CONTROL DATA CORPORATION

CONTROL DATA® MICRO-PROGRAMMABLE COMPUTER FAMILY

MICRO PROCESSOR

Preliminary Edition

HARDWARE REFERENCE MANUAL

CONTROL DATA CORPORATION

CONTROL DATA®
MICRO-PROGRAMMABLE
COMPUTER FAMILY
MICRO PROCESSOR

Preliminary Edition

HARDWARE REFERENCE MANUAL

REVISION RECORD

REVISION	DESCRIPTION
01	Preliminary Release
(10/75)	
Publication No. 88973400	

Address comments concerning this manual to:
Control Data Corporation
Publications and Graphics Division
4455 Eastgate Mall
La Jolla, California 92037
or use Comment Sheet in the back of this manual.

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

Page	Revision	Feature
Cover	---	
Title Page	---	
Revision Record	01	
iii through viii	01	
1-1 through 1-8	01	
2-1 through 2-14	01	
3-1 through 3-6	01	
4-1 through 4-19	01	
5-1 through 5-6	01	
A-1 through A-3	01	
Index-1 through Index-3	01	
Comment Sheet/ Envelope	01	

Page	Revision	Feature

PREFACE

The micro-programmable (MP) computer is the basic building block or engine for the CONTROL DATA family of micro-programmable computers. This manual describes this basic micro-level machine.

This manual is intended to familiarize the reader with the micro-level organization and operation of this computer family. The reader should be familiar with basic micro-control machine architecture.

The reader should understand that the basic engine can be used to construct a controller or higher-level processor such as the micro-programmable 1700 enhanced processor.

Additional information on the micro-programmable computer can be found in the following publications:

Title	Publication No.
Basic Micro-Programmable Processor Hardware Maintenance Manual	39451400
Detail Characterization of TTL Logic Circuits Engineering Specification	52339100
CCP Support Software MICRO Assembler Reference Manual	88988800
Micro-Programmable Computer Family Input/Output Specifications Manual	96709600

CONTENTS

APPENDIX

INDEX

FIGURES

TABLES

INTRODUCTION

The family of micro processors developed by Control Data Corporation consists of parallel mode, stored program digital processors, configured around either a 16-bit or a 32-bit word format. The micro processor (MP) provides micro-programming capabilities for program execution. Inherent in its design is the capability to configure the MP into many forms by using the same basic hardware.

The basic MP configuration consists of:

- Micro processor
- Micro memory
- Power supply
- Maintenance interface/maintenance panel

The MP can be used as a peripheral controller, a special algorithm processor, or an emulator. A typical application of an MP unit is shown in figure 1-1.

A discussion of the general characteristics of the 16-bit and 32-bit processors, including several available options, is included in subsequent portions of this manual. This manual is supplemented by the Basic Micro-Programmable Processor Hardware Maintenance Manual, which contains the information necessary for installation, checkout, operation, and maintenance of the system.

A listing of the general characteristics of the 16-bit processor and the 32-bit processor is presented in table 1-1.

FUNCTIONAL CHARACTERISTICS

The micro processor family has multilevel performance. The control section consists of a micro memory and a hardware transform function which may be set up in a number of ways, allowing the user to adapt the micro processor to meet his particular requirements. Figure 1-2 shows the basic differences in organization between the multilevel processor and the conventional processor.

This design technique provides versatility by making a portion of the hardware variable. Each user can then design his transform logic and micro-memory program to perform his specialized task. By modifying a small portion of the hardware, the MP units can function as general-purpose processors, emulators, algorithm processors, peripheral controllers, and preprocessors.

PHYSICAL CHARACTERISTICS

The MP is modularly designed, with standard TTL medium scale integration (MSI) components and commercial construction. The micro memory is a semiconductor read-only memory, read/write memory, or combined read-only memory and read/write memory. The basic word length is 64 bits, with each word holding two micro instructions. The micro memory is available in two formats. One type of micro-memory card is laid out to contain 512 words of read/write memory and/or 1K words of read-only memory. The other type has 2K of read/write memory. Two chassis locations in the 16-bit processor are prewired and reserved for micro memory; the 32-bit processor has four reserved chassis locations.

The standard chassis for the MP units shown in figure 1-3 is 18.5 inches (46.99 cm) high by 17.5 inches (44.45 cm) wide by 12.0 inches (30.48 cm) deep. This chassis includes cooling fans sufficient for the fully populated unit. The back panel is available in two configurations, depending upon whether it is for the 16-bit or the 32-bit unit. Each card slot is 0.625 inch 1.5875 cm) wide, and the card connectors on the back panel contain keying slots. Circuit cards must be similarly keyed to permit insertion. The front cover panel protects the circuit cards and improves chassis cooling.

The MP circuit card shown in figure 1-4 is 11 by 14 inches (27.94 by 35.56 cm) and has 204 input/output contacts. Construction details for the design and application of user electronic cards are contained in the CDC document, Micro-Programmable Computer Family Input/Output Specifications Manual.

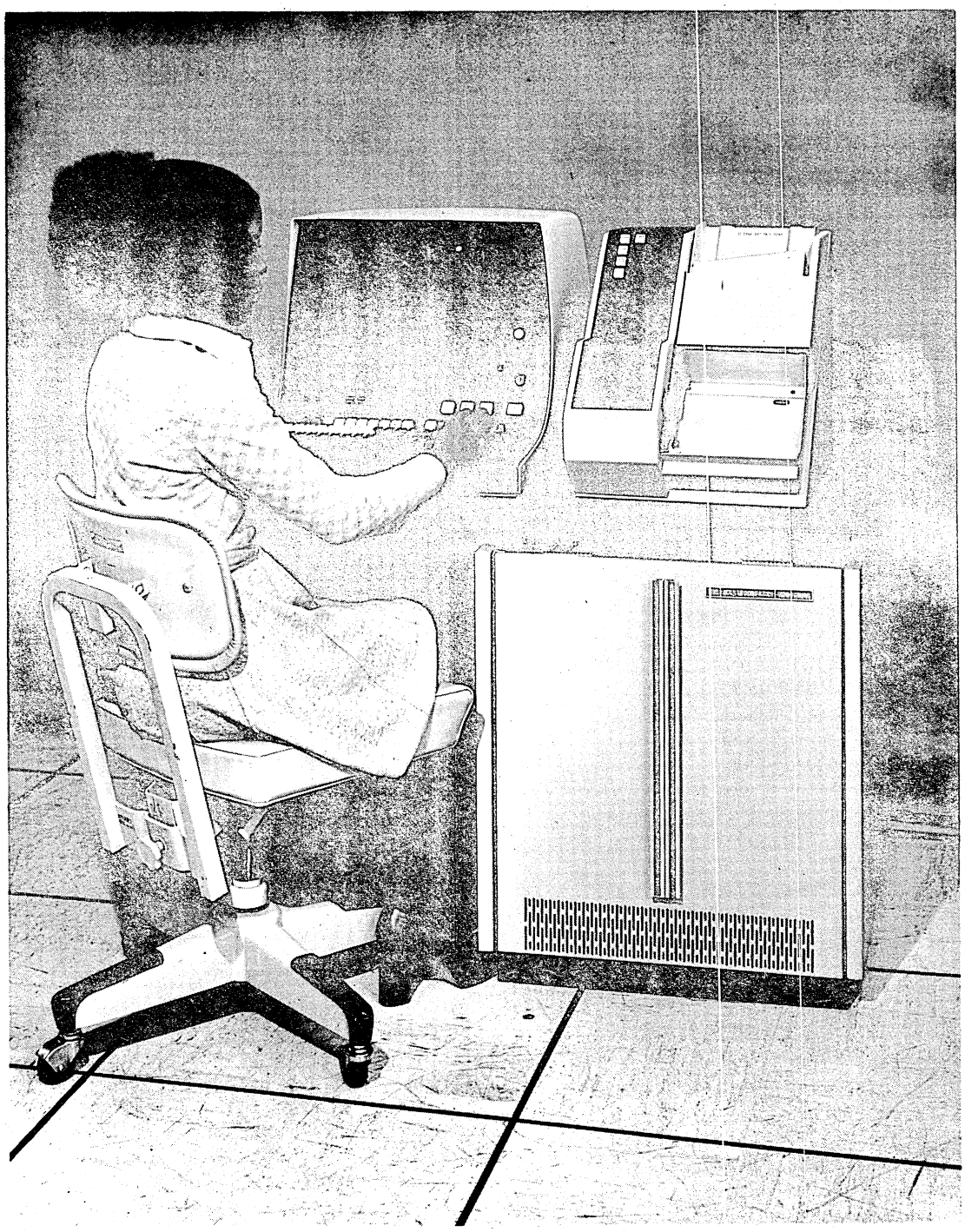


Figure 1-1. Typical MP Application

TABLE 1-1. MP16/32 GENERAL CHARACTERISTICS

BASIC CONFIGURATION	
Processor	
Type	General-purpose, micro-programmable digital processor
Organization	Register oriented or file oriented
Word Length	16 bits or 32 bits
Instruction Word	32-bit format, two micro instructions per micro-memory word
Micro-Memory Type	Semiconductor read/write memory and/or read-only memory
Micro-Memory Size	256 words, 64-bit increments; maximum of 4096 words (8192 instructions)
Micro-Memory Cycle Time	168 nanoseconds cycle time; 70 nanoseconds access time
Arithmetic	Binary with dynamic selection of ones or twos complement mode. Up to four parallel unrelated operations possible in one micro instruction.
Execution Time	Typical: 168 nanoseconds (excluding shifts) Maximum: 504 nanoseconds (excluding shifts) Shifts: 280 + 56(n) nanoseconds (where n = the number of shifts)
Panel Devices	Maintenance panel RS232-compatible console
Mechanical	
Hardware	Modular
Construction	RETMA 19-inch, rack mountable
Dimensions	Logic Chassis: Height — 18.5 inches (47 cm) Width — 17.5 inches (44.5 cm) Depth — 12.0 inches (30.48 cm) Power Supply Chassis: Height — 8.75 inches (22.25 cm) Width — 17.5 inches (44.5 cm) Depth — 16.0 inches (40.64 cm)
Weight	Logic Chassis — 40 pounds (approximately) (18 kg) Power Supply — 50 pounds (approximately) (15 kg) (+5v supply only)
Input Power	115 volts, 50/60 Hz

TABLE 1-1. MP16/32 GENERAL CHARACTERISTICS (Continued)

STANDARD OPTIONS	
Macro Memory	
Requirement	Dependent upon 16-bit MP/32-bit MP application
Type	Core memory — Available in 8K stacks 16-bit processor — Maximum 32K, 16-bit word format MOS memory — Available in 16K or 32K arrays 16-bit processor — Maximum 131K, 16-bit word format 32-bit processor — 8K, 16-bit word format or 8K, 32-bit word format Parity and protect are available in the standard stack.
Core Speed	Access — 250 nanoseconds Read/Restore — 600 nanoseconds Read/Modify/Write — 700 nanoseconds
Direct Memory Access	16-bit processor — Up to five external devices 32-bit processor — Up to five external devices
Input/Output (I/O)	
Interfaces	CDC 1700 A/Q channel TTL level Teletype Display Terminal
Operator Input Devices	Teletype ASR/KSR 33/35 CDC Model 92423 Conversational Display Terminal

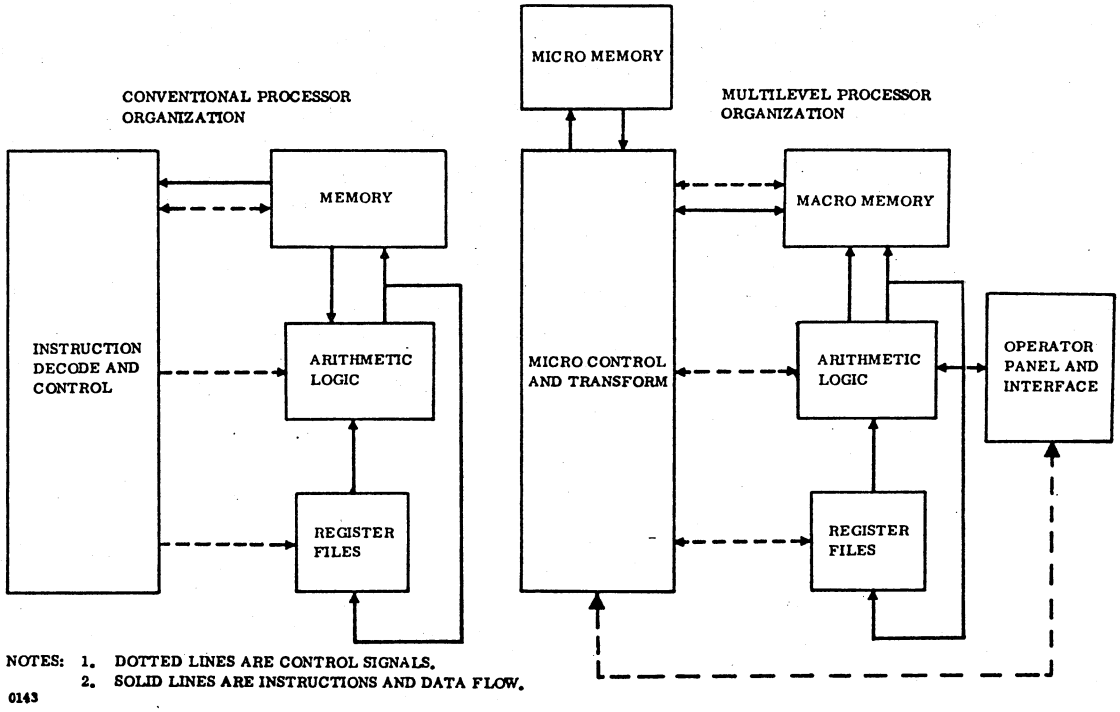


Figure 1-2. Digital Processor Organizations

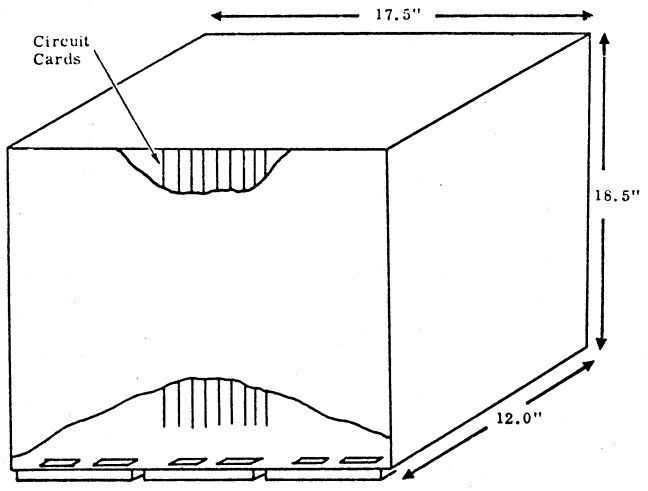


Figure 1-3. Standard Chassis for MP Units

MP power requirements vary with the particular user's application. CDC provides power supplies of ±5, ±12, and ±15 volts with input power requirements of 115 vac, 50 or 60 Hz. Physical dimensions for a power supply chassis are 8.75 inches (22.22 cm) high by 17.5 inches (44.45 cm) wide by 16.0 inches (40.64 cm) deep.

Cooling fans require 110 vac, 50 or 60 Hz. Power supplies should maintain stated values within ±5 percent deviation.

Both the 16-bit processor and the 32-bit processor chassis contain a prewired chassis location for a panel interface card. The maintenance panel is a 16-inch by 4.5-inch (40.64 by 11.43 cm) printed circuit board, connected by a flexible cable to the panel interface card. The panel contains 16 light-emitting diodes (LEDs) for display, momentary pushbuttons, and switches for data and control entry.

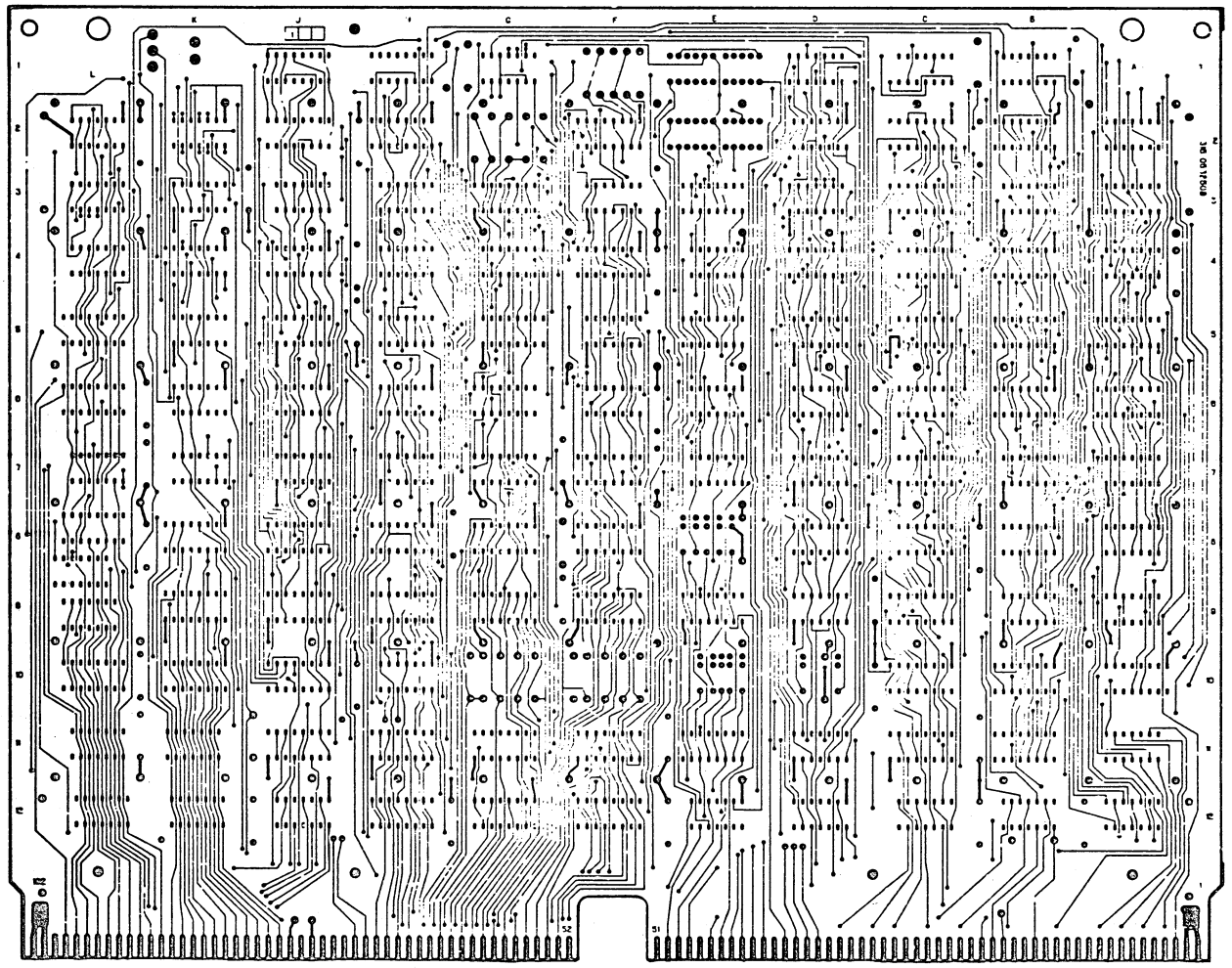


Figure 1-4. A Typical MP Circuit Card

The 16-bit processor and the 32-bit processor operate in computer room, general office, and industrial environments. They operate at temperatures from 59° F (15° C) to 104° F (40° C), withstand a maximum temperature gradient of 20° F (-6.7° C) per hour or at a rate that precludes condensation, and withstand a relative humidity of 5 to 95 percent. For the non-operating environment the temperature range extends from 20° F to 125° F (-6.7° C to 51.7° C) and a maximum thermal gradient not to exceed 60° F (15.6° C) per hour or at a rate that precludes condensation. Storage temperatures with proper packaging protection may be from -60° F to 160° F (-51.1° C to 71.1° C) and relative humidity from 2 to 98 percent with temperature cycles of not more than 60° F (15.6° C) per hour or at a rate that precludes condensation. The user should note that these ranges cover only the 16-bit processor and the 32-bit processor; peripheral equipments may require more stringent environmental controls.

MAJOR SYSTEM COMPONENT DESCRIPTION

Figure 1-5 shows the three chassis layouts available for the MP equipment, and figure 1-6 shows the overall MP system configuration. The dashed lines in figure 1-6 are options or user application interfaces.

MICRO PROCESSOR

The basic 16-bit processor consists of an arithmetic logic unit (ALU) card and a status mode interrupt (SMI) card. Two control cards (Control 1 and 2) provide the timing and control signals. The basic 32-bit processor includes a second ALU card and a second SMI card to perform 32-bit operations. Both the 16-bit processor and the 32-bit processor contain a basic transform board that may be configured in different ways to suit a particular application. Special user-required algorithm cards may also be included. The micro-processor cards are interconnected through the basic back-panel layout; however, nonstandard options require additional wiring.

MICRO MEMORY

The basic MP micro-memory boards contain micro memory of either 512 words of read/write memory with 1K of read-only memory capacity or 2K of read/write memory. The micro memory is connected to the micro processor through the back panel and is accessible only by the micro processor. 1K of read-only micro memory may also be located on the transform board if desired by the user.

MAINTENANCE INTERFACE/ MAINTENANCE PANEL

Two circuit cards (panel interface and I/O-TTY) are available for manual interface to the micro processor. The panel interface card provides the interface for a maintenance panel or the RS232-compatible consoles that have full duplex serial ASCII characteristics. The maintenance panel is a 16 by 4-1/2 inch (40.64 by 10.43 cm) printed circuit board mounted directly above the main chassis. It connects to the panel interface module through a flexible cable that includes power and ground connections.

I/O-TTY CARD

The optional I/O-TTY card can interface with Teletype Corporation Models ASR/KSR 33/35 Teletypes or Control Data 92423 Conversational Display Terminals. A TTL bus is available in the I/O-TTY card for interfacing the peripheral I/O controller cards in the main chassis to the micro processor. The I/O-TTY card interfaces directly to the micro processor through the ALU card, the control cards, and the SMI card.

Card slots are prewired for the panel interface and I/O-TTY cards in the back panel of both the 16-bit and the 32-bit processors. Control and data lines tie directly into the control cards and the ALUs. The choice of interface is determined by the manual operator requirements.

I/O INTERFACE

The main chassis includes nine card slots for I/O use in the 16-bit processor and eight card slots for I/O use in the 32-bit processor. Only the power and ground pins are assigned in seven of the I/O card slots in the 32-bit processor. All I/O card slots can be wired to meet the user's requirements. The 16-bit processor has five card slots prewired for 1700 A/Q channels and four card slots prewired for 1700 A/Q-DMA channels. The 32-bit processor has one card slot prewired for 1700 A/Q.

Z	8K CORE MEMORY STACK†
Y	8K CORE MEMORY STACK†
X	8K CORE MEMORY STACK†
W	8K CORE MEMORY STACK
V	MEMORY INTERFACE†
U	PANEL INTERFACE†
T	MICRO MEMORY†
S	MICRO MEMORY/ALGORITHM†
R	TRANSFORM
P	CONTROL 1
N	CONTROL 2
M	ALU
L	SMI
K	I/O-TTY†
J	1700 A/Q†
H	1700 A/Q-DMA†
G	1700 A/Q†
F	1700 A/Q-DMA†
E	1700 A/Q†
D	1700 A/Q-DMA†
C	1700 A/Q†
B	1700 A/Q-DMA†
A	1700 A/Q†

16-BIT PROCESSOR (CORE MEMORY)

AA	32K MOS MEMORY†
Z	32K MOS MEMORY†
Y	32K MOS MEMORY†
X	32K MOS MEMORY†
W	MEMORY INTERFACE ADD
V	MEMORY INTERFACE DATA
U	PANEL INTERFACE†
T	MICRO MEMORY†
S	MICRO MEMORY/ALGORITHM†
R	1700 TRANSFORM
P	CONTROL 1
N	CONTROL 2
M	ALU
L	SMI
K	I/O-TTY
J	1700 A/Q
H	1700 A/Q-DMA
G	1700 A/Q-DMA
F	1700 A/Q
E	1700 A/Q
D	1700 A/Q-DMA
C	1700 A/Q
B	OPEN
A	1700 A/Q-DMA
AC	1700 A/Q
AB	OPEN

16-BIT PROCESSOR (MOS MEMORY)

AB	8K CORE MEMORY STACK, † UPPER
AA	MEMORY INTERFACE, † UPPER
Z	8K CORE MEMORY STACK, † LOWER
Y	MEMORY INTERFACE, † LOWER
X	PANEL INTERFACE
W	MICRO MEMORY†
V	MICRO MEMORY†
U	MICRO MEMORY†
T	MICRO MEMORY
S	TRANSFORM
R	CONTROL 1
P	CONTROL 2
N	ALU, LOWER
M	SMI, LOWER
L	ALU, UPPER
K	SMI, UPPER
J	I/O-TTY†
H	OPEN
G	OPEN†
F	OPEN
E	OPEN
D	OPEN
C	OPEN
B	OPEN
A	OPEN

†OPTION

32-BIT PROCESSOR

0140

Figure 1-5. Basic MP Chassis Layout

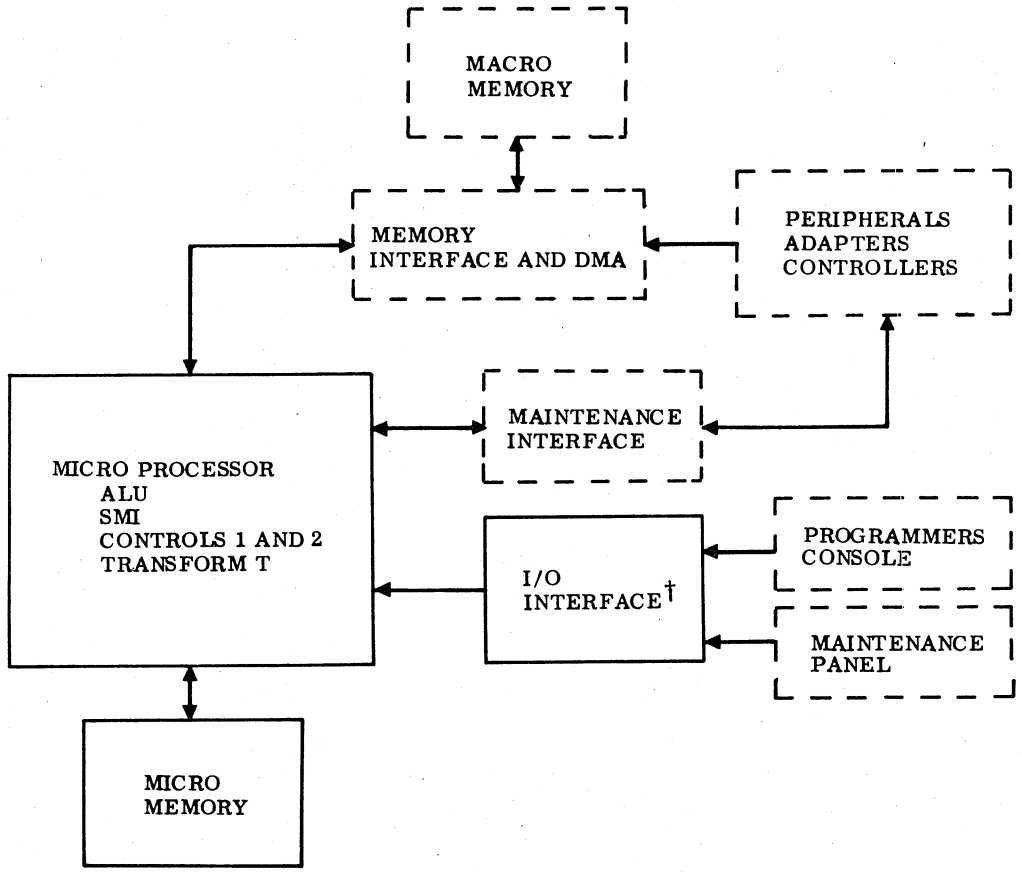


Figure 1-6. MP System Configuration

MACRO MEMORY (CORE) AND MEMORY INTERFACE

The core macro memory consists of memory stacks and interface cards to provide a pluggable option for the 16-bit processor and the 32-bit processor units. Macro-memory core stacks are mounted on standard 11 by 14 inch circuit boards; each stack requires two card spaces in the chassis. Data flow is in 16-bit word format for the 16-bit processor with a maximum of 32K possible in the standard MP chassis. The 32-bit processor can be configured with either an 8K, 16-bit format or an 8K, 32-bit format. The macro memory interface provides direct memory access (DMA) with external peripherals, including Control Data 1700 direct storage access (DSA).

This section provides a general technical description of the micro processor, the micro memory, and the various options available to the user for incorporation into his system. A block diagram of the MP system is shown in figure 2-1.

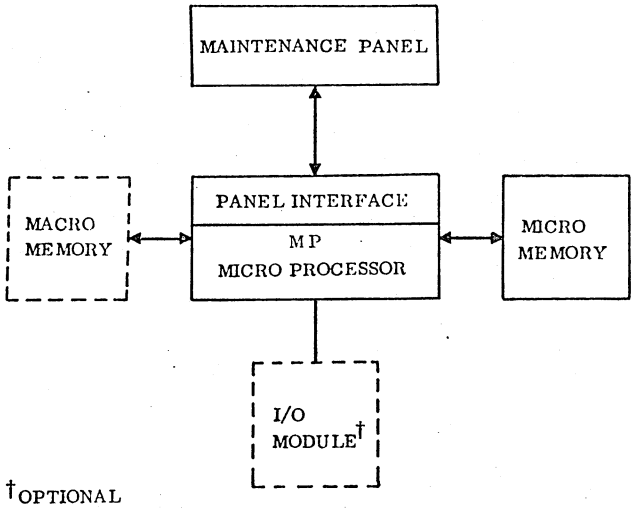


Figure 2-1. MP Block Diagram

MICRO PROCESSOR

The MP consists of an arithmetic/logical unit (ALU) module, a status mode interrupt (SMI) module, two control modules, and an application-dependent transform module for 16-bit processing. For a 32-bit configuration, a second ALU module and a second SMI module are added. The detailed MP organization (figure 2-2) shows the MP register interconnected primarily by selectors. A selector is a multiplexer that transfers one of several inputs to an output. The selectors are numbered from S1 to S9. S1, S2, S3, and S9 are word-width selectors that transfer complete words to the output. Selector S8 transfers eight bits, S5 transfers eight bits, and selector S7 is a single bit selector for transferring one of 16 or 32 inputs to its output. Selectors S4 through S6 are 12-bit selectors and transfer a micro memory address to the page/memory address register (P/MA).

ARITHMETIC/LOGICAL UNIT (ALU) AND DATA TRANSFER ORGANIZATION

The ALU provides the arithmetic and logical capabilities of the MP. This unit combines two input words of the system word length; one from the A input, provided by S1, and the other from the B input, provided by S2. These inputs are combined according to the function code specified in the micro instruction. The result is immediately available at the output of the ALU for possible shifting via S3 and delivery to a destination register, memory interface, panel interface, and I/O. The unshifted output of the ALU is delivered to the status/mode (SM) and mask registers. The ALU output can be ignored on an operation. The results of the ALU operation regarding sign, zero, and magnitude (by means of a carry-out test) are available to the test bit logic for instruction sequencing.

The data transfer organization of the MP provides for storing data in one of six working registers and two files and for selecting data for processing through the ALU. ALU results are transferred back to one of the registers or out of the organization to control external equipment. The primary data (working) registers are I, P, A, F, X, and Q. The following is a brief description of these registers:

- I Register — Input to register I comes directly from the output of selector S1. This enables data on the tristate bus to be stored directly in register I and simultaneously input to the ALU for some other operation. This is particularly useful in configurations using macro memory. The word length output is available at selection S1.
- P Register — This word length register receives data from selector S3 and output to the A input of the ALU via S1. In computer emulation configurations, it is normally used to contain the macro program instruction counter.
- A Register — The word length A register may be used for data shifts, either by itself or in conjunction with the Q register as a double-length shift register. The shift function is independent of the ALU and S3. This general-purpose register inputs from S3 and outputs to the A input of the ALU via S1.

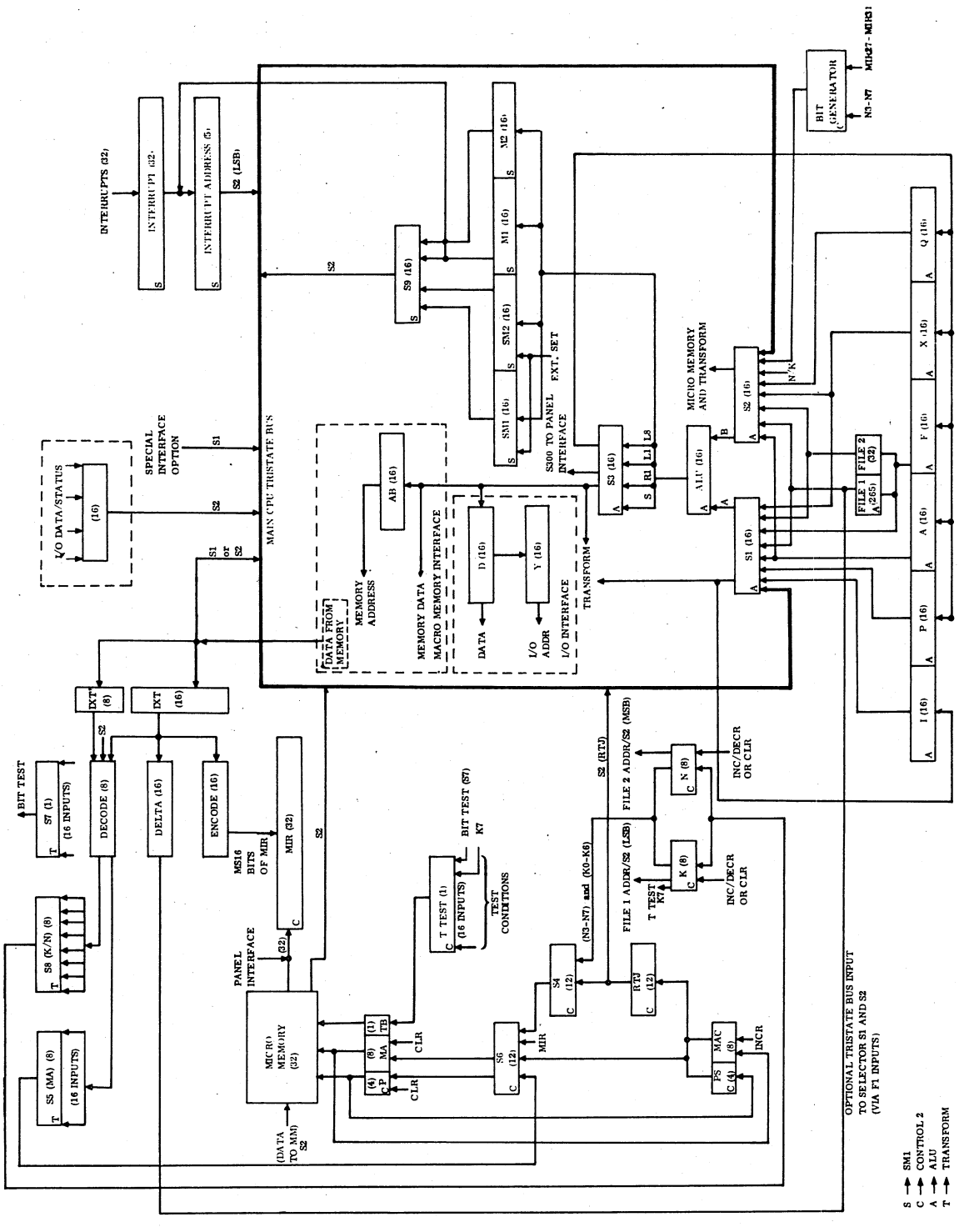


Figure 2-2. Detailed MP Block Diagram

- F Register — This word length general-purpose register inputs from S3 and outputs to the A input of the ALU via S1 or the B input via S2. The F register serves as an entry register for file 2 or file 1 when either of these is a destination of an ALU operation.
- X Register — The X register is a word length general-purpose register whose input is from S3. It outputs to ALU input A via S1 and to input B via S2.
- Q Register — The word length Q register is a general-purpose register that outputs to ALU input B via S2. It may be shifted left or right in conjunction with the A register. This shift is independent of the ALU and S3.
- File 1 — This general-purpose word length contains 256 word length words addressed by the contents of the K register. The register output shares the tristate bus with an optional source. This may be from a transform or an external device. A status mode bit selects either the file 1 output or the option. The data is sent to ALU input A via S1 or to input B via S2.
- File 2 — File 2 contains 32 word length words addressed by the lower five bits of the N register. It delivers its output to ALU input A via S2 and to ALU input B via S1. File 2 is intended as a source for constants, but it may be used as a general-purpose file.
- Bit Generator (BG) — The bit generator is capable of setting one bit to input B of the ALU via S2. The single bit can be set at any of the word length positions in the word. The bit position is established by the least significant five bits in the micro instruction (MIR27 through MIR31) or by the lower five bits of register N. Status mode bit SM102 determines which source has control. If SM102 is set, the bit generator position is determined by bits N03 through N07.
- Selector S1 — S1 provides for the selection of one of eight inputs for delivery to the ALU (A input), to the I register (if I is a destination register), and to the transform module. Input to the selector is indicated below:

Position	Input Source
0	File 2
1	P register
2	I register
3	X register
4	A register
5	F register
6	File 1 or external input
7	Main CPU tristate bus (i.e., data from macro memory, SM and mask registers, algorithm option)

- Selector S2 — Similarly, S2 provides for the selection of one of eight inputs for delivery to ALU (B input), micro memory, and the transform module. Input to selector S2 is indicated below:

Position	Input Source
0	File 2
1	N and K registers
2	Bit generator
3	X register
4	Q register
5	F register
6	File 1 or external input
7	Main CPU tristate bus (i.e., data from macro memory, interrupt address, micro memory, I/O data/status register, maintenance panel, and return jump (RTJ) register)

STATUS MODE INTERRUPT MODULE

The status mode interrupt module contains the status/mode registers, the mask registers, and the interrupt registers for the micro processor.

Status Mode (SM) Register

The SM registers allow the micro program to control the mode of operation and also allow the micro program to examine the status of certain internal and external conditions. The CPU can access one of the two SM registers, SM1 or SM2. Each SM register contains the same number of bits as the basic processor word; 16 bits for a 16-bit processor or 32 bits for a 32-bit processor. Therefore, the maximum total number of SM bits in the CPU is twice the basic processor word length.

The SM register module contains 16 bits of SM1 and 16 bits of SM2. The micro program can set or reset all 32 bits of an SM module by transferring information to the SM register from the output of the ALU. Master clear will clear SM1 and SM2.

The SM register bits are numbered as follows:

16-bit processor: SM1 bits numbered S100 to S115
SM2 bits numbered S200 to S215

32-bit processor: SM1 bits (first module) numbered S100 to S115 (most significant)
SM1 bits (second module) numbered S116 to S131 (least significant)
SM2 bits (first module) numbered S200 to S215 (most significant)
SM2 bits (second module) numbered S216 to S231 (least significant)

The functional characteristics of the individual bits of the SM module are shown in table 2-1. SM bit assignments for the 16-bit and 32-bit processors are shown in table 2-2.

MP Operating Modes

Operating modes are incorporated into the system during manufacture; these modes are controlled by a bit in the SM register. If an option is not needed, the associated mode bit in the SM register may be reassigned as required.

MP Status

Status bit assignments are incorporated into the system during manufacture, and these status bits are set by the condition detected. The clearing of the status bit must be performed by the micro program with the exception of SM1 (bits 14, 15, 30, and 31) and SM2 (bits 2, 14, 15, 30, and 31) which have an external clear input.

TABLE 2-1. SM BIT CHARACTERISTICS

SM Bits		Functional Characteristics †		
16-Bit	32-Bit	Set By	Cleared By	Output Available From
FLAG100 — FLAG103	FLAG100 — FLAG103 FLAG116 — FLAG119	SETF/j command	CLRF/j command or master clear	True
FLAG200 — FLAG203	FLAG200 — FLAG203 FLAG216 — FLAG219	SETF/j command	CLRF/j command or master clear	Complement
SM104 — SM107 SM204 — SM207	SM104 — SM107 SM120 — SM123 SM204 — SM207 SM220 — SM223	External input	Master clear	True/complement
SM108 — SM111	SM108 — SM111 SM124 — SM127	External input	Master clear	True
SM112 — SM113	SM112 — SM113 SM128 — SM129	External input	Master clear	Complement
SM114 — SM115	SM114 — SM115 SM130 — SM131	External input	External input or master clear	Complement
SM208 — SM213	SM208 — SM213 SM224 — SM229	External input	Master clear	True
SM214 — SM215	SM214 — SM215 SM230 — SM231	External input	External input or master clear	True

†All SM bits can be set or reset by transferring information from the ALU to the SM registers.

TABLE 2-2. SM REGISTER BIT ASSIGNMENTS

SM Bit	Type†	Name	Function
100	M	Double-Precision (optional)	1 = The ALU and the ALU* are combined to form a double-word-length ALU if the double-precision hardware option is included. This double-word-length ALU operates for addition and subtraction but not for logical operations. 0 = The double-precision ALU* is disconnected from the ALU, and no operation takes place in ALU*.
101	M	Ones Complement	1 = The ALU (and ALU* for double-precision operation) operates in ones complement arithmetic mode for addition and subtraction. 0 = Operations are in twos complement arithmetic mode for addition and subtraction.
102	M	Bit Generator Input From N	1 = The bit generator is controlled by the lower five bits of the N register (N03 through N07). 0 = The bit generator is controlled by the lower five bits of the micro instruction (MIR27 through MIR31).
103	M	Adder Split (optional)	1 = The ALU is split into two independent ALUs at the adder split point for arithmetic operation. The split point is between bits 07 and 08 on a 16-bit processor and is normally between bits 15 and 16 on a 32-bit processor. If ones complement mode is selected while the adder is split, the upper adder operates in twos complement mode while the lower adder operates in ones complement mode. 0 = The ALU operates as a single-word-size ALU.
104		Open	
105	S	Protect Fault (optional)	Set if a protect violation occurred when data was written into macro memory.
106	M	Active Interrupt System (optional)	1 = The processor interrupt system is activated, and the INTU test can examine the interrupt system. The interrupt lines to be enabled/disabled are selectable in groups of eight; i.e., this bit controls any combination of four groups on a 16-bit processor and any combination of eight groups on a 32-bit processor. Groups not controlled by the mode bit can be constantly enabled or disabled. 0 = The interrupt system is disabled, and the INTU test will always receive a reply of no interrupt present on groups controlled by the mode bit.
107		Open	

†M = Mode
S = Status

TABLE 2-2. SM REGISTER BIT ASSIGNMENTS (Continued)

SM Bit	Type †	Name	Function
108	S	Memory Parity Error (optional)	Set if the macro-memory interface detects a parity error during a read operation with macro memory.
109	M	Micro Halt	1 = Any micro instruction with a HALT code in the S field stops processor operation on completion of the micro instruction. 0 = The HALT code in the S field of any micro instruction is ignored.
110	S	Overflow	Set on detection of an arithmetic overflow condition. The arithmetic operation is add or subtract with an overflow test, and the arithmetic result is inconsistent with the sign of the operands and the arithmetic operation.
111		Enable F1 (optional)	1 = The output of file 1 is enabled to selector S1 or S2. This overrides any selection of an external source. Bit must be set to 1 to read/write file 1. 0 = The output of file 1 is disabled as an input source to selector S1 or S2. If an external source (such as the transform module) is available, it may be enabled to selector S1 or S2 at the same position as defined for file 1.
112		Open	
113	M	Select XT/MA (optional)	1 = The MA transform determines the micro-memory address for read/write micro-memory operand references. 0 = The N/K register determines the micro-memory address for read/write micro-memory operand references.
114, 115		Open	
116 - 131		Open	These SM register bits are only available to the 32-bit processor.
200	M	Auto Data Transfer†† (optional)	Used principally to implement auto data transfer (ADT) I/O (a pseudo DMA method of I/O for an MP using the 1700 A/Q channel). Each data transfer condition results in an interrupt that is processed by the micro code while emulation of the macro instruction set continues between data transfers. 1 = The W=0 and program protect signal lines are generated, and it permits control of the clock interrupts. 0 = The micro interrupt for the clock circuit cannot be cleared, and the macro interrupt cannot be set; the states of the W=0 and program protect lines are determined by alternative logic.

†M = Mode
S = Status

††If the I/O-TTY board is present, SM200 through SM202 must remain assigned as is.

TABLE 2-2. SM REGISTER BIT ASSIGNMENTS (Continued)

SM Bit	Type†	Name	Function
201	M	Read (optional)††	Generates the read line.††† Set if the read line is active. This mode bit is also used in controlling the clock.
202	M	Write (optional)††	Generates the write line.††† Set if the write line is not active. This mode bit is also used in controlling the clock.
203	M	STERM (optional)††	Generates the STERM line.††† Set if the STERM line is active.
204	M	Deadstart	1 = The deadstart mode is selected on the panel-I/O interface. 0 = The deadstart mode is disabled on the panel-I/O interface.
205, 206		Open	
207	M	Select Page (XT/MA) (optional)	1 = An MA transform can be executed across a page boundary by setting the desired page in the S field of the micro instruction. Normal S field operations are disabled. 0 = An MA transform is limited to the page in which the micro instruction containing the MA transform command resides. The S field of the micro instruction is decoded for normal operations.
208 - 213		Open	
214	M	Pre-enable Console Interface to MIR	1 = The output of the console interface is pre-enabled to MIR. A page jump or return jump micro instruction must be executed with this mode bit set to 1 to force the actual selection. If the processor is master-cleared, the console interface is enabled to MIR. 0 = The output of the micro memory is pre-enabled to MIR. A page jump or return jump micro instruction must be executed with this mode bit set to 0 to force the actual selection to the page and address selected.
215		Open	
216 - 231		Open	These SM register bits are only available to the 32-bit processor.

† M = Mode
S = Status

†† If the I/O-TTY board is present, SM200 through SM202 must remain assigned as is.

††† A bus line that is available at all 1700 A/Q I/O ports and is used by 1700 A/Q controllers and adapters that feature the ADT mode of operation

Interrupts and Mask Register

The interrupt system is implemented as a sampled data system at the micro-program level instead of a true interrupt that is used in conventional computers (i.e., the interrupt system provides a sampling capability in which a micro instruction can sample the interrupt system to see if any interrupt present has its corresponding mask register bit set to 1). This sample is taken by performing an INTU operation in the T field of a micro instruction. If there is an interrupt in the system whose mask register bit is 1 and the interrupt system is enabled, the next micro instruction is executed from the upper of the next micro-instruction pair. If there is no such interrupt, the next micro instruction is executed from the lower of the next micro-instruction pair.

When an interrupt is recognized, the micro program samples the interrupt address encoder to identify the most significant (highest priority) interrupt. The interrupt address encoder must be read in the micro instruction following the interrupt test to be sure of a correct interrupt line address. If the interrupt address is read earlier or later, there is a possibility that the address encoder output is unstable due to a newly arrived interrupt. The interrupt address is read by performing an INTA operation in the B' field of any micro instruction.

No standard interrupts are defined; the use of the interrupt system and the design of interrupt are application functions. Interrupts are identified by the corresponding mask bits that are assigned to control the interrupt recognition. The bits in the mask registers are identified as follows:

Mask Register 1 (M1):

M100 through M115 (16-bit processor)
M100 through M131 (32-bit processor)

Mask Register 2 (M2):

M200 through M215 (16-bit processor)
M200 through M231 (32-bit processor)

Interrupt addresses are generated by the interrupt address encoder according to the assignments for 16-bit and 32-bit processors shown in table 2-3.

The interrupt priorities correspond to the interrupt address generated; that is, interrupt address 0 is associated with the highest priority interrupt line and interrupt address 31 is associated with the lowest priority interrupt line in a 16-bit processor. For example, an interrupt associated with M112 would have priority over an interrupt associated with M111 in a 16-bit processor and an interrupt address of 3 would be generated by the interrupt address encoder.

The output from the interrupt address encoder is the complement of the interrupt address for input to S2; thus, the transfer of the interrupt address to the X register, for example, would be coded using a -B code in the F field, INTA in the B' field, and X in the D field. This results in the transfer of the correct interrupt address.

A design option in the interrupt system provides for activating interrupts in groups of eight interrupt lines. Therefore, on a 16-bit processor any combination of four groups can be controlled by the enable interrupt SM bits and on a 32-bit processor any combination of eight groups can be controlled by SM bits. Groups not controlled by SM bits can remain active while the remaining interrupts are enabled or disabled.

Interrupt signals must be steady-state when they are input to the interrupt system and must indicate the presence of an interrupt when set to 0. If a pulse-type interrupt is required, the pulse interrupt signal is used to set a bit in the SM register; this SM bit is then wired to the interrupt system. On recognizing this interrupt, the micro program clears the interrupt condition by clearing the SM bit.

CONTROL MODULES

The micro-processor control logic is divided between two control modules. This logic controls the micro-instruction fetch, decode, and execution. The functions of the controls modules are described as follows:

Control 1

Control 1 contains the following functional processor elements:

- MP operating control enables and fan-outs
- Main timing generator
- Micro-instruction register (MIR02 through MIR15)
- Decode logic for the A, B, D, and F fields of the micro instruction including the control signals for the selectors, registers, and arithmetic/logical unit (ALU)

TABLE 2-3. MP INTERRUPT ADDRESSES

Mask Bit	Interrupt Address — Mask Register 1		Mask Bit	Interrupt Address — Mask Register 2	
	16-Bit Processor	32-Bit Processor		16-Bit Processor	32-Bit Processor
100	15 Lowest Priority (M1)	47 Lowest Priority (M1)	200	31 Lowest Priority (M2)	63 Lowest Priority (M2)
101	14	46	201	30	62
102	13	45	202	29	61
103	12	44	203	28	60
104	11	43	204	27	59
105	10	42	205	26	58
106	9	41	206	25	57
107	8	40	207	24	56
108	7	39	208	23	55
109	6	38	209	22	54
110	5	37	210	21	53
111	4	36	211	20	52
112	3	35	212	19	51
113	2	34	213	18	50
114	1	33	214	17	49
115	0 Highest Priority (M1)	32	215	16 Highest Priority (M2)	48
116	--	15	216	--	31
117	--	14	217	--	30
118	--	13	218	--	29
119	--	12	219	--	28
120	--	11	220	--	27
121	--	10	221	--	26
122	--	9	222	--	25
123	--	8	223	--	24
124	--	7	224	--	23
125	--	6	225	--	22
126	--	5	226	--	21
127	--	4	227	--	20
128	--	3	228	--	19
129	--	2	229	--	18
130	--	1	230	--	17
131	--	0 Highest Priority (M1)	231	--	16 Highest Priority (M2)

Note: The interrupt address generated is the same as its priority level; i.e., the highest priority interrupt generates a 0 interrupt address and the lowest priority interrupt (16-bit configuration) generates a 31 interrupt address.

- Overflow condition detector
- Carry-in generator circuit for the ALU

Control 2

Control 2 contains the following functional processor elements:

- Bit generator
- K and N registers
- Micro-memory address registers and control circuitry

  -Page/memory address register (P/MA)
  -Page storage register (PS)
  -Micro-memory address counter (MAC)
  -Return jump register (RTJ)
- Micro-instruction register: MIR00, MIR01, and MIR16 through MIR31
- M-, S-, C-, and T-field decoders and miscellaneous control circuitry
- Macro-memory interface control and miscellaneous circuitry

The principal micro-processor operating and control registers are contained on the control 2 module. The descriptions that follow describe the functions of each register.

Bit Generator

The bit generator generates a 1 bit to any position in a word as input to the B side of the ALU. Control to drive the bit generator is derived either from a micro instruction (MIR27 through MIR31) or from the lower five bits of the N register (N03 through N07).

K and N Registers

The K register is an eight-bit counter that can be cleared, incremented, or decremented. It is used to address file 1 in addition to any program usage as a counter. The original value of K can be tested against zero by the micro instruction. K is selectable as an input to S2.

The N register is an eight-bit counter that may be cleared, incremented, or decremented. It is used to address file 2, control shifts, and control the scale operations, and it may be used as an iteration counter that controls micro-instruction execution for operations such as multiplication and division. It may also be used as a programmed counter, since the original value of N can be tested against zero by the micro instructions. N is selectable as an input to S2.

The N and K registers may be combined to provide micro-memory operand addresses outside the current operating page. They are gated through S6 to P/MA for this operation. In this operation the five least significant bits of N provide the upper portion of the 12-bit address. The seven most significant bits of K provide the lower portion of the 12-bit address. The least significant bit of K is used to determine the upper or lower 32-bit operand. An SM bit determines whether to use N/K data or the output of the transform module for this operation.

Page/Memory Address (P/MA) Register

The micro memory is addressed by the P/MA register. The P portion of the register is a page control and consists of four bits that specify up to 16 pages of micro memory. A page consists of 256 words or 512 micro instructions. The MA portion of the register is an eight-bit register that specifies one of the 256 micro-instruction pairs within the page that is to be the source of the next micro instructions. Micro-instruction sequencing is designed so that no automatic overflow of addressing from the MA portion to the P portion occurs. Any control transfer between pages is initiated by a page jump operation, MA transform operation, or clear page operation.

Page Storage (PS) Register

The PS register is a four-bit holding register used to determine the page for the next instruction. In operation, the contents of the page register are transferred to the PS at each micro-memory reference. Depending on the sequencing operation specified in the micro instruction, the PS may or may not be used to obtain the next micro instruction.

Memory Address Counter (MAC)

The MAC is a counter used to determine the next location within a page following the current location specified in the MA register. In operation, the contents of the MA register are transferred to the MAC at each micro-memory reference and are then incremented by one to point to the next location. Depending on the sequencing operation specified in the micro instruction, the MAC may or may not be used to obtain the next micro instruction. Sequencing of the MAC is such that location 0 within a page follows location 255 of that page.

Return Jump (RTJ) Register

When specified by a micro instruction, the RTJ register captures the location of the next micro-instruction pair. When this capture is specified, the contents of MAC are incremented and PS and MAC are stored in the RTJ register. The contents of the RTJ register are unchanged until the next command is given to save a new address. This saving of the next instruction pair location is independent of any actual transfer of control. The output of the RTJ register can be gated to the P/MA register to perform operations or may be read into the MP through selector S2.

Micro-Instruction (MIR) Register

The MIR is a 32-bit register used to hold the micro instruction during execution. Micro instructions are normally entered into the MIR from the micro memory; either the upper or lower 32 bits of the micro-memory location are gated to the MIR based on the value of the test bit determined during the preceding micro instruction.

Micro instructions may also be transferred into the MIR from the maintenance panel interface. This is done as a result of an operator request.

TRANSFORMS

The standard transform feature of the MP enables the micro program to select a pattern of bits from the MP data transmission paths which, in turn, forms micro-memory addresses, sequences the micro program, and sets the contents of the N and K registers.

The transform hardware is packaged on a separate module. The standard transforms are implemented on two selectors, S5 and S8, that are located on the transform module with bit test selector S7.

The MP transform module provides unused area for the user to adapt transforms to his application. Specially designed transforms can also be developed to meet user requirements.

SPLIT ADDER OPTION

The split adder option allows the main ALU to be split into two independent adders. This split is activated by setting the adder split flag in the SM register. The split blocks the carry between the two portions of the adder. The upper portion functions as a twos complement adder; the lower portion can function as a ones complement or a twos complement adder, depending upon the state of the ones complement SM register flag (32-bit processor only). In ones complement mode, the carryout of the lower portion is used as the end-around carry bit. In twos complement mode, both portions act as independent twos complement adders. The split adder has no effect on logical operations because no carry is involved in these operations.

DOUBLE PRECISION OPTION

The double-precision hardware arithmetic option provides the capability to perform arithmetic on double-length operands. The double-precision logic contains three additional registers (A*, Q*, and X*) and a second ALU (ALU*) distinct from the main MP elements. The A* register is unconditionally input to ALU*. The output of ALU* can be shifted left and right in a multiply or divide operation, and the output goes to the A* register. The X* and Q* registers are loadable only; they cannot be specified as destinations for the results from ALU*. The Q* register can be shifted during double-precision multiply or divide iterations. The double-precision logic, if present, is enabled when the double-precision flag is set in the SM register.

SCALE POINT

Normally, the scale point is between 0 and 1 of the A register for scale instructions. Through a hardware

change the scale point may be set between different bits in the A register when necessary for efficient floating-point emulation.

MICRO MEMORY

The micro memory is available in read/write memory and read-only memory as described below.

- A read/write random access memory (RAM) is available for use during program development and in applications that require the MP to be reprogrammed or reorganized for multiple applications. This read/write micro memory can either be loaded from an external device or data can be written into micro memory under control of the micro program.
- A read-only micro memory (ROM) is available for fixed applications of the MP. It is programmed during manufacture.

MAINTENANCE PANEL INTERFACE

All configurations of the MP contain a dedicated chassis location for the maintenance panel interface card.

The programmers console is optional on all MP configurations and interfaces through the same card as the maintenance panel. The basic signal flow of the panel interface is shown in figure 2-3.

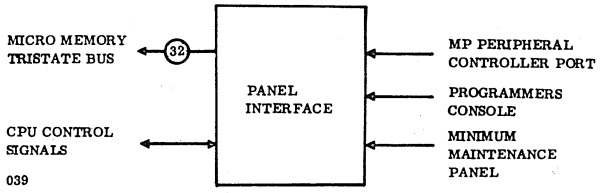


Figure 2-3. Basic Signal Flow of Panel Interface

The interface card interfaces to the MP main bus and to internal MP control signals. The interface of the programmers console is RS232-compatible. No special programmers console is included. The programmers console consists of a teletypewriter, a display, or any device with two-way simultaneous serial (full duplex) ASCII characters; e.g., Control Data 92423 Conversational Display Terminal. The programmers console can be remoted, but no modem or modem control signals are provided. The interface to the maintenance panel is not RS232-compatible.

MACRO MEMORY

The macro memory options consist of core stacks or MOS arrays and interface cards. The interface cards provide the control and interfacing required for MP memory functions and peripheral memory functions. Core memory stacks for the 16-bit and 32-bit processors are available in 8K increments of 19-bit format. Sixteen data bits, a data parity bit, a protect bit, and a protect parity bit make up the 19-bit word. The data parity, protect, and parity protect bits are generated and tested in the interface card. The 16-bit processor, with one memory interface card, can address up to 32K 16-bit words in the main chassis. With the MOS memory, the 16-bit processor can address up to 131K 16-bit words in the main chassis. The 32-bit processor macro memory can be configured in two ways: An 8K core, 16-bit memory is available with one interface card, and an 8K core, 32-bit word format is available using two 8K 16-bit core stacks and two interface cards.

Direct memory access (DMA) is available for both the 16-bit processor and 32-bit processor. DMA for the 16-bit and 32-bit processors provides access to macro memory for five external devices.

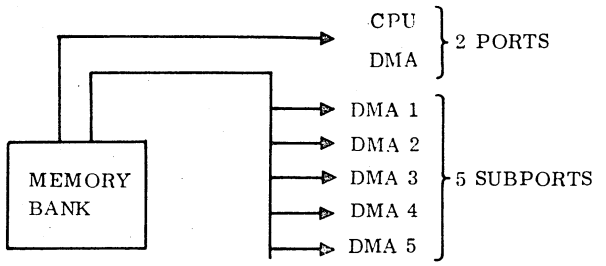
MACRO MEMORY CONFIGURATION

Figure 2-4 shows the macro memory configuration.

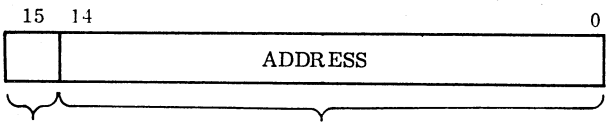
The memory configuration is a one-bank, two port memory. The single bank means that only one reference at a time may take place. Two ports provide two independent data and control paths to the memory; either port, MP or DSA, may request memory independently of any operation underway on the other port.

I/O-TTY MODULE

Figure 2-5 illustrates the major signal flow paths to and from the optional I/O-TTY module. The user may add the I/O-TTY module to his basic micro processor or incorporate his own design to adapt to his unique requirements.

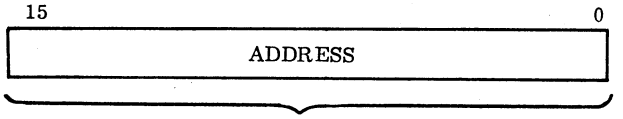


MULTILEVEL INDIRECT MODE (FCR18 = 1)



SET IF AN INDIRECT ADDRESS 0000_{16} TO $7FFF_{16}$ (32K SYSTEM)

DIRECT ADDRESS MODE (FCR18 = 0)



0000_{16} TO $FFFF_{16}$ (65K SYSTEM)

Note: With MOS memory, only four DMA ports are available.

Figure 2-4. Macro Memory Configuration

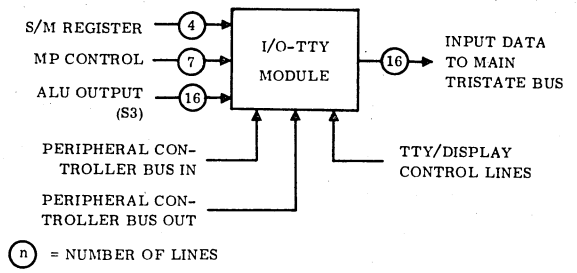


Figure 2-5. Major Signal Flow Paths of the I/O-TTY Module

The basic functions provided by this module include:

- Clock — The clock, in conjunction with the micro code, appears as a peripheral to the macro-level programmer.
- I/O-Teletypewriter/Display Controller — The I/O-teletypewriter controller is an integral part of this module. It interfaces to Teletype Corporation ASR/KSR 33/35 Teletypes and Control Data 92423 Conversational Display Terminal.
- Internal Peripheral Controller Bus — This module provides all I/O data lines, interrupts, and control signals necessary to generate, in conjunction with the micro code, an internal CDC 1700 A/Q I/O bus. This TTL-level bus is intended to interface with controllers located in the basic MP chassis.

The MP is interfaced to the I/O-TTY module as follows:

- ALU Output — All output data and address information is provided from the output of the ALU via S3.
- SM Register — All commands to peripheral controllers are generated by micro-code manipulation of the MP status mode register.
- MP Control — Timing and control information for controlling internal I/O-TTY module data gating is provided from the MP control signals.
- Interrupts — Interrupts from peripheral controllers (within the basic chassis) are wired directly from the peripheral controller module to the MP.
- Input Data and Peripheral Response Signals — All of these are provided to the MP on the main MP tristate bus.
- Clock — The clock circuit is an integral part of the I/O-TTY board and is designed to appear as a 1700 peripheral to the MP macro level software. However, even through the clock is designed primarily to be controlled via an MP ADT table, the circuit can be used and controlled at the micro level in the following manner:

  1. The micro code enables the clock circuit, which causes the clock circuit to generate an interrupt every 3.33 milliseconds (after the first occurrence of the clock interrupt), and enables the detection of a lost count condition:

     -Set ADT line (SM200)
     -Set address lines to $00F3_{16}$
     -Set WRITE line (SM202)
     -Clear WRITE line
     -Clear ADT line

2. When the micro code detects the clock interrupt (the clock is enabled and the corresponding mask bit is set), it clears the clock interrupt:

   -Set ADT line
   -Set address lines to either $00F0_{16}$ or $00F3_{16}$.
   -Set READ line (SM201)
   -Clear READ line
   -Clear ADT line

3. Another clock interrupt will occur in 3.33 milliseconds and step 2 above is repeated.

4. Lost count occurs whenever the micro code fails to clear the existing clock interrupt prior to the occurrence of the next clock interrupt in 3.33 milliseconds. Whenever a lost count condition occurs, the most significant bit of the main tristate bus (B' input to the ALU) will be 1 when the clock is selected. To check for a lost count condition, the micro code must:

   -Set address lines to either $00F0_{16}$ or $00F3_{16}$

   -Gate B' input from I/O-TTY board onto the tristate bus using an INRD B-source code

   -Check if the most significant bit of the word just gated off the tristate bus is 1

NOTE

In the case of a 32-bit processor, the word gated from the I/O-TTY board is only 16 bits wide, whereas the main tristate bus is 32 bits wide (BUS00 to BUS31) and the I/O-TTY board is gated onto the lower 16 bits (BUS16 to BUS31) of the main tristate bus. Thus, whenever a lost count condition is present in a 32-bit processor, BUS16 will be 1 instead of BUS00.

5. Whenever the micro code wants to clear the lost count status, it must repeat the WRITE operation as outlined in step 1 above.

6. Whenever the micro code wants to disable the clock circuit so that no further interrupts will be generated, the WRITE operation outlined in step 1 is repeated, except that the address lines are set to $00F0_{16}$.

This section discusses the operating procedure for the micro-programmable computer in general terms. Since each user will have a different equipment application and setup, it is recommended that the user evaluate and develop his own operating procedure. The following sections present a general outline for startup and shutdown actions. Included is a description of the normal operator's interface to the MP.

STARTUP

The following startup sequence is a suggested outline:

1. Power-On Switch. Turn the MP power-on switch to the ON position.
2. Peripheral Power On Sequence. Turn on all peripherals and auxiliary power units.

SHUTDOWN

De-energize all peripherals. Position the power-on switch to the OFF position.

SYSTEM FAILURE

After a system failure, follow the startup procedure and deadstart/autoload for restart.

OPERATOR INTERFACE FOR THE MP

The panel interface card provides manual operator control of the MP. A maintenance panel and/or programmers console may be connected to this card. The following paragraphs discuss the controls necessary at either the maintenance panel or programmers console. This discussion is followed by a description of the functional features of the maintenance panel.

FUNCTION CONTROL REGISTER (FCR)

The function control register (table 3-1) is the basic means of communication between the MP and the operator. The eight hexadecimal digits (32 bits) of the FCR can be grouped as follows (0 is highest order):

Display:	Digits 0 and 1
Machine modes:	Digits 2 to 5
Machine status:	Digits 6 and 7

The display digits determine which individual registers of two groups of registers (identified in table 3-2) can be displayed and/or modified. Digits 2 to 5 of the FCR are used to set such conditions as selective stop/on/off, step/run mode, etc.

The two least significant digits (6 and 7) of the FCR are set by the MP and indicate the machine status, such as overflow on/off, macro storage parity error, protect fault, etc.

NOTES

1. Bits 14_{16} and 15_{16} of the FCR (Enable Console Echo and Enable Auto Display) are mutually exclusive; that is, the operator may select one or the other, but not both simultaneously.
2. Digit 3 of the FCR (bits $0C_{16}$ to $0F_{16}$), Breakpoint, is applicable only if the user has the optional maintenance panel and panel interface card.
3. Unassigned display codes (table 3-2) should be assumed to be undefined.
4. Selecting BP or P/MA (table 3-2) will result in both BP and P/MA being displayed. BP is the leftmost 16 bits and P/MA is the rightmost 16 bits. BP can be modified only if BP is selected; P/MA cannot be modified in either case.

TABLE 3-1. FUNCTION CONTROL REGISTER (FCR)

Bit		Digit	Bit Definition
(LSB) 31	1F	7	Overflow
30	1E		Not Protected Instruction †
29	1D		Protect Fault †
28	1C		Parity Error †
27	1B	6	Interrupt System Active †
26	1A		Auto-Restart Enabled
25	19		Micro Running
24	18		Macro Running †
23	17	5	Not used
22	16		Not used
21	15		Enable Auto Display
20	14		Enable Console Echo
19	13	4	Enable Micro Memory Write
18	12		Multilevel Indirect Addressing Mode †
17	11		Not used
16	10		Suppress Console Transmit
15	0F	3	0 0 Breakpoint Off 0 1 Instruction Reference Breakpoint 1 0 Storage Operand Breakpoint 1 1 All References Breakpoint
14	0E		
13	0D		Breakpoint Interrupt (BP Stop if Clear)
12	0C		Micro Breakpoint, Step, Go, Stop (Macro if Clear)
11	0B	2	Step
10	0A		Selective Stop †
09	09		Selective Skip †
08	08		Protect Switch †
07	07	1	Display 1
06	06		
05	05		
04	04		
03	03	0	Display 0
02	02		
01	01		
(MSB) 00	00		

Status Only

† These FCR bits may be assigned as desired, depending on the micro-processor application. The definitions shown are standard for enhanced 1700 emulation.

TABLE 3-2. DISPLAY CODE DEFINITIONS

Code		Display 1	Display 0
0	0 0 0 0	FCR	F2 (Addressed by N)
1	0 0 0 1	P†	N (MSBs) ††
2	0 0 1 0	I	K (LSBs) ††
3	0 0 1 1		X
4	0 1 0 0	A†	Q
5	0 1 0 1	MIR	F
6	0 1 1 0	BP-P/MA (Display only)	F1 { Addressed by K / Enabled by SM111
7	0 1 1 1	P/MA (Display only)	MEM
8	1 0 0 0	SM1	
9	1 0 0 1	M1	$\overline{\text{RTJ}}$
A	1 0 1 0	SM2	
B	1 0 1 1	M2	
C	1 1 0 0		MM
D	1 1 0 1	A*	
E	1 1 1 0	X*	
F	1 1 1 1	Q*	

† Used to address macro memory. Automatically incremented after each memory reference.

†† The combined contents of the K and N registers are used to address micro memory. The K register is automatically incremented after each memory reference. The N register does not automatically increment.

5. Selecting N or K (table 3-2) results in both N and K being displayed. N is the left eight bits and K is the right eight bits. However, when N is selected, only the N register can be modified; when K is selected, only the K register can be modified.

AUTO-DISPLAY

When auto-display is enabled, the register selected by the control code and display code will be output to the operator's interface and continuously updated (assuming that operator's interface contains a display and not a teletypewriter). With auto-display enabled, depressing a terminator (: or G) with no characters preceding it will cause a go signal.

PANEL INTERFACE CONTROL COMMANDS

The control commands used in the panel interface mode include: H, I, J, K, L, :, G, and ?. Control commands H through L identify the type of data or operation entered or returned. The colon (:) and G perform an entry termination function. The question mark (?) generates a master clear.

A normal entry consists of one control character H through L; two, four, or eight hexadecimal digits 0 through F; and a terminating entry (: or G), in that order.

A normal response consists of the control character identifying the data that follows the four or eight hexadecimal digits. If a transmission or operator error occurs on the entry, an asterisk (*) precedes the control character and the function control register is unconditionally displayed with the last legal control character. All entries except the ? cause a response, unless bit 10_{16} (suppress console transmit) of the FCR is set. The following are examples of the control functions. The colon (:) is used as the terminating entry.

- Master Clear — A master clear can be generated in several ways:

  -A power-on master clear

  -The MC button on the maintenance panel

  -A signal from a peripheral controller

  -A question mark from a panel device (programmers console)

  NOTE

  Baud rate compatibility between the panel device and the machine must exist for ? master clear.

- Stop/Go Control — The following entry will cause a go:

  I: (Initiate)

  This is a micro go if bit 12 of the FCR is set. It is both a micro and macro go if bit 12 of the FCR is clear.

  The I control function may also be used to set a bit in the FCR.

  The following entry will cause a stop:

  H: (Halt)

  This is a micro stop if bit 12 of the FCR is set. It is a macro stop if bit 12 of the FCR is clear.

  The response to a start or stop entry is a display of the FCR.

  The H control function may also be used to clear a specific bit in the FCR. The entry

  H14:

  would clear bit 14_{16} in the FCR and the response would be a display of the updated FCR.

- J Control Function — The J control function is used to replace the contents of the function control register in a digit mode. While it may be used to change the value of any FCR digit, it is generally used to change digits 0 and 1. The value of display 0 and display 1 specifies which MP parameter is displayed on display requests or entered on enter requests. J functions always consist of J followed by two hexadecimal digits and a terminator (: or G). The first hexadecimal digit specifies the FCR digit 0 through 5 and the second hexadecimal digit specifies the value the digit is to assume, 0 through F.

  The function code

  J14:

  would set FCR digit 1 to 4 (select the A register), and the response would be a display of the updated FCR.

  The J code is also used to alternately display the upper and lower 16 bits of a 32-bit register on the 16-bit maintenance panel display.

- K Control Function — The K control function is used to display or enter data into the parameter specified by display 1. The K function uses two formats. The first format is a request to display the parameter specified by display 1:

  K:

  The second format is an enter data request. The data is entered into the parameter specified by display 1. It consists of K followed by four or eight hexadecimal digits, followed by a

terminator (: or G). The hexadecimal digits are the data to be entered. For example:

-To display the P register, type:

J11: Set display 1 to P register (FCR digit 1 = 1_{16}).

K: Display parameter selected in display 1.

-To enter $14FE_{16}$ into the breakpoint register, type:

J16: Set display 1 to breakpoint register (FCR digit 1 = 6_{16}).

K14FE: Enter data into parameter selected in display 1.

- L Control Function — The L function is operationally the same as the K function, except that it is associated with display 0.

NOTE

When macro memory is displayed or entered, the register selected in display 1 is the macro memory address. The display 1 selection must be the P or A register. This register is incremented by 1 after the display. When micro memory is displayed or entered, the K register is the eight least significant bits of the address, and the N register provides the remaining bits. The K register is incremented by 1 after the display.

- Breakpoint (BP) — There are two types of breakpoint: micro and macro. If bit 12 of the FCR is set, micro breakpoint is selected. If bit 12 is clear, macro breakpoint is selected. In the panel simulation mode there is no micro or macro breakpoint capability.

  Bits 14 and 15 of the FCR are used to select three types of macro breakpoint (BP):

Bit 14	Bit 15	
0	0	BP not selected
0	1	Instruction reference BP
1	0	Store operand BP
1	1	All references BP

A macro breakpoint occurs if the breakpoint register is equal to the macro memory address and the select conditions are met. For example:

J16: Set display 1 to breakpoint register.

K0050: Set breakpoint register to 0050_{16}.

J31: Set macro mode and breakpoint on instruction reference.

A stop will occur after the instruction at macro location 50_{16} is executed.

If bit 13 of the FCR is set, an optional interrupt occurs rather than a stop when the breakpoint conditions are met.

For a micro breakpoint, P/MA is compared to the lower 12 bits of the breakpoint register. In addition, the upper/lower selection (32-bit select) is compared to bit 13 of the breakpoint register. If all bits are equal and the combination of FCR bits 14 and 15 is not zero, then a micro stop occurs. If FCR bit 14 is set, then a comparison of FCR bit 13 and the upper/lower selector is not required.

- Auto Display — When auto display is enabled, the register selected by the control and display codes will be output to the operator's interface and continuously updated as long as the interface is a display terminal and not a teletypewriter. Depressing a terminator (: or G) with no characters preceding it will cause a go signal, which is useful for stepping through a micro or macro program.

NOTE

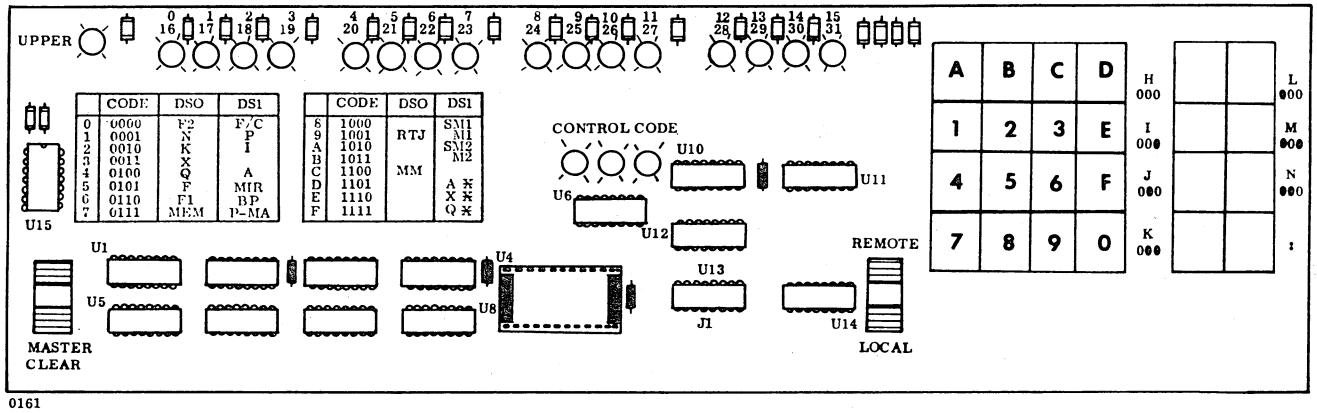
Auto-display mode and echo mode should never be selected simultaneously. In other words, FCR bits 20 and 21 are mutually exclusive.

MAINTENANCE PANEL

Table 3-3 defines and figure 3-1 illustrates the available functional features of the maintenance panel.

TABLE 3-3. MP CONTROLS AND INDICATORS

Panel Device	Type of Device	Function
Data display	16 indicator LEDs	Display current binary data as determined by specific control characters
UPPER	1 indicator LED	Indicates whether the display is upper or lower 16 bits; upper when lit.
CONTROL CODE	3 indicator LEDs	Indicate last control character entered as follows: 0 = H 1 = I 2 = J 3 = K 4 = L 5 = Undefined 6 = Undefined 7 = Error
MASTER CLEAR	Momentary switch	Master clear to CPU, memory, and peripheral controllers
REMOTE/LOCAL	Two-position switch	Enables panel or remote programmer console as follows: REMOTE position = Programmable console is enabled LOCAL position = Panel is enabled
Data entry	16 momentary pushbuttons	For entry of hexadecimal data
Control character	8 momentary pushbuttons	For entry of control character
Function control definition table		Operator assistance information such as function control register definition and control character definition



0161

Figure 3-1. Functional Features of Maintenance Panel

INSTRUCTION FORMAT

Each micro-memory address specifies the location of two micro instructions (upper and lower). Each 32-bit micro instruction is divided into five main sections and is numbered from left to right as bits 0 to 31. Figure 4-1 illustrates the basic MP instruction format.

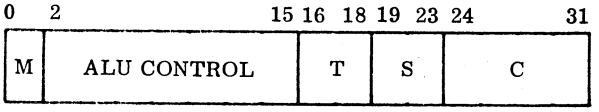


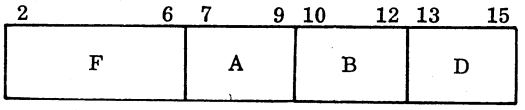
Figure 4-1. Basic MP Instruction Format

The five basic fields of an MP instruction are defined in table 4-1.

The MP instruction field definitions, including the various subfield definitions, are shown in figure 4-2.

The M field specifies one of three addressing modes used to obtain the next micro-instruction pair from micro memory and specifies the format for interpreting bits 19 to 31 of the micro instruction (see table 4-2).

The ALU control fields specify the sources of the two operands on which arithmetic, logical, shift, or scale operation is to be performed and specify the destination of the operation result. For arithmetic and logical operations, the ALU control fields consist of the ALU function (F), A source (A), B source (B), and destination (D) fields as follows:



For shift and scale operations, the A and B fields are interpreted as follows:

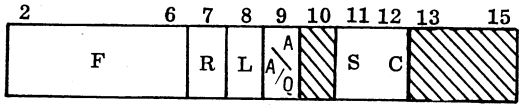
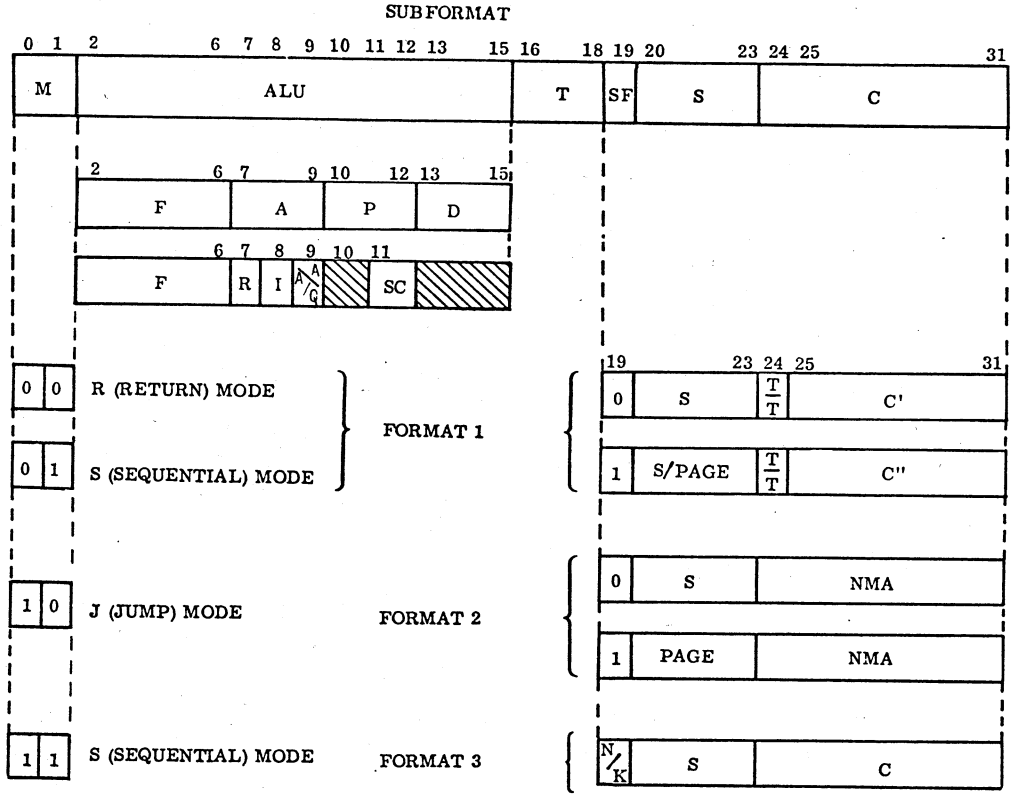


TABLE 4-1. BASIC MP INSTRUCTION FIELDS

Bit Positions	Field Definition
0 and 1	The mode (M) field specifies the addressing used in obtaining the next micro-instruction pair from micro memory and specifies the format used in interpreting bits 19 through 31 (the S and C fields) of the micro instruction.
2 through 15	The ALU control field specifies the arithmetic and logical unit (ALU) operation code, sources of operands, and destination of the operation result.
16 through 18	The test (T) field specifies the method of selecting which micro instruction (upper or lower) of the next micro-instruction pair to execute.
19 through 23	The special (S) field specifies sub-format selection (bit 19) and special operation codes (bits 20 through 23).
24 through 31	The constant (C) or suboperation field (bit 24) specifies constant, jump transform, or micro operation.

The F field specifies shift or scale operation. Bits 7 and 8 specify right or left shifting. Bit 9 specifies whether the A register alone or the A and Q registers together are to be shifted or scaled. Bit 10 is not used, and bits 11 and 12 specify the shift control code. The D field contains a no-operation code for shift and scale operations.

The T field is the conditional branch of the micro instruction. It specifies which micro instruction, upper or lower, of the next micro-instruction pair to execute.



Where:

M is the mode field. It specifies the format of the S and C fields and the sequencing mode used to obtain the next micro-instruction pair.

ALU is ALU control. It specifies ALU operation, sources of operands, and destinations of operations. The arithmetic and logical operations are:

F	Operation
B	B source
A	A source
D	Destination

The shift and scale operations are:

F	Operation
R	Right
L	Left
A	A register
A/Q	A/Q register
SC	Shift control

T is the test field. It specifies the method of selecting which micro instruction of the next micro-instruction pair to execute.

SF is the subformat select bit. It is part of the S field. In format 3, it specifies (N/K) whether the eight bits of the C field are transferred to the N register (bit = 1) or the K register (bit = 0).

S is the special field. It specifies a special operation, such as macro memory read or write, alternate codings to be used in the A, B, and D fields, or a page jump.

PAGE is the page where the next micro-instruction pair is to be found.

C is the constant field. It specifies a constant, jump, transform, or micro operation.

C' is a constant for driving the bit generator, for providing additional information to control the macro memory, or for performing other operations.

C'' is a transform code associated with a transform operation.

NMA is the micro-memory address of the next micro-instruction pair.

Figure 4-2. Detailed MP Instruction Field Definitions

TABLE 4-2. INSTRUCTION MODES

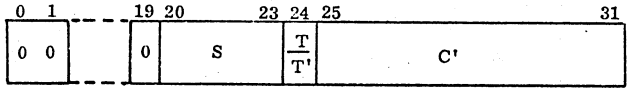
M	Addressing Mode	Bits 19 to 31
00	Return	Format 1
01	Sequential	Format 1
10	Jump	Format 2
11	Sequential	Format 3

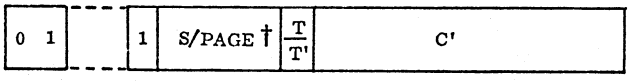
The test may be based on the result of the ALU operation of the current micro instruction or on some other condition.

The codings in the S and C fields depend upon the contents of the M field. The S and C fields are coded in three formats.

FORMAT 1

Format 1 is specified when the M field contains 0 0 (return mode) of 0 1 (sequential mode) as follows:



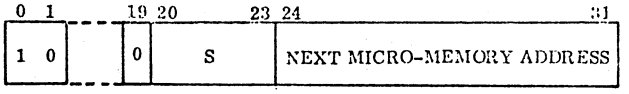


The S field specifies such operations as macro memory read or write and alternate codings to be used in the A, B, and D fields. The T/T' bit specifies that the code in the T field is to be interpreted as the normal T code (T/T' equals 0) or as the alternate T' code (T/T' equals 1). The subformat select bit, bit 19, determines whether bits 25 through 31 are to be interpreted as C' codes or as C'' codes. The C' code can contain a constant for driving the bit generator, additional information to control the macro memory read or write, or another operation. The C'' codes are associated with transforms.

†If SM207 is set, normal S-field commands are disabled and the S field then specifies a micro-memory page for the TMA/j, TMAK/j, GITMAK/j, and GITMAK/XT commands. For a TK/j command, a page is not specified and the S field is not decoded (if SM207 is set). Refer to the definition of SM207.

FORMAT 2

Format 2 is specified when the M field contains 1 0 (jump mode) as follows:

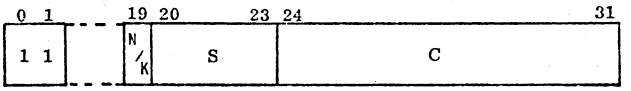


If a jump is specified to a micro-instruction pair within the same micro-memory page, the subformat select bit, bit 19, is 0 and bits 20 to 23 contain a special operation code as in format 1. Bits 24 through 31 contain the micro-memory address of the next micro-instruction pair.

The subformat select bit is 1 when a jump is specified to a different micro-memory page; bits 20 through 31 contain the complete micro-memory address of the next micro-instruction pair.

FORMAT 3

Format 3 is specified when the M field contains 1 1 (sequential mode), as follows:



This format allows one special operation to be performed, as specified by the S code, and also causes the eight bits of the C field to be transferred to the N register (bit 19 = 1) or to the K register (bit 19 = 0).

DESCRIPTION OF INSTRUCTIONS

The detailed coding for each field of the micro instruction is described in the following paragraphs. This coding pertains generally to the 16-bit and 32-bit processors. Coding exclusive to the 32-bit processor is so noted.

ALU CONTROL FIELDS

F Field

The F field, bits 2 through 6, specifies the logical or arithmetic operation performed by the ALU or the shift or scale operations performed with the A and Q registers.

Logical Operations

The logical operations are described in table 4-3.

TABLE 4-3. LOGICAL OPERATIONS

F Code	Mnemonic	A input = 0 0 1 1 B input = 0 1 0 1 Bit Result
0 1 1 0 0	ZERO	0 0 0 0
0 1 1 1 0	A • B	0 0 0 1
0 1 1 0 1	A • -B	0 0 1 0
0 1 1 1 1	A	0 0 1 1
0 1 0 0 0	-A • B	0 1 0 0
0 1 0 1 0	B	0 1 0 1
0 1 0 0 1	EOR	0 1 1 0
0 1 0 1 1	A+B	0 1 1 1
0 0 1 0 0	-A • -B	1 0 0 0
0 0 1 1 0	-EOR	1 0 0 1
0 0 1 0 1	-B	1 0 1 0
0 0 1 1 1	$A+^{-B}$	1 0 1 1
0 0 0 0 0	-A	1 1 0 0
0 0 0 1 0	-A+B	1 1 0 1
0 0 0 0 1	$-A+^{-B}$	1 1 1 0
0 0 0 1 1	ONE	1 1 1 1

Arithmetic Operations

The arithmetic operations listed in table 4-4 operate on either single-precision operands (using the main ALU) or double-precision operands (if the double-precision option is present and the double-precision flag is set in the SM register). Two additional options are provided and coded in the arithmetic function code. The first option is a carry-in to the adder (indicated by a plus sign in the micro-instruction mnemonic). The T-field code checks for a carry-out of the ALU to determine whether a carry-in to the ALU should be used on the next arithmetic operation. With the double-precision bit set in the SM register, the carry-in is entered in the lower bit of the double-precision ALU; otherwise, the carry-in is entered in the main ALU.

The second option allows the capture of the overflow condition in the SM register, indicated by a T in the instruction mnemonic. The overflow test is performed by comparing the sign of the two inputs with the results of the ALU and setting a status/mode bit when the result is inconsistent. The SM overflow bit is set to 1 when the overflow occurs; it must be set to 0 by a micro instruction that sets the SM bit to 0.

Shift Operations

The shift operations in the F field (1 1 1 1 0) specify a shift of the A register or the A/Q register of the main MP organization only; no shift is possible in the double-precision registers from this command. The ALU is not used to perform the shift, but performs some operation based on its decoding of the F field (which should be considered unknown). The destination will receive this meaningless output unless an NOP is chosen for the destination of the D field.

The type of shift is determined by the coding in bits 7 to 12 of the micro instruction, and the amount of the

TABLE 4-4. ARITHMETIC OPERATIONS

F Code	Mnemonic	Operation
1 0 1 0 0	SUB	Subtract B input from A input.
1 1 0 0 0	ADD	Add A and B inputs.
1 0 1 0 1	SUBT	Subtract with an overflow test.
1 1 0 0 1	ADDT	Add with an overflow test.
1 0 1 1 0	SUB-†	Perform A-B-1 input (twos complement only).
	SUB-C †	A-B with forced carry-in (ones complement only).
1 1 0 1 0	ADD+ †	Perform A + B (forced carry-in).
1 0 1 1 1	SUB-T	Perform SUB- with an overflow test (twos complement only).
	SUB-TC†	A-B with forced carry-in (ones complement only).
1 1 0 1 1	ADD+T †	Perform ADD+ with an overflow test.

†If split adder mode is selected, the most significant (upper) adder performs an ADD or SUB without forced carry-in. Forced carry-in is defined as an unconditional hardware logical 1 (used for twos complement arithmetic) gated to the (hardware) adder carry-in input.

shift is determined by the number contained in the N register. The operation examines the N register and, if it is zero, the shift operation is terminated and the next micro instruction is executed. The T-field codes that can be used with a shift are U, L, KZU, INTU, OVFL, K7L, and BTU. Other T codes cannot be used.

If the N register is not zero, a shift of one bit position is taken as specified, N is decremented by one, and the test for zero is repeated as above.

The shift conditions are as follows:

- Shift A — Shift the A register only.
- Shift A/Q — Shift the combined A and Q register. The Q register has the least significant bits of the combined A/Q register.
- Shift left or right.
- Enter 0 — Enter a 0 in the vacated bit position at the end of the register.
- Enter 1 — Enter a 1 in the vacated bit position at the end of the register.
- Extend sign — Extend the sign (for a right shift only).
- End-around carry — Enter the bit coming off the end of the register into the vacated position at the other end.

The shifts are defined in table 4-5.

TABLE 4-5. SHIFT OPERATIONS

Bit Code					Mnemonic	Operation
7	8	9	11	12		
1	0	0	0	0	AR0E	A is right-shifted (N) bits, with 0 entered as the most significant bit.
1	0	0	0	1	ARSE	A is right-shifted (N) bits, with sign extension.
1	0	0	1	0	AREA	A is right-shifted (N) bits, with end-around carry.
0	1	0	0	0	AL0E	A is left-shifted (N) bits, with 0 entered as the least significant bit.
0	1	0	0	1	AL1E	A is left-shifted (N) bits, with 1 entered as the least significant bit.
0	1	0	1	0	ALEA	A is left-shifted (N) bits, with end-around carry.
1	0	1	0	0	AQR0E	A/Q is right-shifted (N) bits, with 0 entered as the most significant bit in A.
1	0	1	0	1	AQRSE	A/Q is right-shifted (N) bits, with sign extension.
1	0	1	1	0	AQREA	A/Q is right-shifted (N) bits, eith end-around carry.
0	1	1	0	0	AQL0E	A/Q is left-shifted (N) bits, with 0 entered as the least significant bit in Q.
0	1	1	1	0	AQLEA	A/Q is left-shifted (N) bits, with end-around carry.

NOTE: (N) = Contents of register N.

Scale Operations

Scale operations are similar to shift operations, but the shift is stopped by bits A00 and A01 not being equal. (The scale point is normally between bits 0 and 1 of the A register. A hardware design option allows the scale point to be specified between different bits in the A register when necessary for efficient floating-point emulation.) The maximum number of bits to scale is contained in the N register and, on completion of the scale, N is decremented by the number of shifts necessary to scale the number.

The scale operation is performed as follows:

1. Examine N. If it is zero, exit the micro instruction.
2. Examine bits 0 and 1 of the A register. If they differ, exit the micro instruction.
3. Shift the A or A/Q register left by one bit position as specified in the instruction.
4. Decrement the N register by one count and go to step 2.†

The scale operation is coded in bits 7 through 12 of the micro instruction in the same number as the shift operation, and allows the same left shift options. When the N register is zero, the scale operation is terminated and the next micro instruction is executed. The usable T field codes for the scale operation are U, L, KZU, INTU, OVFL, K7L, and BTU. Other T codes cannot be used.

All scale operations are performed with an F code of 1 1 1 1 1. The scales are given in table 4-6.

TABLE 4-6. SCALE OPERATIONS

Bit Code					Mnemonic	Operation
7	8	9	11	12		
0	1	0	0	0	SL0E	A is scaled left, with 0 entered as the least significant bit.
0	1	0	0	1	SL1E	A is scaled left, with 1 entered as the least significant bit.
0	1	0	1	0	SLEA	A is scaled left, with end-around carry.
0	1	1	0	0	SDL0E	A/Q is scaled left, with 0 entered as the least significant bit in Q.
0	1	1	1	0	SDLEA	A/Q is scaled left, with end-around carry.

A Field

The A field, bits 7 through 9, specifies the input to S1 and thus to the A side of the ALU.

The eight A codes in table 4-7 are used as input if the S field is not 1 0 1 0 or 0 1 1 1. If the S field is 1 0 1 0 or 0 1 1 1, the eight A' codes in table 4-8 are used as input.

†If the number being scaled is all 0s or all 1s (i.e., the number cannot be scaled), then the scale operation is terminated when N = FE_{16} (after passing through N = 0). To avoid exiting the micro instruction before the scale operation is actually completed, N should be at least equal to the number of bits in the word to be scaled.

B Field

The B field, bits 10 through 12, specifies the input to S2 and thus to the B side of the ALU.

Codes available for the B field are the B codes and the B' codes. Eleven B codes are defined for the S field not equal to 1 0 0 0. Code 0 0 1 of the B field is expanded by micro-instruction bits 28 and 29 for enabling the N or K register to S2. As long as there is no conflict, the input for the N and K registers may be used in conjunction with commands or constants in the C field. The B and B' codes are given in tables 4-9 and 4-10, respectively.

D Field

The D field, bits 13 through 15, specifies the destination of the information from the main organization of the MP. The four sources of this information are:

- An optionally shifted ALU output; this shifting occurs in the S3 shift network that connects the ALU output to the P, A, F, X, and Q registers and external destination (e.g., I/O).
- The output of the ALU
- The A source
- The B source

TABLE 4-7. A INPUT OPERATIONS

A Code	Mnemonic	Operation
000	F2†	Use the contents of the file 2 register as the A source input. The current value of the N register is used to address register file 2. If the value of N is changed in the current micro instruction, its initial value is used to reference the file register. F2 must not have been written during the previous instruction.
001	P	Use the contents of the P register as the A source.
010	I	Use the contents of the I register as the A source.
011	X	Use the contents of the X register as the A source.
100	A	Use the contents of the A register as the A source.
101	F	Use the contents of the F register as the A source.
110	F1†	Use the contents of the optional file 1 register or external source as the A source. The current value of the K register is used to address register file 1. If the value of K is changed in the current micro instruction, the initial value of K is used to reference the file register. SM111 controls the selection of F1/external. F1 must not have been written during the preceding micro instruction.
111	MEM††	Obtain data read from macro memory and use it as the A source.

† Restriction: The value of the addressing register (N or K) cannot have been modified by a C' increment or decrement command in the preceding micro instruction.

†† Restriction: If the macro memory READ command was not given in the preceding micro instruction, all 1s are input to the A source. If the B source is a prime code, the B source data will also be input to S1. This command is restricted to a micro instruction, with type A, B, or C execution time.

TABLE 4-8. A' INPUT OPERATIONS

A' Code	Mnemonic	Operation
000	SM1	Use the contents of SM register 1 as the A source.
001	M1	Use the contents of interrupt mask register 1 as the A source.
010	SM2	Use the contents of SM register 2 as the A source.
011	M2	Use the contents of interrupt mask register 2 as the A source input.
100	A*R8	Use the contents of the double-precision A* register, shifted right eight bits with end-around carry, as the A source. The A* register remains unshifted.
101	A*	Use the contents of the double-precision A* register as the A source.
110	X*	Use the contents of the double-precision X* register as the A source.
111	Q*	Use the contents of the double-precision Q* register as the A source.

Note: The A' codes are specified by the S field equal to 0 1 1 1 or 1 0 1 0.

TABLE 4-9. B CODES

B Code	MIR28 - MIR29	Mnemonic	Operation
000		F2 †	Use the contents of the file 2 register as the B source. The value of the N register, before the instruction is executed, is used to address register file 2. If the value of N is changed in the current micro instruction, its initial value is used to reference the file register. F2 must not have been written during the previous instruction.
001	1 1	Zero	The B source is all zeros.
001	1 0	N ††	Use the contents of the N register as the B source. Since N is an eight-bit register, this source uses N as the upper eight bits and zeros as the lower bits.
001	0 1	K ††	Use the contents of the K register as the B source. Since K is an eight-bit register, the upper bits are zeros and K serves as the lower eight bits.
001	0 0	N, K ††	Use the contents of the N and K registers as the B source. These registers are combined with the N register as the upper eight bits of source and with K as the lower eight bits.
010		BG	Use the contents of the BG register as the B source. The register has only one bit set to 1, and the position of the bit in the BG register is specified either on value in the N register or by a number in the C field, depending on the state of the controlling SM register bit.
011		X	Use the contents of the X register as the B source.
100		Q	Use the contents of the Q register as the B source.
101		F	Use the contents of the F register as the B source.
110		F1	This code is similar to F2, but it uses the contents of optional file 1 register addressed by (K) or an external source as the B source input. SM111 controls the selection of F1/external. F1 must not have been written during the preceding micro instruction.
111		MEM †††	This code obtains data read from macro memory and uses it as the B source.

† Restriction: The value of the addressing register (N or K) cannot have been modified by a C' increment or decrement command in the preceding micro instruction.

†† The most significant 16 bits of the 32-bit processor are zeros. These codes will control only the two lower-order eight-bit bytes in the 16 least significant bits of the B source input in a 32-bit processor.

††† Restriction: If the macro memory READ command was not given in the preceding micro instruction, all ones are input to the B source. Exception: If the A source is a prime code, the A prime code source data will be inputted to S2. This command (MEM) is restricted to a micro instruction with type A, B, or C execution time.

All D destinations, except the I register, are optionally shiftable by S3 when specified by a code in the C field or by the L8EA command in the S field if alternate codings D' or DD" are not specified. The I destination differs from the others in that the output of the A source is the input to the I register. The codes and their operations are given in table 4-11.

The D' destinations are specified by the D field when the S field is set to 1001 or 1010. The codes and action are given in table 4-12.

The D" destinations are specified by the D field if the S field is set to 1011. These destinations transfer data to the double-precision logic from S1. The codes and actions are given in table 4-13.

TABLE 4-10. B' CODES

B' Code†	Mnemonic	Operation
000	OPEN	
001	CRTJ	Transfer the complement of the RTJ register to the 12 least significant bits of S2. Transfer 1s to the four most significant bits of S2.
010	INRD	Input data/status from the I/O channel.
011	INRS	Input to S2 I/O response signals.
100	MMU	Transfer the upper 16 bits of data from the micro memory to the X register in the 16-bit processor. The 32-bit processor transfers the total 32-bit word. The F field must make a reference to the B source. The address is specified by transform or NK. The D field must be an NOP. (See micro-memory operand references section for further details.)
101	MML	Transfer the lower 16 bits of data from the micro memory to the X register in the 16-bit processor. This code is not operative in a 32-bit processor.
110 or 111	INTA††	Use the contents of the interrupt address encoder as the B source. The output of this encoder represents the complement of the interrupt address of the highest priority interrupt line that is active having its corresponding mask bit set.

† The B' codes are specified by the S field equal to 1000.

†† Restriction: An INTU test command must be given in the preceding micro instruction.

TABLE 4-11. D CODE TRANSFERS

D Code	Mnemonic	Operation
000	NOP	Do not transfer data to any destination.
001	P†	Transfer output of S3 to P, AB (macro memory address buffer register).
010	I	Transfer output of S1 to I, AB.
011	Q	Transfer output of S3 to Q, AB.
100	F1††	Transfer output of S3 to F register, AB, and write this data in file 1 at the address specified by K at the completion of this instruction.
101	A	Transfer output of S3 to A, AB.
110	X	Transfer output of S3 to X, AB.
111	F	Transfer output of S3 to F, AB.

† If a D-field command to load AB is issued in the next micro instruction following the micro instruction with this command, the transfer to AB is inhibited.

†† Data is written into the file 1 register during the first part of the next micro instruction, taking advantage of the updated value of K from this micro instruction. The next micro instruction must not specify a read of file 1.

TABLE 4-12. D' CODE TRANSFERS

D' Code	Mnemonic	Operation
000	IOD	Transfer the output of S3 to the I/O data register.
001	IOA	Transfer the output of S3 via the I/O data register to the I/O address register. It destroys the contents of the I/O data register.
010	MMU	Transfer the output of S2 to the upper 16 bits of micro memory in the 16-bit processor, or transfer the output of S2 to the 32-bit word in micro memory in the 32-bit processor. (See micro-memory operand reference section for further details.)
011	MML	Transfer the output of S2 to the lower 16 bits of micro-memory location in the 16-bit processor, or transfer the output of S2 to the 32-bit word in micro memory in the 32-bit processor. (See micro-memory operand reference section for further details.)
100	M1	Transfer the output of ALU to mask register 1.
101	SM1	Transfer the output of ALU to SM register 1.
110	M2	Transfer the output of ALU to mask register 2.
111	SM2	Transfer the output of ALU to SM register 2.

Note: Outputs to the mask and SM registers are direct from the ALU and are not shiftable.

TABLE 4-13. D'' CODE TRANSFERS

D'' Code	Mnemonic	Operation
000	NOP	Do not transfer data to any destination.
001	A*LHW	Transfer the output of S1 to the A* register, shifted left one-half word, with end-around carry.
010	X*LHW	Transfer the output of S1 to the X* register, shifted left one-half word, with end-around carry.
011	Q*LHW	Transfer the output of S1 to the Q* register, shifted left one-half word, with end-around carry.
100	NOP	Do not transfer data to any destination.
101	A*	Transfer the output of S1 to the A* register.
110	X*	Transfer the output of S1 to the X* register.
111	Q*	Transfer the output of S1 to the Q* register.

The DD'' option allows the performance of an operation on the A, X, or F register. This changes the register but keeps a copy of the original register in a double-precision register. This is another way of getting data to the double-precision registers. The DD'' option is specified when the S field contains 0001. Table 4-14 lists the DD'' codes and their operations.

TABLE 4-14. DD'' CODES

DD'' Code	Mnemonic	Operation
101	AA*	Transfer the output of S3 to the A register, and transfer the output of S1 to the A* register.
110	XX*	Transfer the output of S3 to the X register, and transfer the output of S1 to the X* register.
111	FQ*	Transfer the output of S3 to the F register, and transfer the output of S1 to the Q* register.

T Field

The T field, bits 16 through 18, selects the upper or lower micro instruction of the next micro-instruction pair to execute. This may be a conditional selection or an unconditional selection, depending on the T-field code. This field is available to all addressing modes in addition to I/O operations. A conditional selection may test the ALU output, the value of certain registers, certain internal conditions such as interrupts, and particular bits wired to the transform board. The only exception to these uses of the T field is when micro-memory data is being read or written; in these cases, the T field is used as part of the micro-memory data reference address, and the upper instruction in the next sequential micro-instruction pair is always selected.

The T-field codes consist of two groups: T codes and T' codes. Similar to the A, B, and D fields that are extended by using the S field, the T field is always extended in the following sense: Bit 24 of format 1 micro instructions is either 0 or 1 if T or T', respectively, is specified. The T' codes are available only for the return and sequential addressing modes (format 1). The T/T' codes select the upper or lower portion of the next micro-instruction pair as the next micro instruction to execute. The T codes are listed in table 4-15; the T' codes are listed in table 4-16.

FORMAT MODES

The format modes are described in the following paragraphs.

M Field

The M field (bits 0 and 1) defines the major operation taking place in the micro instruction and specifies the type of sequencing that will be used to obtain the next instruction pair. The operations specified in the M field are listed in table 4-17.

Subformat Select Bit (SF)

Bit 19 (SF) is used to select either variation in format 1 and format 2 decoding or the choice of loading the N or K register in format 3.

S Field

The S field (bits 20 through 23) of the micro instruction is used to specify a special command (including alternate codings in the A, B, and D fields) in addition to page or constant information (as required by the code in the C field). The S codes specify action that occurs at the same time as the ALU operation specified in the F, A, B, and D fields. The codes and operations are given in table 4-18.

C Field

The C field (bits 24 through 31) is used to specify an additional special operation, an address for a jump, or a constant for setting the K or N register. Bit 24 in format 1 specifies the T-field interpretation. The codes for this field are listed in tables 4-19 through 4-21.

MICRO-INSTRUCTION TIMING

The basic MP micro-instruction execution time is 168 nanoseconds. Some micro instructions have longer

TABLE 4-15. T ADDRESSING MODES

T Code	Mnemonic	Operation
000	*L	Execute the lower micro instruction of this pair as the next micro instruction. This operation over-rides the M field addressing mode.
001	U	Execute the upper micro instruction of the next micro-instruction pair.
010	L	Execute the lower micro instruction of the next micro-instruction pair.
011	KZU†	If the initial contents of the K register are zero, execute the upper micro instruction of the next micro-instruction pair; otherwise, execute the lower micro instruction. If the decremented K command is included in the same micro instruction, K will contain all 1s on satisfying the zero test.
100	NZU†	If the initial contents of the N register are zero, execute the upper micro instruction of the next micro-instruction pair; otherwise, execute the lower micro instruction. If the decremented N command is included in the same micro instruction, N will contain all 1s on satisfying the zero test.
101	INTU	If there is an interrupt and its corresponding interrupt mask bit is set, execute the upper micro instruction of the next micro-instruction pair; otherwise, execute the lower micro instruction of the next micro-instruction pair.
110	NU	If the sign bit of ALU output is negative on completion of this micro instruction, execute the upper micro instruction to the next micro-instruction pair; otherwise, execute the lower micro instruction of the next pair.
111	ZL	If the output of ALU is zero on completion of this instruction, execute the lower micro instruction of the next micro-instruction pair; otherwise, execute the upper micro instruction of the next pair.

†Restriction: These T-field commands cannot be used in a micro instruction with a C-field INCK or INCN command.

execution times to allow certain operations to be completed. The micro instructions have been grouped according to execution times as types A through G. Execution time may vary ±4 percent over the temperature range (40° F to 120° F) and over the voltage range (±5v).

The classification of micro instructions is shown in table 4-22. This table provides execution time by types for all legal combinations of micro commands that extend the basic cycle time. The instruction type can be ascertained by examining its operation and checking the table.

Exceptions to table 4-22 are:

- The combination of ones complement arithmetic with an ADD+ or ADD+T arithmetic operation is classified as a type B rather than a type C.
- The MMU and MML B' commands are included under read/write micro-memory operand commands; therefore, they are a type F rather than a type B.
- A type G micro instruction followed by a micro instruction with a GITMAK command (C field) has an additional execution time of 185 nanoseconds (typical); thus, the total execution time becomes 625 nanoseconds.

Analysis of a micro program for execution time starts by classifying each of the micro instructions as to type. This is done by using the micro-instruction classification table or by examining the assembler output listing.

TABLE 4-16. T' ADDRESSING MODES

T' Code	Mnemonic	Operation
000	*L	Execute the lower micro instruction of this micro-instruction pair. This operation overrides the M field addressing mode.
001	LQL	If at the start of the micro instruction, the LSB of Q is 1, execute the lower micro instruction of the next micro-instruction pair. Otherwise, execute the upper micro instruction.
010	K7L	If the LSB of the K register is set, execute the lower micro instruction of the next micro-instruction pair. If clear, execute the upper micro instruction.
011	OVFL	If overflow exists, execute the lower micro instruction of the next micro-instruction pair; if not, execute the upper micro instruction.
100	BTU	Bit test — The lower-order four bits in the C field of this micro instruction specify a setting of the bit test selector. If the bit at this position is 1, execute the upper micro instruction of the next micro-instruction pair; otherwise, execute the lower micro instruction of the next micro-instruction pair. The bit test is a general-purpose testing facility that allows wiring any bit of organization available on the machine's backpanel to the bit test selector. This wiring is defined on the transform board.
101	LQ*L	If at the start of this micro instruction, the LSB of Q* is 1, execute the lower micro instruction of the next micro-instruction pair 1; otherwise, execute the upper micro instruction.
110	COL	Carry-out lower — If the carry-out of ALU results from an arithmetic operation, execute the lower micro instruction of the next micro-instruction pair; otherwise, execute the upper micro instruction. This instruction allows a test for carry-out of ALU during multiple-precision arithmetic.
111	Z*L	If the output of ALU* is 0 on completion of this instruction, execute the lower micro instruction of the next pair; otherwise, execute the upper micro instruction.

TABLE 4-17. M FIELD OPERATIONS

M Code	Mnemonic	Operation
00	R	Select the next micro-instruction pair from the address contained in the RTJ register. Use format 1 for special operations.
01	S	Select the next micro-instruction pair in the current page from the address contained in MAC (normally the next sequential pair, unless suppressed by T-field coding). Use format 1 for special operations.
10	J	A jump or page jump. Select the next micro-instruction pair from the address specified by bits 24 to 31 of this micro instruction. The address is in the current MM page if bit 19 is 0 or from the page specified in bits 20 through 23 if bit 19 is 1. Use format 2 for special operations.
11	S	Transfer bits 24 through 31 of this micro instruction to the N or K register as specified by bit 19. The N register is specified if bit 19 is 1, and the K register is specified if bit 19 is 0. Select the next micro-instruction pair in the current page from the address contained in MAC (normally the next sequential micro-instruction pair). Use format 3 for special operations.

TABLE 4-18. S FIELD CODES

S Code	Mnemonic	Operation
0000	NOP	No operation for S field
0001	DD	Alternate D field coding, DD''
0010	RPT	If the N register is not equal to 0, selection of the next micro-instruction pair is inhibited and the current micro-instruction pair is the next micro-instruction pair. The N register is decremented by one. Normal T-field selection applies. If the N register is equal to 0, the normal next micro-instruction pair is used.
0011	READ†	Read the macro memory command. Read the word from macro memory at the address contained in the address buffer (AB). Instruction execution is delayed until a resume is received from memory acknowledging command.
0100	WRITE††	Write macro memory. Transmit the output of S3 to macro memory as data to be written at the address contained in AB. Instruction execution is delayed until a resume is received from memory acknowledging command. Data is stored in memory at the completion of this instruction.
0101	L8EA	The output of ALU via S3 is shifted left eight bits, end-around.
0110	F2WR	Write data contained in the F register into file 2 at the address specified by the contents of the N register at the beginning of the current micro instruction. The actual writing takes place during the first part of the instruction.
0111	AP	Alternate A field coding, A'
1000	BP	Alternate B field coding, B'
1001	DP	Alternate D field coding, D'
1010	APDP	Alternate A and D field coding, A'D'
1011	DPP	Alternate D field coding, D''
1100	GATEI	Gate the output of S1 to the I register.
1101	HALT	If the halt bit of the SM register is 1, stop operation of the MP on completion of this micro instruction. When the start signal is received, continue with the micro instruction specified by the addressing mode and T field. If the halt bit is 0, continue with micro-instruction sequencing.
1110	RTJ	Transfer the address of the next sequential micro-instruction pair to the RTJ register. This is done regardless of the actual addressing mode used in this instruction.
1111	CLRNP	Clear the N register and page register.

† Restrictions: AB must be loaded in a previous micro instruction (D field command). If D field command reloading AB is issued in the same micro instruction as the READ command, the new data will not be loaded into AB until the completion of the read portion of the cycle. A READ command can only be given in a micro instruction with type A, B, C, or D execution time. Memory data must be input to the system in the following micro instruction with type A, B, or C execution time only.

†† Restrictions: AB must be loaded in a previous micro instruction (D field command). If D field command reloading AB is issued in the same micro instruction as the WRITE command, the new data will not be loaded into AB until completion of the WRITE portion of the cycle.

TABLE 4-19. C' CODE ACTIONS

C' Code	Mnemonic	Operation †
00xxxxx		xxxxx is a constant for use in driving the bit generator or in any other commands using the lower five bits of instruction for control.
0100000 0100001 0100010 0100011	WRCH/0 WRCH/1 WRCH/2†† WRCH/3††	Write the eight-bit character specified from the output of S3 at the memory address specified by the output of AB. See the WRITE command in the S field for details of operation and restrictions. Character 0 is bits 0 through 7 (MSBs); character 1 is bits 8 through 15, etc. The remainder of the word in memory is unchanged. The character is not repositioned in the WRITE command.
0100100	RMW	Read modify write — Perform a read of information from the macro memory in the same manner as the READ instruction in the S field. The memory system performs a read cycle and locks up before performing the write cycle. Memory must be forced to complete the write cycle by issuing the WRITE instruction, using the same address, before it will respond to any other read operations.
0100101	WRHW0	Write bits 0 through 15 (MSBs) from the output of S3 at the memory address specified by the output of AB. Bits 16 through 31 (in the 32-bit processor) in memory are unchanged. See the WRITE command in the S field for details of operation and restrictions.
0100111	WRHW1††	Write bits 16 through 31 (LSBs) from the output of S3 at the memory address specified by the output of AB. Bits 0 through 15 (in the 32-bit processor) in memory are unchanged. See the WRITE command in the S field for details of operation and restrictions.
0101000	WRPB	Write protect bit.
011xxxx	GATEIXT	Open for special applications. General-purpose strobe at T4 time
1000101	INCK	Increment the number contained in the K register by one.
1001101	INCN	Increment the number contained in the N register by one.
1000100	DECK	Decrement the number contained in the K register by one.
1001100	DECN	Decrement the number contained in the N register by one.
1000000	CLRK	Clear the K register.
1001000	CLRN	Clear the N register.
101xxxx	SETF/j	xxxx is the value of j, from 0 to 15. Set the SM register flag j to 1.
110xxxx	CLRF/j	xxxx is the value of j, from 0 to 15. Clear the SM register flag j to 0.
1110000 or 1110001	RQLXN†††	The destination register (P, A, F, or X) and Q register are considered as one double-length register with the Q register as the lower order bits. The combined register is shifted left one bit position with the complement of ALU sign bit entered into the lowest bit position of the Q register.
1110011	RQR1E†††	Shift the combined destination and Q register right one bit, and enter 1 in the sign position of the destination register. This command is used in multiply iteration.

† These special operations for format 1 have a 0 in bit 19 (SF) to specify C' codes (bits 25 through 31). Bit 24 specifies the T field interpretation.

†† Commands applicable to the 32-bit processor only.

††† This operation cannot be performed when the S field command (L8EA) is selected.

TABLE 4-19. C' CODE ACTIONS (Continued)

C' Code	Mnemonic	Operation†
1110010	RQR0E††	Shift the combined destination and Q register right one bit, and enter 0 in the sign position of the destination register. This command is used in multiply iteration.
1110100	RL0E ††	Shift the destination register left one bit, entering 0 in the lowest bit position of the register. This operation cannot be performed when Q is the destination register.
1110101	RL1E ††	Shift the destination register left one bit, entering 1 in the lowest bit position. This operation cannot be performed when Q is the destination register.
1110110	RR0E ††	Shift the destination register right one bit, entering 0 in the sign position of the register. This operation cannot be performed when Q is the destination register.
1110111	RR1E ††	Shift the destination register right one bit, entering 1 in the sign position. This operation cannot be performed when Q is the destination register.

† These special operations for format 1 have a 0 in bit 19 (SF) to specify C' codes (bits 25 through 31). Bit 24 specifies the T field interpretation.

†† This operation cannot be performed when the S field command (L8EA) is selected.

TABLE 4-20. C'' CODE ACTIONS

C'' Code†	Mnemonic	Operation
000xxxx	TMA/j	xxxx = j, with values from 0 to 15. Obtain the next micro-instruction pair from the address specified by MA transform selector setting j.
001xxxx	TMAK/j	xxxx = j, with values from 0 to 15. Obtain the next instruction pair from the address specified by MA transform selector setting j. Also, set the K register to the value specified by K transform selector setting j.
0100xxx	GITMAK/j††	Gate the output of macro memory to the IXT register (on the transform module) and perform the TMAK/j operation. Note that j = xxx with values of 0 to 7.
0101xxx	GITMAK/XT ††	Gate the output of macro memory to the IXT register (on the transform module) and perform a transform of the upper 16 bits of MIR; then select one of eight transforms of the MA from the decoded and encoded macro instruction loaded into IXT.
011xxxx	TK/j	xxxx = j, with values from 0 to 15. Set the K register to the value specified by K transform selector setting j.
100xxxx	TN/j	xxxx = j, with values from 0 to 15. Set the N register to the value specified by N transform selector setting j.

† The special operations for format 1 have a 1 in bit 19 (SF) to specify the C'' codes (bits 25 through 31). Bit 24 specifies the T field interpretation.

Note: If the select page XT/MA bit (SM207) is set to a 1, the MA transform will use the S field as a page to allow the MA transforms to execute across a page boundary; the TK transform will disregard the S/page field. See the discussion of transforms and the transform module in section 2.

†† Restrictions: This command must be executed in the micro instruction following a READ command. This command is applicable only if the configuration includes the required specialized transform module hardware.

TABLE 4-20. C'' CODE ACTIONS (Continued)

C'' Code†	Mnemonic	Operation
101xxx	SUB ††	Set upper bounds — Transfer the output of S3 to the upper bounds register in macro memory interface.
110xxxx	SLB††	Set lower bounds — Transfer the output of S3 to the lower bounds register in macro memory interface.

† The special operations for format 1 have a 1 in bit 19 (SF) to specify the C'' codes (bits 25 through 31). Bit 24 specifies the T field interpretation.

Note: If the select page XT/MA bit (SM207) is set to a 1, the MA transform will use the S field as a page to allow the MA transforms to execute across a page boundary; the TK transform will disregard the S/page field. See the discussion of transforms and the transform module in section 2.

††This code is used if the optional program protect logic is included.

TABLE 4-21. C FIELD ACTIONS

C Field	Mnemonic	Operation
Value †	K = value	When micro-instruction bit 19 = 0, transfer the C field value (bits 24 to 31) to the K register and execute the next sequential micro-instruction pair.
Value †	N = value	When micro-instruction bit 19 = 1, transfer the C field value (bits 24 to 31) to the N register and execute the next sequential micro-instruction pair.
Number††	Number	Number is the address of the next instruction pair. If a page jump is required (bit 19 = 1), the S field contains a page setting instead of the S field code.

† These format 3 codings are for setting K and N registers; this format has the M field bits 0 and 1 set to 11, while bit 19 selects the register.

†† These format 2 codings are used to perform a jump, specified if the M field (bits 0 and 1) is 10.

Each micro instruction with a macro memory read or write command has a 440-nanosecond (typical) execution time regardless of the type of cycle (assuming there is no DMA activity in the macro memory interface). However, type E and F micro instructions cannot contain a macro memory read command. An additional execution time of 185 nanoseconds (typical) is required on all read commands followed by a micro instruction containing a GITMAK command.

The total execution time required is then calculated from one macro memory command to the next. If the total time, starting with a micro instruction with a read or write command and including all micro instructions up to the following read or write command, is less than the memory cycle time (600 nanoseconds), 600 nanoseconds is the execution time for that sequence. If the total time is greater than 600 nanoseconds, the execution time for that sequence is the calculated time.

TABLE 4-22. MICRO-INSTRUCTION CLASSIFICATION

Micro Instruction	Micro-Instruction Type †																			
	A	B	C	C	B	C	C	D	A	B	C	C	C	D	C	D	E	E	F	G
Read/Write Macro Memory	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Read/Write Micro-Memory Operand	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Shift or Scale	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
Ones Complement Arithmetic	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	X
Add or Subtract or (A'+B') (ZL, COL, NU, Z*L) on 32-Bit Processor Only	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	X
TMA, TMAK, GITMAK, (COL, ZL, NU, Z*L)	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	X
A', B', NU, ZL, COL, Z*L	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	X

† A = 168 ns execution time
B = 224 ns execution time
C = 280 ns execution time
D = 336 ns execution time
E = 280 + 56s (where s = number of shifts) ns execution time
F = 504 ns execution time
G = 440 ns execution time (typical)
1 = True
0 = False
X = Don't care

For example, the execution time for a micro-program sequence would be calculated as in table 4-23.

MICRO-MEMORY OPERAND REFERENCES

The MP can transfer information between micro memory and the registers of the processor if it has read/write micro memory.

Micro memory is addressed as one to 16 pages of 256 words each, where each word contains 64 bits, divided into upper and lower 32-bit words. A 32-bit processor can reference 32-bit micro-memory words by specifying page, address, and upper or lower micro instruction. A 16-bit processor can reference only 16 bits at a time, so an additional specification of the upper or lower 16 bits of each 32-bit micro-memory half-word is required to address all bits of micro memory.

TABLE 4-23. SAMPLE MICRO-PROGRAM TIMINGS

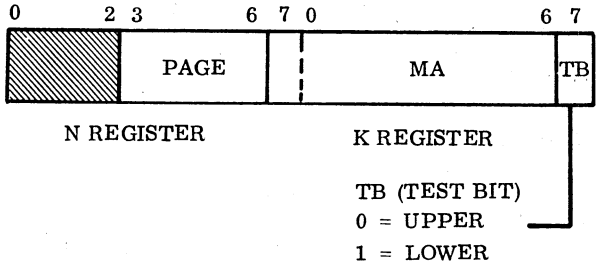
Instruction Type	Time	Sequenced Time (Nanoseconds)
G (Memory reference)	440	888
A	168	
C	280	
G (Memory reference)	440	608
A	168	
G (Memory reference)	625	1129 (440 + 185 = 625)
C (GITMAK)	280	
B	224	
G (Memory reference)	440	1112
A	168	
F	504	
G (Memory reference)		
⋮	⋮	

The micro instruction for micro-memory operand references may be an upper or a lower micro instruction; the next micro instruction executed following the referencing type micro instruction will always be the upper micro instruction of the next sequential location. Micro instructions referencing a micro-memory operand do not have to reside in the same page as the micro-memory operand being referenced when the combined N/K register is used to reference the operands.

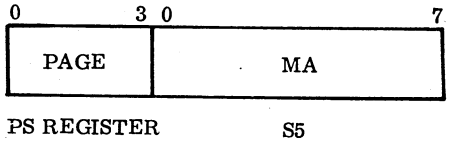
The commands for reading an operand from micro memory are coded in the B' field as MMU and MML; the commands for writing an operand into micro memory are coded in the D' field as MMU and MML.

Two addressing modes are available for operand references via status mode bit 13 (SM113).

If SM113 equals 0, the contents of the combined N/K registers are used to reference operands as indicated. The least significant bit of K (bit 07) determines which 32-bit half-word is referenced via T' code 010.



If SM113 equals 1, an MA transform via selector S5 determines the 64-bit word. The 32-bit half-word to be referenced must be selected by a T-field test.



PROGRAMMING AIDS

This section provides a summary of the micro code for the micro processor. The user will find a synopsis of the codes used to micro program the unit in table 5-1. This is to be used as a reference while programming the equipment.

An assembler for the 16-bit and 32-bit micro programmable processors provides the mnemonic language for the programmer to write a micro program. The assembler translates symbolic source program instructions into object machine instructions and provides a listing of assembly results.

The characteristics of the assembler as written for the CDC® CYBER 170/70/6000 and 1700 Series computers is described in the CCP Support Software 1 MICRO Assembler Reference Manual.

TABLE 5-1. MICRO CODE SUMMARY

ADDRESSING MODES

M	Addressing Mode	Format
0 0	Return	1
0 1	Sequential	1
1 0	Jump	2
1 1	Sequential	3

LOGICAL OPERATIONS

F Code	Mnemonic
0 1 1 0 0	ZERO
0 1 1 1 0	A•B
0 1 1 0 1	A• -B
0 1 1 1 1	A
0 1 0 0 0	-A • B
0 1 0 1 0	B
0 1 0 0 1	EOR
0 1 0 1 1	A+B
0 0 1 0 0	-A • -B
0 0 1 1 0	-EOR
0 0 1 0 1	-B
0 0 1 1 1	$A+^{-B}$
0 0 0 0 0	-A
0 0 0 1 0	-A+B
0 0 0 0 1	$-A+^{-B}$
0 0 0 1 1	ONE

ARITHMETIC OPERATIONS

F Code	Mnemonic
1 0 1 0 0	SUB
1 1 0 0 0	ADD
1 0 1 0 1	SUBT
1 1 0 0 1	ADDT

ARITHMETIC OPERATIONS (Continued)

F Code	Mnemonic
1 0 1 1 0	SUB-, SUB-C
1 1 0 1 0	ADD+
1 0 1 1 1	SUB-T, SUB-TC
1 1 0 1 1	ADD+T

SHIFT INSTRUCTIONS
F = 1 1 1 1 0; (N) = Number of shifts

Bit Code					Mnemonic
7	8	9	11	12	
1	0	0	0	0	AR0E
1	0	0	0	1	ARSE
1	0	0	1	0	AREA
0	1	0	0	0	AL0E
0	1	0	0	1	AL1E
0	1	0	1	0	ALEA
1	0	1	0	0	AQR0E
1	0	1	0	1	AQRSE
1	0	1	1	0	AQREA
0	1	1	0	0	AQL0E
0	1	1	1	0	AQLEA

SCALE INSTRUCTIONS
F = 1 1 1 1 1

Bit Code					Mnemonic
7	8	9	11	12	
0	1	0	0	0	SL0E
0	1	0	0	1	SL1E
0	1	0	1	0	SLEA
0	1	1	0	0	SDL0E
0	1	1	1	0	SDLEA

TABLE 5-1. MICRO CODE SUMMARY (Continued)

A CODE

A Code	Mnemonic
0 0 0	F2
0 0 1	P
0 1 0	I
0 1 1	X
1 0 0	A
1 0 1	F
1 1 0	F1
1 1 1	MEM

A' CODE

A' Code	Mnemonic
0 0 0	SM1
0 0 1	M1
0 1 0	SM2
0 1 1	M2
1 0 0	A*R8
1 0 1	A*
1 1 0	X*
1 1 1	Q*
A' when S = 0 1 1 1 or 1 0 1 0	

B CODE

B Code	28	29	Mnemonic
0 0 0			F2
0 0 1	1	1	ZERO
0 0 1	1	0	N
0 0 1	0	1	K
0 0 1	0	0	N, K
0 1 0			BG
0 1 1			X
1 0 0			Q
1 0 1			F
1 1 0			F1
1 1 1			MEM

B' CODE

B' Code	Mnemonic
0 0 0	OPEN
0 0 1	NRTJ
0 1 0	INRD
0 1 1	INRS
1 0 0	MMU
1 0 1	MML
1 1 0 or 1 1 1	INTA
B' when S = 1 0 0 0	

TABLE 5-1. MICRO CODE SUMMARY (Continued)

D CODE

D Code	Mnemonic
0 0 0	NOP
0 0 1	P
0 1 0	I
0 1 1	Q
1 0 0	F1
1 0 1	A
1 1 0	X
1 1 1	F

D' CODE

D' Code	Mnemonic
0 0 0	IOD
0 0 1	IOA
0 1 0	MMU
0 1 1	MML
1 0 0	MI
1 0 1	SM1
1 1 0	M2
1 1 1	SM2
D' when S = 1 0 0 1 or 1 0 1 0	

D'' CODE

D'' Code	Mnemonic
0 0 0	NOP
0 0 1	A*LHW
0 1 0	X*LHW
0 1 1	Q*LHW
1 0 0	NOP
1 0 1	A*
1 1 0	X*
1 1 1	Q*
D'' when S = 1 0 1 1	

DD'' CODE

DD'' Code	Mnemonic
1 0 1	AA*
1 1 0	XX*
1 1 1	FQ*
DD'' when S = 0 0 0 1	

T CODE

T Code	Mnemonic
0 0 0	*L
0 0 1	U
0 1 0	L
0 1 1	KZU
1 0 0	NZU
1 0 1	INTU
1 1 0	NU
1 1 1	ZL
T when MIR24 = 0	

T' CODE

T' Code	Mnemonic
0 0 0	*L
0 1 0	LQL
0 1 0	K7L
0 1 1	OVFL
1 0 0	BTU
1 0 1	LQ*L
1 1 0	COL
1 1 1	Z*L
T' when MIR24 = 1	

TABLE 5-1. MICRO CODE SUMMARY (Continued)

S CODE

S Code	Mnemonic
0 0 0 0	NOP
0 0 0 1	DD
0 0 1 0	RPT
0 0 1 1	READ
0 1 0 0	WRITE
0 1 0 1	L8EA
0 1 1 0	F2WR
0 1 1 1	AP
1 0 0 0	BP
1 0 0 1	DP
1 0 1 0	APDP
1 0 1 1	DPP
1 1 0 0	GATEI
1 1 0 1	HALT
1 1 1 0	RTJ
1 1 1 1	CLRNP

FORMAT 1
Bit 19 = 0

C' Code	Mnemonic
0 0 x x x x x	
0 1 0 0 0 0 0	WRCH/0
0 1 0 0 0 0 1	WRCH/1
0 1 0 0 0 1 0	WRCH/2
0 1 0 0 0 1 1	WRCH/3
0 1 0 0 1 0 0	RMW
0 1 0 0 1 0 1	WRHW0
0 1 0 0 1 1 1	WRHW1
0 1 0 1 0 0 0	WRPB
0 1 1 x x x x	GATEIXT
1 0 0 0 1 0 1	INCK
1 0 0 1 1 0 1	INCN
1 0 0 1 0 0	DECK

FORMAT 1 (Continued)
Bit 19 = 0

C' Code	Mnemonic
1 0 0 1 1 0 0	DECN
1 0 0 0 0 0 0	CLRK
1 0 0 1 0 0 0	CLRN
1 0 1 x x x x	SETF/j
1 1 0 x x x x	CLRF/j
1 1 1 0 0 0 0 or 1 1 1 0 0 0 1	RQLXN
1 1 1 0 0 1 1	RQR1E
1 1 1 0 0 1 0	RQR0E
1 1 1 0 1 0 0	RL0E
1 1 1 0 1 0 1	RL1E
1 1 1 0 1 1 0	RR0E
1 1 0 1 1 1 1	RR1E

FORMAT 1
Bit 19 = 1

C'' Code	Mnemonic
0 0 0 x x x x	TMA/j
0 0 1 x x x x	TMAK/j
0 1 0 0 x x x	GITMAK/j
0 1 0 1 x x x	GITMAK/XT
0 1 1 x x x x	TK/j
1 0 0 x x x x	TN/j
1 0 1 x x x x	SUB } Require program protect
1 1 0 x x x x	SLB } Require program protect

TABLE 5-1. MICRO CODE SUMMARY (Continued)

FORMAT 2

C Field	Action
x x x x x x x x	Address of the next instruction. If bit 19 = 1, then S = Page (Page Jump)

FORMAT 3

C Field	Action
x x x x x x x x	If bit 19 = 0, transfer the C-field value (bits 24 to 31) to the K register. If bit 19 = 1, transfer the C-field value (bits 24 to 31) to the N register.

A FIELD	In the micro instruction, the A field specifies the source of the operand to be sent to the ALU from selector 1.
A REGISTER	General-purpose register.
A* REGISTER	Register included in systems that contain the hardware double-precision option.
AB	Address buffer register — macro memory address register.
ALU	Arithmetic logic unit. Performs arithmetic and logical operations on two operands sent to it from two selectors.
B FIELD	In the micro instruction, the B field specifies the source of the operand to be sent to the ALU from selector 2.
BG	Bit generator. Allows a word to be sent to the ALU with all zeros except one bit set at any bit position; used for masking or arithmetic operations.
C FIELD	Constant field. May contain constants, micro-memory addresses, or other codes, depending on the micro-instruction format.
D FIELD	Destination field — Specifies the destination for the results of the operation performed by the ALU.
DEADSTART	Optional logic that allows loading of read/write micro memory from an external input device
DMA	Direct memory access.
DSA	1700 direct storage access.
EMULATION	Process combining hardware and firmware design, by which one processor (emulator) executes programs designed for a different processor, even though one-to-one hardware correspondence does not exist.
F FIELD	Function field. Specifies operation to be performed by the ALU or shift or scale of A or A/Q registers.
F REGISTER	General-purpose register.
FILE 1	Register file addressed by the contents of the K register.
FILE 2	Register file addressed by the contents of the N register.
FIRMWARE	General term for the combination of micro instructions used in a micro program to perform a certain operation.
I REGISTER	General-purpose register.
IC	Integrated circuit.
IXT	I register external on the transform.

I/O	Input/output.
K REGISTER	Eight-bit counter that can be cleared, incremented, or decremented under micro-instruction control. Also used to address file 1.
M FIELD	Mode field. Specifies addressing mode for use in obtaining next micro-instruction pair from micro memory.
MA REGISTER	Micro-memory address register. Holds the micro-memory address of the current micro-instruction pair.
MA TRANSFORM	Micro-memory address transform.
MAC	Memory address counter. Holds the address of the next sequential micro-instruction pair.
MACRO MEMORY	Core memory used by the MP for the storage of operands, etc.
MASK REGISTER	Used to control processing of internal and external interrupts.
MICRO INSTRUCTION	32-bit instruction from micro memory that controls all operations throughout the MP system.
MICRO MEMORY	High-speed semiconductor memory which contains micro programs.
MICRO PROGRAM	Set of micro instructions stored in micro memory.
MIR	Micro-instruction register. Holds the micro instruction being executed.
MP	Micro processor. The basic micro programmable processor which can be configured in many forms/applications.
N REGISTER	Eight-bit counter that can be cleared, incremented, or decremented under micro-instruction control. Also used to address file 2.
P REGISTER	General-purpose register used to hold macro-memory address of the software instruction being executed when the MP is configured as an emulator.
PAGE	The page where the next micro-instruction pair is to be found.
PS	Page storage register used with RTJ.
PROGRAM PROTECT	Optional logic that, when enabled, prevents unprotected programs and I/O users from changing the contents of protected areas of macro memory.
Q REGISTER	General-purpose register used in multiply and divide operations.
Q* REGISTER	Register included in systems that contain the hardware double-precision option.
RTJ REGISTER	Return jump register. Holds the micro-memory address to which control returns at the completion of a subroutine.
S FIELD	Special field. Specifies the operation to be performed in parallel with ALU operation.
S1, S2, ETC.	Selector 1, selector 2, etc.

SELECTOR	Multiplexer that allows the selection of one of several sources of data for transfer from one location in the MP organization to another under micro-instruction control.
16-BIT PROCESSOR	A 16-bit processing element with an application-defined micro program. It may have no macro memory or may have up to 65K macro memory, as defined by the application.
SM	Status/mode register. Contains flag bits and status/mode bits. Flag bits are set under micro-instruction control to enable certain internal MP operations. Status/mode bits indicate internal or external conditions (e.g., memory parity error).
T FIELD	Test field. Specifies whether the upper or lower micro instruction of the next micro-instruction pair is to be executed.
32-BIT PROCESSOR	A 32-bit processing element with an application-defined micro program. It may have no macro memory or may have 8K of either 16-bit or 32-bit macro memory.
TRANSFORM MATRIX	Selects bits from various sources in the MP organization and translates them into a micro-memory address in the MA register or transfers them to the K or N register. Used in converting user application to the MP micro code.
X REGISTER	General-purpose processor register.
X* REGISTER	Register included in systems that contain the hardware double-precision option.

INDEX

COMMENT SHEET

MANUAL TITLE CONTROL DATA® Micro-Programmable Computer Family

Micro Processor Hardware Reference Manual

PUBLICATION NO. 88973400 REVISION 01

FROM NAME: _____

BUSINESS
ADDRESS: _____

COMMENTS: This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number.

CUT ALONG LINE

STAPLE

STAPLE

FOLD

FIRST CLASS
PERMIT NO. 333

LA JOLLA, CA.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
PUBLICATIONS AND GRAPHICS DIVISION
4455 EASTGATE MALL
LA JOLLA, CALIFORNIA 92037

CUT ALONG LINE

FOLD

STAPLE

STAPLE