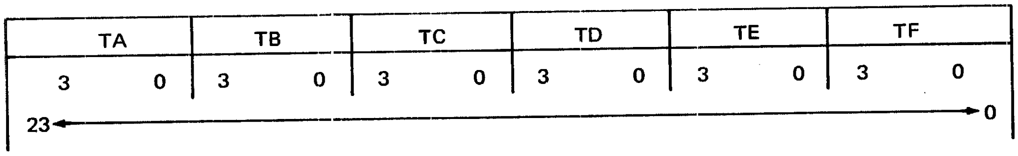


LA		LB		LC		LD		LE		LF	
3	0	3	0	3	0	3	0	3	0	3	0

Figure 3-5. L Register

The Dispatch micro-instruction uses the contents of the L register as the 24-bit address of the first I/O description in main memory.

T REGISTER. The T register (figure 3-6) is a 24-bit general-purpose register. It is addressable as a source or a destination register, either 24 bits at a time or in 4-bit groups. Since the T register is addressable in 4-bit groups, its contents are available for analysis and alteration via the 4-bit function box. Manipulate, Skip, and Bit-Test-Branch instructions can operate on the data in the T register. It is one of the four registers (X, Y, T, L) capable of Read/Write/Swap operations with main memory. The T register is capable of Shift/Rotate and Extract operations.



TA		TB		TC		TD		TE		TF	
3	0	3	0	3	0	3	0	3	0	3	0

Figure 3-6. T Register

X AND Y REGISTERS. The X register and the Y register are 24-bit, general-purpose registers. They are used primarily to hold and act as sources for two of the operands of the combinatorial section (input to the 24-bit function box) of the processor. The other operand is the CYF register (carry flip-flop). The X and Y registers are addressable as source and destination registers.

Both registers, along with the L register and the T register, are capable of Read/Write Swap operations with main memory. Both registers are capable of the Shift/Rotate operation.

FIELD (F) DEFINITION REGISTER.
The F register (field definition) specifies the address and lengths of data fields in main memory (see figure 3-7). It is a 48-bit register which is functionally divided into two portions: a 24-bit FA (field address) register and a 24-bit FB register which is divided into a 4-bit FU (field unit) register, a 4-bit FT (field type) register, and a 16-bit FL (field length) register. It may be loaded, stored, or swapped 48 bits in parallel with scratchpad memory.

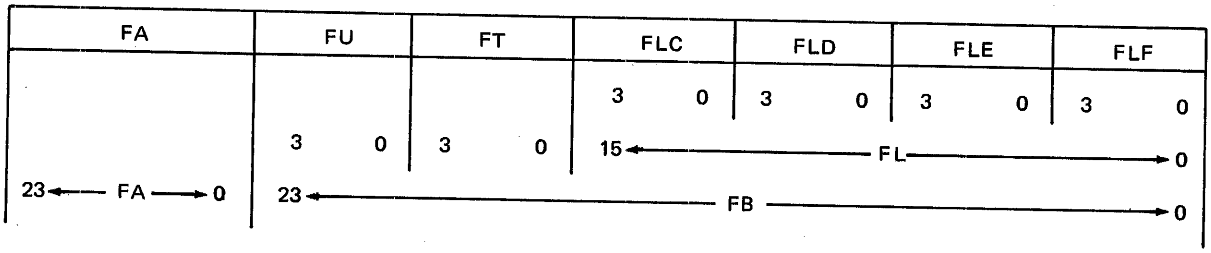


Figure 3-7. F Register

FA PORTION OF THE F REGISTER. The FA register holds the absolute binary main memory address of a data field and has the capability of directly addressing 16,777,216 bits without regard to physical boundaries. FA may be addressed as either a 24-bit source or destination register.

FB PORTION OF THE F REGISTER. FB may be considered as one 24-bit register or various subregisters: FU, FT, and FL. The FU register holds descriptive information regarding the units of data which make up the addressed field in main memory (binary, 4-bit, or 8-bit groups). The FT register holds additional descriptive information. The FU and FT values are usually supplied by S-instructions and the FU may also be used with the Bias micro-instruction to set the CPU and CPL registers. The FL register holds the total length in binary of the field being operated on in main memory, and is capable of describing fields up to 65,536 in length.

FB may be addressed as a source or destination or in part as FU (4 bits), FT (4 bits), and FL (16 bits). FL may also be addressed as 4-bit subfields (FLC, FLD, FLE, or FLF).

SCRATCHPAD.
A scratchpad of 16 words, each 48 bits wide, is provided to hold field descriptors of operands (see figure 3-8). Scratchpad may be addressed as 16 48-bit words (S00-S15) or 32 24-bit words (S00A-S15A, S00B-S15B). Any word may be a source or a destination and may be used to hold pointers, values, or any other information.

Revised 9/28/73 by
PCN 1057155-002

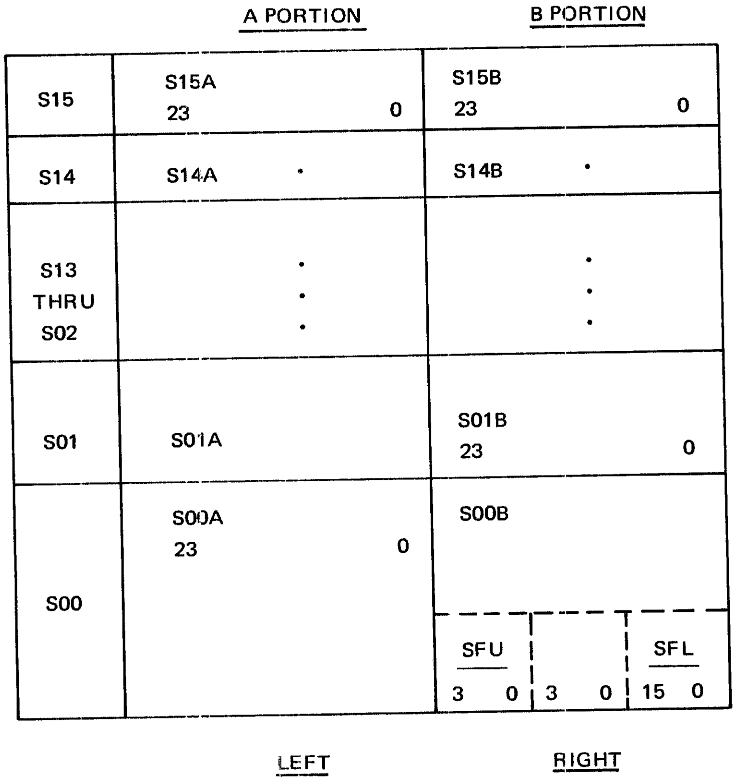


Figure 3-8. Scratchpad

The second half of the first 48-bit word (S00B), along with FU and FL, is always entered as input to the 4-bit function box and is used in certain decision-making logic. The field length value in the FL register is continually being compared against a corresponding portion (SFL) of the S00B word to determine the relationship of their contents: high, low, equal, or zero. This relationship may then be used in the Bit-Test-Brahch micro-instruction (refer to FLCN register). The FU and FL portions of the F register and the like portions (SFU and SFL) of S00B may also be used in the combinatorial section to set the value in the CP portion of the C register.

FIELD LENGTH CONDITIONS (FLCN) REGISTER. The FLCN register (figure 3-9) is a 4-bit register which contains the result of a comparison between the FL portion of the F register and the corresponding portion of the first scratchpad word (right-most 16 bits of S00B) which is called SFL. It has the following interpretation:

FL ≠ 0 - FL not equal to 0

FL<SFL - FL less than SFL
FL>SFL - FL greater than SFL
FL=SFL - FL equal to SFL

FL=SFL	FL>SFL	FL<SFL	FL≠0
3	2	1	0

Figure 3-9. FLCN Register

The relevant bits of FLCN are SET by comparing FL and SFL as positive 16-bit quantities. A one bit in a position of the FLCN register indicates that the respective condition is true.

BASE AND LIMIT REGISTERS (BR, LR).
The base register (BR) and limit register (LR) may be used as source or destination registers. They are used for main memory protection and for base relative addressing.

Memory protection is provided for in the MCP by checking the main memory address (in FA) on all Read/Write/Swap operations with the BR and LR allowing the operation to take place only within those limits. All out-of-bound requests, whether allowed or not, are flagged in the CD portion of the C register (CD bits 0 and 1). Any Write operation or the write portion of a Swap operation outside the values in the base and limit registers is inhibited unless the out-of-bounds override bit is on in the C register (CD bit 2 = 1). Read out-of-bound operations are executed.

Memory protection is only provided on the main memory address in the FA register and not on any of the 23 adjacent bits when the value in the field length (FL) register is greater than one.

CONTROL (C) REGISTER.
The C register (control) is a collection of independent registers which are utilized by the interrupt system of the processor and the combinatorial section. The C register is 24 bits wide and is divided into three 8-bit functional parts (see figure 3-10).

CA	CB	CC	CD	CP		
				CYF	CPU	CPL
3 0	3 0	3 0	3 0	0	1 0	4 0
8 BITS		8 BITS		8 BITS		
24 BITS						

Figure 3-10. C Register

CA AND CB REGISTERS. This portion of the C register is addressable in 4-bit units as CA and CB. These portions are available as general-purpose storage registers.

CC AND CD REGISTERS. This portion of the C register is addressable in two 4-bit units defined as CC and CD. CC and CD are used for storage of the processor states and for processor interrupts as described in the following paragraphs.

<u>CC Register.</u>
The bit meanings for the CC register are as follows:

a. Bit 0 - console interrupt.
b. Bit 1 - I/O service request interrupt.
c. Bit 2 - timer interrupt.
d. Bit 3 - console state.

The console interrupt bit is set from the console INT (interrupt) switch and allows the software to come to an orderly stop. The I/O service request interrupt bit is set by an I/O controller when it issues a service request to the processor. The timer interrupt bit is set by the hardware timer every 100 milliseconds. Once the timer interrupt is set, it stays set until the MCP resets it. The console STATE light is also under control of the micro-programmer.

<u>CD Register.</u>
The bit meanings for the CD register are as follows:

a. Bit 0 - memory Write/Swap address out of bounds interrupt.
b. Bit 1 - memory Read address out of bounds interrupt.
c. Bit 2 - memory Write/Swap address out of bounds override.
d. Bit 3 - memory parity error interrupt.

The out-of-bound signals used to set CD bits 0 and 1 are derived from logic which compares the contents of the FA register with the contents of the base (BR) and limit (LR) registers. The out-of-bounds override bit (CD bit 2 set to 1) allows a Write operation or the write portion of a Swap operation to be executed but does <u>not</u> inhibit the setting of the Write/Swap address out-of-bounds bit (CD bit 0). The Read cycle is never inhibited but sets the Read Address out-of-bounds bit (CD bit 1) if the address is not between the values in the base and limit registers. Any parity error detected during a main memory operation sets the memory parity error interrupt (CD bit 3). All interrupts in the processor are soft and no reaction occurs as a result of an interrupt bit being set until the micro-program tests for such settings.

CYF, CPU, AND CPL REGISTERS. The least significant eight bits of the C register (CP) are composed of the arithmetic unit carry flip-flop (CYF), the 2-bit arithmetic unit type (CPU), and the 5-bit combinatorial data length control (CPL). Moves from CP to a 24-bit register automatically insert 16 leading zeros, and moves to CP from any register use only the least significant eight bits of the field being moved.

The Carry micro-instruction is supplied for manipulating CYF and thereby remembering the status of a carry or borrow beyond the length of an operand. For SUM and DIFF results, CYF is always an input to the combinatorial section along with the X and Y registers. CPU is addressable only as a destination. CPU controls the arithmetic unit type of the combinatorial section. The possible settings and their meanings are listed below.

CPU	Data Type
00	Binary
01	4-bit binary
10	Undefined
11	8-bit numeric (EBCDIC)

CPL controls the length of the operands being entered as input to the combinatorial section and should be an integral multiple of the data type specified by CPU for valid arithmetic results. The maximum allowable value of CPL is 24, which is an integral multiple of all allowable data types. Larger values are reserved.

MAXIMUM SIZE OF MAIN MEMORY (MAXS) REGISTER.
The MAXS register is a 24-bit pseudo register which is set by a field engineer to indicate the maximum size of the installed main memory. The MAXS register is addressable as a source register only.

MAXIMUM SIZE OF CONTROL MEMORY (MAXM) REGISTER.
The MAXM register is a 24-bit pseudo register which is set by a field engineer to indicate the maximum size of the installed control (micro-instruction) memory. The MAXM register is addressable as a source register only.

TAPE CASSETTE INPUT (U) REGISTER.
The U register is a 16-bit register used to accumulate input (micro-instructions) from the console tape cassette. The U register is addressable only as a 16-bit source register.

Only the micro-instruction Register Move may access the U register. If data is not yet available in the register, the micro-operator is delayed. During TAPE mode the contents of this register is moved by a micro-program to the M register for execution if the previous micro-instruction did not move the contents out.

DATA REGISTER.
The DATA register is a 24-bit pseudo register which can act as a source or destination. It is used to transfer data to and from the respective I/O controllers. When used as a source, the DATA register accepts 24 bits of data from the respective I/O controller. When used as a destination, the DATA register transfers up to 24 bits of data from a source in memory to the I/O controllers.

COMMAND (CMND) REGISTER.

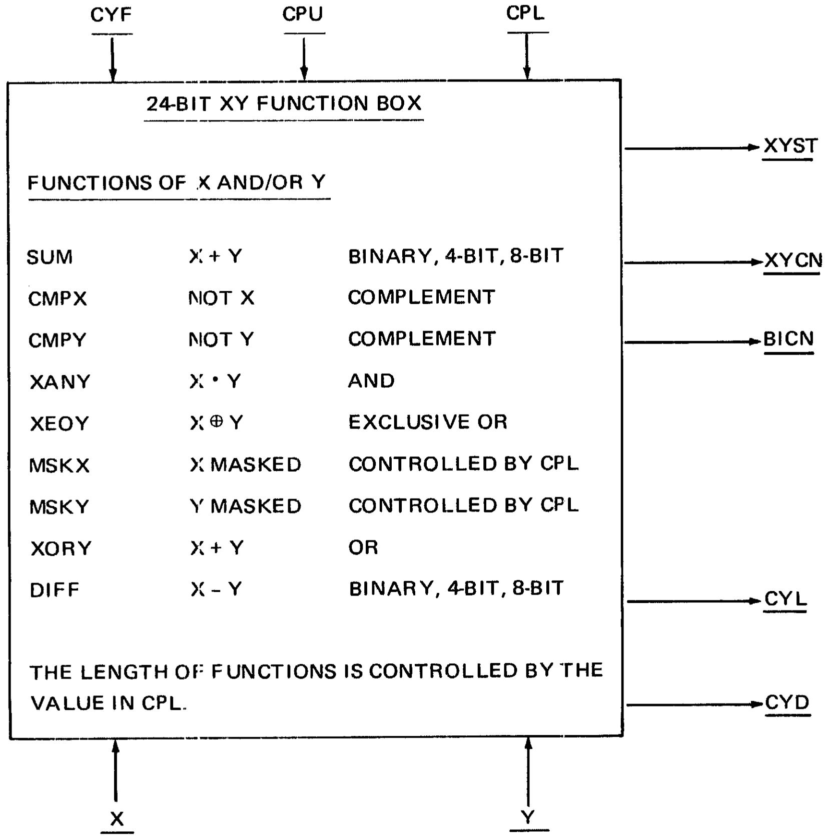
The CMND register (command) is a 24-bit pseudo register which can act as a destination only. It is used to transfer I/O commands to I/O devices on the respective I/O controller.

NULL REGISTER.

The NULL register is a 24-bit pseudo register which contains zeros. It may be addressed only as a source register. Moves from NULL may be used for setting various registers to zeros.

ARITHMETIC AND COMBINATORIAL SECTION (24-BIT FUNCTION BOX).

The 24-bit arithmetic and combinatorial section (figure 3-11) is composed of a 24-bit arithmetic unit and a 24-bit combinatorial unit. It has as data inputs the contents of the X and Y registers and the carry flip-flop (CYF). It also uses CPU (control for the arithmetic unit) and CPL (the 5-bit variable operand length) from the CP portion of the C register.



FUNCTIONS OF X AND/OR Y

SUM	X + Y	BINARY, 4-BIT, 8-BIT
CMPX	NOT X	COMPLEMENT
CMPY	NOT Y	COMPLEMENT
XANY	X • Y	AND
XEOY	X ⊕ Y	EXCLUSIVE OR
MSKX	X MASKED	CONTROLLED BY CPL
MSKY	Y MASKED	CONTROLLED BY CPL
XORY	X + Y	OR
DIFF	X - Y	BINARY, 4-BIT, 8-BIT

THE LENGTH OF FUNCTIONS IS CONTROLLED BY THE VALUE IN CPL.

Figure 3-11. Arithmetic and Combinatorial Section

All results from the combinatorial section are generated immediately and are continuously available to the micro-programmer. A move to one of the input registers or an alteration of a value in the CP portion of the C register immediately generates a new result. The results are available to the next micro-instruction and are accessed by moving the contents of a result register to a destination register or by testing one of the 4-bit condition registers.

The results are most of the commonly used functions between two operands. These include the AND, OR, EXCLUSIVE-OR, sum, carry out, difference, and borrow functions and the set of equal to, greater than, and less than relationals. The results of the unary operations of complementation and masking are also available.

The results of the arithmetic unit are under control of the CPU and the CPL registers as follows.

CPU	Unit Type	Possible CPL Values	Data Type
00	1-bit operands	1 to 24	Binary
01	4-bit operands	4, 8, 12, 16, 20, or 24	4-bit binary
10	Undefined		
11	8-bit operands	8, 16, or 24	EBCDIC

For valid arithmetic operations, the operand length (as specified by CPL) must be an exact multiple of the length of the unit specified by CPU.

Each of the following register contents is generated immediately and is available to the micro-programmer.

SUM RESULT REGISTER.
The sum of the X, Y, and CYF registers (X, Y, and CYF are entered as inputs to the 24-bit function box) is produced and placed in the 24-bit pseudo SUM register. Zeros in the most-significant bit positions (left-most) of the 24-bit result are produced when the length given by CPL is less than 24. CYF should be set to zero at the start of a sum generation.

The adder logic performs differently for the four possible values in the CPU register as follows:

a. If CPU = 00, the binary sum is produced.

b. If CPU = 01, the decimal sum is produced by considering the input to be comprised of up to six 4-bit units. The 4-bit units are added decimally and a carry is propagated from one 4-bit unit to the next most-significant 4-bit unit whenever the sum of two 4-bit units exceeds 9.

c. If CPU = 10, the sum is undefined.

d. If CPU = 11, the decimal sum is produced by considering the input to be comprised of up to three 8-bit units. The least-significant 4 bits of each unit are added in a manner similar to the 4-bit unit, including carries. The most-significant bits of each unit contain the inclusive-OR of the corresponding bits of the inputs.

DIFFERENCE (DIFF) RESULT REGISTER.
The difference of the X, Y, and CYF registers (X, Y, and CYF are entered as inputs to the 24-bit function box) is produced and placed in the 24-bit pseudo DIFF register. Zeros in the most-significant bit positions (left-most) of the 24-bit result are produced when the length given by CPL is less then 24. Difference results are generated by adding the contents of X to the one's complement of both Y and CYF The complement values are hardware generated and do not alter the values of Y or CYF.

The absolute value of (Y + CYF) may be greater than the value in X and produces a negative result. Such negative results are in complement form and are indicated by CYD (carry difference) = 1. If CYD = 0, the difference result value is a positive number. CYD is generated from all 24 bits of X and Y and is not controlled by the value in CPL. CYD may be used to alter the value of CYF. Intermediate underflow may thus be remembered by CYF through iterations of a field.

The difference logic produces results under the control of the CPU register as follows:

a. If CPU = 00, the binary difference is produced. Negative results are expressed in 2's complement form when CYD = 1.

b. If CPU = 01, the decimal difference is produced by considering the input to be comprised of up to six 4-bit units. The 4-bit units are subtracted and any borrow is propagated from one 4-bit unit to the next as in decimal subtraction. Negative results are expressed in 10's complement form when CYD = 1.

c. If CPU = 10, the difference is undefined.

d. If CPU = 11, the decimal difference is produced by considering the input to be comprised of up to three 8-bit units. The least-significant 4 bits of each unit are subtracted in a manner similar to the 4-bit units including borrows. The most-significant bits of each unit contain the inclusive-OR of the corresponding bits of the inputs. Negative results are expressed in 10's complement form when CYD = 1.

AND/OR/EXCLUSIVE-OR (XANY, XEOY, XORY) RESULT REGISTERS.
The result of the appropriate logical function AND/OR/EXCLUSIVE-OR of the X and Y registers (X and Y are inputs to the 24-bit function box) is produced and placed in the appropriate 24-bit pseudo register (XANY, XEOY, XORY). Zeros in the most-significant bit positions in the 24-bit

result are produced when the length given by CPL is less than 24.

COMPLEMENT X/COMPLEMENT Y (CMPX, CMPY) RESULT REGISTER.
The one's complement of the appropriate register X or Y (X or Y and CPL are inputs to the 24-bit function box) is produced and placed in the appropriate 24-bit pseudo register, CMPX or CMPY, respectively. Zeros in the most-significant bit positions of the 24-bit result are produced when the length given by CPL is less than 24.

MASKED X/MASKED Y (MSKX, MSKY) RESULT REGISTER.
The mask of the contents of the appropriate register X or Y (X or Y and CPL are inputs to the 24-bit function box) is produced and placed in the appropriate 24-bit pseudo register, MSKX or MSKY, respectively (see figure 3-12). The value of CPL determines the number of bits in MSKX or MSKY. Zeros in the most-significant bit positions of the 24-bit result are produced when the length given by CPL is less than 24.

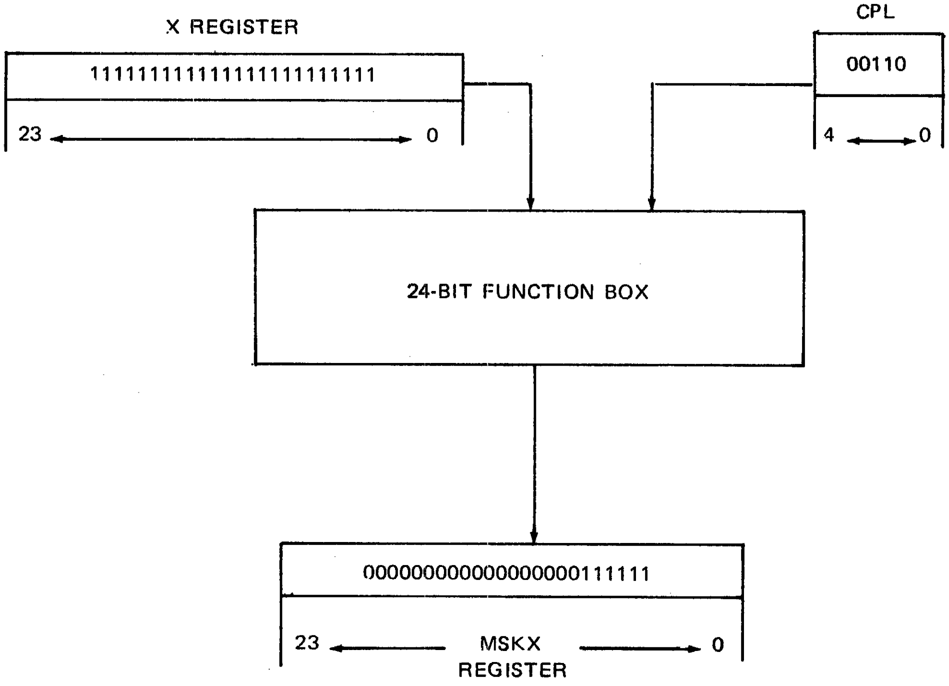


Figure 3-12. MSKX Register

BINARY CONDITIONS (BICN) REGISTER.
The BICN register (figure 3-13) is a 4-bit pseudo register which may act as a source only and indicates the following binary conditions: CYL (carry out level), CYD (borrow out level), CYF (carry flag), and LSUY (least significant unit of Y).

LSUY	CYF	CYD	CYL
3	2	1	0

Figure 3-13. BICN Register

The LSUY is true if: the least significant unit of the Y register is equal to 1 and CPU equals 00, or the right-most 4 bits of the least significant unit of the Y register is equal to 1001 (nine) and CPU is equal to 01 or 11. If CPU is equal to 10, LSUY is undefined.

The carry out and borrow out levels are a function of the inputs (X, Y, and CYF) and CPL. CYF may be manipulated by a special micro-instruction or as data in the high-order position of the CP register.

X/Y CONDITIONS (XYCN) REGISTER.
The XYCN register (figure 3-14) is a 4-bit pseudo register which may act as a source only and contains the following relational conditions: X>Y (X is greater than Y, X<Y (X is less than Y), X=Y (X equals Y), and MSBX (most significant bit of X).

MSBX	X=Y	X<Y	X>Y
3	2	1	0

Figure 3-14. XYCN Register

The relevant bits of XYCN are SET by comparing X and Y as positive 24-bit quantities. MSBX is true if the bit to which CPL points is a 1.

X/Y STATES (XYST) REGISTER.
The XYST register (figure 3-15) is a 4-bit pseudo register which may act as a source only and contains the following relational conditions: X = 0 (X is not equal to zero); Y = 0 (Y is not equal to zero); INT (interrupt) is true if any of the following conditions reflected by INCN, CC, and CD are true: missing port device, port interrupt, timer interrupt, I/O service request interrupt, console interrupt, memory parity error interrupt, and memory Write/Swap address out-of-bounds interrupt; LSUX (the least significant unit of X).

LSUX is true when the least significant unit of the X register is a 1 and CPU equals 00 or when the low-order 4 bits of the least significant unit of the X register equals 1001 (nine) and CPU is equal to 01 or 11. If CPU equals 10, LSUX is undefined.

LSUX	INT	Y≠0	X≠0
3	2	1	0

Figure 3-15. - XYST Register

FOUR-BIT ARITHMETIC AND COMBINATORIAL SECTION (FOUR-BIT FUNCTION BOX). The 4-bit arithmetic and combinatorial section of the processor is used to generate most of the normally used functions between two 4-bit operands and can accept as an input the contents of any one of the following 4-bit registers. A second input is obtained from the Four-Bit Manipulate micro-instruction itself.

TA	TB	TC	TD	TE	TF
LA	LB	LC	LD	LE	LF
FU	FT	FLC	FLD	FLE	FLF
CA	CB	CC	CD	TOPM	
BICN	XYCN	XYST	FLCN	INCN	

It has as possible results most of the functions between two operands. These include: SET, AND, OR, EXCLUSIVE-OR, binary modulo 16 sum, and binary modulo 16 difference functions. The results may only be directed to the same register which acted as the source. The sum and difference results can be tested for overflow and underflow, respectively.

The BICN, XYCN, XYST, FLCN, and INCN registers are available as source registers only.

The 4-bit combinatorial section also provides for the selective testing of any of the bits within the 4-bit registers and relative branching based on the results of the test (Bit-Test-Branch micro-instruction). The Skip When micro-instruction also tests any combination of bits, up to four, and branches on the result.

INTERRUPT CONDITIONS (INCN) REGISTER. The INCN register (figure 3-16) is a 4-bit register which contains the interrupt conditions of particular interface lines between the processor and port interchange.

PORT MISSING DEVICE	PORT HIGH PRIORITY INTERRUPT	PORT INTERRUPT (NORMAL)	PORT LOCKOUT
3	2	1	0

Figure 3-16. - INCN Register

Listed in table 3-1 is a summary of the various 4-bit conditions which are available to the micro-programmer. (Refer to table 1-3 for the registers and functions in the column-row matrix.)

Table 3-1

4-Bit Conditions

Register	Bit 3	Bit 2	Bit 1	Bit 0
BICN	LSUY	CYF	CYD	CYL
XYCN	MSBX	X=Y	X<Y	X>Y
XYST	LSUX	INT	Y≠0	X≠0
FLCN	FL=SFL	FL>SFL	FL<SFL	FL≠0
INCN	Port device missing	Port high priority	Port interrupt	Port lockout
CC	Unassigned	Timer interrupt	I/O interrupt	Console interrupt
CD	Memory parity error interrupt	Memory Write/Swap memory address out-of-bounds override	Memory Read address out-of-bounds interrupt	Memory Write/Swap address out-of-bounds interrupt

SECTION 4

MICRO-INSTRUCTIONS

GENERAL.

Micro-instruction routines can be related to read-only memories of past systems that contain a hard set of micro-programs which control most of the action taken by the processor. Because read-only memories are expensive and difficult to change once a system has been designed for a certain environment, it is, in most cases, destined to stay suited for that environment.

With the advent of Read/Write control memories and micro-programming techniques, Burroughs Corporation now provides the ability to change the system architecture to whatever is desired. By changing the micro-program (interpreter), a system efficient in COBOL can be changed to a system efficient in FORTRAN.

MICRO-INSTRUCTION DESCRIPTION.

All micro-instructions are 16 bits in length and are placed in the M register (micro-instruction) for execution. The M register (figure 4-1) is 16 bits long and is divided into four 4-bit sections which are not addressable as individual registers.

4 BITS				4 BITS				4 BITS				4 BITS				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT NUMBER
16 BITS																

Figure 4-1. M Register

The micro-instructions are referred to by name and hexadecimal value within the specific portion of the M register. For example, the Register Move micro-instruction has a hexadecimal value of 1 in the first 4-bit portion of the M register; the micro in machine code for a Register Move is 1. The Halt micro-instruction has a hexadecimal value of 1 in the last 4-bit portion of the M register; the micro in machine code for a Halt command is 0001. The other portions of the M register (if not part of machine code) are used for the selection of source and destination register and also for any variants associated with the respective micro-commands.

Source and destination register selection and other variables are determined by the hexadecimal value placed in the remaining 4-bit sections. For the Register Move command, these hexadecimal values point to the desired register as shown in table 4-1. If the hexadecimal value in the second 4-bit portion is equal to a 1100 (row C) and the most-significant two bits of the third 4-bit portion are equal to 01 (column 1), XYCN is selected as the source register for the Register Move micro-instruction.

Table 4-1

Register Select Switch
Column-Row Matrix

Row	Column 0	Column 1	Column 2	Column 3
0	TA	FU	X	SUM
1	TB	FT	Y	CMPX
2	TC	FLC	T	CMPY
3	TD	FLD	L	XANY
4	TE	FLE	A(MAR)	XEOY
5	TF	FLF	M	MSKX
6	CA	BICN	BR	MSKY
7	CB	FLCN	LR	XORY
8	LA	TOPM(*2)	FA	DIFF
9	LB	Reserved	FB	MAXS
A	LC	Reserved	FL	MAXM
B	LD	Reserved	TAS	U
C	LE	XYCN	CP	MBR(*2)
D	LF	XYST	MSM(*2)	DATA
E	CC	INCN(*2)	Console READ	CMND
F	CD	CPU	Console WRIT	NULL

REGISTER MOVE (1NNN).
The function of this micro-instruction (figure 4-2) is to move the contents of the source register to the destination register. If the move is between registers of unequal lengths, the data is right justified with left zero bits supplied or with data truncated from the left, whichever is appropriate. The source register is unaffected unless it is also the destination.

MSM and CPU are excluded as source registers.

When M is used as a destination register, the operation is changed to a Bit-OR which modifies the next micro-instruction to be executed. It does not modify the micro-instruction as stored in memory.

BICN, FLCN, XYCN, XYST, and INCN are excluded as destination registers.

MSM is not available as a destination register in STEP and RUN mode. It is permitted as a destination register in TAPE mode only.

[2] Not available on B 1710 Systems

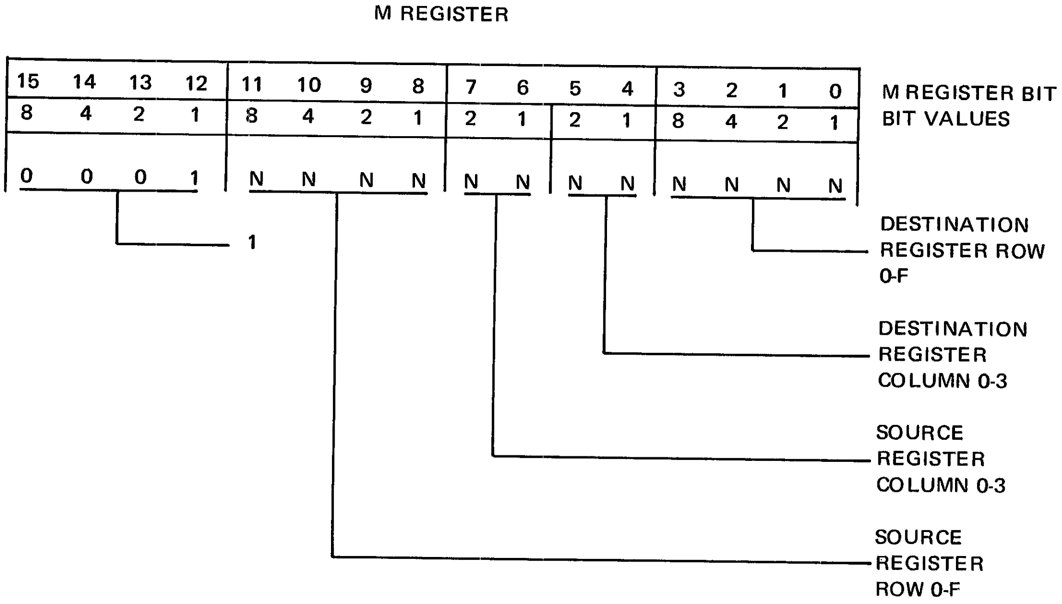


Figure 4-2. Register Move (1NNN)

The U register is excluded as a source register in STEP and TAPE mode but is permitted as a source register in the RUN mode.

All registers and pseudo registers in column select three are excluded as destination registers except MBR, CMND, and DATA.

When CMND or DATA is designated as a source register, CMND and DATA are prohibited from being destination registers.

SCRATCHPAD MOVE (2NNN).
The function of this micro-instruction (figure 4-3) is to move the contents of either scratchpad memory to the selected register or the contents of the selected register to scratchpad memory as designated by the micro-instruction. The contents of the source register/scratchpad word is unchanged by the move.

MSM, U, and CPU are excluded as source registers.

When M is used as a destination register, the operation is changed to a Bit-OR which modifies the next micro-instruction to be executed but does not modify the micro-instruction as stored in memory.

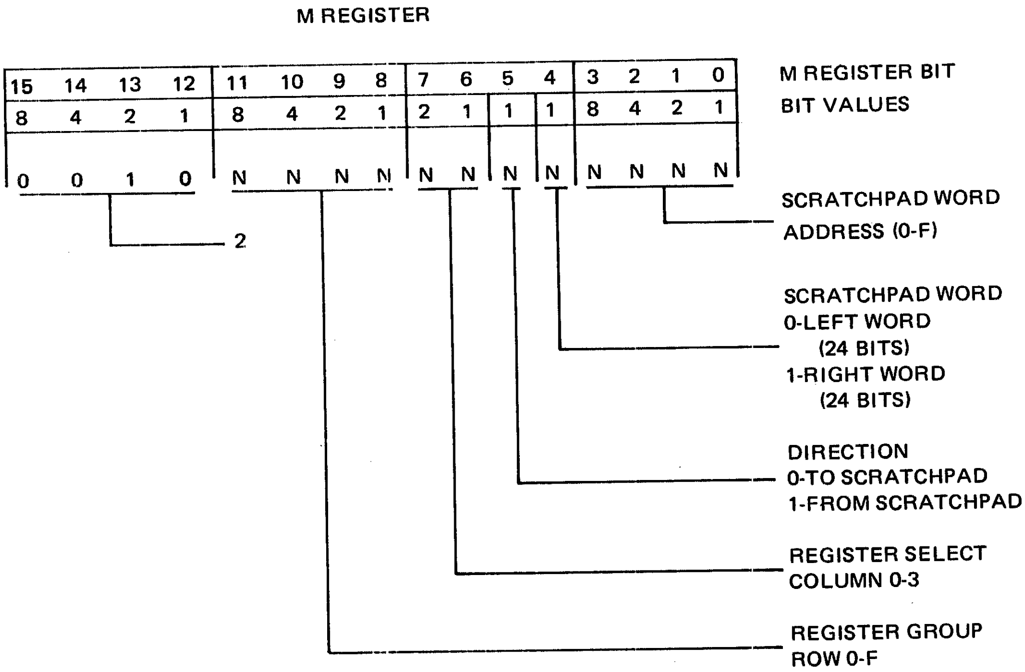


Figure 4-3. - Scratchpad Move (2NNN)

BICN, FLCN, XYCN, XYST, INCN, and MSM are excluded as destination registers.

All registers and pseudo registers in column select three are excluded as destination registers except MBR, CMND, and DATA.

FOUR-BIT MANIPULATE (3NNN).

The function of this micro-instruction (figure 4-4) is to perform the Manipulate operation as specified by the micro-instruction variants on the addressed 4-bit register utilizing the literal specified in MF.

Only two columns of registers are available to the Four-Bit Manipulate micro-instruction (columns 0 and 1). Bit seven contains either a 0 or 1 to enable selection of the registers in columns 0 or 1 as expressed in table 4-1.

Bits four through six designate the specific operation to be performed on the addressed register and its relationship with the 4-bit literal expressed in bits zero through three. The possible operations permitted by respective variant values are given below.

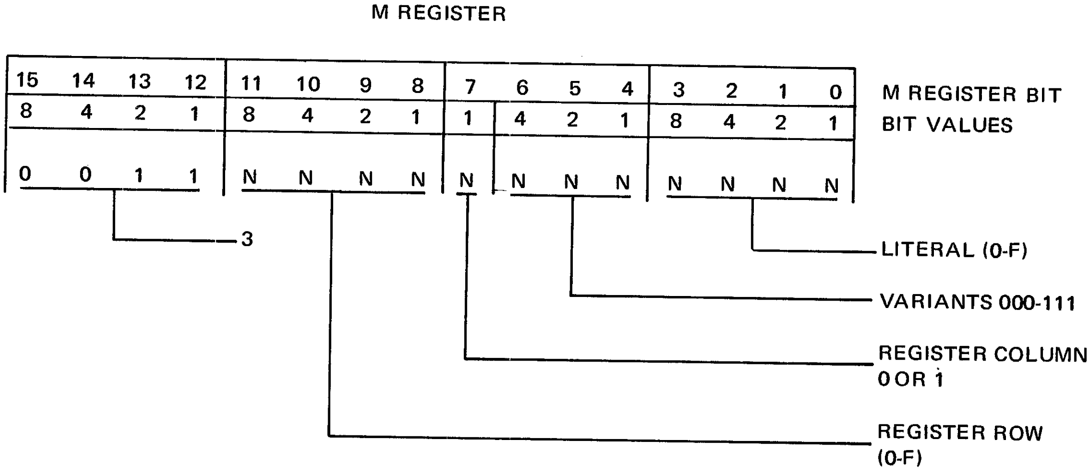
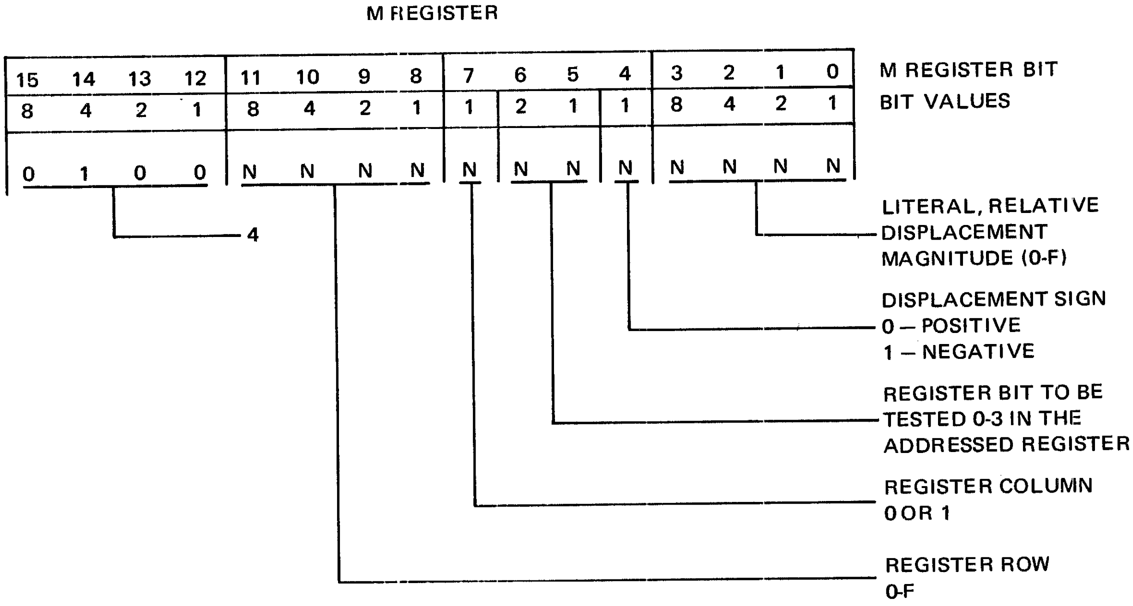


Figure 4-4. Four-Bit Manipulate (3NNN)

Variant	Operation
000	Set the addressed register equal to the value of the literal.
001	Set the addressed register to the logical AND of the addressed register and the literal.
010	Set the addressed register to the logical OR of the addressed register and the literal.
011	Set the addressed register to the logical EXCLUSIVE OR of the addressed register and the literal.
100	Set the addressed register to the binary sum of the addressed register and the literal.
101	Set the addressed register to the binary difference of the addressed register and the literal.
110	Set the addressed register to the binary sum of the addressed register and the literal, and skip the next micro-instruction if a carry is produced.
111	Set the addressed register to the binary difference of the addressed register and the literal, and skip the next micro-instruction if a borrow is produced.

BIT TEST BRANCH ON FALSE (4NNN).

The function of this micro-instruction (figure 4-5) is to test the designated bit (0-3) within the specified register and branch relative to the next instruction by the displacement magnitude value if the displacement sign bit is zero. If the displacement sign bit is one, a displacement value of zero is assumed and the next micro-instruction in line is executed. A displacement value indicates the number of 16-bit words from the next in-line micro-instruction. This value may not exceed 15 (hexadecimal F).


M REGISTER
4
LITERAL, RELATIVE DISPLACEMENT MAGNITUDE (0-F)
DISPLACEMENT SIGN 0 – POSITIVE 1 – NEGATIVE
REGISTER BIT TO BE TESTED 0-3 IN THE ADDRESSED REGISTER
REGISTER COLUMN 0 OR 1
REGISTER ROW 0-F

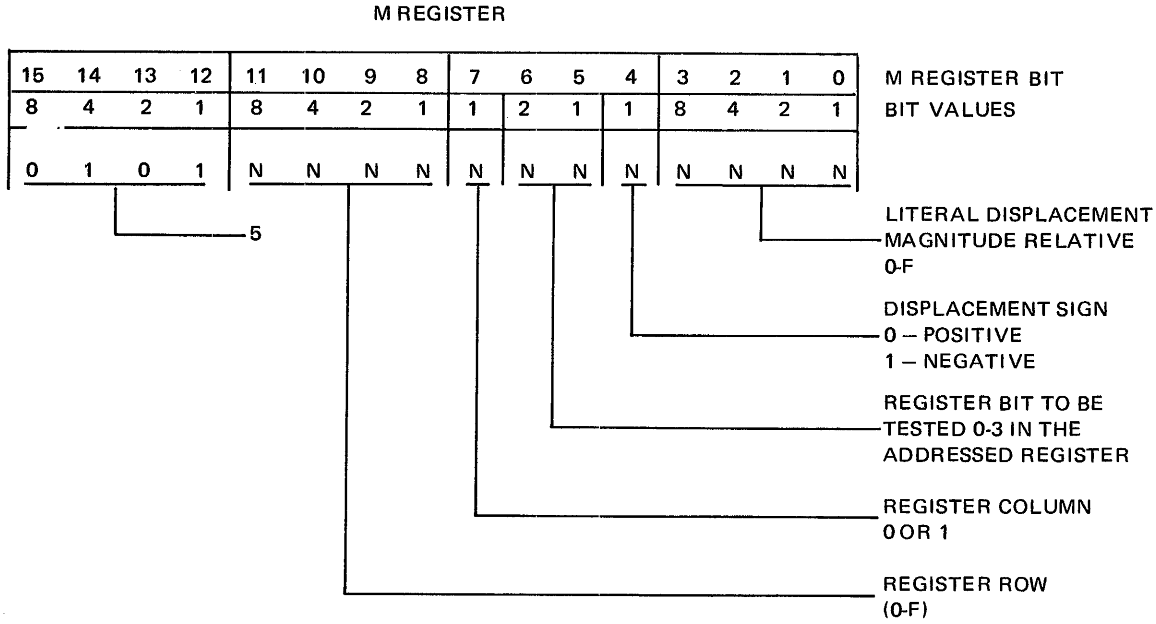
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	1	2	1	1	8	4	2	1	BIT VALUES
0	1	0	0	N	N	N	N	N	N	N	N	N	N	N	N	

Figure 4-5. Bit Test/Branch on False (4NNN)

Only the 4-bit registers shown in columns 0 and 1 in table 4-1 may be used as the addressed register on which this micro-instruction may act.

BIT TEST/BRANCH ON TRUE (5NNN).

The function of this micro-instruction (figure 4-6) is to test the designated bit (0-3) within the specified register and branch relative to the next instruction by the signed displacement value if the displacement sign bit is one. If the displacement sign bit is zero, a displacement value of zero is assumed and control passes to the next in-line micro-instruction. a displacement value indicates the number of 16-bit words (micro-instructions) from the next in-line micro-instruction. This value may not exceed 15 (hexadecimal F).



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	1	2	1	1	8	4	2	1	BIT VALUES
0	1	0	1	N	N	N	N	N	N	N	N	N	N	N	N	

5
LITERAL DISPLACEMENT MAGNITUDE RELATIVE 0-F
DISPLACEMENT SIGN 0 – POSITIVE 1 – NEGATIVE
REGISTER BIT TO BE TESTED 0-3 IN THE ADDRESSED REGISTER
REGISTER COLUMN 0 OR 1
REGISTER ROW (0-F)

Figure 4-6. Bit Test/Branch on True (5NNN)

Only the 4-bit registers shown in columns 0 and 1 in table 4-1 may be used as the addressed register on which this micro-instruction may act.

SKIP WHEN (6NNN).
The function of this micro-instruction (figure 4-7) is to test only the bits in the designated register that are referenced by the 1 bits in the mask contained in the micro-instruction, to ignore all others, and to perform the action as specified by the variants.

The possible actions which may be specified by the 3-bit variant field are as follows.

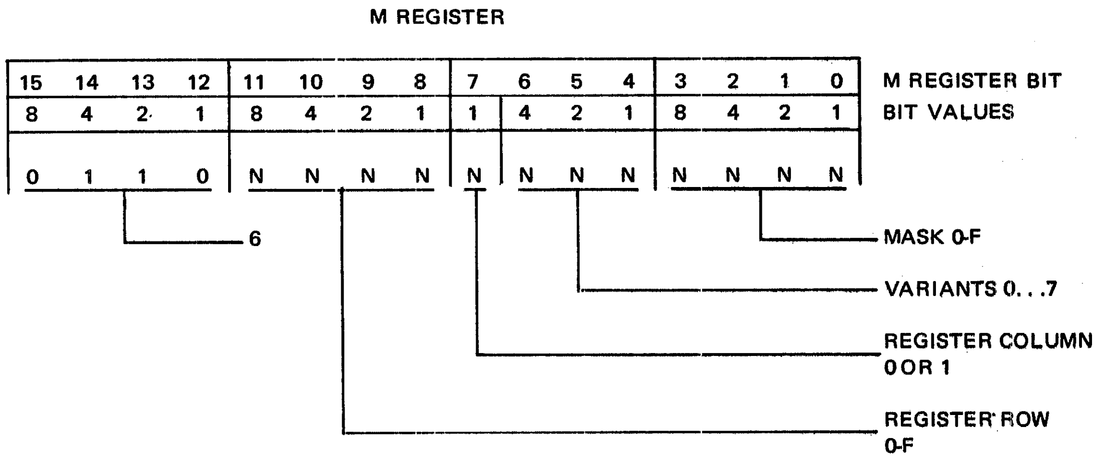


Figure 4-7. Skip When (6NNN)

Variant	Operation
000	If any of the referenced bits are 1, skip the next micro-instruction.
001	If all the referenced bits are 1, skip the next micro-instruction.
010	If the register is EQUAL to the mask, skip the next micro-instruction.
011	If all the referenced bits are 1, skip the next micro-instruction and CLEAR the referenced bits to zero.
100	If any of the referenced bits are 1, do not skip the next micro-instruction.
101	If all the referenced bits are 1, do not skip the next micro-instruction.
110	If the register is EQUAL to the mask, do not skip the next micro-instruction.
111	If all the referenced bits are 1, do not skip the next micro-instruction; and clear the referenced bits to zero.

NOTE

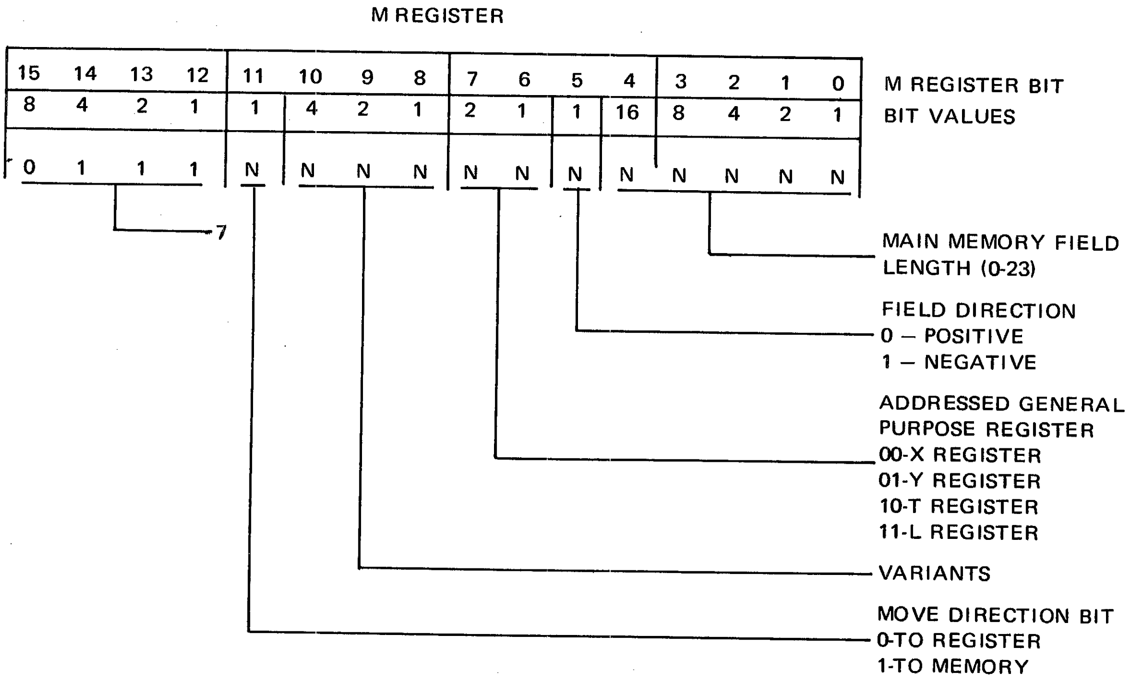
If the mask equals 0000,the (any) result is false. The skip is made if the variant is a 001 or 011 but is not made for 101 and 111.

BICN, FLCN, XYCN, XYST, and INCN are excluded as operand registers when the variant is equal to 011 or 111.

Only the 4-bit registers shown in columns 0 and 1 in table 4-1 may be used as the addressed register on which this micro-instruction is to act.

READ/WRITE MEMORY (7NNN).

The function of this micro-instruction (figure 4-8) is to move the contents of the addressed register to the main memory location that is specified in FA (field address register) or move the contents of the main memory location specified by FA to the addressed register, depending on the direction bit in the instruction.



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	1	4	2	1	2	1	1	16	8	4	2	1	BIT VALUES
0	1	1	1	N	N	N	N	N	N	N	N	N	N	N	N	

Figure 4-8. Read/Write Memory (7NNN)

If the value specified in the main memory field length is less than 24, the data from memory is right justified into the addressed register with left-most zero bits supplied while data from the register is truncated from the left when placed into memory. Values above 24 in the field length are reserved. The contents of the source is unchanged.

If the value of the main memory field length as given in the micro-instruction is zero, the value in the CPL portion of the C (control) register is utilized instead.

The possible variant conditions are as follows: (FA = field address register and FL = field length register.)

Variant	Condition
000	No incrementation
001	Binarily increment FA (FA↑)
010	Binarily increment FL (FL↑)
011	Binarily increment FA and decrement FL (FA↑-FL↓)
100	Binarily decrement FA and increment FL (FA↓-FL↑)
101	Binarily decrement FA (FA↓)
110	Binarily decrement FL (FL↓)
111	Binarily decrement FA and FL (FA↓-FL↓)

MOVE 8-BIT LITERAL (8NNN).

The function of this micro-instruction (figure 4-9) is to move the 8-bit literal contained in the micro-instruction to the destination register. If the move is between registers of unequal lengths, the data is right justified with left zero bits supplied. Only registers X, Y, T, L, A, BR, LR, FA, FB, FL, TAS, and CP can be specified. The register row number is assumed to be two. MSM can be specified as a destination in TAPE mode only.

MOVE 24-BIT LITERAL (9NNN).

The function of this micro-instruction (figure 4-10) is to move the 24-bit literal given in the micro-instruction to the destination register. If the move is between registers of unequal lengths, the literal is truncated from the left. The least-significant 16 bits of the literal to be moved must be contained in the next word of memory following the Move 24-Bit Literal micro-instruction.

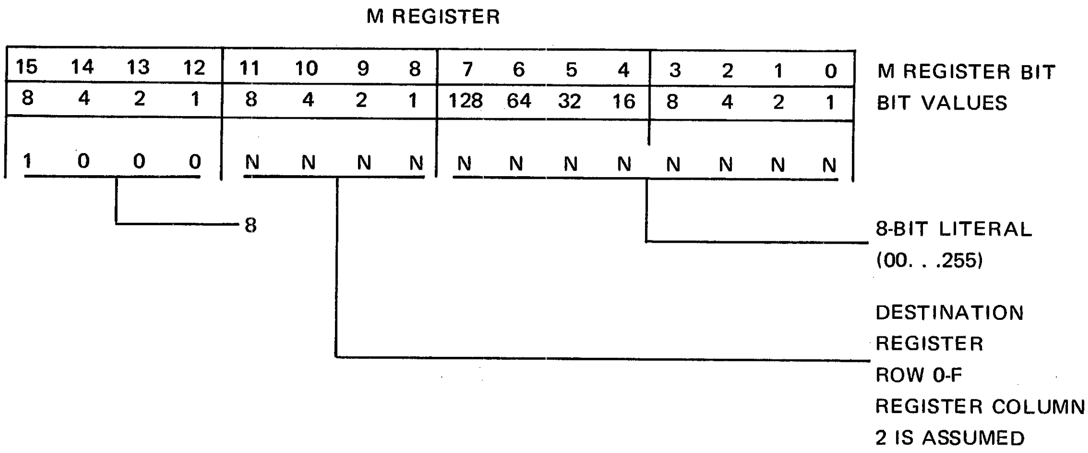


Figure 4-9. Move 8-Bit Literal (8NNN)

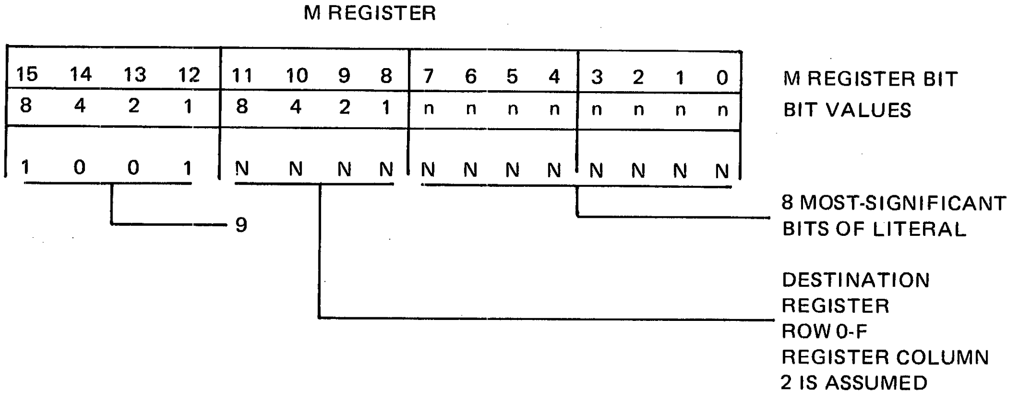


Figure 4-10. Move 24-Bit Literal (9NNN)

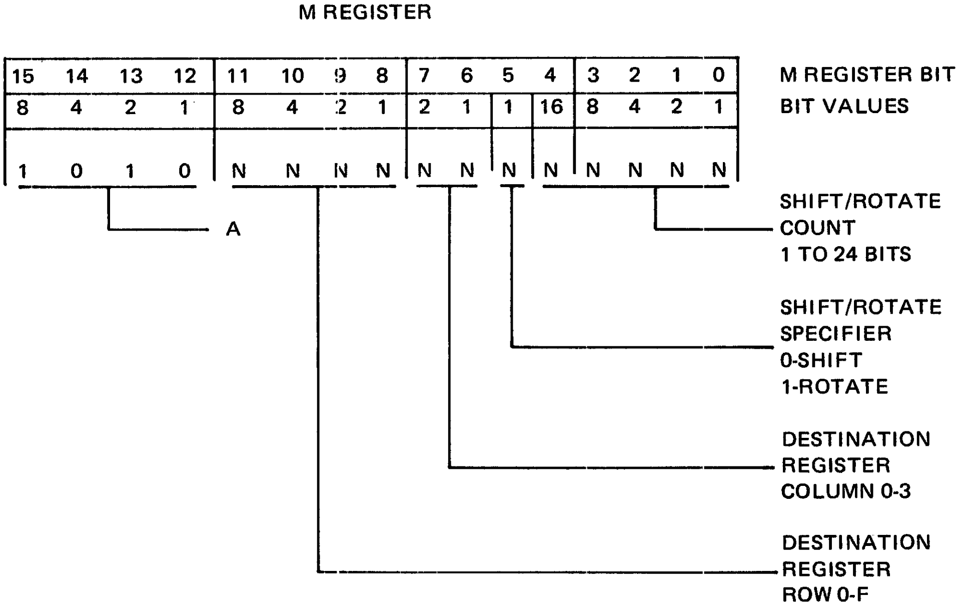
Only the registers X, Y, T, L, A, BR, LR, FA, FB, FL, TAS, and CP can be specified. The register row number is assumed to be two. MSM (only permitted on B 1726) can be specified as a destination register in the TAPE mode only.

NOTE

The least-significant 16 bits of the literal are found automatically by binarily incrementing the address (A) register by 16 (10000).

SHIFT OR ROTATE T REGISTER LEFT (ANNN).
The function of this micro-instruction (figure 4-11) is to shift or rotate the T register left by the number of bits specified and then move the 24-bit result to the destination register. If the move is between registers of unequal lengths, the data is right justified with data truncated from the left. For the shift operation, zero fill on the right and truncation on the left occur. The contents of the source register is unchanged unless it is also the designated destination.

M REGISTER



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	2	1	1	16	8	4	2	1	BIT VALUES
1	0	1	0	N	N	N	N	N	N	N	N	N	N	N	N	

A
SHIFT/ROTATE COUNT 1 TO 24 BITS
SHIFT/ROTATE SPECIFIER 0-SHIFT 1-ROTATE
DESTINATION REGISTER COLUMN 0-3
DESTINATION REGISTER ROW 0-F

Figure 4-11. Shift or Rotate T Register Left (ANNN)

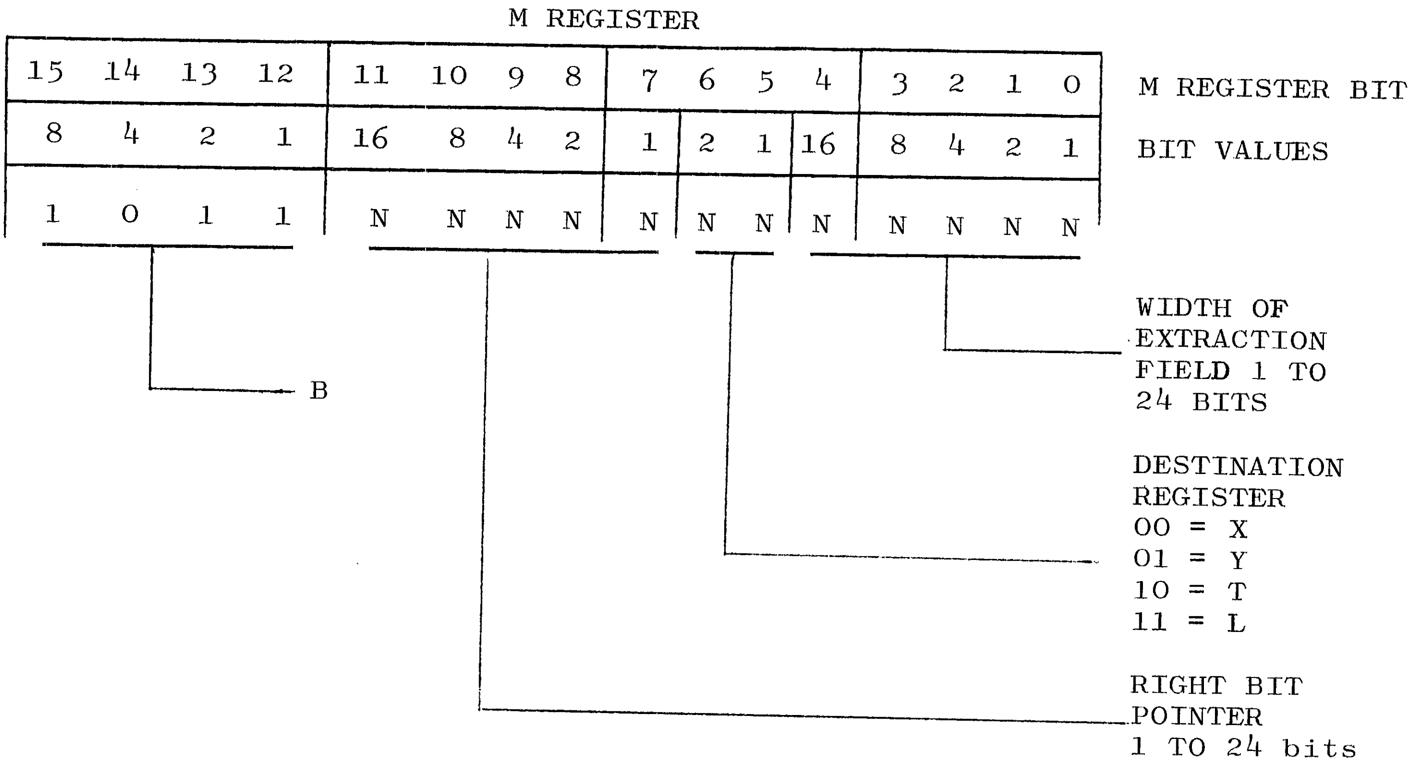
If the value of the shift/rotate count as given in the instruction is zero, the value given in CPL is used.

When M is used as a destination register, the operation is changed to a Bit-OR which modifies the next micro-instruction to be executed but does not modify the micro-instruction as stored in control memory.

BICN, FLCN, XYCN, XYST, INCN, and MSM are excluded as destination registers.

EXTRACT FROM T REGISTER (BNNN).
The function of this micro-instruction (figure 4-12) is to rotate the T-register left by the number of bits specified and then extract the

number of bits specified and move the results to the destination register. If the extract bit count is less then 24, the data is right justified with left zero bits supplied. The contents of the T register is unchanged unless it is also the designated destination. A rotate value of 24 is equivalent to zero.



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	16	8	4	2	1	2	1	16	8	4	2	1	BIT VALUES
1	0	1	1	N	N	N	N	N	N	N	N	N	N	N	N	

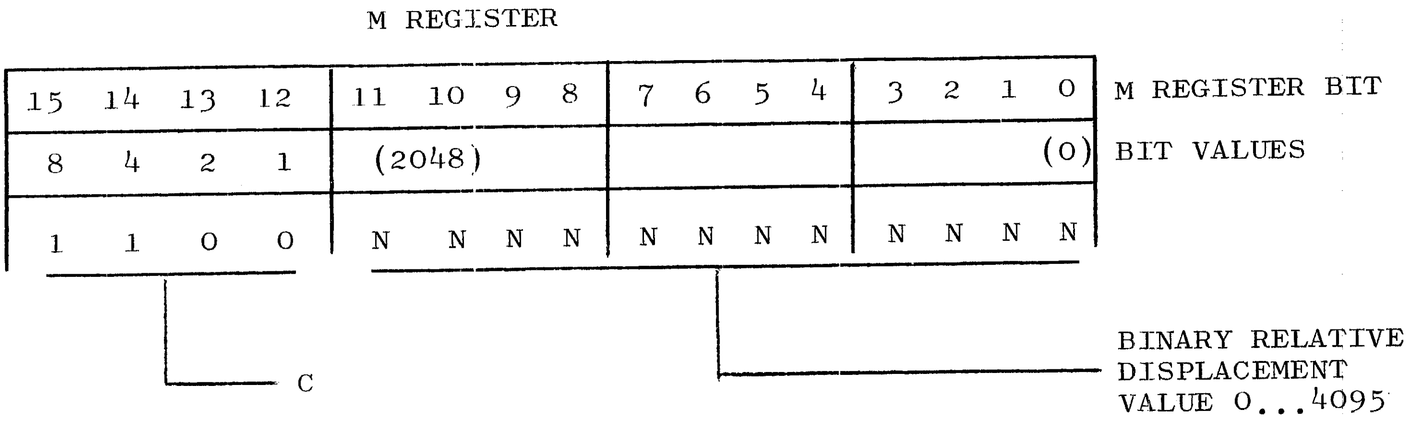
Figure 4-12. Extract from T register (BNNN)

If the right bit pointer value is equivalent to zero, the value expressed in CPL is used instead.

BRANCH.

BRANCH FORWARD (CNNN).
The function of this micro-instruction (figure 4-13) is to fetch the next micro-instruction from the location obtained by binarily adding the displacement value given in the micro-instruction to the address of the next in-line micro-instruction. The displacement value indicates the number of 16-bit words.

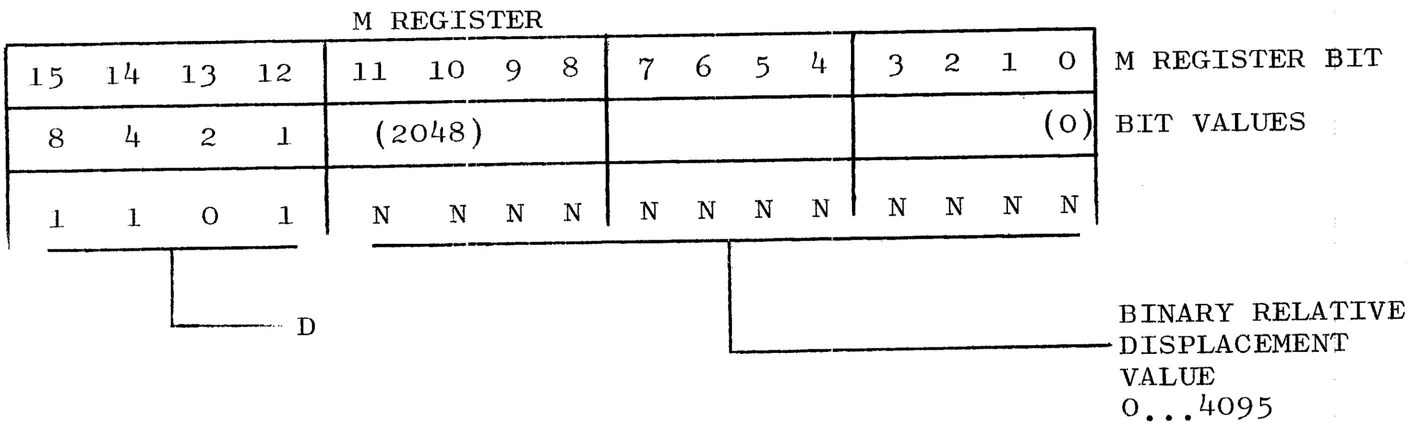
BRANCH BACKWARD (DNNN).
The function of this micro-instruction (figure 4-14) is to fetch the next micro-instruction from the location obtained by binarily subtracting the displacement value given in the micro-instruction from the address of the next in-line micro-instruction. The displacement value indicates the number of 16-bit words.



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	(2048)											(0)	BIT VALUES
1	1	0	0	N	N	N	N	N	N	N	N	N	N	N	N	

Figure 4-13. Branch Forward (CNNN)



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	(2048)											(0)	BIT VALUES
1	1	0	1	N	N	N	N	N	N	N	N	N	N	N	N	

Figure 4-14. Branch Backward (DNNN)

NOTE

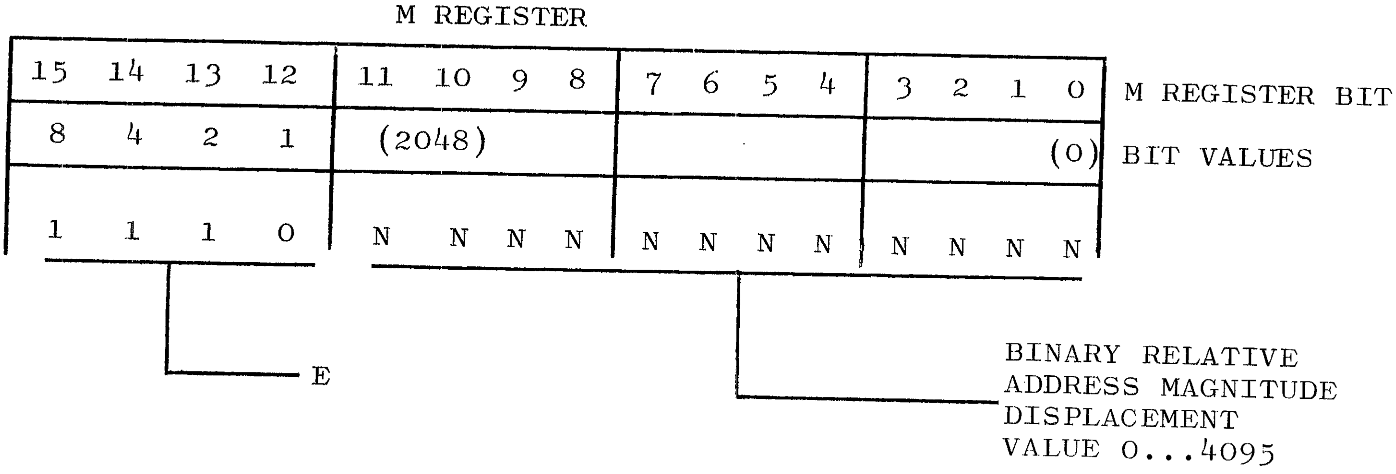
For both of the above micro-instructions the address of next in-line micro-instructions is determined by hardware by binarily incrementing the address (A) register by 16 (10000).

CALL.

CALL FORWARD (ENNN).
The function of this micro-instruction is to push the address of the next in-line instruction into the ASTACK and then Fetch the next micro-instruction from the location obtained by binarily adding the displacement value given in the micro-instruction to the address of the next in-line micro-instruction. The displacement value indicates the number of 16-bit words.

NOTE

When the A address is stored in the A stack it is multiplied by 16 and stored as a bit address.



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	(2048)											(0)	BIT VALUES
1	1	1	0	N	N	N	N	N	N	N	N	N	N	N	N	

E
BINARY RELATIVE ADDRESS MAGNITUDE DISPLACEMENT VALUE 0...4095

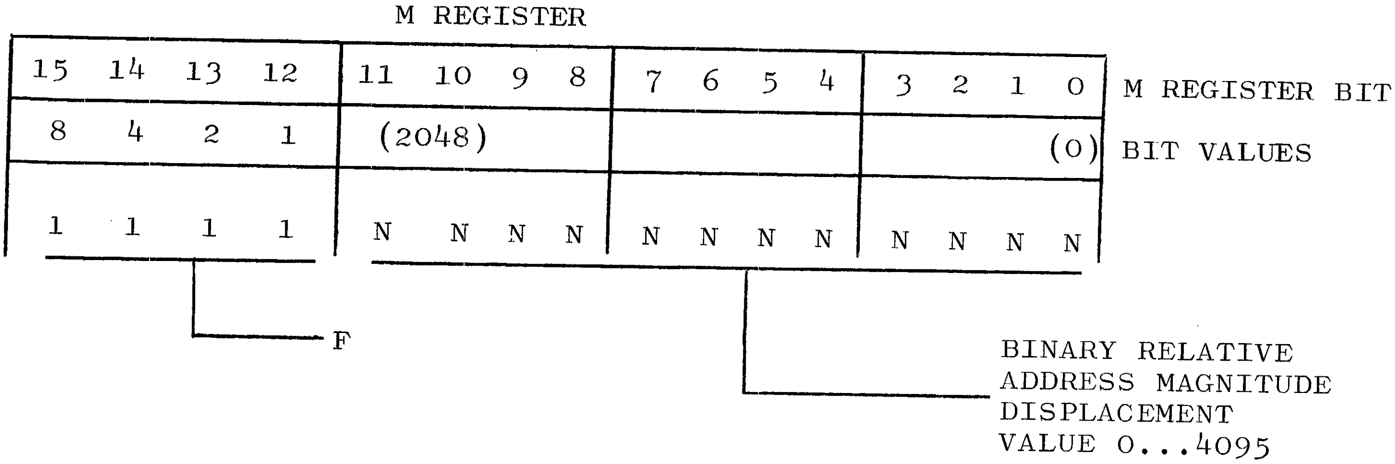
Figure 4-14A. Call Forward (ENNN)

CALL REVERSE (FNNN).

The function of this micro-instruction is to push the address of the next in-line instruction into the ASTACK and then fetch the next micro-instruction from the location obtained by binarily subtracting the displacement value given in the micro-instruction from the address of the next in-line micro-instruction. The displacement value indicates the number of 16-bit words.

NOTE

When the A address is stored in the A stack, it is multiplied by 16 and stored as a bit address.



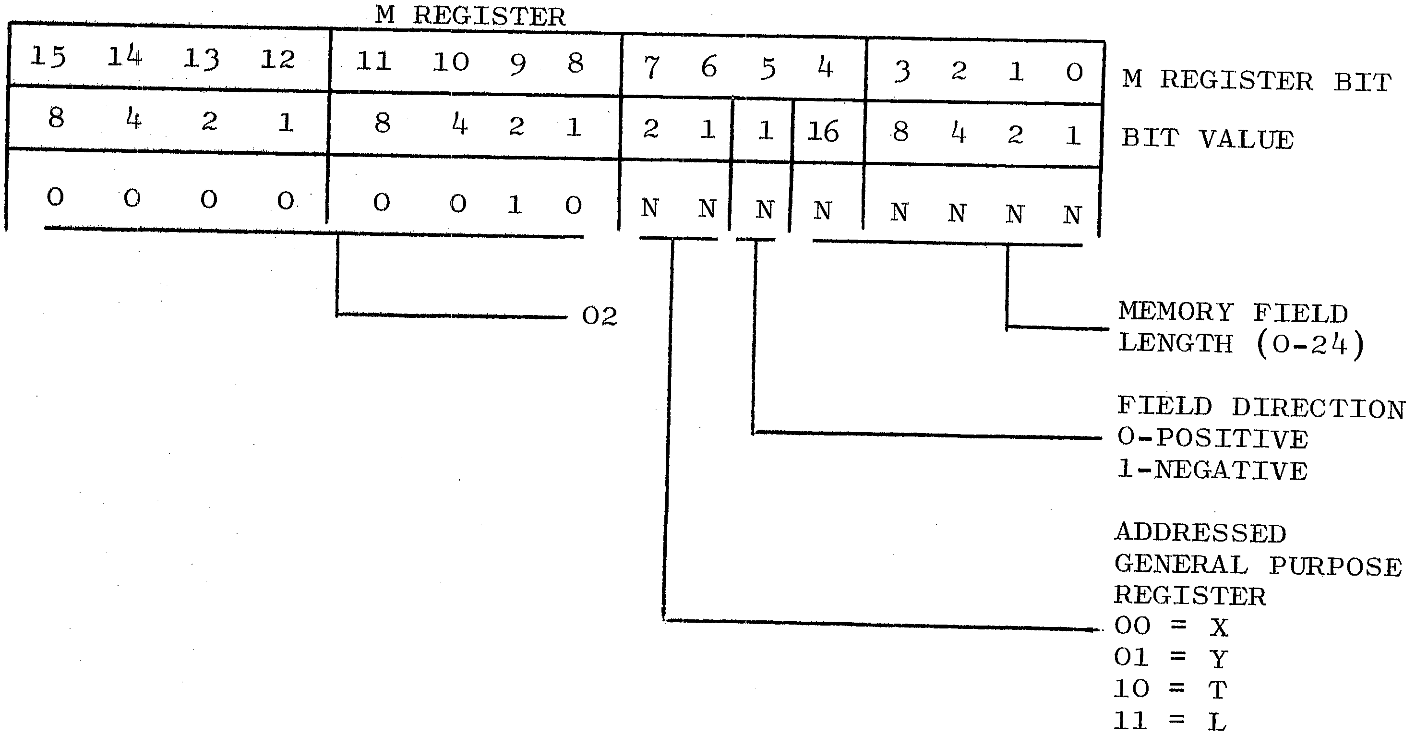
M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	(2048)											(0)	BIT VALUES
1	1	1	1	N	N	N	N	N	N	N	N	N	N	N	N	

F
BINARY RELATIVE ADDRESS MAGNITUDE DISPLACEMENT VALUE 0...4095

Figure 4-14B. Call Reverse (FNNN)

SWAP MEMORY (02NN).
The function of this micro-instruction (figure 4-15) is to swap up to 24 bits of data from main memory with the data in the specified register. If the value of the main memory field specified is less than 24 bits, the data from memory is right justified into the specified register with left zero bits supplied while the data from the register is truncated from the left as it is placed into memory.



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	2	1	1	16	8	4	2	1	BIT VALUE
0	0	0	0	0	0	1	0	N	N	N	N	N	N	N	N	

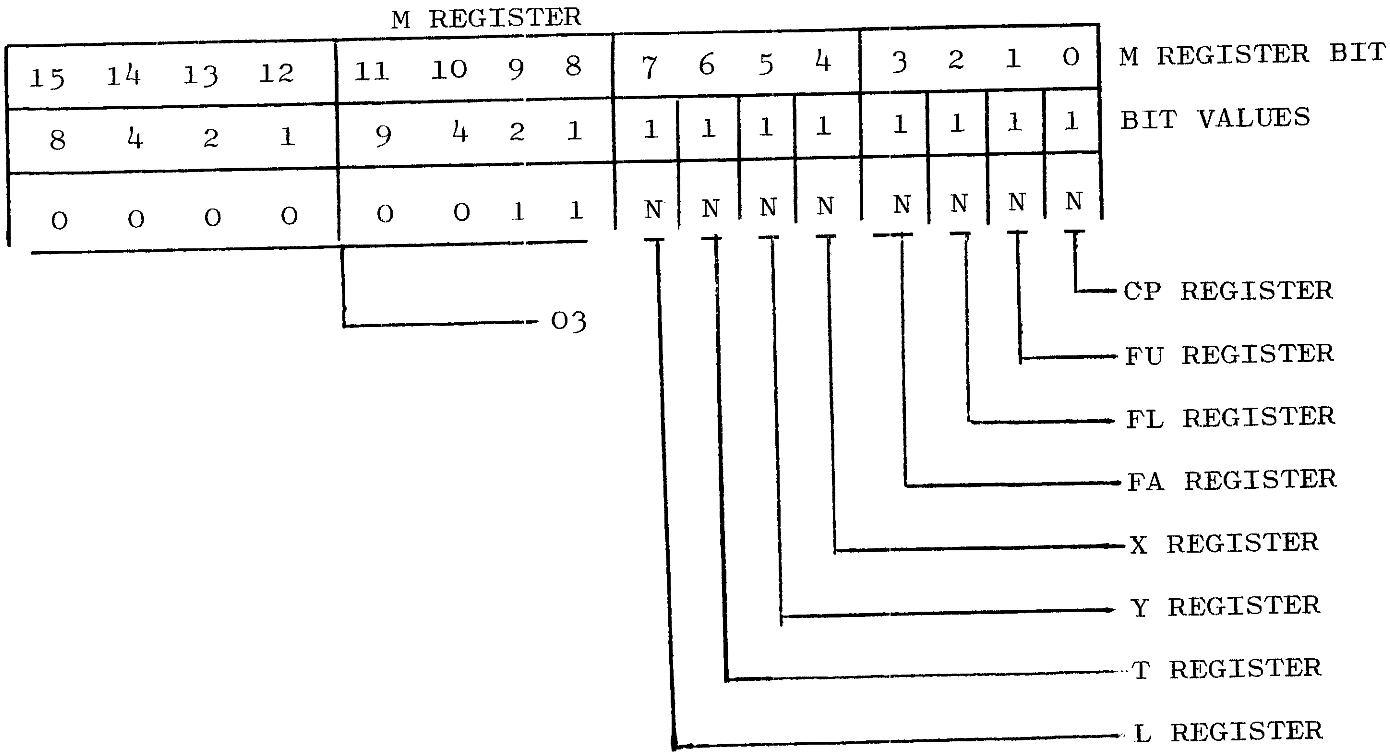
Figure 4-15. Swap Memory (02NN)

The field address register (FA) contains the address of the main memory location affected while the field direction sign and field length are given in the instruction. If the value of the main memory field length given in the instruction is zero, the value contained in CPL is used.

NOTE
This micro-instruction is available only on the B 1726 System.

CLEAR REGISTER (03NN).
The function of this micro-instruction (figure 4-16) is to clear the designated register to zero if its respective flag bit is set. Registers not designated are unaffected.

NOTE
This micro-instruction is available only on the B 1726 System.



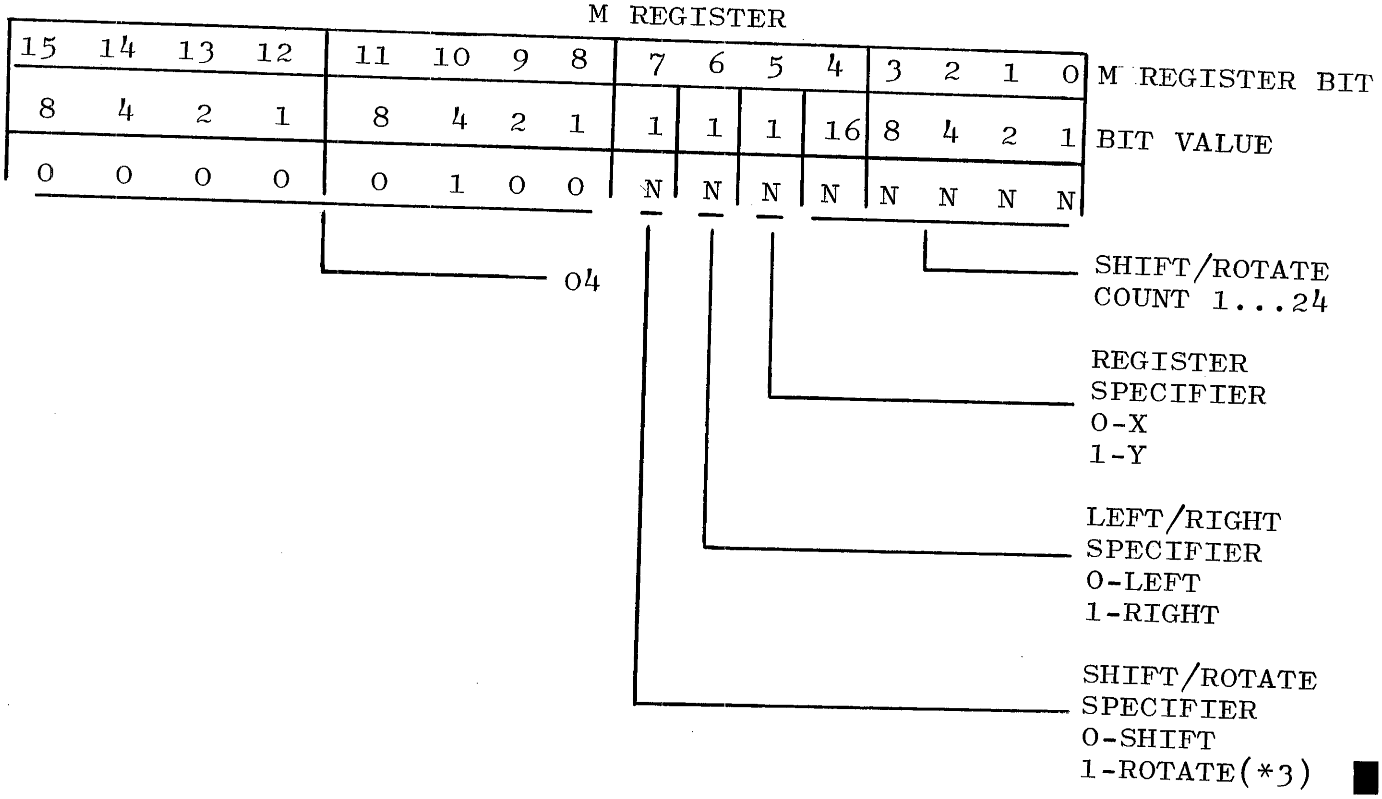
M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	9	4	2	1	1	1	1	1	1	1	1	1	BIT VALUES
0	0	0	0	0	0	1	1	N	N	N	N	N	N	N	N	

Figure 4-16. Clear Register (03NN)

SHIFT/ROTATE X OR Y (04NN). The function of this micro-instruction (figure 4-17) is to shift or rotate the X or Y register either left or right by the number of bits specified by the shift/rotate count portion of the micro-instruction. If the value of the shift/rotate count given in the micro-instruction is zero, the shift or rotate is determined by CPU as follows.

CPU	Shift/Rotate Count
00	1 bit
01	4 bits
10	Undefined
11	8 bits (not available on B 1710 Systems)



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	1	1	1	16	8	4	2	1	BIT VALUE
0	0	0	0	0	1	0	0	N	N	N	N	N	N	N	N	

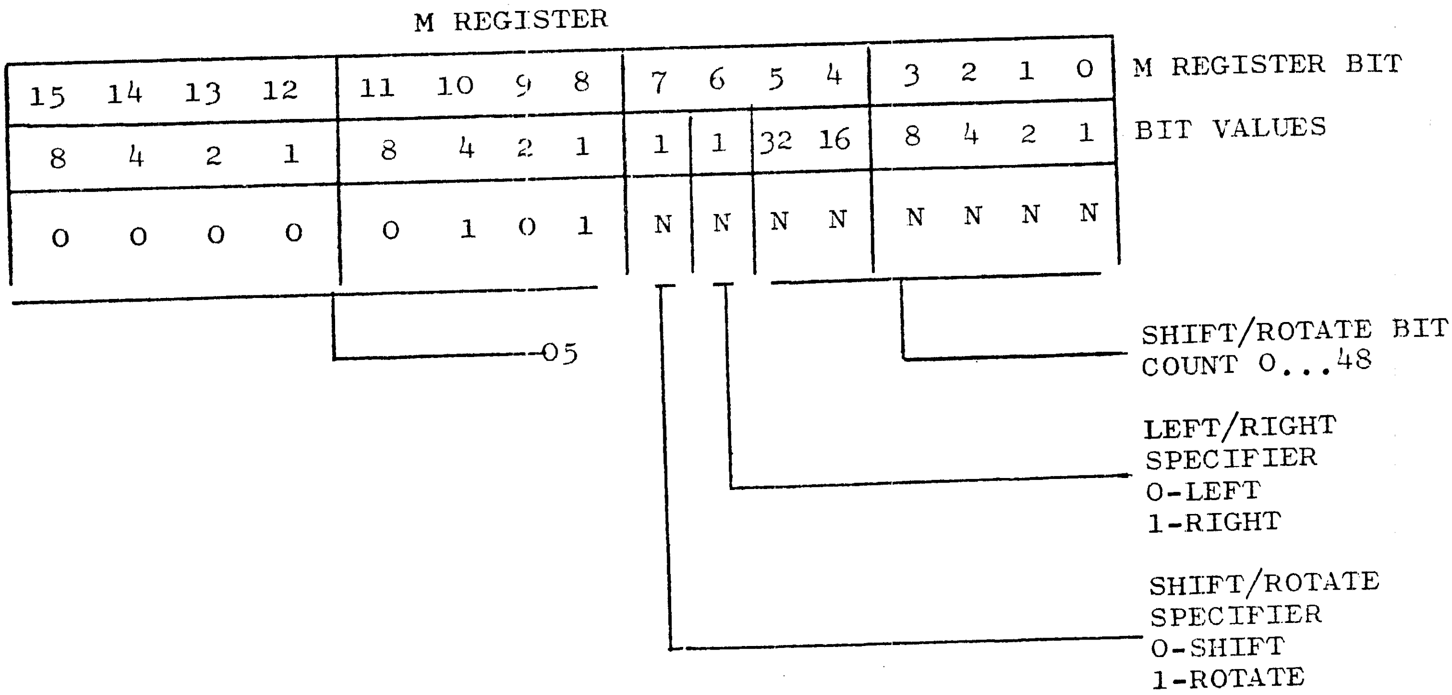
Figure 4-17. Shift/Rotate X or Y (04NN)

On the shift left the most-significant bits are truncated and zero bits occur on the right. On the shift right the least-significant bits are truncated and zero bits occur on the left.

SHIFT/ROTATE X AND Y (05NN).
The function of this micro-instruction(*3) (figure 4-18) is to concatenate the X and Y registers to become one 48-bit register with the X register being the left-most or most-significant. The combined registers are either shifted or rotated left or right by the number of bits specified. If the shift/rotate count is zero, the amount to be shifted or rotated is determined by the contents of CPU as follows:

CPU	Shift/Rotate(*3)
00	1 bit
01	4 bits
10	Undefined
11	8 bits (not available on B 1710 Systems)

*3 B 1726's allow SHIFT/ROTATE
B 1710's allow only SHIFT



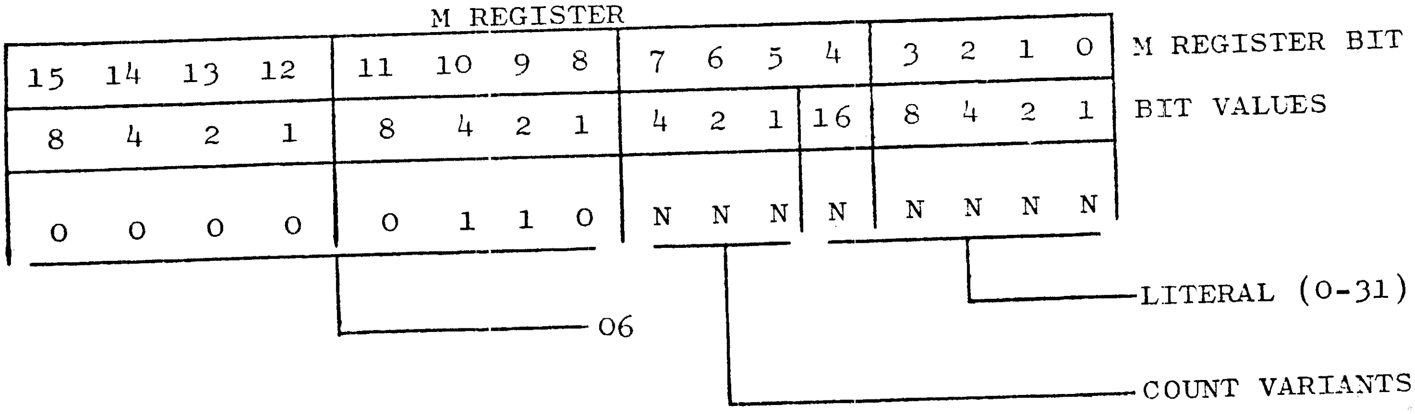
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	1	1	32	16	8	4	2	1	BIT VALUES
0	0	0	0	0	1	0	1	N	N	N	N	N	N	N	N	

M REGISTER
05
SHIFT/ROTATE BIT COUNT 0...48
LEFT/RIGHT SPECIFIER 0-LEFT 1-RIGHT
SHIFT/ROTATE SPECIFIER 0-SHIFT 1-ROTATE

Figure 4-18. Shift/Rotate X and Y (05NN)

If the micro-instruction indicates a shift left operation, zero fills on the right and truncation on the left occur. For the shift right operation, zero fills on the left and truncation on the right occur.

COUNT FA/FL (06NN).
The function of this micro-instruction (figure 4-19), depending on the variant field, is to binarily increment or decrement the designated register(s) by the value of the literal contained in the micro-instruction or by the value of CPL if the literal is zero. The field address (FA) register and the field length (FL) register are the only two registers affected by this micro-instruction. Results of values of greater than 24 for FL are undefined.



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	4	2	1	16	8	4	2	1	BIT VALUES
0	0	0	0	0	1	1	0	N	N	N	N	N	N	N	N	

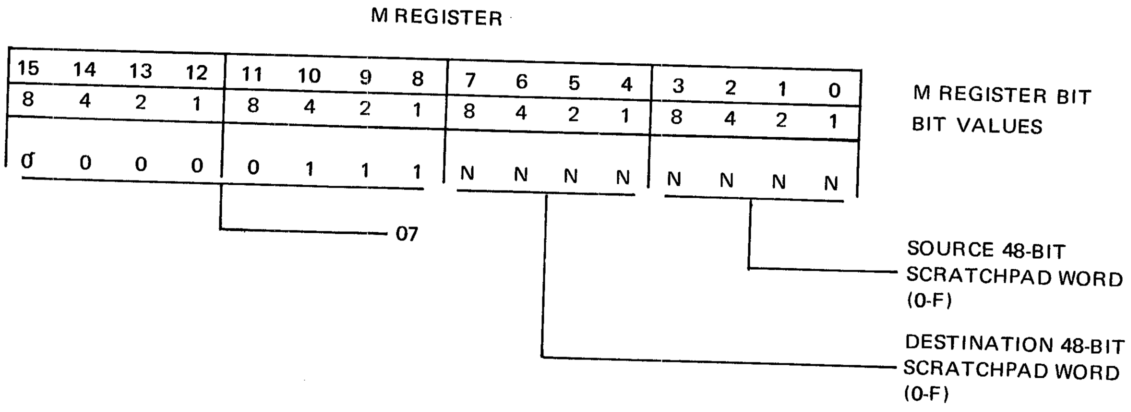
Figure 4-19. Count FA/FL (06NN)

Neither overflow nor underflow of FA is detected. The value of FA may exceed its maximum value and wraparound. Overflow of FL is also not detected. The value of FL may exceed its maximum value and wraparound. Underflow of FL is detected and does not wraparound, in which case a binary value of zero is left in FL.

The possible variant conditions are as follows.

Variant	Condition
000	No operation
001	Binarily increment FA (FA↑)
010	Binarily increment FL (FL↑)
011	Binarily increment FA and decrement FL (FA↑-FL↓)
100	Binarily decrement FA and increment FL (FA↓-FL↑)
101	Binarily decrement FA (FA↓)
110	Binarily decrement FL (FL↓)
111	Binarily decrement FA and FL (FA↓-FL↓)

EXCHANGE DOUBLEPAD WORD (07NN).
The function of this micro-instruction (figure 4-20) is to move the 48 bits of the field (F) definition register to a holding register (refer to note) and then move 48 bits from a source register in scratchpad to the F register. After this is accomplished the 48 bits are taken from the holding register and placed into the specified destination register in scratchpad. A doublepad word is equal to 48 bits of scratchpad memory.


M REGISTER
M REGISTER BIT
BIT VALUES
07
SOURCE 48-BIT SCRATCHPAD WORD (0-F)
DESTINATION 48-BIT SCRATCHPAD WORD (0-F)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
0	0	0	0	0	1	1	1	N	N	N	N	N	N	N	N

Figure 4-20. Exchange Doublepad Word (07NN)

NOTE
The holding register is imbedded in the hardware. It is not specified within the micro-instruction and is not accessible to the micro-programmer.

SCRATCHPAD RELATE FA (08NN).
This micro-instruction (figure 4-21), depending on the sign of operation, binarily adds or subtracts the contents of the left half of the addressed portion of scratchpad memory (scratchpad locations S00A

through S15A) to/from the contents of the 24-bit field address (FA) register, and places the results in FA.

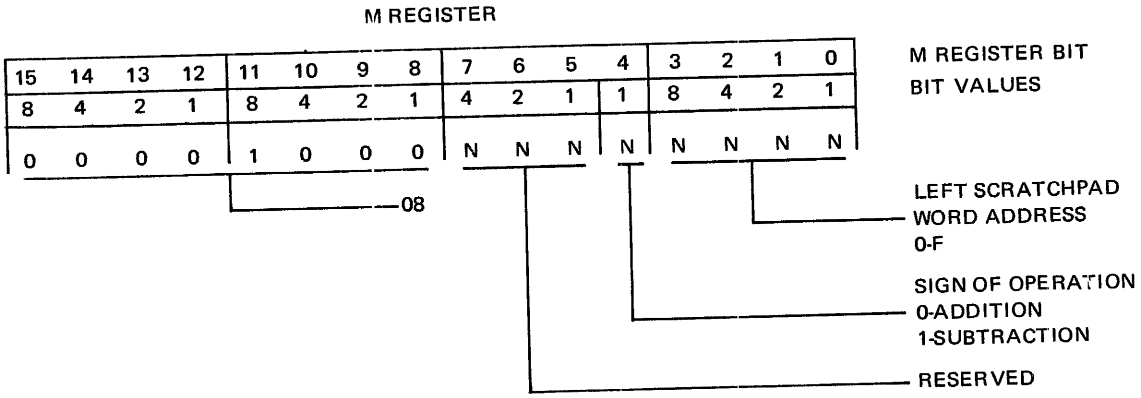


Figure 4-21. Scratchpad Relate FA (08NN)

MONITOR (09NN).
This micro-instruction (figure 4-22) skips to the next sequential instruction. Execution of this instruction brings a level to the backplane. The eight, least-significant bits (0 thru 7) of the M Register may be monitored concurrently. Also, the two, least-significant bits of the micro-instruction are decdded; and one of four levels is brought to the backplane on the B 1720 series.

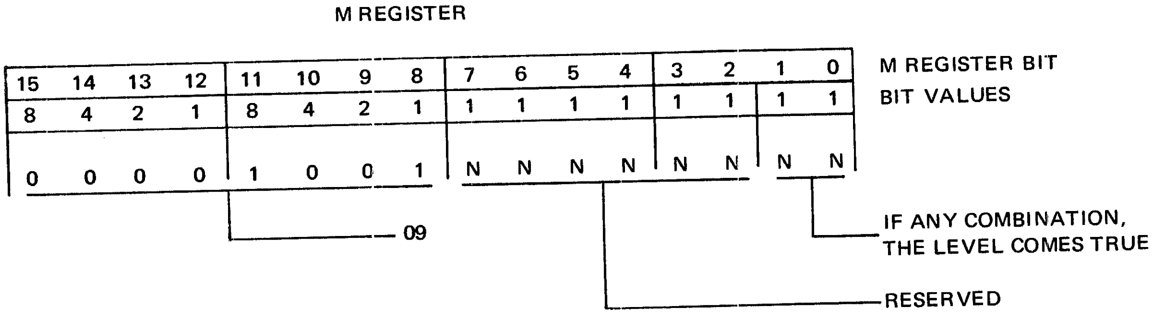
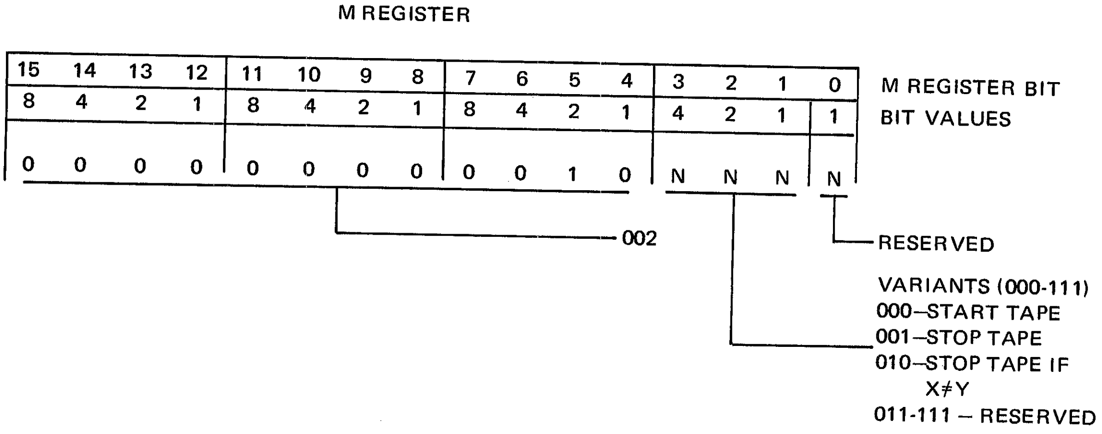


Figure 4-22. Monitor (09NN)

CASSETTE CONTROL (002N).
The function of this micro-instruction (figure 4-23) is to control the cassette tape movement.

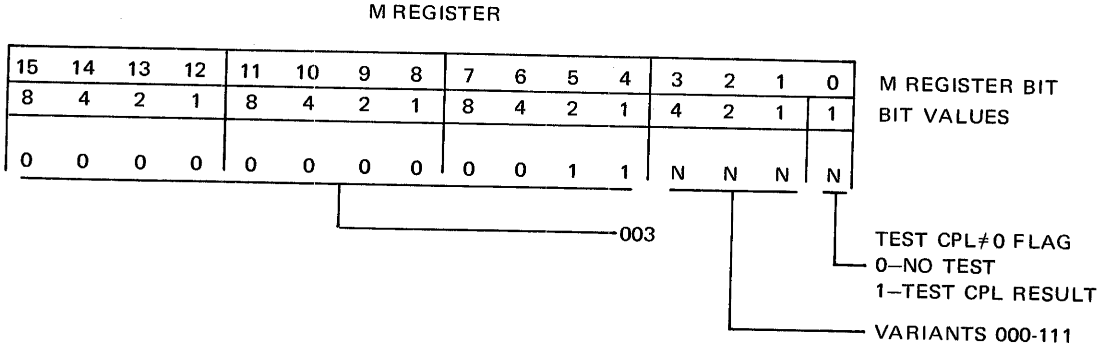


15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
8	4	2	1	8	4	2	1	8	4	2	1	4	2	1	1
0	0	0	0	0	0	0	0	0	0	1	0	N	N	N	N

Figure 4-23. Cassette Control (002N)

All tape stop variants cause the cassette tape to halt in the next available tape gap, not immediately. If the processor is in the TAPE mode and a stop tape cassette condition occurs, the processor also halts.

BIAS (003N).
The function of this micro-instruction (figure 4-24) is to set the CPU register to the value of 1, 3, or 0 depending on whether the value of the field unit (FU) portion of the F register is equal to 4, 8, or any other value 0-F, respectively.



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
8	4	2	1	8	4	2	1	8	4	2	1	4	2	1	1
0	0	0	0	0	0	0	0	0	0	1	1	N	N	N	N

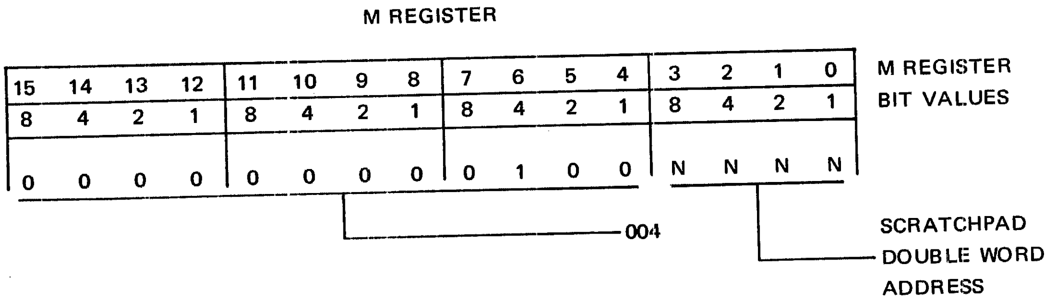
Figure 4-24. Bias (003N)

Depending on the variants the value of CPL is set to the value denoted or to the smallest of the values listed below.

Variant	Values
000	FU
001	24 or FL
010	24 or SFL
011	24 or FL or SFL
100	CPL
101	24 or CPL or FL
110	CPL
111	CPL

When the variant field is equal to 010, the value of CPU is determined by SFU instead of FU. SFU is contained in the first four bits of scratchpad word S00B. If the test flag is equal to 1 and CPL is not equal to zero, the next micro-instruction is skipped.

STORE F IN DOUBLEPAD WORD (004N).
The function of this micro-instruction (figure 4-25) is to store the contents of the F register (FA portion = 24 bits and FB portion = 24 bits) in the designated scratchpad word. FA goes to the left half (A) of the scratchpad word and FB to the right half (B) of the scratchpad word.



M REGISTER

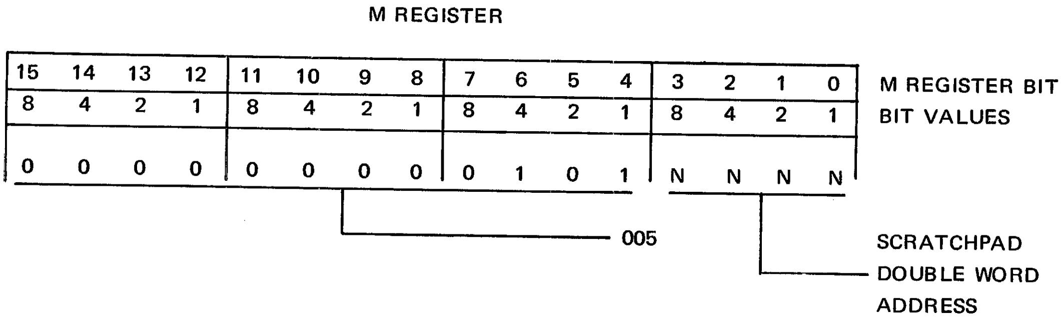
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1	BIT VALUES
0	0	0	0	0	0	0	0	0	1	0	0	N	N	N	N	

Figure 4-25. Store F into Doublepad Word (004N)

NOTE
This micro-instruction is available only on the B 1726 System.

LOAD F FROM DOUBLEPAD WORD (005N).
The function of this micro-instruction (figure 4-26) is to move all 48 bits of the designated scratchpad word to the F register. The left half (A) of the doublepad word is placed into the FA portion of the F register, and the right half (B) of the doublepad word is placed in the

FB portion of the F register.



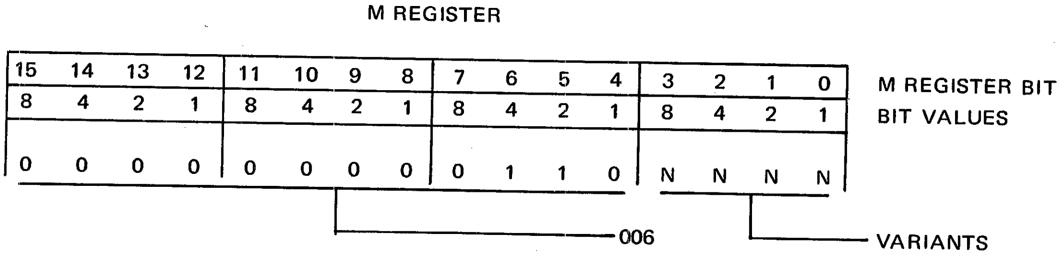
M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1	BIT VALUES
0	0	0	0	0	0	0	0	0	1	0	1	N	N	N	N	

Figure 4-26. Load F from Doublepad Word (005N)

NOTE
This micro-instruction is available only on the B 1726 System.

SET CYF (006N).
The function of this micro-instruction (figure 4-27) is to set CYF (carry flip-flop) as specified by the variant.



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1	BIT VALUES
0	0	0	0	0	0	0	0	0	1	1	0	N	N	N	N	

Figure 4-27. Set CYF (006N)

The available variants are represented below; all others are reserved.

Variant	Action
0001	Set CYF to 0.
0010	Set CYF to 1.
0100	Set CYF to CYL (carry total for sums).
1000	Set CYF to CYD (carry borrow from difference).

CYL is generated under control of the length in CPL, and CYF is an input to the arithmetic logic along with the X and Y register's.

HALT (0001).
The function of this micro-instruction (figure 4-28) is to halt the sequence of executable micro-instruction's.

M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1	BIT VALUES
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	

0001

Figure 4-28. Halt (0001)

The system halts with the next in-line micro-instruction in the M register and the address of the second in-line micro-instruction in the address (A) register's. Whatever register is pointed to by the console controls is displayed on the register display light's.

OVERLAY CONTROL MEMORY (0002).
The function of this micro-instruction (figure 4-29) is to overlay control memory from main memor'y. This is desired because of faster micro-instruction execution time in control memor'y.

M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1	BIT VALUES
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	

0002

Figure 4-29. Overlay Control Memory (0002)

The address for data from main memory is obtained from the field address (FA) register, and the address where data is to be placed into control memory is contained in the L registe'r. The length of the data overlay bits is taken from the FL register, and overlaying occurs until FL=0 or the maximum amount of control memory is exceede'd.

The execution of the micro-instructions proceeds as follows:

a. A is moved to TAS.

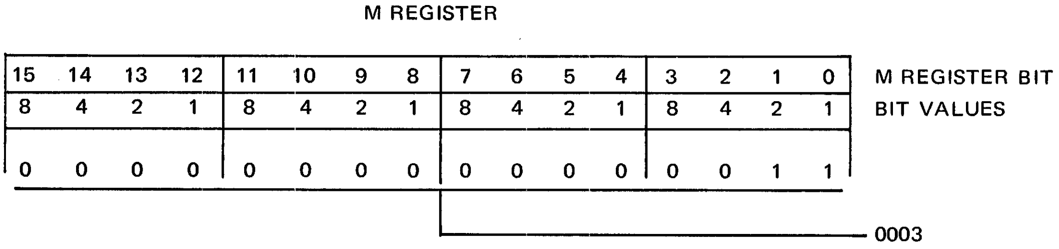
b. L is moved to A.

c. The first 16 bits are read from main memory and stored in control memory. Register FL is decremented by 16 bits, FA incremented by 16 bits, and A incremented by one word.

d. Step c is repeated until one of the two above mentioned ending conditions are met.

NOTE
This instruction is available only on the B 1726 System.

NORMALIZE X (0003).
The function of this micro-instruction (figure 4-30) is to shift the X register left while counting FL down until FL is equal to zero or the bit in X which is referenced by CPL is equal to one. Zero bits are shifted into the least-significant end of X during the shift operation.



M REGISTER

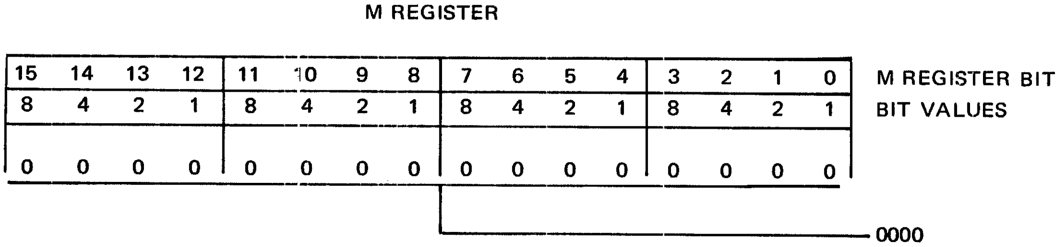
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1	BIT VALUES
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	

Figure 4-30. Normalize X (0003)

If CPL is equal to 1, the least-significant bits of X are referenced; if CPL is equal to 24, the most-significant bits of X are referenced.

NO OPERATION (0000).
The function of this micro-instruction (figure 4-31) is to reserve 16 bits of memory (the length of one micro-instruction) and to utilize one machine cycle.

When this micro-instruction is encountered, a fall-through to the next sequential micro-instruction occurs.



M REGISTER

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	M REGISTER BIT
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1	BIT VALUES
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figure 4-31. No Operation (0000)

SECTION 5

B 1726 I/O SUBSYSTEM

GENERAL.

The I/O subsystem on the B 1726 System is designed to enhance simultaneity of I/O operation. The MCP creates the various I/O descriptors and sends an interrupt message to the desired I/O controller. At the completion of the I/O event, either normal or abnormal, the I/O controller returns the result descriptor which contains descriptive information concerning the I/O operation just completed. Immediately following the result descriptor there may exist the address (link) of another I/O descriptor which is initiated by the MCP. This provides for chaining of I/O's.

I/O COMPONENTS.

The I/O subsystem has an ordered relationship among its components. From peripheral device to main memory the order is as follows. A peripheral device (disk cartridge, printer, etc.) interfaces and is controlled by an I/O controller which interfaces to the processor. Up to eight I/O controllers may interface with the processor which is connected to main memory. (See figure 5-1.)

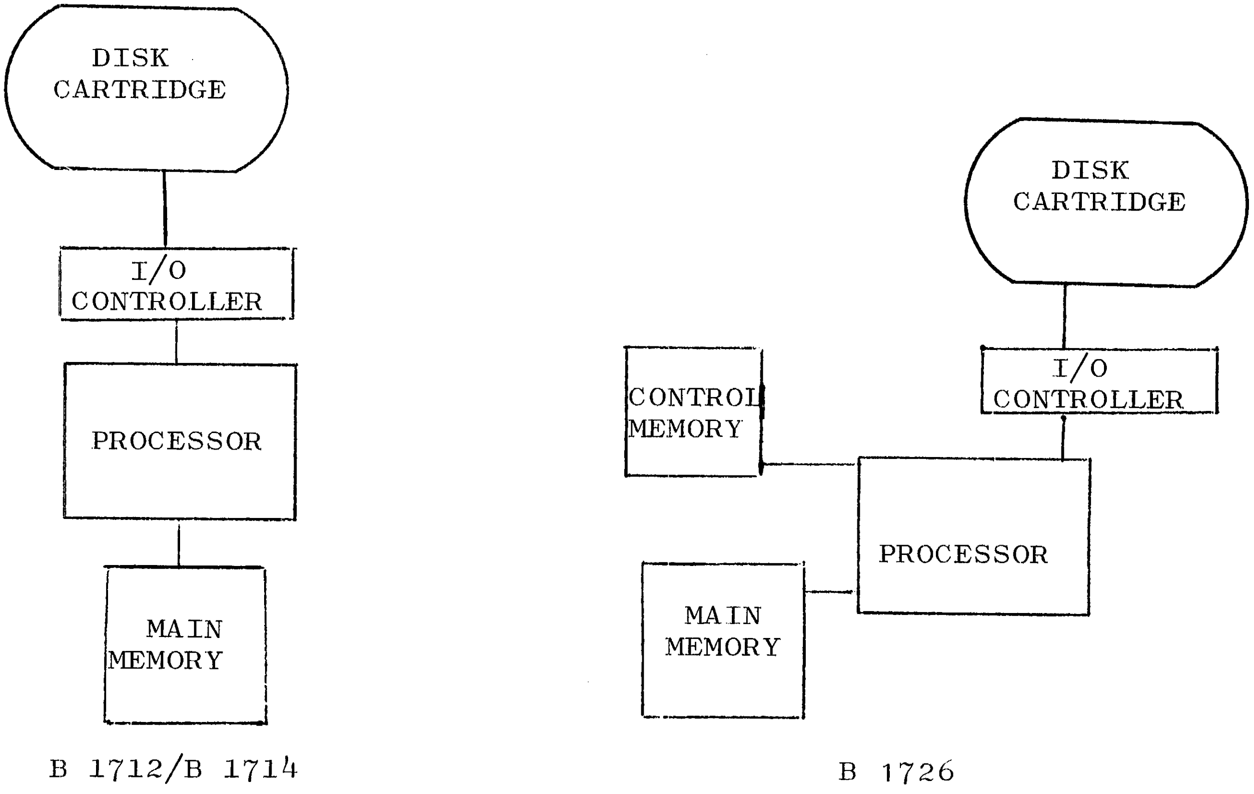


Figure 5-1. I/O Subsystem

I/O CONTROLLERS.

The peripheral I/O controllers are physically separate, logically independent devices. The peripheral side of each I/O controller is unique for each type of device.

Each I/O controller is a special-purpose device specifically designed to interface one type of peripheral to the system. In most cases an I/O controller interfaces to a single device; in a few cases a single I/O controller interfaces with two or more identical devices on a time-shared basis (i.e., serial disk drives or serial tape drives). I/O controllers have individual hardware buffers to aid in simultaneity of I/O operations.

I/O CONTROLLER FUNCTIONS.

The functions performed by an I/O controller are to accept an I/O descriptor from the processor and to determine if it is a valid I/O descriptor. In the case of an invalid I/O descriptor an appropriate result descriptor is returned. Otherwise, the controller signals the processor that it is now ready to receive data. Hardware translation between the peripheral data codes and the internal memory code is performed by the I/O controller (i.e., BCL card code to EBCDIC).

At the conclusion of an I/O operation the I/O controller returns a result descriptor to the processor.

SECTION 6

I/O DESCRIPTORS

GENERAL.

The purpose of input/output descriptors is to communicate control information to the various I/O controllers which perform the desired operation on the peripheral.

I/O DESCRIPTOR FORMAT.

Each I/O descriptor (figure 6-1) consists of a series of 24-bit fields. The number of these 24-bit fields varies with the peripheral to which the I/O descriptor is directed. These differences are explained in section 7.

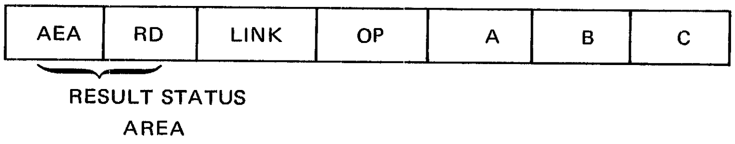


Figure 6-1. I/O Descriptor Format

The fields shown above are each 24 bits. The maximum size of any I/O descriptor is 168 bits; the minimum size is 96 bits. Fields A, B, and C may be omitted depending on the I/O descriptor initiated and the peripheral involved. The various fields are defined in the following paragraphs.

RESULT STATUS AREA.

The result status area (RSA) consists of two 24-bit fields: the actual ending address (AEA), which is the ending binary address + 1 of the last data bit affected by the operation; and a 24-bit result descriptor (RD), which indicates the status of the just terminated I/O operation. The RSA is returned by the I/O controller at the completion of each I/O operation.

LINK.

The LINK field contains an address which points to the beginning address of a result descriptor which is contained in the next desired I/O descriptor. Linking is further explained later in this section.

OP.

The operation (OP) field contains the desired I/O command and its variants.

A.

The A field contains the beginning binary address of the input/output main memory area.

B.
The B field contains the ending binary address + 1 of the input/output main memory area.

C.
The C field is used in conjunction with disk I/O's only and contains the absolute binary disk address.

I/O DESCRIPTOR CHAINING.

The ability to link or chain together I/O descriptors provides for maximum utilization of the processor and peripheral devices. Shown in figure 6-2 is a sample chain of I/O descriptors which are created by the MCP because a user program has requested three output print buffers.

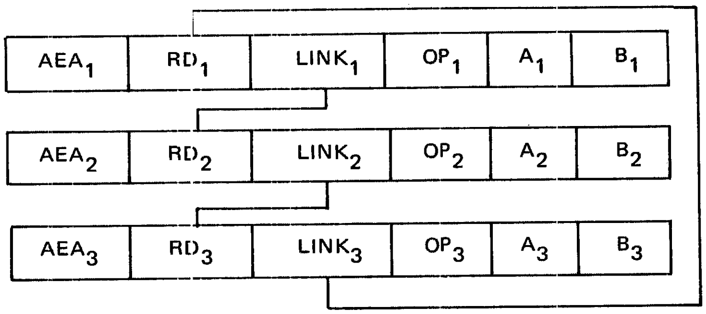


Figure 6-2. Sample Chain of I/O Descriptors

LINK-sub1 contains the address of RD-sub2, LINK-sub2 contains the address of RD-sub3, and LINK-sub3 contains the address of RD-sub1. This forms a continuous chain of I/O descriptors, one associated with each of the requested buffers.

The address provided to the I/O controller by the processor contains the address of RD-sub1. This action locks in the particular I/O controller and device to the proper chain of I/O descriptors. At the address provided, the I/O-complete bit of that result descriptor is examined; if it is a zero, the I/O descriptor is executed and a result descriptor is returned to location RD-sub1. If no exception condition has occurred, the I/O descriptor pointed to by LINK-sub2 is executed, and so on, thus providing a continuous chain of I/O descriptors. If the I/O-complete bit is set to one when examined, a pause of a specified number of milliseconds occurs and re-examination of the I/O complete-bit occurs. The I/O-complete continues to be examined until it becomes zero; in this manner execution of the I/O descriptors is under MCP control. The MCP can get ahead of the actual execution of the I/O descriptors; that is, the MCP may be releasing buffers faster than the I/O descriptors can be executed. The I/O controllers can never get ahead of the MCP because each I/O controller remains locked to each I/O descriptor until the I/O is complete. A stop I/O descriptor is provided by which any I/O controller may be stopped from executing an I/O descriptor. If an error condition occurs on any given I/O operation, that particular chain is broken and the MCP is notified.

This process is completely under control of the MCP. The MCP examines the I/O-complete bits and performs the above operation.

DISK FILE AND DISK PACK CHAINING OPERATION.
Since disk I/O descriptors all contain an absolute disk address, there is no danger of an operation being performed out of logical sequence as long as the controller does not attempt to execute a descriptor that is already complete. Also, since disk addresses may be in the chain for many different programs and files, the controller must not wait for any given descriptor to be released. On discovering an I/O descriptor that is already complete, the control must proceed to the next I/O descriptor immediately.

As a consequence of this mode of operation, the controller could conceivably be continually searching the same chain of I/O descriptors for one that is executable and therefore could be using all available memory accesses. For this reason, the test/pause I/O descriptor has been implemented.

In summary, once a disk file is opened and its descriptors are linked into the subsystem chain, they may or may not be executable depending on the setting of the I/O-complete bit in the result descriptor, which in turn is dependent on the read, write, and seek instructions in the object program.

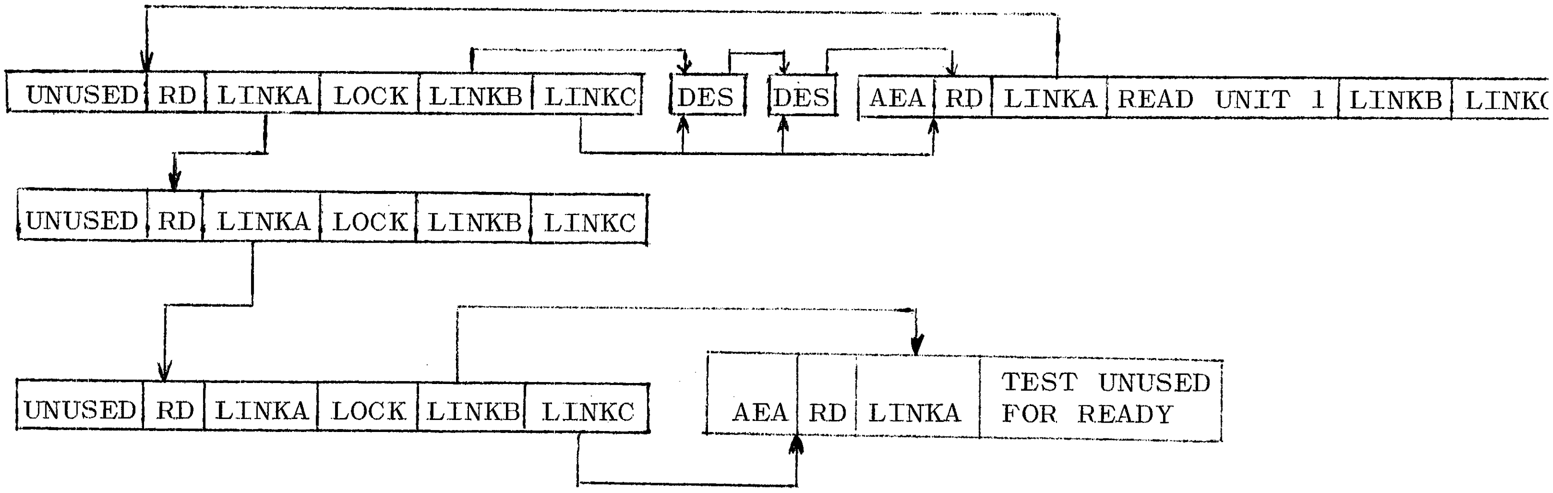
MAGNETIC TAPE OPERATION.
Because of the serial nature of magnetic tape operation, it is slightly more complex than that of disk. There is one chain of I/O descriptors for the entire tape subsystem, similar to the disk operation previously described. The descriptors must, however, be executed in logical sequence. To accomplish this, the lock descriptor has been implemented in the tape controls only.

The lock descriptor contains three link fields, as shown in figure 6-3.

There is one lock descriptor in the subsystem chain for each tape unit in the system. All I/O descriptors for that unit are linked, directly or indirectly, from the lock descriptor for the unit. LINK-A of the lock descriptor always contains the memory address of the lock descriptor for the next unit in the subsystem or the address of a pause descriptor. There is one pause descriptor for the entire subsystem. LINK-B of the lock descriptor always contains the memory address of the first, physical I/O descriptor for the unit. This field is initialized by the MCP when the file is opened. LINK-C contains the memory address of the logically next I/O descriptor to be executed on the unit. LINK-C is initialized by the MCP but is maintained by the magnetic tape controller.

STOP I/O DESCRIPTOR.
All I/O controllers are capable of receiving and recognizing a stop I/O descriptor. When this descriptor is received, the I/O controller stops operation and idles. The main use of this descriptor is to allow the MCP the ability to terminate a chaining operation.

The format for a stop I/O descriptor is shown in figure 6-4.



LOCK DESCRIPTOR

LINKA ALWAYS POINTS TO NEXT LOCK DESCRIPTOR
LINKB ALWAYS POINTS TO FIRST NON-LOCK DESCRIPTOR.
LINKC ALWAYS POINTS TO CURRENT NON-LOCK DESCRIPTOR.

Figure 6-3. Typical Magnetic Tape Chain

VVV	RRRRRRRRRRRRRRRRRRRRR
3 BITS	21 BITS

VVV = 111 - STOP I/O COMMAND
R = RESERVED

Figure 6-4. Stop I/O Descriptor

DISPATCH OF I/O DESCRIPTORS.
The Dispatch operation is used to send the address of an I/O descriptor to a device on an I/O port. Information is also received from the I/O port via this operation.

SECTION 7

INPUT/OUTPUT CONTROLLERS

GENERAL.

This section describes the I/O controllers which may be connected to the B 1700 series of computers. All input/output operations are initiated but not executed by the processor. The execution of any specified I/O operation is accomplished by the respective I/O controllers. The I/O operation may be executed simultaneously with a processor operation or other previously initiated I/O operations. The type of I/O operation is determined by an I/O descriptor that is transferred to the I/O controller. At the conclusion of an I/O operation, a result descriptor that specifies any exception conditions and other pertinent information is returned by the I/O controller. All I/O's are initiated by the Burroughs Master Control Program.

CARD READER CONTROLLER.

The card read I/O controller executes the card reader as initiated by the MCP. It reads data into ascending memory locations beginning with the location specified by the A address.

CARD READ DESCRIPTOR (EBCDIC).

This I/O descriptor controller (figure 7-1) is capable of translating the standard EBCDIC card code into the internal 8-bit EBCDIC code. When reading EBCDIC, 256 card-hole combinations are valid. All invalid combinations are translated to the ? character. The card is read into ascending memory locations beginning with the location specified by the A address and continuing to, but not including, the terminal location specified by the B address. The contents of each card column occupies one 8-bit character position in memory. Reading is also terminated after 80 columns of information have been transferred to main memory.

VVV	T	RRRRRRRRRRRRRRRRRRRR	A	B
3 BITS	1	20 BITS	24 BITS	24 BITS

Figure 7-1. Card Read I/O Descriptor

Referring to figure 7-1:

a. VVV = 000 - card read operation code.
b. T = 0 - card read EBCDIC.
c. T = 1 - card read binary.
d. R = reserved.
e. A = beginning address of data.
f. B = ending address of data + 1.

CARD READ DESCRIPTOR (BINARY).

The card read controller is also capable of a binary read operation in which each hole in a card column is interpreted as a one bit and each non-hole is interpreted as a zero bit. The binary card image is stored in ascending memory locations beginning at the locations specified by the A address continuing until, but not into, the end location specified by the B address, or until 80 columns of information have been read, whichever occurs first. Each column is stored as 12 bits of data in main memory. (See figure 7-1.)

CARD TEST.
This I/O descriptor (figure 7-2) returns a result descriptor indicating that the test operation has been completed and the card reader is ready (operation complete) or the reader is not ready (operation not complete or exception). The test result descriptor also contains the controller ID.

VVV	TT	RRRRRRRRRRRRRRRRRRR
3 BITS	2	19 BITS

Figure 7-2. Card Read Test Descriptor

Referring to figure 7-2:

a. VVV = 100 - test card reader and controller.
b. TT = 00 - result store immediately.
c. TT = 10 - result is delayed until card reader is ready.
d. TT = 01 - result is stored when card reader is not ready.
e. R = reserved.

CARD READ RESULT DESCRIPTOR.
At the completion of a card read operation, the controller returns a result descriptor which is described as follows:

a. Bit 0 - I/O complete.
b. Bit 1 - exception condition (any bit on in bits 3-16).
c. Bit 2 - not ready.
d. Bit 3 - validity error.
e. Bit 4 - memory access error.
f. Bit 5 - reserved.
g. Bit 6 - read check.
h. Bits 7-16 - reserved.
i. Bits 17-23 - controller ID = 0101010 (test I/O only).

CARD PUNCH CONTROLLER.
The card punch I/O controller executes card punch I/O descriptors and punches data from ascending memory locations beginning at the location specified by the A address of the operand until 80 columns of information are punched. It allows EBCDIC card punch operation and a binary punch operation.

Information is transferred to the punch bit-serially for each punch position, with 12 separate 80-bit transfers being required to punch a card. Transfer of a given row to the punch consists of 40 memory accesses, obtaining punch information for two columns per access. The number of memory accesses required to punch a card is 480. A full card is always punched. The source program need not be concerned with the transfer of data to the punch.

The controller provides a punch echo check which checks if fewer than 80 data bits are received for each row or if fewer than 12 row cycles are counted. It also checks whether the number of punched holes agrees with the number of bits in the original data received by the I/O control unit.

CARD PUNCH DESCRIPTOR (BINARY).
A card is punched from ascending main memory locations beginning with the location specified by the A address (see figure 7-3). Punching is terminated after the punching of 80 columns, where each hole in A to be punched in a card column is represented by a one bit in memory.

VVV	T	SS	RRRRRRRRRRRRRRRRRR
3 BITS	1	2 BITS	18 BITS

A
24 BITS

Figure 7-3. Card Punch I/O Descriptor

Referring to figure 7-3:

a. VVV = 010 - card punch operation command.
b. T = 0 - card punch EBCDIC.
c. T = 1 - card punch binary.
d. SS = 00 - normal stacker.
e. SS = 01 - auxiliary stacker.
f. SS = 10 - error stacker.
g. R = reserved.
h. A = beginning address of data.

CARD PUNCH DESCRIPTOR (EBCDIC).
A card is punched from ascending main memory locations beginning with the address specified in A (see figure 7-3). One column is punched for each EBCDIC memory character and 80 columns are always punched.

CARD PUNCH TEST DESCRIPTOR.
The card punch and card punch controller are tested, and a result descriptor containing the controller ID number is returned (see figure 7-4).

VVV	T	RRRRRRRRRRRRRRRRRRRR
3 BITS	1	20 BITS

Figure 7-4. Card Punch Test Descriptor

Referring to figure 7-4:

a. VVV = 100 - test the card punch and controller.
b. T = 0 - the result is stored immediately.
c. T = 1 - the result is delayed until the reader is ready.
d. R = reserved.

CARD PUNCH RESULT DESCRIPTOR.
At the completion of an I/O operation the I/O controller retains a result descriptor, for the just-completed I/O, in the second 24 bits of the result status area. The result descriptor information for the card punch is as follows:

a. Bit 0 - I/O complete.
b. Bit 1 - exception condition (any bit on in bits 3-16).
c. Bit 2 - not ready.
d. Bit 3 - punch check error.
e. Bit 4 - memory access error.
f. Bit 5 - memory parity error.
g. Bits 6-16 - reserved.
h. Bits 17-23 - controller ID = 0000100 (test I/O only).

<u>PAPER TAPE READER CONTROLLER.</u>
This controller stores one 8-bit character of data per memory cycle. If the number of characters specified is not read, spaced, or backspaced because of termination by the reader, the short record bit (bit 9 of the result descriptor) is set. If the termination is caused by the end-of-tape or beginning-of-tape reflective strip, the appropriate EOT or BOT is set in the result descriptor.

PAPER TAPE READ DESCRIPTOR.
A record is read from the paper tape reader into ascending memory locations beginning at the location specified by the A address and continuing to, but not into, the address specified by the B address (see figure 7-5). If the T (type) variant of the read I/O contains a zero, eight bits received from the reader are transferred to memory via a BCL/EBCDIC translator. If the T variant contains a one, no translation occurs and eight bits received from the reader are transferred to memory.

VVV	T	RRRRRRRRRRRRRRRRRRRR	A	B
3 BITS	1	20 BITS	24 BITS	24 BITS

Figure 7-5. Paper Tape Read, Space, and Backspace I/O Descriptor

Referring to figure 7-5:

a. VVV = 000 - paper tape read operation command.
b. VVV = 101 - paper tape space.
c. VVV = 110 - paper tape backspace.
d. T = 0 - translate BCL to EBCDIC and check for odd parity.
e. T = 1 - no translation and check for even parity.
f. R = reserved for unit address.
g. A = beginning address of data.
h. B = ending address of data + 1.

PAPER TAPE SPACE DESCRIPTOR.
The paper tape is spaced forward the number of characters specified unless stopped by an end-of-tape condition. The number of characters is specified by the difference between the A and B addresses of the I/O descriptor (figure 7-5).

PAPER TAPE BACKSPACE DESCRIPTOR.
The paper tape is spaced backward the number of characters specified unless stopped by a beginning-of-tape condition. The number of characters is specified by the difference between the A and B addresses of the I/O descriptor (figure 7-5).

PAPER TAPE REWIND I/O DESCRIPTOR.
The paper tape is rewound to the beginning-of-tape position (see figure 7-6).

VVV	RRRRRRRRRRRRRRRRRRRRR
3 BITS	21 BITS

Figure 7-6. Paper Tape Rewind I/O Descriptor

Referring to figure 7-6:

a. VVV = 011 - paper tapc rewind operation command.
b. R = reserved for unit address.

PAPER TAPE TEST I/O DESCRIPTOR.
This descriptor (figure 7-7) tests the designated unit and returns a result descriptor.

VVV	T	RRRRRRRRRRRRRRRRRRRR
3 BITS	1	20 BITS

Figure 7-7. Paper Tape Test Descriptor

Referring to figure 7-7:

a. VVV = 100 - tests the paper tape reader and controller.

b. T = 0 - the result is stored immediately on completion.

c. T = 1 - the result is delayed until the reader is ready and it is not rewinding.

PAPER TAPE READ RESULT DESCRIPTOR.
At the completion of an I/O operation, the I/O controller returns a result descriptor, for the just-completed I/O, in the second 24 bits of the result status area. The result descriptor for the paper tape reader operation is:

a. Bit 0 - I/O complete.
b. Bit 1 - exception condition (any bit on in bits 3-16).
c. Bit 2 - not ready.
d. Bit 3 - tape parity error.
e. Bit 4 - memory access error.
f. Bit 5 - reserved.
g. Bit 6 - end of tape.
h. Bit 7 - beginning of tape.
i. Bit 8 - reserved.
j. Bit 9 - short record.
k. Bit 10 - rewinding.
l. Bits 11-16 - reserved.
m. Bits 17-23 - controller ID = 0001110 (test I/O only).

UNBUFFERED TRAIN PRINTER CONTROLLER.
This controller provides printer speeds of 400 and 750 LPM with 132 characters per line. It provides code conversion from internal EBCDIC to a binary code which is related to the graphic position for the standard 16, 48, 64, and 96 chains. If memory access is missed, the control waits until the same graphic is presented to it a second time and then continues, thus providing automatic recovery.

This controller also includes a quick release feature which terminates the operation and enables the start of paper motion as soon as all nonblank characters have been printed. Printing speed with this feature is dependent on the number of blank characters; however, the effective rate is usually greater than the rated speed.

BUFFERED LINE PRINTER CONTROLLER.
This controller provides for EBCDIC to BCL translation on the data as it is transferred from memory to the printer. The I/O controller contains a one-character buffer which receives information from memory serially by character and transfers the information serially by character to the printer buffer. Transfer of paper motion control information takes place after the printer buffer is filled. The I/O controller is capable of loading the printer buffer, while paper motion is occurring, in response to a previous I/O descriptor. After the paper motion of the previous command is completed, format information is received for the information in the buffer, which can now be printed. The controller provides for printer speed options of 300, 400, 475, 700, and 750 LPM with 132 characters per line. A 1040 LPM buffered line printer controller is available on the B 1726.

LINE PRINTER WRITE DESCRIPTOR.
A line of data is printed from ascending memory locations beginning at the memory location specified by the A address (see figure 7-8). The length of the line is determined by the number of printer columns, 80, 120. 132, or until the location specified by the B address is reached, whichever occurs first. Spacing or skipping takes place after printing, and skipping takes precedence over spacing.

VVV	I	TTTT	RRRRRRRRRRRRRRR	U	A	B
3 BITS	1	4 BITS	15 BITS	1	24 BITS	24 BITS

Figure 7-8. Line Printer Write Descriptor

Referring to figure 7-8:

a. VVV = 010 - printer write operation command.

b. I = 1 - inhibit reporting of EOP and skip to channel one if this operation calls for single or double spacing.

c. TTTT = 0000 - no paper advance.

d. TTTT = 1110 - single space after printing.

e. TTTT = 1111 - double space after printing.

f. TTTT = 0001 - skip to channel 1 after printing.

g. TTTT = 1100 - skip to channel 12 after printing.

h. TTTT = 1101 - skip to next channel after printing.

i. R = reserved.

j. U = 0 - printer unit 1.

k. U = 1 - printer unit 2.

l. A = beginning address of data.

m. B = ending address of data + 1.

LINE PRINTER SKIP DESCRIPTOR.
Skipping is controlled by bits 4-8 as in the write descriptor. Skipping takes precedence over spacing. The skip I/O descriptor format is the same as for the printer write (see figure 7-9).

VVV	I	TTTT	RRRRRRRRRRRRRRR	U
3 BITS	1	4 BITS	15 BITS	1

Figure 7-9. Line Printer Skip Descriptor

The printer space or skip operation command has VVV equal to 101. Variables are the same as for the write I/O descriptor.

LINE PRINTER TEST DESCRIPTOR.
The status of the designated unit is tested and a result descriptor is returned which contains the controller ID (see figure 7-10).

VVV	T	RRRRRRRRRRRRRRRRRRR	U
3 BITS	1	19 BITS	1

Figure 7-10. Line Printer Test Descriptor

Referring to figure 7-10:

a. VVV = 100 - test the printer and the printer controller.

b. T = 0 - result is stored immediately.

c. T = 1 - result is delayed until the printer is ready and paper motion is stopped.

d. R, U - same as for the write I/O descriptor.

LINE PRINTER RESULT DESCRIPTOR.
At the completion of the I/O operation the controller returns a result descriptor to the second 24 bits of the result status area. The result descriptor information is as follows:

a. Bit 0 - I/O complete.
b. Bit 1 - exception condition (any bit on in bits 3-16).
c. Bit 2 - not ready.

d. Bit 3 - code parity.
e. Bit 4 - reserved.
f. Bit 5 - memory parity error.
g. Bit 6 - end of page.
h. Bits 7-8-9:
   1) 000 - 64 character set (test I/O only).
   2) 001 - 48 character set.
   3) 010 - 16 character set.
   4) 011 - 96 character set.
   5) 100 - 192 character set.
i. Bits 10-11:
   1) 00 - 860 LPM or 400 LPM (test I/O only).
   2) 01 - 300 LPM.
   3) 10 - 600 LPM.
   4) 11 - 1100 LPM.
j. Bit 12 - paper in motion (test I/O only).
k. Bit 13 - motor-on test (test I/O only).
l. Bits 14-15:
   1) 00 - 132 columns per print line (test I/O only).
   2) 10 - 120 columns per print line.
   3) 11 - 80 columns per print line.
m. Bit 16 - reserved.
n. Bits 17-23 - controller ID = 0010000 (test I/O only).

CONSOLE PRINTER (SPO) CONTROLLER.

This I/O controller is used to control the console printer. The control provides a bidirectional translator which translates EBCDIC to USASCII (also referred to as ASCII and ANSCII) for the 128 USASCII codes. Translation for codes not specified is undefined.

Information between the console printer and the controller is transferred serially by bit, whereas it is transferred serially by character between the controller and main memory.

The controller can be "input request enabled," at which time the controller is sensitive only to the USASCII ENQ code from the keyboard. When the controller receives this code, an interrupt is set. The ENQ signal is remembered and is reported to main memory. Once reported, the interrupt remembering an ENQ signal is cleared.

Upon receipt of a read descriptor, the controller automatically generates a signal to light an indicator on the console printer and to await an input message from the systems operator. The controller is sensitive to the USASCII ETX (end of text) in an output message, which is translated to EBCDIC and transferred to main memory.

The controller is also sensitive to the USASCII NAK code which is also translated and stored in main memory. The NAK and ETX signals terminate a read operation.

The A and B addresses of the operand must define an integral number of 8-bit characters.

Data entered into the console printer is first sent to the I/O controller and then is retransmitted (echo back) to the teletypewriter for printing.

Console printer I/O descriptors are described in the following paragraphs.

CONSOLE PRINTER READ DESCRIPTOR.
The message being typed on the typewriter keyboard is read into ascending memory locations beginning with the location specified by the A address and continuing until an ETX code is detected, or up to the location specified by the B address (see figure 7-11). The ETX code is stored following the data stream only if the ETX signal terminates the operation.

VVV	T	RRRRRRRRRRRRRRRRRRRR	A	B
3 BITS	1	20 BITS	24 BITS	24 BITS

Figure 7-11. Console Printer
Read Descriptor

Referring to figure 7-11:

a. VVV = 000 - console printer read operation command.
b. T = 0 - EBCDIC translation.
c. T = 1 - no translation.
d. R = reserved.
e. A = beginning address of data.
f. B = ending address of data + 1.

CONSOLE PRINTER WRITE DESCRIPTOR.
A message is typed on the console printer for ascending memory locations beginning with the location specified by the A address and continuing until an ETX code is detected, or up to the location specified by the B address (see figure 7-12).

VVV	T	RRRRRRRRRRRRRRRRRRRR	A	B
3 BITS	1	20 BITS	24 BITS	24 BITS

Figure 7-12. Console Printer
Write Descriptor

The console printer write command has VVV equal to 010. Other variables are the same as for the read descriptor.

CONSOLE PRINTER TEST DESCRIPTOR.
The I/O controller and console printer are tested, and a result descriptor which contains the controller ID is returned (see figure 7-13).

VVV	T	RRRRRRRRRRRRRRRRRRRR
3 BITS	1	20 BITS

Figure 7-13. Console Printer Test Descriptor

Referring to figure 7-13:

a. VVV = 100 - console printer test.
b. T = 0 - store result descriptor immediately.
c. T = 1 - store result descriptor when ENQ is sensed.
d. R = reserved.

CONSOLE PRINTER RESULT DESCRIPTOR.
The result descriptor for the console printer is as follows:

a. Bit 0 - I/O complete.
b. Bit 1 - exception condition (any bit on in bits 3-17).
c. Bit 2 - not ready.
d. Bit 3 - NAK received.
e. Bit 4 - reserved.
f. Bit 5 - memory parity error.
g. Bit 6 - attempt to exceed maximum address.
h. Bit 7 - ENQ received.
i. Bits 8-16 - reserved.
j. Bits 17-23 - controller ID = 0010110 (test I/O only).

READER SORTER CONTROLLER.
The reader sorter controllers are used to execute I/O operations on reader sorters. One result descriptor is returned for each I/O operation initiated. The reader sorter control can store one character per memory access. The time between memory access is approximately 420 micro-seconds for an MICR read. The A and B addresses of the I/O descriptor must specify an integral number of 8-bit characters. Demand read operation is not permitted.

The controls provide an MICR-USASCII-8 translator which translates the first 16 USASCII codes (0000 0000 to 000 1111 inclusive) to EBCDIC as shown in table 7-1.

Table 7-1

Reader Sorter
MICR/USASCII-8 Translation

Sorter Symbol	Sorter Code	I/O Controller Internal Code	System Internal Graphic	Internal EBCDIC Code
0	000 0000	0011 0000	0 (F0)	1111 0000
1	000 0001	0011 0001	1 (F1)	1111 0001
2	000 0010	0011 0010	2 (F2)	1111 0010
3	000 0011	0011 0011	3 (F3)	1111 0011
4	000 0100	0011 0100	4 (F4)	1111 0100
5	000 0101	0011 0101	5 (F5)	1111 0101
6	000 0110	0011 0110	6 (F6)	1111 0110
7	000 0111	011 0111	7 (F7)	1111 0111
8	000 1000	0011 1000	8 (F8)	1111 1000
9	000 1001	0011 1001	9 (F9)	1111 1001
Amount (S2)	000 1010	0010 0100	# (7B)	0111 1011
Transit (S2)	000 1011	0011 1100	@ (7C)	0111 1100
On-Us (S3)	000 1100	0010 0011	: (7A)	0111 1010
Hyphen (S4)	000 1101	0010 1101	- (60)	0110 0000
S5	000 1110	0010 0111	' (7D)	0111 1101
Cannot read	000 1111	0011 1111	? (6F)	0110 1111

The MICR control provides a USASCII/EBCDIC translator which translates the center 64 USASCII codes (010 0000 to 101 1111 inclusive) to the corresponding EBCDIC code's. (Refer to table 7-2.) In addition, the USASCII code 001 1010 for SUB is translated to the USASCII-8 code 0001 1010 for SUB.

Validity checking for MICR reading is as follows:

a. Amount field - 1st and 12th character for amount symbols and intervening 10 characters for decimal digit's.

b. Transit field - 40th and 50th characters for transit symbols and intervening nine characters for decimal digits except for hyphen in position 45.

Table 7-2

Reader Sorter
USASCII/EBCDIC Translation

Sorter Symbol	Sorter Code	EBCDIC Code	Sorter Symbol	Sorter Code	EBCDIC Code
Blank	010 0000	0100 0000	@	100 0000	0111 1100
Vertical bar	010 0001	0100 1111	A	100 0001	1100 0001
"	010 0010	0111 1111	B	100 0010	1100 0010
#	010 0011	0111 1011	C	100 0011	1100 0011
$	010 0100	0101 1011	D	100 0100	1100 0100
%	010 0101	0110 1100	E	100 0101	1100 0101
&	010 0110	0101 0000	F	100 0110	1100 0110
'	010 0111	0111 1101	G	100 0111	1100 0111
(	010 1000	0100 1101	H	100 1000	1100 1000
)	010 1001	0101 1101	I	100 1001	1100 1001
*	010 1010	0101 1100	J	100 1010	1101 0001
+	010 1011	0100 1110	K	100 1011	1101 0010
,	010 1100	0110 1011	L	100 1100	1101 0011
-	010 1101	0110 0000	M	100 1101	1101 0100
.	010 1110	0100 1011	N	100 1110	1101 0101
/	010 1111	0110 0001	O	100 1111	1101 0110
0	011 0000	1111 0000	P	101 0000	1101 0111
1	011 0001	1111 0001	Q	101 0001	1101 1000
2	011 0010	1111 0010	R	101 0010	1101 1001
3	011 0011	1111 0011	S	101 0011	1110 0010
4	011 0100	1111 0100	T	101 0100	1110 0011
5	011 0101	1111 0101	U	101 0101	1110 0100
6	011 0110	1111 0110	V	101 0110	1110 0101
7	011 0111	1111 0110	W	101 0111	1110 0110
8	011 1000	1111 1000	X	101 1000	1110 0111
9	011 1001	1111 1001	Y	101 1001	1110 1000
:	011 1010	0111 1010	Z	101 1010	1110 1001
;	011 1011	0101 1110	[	101 1011	0100 1010
<	011 1100	0100 1100	\	101 1100	1110 0000
=	011 1101	0111 1110	]	101 1101	0101 1010
>	011 1110	0110 1110	¬	101 1110	0101 1111
?	011 1111	0110 1111	-	101 1111	0110 1101
Cannot read (SUB)	001 1010	0011 1111			

NOTE

Leading unreadable characters are not stored and are not reported as errors.

Formatting for MICR reading is accomplished by storing all data except for hyphens in the ON-US field into descending memory locations until the first transit symbol is detected and then storing blanks until the

40th character location is reached. The transit symbol code is stored in the 40th character position followed by the remaining data. In no case is data stored more often than every eight micro-seconds.

In flow mode, the appropriate operation complete (OC) bit is inspected just prior to the too-late-to-read time. If the OC bit is false, the read takes place and the link address is used to obtain a new set of addresses for the next read operation. If an OC bit is true, all subsequent read operations associated with that station are discontinued. However, the other read station, if active, continues to read and store data. The feeder is stopped by the presence of the OC bit for either read stations or by a too-late-to-pocket-select condition or by a halt variant contained in the read OP code.

Pocket select and interrupt information, if present, are obtained from memory via a swap operation just prior to the too-late-to-pocket-select time. The controller first swaps a lockout pattern of 24 zero bits and then determines whether valid information is present. The controller then stores the pocket select result information. If valid information is not present, the feeder is stopped; the current item and all subsequent items are rejected. The too-late bit is reported in the pocket select result descriptor area immediately. A new initiate can be accepted while the controller is rejecting previous items.

An interrupt can be requested at the completion of any read from the first station, the second station, or both. A pending interrupt does not prevent the storing of subsequent read data but does cause any subsequent request for an interrupt to be ignored.

If an unencoded item is encountered, it is reported as such in the read result descriptor. The read area in memory is unaffected.

If a batch ticket is encountered, it is reported as such in the read result descriptor for the read area in which the batch ticket data has been stored.

If underspaced, overlength, or double items are detected, the control does <u>not</u> read the fault item(s) but rejects them <u>without</u> any result indication. The feeder is not stopped and subsequent items are read normally. However, if the fault item is a batch ticket, it is reported in the read result descriptor as a rejected item. The read information is not valid.

If an empty hopper, full pocket, or sorter stop button is detected, all items in motion are processed. The not-ready bit is reported in the pocket select result descriptor area when the last item read is pocketed.

If a jam or missort is detected, the sorter stops the feeder and the control rejects all items in the feed line which have not been read. The jam-missort bit is reported in the pocket select result descriptor area when the last item read is pocketed. After a pocket light descriptor is sent to the sorter, the sorter is placed in a not-ready status and accepts only pocket light operations.

The I/O descriptor for a reader sorter operation is shown in figure 7-14. Each field is 24 bits wide.

AEA$_1$	RD$_1$	LINK$_1$	OP$_1$	A$_1$	B$_1$	PS	AEA$_2$	RD$_2$	A$_2$	B$_2$
RSA$_1$							RSA$_2$			

Figure 7-14. Reader Sorter I/O Descriptor

Referring to figure 7-14:

a. Read station one.

1) AEA-sub1 - actual ending address.
2) RD-sub1 - result descriptor.
3) LINK-sub1 - link address.
4) OP-sub1 - operation code and variants.
5) A-sub1 - start address.
6) B-sub1 - ending address of data + 1.
7) PS - pocket select information and result.

b. Read station two (present only if variants specify a read for both stations).

1) AEA-sub2 - actual ending address.
2) RD-sub2 - result descriptor.
3) A-sub2 - start address of data.
4) B-sub2 - ending address of data + 1.

POCKET SELECT INFORMATION.
Pocket select information, if present, is obtained from the 24-bit PS portion of the I/O descriptor (see figure 7-15).

VV	PPPPP	RRRRRRRRRRRRRRRRR
2 BITS	5 BITS	17 BITS

Figure 7-15. PS (Pocket Select) Portion of Reader Sorter I/O Descriptor

Referring to figure 7-15:

a. VV = 00 - valid information is present and a normal pocket selection is performed.

b. VV not equal to 00 - valid pocket select information is not present and the control waits and again checks if valid information is present.

c. PPPPP = 00000-11111 - from four to 32 pockets are available for selection.

d. R = reserved.

READ READER SORTER DESCRIPTOR.
Read the data from the appropriate read station(s) into descending memory locations starting at the location specified by the B address of the I/O descriptor minus one and continuing until, but not into, the location specified by the A address plus one or until all data from the item has been read, whichever occurs first. (See figure 7-16.)

VVV	R	C	D	E	F	G	H	I	J	R	H	RRRRRRRRRR
3 BITS	1	1	1	1	1	1	1	1	1	1	1	10 BITS

Figure 7-16. Read Reader Sorter Descriptor

Referring to figure 7-16:

a. VVV = 000 - read reader sorter operation command.
b. R = reserved.
c. C = 1 - read first station.
d. G = 1 - read second station.
e. CG = 11 - read both stations.
f. D = 1 - format and validity check first station.
g. H = 1 - format and validity check second station.
h. DH = 11 - format and validity check both stations.
i. E = 1 - do not store delimiter, first station.
j. I = 1 - do not store delimiter, second station.
k. F = 1 - delimiter = blank, first station.
l. J = 1 - delimiter = blank, second station.
m. H = 1 - halt the feeder.

BATCH COUNT DESCRIPTOR.
The batch counter in the sorter is advanced by one. Flow must be stopped and all items must be pocket-selected. (See figure 7-17.)

VVV	RRRRRRRRRRRRRRRRRRRRR
24 BITS	21 BITS

Figure 7-17. Batch Count Descriptor

Referring to figure 7-17, VVV is equal to 101, and R indicates reserved.

POCKET LIGHT DESCRIPTOR.

The light in the pocket specified (0-30) in the I/O descriptor is illuminated. Flow must be stopped and all items must be pocket-selected. (See figure 7-18.)

VVV	NNNNN	RRRRRRRRRRRRRRRR
3 BITS	5 BITS	16 BITS

Figure 7-18. Pocket Light Descriptor

Referring to figure 7-18:

a. VVV = 010 - pocket light enable operation command.

b. NNNNN = 00000-11111 - light specific pocket q through 32 (0-31).

c. R = reserved.

READER SORTER TEST DESCRIPTOR.

The reader sorter and controller are tested, a not-ready, jam, or missort condition is reported, if present, and an appropriate result descriptor is returned. (See figure 7-19.)

VVV	T	RRRRRRRRRRRRRRRRRRRR
3 BITS	1	20 BITS

Figure 7-19. Reader Sorter Test Descriptor

Referring to figure 7-19:

a. VVV = 100 - test reader sorter I/O controller and reader sorter.

b. T = 0 - store result immediately.

c. T = 1 - store result when reader sorter is ready.

d. R = reserved.

READER SORTER RESULT DESCRIPTOR.

At the completion of the operation, the result status information is stored at the specified location. The result descriptor information is as follows:

a. Bit 0 - operation complete.
b. Bit 1 - exception condition (any bit on 3 through 16 set).
c. Bit 2 - not ready.

d. Bit 3 - unencoded document.
e. Bit 4 - memory access error.
f. Bit 5 - cannot read document.
g. Bit 6 - amount field error.
h. Bit 7 - on-us field error.
i. Bit 8 - transit field error.
j. Bit 9 - double documents.
k. Bit 10 - too late to read.
l. Bit 11 - jam.
m. Bit 12 - missort.
n. Bit 13 - batch ticket (no item left in path to be read).
o. Bit 14 - flow stopped (no items left in path to be read).
p. Bit 15 - empty hopper, full stacker, stop.
q. Bit 16 - reserved.
r. Bits 17-23 - control ID = 0010100 (test I/O only).

DISK PACK CONTROLLER (B 1726 ONLY).
This I/O controller provides for the attachment of disk packs to the system. The maximum capacity of each pack is 30 million bytes.

The controller can be used with two disk pack drives in a 1 x 2 configuration.

The disk pack control can store or fetch 24 bits of data per memory access and is capable of receiving and recording information at a five megabit per second rate.

Correctness of information in main memory is verified by the port interchange. If a parity error is detected during the fetch of any portion of an I/O descriptor, a message is returned by the controller to the processor containing the address and 24 of the bits which have caused the memory parity error. Any error detected during data transfer causes the return of an appropriate result descriptor.

The disk pack controller assumes a linked list of I/O descriptors. Once a control is initiated with an address which points to a result descriptor, the control reads the first two bits (00) of the result descriptor. The control fetches that I/O descriptor and executes it. If not, the control exits by fetching the link address to the next result descriptor.

Each disk pack consists of 10 disk platters; each has 20 data surfaces numbered 00-19. There are 33 available segments on each surface in a cylinder except the first, on which five alternate segments are reserved. These alternate segments are used for relocating segments which had been unrecordable in their original location. Continuous operation from segment to segment, track to track, and cylinder to cylinder is permitted, but continuous operation from disk pack unit to unit is not.

DISK PACK WRITE DESCRIPTOR.
Data is written to the disk pack starting at the given file address (C) from ascending memory locations beginning at the location specified by the A address continuing to but not from the end location specified by

the B address. Zero bits are written to complete the last segment. (See figure 7-20.)

VVV	RR	E	C	T	NNN	RRRRRRRRR	UUUU	A	B	C
3 BITS	2	1	1	1	3 BITS	9 BITS	4 BITS	24 BITS	24 BITS	24 BITS

Figure 7-20. Disk Pack Write Descriptor

Referring to figure 7-20:

a. VVV = 010 - disk pack write operation command.

b. R = Reserved

c. E = 0 - enable automatic restore after seek error.

d. E = 1 - disable automatic restore after seek error.

e. C = 0 - do not parity-check segment written.

f. C = 1 - parity-check all segments written (to be specified if required).

g. T = 0 - segment mode.

h. T = 1 - track mode.

i. NNN = 000 - normal segment address.

j. NNN = 001 - spare segment #1 on selected track.

k. NNN = 010 - spare segment #2 on selected track.

l. NNN = 011 - spare segment #3 on selected track.

m. NNN = 100 - spare segment #4 on selected track.

n. NNN = 101 - spare segment #5 on selected track.

o. UUUU = 0-15 - unit number.

p. R = Reserved

q. A = beginning address of data.

r. B = ending address of data + 1.

s. C = binary file address.

DISK PACK READ DESCRIPTOR.
Data is read from the disk pack starting at the given file address (C) into ascending memory locations beginning at the location specified by the A address and ending at but not in the end location specified by the B address. A complete segment need not be stored but is parity-checked. (See figure 7-21.)

VVV	M	R	E	C	T	NNN	RRRRRRRRR	UUUU	A	B	C
3 BITS	1	1	1	1	1	3 BITS	9 BITS	4 BITS	24 BITS	24 BITS	24 BITS

Figure 7-21. Disk Pack Read Descriptor

Referring to figure 7-21:

a. VVV = 000 - disk pack read operation.

b. M = 1 - normal read operation.

c. M = 0 - if reading a relocated segment, the address field returned contains all ones.

d. C = 0 - enable error correction.

e. C = 1 - disable error correction.

f. Other variables are the same as for the disk pack write descriptor.

DISK PACK INITIALIZE DESCRIPTOR.
Segment addresses, data (all zeros), and gaps in all tracks are written starting after the index pulse on the track decoded from the given file address continuing through the entire track, cylinder, or pack. (See figure 7-22.) The data consist of a 16-bit pattern obtained from ascending memory locations at the specified A address. This 10-bit pattern is repeated 90 times throughout each data segment.

VVV	R	R	E	R	T	PP	RRRRRRRRRR	UUUU	A	B	C
3 BITS	1	1	1	1	1	2	10 BITS	4 BITS	24 BITS	24 BITS	24 BITS

Figure 7-22. Disk Pack Initialize Descriptor

Referring to figure 7-22:

a. VVV = 011 - disk pack initialize operation.

b. E = 0 - enable automatic restore after seek error.

c. E = 1 - disable automatic restore after seek error.

d. T = 0 - segment mode.

e. T = 1 - track mode.

f. PP = 00 - entire pack.

g. PP = 01 - cylinder only.

h. PP = 10 - track only.

i. PP = 11 - undefined.

j. UUUU = 0-15 - unit number.

k. R = reserved.

DISK PACK VERIFY DESCRIPTOR.
The disk pack is read and checked for address errors and information parity errors beginning with the first segment after the index pulse on the track decoded from the given file address, continuing through the entire track, cylinder, or pack. The positions of segments are verified by counting from the index on each track.

The binary file address of the detected error(s) is reported into ascending memory locations beginning at the location specified by the A address plus 16. The data is checked by comparing the 16-bit pattern at the location specified by the A address with each 16-bit group of data in the segment.

NOTE
Relocated segments are also checked.

VVV	R	R	E	R	T	PP	RRRRRRRRRR	UUUU
3 BITS	1	1	1	1	1	2	10 BITS	4 BITS

A	B	C
24 BITS	24 BITS	24 BITS

Figure 7-23. Disk Pack Verify Descriptor

Referring to figure 7-23:

a. VVV = 001 - verify disk pack operation.

b. E = 0 - enable automatic restore after seek error.

c. E = 1 - disable automatic restore after seek error.

d. T = 0 - segment mode.

e. T = 1 - track mode.

f. PP = 00 - entire pack.

g. PP = 01 - cylinder only.

h. PP = 10 - track only.

i. PP = 11 - undefined.

j. UUUU = 0-15 - unit number.

k. R = reserved.

DISK PACK RELOCATE DESCRIPTOR.
The segment address designated by the given file address (C) is flagged with an error configuration, and is rewritten on track 00 in the space sector specified by N. Both the original address field and the relocated address field on track 00 are located by counting from the index on each track.

NOTE

> An error configuration is one byte of binary "ones" with clock pulses omitted, followed by one byte of binary "ones" (with clock pulses), followed by one byte of binary "ones" with clock pulses omitted. Also the Sync Code is changed to eight zeros.

The error configuration is written in the address field of the designated sector. The standard test data pattern is also written in the relocated segment data field. (See figure 7-24.)

VVV	R	R	E	R	T	NNN	RRRRRRRR	UUUU	A	B	C
3 BITS	1	1	1	1	1	3 BITS	8 BITS	4 BITS	24 BITS	24 BITS	24 BITS

Figure 7-24. Disk Pack Relocate Descriptor

Referring to figure 7-24:

a. VVV = 101 - relocate segment operation.

b. E = 0 - enable automatic restore after seek error.

c. E = 1 - disable automatic restore after seek error.

d. T = 0 - segment mode.

e. T = 1 - track mode.

f. NNN = 1-5 - indicates the spare segment in 28 through 32 on track 00.

g. R = reserved.

DISK PACK TEST DESCRIPTOR.
The disk pack drive is tested for the following conditions (see figure 7-25):

a. Readiness of drive.
b. Busyness of drive.
c. Timeout of seek operation.
d. Seek complete status.
e. Seeking status.
f. Control identification.
g. Write lockout.
h. Drive type.

VVV	E	E	P	P_1	RRRRRRRRRRRRR	UUUU
3 BITS	1	1	1		13 BITS	4 BITS

Figure 7-25. Disk Pack Test Descriptor

Referring to figure 7-25:

a. VVV = 100 - disk pack test operation.

b. EEP = 000 - return a result immediately. Do not pause.

c. EEP = 100 - return a result only if the drive is ready and in a seek complete status; otherwise, unlock and fetch the next descriptor.

d. EEP = 010 - returns a result only if the drive is present else unlock and fetch the next descriptor.

e. EEP = 001 - if no I/O descriptor has been executed since the last encounter with a test descriptor with EEP = 001, unlock and then pause one to two milliseconds before proceeding to fetch the result descriptor field of the next descriptor. If an I/O descriptor has been executed, do not pause; unlock and then proceed immediately to fetch the result descriptor field of the next descriptor. Do not store a result in either case.

f. P_1 = 1 - put the drive off time for pack removal.

g. R = reserved.

The controller can store and fetch 16 bits of data per memory access and is capable of handling a data transfer rate from and to a drive of 1.55 and 3.10 megabits per second.

Segment to segment, track to track, and cylinder to cylinder operation is permitted but continous operation from unit to unit is not permitted.

Bad tracks should not be addressed by the software in any routines other than initialization routines. Bad tracks are not detected by the control, and reading and writing on or over bad tracks produces an undefined result.

DISK CARTRIDGE READ DESCRIPTOR.
Data is read from the disk cartridge starting at the given file address (C) into ascending memory locations beginning at the location specified by the A address and ending at but not in the end location specified by the B address. A complete segment need not be stored but is parity-checked. (See figure 7-27.)

VVV	R	M	R	S	RRRRRRRRRRRR	UUU	A	B	C
3 BITS	1	1	1	1	12 BITS	3 BITS	24 BITS	24 BITS	24 BITS

Figure 7-27. Disk Cartridge Read Descriptor

Referring to figure 7-27:

a. VVV = 000 - disk cartridge read.
b. MS = 00 - normal read.
c. MS = 01 - undefined.
d. MS = 10 - read immediately following the segment pulse.
e. MS = 11 - undefined.
f. UUU = 000-111 - disk cartridge unit number.
g. R = reserved.
h. A = beginning address of data.
i. B = ending address of data + 1.
j. C = binary absolute disk address.

DISK CARTRIDGE WRITE DESCRIPTOR.
Data is written to the disk cartridge starting at the given file address (C) from ascending memory locations beginning at the location specified by the A address continuing to but not from the end location specified by the B address. Zero bits are written to complete the last segment. (See figure 7-28.)

VVV	R	M	R	S	RRRRRRRRRRRRRR	UUU	A	B	C
3 BITS	1	1	1	1	14 BITS	3 BITS	24 BITS	24 BITS	24 BITS

Figure 7-28. Disk Cartridge Write Descriptor

Referring to figure 7-28:

a. VVV = 010 - disk cartridge write.

b. MS = 00 - normal write.

c. MS = 01 - undefined.

d. MS = 10 - write immediately following the segment pulse.

e. MS = 11 - undefined.

f. Other variables are the same as for the disk cartridge read descriptor.

DISK CARTRIDGE TEST DESCRIPTOR.
The designated controller is tested for the following conditions:

a. Readiness of cartridge drive.
b. Write lockout.
c. Timeout of seek operation.
d. Seek status.
e. Seeking status.

f. Control identification.

The disk cartridge test descriptor is shown in figure 7-29.

VVV	MS	RRRRRRRRRRRRRRRR	UUU
3 BITS	2	16 BITS	3 BITS

Figure 7-29. Disk Cartridge Test Descriptor

Referring to figure 7-29:

a. VVV = 100 - disk cartridge test.

b. MS = 00 - return a result immediately unconditionally.

c. MS = 01 - do not return any result unless ready.

d. MS = 01 - do not return any result but pause one to two milliseconds before fetching the next I/O descriptor.

DISK CARTRIDGE RESULT DESCRIPTOR.
The disk cartridge result descriptor is defined as follows:

a. Bit 0 - operation completed.

b. Bit 1 - exception condition.

c. Bit 2 - not ready.

d. Bit 3 - read parity error.

e. Bit 4 - memory access error.

f. Bit 5 - memory parity error.

g. Bit 6 - write lockout operation not performed (test).

h. Bit 8 - overrun.

i. Bits 7-8-9:

1) 000 not present (test).
2) 100 2200 BPI, 203T (test).

j. Bit 10 - reserved.

k. Bit 11 - seek timeout operation not performed.

l. Bit 12 - seek status (test).

m. Bit 13 - reserved.

n. Bit 14 - seeking (test).

o. Bit 15 - reserved.

p. Bit 16 - reserved.

q. Bits 17-23 - controller ID.

MAGNETIC TAPE CONTROLLERS.

7-TRACK MAGNETIC TAPE CONTROLLER.

This controller is used with all 7-track free-standing transports. It has the capability of operating at 200, 556, and 800 BPI densities. A maximum of six transports can be used on the controller. The ability to perform non-stop forward operations is also included.

9-TRACK MAGNETIC TAPE CONTROLLER.

This controller is used with any 9-track free-standing transports and 9-track magnetic tape clusters. It has the capability of operating at 1600 and 800 BPI density.

NOTE
200 and 556 BPI are not provided.

A maximum of six transports can be used on the controller.

The magnetic tape I/O descriptors are described below.

MAGNETIC TAPE READ DESCRIPTOR.

This operation reads data from magnetic tape in either a forward or backward direction into ascending or descending memory locations as specified by the A address and continuing until the end location specified by the B address (see figure 7-30).

VVV	T	C	FFF	PPPP	RRRRRRRR	UUUU	A	B
3 BITS	1	1	3 BITS	4 BITS	8 BITS	4 BITS	24 BITS	24 BITS

Figure 7-30. Magnetic Tape Read Descriptor

Referring to figure 7-30:

a. VVV = 000 - read operation command.
b. T = 0 - read forward.
c. T = 1 - read reverse.
d. C = 1 - current track.
e. FFF = 000 through 111 - track number.
f. PPPP = 0110 - 800 BPI even parity 7-track.
g. PPPP = 0111 - 800 BPI odd parity 7-track.
h. PPPP = 1011 - 800 BPI odd parity 9-track.
i. PPPP = 0010 - 556 BPI even parity 7-track.

j. PPPP = 0011 - 556 BPI odd parity 7-track.
k. PPPP = 0000 - 200 BPI even parity 7-track.
l. PPPP = 0001 - 200 BPI odd parity 7-track.
m. PPPP = 1111 - 1600 BPI odd parity 9-track (phase encoded).
n. PPPP = 0100 - (as selected by switch - even parity).
o. PPPP = 0101 - (as selected by switch - odd parity).
p. R = reserved.
q. A = beginning address of data.
r. B = ending address of data + 1.
s. UUUU = 0000 through 1111 - tape unit number.

MAGNETIC TAPE WRITE DESCRIPTOR.
This operation writes a data to tape moving in the forward direction from ascending memory locations beginning at the location specified by the A address and continuing until the location specified by the B address. (See figure 7-31.)

MAGNETIC TAPE ERASE DESCRIPTOR.
Magnetic tape is erased in the forward direction on the designated unit. The number of characters erased is the number of characters which would be written for the same A and B addresses. No memory space is used, but memory cycle time is used. (See figure 7-31.)

VVV	E	T	RRRR	FFFF	RRRRRRRR	UUUU	A	B
3 BITS	1	1	4 BITS	4 BITS	8 BITS	4 BITS	24 BITS	24 BITS

Figure 7-31. Magnetic Tape Write/Erase Descriptor

Referring to figure 7-31:

a. VVV = 010 - tape write operation command.
b. E = 1 - erase.
c. T = 1 - write tape mark.
d. All other variables are the same as in tape read.

MAGNETIC TAPE REWIND DESCRIPTOR.
Magnetic tape is rewound on the designated unit to the beginning of tape (BOT). After the operation is initiated, a result descriptor is returned with the I/O complete bit set ON without waiting until the rewind is complete. (See figure 7-32.)

VVV	RRRRRRRRRRRRRRRRR	UUUU
3 BITS	17 BITS	4 BITS

Figure 7-32. Magnetic Tape Rewind Descriptor

Referring to figure 7-32:

a. VVV = 011 - tape rewind operation command.
b. R = reserved.
c. UUUU = 0000-1111 - tape unit number.

MAGNETIC TAPE SPACE DESCRIPTOR.
Magnetic tape is spaced forward or backward by the number of records specified by bits 13-20 of the tape space I/O descriptor (001 through 255 physical records) unless tape is stopped by detection of an end of file (EOF), end of tape (EOT), or beginning of tape (BOT) condition. (See figure 7-33.)

VVV	T	RRRR	PPPP	RRRRRRR	S	UUUU
3 BITS	1	4 BITS	4 BITS	7 BITS	1	4 BITS

Figure 7-33. Magnetic Tape Space Descriptor

Referring to figure 7-33:

a. VVV = 110 - space operation command.
b. S = 0 - space to next end of file (EOF).
c. S = 1 - space one record.
d. All other variables are the same as in the tape descriptor.

MAGNETIC TAPE TEST DESCRIPTOR.
This descriptor tests the unit specified and returns a result descriptor (see figure 7-34).

VVV	TT	RRRRRRRRRRRRRRR	UUUU
3 BITS	2	15 BITS	4 BITS

Figure 7-34. Magnetic Tape Test Descriptor

Referring to figure 7-34:

a. VVV = 100 - test operation command.

b. TT = 00 - always return a result descriptor.

c. TT = 10 - return a result descriptor only if the unit is ready.

d. TT = 01 - return a result descriptor only if the unit is not ready.

e. All other variables are the same as for the tape read descriptor.

MAGNETIC TAPE LOCK DESCRIPTOR.
There is one lock descriptor in the subsystem chain for each magnetic tape unit (see figure 7-35).

VVV	RRRRRRRRRRRRRRRRRRRRR	A	B
3 BITS	21 BITS	24 BITS	24 BITS

Figure 7-35. Magnetic Tape Lock Descriptor

Referring to figure 7-35:

a. VVV = 101 - magnetic tape lock operation.
b. R = reserved.
c. A = address of first non-lock descriptor.
d. B = address of current non-lock descriptor.

MAGNETIC TAPE RESULT DESCRIPTOR.
At the completion of an I/O operation the following result descriptor information is returned:

a. Bit 0 - I/O complete.
b. Bit 1 - exception condition (any bit on in bits 3-16).
c. Bit 2 - not ready.
d. Bit 3 - tape parity error.
e. Bit 4 - memory access error.
f. Bit 5 - memory parity error.
g. Bit 6 - end of tape (EOT).
h. Bit 7 - beginning of tape (BOT).
i. Bit 8 - write lockout.
j. Bit 9 - end of file (EOF).
k. Bit 10 - tape is rewinding.
l. Bit 11 - spacing of three feet of blank tape (timeout).
m. Bit 12 - cyclic redundancy check.
n. Bits 13-15 - track in error (000-111).
o. Bit 16 - reserved.
p. Bit 17 - short record (incomplete record).
q. Bits 18-23 - reserved.

The magnetic tape test result descriptor is as follows:

a. Bit 0 - I/O complete.

b. Bit 1 - exception condition (any bit on in bits 3-16).

c. Bit 2 - not ready.

d. Bit 3 - busy.

e. Bits 4-5 - reserved.

f. Bit 6 - end of tape.

g. Bit 7 - beginning of tape.

h. Bit 8 - write lockout.

i. Bit 9 - reserved.

j. Bit 10 - rewinding.

k. Bit 11 - reserved.

l. Bits 12-14:

1) 000 - 7-track 200 BPI.
2) 001 - 7-track 556 BPI.
3) 011 - 7-track 800 BPI.
4) 101 - 9-track 800 BPI.
5) 111 - 9-track 1600 BPI.

m. Bits 15-16 - reserved.

n. Bits 17-23 - controller ID.

1) ID #1 - 0110010.
2) ID #2 - 0110000.
3) ID #3 - 0110100.
4) ID #4 - 0110110.
5) ID #5 - 0111000.

NOTE

The cyclic redundancy check is a means of generating another check character during write operations and provides error correcting capabilities. During a read, the generated check character must agree with the one that has been written.

DATA COMMUNICATIONS CONTROLLERS.
A data communications controller provides for certain basic functions described below with non-standard functions peculiar to particular line disciplines provided for by the specific line adapter. Basic functions of data communications controllers are:

a. I/O initiate handling.

b. I/O descriptor fetch for an adapter.

c. Provides for storage in an adapter for address of result status area, beginning of data address, and end of data address + 1.

d. Initiating an adapter to the appropriate operating mode.

e. Servicing an adapter request for character transfers.

f. USASCII-7 to/from EBCDIC translation as selected by variant.

g. Result descriptor and interrupt handling.

The data communications controller accepts all I/O initiate descriptors directed to it regardless of the state of an adapter. If an I/O descriptor is directed to a busy adapter, the control forces the adapter to the new operating state, destroying all traces of the previous operation including any pending interrupts.

On the controller a parity error or a memory access error does not terminate a read or write operation. On a read, an access failure results in a loss of the character. On a write, an access failure results in the transmittal of a sync character on a synchronous line and a steady marking condition on an asynchronous line.

SINGLE-LINE CONTROLLER.
A single-line controller provides connection between a single I/O channel and a single communications line. A line adapter is required to equip the single-line controller for use with a particular type of remote I/O device. The adapter determines code sensitivity, transmission rate, and character length in bits.

LINE ADAPTERS FOR SINGLE-LINE CONTROLS.
The adapters are input/output devices that provide the terminal connections between a telephone or telegraph facility and an I/O channel via the single-line controls.

During input, all adapters receive data from the line serially by bit. When a bit has been stored, the adapter signals the control to accept the bit. During output, all adapters receive data serially by bit from the I/O control. As each bit is transmitted, the adapter requests the next bit from the control.

The adapters provide for both a receive and a transmit break. On receipt of a break I/O descriptor, the adapter terminates the operation in process, if any, and transmits the break signal. It does not return a result descriptor for an operation in process. The result descriptor for the break is returned after completion of the break operation.

A time-out feature is provided for the I/O controller/adapter hookup when using leased lines or direct line operation and provides a read time-out which detects the absence of a data character for a period of time greater than one second. The time-out starts immediately after the receipt of the I/O descriptor. The time-out can be inhibited programmatically. The time-out period can be changed by a field engineer to a 20-second period.

Time-out also occurs if the clear-to-send signal from the data set is not returned within 20 seconds after the request to send. This time-out cannot be inhibited.

Three types of standard line adapters are available.

a. Standard synchronous line adapter.
b. Standard asynchronous line adapter.
c. Standard direct asynchronous line adapter.

Standard line adapters provide for the attachment of the following terminals:

a. Teletypes (models 33, 35, and 37).
b. Burroughs Input and Display - (B 9353).
c. CRT Input and Display (B 9352).
d. Burroughs TC-Series.
e. Burroughs TU-Series.
f. Burroughs RT-Series.

SYNCHRONOUS LINE ADAPTER.
The standard synchronous line adapter(s) provide for a connection of the listed terminals via the following common carrier services and data sets. Data sets equivalent to those listed can be used but must first be approved and specified. The clock is supplied by the data set.

Speed (BPS)	Service	Data Set	Burroughs Data Sets
2000	Dial	WE 201A3	
2400	Leased	WE 201B1	TA 713/TA 783
4800	Leased	WE 203	TA 713/TA 783
9600	Leased	WE 203	

ASYNCHRONOUS LINE ADAPTER.
Standard asynchronous line adapter(s) provide for a connection of the listed terminals via the following common carrier services and data sets. Data sets equivalent to those listed can be used but must first be approved and specified. The adapters may be modified to accommodate any one of the listed transmission speeds.

Speed (BPS)	Service	Data Set	Burroughs Data Sets
150	Dial	WE 103A	
150	Leased	WE 103F	TA 713/753
300	Dial	WE 202C	
300	Leased	WE 202D	TA 713/753
600	Dial	WE 202C	
600	Leased	WE 202D	TA 713/753

Speed (BPS)	Service	Data Set	Burroughs Data Sets
1200	Dial	WE 202C	
1200	Leased	WE 202D	TA 713/753
1800	Leased	WE 202D	TA 783

DIRECT ASYNCHRONOUS LINE ADAPTER.
The standard direct asynchronous line adapter provides for a connection of the listed terminals via a 2-wire direct connect interface. The adapter may be modified to accommodate any of the following listed transmission speeds (in BPS):

a. 110.
b. 150.
c. 300.
d. 1200.
e. 1800.
f. 2400.
g. 4800.
h. 9600.

ADAPTER CHARACTERISTICS.
Transmission code is USASCII-7. The seven data bits are transmitted least significant bits first followed by one parity bit. For synchronous operation, character parity is odd. For asyhchronous operation, character parity is even. The adapter performs the parity generation and checking. One start bit and one stop bit frame the seven bits plus parity that make up the character; therefore, for each character of data, 10 bits are sent down the line.

A block check code (BCC) is normally transmitted following all messages that contain a start code. The BCC is formed by taking the modulo two sum on each data bit following a start code, up to and including an end code, and is transmitted after the end code with appropriate character parity.

Both EBCDIC and USASCII-8 codes are capable of being received from memory by the line adapters. EBCDIC code must be translated to USASCII-8 by the control. USASCII-8 is translated to USASCII-7 in the adapter by ignoring the high order bit. On input, USASCII-7 is translated to USASCII-8 in the adapter and then translated to EBCDIC by the control if requested by a variant in the I/O descriptor.

The adapter assembles one character of data before requesting service from the I/O controller.

The following code sensitivity is provided by the adapter's.

Function	Code
Start	SOH, STX
End	ETX, ETB
Positive response	ACK, ENQ, BEL, ETX, ETB
Negative response	NAK, EOT
Enquiry	ENQ
Sync	SYNC

Two SYNC codes must be received before a synchronous adapter is sensitive to any other codes. SYNC codes are not stored in main memory. After the receipt of the last leading SYNC code, an adapter stores in memory all subsequent codes, except the SYNC character, up to but not including a block check character. SYNC characters are not included in a BCC sum. At least four SYNC codes must be transmitted prior to data. Four SYNC codes are automatically generated by the I/O controller prior to transmittal of data.

A control code listed under the classifications of positive responses and negative responses is also defined to be an ending code if it is not preceded by a code classified as a start code.

The control characters ETX and ETB are not considered to be a positive response unless followed by a correct block check character (BCC).

The negative response code EOT is a response code on a read only and never on a write. To enable an on-hook condition at a remote site, the adapter must receive a message such as DLE, EOT, ENQ. To enable an on-hook condition at the central site, the adapter must receive an I/O descriptor with the disconnect variant.

The receipt of an ENQ code or a ringing condition is recognized by the adapter when the adapter is in a test and wait state. It is not remembered or recognized in any other state.

A variant in the I/O descriptor is provided to ignore EOT as an ending code for read. This variant allows the controller and the system to which it is attached to act as a terminal and receive a polling sequence from other systems. No hardware recognition of addresses is provided, however.

The SLC I/O controller and line adapters are designed to operate over the communications facilities defined in table 7-3 and table 7-4.

Table .7-3

Transmission Facility Classification

Designation	Bandwidth	Transmission Rate	Remarks
Narrowband	Variable - generally up to 300 Hz	15.0-300 bits/sec	Generally private line except for TELEX and TWX services
Voiceband	Nominal 4 kHz	600/1200/1800/2000/2400 bits/sec and higher with WE or COAM modems	Private and dial lines
Wideband	48 kHZ and up	40.8 K bits/sec and higher	Generally private line except DATA-PHONE 50 service from the Bell System

Table .7-4

Conditioning

Voiceband	Conditioning	Delay Distortion	Band
3002	None	1750 us	800-2600 Hz
3002	C1	1000 us	1000-2400 Hz
		1750 us	800-2600 Hz
3002	C2	500 us	1000-2600 Hz
		1500 us	600-2600 Hz
		3000 us	500-2800 Hz
3002	C4	300 us	1000-2600 Hz
		1500 us	600-3000 Hz

DATA COMMUNICATIONS READ DESCRIPTOR.
Data is read from the remote device in ascending memory locations beginning with the location specified by the A address. Reading is continued until a control code denoting the end of text is detected. In no case is the data stored in the ending location specified by the B address. The complete message read must be terminated by a control code. (See figure .7-36.)

VVV	TTT	W	RR	D	CC	RRRRRRRR	UUUU	A	B
3 BITS	3 BITS	1	2	1	2	8 BITS	4 BITS	24 BITS	24 BITS

Figure 7-36. Data Communications
Read Descriptor

Referring to figure 7-36:

a. VVV = 000 - data communications read operation.
b. TTT = 000 - no translation.
c. TTT = 010 - USASCII-7 to EBCDIC translation.
d. W = 0 - do not disable time-out timer.
e. W = 1 - disable time-out timer.
f. D = 1 - ignore EOT as a response cycle.
g. CC = 00 - normal I/O chaining.
h. CC = 01 - I/O chaining if negative response is received.
i. CC = 10 - I/O chaining if positive response is received.
j. R = reserved.
k. UUUU = 0-15 - adapter number.
l. A = beginning address of data.
m. B = ending address of data + 1.

DATA COMMUNICATIONS WRITE DESCRIPTOR.
Data is written to the remote device from ascending memory locations beginning with the location specified by the A address and continuing until a control code which denotes end of text is detected or up to the location specified by the B address. (See figure 7-37.)

VVV	TTT	RRRRRRRRRRRRRR	UUUU	A	B
3 BITS	3 BITS	14 BITS	4 BITS	24 BITS	24 BITS

Figure 7-37. Data Communications
Write Descriptor

VVV is equal to 010 for the data communications write operation. Other variables are the same as for the read descriptor.

DATA COMMUNICATIONS BREAK DESCRIPTOR.
Send a break signal to the remote terminal or go on hook as specified by the variants in the I/O descriptors. (See figure 7-38.)

VVV	RRR	T	RRRRRRRRRRRRR	UUUU
3 BITS	3 BITS	1	13 BITS	4 BITS

Figure 7-38. Data Communications Break Descriptor

Referring to figure 7-38:

a. VVV = 111 - data communications break.
b. T = 0 - send break and do not disconnect.
c. T = 1 - disconnect and do not break.
d. UUUU = 0-15 - adapter number.
e. R = reserved.
f. D = address of result status area.

DATA COMMUNICATIONS TEST DESCRIPTOR.
Test the adapter identification and for a received ENQ, and return a result descriptor which contains the adapter I/O. (See figure 7-39.)

VVV	T	RRRRRRRRRRRRRRRR	UUUU
3 BITS	1	16 BITS	4 BITS

Figure 7-39. Data Communications Test Descriptor

Referring to figure 7-39:

a. VVV = 100 - test the I/O controller.

b. VVV = 111 - test the I/O controller, return a result descriptor, and go idle.

c. T = 1 - wait until ENQ is received before completing I/O.

d. R = reserved.

e. D = address of result status area.

DATA COMMUNICATIONS RESULT DESCRIPTOR.
The data communications result descriptor is as follows:

a. Bit 0 - I/O complete.

b. Bit 1 - exception conditions (any bit on in bits 3-16).

c. Bit 2 - not ready.

d. Bit 3 - parity error - character or BCC.

e. Bit 4 - memory access error.

f. Bit 5 - memory parity error.

g. Bit 6 - time-out.

h. Bit 7 - break.

i. Bit 8 - ending control code expected but not received.

j. Bit 9 - chaining terminated.

k. Bit 10 - reserved.

l. Bit 11 - loss of clear to send.

m. Bit 12 - carrier loss.

n. Bit 13 - reserved.

o. Bit 14 - off hook (dial%

p. Bit 15 - ringing or enquiry received.

q. Bit 16 - reserved.

r. Bits 17-23 - adapter ID (test I/O only).

   1) 1000000 = adapter not present.
   2) 10nnnnn = leased or direct connect.
   3) 11nnnnn = switched line.

NOTE

The variable nnnnn is defined as:

00010 - standard line adapter
01000 - Teletype adapter
00100 - ACU adapter

DISK PACK RESULT DESCRIPTOR.
At the completion of a disk pack I/O operation, the following result descriptor information is returned:

a. Bit 0 - operation complete.

b. Bit 1 - exception condition.

c. Bit 2 - not ready.

d. Bit 3 - read data parity error; write data parity error.

e. Bit 4 - memory access error.

f. Bit 5 - memory parity.

g. Bit 6 - write lockout operation not performed.

h. Bit 7 - transmission parity error; read data error corrected.

i. Bit 8 - overrun.

j. Bits 7-9:

1) 000 not present.
2) 001 20 surface - 203 track.
3) 010 20 surface - 406 track.

k. Bit 10 - address parity error or verify error.

l. Bit 11 - seek timeout.

m. Bit 12 - seek status.

n. Bit 13 - control no. 0...1.

o. Bit 14 - seeking; seek initiated flag; wrong cylinder.

p. Bit 15 - segment address error.

q. Bit 16 - reserved.

r. Bits 17-23 - controller ID = 0011110 (test I/O only).

DISK CARTRIDGE CONTROLLER.
This I/O controller provides for the attachment of disk cartridge drives to the system (see figure 7-26).

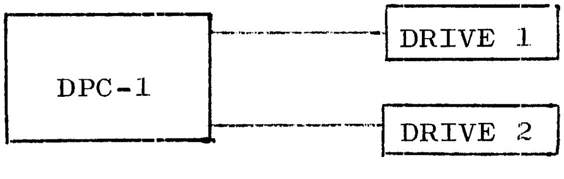


Figure 7-26. Disk Cartridge Controller

APPENDIX A

REGISTER SUMMARY

MICRO-INSTRUCTION CONTROL REGISTERS.

Mnemonic	Width	Function	Usage	Remarks
A	19	Address reg.	Source/dest.	32K micro-inst. (max memory on B 1720 Series) 256K micro-inst. on B 1720 Series
M	16	Micro-inst.	Source/dest.	
TAS	24	Top of stack	Source/dest.	Pseudo
TOPM	4	Top ctl. mem.	Source/dest.	Not avl. on B 1710 Series
MBR	24	Mem. base reg.	Source/dest.	Not avl. on B 1710 Series

MAIN MEMORY CONTROL REGISTERS.

Mnemonic	Width	Function	Usage	Remarks
BR	24	Base reg.	Source/dest.	
LR	24	Limit reg.	Source/dest.	
FA	24	Field Adr.	Source/dest.	B 1700 64 KB max.
FL	16	Field lgth.	Source/dest.	FLC/D/E/F
FU	4	Field unit	Source/dest.	Part of FB
CP	8	Arith. ctl.	Source/dest.	CYF, CPU, CPL
CYF	1	Carry flag	Source/dest.	
CPU	2	Arith. unit	Dest.	00 = binary 01 = 4-bit 11 = 8-bit (Not avl. on B 1710 Series)
CPL	5	Data length	Source/dest.	Max. val. 24
MAXS	24	Main mem. size	Source	

Mnemonic	Width	Function	Usage	Remarks
MAXM	24	Control mem. size	Source	
FLCN	4	Fld. lgth. cond.	Source/dest.	

INTERRUPT CONTROL REGISTERS.

Mnemonic	Width	Function	Usage	Remarks
CC	4	Intr. cond.	Source/dest.	0 = console intr. 1 = I/O ser. rq. 2 = Clock intr. 3 = State flag
CD	4	Intr. cond.	Source/dest.	0 = Wrt. out bds. 1 = Rd. out bds. 2 = Out bds. ovride. 3 = Rd. par. er.
INCN	4	Intr. cond.	Source/dest.	0 = Disp. lkout. 1 = Disp. intr. 2 = Priority disp. 3 = Missing port (Not aval. on B 1710 Series)

GENERAL PURPOSE REGISTERS.

Mnemonic	Width	Function	Usage	Remarks
X	24		Source/dest.	Gen. and inp. to funct. box
Y	24		Source/dest.	Gen. and inp. to funct. box
T	24		Source/dest.	Gen.
L	24		Source/dest.	Gen. and used by DISPATCH
CA	4		Source/dest.	
CB	4		Source/dest.	
FT	4		Source/dest.	Gen. - part of FB

INPUTS TO 24-BIT FUNCTION BOX.

Mnemonic	Width	Function	Usage	Remarks
X	24	Input	Source/dest.	
Y	24	Input	Source/dest.	
CYF	1	Carry flag	Source/dest.	
CPU	2	Arith. unit	Dest.	
CPL	5	Data length	Source/dest.	

OUTPUTS FROM 24-BIT FUNCTION BOX.

Mnemonic	Width	Function	Usage	Remarks
BICN	4	Binary cond.	Source	
XYCN	4	X and Y cond.	Source	
XYST	4	X and Y states	Source	
CYL	1	Carry out level	Source	
CYD	1	Borrow out level	Source	

AVAILABLE FROM 24-BIT FUNCTION BOX.

Mnemonic	Width	Function	Usage	Remarks
SUM	24	X + Y	Source	AND CYF
DIFF	24	X - Y	Source	AND CYD
XORY	24	X OR Y	Source	
XEOY	24	X EXOR Y	Source	
XANY	24	X AND Y	Source	
CMPX	24	Compl. X	Source	AND CPL
CMPY	24	Compl. Y	Source	AND CPL
MSKX	24	Mask X	Source	AND CPL
MSKY	24	Mask Y	Source	AND CPL

MISCELLANEOUS REGISTERS.

Mnemonic	Width	Function	Usage	Remarks
U	16	Cassette input	Source	
NULL	24		Source	Contains zeros
DATA	24	I/O data	Source/dest.	
CMND	24	I/O command	Dest.	

APPENDIX B

HEXADECIMAL-DECIMAL

CONVERSION TABLE

The table in this appendix provides for direct conversion of decimal and hexadecimal numbers in the ranges:

Hexadecimal	Decimal
000 to FFF	0 to 4095

For numbers outside the range of the table, add the following values to the table figure's.

Hexadecimal	Decimal
1000	4096
2000	8192
3000	12288
4000	16384
5000	20484
6000	24576
7000	28672
8000	32768
9000	36864
A000	40960
B000	45056
C000	49152
D000	53248
E000	57344
F000	61440

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
010	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
020	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
030	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
040	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
050	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
060	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
070	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
080	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
090	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
0A0	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
0B0	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
0C0	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
0D0	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
0E0	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
0F0	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255
100	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271
110	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287
120	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303
130	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319
140	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335
150	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351
160	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367
170	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383
180	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399
190	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415
1A0	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431
1B0	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447
1C0	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463
1D0	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479
1E0	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495
1F0	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
200	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527
210	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543
220	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559
230	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575
240	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591
250	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607
260	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623
270	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639
280	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654	655
290	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671
2A0	672	673	674	675	676	677	678	679	680	681	682	683	684	685	686	687
2B0	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702	703
2C0	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719
2D0	720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735
2E0	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751
2F0	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767
300	768	769	770	771	772	773	774	775	776	777	778	779	780	781	782	783
310	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799
320	800	801	802	803	804	805	806	807	808	809	810	811	812	813	814	815
330	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831
340	832	833	834	835	836	837	838	839	840	841	842	843	844	845	846	847
350	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863
360	864	865	866	867	868	869	870	871	872	873	874	875	876	877	878	879
370	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895
380	896	897	898	899	900	901	902	903	904	905	906	907	908	909	910	911
390	912	913	914	915	916	917	918	919	920	921	922	923	924	925	926	927
3A0	928	929	930	931	932	933	934	935	936	937	938	939	940	941	942	943
3B0	944	945	946	947	948	949	950	951	952	953	954	955	956	957	958	959
3C0	960	961	962	963	964	965	966	967	968	969	970	971	972	973	974	975
3D0	976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991
3E0	992	993	994	995	996	997	998	999	1000	1001	1002	1003	1004	1005	1006	1007
3F0	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
400	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039
410	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1050	1051	1052	1053	1054	1055
420	1056	1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1070	1071
430	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081	1082	1083	1084	1085	1086	1087
440	1088	1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103
450	1104	1105	1106	1107	1108	1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119
460	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129	1130	1131	1132	1133	1134	1135
470	1136	1137	1138	1139	1140	1141	1142	1143	1144	1145	1146	1147	1148	1149	1150	1151
480	1152	1153	1154	1155	1156	1157	1158	1159	1160	1161	1162	1163	1164	1165	1166	1167
490	1168	1169	1170	1171	1172	1173	1174	1175	1176	1177	1178	1179	1180	1181	1182	1183
4A0	1184	1185	1186	1187	1188	1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199
4B0	1200	1201	1202	1203	1204	1205	1206	1207	1208	1209	1210	1211	1212	1213	1214	1215
4C0	1216	1217	1218	1219	1220	1221	1222	1223	1224	1225	1226	1227	1228	1229	1230	1231
4D0	1232	1233	1234	1235	1236	1237	1238	1239	1240	1241	1242	1243	1244	1245	1246	1247
4E0	1248	1249	1250	1251	1252	1253	1254	1255	1256	1257	1258	1259	1260	1261	1262	1263
4F0	1264	1265	1266	1267	1268	1269	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279
500	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289	1290	1291	1292	1293	1294	1295
510	1296	1297	1298	1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	1310	1311
520	1312	1313	1314	1315	1316	1317	1318	1319	1320	1321	1322	1323	1324	1325	1326	1327
530	1328	1329	1330	1331	1332	1333	1334	1335	1336	1337	1338	1339	1340	1341	1342	1343
540	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353	1354	1355	1356	1357	1358	1359
550	1360	1361	1362	1363	1364	1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375
560	1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	1386	1387	1388	1389	1390	1391
570	1392	1393	1394	1395	1396	1397	1398	1399	1400	1401	1402	1403	1404	1405	1406	1407
580	1408	1409	1410	1411	1412	1413	1414	1415	1416	1417	1418	1419	1420	1421	1422	1423
590	1424	1425	1426	1427	1428	1429	1430	1431	1432	1433	1434	1435	1436	1437	1438	1439
5A0	1440	1441	1442	1443	1444	1445	1446	1447	1448	1449	1450	1451	1452	1453	1454	1455
5B0	1456	1457	1458	1459	1460	1461	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471
5C0	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	1482	1483	1484	1485	1486	1487
5D0	1488	1489	1490	1491	1492	1493	1494	1495	1496	1497	1498	1499	1500	1501	1502	1503
5E0	1504	1505	1506	1507	1508	1509	1510	1511	1512	1513	1514	1515	1516	1517	1518	1519
5F0	1520	1521	1522	1523	1524	1525	1526	1527	1528	1529	1530	1531	1532	1533	1534	1535

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
600	1536	1537	1538	1539	1540	1541	1542	1543	1544	1545	1546	1547	1548	1549	1550	1551
610	1552	1553	1554	1555	1556	1557	1558	1559	1560	1561	1562	1563	1564	1565	1566	1567
620	1568	1569	1570	1571	1572	1573	1574	1575	1576	1577	1578	1579	1580	1581	1582	1583
630	1584	1585	1586	1587	1588	1589	1590	1591	1592	1593	1594	1595	1596	1597	1598	1599
640	1600	1601	1602	1603	1604	1605	1606	1607	1608	1609	1610	1611	1612	1613	1614	1615
650	1616	1617	1618	1619	1620	1621	1622	1623	1624	1625	1626	1627	1628	1629	1630	1631
660	1632	1633	1634	1635	1636	1637	1638	1639	1640	1641	1642	1643	1644	1645	1646	1647
670	1648	1649	1650	1651	1652	1653	1654	1655	1656	1657	1658	1659	1660	1661	1662	1663
680	1664	1665	1666	1667	1668	1669	1670	1671	1672	1673	1674	1675	1676	1677	1678	1679
690	1680	1681	1682	1683	1684	1685	1686	1687	1688	1689	1690	1691	1692	1693	1694	1695
6A0	1696	1697	1698	1699	1700	1701	1702	1703	1704	1705	1706	1707	1708	1709	1710	1711
6B0	1712	1713	1714	1715	1716	1717	1718	1719	1720	1721	1722	1723	1724	1725	1726	1727
6C0	1728	1729	1730	1731	1732	1733	1734	1735	1736	1737	1738	1739	1740	1741	1742	1743
6D0	1744	1745	1746	1747	1748	1749	1750	1751	1752	1753	1754	1755	1756	1757	1758	1759
6E0	1760	1761	1762	1763	1764	1765	1766	1767	1768	1769	1770	1771	1772	1773	1774	1775
6F0	1776	1777	1778	1779	1780	1781	1782	1783	1784	1785	1786	1787	1788	1789	1790	1791
700	1792	1793	1794	1795	1796	1797	1798	1799	1800	1801	1802	1803	1804	1805	1806	1807
710	1808	1809	1810	1811	1812	1813	1814	1815	1816	1817	1818	1819	1820	1821	1822	1823
720	1824	1825	1826	1827	1828	1829	1830	1831	1832	1833	1834	1835	1836	1837	1838	1839
730	1840	1841	1842	1843	1844	1845	1846	1847	1848	1849	1850	1851	1852	1853	1854	1855
740	1856	1857	1858	1859	1860	1861	1862	1863	1864	1865	1866	1867	1868	1869	1870	1871
750	1872	1873	1874	1875	1876	1877	1878	1879	1880	1881	1882	1883	1884	1885	1886	1887
760	1888	1889	1890	1891	1892	1893	1894	1895	1896	1897	1898	1899	1900	1901	1902	1903
770	1904	1905	1906	1907	1908	1909	1910	1911	1912	1913	1914	1915	1916	1917	1918	1919
780	1920	1921	1922	1923	1924	1925	1926	1927	1928	1929	1930	1931	1932	1933	1934	1935
790	1936	1937	1938	1939	1940	1941	1942	1943	1944	1945	1946	1947	1948	1949	1950	1951
7A0	1952	1953	1954	1955	1956	1957	1958	1959	1960	1961	1962	1963	1964	1965	1966	1967
7B0	1968	1969	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979	1980	1981	1982	1983
7C0	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
7D0	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015
7E0	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031
7F0	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
800	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063
810	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079
820	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095
830	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111
840	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127
850	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143
860	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159
870	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175
880	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191
890	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207
8A0	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223
8B0	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239
8C0	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255
8D0	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271
8E0	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287
8F0	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303
900	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319
910	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335
920	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351
930	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367
940	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383
950	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399
960	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415
970	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431
980	2432	2433	2434	2435	2436	2437	2438	2439	2440	2441	2442	2443	2444	2445	2446	2447
990	2448	2449	2450	2451	2452	2453	2454	2455	2456	2457	2458	2459	2460	2461	2462	2463
9A0	2464	2465	2466	2467	2468	2469	2470	2471	2472	2473	2474	2475	2476	2477	2478	2479
9B0	2480	2481	2482	2483	2484	2485	2486	2487	2488	2489	2490	2491	2492	2493	2494	2495
9C0	2496	2497	2498	2499	2500	2501	2502	2503	2504	2505	2506	2507	2508	2509	2510	2511
9D0	2512	2513	2514	2515	2516	2517	2518	2519	2520	2521	2522	2523	2524	2525	2526	2527
9E0	2528	2529	2530	2531	2532	2533	2534	2535	2536	2537	2538	2539	2540	2541	2542	2543
9F0	2544	2545	2546	2547	2548	2549	2550	2551	2552	2553	2554	2555	2556	2557	2558	2559

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A00	2560	2561	2562	2563	2564	2565	2566	2567	2568	2569	2570	2571	2572	2573	2574	2575
A10	2576	2577	2578	2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591
A20	2592	2593	2594	2595	2596	2597	2598	2599	2600	2601	2602	2603	2604	2605	2606	2607
A30	2608	2609	2610	2611	2612	2613	2614	2615	2616	2617	2618	2619	2620	2621	2622	2623
A40	2624	2625	2626	2627	2628	2629	2630	2631	2632	2633	2634	2635	2636	2637	2638	2639
A50	2640	2641	2642	2643	2644	2645	2646	2647	2648	2649	2650	2651	2652	2653	2654	2655
A60	2656	2657	2658	2659	2660	2661	2662	2663	2664	2665	2666	2667	2668	2669	2670	2671
A70	2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684	2685	2686	2687
A80	2688	2689	2690	2691	2692	2693	2694	2695	2696	2697	2698	2699	2700	2701	2702	2703
A90	2704	2705	2706	2707	2708	2709	2710	2711	2712	2713	2714	2715	2716	2717	2718	2719
AA0	2720	2721	2722	2723	2724	2725	2726	2727	2728	2729	2730	2731	2732	2733	2734	2735
AB0	2736	2737	2738	2739	2740	2741	2742	2743	2744	2745	2746	2747	2748	2749	2750	2751
AC0	2752	2753	2754	2755	2756	2757	2758	2759	2760	2761	2762	2763	2764	2765	2766	2767
AD0	2768	2769	2770	2771	2772	2773	2774	2775	2776	2777	2778	2779	2780	2781	2782	2783
AE0	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	2795	2796	2797	2798	2799
AF0	2800	2801	2802	2803	2804	2805	2806	2807	2808	2809	2810	2811	2812	2813	2814	2815
B00	2816	2817	2818	2819	2820	2821	2822	2823	2824	2825	2826	2827	2828	2829	2830	2831
B10	2832	2833	2834	2835	2836	2837	2838	2839	2840	2841	2842	2843	2844	2845	2846	2847
B20	2848	2849	2850	2851	2852	2853	2854	2855	2856	2857	2858	2859	2860	2861	2862	2863
B30	2864	2865	2866	2867	2868	2869	2870	2871	2872	2873	2874	2875	2876	2877	2878	2879
B40	2880	2881	2882	2883	2884	2885	2886	2887	2888	2889	2890	2891	2892	2893	2894	2895
B50	2896	2897	2898	2899	2900	2901	2902	2903	2904	2905	2906	2907	2908	2909	2910	2911
B60	2912	2913	2914	2915	2916	2917	2918	2919	2920	2921	2922	2923	2924	2925	2926	2927
B70	2928	2929	2930	2931	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941	2942	2943
B80	2944	2945	2946	2947	2948	2949	2950	2951	2952	2953	2954	2955	2956	2957	2958	2959
B90	2960	2961	2962	2963	2964	2965	2966	2967	2968	2969	2970	2971	2972	2973	2974	2975
BA0	2976	2977	2978	2979	2980	2981	2982	2983	2984	2985	2986	2987	2988	2989	2990	2991
BB0	2992	2993	2994	2995	2996	2997	2998	2999	3000	3001	3002	3003	3004	3005	3006	3007
BC0	3008	3009	3010	3011	3012	3013	3014	3015	3016	3017	3018	3019	3020	3021	3022	3023
BD0	3024	3025	3026	3027	3028	3029	3030	3031	3032	3033	3034	3035	3036	3037	3038	3039
BE0	3040	3041	3042	3043	3044	3045	3046	3047	3048	3049	3050	3051	3052	3053	3054	3055
BF0	3056	3057	3058	3059	3060	3061	3062	3063	3064	3065	3066	3067	3068	3069	3070	3071

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
C00	3072	3073	3074	3075	3076	3077	3078	3079	3080	3081	3082	3083	3084	3085	3086	3087
C10	3088	3089	3090	3091	3092	3093	3094	3095	3096	3097	3098	3099	3100	3101	3102	3103
C20	3104	3105	3106	3107	3108	3109	3110	3111	3112	3113	3114	3115	3116	3117	3118	3119
C30	3120	3121	3122	3123	3124	3125	3126	3127	3128	3129	3130	3131	3132	3133	3134	3135
C40	3136	3137	3138	3139	3140	3141	3142	3143	3144	3145	3146	3147	3148	3149	3150	3151
C50	3152	3153	3154	3155	3156	3157	3158	3159	3160	3161	3162	3163	3164	3165	3166	3167
C60	3168	3169	3170	3171	3172	3173	3174	3175	3176	3177	3178	3179	3180	3181	3182	3183
C70	3184	3185	3186	3187	3188	3189	3190	3191	3192	3193	3194	3195	3196	3197	3198	3199
C80	3200	3201	3202	3203	3204	3205	3206	3207	3208	3209	3210	3211	3212	3213	3214	3215
C90	3216	3217	3218	3219	3220	3221	3222	3223	3224	3225	3226	3227	3228	3229	3230	3231
CA0	3232	3233	3234	3235	3236	3237	3238	3239	3240	3241	3242	3243	3244	3245	3246	3247
CB0	3248	3249	3250	3251	3252	3253	3254	3255	3256	3257	3258	3259	3260	3261	3262	3263
CC0	3264	3265	3266	3267	3268	3269	3270	3271	3272	3273	3274	3275	3276	3277	3278	3279
CD0	3280	3281	3282	3283	3284	3285	3286	3287	3288	3289	3290	3291	3292	3293	3294	3295
CE0	3296	3297	3298	3299	3300	3301	3302	3303	3304	3305	3306	3307	3308	3309	3310	3311
CF0	3312	3313	3314	3315	3316	3317	3318	3319	3320	3321	3322	3323	3324	3325	3326	3327
D00	3328	3329	3330	3331	3332	3333	3334	3335	3336	3337	3338	3339	3340	3341	3342	3343
D10	3344	3345	3346	3347	3348	3349	3350	3351	3352	3353	3354	3355	3356	3357	3358	3359
D20	3360	3361	3362	3363	3364	3365	3366	3367	3368	3369	3370	3371	3372	3373	3374	3375
D30	3376	3377	3378	3379	3380	3381	3382	3383	3384	3385	3386	3387	3388	3389	3390	3391
D40	3392	3393	3394	3395	3396	3397	3398	3399	3400	3401	3402	3403	3404	3405	3406	3407
D50	3408	3409	3410	3411	3412	3413	3414	3415	3416	3417	3418	3419	3420	3421	3422	3423
D60	3424	3425	3426	3427	3428	3429	3430	3431	3432	3433	3434	3435	3436	3437	3438	3439
D70	3440	3441	3442	3443	3444	3445	3446	3447	3448	3449	3450	3451	3452	3453	3454	3455
D80	3456	3457	3458	3459	3460	3461	3462	3463	3464	3465	3466	3467	3468	3469	3470	3471
D90	3472	3473	3474	3475	3476	3477	3478	3479	3480	3481	3482	3483	3484	3485	3486	3487
DA0	3488	3489	3490	3491	3492	3493	3494	3495	3496	3497	3498	3499	3500	3501	3502	3503
DB0	3504	3505	3506	3507	3508	3509	3510	3511	3512	3513	3514	3515	3516	3517	3518	3519
DC0	3520	3521	3522	3523	3524	3525	3526	3527	3528	3529	3530	3531	3532	3533	3534	3535
DD0	3536	3537	3538	3539	3540	3541	3542	3543	3544	3545	3546	3547	3548	3549	3550	3551
DE0	3552	3553	3554	3555	3556	3557	3558	3559	3560	3561	3562	3563	3564	3565	3566	3567
DF0	3568	3569	3570	3571	3572	3573	3574	3575	3576	3577	3578	3579	3580	3581	3582	3583

APPENDIX B (cont'd)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
E00	3584	3585	3586	3587	3588	3589	3590	3591	3592	3593	3594	3595	3596	3597	3598	3599
E10	3600	3601	3602	3603	3604	3605	3606	3607	3608	3609	3610	3611	3612	3613	3614	3615
E20	3616	3617	3618	3619	3620	3621	3622	3623	3624	3625	3626	3627	3628	3629	3630	3631
E30	3632	3633	3634	3635	3636	3637	3638	3639	3640	3641	3642	3643	3644	3645	3646	3647
E40	3648	3649	3650	3651	3652	3653	3654	3655	3656	3657	3658	3659	3660	3661	3662	3663
E50	3664	3665	3666	3667	3668	3669	3670	3671	3672	3673	3674	3675	3676	3677	3678	3679
E60	3680	3681	3682	3683	3684	3685	3686	3687	3688	3689	3690	3691	3692	3693	3694	3695
E70	3696	3697	3698	3699	3700	3701	3702	3703	3704	3705	3706	3707	3708	3709	3710	3711
E80	3712	3713	3714	3715	3716	3717	3718	3719	3720	3721	3722	3723	3724	3725	3726	3727
E90	3728	3729	3730	3731	3732	3733	3734	3735	3736	3737	3738	3739	3740	3741	3742	3743
EA0	3744	3745	3746	3747	3748	3749	3750	3751	3752	3753	3754	3755	3756	3757	3758	3759
EB0	3760	3761	3762	3763	3764	3765	3766	3767	3768	3769	3770	3771	3772	3773	3774	3775
EC0	3776	3777	3778	3779	3780	3781	3782	3783	3784	3785	3786	3787	3788	3789	3790	3791
ED0	3792	3793	3794	3795	3796	3797	3798	3799	3800	3801	3802	3803	3804	3805	3806	3807
EE0	3808	3809	3810	3811	3812	3813	3814	3815	3816	3817	3818	3819	3820	3821	3822	3823
EF0	3824	3825	3826	3827	3828	3829	3830	3831	3832	3833	3834	3835	3836	3837	3838	3839
F00	3840	3841	3842	3843	3844	3845	3846	3847	3848	3849	3850	3851	3852	3853	3854	3855
F10	3856	3857	3858	3859	3860	3861	3862	3863	3864	3865	3866	3867	3868	3869	3870	3871
F20	3872	3873	3874	3875	3876	3877	3878	3879	3880	3881	3882	3883	3884	3885	3886	3887
F30	3888	3889	3890	3891	3892	3893	3894	3895	3896	3897	3898	3899	3900	3901	3902	3903
F40	3904	3905	3906	3907	3908	3909	3910	3911	3912	3913	3914	3915	3916	3917	3918	3919
F50	3920	3921	3922	3923	3924	3925	3926	3927	3928	3929	3930	3931	3932	3933	3934	3935
F60	3936	3937	3938	3939	3940	3941	3942	3943	3944	3945	3946	3947	3948	3949	3950	3951
F70	3952	3953	3954	3955	3956	3957	3958	3959	3960	3961	3962	3963	3964	3965	3966	3967
F80	3968	3969	3970	3971	3972	3973	3974	3975	3976	3977	3978	3979	3980	3981	3982	3983
F90	3984	3985	3986	3987	3988	3989	3990	3991	3992	3993	3994	3995	3996	3997	3998	3999
FA0	4000	4001	4002	4003	4004	4005	4006	4007	4008	4009	4010	4011	4012	4013	4014	4015
FB0	4016	4017	4018	4019	4020	4021	4022	4023	4024	4025	4026	4027	4028	4029	4030	4031
FC0	4032	4033	4034	4035	4036	4037	4038	4039	4040	4041	4042	4043	4044	4045	4046	4047
FD0	4048	4049	4050	4051	4052	4053	4054	4055	4056	4057	4058	4059	4060	4061	4062	4063
FE0	4064	4065	4066	4067	4068	4069	4070	4071	4072	4073	4074	4075	4076	4077	4078	4079
FF0	4080	4081	4082	4083	4084	4085	4086	4087	4088	4089	4090	4091	4092	4093	4094	4095

APPENDIX C

SYSTEM COMPARISON

	B 1712/B 1714	B 1726
MSM	No	Yes
TOPM	No	Yes
MBR	No	Yes
INCN	No	Yes
A stack	16 elements	32 elements
BR & LR	MCP checking only	CD reg. set out of bounds
CPU	00 = binary 01 = 4 bit	00 = binary 01 = 4 bit 11 = 8 bit
MAXM	0	0, 4, 8
DATA/CMND	Utilized	Utilized

XYST/conditions	Least-significant unit of X
	ANY-interrupt
	Y = 0
	X = 0
		The ANY-interrupt bit has several more inputs.

Swap memory micro	No	Yes
Clear register micro	No	Yes
Load/Store F from/into doublepad word micro	No	Yes
Overlay M-string micro	No	Yes
Control memory	No	Yes
Port interchange	No	Yes
Buffered I/O controllers	Yes	Yes

	B 1712/B 1714	B 1726
Processor clock rate	2 MHz/4 MHz	6 MHz
Read cycle	4 clocks	4 clocks
Write cycle	6 clocks	6 clocks
Main memory minimum	16 KB	24 KB
Main memory maximum	40 KB/65 KB	96 KB

NOTE

6 MHz = 167 nanoseconds (B 1726)
4 MHz = 250 nanoseconds (B 1714)
2 MHz = 500 nanoseconds (B 1712)
1 KB = 1024 bytes
1 byte = 8 bits
1 micro-instruction = 16 bits

APPENDIX D

CONVERSIONS

POWERS OF TWO, BASE 10.

2**n	n	2**n	n
1	0	1 048 576	20
2	1	2 097 152	21
4	2	4 194 304	22
8	3	8 388 608	23
16	4	16 777 216	24
32	5	33 554 432	25
64	6	67 108 864	26
128	7	134 217 728	27
256	8	268 435 456	28
512	9	536 870 912	29
1 024	10	1 073 741 824	30
2 048	11	2 147 483 648	31
4 096	12	4 294 967 296	32
8 192	13	8 589 934 592	33
16 384	14	17 179 869 184	34
32 768	15	34 359 738 368	35
65 536	16	68 719 476 736	36
131 072	17	137 438 953 472	37
262 144	18	274 877 906 944	38
524 288	19	549 755 813 888	39

NOTE
2**n means 2 raised
to the power of n.

BINARY/HEXADECIMAL/DECIMAL.

Binary	Hexadecimal	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13

Binary	Hexadecimal	Decimal
1110	E	14
1111	F	15

APPENDIX E

MICRO-INSTRUCTIONS

MICROMNEMONICS	OP CODE (HEXADECIMAL)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REGISTER MOVE	1nnn	0	0	0	1	SOURCE REG. ROW				SOURCE REG. COL.		DESTINATION REG. COL.		DESTINATION REG. ROW			
SCRATCHPAD MOVE	2nnn	0	0	1	0	SOURCE/DESTINATION REG. ROW				SOURCE/DEST REG. COL.		REG. TO S.P./S.P. TO REG	A/B	SCRATCHPAD WORD ADDRESS			
FOUR-BIT MANIPULATE	3nnn	0	0	1	1	AFFECTED REGISTER ROW				REG. COL.	MANIPULATE VARIANTS			FOUR-BIT MANIPULATE LITERAL			
BIT TEST REL BR ON FALSE	4nnn	0	1	0	0	SOURCE REG. (FOUR-BIT) ROW				REG. COL.	TEST BIT NUMBER		+/-	RELATIVE BRANCH DISPLACEMENT VALUE			
BIT TEST REL BR ON TRUE	5nnn	0	1	0	1	SOURCE REG. (FOUR-BIT) ROW				REG. COL.	TEST BIT NUMBER		+/-	RELATIVE BRANCH DISPLACEMENT VALUE			
SKIP WHEN	6nnn	0	1	1	0	SOURCE REG. (FOUR-BIT) ROW				REG. COL.	SKIP TEST VARIANTS			FOUR-BIT TEST MASK			
READ/WRITE MEMORY	7nnn	0	1	1	1	R/W	COUNT FA/FL VARIANTS			DATA REG. (XYTL)		FOR/REV	MEMORY FIELD LENGTH·				
MOVE EIGHT-BIT LITERAL	8nnn	1	0	0	0	DESTINATION REG. ROW COL 2 ASSUMED				EIGHT-BIT LITERAL							
MOVE 24-BIT LITERAL	9nnn	1	0	0	1	DESTINATION REG. ROW COL 2 ASSUMED				FIRST EIGHT BITS OF LITERAL							
SHIFT/ROTATE T REGISTER	Annn	1	0	1	0	DESTINATION REG. ROW				DEST REG. COL.		SFT./ROT.	SHIFT/ROTATE COUNT (1-24)				
EXTRACT FROM T REGISTER	Bnnn	1	0	1	1	RIGHT BIT POINTER (1-24) FOR EXTRACTION FIELD OP.					DEST. REG. CODE (XYTL)		WIDTH OF EXTRACTION FIELD (1-24)				
BRANCH REL. FORWARD	Cnnn	1	1	0	0	RELATIVE DISPLACEMENT MAGNITUDE											
BRANCH REL. REVERSE	Dnnn	1	1	0	1	RELATIVE DISPLACEMENT MAGNITUDE											
CALL REL. FORWARD	Ennn	1	1	1	0	RELATIVE CALLED ADDRESS MAGNITUDE											
CALL REL. REVERSE	Fnnn	1	1	1	1	RELATIVE CALLED ADDRESS MAGNITUDE											
SWAP MEMORY	02nn	0	0	0	0	0	0	1	0	DEST. GEN. PURPOSE REG. (XYTL)		FOR./REV.	MEMORY FIELD LENGTH				
CLEAR REGISTERS	03nn	0	0	0	0	0	0	1	1	L REG.	T REG.	Y REG.	X REG.	FA REG.	FL REG.	FU REG.	CP REG.
SHIFT/ROTATE X OR Y	04nn	0	0	0	0	0	1	0	0	SFT./ROT.	LFT./RT.	X/Y	SHIFT/ROTATE COUNT (1-24)				
SHIFT/ROTATE X AND Y	05nn	0	0	0	0	0	1	0	1	SFT. • /ROT.	LFT./RT.	RES.	SHIFT/ROTATE • COUNT (1-24)				
COUNT FA AND FL	06nn	0	0	0	0	0	1	1	0	COUNT VARIANTS			LITERAL MAGNITUDE				
EXCHANGE DOUBLEPAD WORD	07nn	0	0	0	0	0	1	1	1	DESTINATION 48-BIT SCRATCHPAD ADDR.				SOURCE 48-BIT SCRATCHPAD ADDR.			
SCRATCHPAD RELATE FA	08nn	0	0	0	0	1	0	0	0	RESERVED			+/-	A(LEFT) SCRATCHPAD WORD ADDRESS			
MONITOR	09nn	0	0	0	0	1	0	0	1	LITERAL OCCURRENCE IDENTIFIER							
CASSETTE CONTROL	002n	0	0	0	0	0	0	0	0	0	0	1	0	CASSETTE MANIPULATE VARIANTS			RES.
BIAS	003n	0	0	0	0	0	0	0	0	0	0	1	1	BIAS VARIANTS			TEST NOT/TEST
STORE F INTO * DOUBLEPAD WORD	004n	0	0	0	0	0	0	0	0	0	1	0	0	DESTINATION SCRATCHPAD WORD (48 BITS)			
LOAD F FROM * DOUBLEPAD WORD	005n	0	0	0	0	0	0	0	0	0	1	0	1	SOURCE SCRATCHPAD WORD (48 BITS)			
SET CYF	006n	0	0	0	0	0	0	0	0	0	1	0	1	CYF TO CYD	CYF TO CYL	CYF TO 1	CYF TO 0
HALT	0001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
OVERLAY * M-STRING	0002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
NORMALIZE X	0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
NO OPERATION	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

* NOT AVAILABLE ON B 1710 SYSTEMS

ONE BIT VARIANTS = 0/1

FOUR-BIT MANIPULATE (3nnn) VARIANTS

BITS 4-6	CONDITIONS
000	SET
001	AND
010	OR
011	EOR
100	INC
101	INC/TEST
110	DEC
111	DEC/TEST

SKIP WHEN (6nnn) SKIP TEST VARIANTS

BITS 4-6	CONDITIONS
000	ANY. SKIP
001	ALL. SKIP
010	EQU. SKIP
011	ALL CLR. SKIP
100	NOT ANY. SKIP
101	NOT ALL. SKIP
110	NOT EQU. SKIP
111	NOT ALL. CLR. SKIP

READ/WRITE MEMORY (7nnn) VARIANTS

BITS 6-7	CONDITIONS
00	X REG.
01	Y REG.
10	T REG.
11	L REG.

BITS 8-10	CONDITIONS
000	NOP
001	FA↑
010	FL↑
011	FA↑ FL↓
100	FA↓ FL↑
101	FA↓
110	FL↓
111	FA↓ FL↓

EXTRACT FROM T REGISTER (8nnn) VARIANTS

BITS 5-6	CONDITIONS
00	X REG.
01	Y REG.
10	T REG.
11	L REG.

SWAP MEMORY (02nn) VARIANTS

BITS 6-7	CONDITIONS
00	X REG.
01	Y REG.
10	T REG.
11	L REG.

CASSETTE CONTROL (002n) VARIANTS

BITS 3-1	CONDITIONS
000	START TAPE
001	STOP ON GAP
010	STOP ON X≠Y
011-111	RESERVED

COUNT FA AND FL (06nn) VARIANTS

BITS 5-7	CONDITIONS
000	NOP
001	FA↑
010	FL↑
011	FA↑ FL↓
100	FA↓ FL↑
101	FA↓
110	FL↓
111	FA↓ FL↓

DISPATCH (001n) VARIANTS

BITS 1-3	CONDITIONS
000	DISPATCH LOCK
001	DISPATCH WRITE
010	DISPATCH READ
011	DISPATCH RD & CLR
100	RESERVED
101	RESERVED
110	RESERVED
111	RESERVED

BIAS (003n) VARIANTS

BITS 3-1	CONDITIONS
000	FU
001	24 OR FL
010	24 OR SFL
011	24 OR FL OR SFL
100	NOP
101	24 OR CPL OR FL
110	NOP
111	24 OR CPL OR FL OR SFL

CC REGISTER

0 = CONSOLE INTR.
1 = I/O SERVICE REQ.
2 = CLOCK INTR (100 MS)
3 = STATE FLAG

CD REGISTER

0 = WRT/SWAP OUT OF BDS*
1 = READ OUT OF BDS*
2 = OUT OF BDS OVERRIDE*
3 = MEM. RD. PARITY ERR.

INCN REGISTER*

0 = PORT DISP. LOCKOUT
1 = PORT DISP. INTR.
2 = PORT PRIORITY INTR.
3 = MISSING CONTROLLER ON PORT OR CHANNEL

REGISTER COLUMN

REGISTER ROW	0	1	2	3
0	TA	FU	X	SUM
1	TB	FT	Y	CMPX
2	TC	FLC	T	CMPY
3	TD	FLD	L	XANY
4	TE	FLE	A(MAR)	XEOY
5	TF	FLF	M	MSKX
6	CA	BICN	BR	MSKY
7	CB	FLCN	LR	XORY
8	LA	TOPM*	FA	DIFF
9	LB	RES.	FB	MAXS
10	LC	RES.	FL	MAXM
11	LD	RES.	TAS	U
12	LE	XYCN	CP	MBR*
13	LF	XYST	MSMA*	DATA
14	CC	INCN*	READ	CMND
15	CD	CPU	WRIT	NULL

* NOT AVAILABLE ON B 1710 SYSTEMS

APPENDIX F

I/O RESULT DESCRIPTORS

RD BITS / UNIT	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	CONTROLLER * ID. (17-23)
CARD READER	VALIDITY CHECK	MEMORY ACCESS ERROR		READ CHECK												CTL. 1 0101010 CTL. 2 0010100
CARD PUNCH	PUNCH CHECK	MEMORY ACCESS ERROR	MEMORY PARITY ERROR													CTL. 0000100
PAPER TAPE READER	TAPE PARITY ERROR	MEMORY ACCESS ERROR		END OF TAPE	BEGINNING OF TAPE		SHORT REC. RD.	UNIT REWINDING								CTL. 10001110 CTL. 20001100
LINE PRINTER	PRINT CHECK		MEMORY PARITY ERROR	END OF PAGE	7-9 = CHAR. SET* 000 = 64 001 = 48 010 = 16 011 = 96 100 = 192			10-11 PRT. SPD.* 00 = 400/860 01 = 300 10 = 600 11 = 1100		PAPER * IN MOTION	MOTOR * ON TEST	NO. PRT. POS. 00 = 132 01 = 120 11 = 80				CTL. 0010110
CONSOLE PRINTER	KEYBOARD E. CANCEL		ATTP. TO EXECUTE END ADDR.		INPUT REQUEST											CTL. 0010110
READER SORTER	UNENCODED DOCUMENT	MEMORY ACCESS ERROR	CANNOT READ	AMOUNT FIELD ERROR	ON-US FIELD ERROR	TRANSIT FIELD ERROR	DOUBLE DOCUMENTS	TOO LATE TO READ	JAM	MISSORT	BATCH TICKET	FLOW STOPPED	EMPTY HOPPER, FULL STACKER			CTL. 0010100
MAGNETIC TAPE	TAPE PARITY ERROR. BUSY	MEMORY ACCESS ERROR	MEMORY PARITY ERROR	END OF TAPE	BEGINNING OF TAPE	WRITE LOCKOUT	END OF FILE	UNIT REWINDING	TIME-OUT (3 FT. BLANK TAPE)	CRC ERROR TRACK IN ERROR TRACK/DENSITY * 000 = 7T. 200 001 = 7T. 556 011 = 7T. 800 101 = 9T. 800 111 = 9T. 1600			000-111 (0-7)	LONG REC.	SHORT REC.	MTC1–0110010 MTC2–0110000 MTC3–0110100 MTC4–0110110 MTC5–0111000
MFCU	INV. CH. COL. 1	MEMORY ACCESS ERROR	MEMORY PARITY ERROR	READ CHECK	PUNCH CHECK	PRI. HOP. EMPTY	SEC. HOP. EMPTY									CTL. 0001000
DISK CARTRIDGE	READ PARITY ERROR	MEMORY ACCESS ERROR	MEMORY PARITY ERROR	WRITE LOCKOUT	TRACK 100 = 203T/2200 110 = 406/2200	OVERRUN	DENSITY		SEEK	SEEK * STATUS		SEEKING				CTL. 0011100
SINGLE LINE	PARITY ERROR	MEMORY ACCESS ERROR	MEMORY PARITY ERROR	TIME-OUT	BREAK DET.	END. CTL. CODE NOT RECEIVED	CHAINING TERM	LOSS OF CLR. TO SD.	CARRIER LOSS		OFF HOOK	RINGING OR ENQ.				SEE BELOW
DISK PACK	READ PARITY ERROR	MEMORY ACCESS ERROR	MEMORY PARITY ERROR	WRITE LOCKOUT	TRANS. P.E. 001 = 20SUR/203T 010 = 20SUR/406T	OVERRUN	TRK/DENS.*	ADDRESS PARITY ERROR	SEEK T.O.	SEEK * STATUS	CTL. NO. 0/1	SEEKING				CTL. 0011110

* TEST DESCRIPTOR ONLY

COMMON RESULT DESCRIPTOR BITS:
0 = I/O COMPLETE
1 = EXCEPTION CONDITION
2 = NOT READY

DATA COMMUNICATIONS CTL. ID. BITS 17-23
1000000 = ADAPTER NOT PRESENT
10nnnnn = LEASED OR DIRECT CONNECT
11nnnnn = SWITCHED LINE
nnnnn = 00010 – STANDARD LINE ADAPTER
01000 – TELETYPE ADAPTER
00100 – A.C.U. ADAPTER