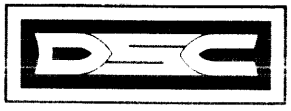
META 4™

series 16 computer system

Digital Scientific



THE DIGITAL SCIENTIFIC

META 4!™

SERIES 16

COMPUTER SYSTEM

Publication No. 7006MO

(Revision E)

PRELIMINARY
SYSTEM MANUAL

June 1970

DIGITAL SCIENTIFIC CORPORATION
11455 Sorrento Valley Road
San Diego, California 92121

PRINTED IN U.S.A.

RECORD OF REVISIONS

Title: Digital Scientific META 4 Series 16 Computer System, Preliminary System Manual

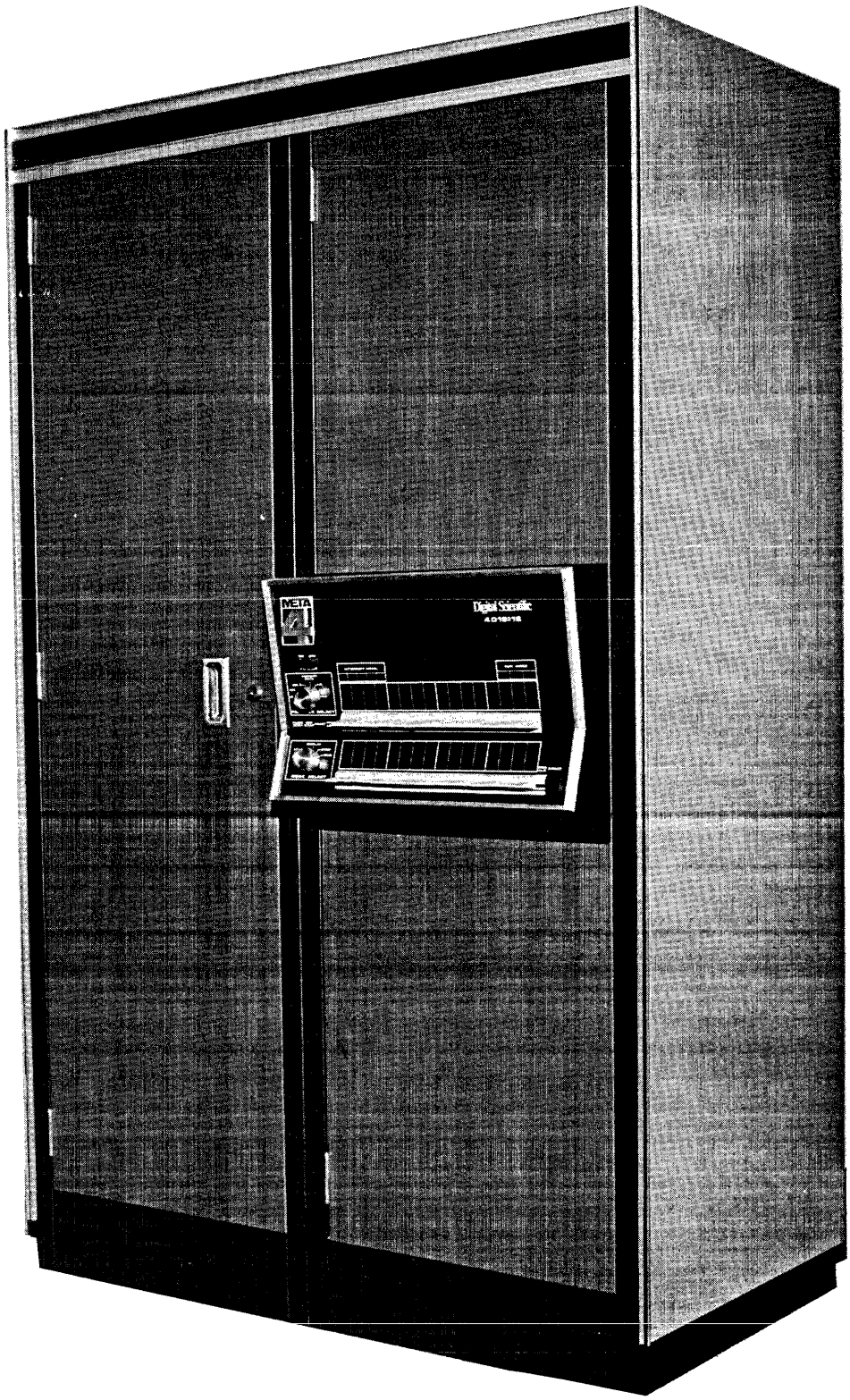
Publication Number: 7006MO (Original)

DATE	ISSUE	
6/6/69	Original	
6/20/69	Revision A	
10/31/69	Revision B	
11/7/69	Revision C	
3/3/70	Revision D	(This revision reflects the adoption of the BNZ, BRZ, and SK mnemonics to replace the BTC, BNC, and AV mnemonics. The table on page 2-4a has also been corrected.)
6/25/70	Revision E	(This revision reflects only minor changes on pages 1-10, 1-13, 2-3, 3-3, and 3-4.)

Address comments to (refer to Comment Sheet at back of manual):

DIGITAL SCIENTIFIC CORPORATION
11455 Sorrento Valley Road
San Diego, California 92121

Phone: 714/453-6050



FRONTISPIECE. DIGITAL SCIENTIFIC META 4 COMPUTER SYSTEM

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

SECTION 4

META 4 SERIES 16 COMPUTER SYSTEM PROGRAMMING TECHNIQUES AND EXAMPLES

LIST OF ILLUSTRATIONS

SECTION 1

DIGITAL SCIENTIFIC META 4 SERIES 16 COMPUTER SYSTEM

1.1 INTRODUCTION

1.1.1 META 4 SYSTEM CONCEPTS [1]

Organization of Digital Scientific's META 4 provides "general-purpose applicability" not only in a main memory instruction set, but also in its underlying, microprogram instruction set and in its flexible hardware organization. Contrary to the traditional computer organization, which bars the system designer from using registers and data paths other than as defined by a core memory instruction set, the system designer determines the number and functional assignment of general-purpose registers for data handling, external input/output, core memory input/output, accumulators or indexing, and the core memory instruction set to use.

The object of META 4 microprogramming capability is to allow the system designer to exercise the hardware as directly as possible without any intervening constraints. The system designer can act both as programmer and logic designer and can manipulate not only algorithms, but architecture.

The META 4 microprogrammed processor can be described as a "computer within a computer," where inner computer sequences emulate the machine language instructions for the outer computer and also execute special sequences not directly related to ordinary outer-machine instructions.

Conventional computer system organization limits the permissible interconnections among functional elements. Since machine language formats and logical organization of conventional computers are closely related, only one machine language instruction set can generally be executed efficiently and the full potential of possible interconnections among the functional elements cannot be realized.

---

[1] Digital Scientific Corporation is indebted to Dr. Robert Rosin for permission to include some of his concepts on emulation in this introduction.

The META 4 microprogrammed computer organization offers the system designer an opportunity to match internal hardware facilities to a particular set of requirements.

Organization of the META 4 system is independent of the machine language instruction set in main memory and permits flexibility in specifying interconnections for functional elements such as arithmetic and Boolean operational units, registers, memory, and input/output devices.

For example, core memory instruction execution in every computer involves a sequence of steps such as:

- Place program location counter in memory address register.
- Fetch instruction from main memory and place in memory data register.
- Increment program location counter to prepare for next instruction.
- Move current instruction to instruction register.
- Decode current instruction and addressing mode.
- Calculate operand address.
- Place operand address in memory address register.
- Fetch operand from main memory and place in memory data register.
- Perform operation.

In a conventional computer, specially wired circuits control each sequence. The main memory instruction set is fixed and can be changed only by rewiring the computer.

In the META 4 Computer, a high-speed control memory replaces the specially wired circuitry. Control memory instructions (the microprogram) specify interconnections of functional elements. The control memory contents can be altered easily by the system designer, thus permitting great flexibility in specifying interconnections and allowing almost any desired main memory instruction set.

The META 4 microprogram emulator differs from a conventional computer program interpreter in several features, as described below.

- Microprograms are stored in a control memory which is distinct from the main memory of the emulated machine and is read-only memory (ROM) for the purpose of optimizing speed.

- Facilities to improve implementation and operation of emulators can be implemented at speeds comparable to those available from hardware. Examples are floating point arithmetic, character code conversion, and control panel operation.

The high-speed computing capability and flexibility are controlled by Digital Scientific Corporation's unique Read-Only Memory (ROM). ROM-controlled instruction cycle times are less than 90 nanoseconds, enabling 10 or more microintructions to be executed during each core memory cycle.

META 4 ROM microprograms (firmware) allow both core memory instructions and special algorithms to be executed much more rapidly than in other "general-purpose" devices. These factors — the computer organization and the high-speed ROM firmware — mean that applications which would normally require special hardware are standard capabilities for a META 4 Computer. For example:

- Instruction Set Emulator for Other Computers (with improved performance): An IBM 1130/1800 can be emulated completely and can be improved upon by adding floating-point instructions and register-to-register instructions.

- Channel Interface or Peripheral Equipment Controller for Other Computers: A disc controller with code and format conversion capabilities can be microprogrammed to operate at high speeds to provide economical standard interfacing to a variety of other computers.

- Communications Line Controller, Buffer, Editor, and Preprocessor: Serial-to-parallel conversion and data editing for multiple nonsynchronous or synchronous lines can be done at high speed to relieve a data processing system of a substantial overhead load.

- Digital Algorithm Processor: Convolution, fast Fourier transformation, correlation, high-level language compilation, or queue optimization algorithms can be executed at high speed, among multiple registers, using core memory for data only, not for program execution.

1.2 PROCESSOR ORGANIZATION

The processing unit (see Figure 1-1) consists of data registers, data processing logic paths, sequence control, input/output facilities, core memory, and integrated scratch-pad memory.

- Data Registers are 16-bit, integrated-circuit registers. Up to 31 directly addressable registers may be installed. During certain operations, data from the Read-Only Memory (ROM) may be used in place of register data.

  In Figure 1-1, registers for addresses 4 through 31 are optional. The requirement for and the choice of a particular type of register depend upon the user's system requirements for accumulator and scratch-pad registers, core memory registers, and input/output registers.

- Data Processing Logic Paths consist primarily of an arithmetic/Boolean unit, which processes data received via the A-bus and the B-bus, followed by a shifter unit, which transmits data to a destination register via the D-bus. The arithmetic unit is a 16-bit, high-speed parallel adder. Carry-in controls, together with overflow and carry-out condition register bits, allow multiple precision operation. The Boolean functions comprise the logical connectives AND, OR, or Exclusive OR. The shifter unit manipulates the result of either an arithmetic or a Boolean operation.

- Sequence Control for the processor is a program stored in high-speed, Read-Only Memory (ROM) and coded in a manner similar to Assembly language instructions for a conventional (hardware-sequenced) computer. Addresses in ROM instructions or in Register 2 (refer to page 1-8) are

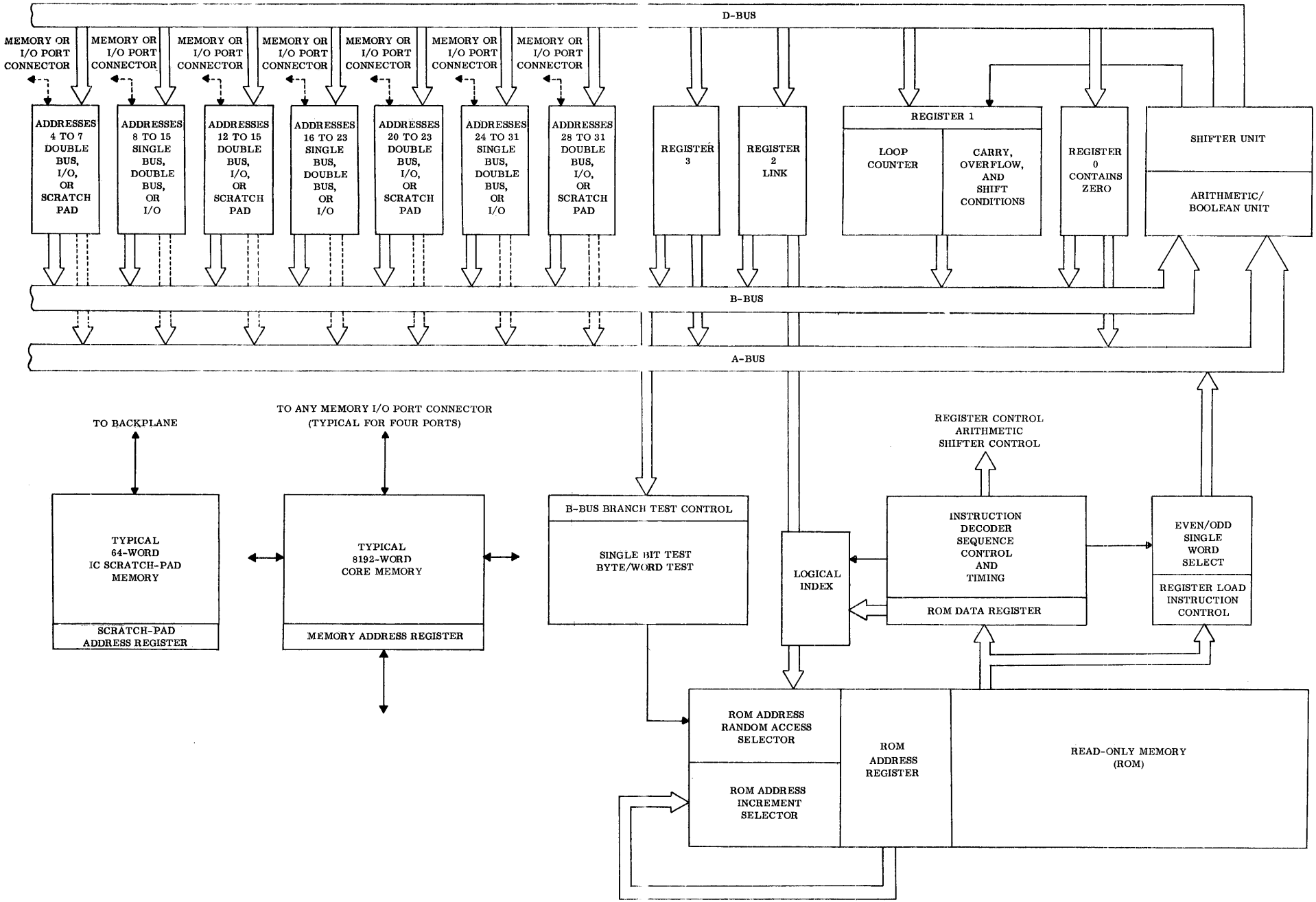


FIGURE 1-1. META 4 COMPUTER SYSTEM, BLOCK DIAGRAM

used by the branch-control unit to shift control between various sequences as the result of testing operations.

Any single bit of any addressable register may be tested for zero or nonzero, 8-bit or 16-bit fields may be tested for zero or nonzero, and a self-decrementing register may be tested for zero concurrently with operations of functional units.

- Input/Output Facilities are implemented at three levels:
  1. Direct cable connections to special types of directly addressable registers. The sequence control program may communicate with the system peripheral equipment through these registers.
  2. Chassis accepting standard controller for various peripheral equipment on a plug-in basis. No field wiring changes are required to add or delete peripheral equipment. Peripheral equipment controllers operate on a party-line I/O bus or directly to memory, as applicable.
  3. Direct access to core independently of the adapter chassis.

- Core Memory is operated by the control program through special registers and controls. Four standard memory ports allow multiple processors or special equipment to share memory. Each 8192-word bank of core memory is an independent unit. The processor can use additional memory registers to overlap accesses to several banks.

- Integrated Circuit Scratch-Pad Memory is operated by the control program through special registers.

1.3 PROCESSOR HARDWARE DESCRIPTIONS

The complete processor, including control memory, mounts in a 19-inch-wide rack housing and requires a 14-inch height for the logic and control memory chassis, a 14-inch height for up to two 8192-word memory banks, and a 14-inch height for the input/output adaptor chassis. Power supplies are normally mounted on the rear rails of a cabinet behind the processor chassis. (See Figure 1-2.)

Processor logic uses high-speed, emitter-coupled, integrated circuits for reliable operation at high speed. Major operation cycle time is 90 nanoseconds including

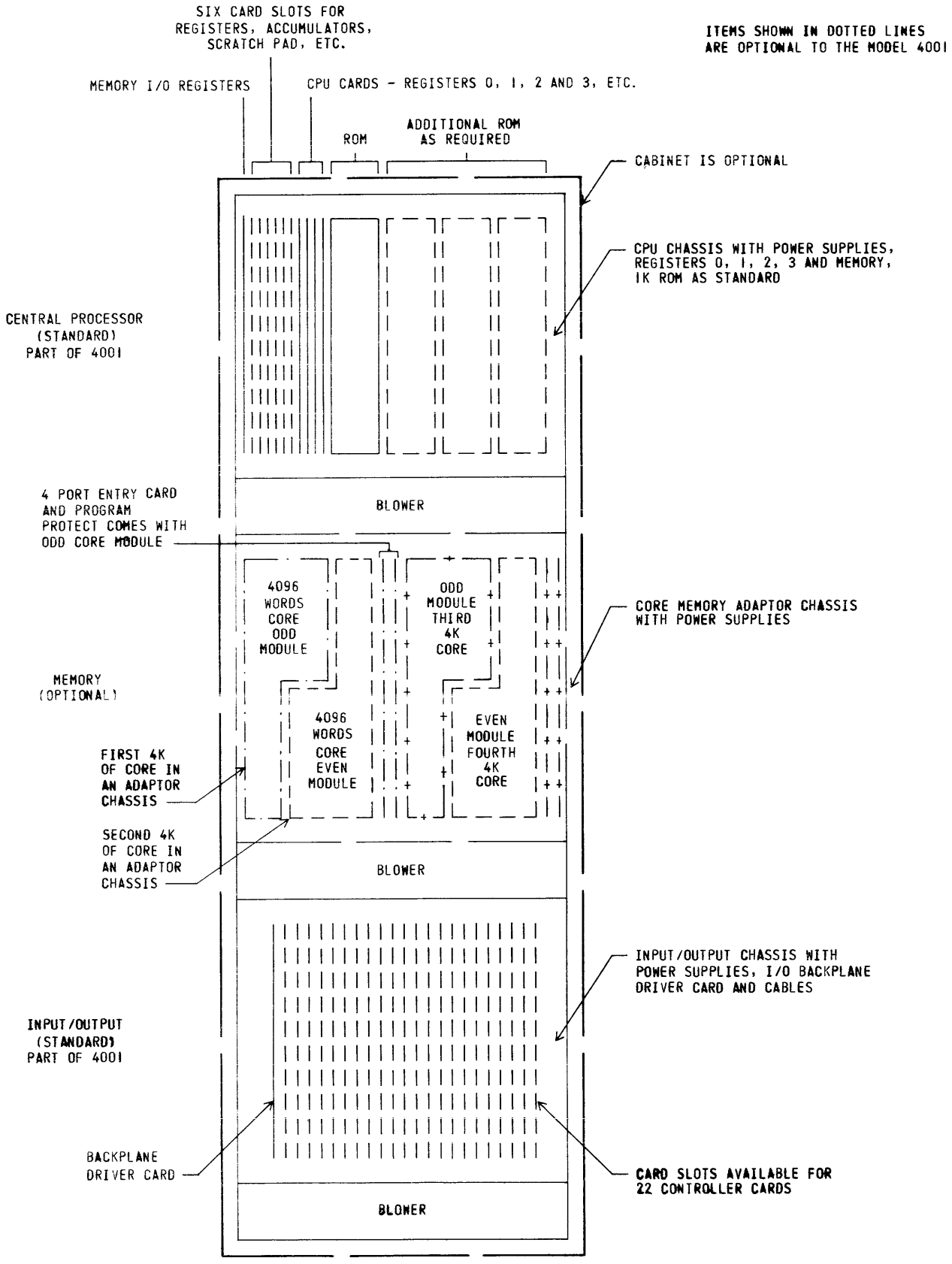


FIGURE 1-2. DIGITAL SCIENTIFIC META 4 SERIES 16 PROCESSOR, HARDWARE ORGANIZATION

Read-Only Memory (ROM) and data source register accesses, arithmetic and/or logical shift processing, and storage in a destination register.

Input/output logic uses a mixture of DTL and TTL integrated circuits.

1.3.1 REGISTERS AND SCRATCH-PAD MEMORY

Optional assemblies for directly addressed registers are available in several versions. Optional assemblies for integrated circuit scratch-pad memory are also available. Register types differ in internal bus connections, external I/O connections, and associated control functions. Register assemblies differ in the number and types of registers.

Type of Assembly	Number and Type of Registers per Assembly	Bus Connections
Double-Bus Accumulators/Index	4 read/write	B, D, A
Single-Bus Accumulators/Index	8 read/write	B, D
Input/Output	4 read, 4 write	B, D
Memory/Input/Output	2 read, 2 write for memory	B, D
	2 read, 2 write for I/O	B, D
Integrated Circuit Memory	64 read/write	B, D (both indirectly)

Registers may be data sources on the A-bus and B-bus and data destinations on the D-bus. Registers without physical connection to the A-bus will function as if the contents were zero for the A-bus only.

Register addresses 0, 1, 2, and 3 are assigned to the basic machine structure (see Figure 1-1). Addresses 4, 5, 6, and 7 are normally assigned to core memory and general I/O functions. All other addresses (8 through 31) are available for general use and are assigned in groups of four registers to seven connectors. Register assemblies containing eight assigned addresses can be installed only in the positions indicated and preclude use of the connector with redundant addresses.

The dedicated register address functions and connections are indicated below:

- Register 0 — Zero register: contains zero for operand use and serves as a dummy destination.

- Register 1 — Condition/Counter register: bits 8 through 15 contain a self-decrementing counter which may be initialized from the D-bus and decremented and tested by instruction control bits. Bits 0, 1, and 2 represent carry-out, overflow, and shift-out conditions and are not controllable from the D-bus. Bits 3 through 7 are fixed at zero. All register bits can be gated to the B-bus for operand use or program testing with the limitation that if Register 1 is specified as both the B-bus source and the D-bus destination of a single instruction the counter contents are indeterminate.

- Register 2 — Link register: serves as an address source for the ROM address selector during specific instructions. The Link register may be set from the D-bus and gated to the B-bus as required and may serve as a single bus accumulator if not required for ROM addressing.

- Register 3 — General-purpose, double-bus accumulator: has no special properties.

1.3.2 DATA PROCESSING LOGIC

1.3.2.1 Arithmetic Unit

The high-speed, 16-bit parallel adder operates in two's complement mode with carry input under program control. Carry-out and overflow automatically force the appropriate condition register bits. The carry-out and the overflow conditions may be tested in the Condition/Counter register. Carry input during an

instruction may be either inhibited, selected to be the previous carry output, or forced unconditionally. The ability to select a previous carry-out as a carry input simplifies multiple precision operation. The ability to force a carry input facilitates two's complement subtraction operations using logical complementing of operands rather than arithmetic complementing. If one's complement arithmetic operation is required, the processor program may use two-step additions in which the second step provides the end-around-carry characteristics of one's complement operations.

Two special addition operations expedite multiply and divide operations:

- Multiply step is addition which is completed only if the shift condition was previously true.
- Divide step is a trial addition where a negative sum inhibits changing the destination register.

1.3.2.2 Boolean Unit

The Boolean unit provides the logical connectives of AND, OR, or Exclusive OR of the A-bus and B-bus sources. Since Register 0 (containing zero) may be used as one of the operands, the Boolean unit may be used to zero registers using the AND function and to copy data using the OR function. An Exclusive OR using a data field from the ROM with all 16 bits true is used to complement data.

1.3.2.3 Shift Unit

The Shift unit provides bit manipulations on the output of the Arithmetic/Boolean unit. Both shift carry-out and shift carry-in for shift operations may be inhibited or selected independently. The shift carry-out status may be tested in the Condition/Counter register (Register 1) and represents the data spill from the most recent shift operation having carry-out enabled.

Shift operations comprise:

- One-place left or right shift
- Eight-place left or right shift/rotate
- Sign extend (copy bit 8 into bits 0 through 7)
- Average (one-place end-off right shift with arithmetic carry entering at left)
- No shift.

1.3.3 SEQUENCE CONTROLS AND READ-ONLY MEMORY (ROM)

The ROM is organized into 16-bit words. Addresses of instruction words must be even. Logically indexed references to data words may use either even or odd addresses. Up to 4096 single words may be installed in multiples of 1024 words. Each reference to the ROM calls up a double word so that access time is identical for single words and double words. Contents of the ROM can be readily modified or replaced in the field by either Digital Scientific Corporation or user personnel.

ROM instructions are executed in sequence unless a Branch causes transfer to another sequence. Branches occur in one of three ways: if specified during an RR format instruction, the next instruction is unconditionally taken from the address in the Link register (Register 2); if the counter section of the Condition/Counter register (Register 1) does not decrement to zero during an RR format instruction when tested, the next instruction is taken from the address in the Link register; if a Branch instruction to test various data or machine conditions is successful, the next instruction is taken from the data field of the instruction and logical indexing by the Link register is selectable.

A 4-bit field in the branch instruction "points" at any single bit of any addressable register. Branching may be selected for the true or false state of the specified bit, allowing tests for data sign, arithmetic carry/overflow, shift carry,

or any other single bit condition. Branching on zero or nonzero half words or single words is selected by a modified branch instruction.

The system is initialized by an externally applied signal which clears the I/O register controls and the ROM address register. Execution of the instruction at ROM address 000_{16} (normally a branch) can lead to a firmware routine that initializes other parts of the system such as internal working registers.

1.3.4 INPUT/OUTPUT TRANSMISSION AND CONTROL

Input/output transmission uses register reference instructions to gate 16 bits of input data directly to the B-bus and to store 16 bits of output data in the latch registers. Voltage-level shifter circuits between emitter-coupled logic and 0 to +5-volt levels are provided and an external signal may be used to enable output from the latch registers.

For input and output operations with the special I/O registers only:

- A program sequence pause occurs if the Pause control instruction bit is used. The input/output operation is delayed until a signal from the external equipment is received.

- A control signal is transmitted to the external equipment upon completion of the instruction if the input/output control instruction bit is used.

The Pause and I/O control bits are ignored if the referenced register is not an I/O-type register. Interlocked transmissions of data are readily made using control signals. Noninterlocked transmissions of data are made by not using one or both control bits.

Input/output pairs share a register address, but are not internally interconnected. Data in an output register cannot be gated back to the B-bus unless the input/output pair are strapped externally. Each pair shares a single ex-

terminal connector and any pair of one processor may be connected to any pair of the same or another processor by using a standard cable.

1.3.5 CORE MEMORY READ/WRITE TRANSMISSION AND CONTROL

Core memory read/write transmission uses register reference instructions in a manner similar to ordinary input/output register reference instructions. The two control instruction bits are interpreted as Read and Write rather than Pause and I/O control. The program sequence Pause function is implied as active when addressing any memory register.

The core memory is a coincident current system with a 900-nanosecond full cycle (read/write). Each completely independent bank of 4096 or 8192 words has four independent access ports. Port priority may be assigned at the discretion of the user and may differ between banks. Memory is protected against power failure.

One 16-bit output register is assigned as the core memory address register 4 and a second 16-bit output register is assigned as the core memory data output register 5. One of the corresponding 16-bit input register addresses is assigned to the core memory data input gate and the second corresponding 16-bit input register address is assigned to the input path for memory parity, protect status condition, and memory control signals. Input/output pairs are not externally connected except for memory data. Only one standard cable is required to connect the core memory with the four register paths. The memory data lines are bidirectional and are shared for input and output.

Standard memory feature:

- One 16-bit odd parity and one protect bit for each 18-bit storage word with automatic abort of Write instructions when the memory cell is protected and the Write control does not indicate a protected write status. Error conditions are transmitted to the processor.

1.4 PERIPHERAL EQUIPMENT

Peripheral equipment is operated either by dedicated registers or by a multiplexed signal bus using one pair of standard I/O registers. One output register is used for addressing and a second output register is used for control and data output. One of the corresponding input register addresses is used for data input and the second corresponding input register address is used for miscellaneous status and data bit inputs. Two standard cables are required to connect the two register paths with the chassis of the peripheral devices.

Standard peripheral devices are listed below:

- Keyboard/Printer
- IBM 1130 Computer Package, including SAC channel, control panel, and ROM containing instruction set firmware
- IBM 1800 Computer Data Channel
- 300-character-per-second Paper Tape Reader
- 50-character-per-second Paper Tape Punch
- 300-card-per-minute Hollerith Card Reader
- 200-card-per-minute Hollerith Card Punch and Reader combination
- Moveable Head Disc with Removeable Single Disc Pack
- 300-line-per-minute Line Printer
- Magnetic Tape Transports (1 x 2 controller, 7- or 9-track)
- Digital Input/Output Interface
- Analog-to-Digital (A/D) and Digital-to-Analog (D/A) Converters
- 300-step-per-second Incremental Plotter

- Teletype Line Adapters
- High-Speed Communications Line Adapters
- Computer-to-Computer Channel Interface for other manufacturers' computers
- Computer Channel Simulator for other manufacturers' computers
- Real-Time Clock/Time-of-Day Clock
- Stall Alarm/Timer

SECTION 2

READ-ONLY MEMORY (ROM) INSTRUCTIONS AND INSTRUCTION MODIFIERS

2.1 GENERAL DESCRIPTION

ROM instructions select specific operations of the arithmetic/Boolean, branch, and register load functions of the computer. Instructions are grouped into four categories, as shown in Figure 2-1.

- BR is the Branch format.
- RR is the Register-Register format.
- RI is the Register-Immediate format.
- RL is the Register Load format.

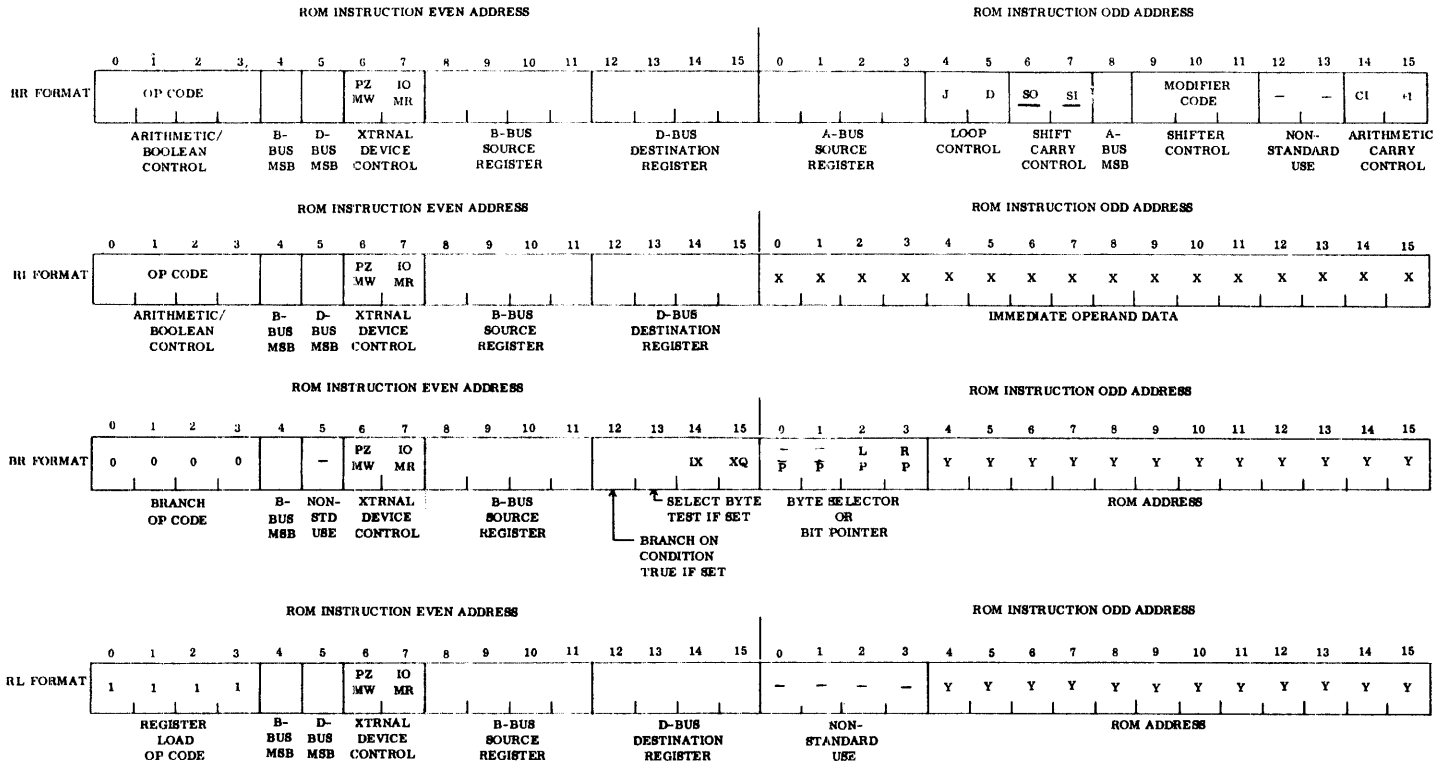
Instruction execution times are each one machine cycle (90 nanoseconds) except for the RL format instruction, which requires two machine cycles (180 nanoseconds).

ROM instruction modifiers select operations of the computer in addition to those selected by the basic instructions. Multiple modifiers may be specified and will operate within the basic instruction cycle times.

2.2

Modifiers are grouped into five categories, as follows:

- Shifter Control
- Arithmetic Control
- Branch Control
- Input/Output and Memory Control
- Instruction Loop Repeat Control



ARITHMETIC/BOOLEAN CONTROL OP CODES

MNEMONIC	HEXADECIMAL	FORMAT
BRZ BNZ	0	BR
AND	1	RR
OR	2	RR
XOR	3	RR
ADD	4	RR
MULT	5	RR
DIV	6	RR
ANDI	9	RI
ORI	A	RI
XORI	B	RI
ADDI	C	RI
LOAD	F	RL

SHIFTER CONTROL MODIFIER CODES

MNEMONIC	HEXADECIMAL
(NONE)	0
RI	1
LI	2
SK	3
SE	4
R8	5
L8	6
EX	7

MISCELLANEOUS MODIFIERS

SYMBOL	SYMBOL
CI	L
D	PZ
IO	R
IX	SI
J	SO
MR	XQ
MW	+1
W	

ARITHMETIC/BOOLEAN CONTROL ASSEMBLER PSEUDO OP CODES

MNEMONIC	HEXADECIMAL	FORMAT
COPY	2	RR
LDI	A	RR
NOP	B	RR
JMP	0	BR
SUBI	C	RI

ASSEMBLER CONTROL PSEUDO OP CODES

MNEMONIC
ORG
HEX
EQU
EQUR
END

MISCELLANEOUS SYMBLOS

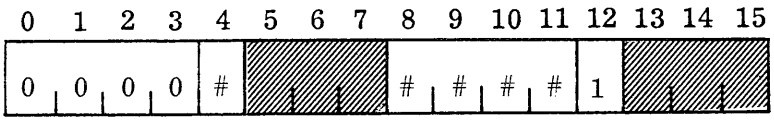
SYMBOLS
P
X
Y

ASSEMBLER SYMBOLS

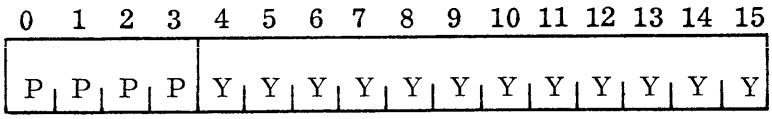
SYMBOL
* (asterisk)
. (period)
$ (dollar)
, (comma)
; (semi-colon)

FIGURE 2-1. META 4 SERIES 16 COMPUTER CONTROL INSTRUCTIONS

BNZ BRANCH IF NONZERO CONDITION **TIMING: 1 CYCLE(S)**



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The condition specified by modifiers or a register bit position is tested. If the test result is nonzero, the next instruction is taken from ROM location Y. If the test result is zero, the next sequential instruction is executed. The least significant bit of Y is ignored and interpreted as zero so that Y will be even. Registers and machine conditions are not changed by a Branch instruction.

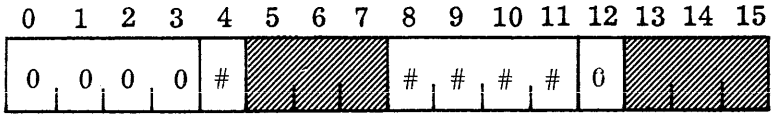
The 5-bit, B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

VALID MODIFIERS:

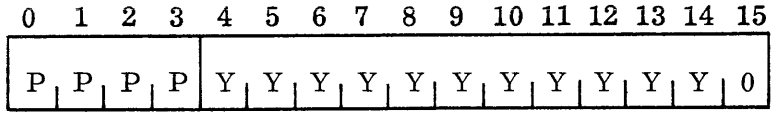
L R W IX XQ PZ IO (Refer also to page 2-4a.)

NOTE: See page 2-16, paragraph 2.3, for a discussion of instruction modifiers. Also, see page 3-4, paragraph 3.2.5, for a discussion of modifiers and their mnemonics as used in writing code for the firmware assembler.

BRZ BRANCH IF ZERO CONDITION **TIMING: 1 CYCLE(S)**



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The condition specified by modifiers or a register bit position is tested. If the test result is zero, the next instruction is taken from ROM location Y. If the test result is nonzero, the next sequential instruction is executed. The least significant bit of Y is ignored and interpreted as zero so that Y will be even. Registers and machine conditions are not changed by a Branch instruction.

The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

VALID MODIFIERS:

L R W IX XQ PZ IO (Refer also to page 2-4a.)

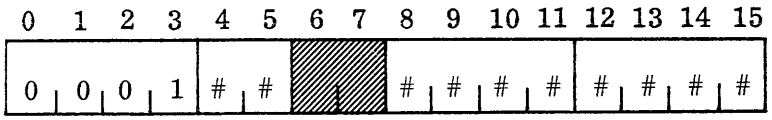
The use of mnemonics BNZ and BRZ is shown in the following table:

	Bits 0 - 7 zero Bits 8 - 15 zero	Bits 0 - 7 zero Bits 8 - 15 nonzero	Bits 0 - 7 nonzero Bits 8 - 15 zero	Bits 0 - 7 nonzero Bits 8 - 15 nonzero
BRZ W	•			
BRZ R	•		•	
BRZ L	•	•		
BRZ R, L	•	•	•	
BNZ R, L				•
BNZ R		•		•
BNZ L			•	•
BNZ W		•	•	•

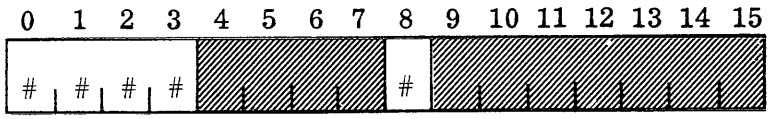
• indicates conditions for successful branch

NOTE: Branch testing of an I/O register input must not be attempted unless the I/O system is stabilized at the time. Stabilization is assured by input/output system data via timing interlocks. Stabilization is not assured for nonsynchronized inputs such as those used for interrupts. The effect of testing a nonstabilized input may be a ROM program branch to an address which is neither the next sequential address nor the expected branch address.

AND — AND B-SOURCE REGISTER WITH A-SOURCE REGISTER — TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified registers are placed on the B-bus and the A-bus, respectively. The AND'ed result is transmitted to the shifter unit for further processing and ultimate storage in the D-bus destination register. Only the D-bus destination register is changed by the AND instruction.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address

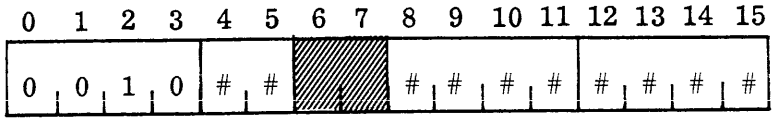
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

The A-bus source register address occupies bit positions 8, 0, 1, 2, 3 of the odd ROM address.

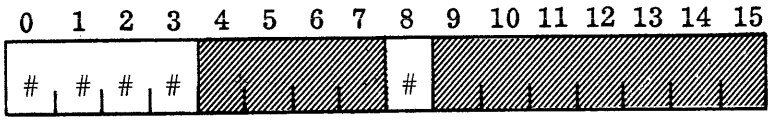
VALID MODIFIERS:

R1 L1 SK SE R8 L8 EX SI SO J D PZ IO MR MW

OR INCLUSIVE OR B-SOURCE REGISTER WITH A-SOURCE REGISTER TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified registers are placed on the B-bus and the A-bus, respectively. The Inclusive OR'ed result is transmitted to the shifter unit for further processing and ultimate storage in the D-bus destination register.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

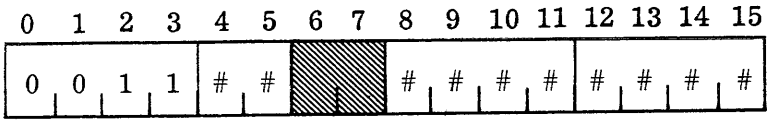
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

The A-bus source register address occupies bit positions 8, 0, 1, 2, 3 of the odd ROM address.

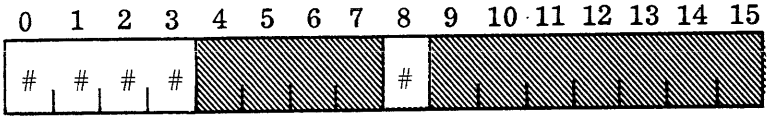
VALID MODIFIERS:

R1 L1 SK SE R8 L8 EX SI SO J D PZ IO MR MW

XOR — EXCLUSIVE OR B-SOURCE REGISTER WITH A-SOURCE REGISTER — **TIMING: 1 CYCLE(S)**



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified registers are placed on the B-bus and the A-bus, respectively. The Exclusive OR'ed result is transmitted to the shifter unit for further processing and ultimate storage in the D-bus destination register.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

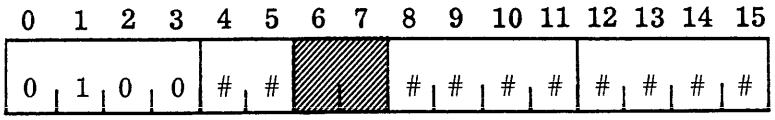
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

The A-bus source register address occupies bit positions 8, 0, 1, 2, 3 of the odd ROM ADDRESS.

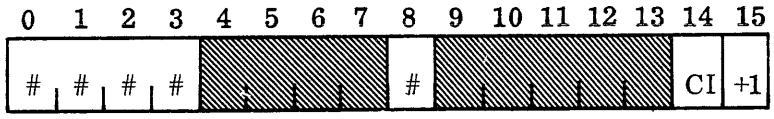
VALID MODIFIERS:

R1 L1 SK SE R8 L8 EX SI SO J D PZ IO MR MW

ADD — ADD B-SOURCE REGISTER TO A-SOURCE REGISTER — TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified registers are placed on the B-bus and A-bus, respectively. The sum is transmitted to the shifter unit for further processing and storage in the D-bus destination register. Addition is carried out in two's complement format. Carry input to the least significant bit is controlled by CI, which enables the previous carry condition as input, and +1, which forces carry input. The carry condition is set to correspond to the carry from bit 0 and the overflow condition is set to correspond to the Exclusive OR of the carries from bits 1 and 0. The D-bus destination register and the carry and overflow conditions are changed by the ADD instruction.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

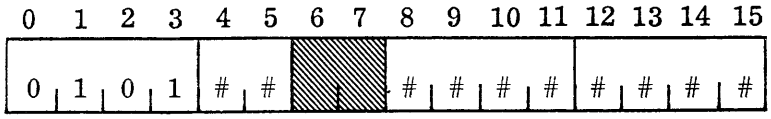
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

The A-bus source register address occupies bit positions 8, 0, 1, 2, 3 of the odd ROM address.

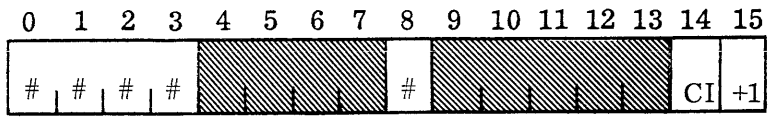
VALID MODIFIERS:

R1 L1 SK SE R8 L8 EX SI SO J D PZ IO MR MW CI +1

MULT — MULTIPLY STEP; CONDITIONAL ADD B-SOURCE REGISTER TO A-SOURCE REGISTER — TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified registers are placed on the B-bus and A-bus, respectively. The sum is transmitted to the shifter unit for further processing and storage in the D-bus destination register. If the shift condition is zero, prior to the MULT instruction execution, the B-bus data source is inhibited so that the A-bus data source passes through the adder unchanged.

Addition is carried out in two's complement format. Carry input to the least significant bit is controlled by CI, which enables the previous carry condition as input, and +1, which forces carry input. The carry condition is set to correspond to the carry from bit 0 and the overflow condition is set to correspond to the Exclusive OR of the carries from bits 1 and 0. The D-bus destination register and the carry and overflow conditions are changed by the MULT instruction.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

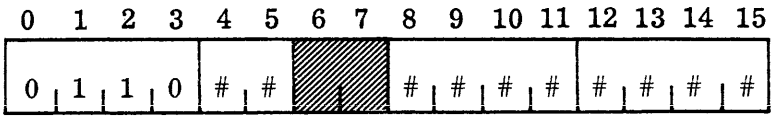
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

The A-bus source register address occupies bit positions 8, 0, 1, 2, 3 of the odd ROM address.

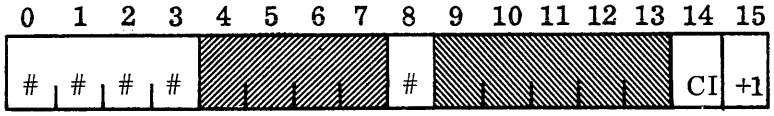
VALID MODIFIERS:

R1 L1 SK SE R8 L8 EX SI SO J D PZ IO MR MW CI +1

DIV DIVIDE STEP; TRIAL ADD B-SOURCE REGISTER TO A-SOURCE REGISTER TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified registers are placed on the B-bus and A-bus, respectively. The sum is transmitted to the shifter unit for further processing and storage in the D-bus destination register.

If the sign of the sum is negative, the destination register is not changed. Addition is carried out in two's complement format. Carry input to the least significant bit is controlled by CI, which enables the previous carry condition as input, and +1, which forces carry input. The carry condition is set to correspond to the carry from bit 0 and the overflow condition is set to correspond to the Exclusive OR of the carries from bits 1 and 0. The D-bus destination register and the carry and overflow conditions are changed by the DIV instruction.

Normal use of the DIV instruction requires that the SO modifier be specified concurrently in order to form the complement of the quotient, bit by bit.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

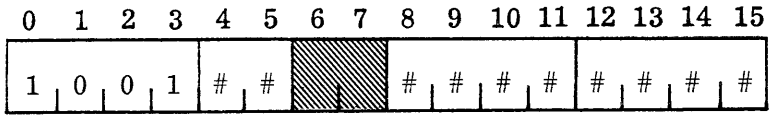
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

The A-bus source register address occupies bit positions 8, 0, 1, 2, 3 of the odd ROM address.

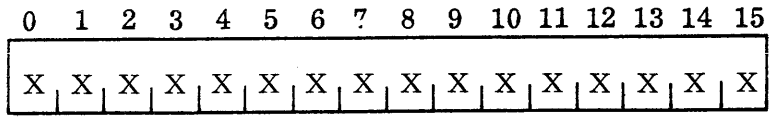
VALID MODIFIERS:

R1 L1 SK SE R8 L8 EX SI SO J D PZ IO MR MW CI +1

ANDI AND B-SOURCE REGISTER WITH IMMEDIATE OPERAND **TIMING: 1 CYCLE(S)**



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified register and the X field of the instruction are placed on the B-bus and A-bus, respectively. The AND'ed result is stored in the D-bus destination register.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

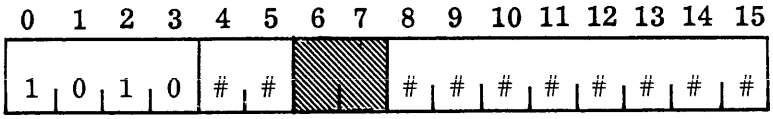
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

VALID MODIFIERS:

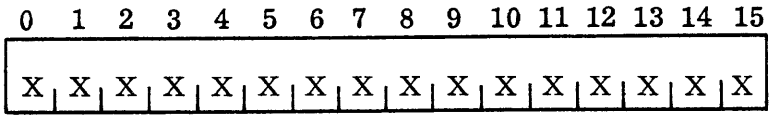
PZ IO MR MW

ORI INCLUSIVE OR B-SOURCE REGISTER WITH IMMEDIATE OPERAND

TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified register and the X field of the instruction are placed on the B-bus and the A-bus, respectively. The OR'ed result is stored in the D-bus destination register.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

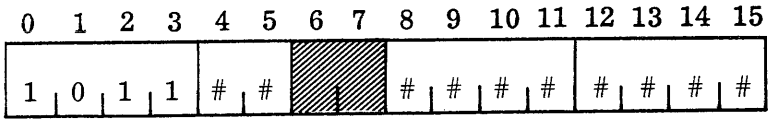
The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

VALID MODIFIERS:

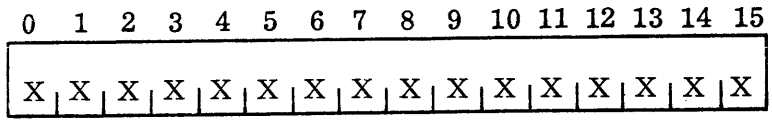
PZ IO MR MW

XORI EXCLUSIVE OR B-SOURCE REGISTER WITH IMMEDIATE OPERAND

TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified register and the X field of the instruction are placed on the B-bus and the A-bus, respectively. The XOR'ed result is stored in the D-bus destination register.

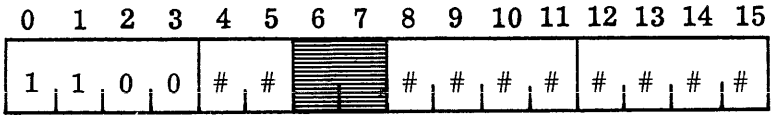
\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

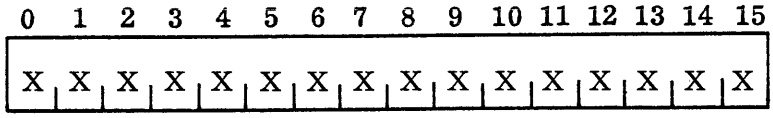
VALID MODIFIERS:

PZ IO MR MW

ADDI — ADD B-SOURCE REGISTER TO IMMEDIATE OPERAND — TIMING: 1 CYCLE(S)



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified register and the X field of the instruction are placed on the B-bus and the A-bus, respectively, and the sum is stored in the D-bus destination. Addition is carried out in two's complement format. The carry condition is set to correspond to the carry from bit 0 and the overflow is set to correspond to the Exclusive OR of the carries from bits 1 and 0. The D-bus register only may be changed by the ADDI instruction.

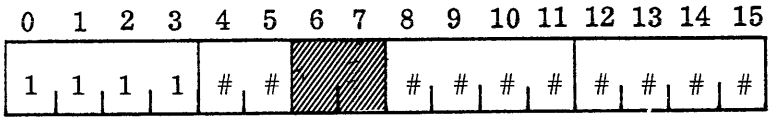
\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

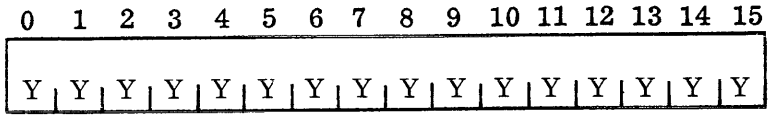
VALID MODIFIERS:

PZ IO MR MW

LOAD LOAD D-DESTINATION REGISTER WITH ROM FIELD **TIMING: 2 CYCLE(S)**



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

The contents of the specified register and the contents of the ROM cell are placed on the B-bus and the A-bus, respectively. The Exclusive OR'ed result is stored in the D-bus destination register.

The ROM cell is specified by the contents of Register 2 logically indexed (Inclusive OR) with the Y field of the instruction.

\# The B-bus source register address occupies bit positions 4, 8, 9, 10, 11 of the even ROM address.

The D-bus destination register address occupies bit positions 5, 12, 13, 14, 15 of the even ROM address.

VALID MODIFIERS:

PZ IO MR MW

2.3 ROM INSTRUCTION MODIFIERS

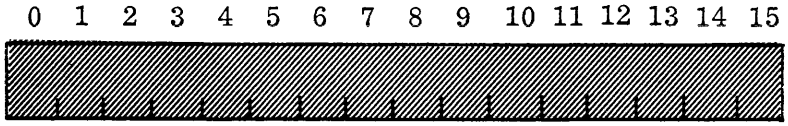
The ROM instruction modifiers are described in detail on the following pages. They are grouped as outlined below.

- Shifter Control Modifiers
- Arithmetic Control Modifiers
- Instruction Loop Repeat Control Modifiers
- Branch Control Modifiers
- Input/Output and Memory Control Modifiers

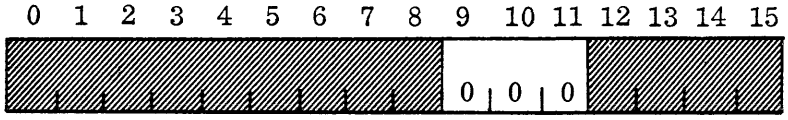
(No Shift Control Modifier)

APPLICABLE INSTRUCTION FORMAT(S): RR

TRANSMIT DATA WITHOUT MODIFICATION



ROM INSTRUCTION, EVEN ADDRESS



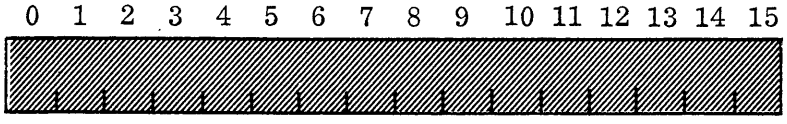
ROM INSTRUCTION, ODD ADDRESS

Output from the Arithmetic/Boolean unit is transmitted without modification.

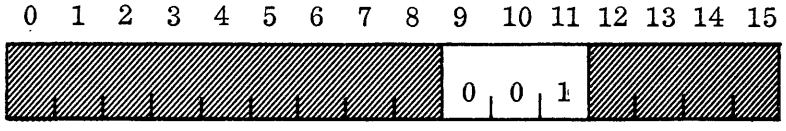
R1

APPLICABLE INSTRUCTION FORMAT(S): RR

SHIFT RIGHT ONE PLACE



ROM INSTRUCTION, EVEN ADDRESS



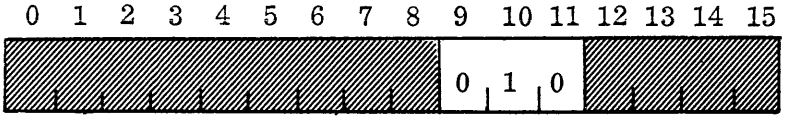
ROM INSTRUCTION, ODD ADDRESS

Output from the Arithmetic/Boolean unit is displaced right one place. Spill from bit 15 may be saved in the Shift Condition register bit by the SO modifier. Entry to bit 0 from the previous shift condition is controlled by the SI modifier.

L1 APPLICABLE INSTRUCTION FORMAT(S): RR

SHIFT LEFT ONE PLACE

ROM INSTRUCTION, EVEN ADDRESS

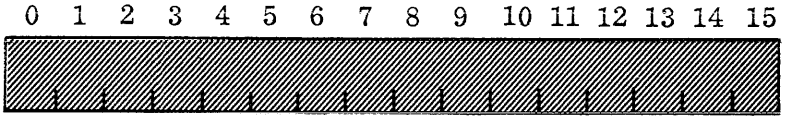


ROM INSTRUCTION, ODD ADDRESS

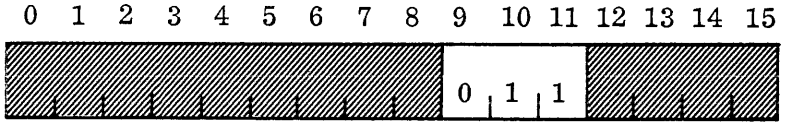
Output from the Arithmetic/Boolean unit is displaced left one place. Spill from bit 0 may be saved in the Shift Condition register bit by the SO modifier. Entry into bit 15 from the previous shift condition is controlled by the SI modifier.

SK APPLICABLE INSTRUCTION FORMAT(S): RR

SCALE



ROM INSTRUCTION, EVEN ADDRESS



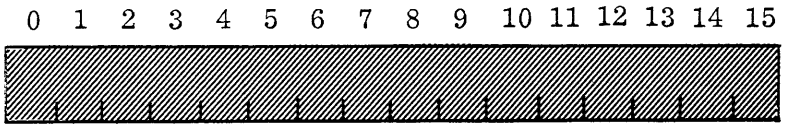
ROM INSTRUCTION, ODD ADDRESS

Output from the Arithmetic/Boolean unit is displaced one place to the right. Spill from bit 15 may be saved by the SO modifier. Entry to bit 0 is made from the current arithmetic carry during an ADD operation or from the Carry Condition register bit during operations other than ADD. If the SI modifier is specified concurrently with SK, entry to bit 15 is the OR between the Shift Condition register bit and the proper carry condition.

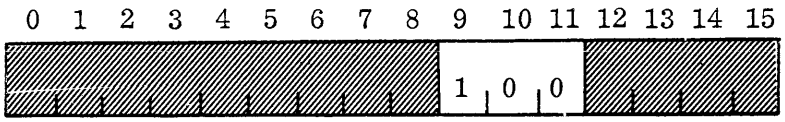
SE

APPLICABLE INSTRUCTION FORMAT(S): RR

SIGN EXTEND



ROM INSTRUCTION, EVEN ADDRESS



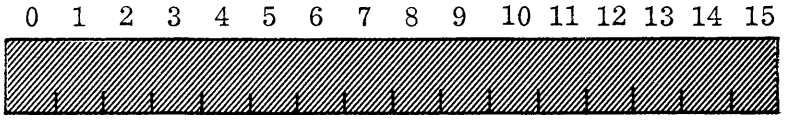
ROM INSTRUCTION, ODD ADDRESS

Output from the Arithmetic/Boolean unit is transmitted with bits 0 through 7 replaced by copies of bit position 8.

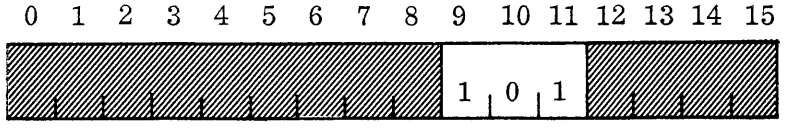
R8

APPLICABLE INSTRUCTION FORMAT(S): RR

SHIFT RIGHT EIGHT PLACES



ROM INSTRUCTION, EVEN ADDRESS

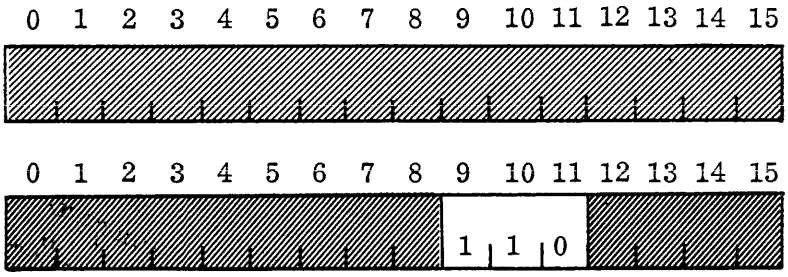


ROM INSTRUCTION, ODD ADDRESS

Output from the Arithmetic/Boolean unit is displaced right eight places. Spill from the right is lost; zeros enter at the left.

L8 APPLICABLE INSTRUCTION FORMAT(S): RR

SHIFT LEFT EIGHT PLACES



Output from the Arithmetic/Boolean unit is displaced left eight places. Spill from the left is lost; zeros enter at the right.

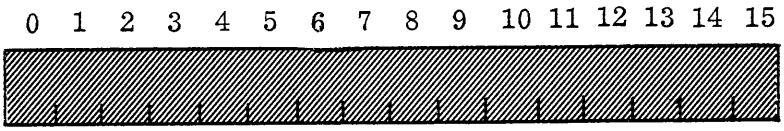
EX APPLICABLE INSTRUCTION FORMAT(S): RR

EXCHANGE BYTES

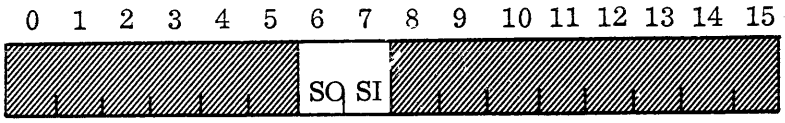
Output from the Arithmetic/Boolean unit is rotated eight places so that bits 0 through 7 and bits 8 through 15 are interchanged.

SO
SPILL SHIFTER TO SHIFT CARRY BIT APPLICABLE INSTRUCTION FORMAT(S): RR

SI
ENTER SHIFT CARRY BIT TO SHIFTER



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

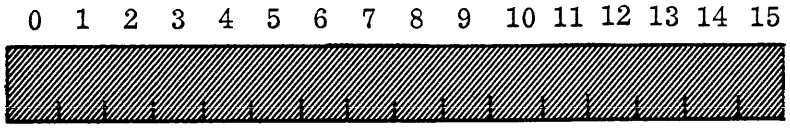
Shifter spill is always from either bit 0 or bit 15 of the operand. SO is effective for any shifter control modifier code. Spill is from bit 0 for no shift, L1, SE, and L8; and from bit 15 for R1, SK. R8, and EX. Refer to DIV instruction description for use of the SO modifier with operations other than a shift.

Shifter input is taken from the Shift Condition register bit when the SI modifier is specified. The SI modifier is enabled only for R1 and L1 modifiers and controls either bit 0 or bit 15 entry, as appropriate. If SK and SI modifiers are specified concurrently, the entry tc bit 0 is the OR between the shift condition and the arithmetic carry.

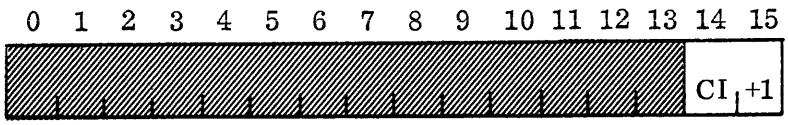
Circular shifts (end around) may be implemented by first executing a single shift operation (right or left, as appropriate) with register zero as the destination and SO specified. The shift carry bit will then be properly set so that subsequent shift operations with both SO and SI specified will be a circular shift.

CI
ENABLE ARITHMETIC CARRY INPUT APPLICABLE INSTRUCTION FORMAT(S): RR

+1
FORCE ARITHMETIC CARRY INPUT



ROM INSTRUCTION, EVEN ADDRESS



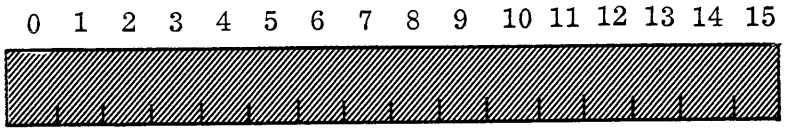
ROM INSTRUCTION, ODD ADDRESS

Carry input to the adder is controlled by CI and +1 modifiers. If neither is specified, the add is without carry input. If CI is specified, the previous carry condition is used as carry input to the least significant stage of the adder. If +1 is specified, a carry input is forced unconditionally.

D
DECREMENT COUNTER

APPLICABLE INSTRUCTION FORMAT(S): RR

J
JUMP ON COUNTER TEST



ROM INSTRUCTION, EVEN ADDRESS

ROM INSTRUCTION, ODD ADDRESS

The low-order 8 bits of Register 1 (the Counter) are decremented and tested using the J and D modifiers. If J is specified without D, a branch to the address specified by the contents of Register 2 (the Link) occurs.

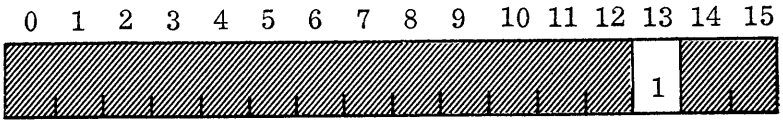
If D is specified without J, the counter is decremented at the conclusion of the instruction.

If J is specified concurrently with D, a branch to the address specified by the contents of Register 2 occurs unless the counter decrements to zero.

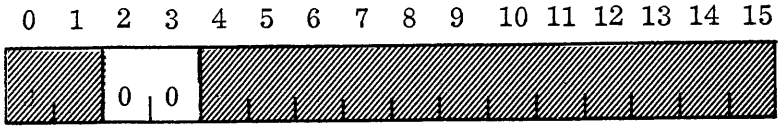
W

APPLICABLE INSTRUCTION FORMAT(S): BR

TEST WORD (RIGHT AND LEFT BYTES)



ROM INSTRUCTION, EVEN ADDRESS

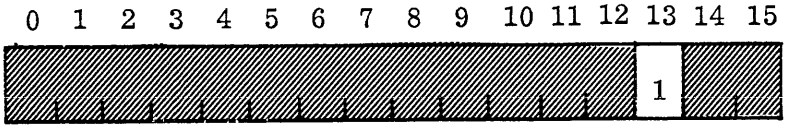


ROM INSTRUCTION, ODD ADDRESS

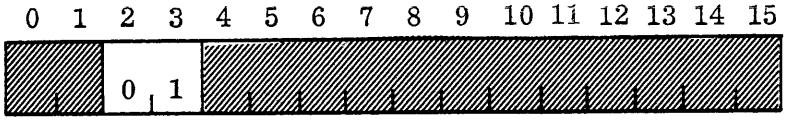
The contents of the register referenced by the B-bus address is tested for zero or nonzero condition.

R APPLICABLE INSTRUCTION FORMAT(S): BR

TEST RIGHT BYTE



ROM INSTRUCTION, EVEN ADDRESS

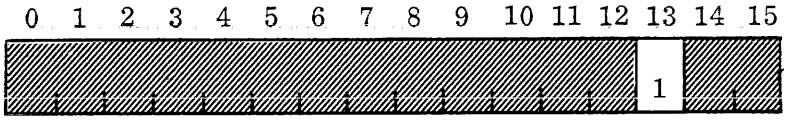


ROM INSTRUCTION, ODD ADDRESS

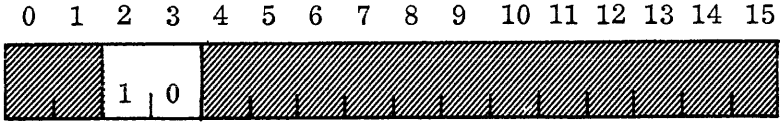
The right byte (8 bits) of the control of the register referenced by the B-bus address is tested for zero or nonzero condition.

L APPLICABLE INSTRUCTION FORMAT(S): BR

TEST LEFT BYTE



ROM INSTRUCTION, EVEN ADDRESS



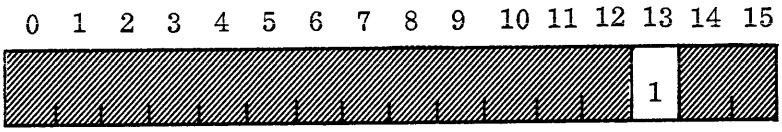
ROM INSTRUCTION, ODD ADDRESS

The left byte (8 bits) of the contents of the register referenced by the B-bus address is tested for zero or nonzero condition.

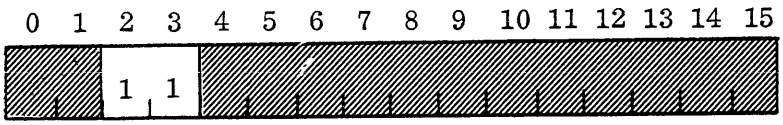
R, L

APPLICABLE INSTRUCTION FORMAT(S): BR

TEST RIGHT OR LEFT BYTE



ROM INSTRUCTION, EVEN ADDRESS



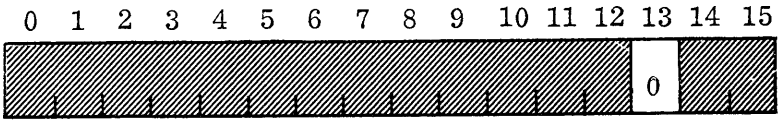
ROM INSTRUCTION, ODD ADDRESS

The right byte (8 bits) and left byte (8 bits) of the contents of the register referenced by the B-bus address are checked independently for zero or nonzero, with the Inclusive OR of the results tested for the zero or nonzero condition.

(No Byte Test Modifiers)

APPLICABLE INSTRUCTION FORMAT(S): BR

TEST SPECIFIED BIT



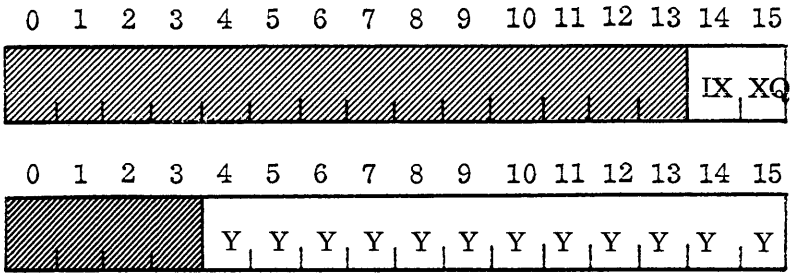
ROM INSTRUCTION, EVEN ADDRESS

ROM INSTRUCTION, ODD ADDRESS

One bit of the contents of the register referenced by the B-bus address and the P field is tested for zero or nonzero. This modifier enables a test and branch capability on any bit of any register and encompasses tests for even/odd, positive/negative, arithmetic carry, arithmetic overflow, and shift carry. The 4-bit P field (pointer) is decoded to define one of 16 bit positions with the tested word.

IX
LOGICAL INDEX APPLICABLE INSTRUCTION FORMAT(S): BR

XQ
EXECUTE ONE INSTRUCTION AFTER BRANCH



Logical indexing, if selected, OR's the contents of the Link register (Register 2) with the Y field to form the effective address.

Execute mode inhibits changing the ROM address register if the Branch instruction test is successful. The effective address is used directly for the execution of one instruction and the control sequence then reverts back to that instruction which would have been executed had the Branch not been successful, unless that one instruction is itself a Branch instruction. Multilevel Branch and Execute instructions may be used with ultimate reversion of control back to that instruction which would have been done with only one level of Branch and Execute. If a Branch without Execute is in the multilevel Branch sequence, then reversion of control will not occur if the Branch without Execute is successful.

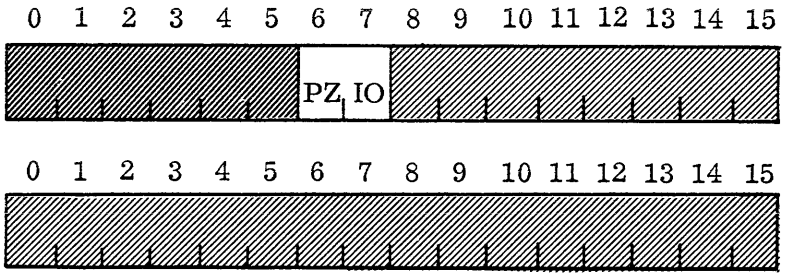
IO

OUTPUT I/O REGISTER CONTROL SIGNALS

APPLICABLE INSTRUCTION FORMAT(S): RR, RI, BR, RL

PZ

PROGRAM PAUSE FOR CONTROL SIGNAL INPUT



Pause occurs prior to any instruction execution; I/O control signals are output at the conclusion of the instruction execution.

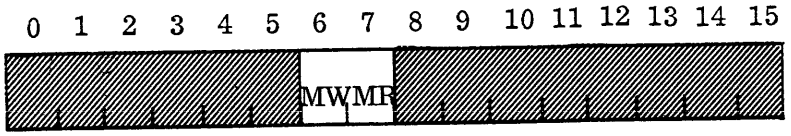
PZ and IO functions are enabled when any I/O register is specified by the B-bus or D-bus addresses. When PZ is specified, the program will pause until a control flip-flop is cleared by an external signal and the clearing pulse has terminated. If more than one I/O register is specified by the bus addresses, the pause condition occurs while any one of the associated control flip-flops is set and the control signal is output on all of the control lines. When IO is specified, the I/O register control line signals are output and control flip-flops are set. The control flip-flops are cleared by the external equipment or by the computer Master Clear.

Note the possible conflicts: MW and MR modifiers use the same control word bit positions as PZ and IO modifiers. Conflicts may occur if memory and I/O registers are specified concurrently.

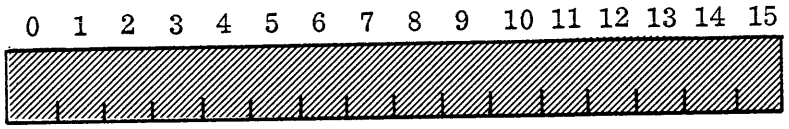
MW
INITIATE MEMORY WRITE

APPLICABLE INSTRUCTION FORMAT(S): RR, RI
BR, RL

MR
INITIATE MEMORY READ



ROM INSTRUCTION, EVEN ADDRESS



ROM INSTRUCTION, ODD ADDRESS

MW and MR modifiers are recognized only when a core memory register is specified by the D-bus address. When either MW or MR is specified, control flip-flops are set. The control flip-flops are cleared by signals from the memory. While the control flip-flops set, the program will pause if the register is specified on either the B-bus or the D-bus and will resume when the memory has completed the appropriate portions of a cycle.

Note the possible conflicts: When MR and MW are specified simultaneously, the core memory will change protect bits to the state specified by bit 15 of the memory data register. Data will be restored without change. PZ and IO modifiers use the same control word bit positions as MW and MR modifiers. Conflicts may occur if memory and I/O registers are addressed concurrently.

A timing restriction must be observed when reading data from the Memory Data register subsequent to initiation of a Read operation while loading the Memory Address register. Memory Read data is valid only if read during the approximately 8 ROM instruction cycles following initiation of the Memory Read cycle. Refer to the Memory Interface description (page 1-12) for Memory Read timing details. Memory Read data is destroyed by a Memory Write command.

SECTION 3

META 4 SERIES 16 COMPUTER FIRMWARE ASSEMBLER CODING

3.1 FUNCTION

The Firmware Assembler converts mnemonics, labels, constants, and modifiers into the various bit patterns of the instructions of the Digital Scientific META 4 Series 16 Computer

3.2 CODING

Instructions, modifiers, and comments are coded on the Firmware Assembler Coding Form (see Figure 3-1).

3.2.1 LOCATION FIELD

Columns 1 - 4 contain an absolute location, a label, or spaces. Absolute locations are represented in (even) hexadecimal notation, left-justified within the field and followed by a $. Labels are left-justified within the field and are terminated by a space. Labels may contain any combination of four alphanumeric characters. An * in column 1 indicates a full line of comments.

3.2.2 OPERATION FIELD

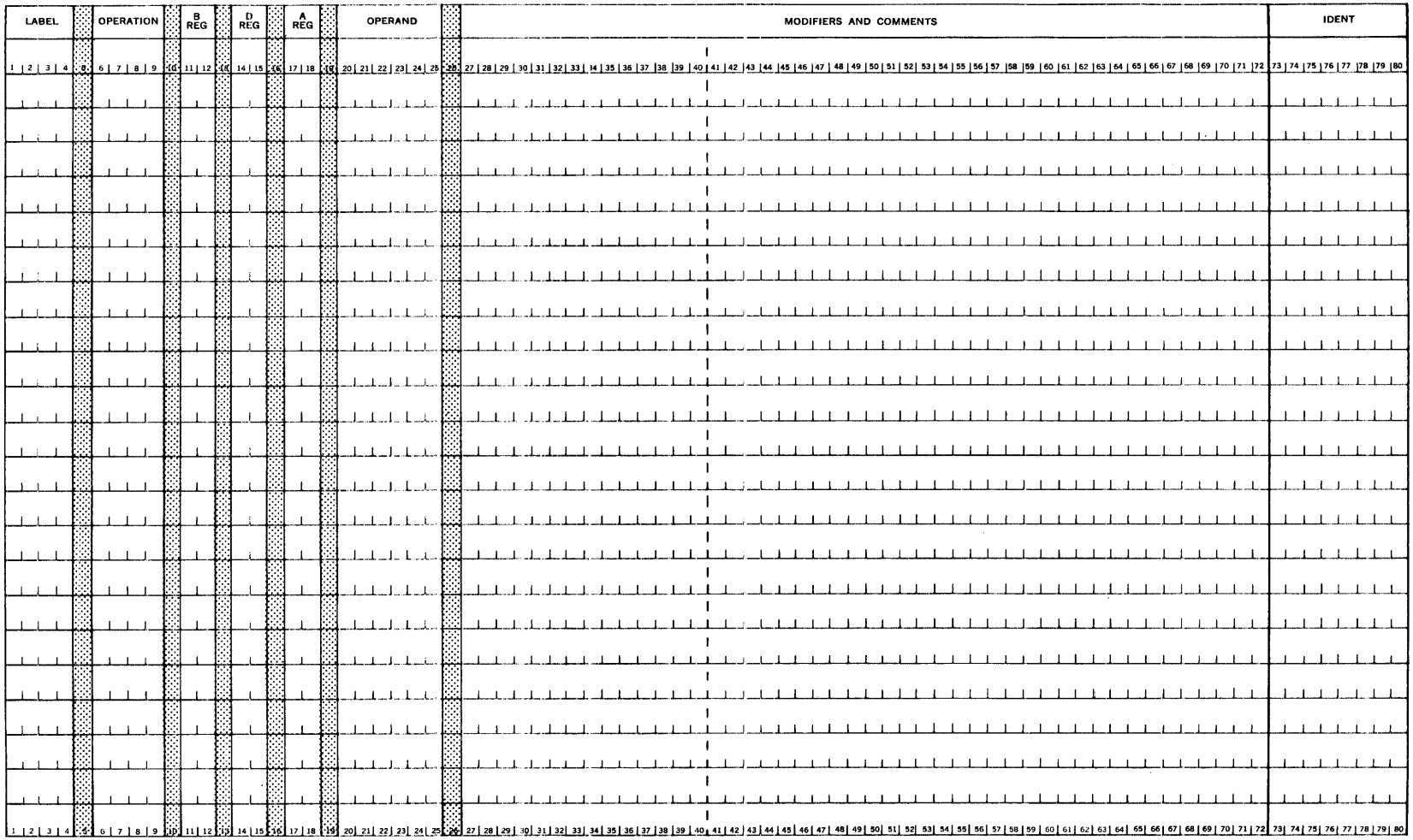
Columns 6 - 9 contain an operation mnemonic or pseudo-op, left-justified within the field. (Operation mnemonics are described in Section 2.) Assembler pseudo-ops are described below.

MNEMONIC	OP_{16}	DESCRIPTION	FORMAT
COPY	2	Copy the B Source to the Destination	RR
LDI	A	Load the Data Field into the Destination	RR
NOP	B	Copy the Data Field to Destination Zero	RR
JMP	0	Unconditional Jump (Test Source Zero = 0)	BR

Digital Scientific META 4 T.M. COMPUTER
FIRMWARE ASSEMBLER CODING FORM

PROGRAM _____ DATE _____

PROGRAMMED BY _____ SHEET _____ OF _____



DSC8A 12/69

FIGURE 3-1. FIRMWARE ASSEMBLER CODING FORM

(Assembler Pseudo-Ops Continued)

MNEMONIC	OP_{16}	DESCRIPTION	FORMAT
SUBI	C	Subtract Immediate: use the Two's Complement of the Data Field in an ADDI Instruction	RI
ORG		Set Origin Address of Routine	
HEX		Permit HEX Data (tables, etc.) to be Assembled with a Routine	
EQU		Equate a Tag with an Absolute Address, Value, or Another Tag	
EQUR		Equate a 2 Character Tag with a Register	
LIST		Allows Control of Output Listing if OFF is Specified in the Operand Field. The Listing will be Terminated until a List Card with ON is Encountered. The On Mode is Assumed if No List Card is Present	
PNCH		Allows Control of Output Binary Cards. Same Rules as List Card	
EJCT		Causes Listing to Start at the Top of a New Page	
END		Terminate Assembly	

3.2.3 SOURCE AND DESTINATION FIELD (BDA)

Columns 11 - 18 contain the source and destination registers to be used in the instruction.

Columns 11 - 12 = B source

Columns 14 - 15 = D destination

Columns 17 - 18 = A source of branch pointer

3.2.4 OPERAND FIELD

Columns 20 - 25 are used as a data field for BR and IM format instructions and must contain a label or a hexadecimal constant. Labels and constants are left-justified within the field and are terminated by a space or a $. A constant must be followed by a $. After a HEX pseudo-op, two 16-bit fields may be specified using a comma (,) as separator. A slash (/) separating two labels indicates that the low order 8 bits of each label are linked to form a 16-bit field.

3.2.5 MODIFIERS

Modifiers are added to instructions in order to control operations in addition to the basic function defined by the operation mnemonic. Modifiers must start in column 27 and can be in any order, spaced by commas. The first space encountered by the Assembler after column 26 indicates that all remaining characters are comments. The instruction modifiers are listed below.

MNEMONIC		VALID FOR
ARITH		
CI	Enable Arithmetic Carry-In	Add, Multiply, Divide
+1	Force Carry-In to 1 and Enable	Add, Multiply, Divide
SHIFT		
SO	Enable Shift Out	All RR Instructions
SI	Enable Shift In	All RR Instructions
L1	Shift Left 1	All RR Instructions
R1	Shift Right 1	All RR Instructions
L8	Shift Left 8	All RR Instructions
R8	Shift Right 8	All RR Instructions
EX	Exchange Bytes	All RR Instructions
SK (SCALE)	Shift Right 1 with Arithmetic Carry Shifted In	All RR Instructions
SE	Sign Extend (copy bit 8 into bits 0 - 7)	All RR Instructions
LOOP		
J	Jump through Link (if D set, jump only if count = 0)	All RR Instructions

MNEMONIC		VALID FOR
BRANCH		
R	Test Right Byte 0	BRZ, BNZ
L	Test Left Byte 0	BRZ, BNZ
W	Test Word 0	BRZ, BNZ
0 through F followed by $	Test Specified Bit Position (0 - 15) 0	BRZ, BNZ
IX	Logical Index With Link	BNZ, BRZ
XQ	Execute One Instruction Out Of Sequence	BNZ, BRZ
I/O		
MR	Initiate Memory Read	All Instructions
MW	Initiate Memory Write	All Instructions
PZ	Pause	All Instructions
IO	Initiate I/O Operations	All Instructions

3.2.6 COMMENT FIELD

Columns 27 - 72 may contain any combination of alphanumeric or special characters.

3.2.7 ASSEMBLER ERROR FLAGS

The following error flags will be typed in the three columns at the left of the assembly listing if any of the conditions are true.

U	Undefined Symbol
R	Undefined Register
O	Invalid Operation
M	Illegal Modifier
D	Duplicate Symbol
F	Format Error
S	Symbol Table Overflow
H	HEX Conversion Error

SECTION 4

META 4 SERIES 16 COMPUTER SYSTEM

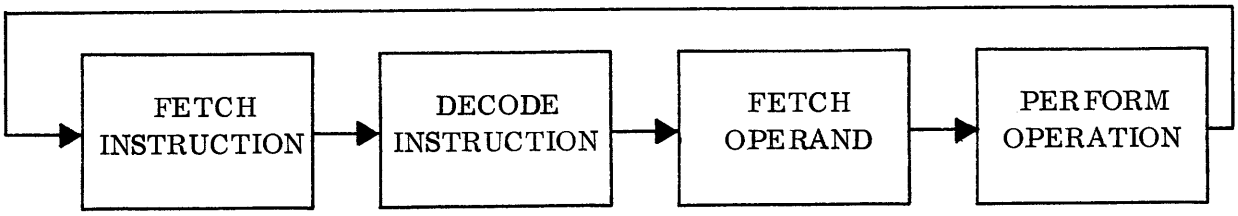
PROGRAMMING TECHNIQUES AND EXAMPLES

4.1 EMULATION

Instruction sets for other computers are readily implemented in the Digital Scientific META 4 Computer's Read-Only Memory (ROM). Firmware performs the operations which are carried out by hardware in computers that lack control memories for sequencing.

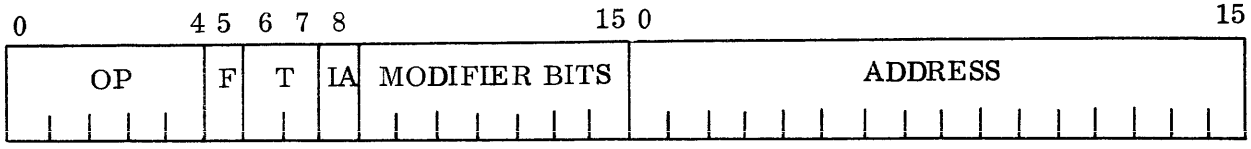
Emulating an IBM 1130 Computer typifies the programming techniques for 16-bit systems. The sequences shown here have been prepared directly from instruction descriptions in the programming manual and from other sources such as timing charts.

The basic functions of any emulation are diagrammed below:



Formats for the instruction to be emulated are as follows.

4.1.1 LONG INSTRUCTION FORMAT (IBM 1130)



OP (OPeration) Code. These five bits specify the operation to be performed.

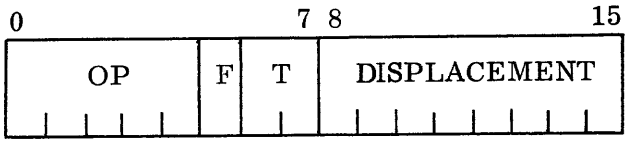
F (Format). The F bit controls the instruction format: 0 = short format; 1 = long format.

T (Tag). These two bits specify the register to be used in effective address generation: 00 indicates the I-register; 01 indicates XR1; 10 indicates XR2; and 11 indicates XR3.

IA (Indirect Address). A zero indicates a direct address (contained in the second word). A 1 bit designates an indirect address.

Modifier Bits. Bit positions 9 through 15 have various uses as modifiers.

4.1.2 SHORT INSTRUCTION FORMAT (IBM 1130)



The first eight bits of the short format are the same as those of the long format. The second eight bits contain the displacement, which is added to data in the register specified by the tag bits to form the effective address. Bit 8 is treated as a sign bit and is extended into bit positions 0 through 7 to obtain a 16-bit number. Negative numbers are expressed as two's complement.

4.1.3 INSTRUCTION DECODING

The instructions decode readily using a table look-up on eight bits of the instruction, which establishes not only the operation to be performed, but also the format and the method of operand-effective address generation.

Since most operations require an operand, a table can be devised to direct the reading of the operand and the subsequent operation to be performed. Using the Subtract instruction as an example, assume that Subtract is located at 044_{16} of the ROM and that the Operand Read routines are located as follows:

SUBROUTINE NAME	ROM ADDRESS
ORSI (Operand Read, Short Format, Relative to I-Register)	F00
ORS1 (Operand Read, Short Format, Index Register 1)	F08
ORS2 (Operand Read, Short Format, Index Register 2)	F10
ORS3 (Operand Read, Short Format, Index Register 3)	F18
OPRL (Operand Read, Long Format)	F40

The table is written in META 4 Computer Assembly language.

The Assembler generates the following table:

```
LOC.  INST.    LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

0E90 44004408  E90SHEX          S ORS1,S ORS1          SUBTRACT
0E92 44104418      HEX          S ORS2,S ORS3
0E94 44404440      HEX          S OPRL,S OPRL
0E96 44404440      HEX          S OPRL,S OPRL
0E98 6C206C28      HEX          SD OWS1,SD OWS1        DBL SUB
0E9A 6C306C38      HEX          SD OWS2,SD OWS3
0E9C 6C486C48      HEX          SD OPWL,SD OPWL
0E9E 6C486C48      HEX          SD OPWL,SD OPWL
```

A META 4 Computer address from the table may not exceed eight bits; therefore, the most significant portion of the address must be "understood" by the program using the logical index facility.

Referencing the numbers adjacent to the boxes of the sample flowcharts as a guide (see Figures 4-1 through 4-3, pages 4-9 through 4-14), the cracking and execution of an instruction proceed as follows. The area between F00 and FFF can be addressed by the least significant eight bits indexed logically by F00. This area is designated P1 and is used for preprocessing such as computing operand addresses, etc.

4.1.3.1 System Conditions

- A Subtract instruction (OP 10010) is located at core storage location 500.
- The accumulator contains 300_{16}.
- The operand in location 520 contains 150_{16}.
- Index Register 2 contains 510_{16}.

4.1.3.2 META 4 Computer Registers

The META 4 Computer registers are listed with their nomenclature on the following page.

META 4 COMPUTER REGISTERS

I	=	SIMULATED INSTRUCTION ADDRESS REGISTER	=	17_{16}
M	=	MEMORY ADDRESS REGISTER	=	4
T	=	INTERRUPT REGISTER	=	6
K	=	PRIORITY MASK REGISTER	=	$1D_{16}$
S	=	SCRATCH REGISTER	=	3
D	=	MEMORY DATA REGISTER	=	5
L	=	LINK REGISTER	=	2
U	=	TEMPORARY ACCUMULATOR	=	16_{16}
A	=	SIMULATED 1130 ACCUMULATOR	=	14_{16}
C	=	CONDITION /COUNTER REGISTER	=	1
X	=	SIMULATED 1130 STATUS REGISTER	=	18_{16}
Q	=	ACCUMULATOR EXTENSION	=	15_{16}
1	=	INDEX REGISTER 1	=	$1A_{16}$
2	=	INDEX REGISTER 2	=	$1B_{16}$
3	=	INDEX REGISTER 3	=	$1C_{16}$

NOTE

In the following listings, Register 1 is incorrectly assigned as the Link register and Register 2 is incorrectly assigned as the Contition/Counter register.

RNI

```
LOC.  INST.    LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

022E 29740000 RNI   COPY I  M            MR,             MOVE I TO MEM ADDRESS
              *                    AND READ NEXT INST
0230 28030000       COPY K  S                            MOVE MASK
0232 10633000       AND  Y  S  S                         AND MASK WITH RAW INT
0234 003C025A       BTC  S        INT    W,              VALID INTERRUPT
0236 CC770001       ADDI I  I     1$                     INCREMENT I
0238 20520050       COPY D  L            R8,             SHIFT OP CODE INTO L
023A F0020E00       LOAD 0  L     E00$                   LOAD THE LINK FROM
              *                    THE TABLE STARTING AT
              *                    E00 AND INDEXED BY
              *                    THE CONTENTS OF L
023C 00020F00       JMP           F00$   IX              JUMP TO P1 AREA
```

ORSI

```
LOC.  INST.    LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

              *ORSI COMPUTES THE OPERAND ADDRESS OF
              *SHORT FORMAT INSTRUCTIONS (REL TO I) AND
              *INITIATTEES THE READ OF THE OPERAND
0F00 20530040 ORSI COPY D  S            SE,             EXTEND SIGN OF DISP
0F02 41347080      ADD  S  M  I         MR,             COMP OP ADDR,READ OP
0F04 20220050      COPY L  L            R8              SHIFT P2 ADDRESS INTO
              *                    LOW 8 OF LINK
0F06 00020000      JMP           0$     IX              JUMP TO P2 AREA
              *                    INDEXED BY CONT OF L
              *
```

ORS1, ORS2, AND ORS3

```
LOC.  INST.    LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

                     *ORS1,ORS2 AND ORS3 ARE IDENTICAL TO ORSI
                     *EXCEPT FOR INDEX REGISTER USED TO
                     *COMPUTE THE EFFECTIVE ADDRESS (EA) OF
                     * HE OPERAND
0F08 20530040 ORS1   COPY D  S              SE,
0F0A 49A43000        ADD  1  M  S           MR,
0F0C 20220050        COPY L  L              R8,
0F0E 00020000        JMP                0$  IX
0F10 20530040 ORS2   COPY D  S              SE,
0F12 49B43000        ADD  2  M  S           MR,
0F14 20220050        COPY L  L              R8,
0F16 00020000        JMP                0$  IX,
0F18 20530040 ORS3   COPY D  S              SE;
0F1A 49C43000        ADD  3  M  S           MR,
0F1C 20220050        COPY L  L              R8,
0F1E 00020000        JMP                0$  IX,
```

ADD

```
LOC.  INST.    LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

0040 44544080 A      ADD  U  A  A                        ADD DATA TO (A)
0042 00000048        JMP             AOV
                     *
```

SUBTRACT

```
LOC.  INST.    LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

0044 B456FFFF S      XORI D  U     FFFF$                 1 S COMP DATA
0046 4C644081        ADD  U  A  A        +1              ADD WITH PLUS 1 = SUB
0048 0010104C AOV    BFC  C     1  *+2                   BR IF NO OVERFLOW
004A AC880001        ORI  X  X     1$                    SET OVFL INDICATOR
004C 00180052        BTC  C     0  *+3                   BR IF CARRY = 1
004E 9C880001        ANDI X  X     1$                    CLEAR CARRY INDICATOR
0050 0000022E        JMP           RNI
0052 AC880002        ORI  X  X     2$                    SET CARRY INDICATOR
0054 0000022E        JMP           RNI
```

MULTIPLY

```
LOC.  INST.    LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

007A A0010010 M     LDI      C      10$                      LOAD 16 INTO COUNTER
007C 08480126       BTC  A        0  MNEG                     BR IF MULTIPLIER NEG
007E 2C450210       COPY A   Q              SO,R1             COPY MULT. INTO Q, RIGHT
                    *                                        SHIFTED 1 PLACE TO COND.
                    *                                        A SUBSEQUENT MULT. STEP
0080 A0030000       LDI      S      0$                       SET SIGN PLUS
0082 0000012C       JMP              MNEG+3
                    *MNEG IS A CONTINUATION OF THE MULTIPLY
                    *ROUTINE IN P2 AREA
0126 BC45FFFF MNEG  XORI A   Q      FFFF$                    COMP MULTIPLIER TO(Q)
0128 4C550211       ADD  Q   Q  0           +1,R1,SO,        +1 FOR 2'S COMP
                    *ALSO SHIFT MULTIPLIER TO CONDITION A
                    *SUBSEQUENT MULTIPLY STEP
012A A0030001       LDI      S      1$                       SET SIGN NEG
012C A4040000       LDI      A      0$
012E 00580134       BTC  D        0  *+3                    BR IF DATA NEG
0130 24560000       COPY D   U                               POS MULTIPLICAND
0132 0000013A       JMP              *+4
0134 B456FFFF       XORI D   U      FFFF$                    COMP MULTIPLIER
0136 CC660001       ADDI U   U      1$
0138 B0330001       XORI S   S      1$                       FLIP SIGN
013A A002013C       LDI      L      MTOP
013C 5C644290 MTOP  MULT U   A  A           SO,R1,           SHIFT AND ADD
                    *CONDITIONED BY PREVIOUS CONTENTS OF THE
                    *SHIFT FLIP-FLOP
013E 2C550F10       COPY Q   Q              SO,R1,SI,D,J,    FORM LEAST
                    *SIGNIFICANT  PORTION OF RESULT,ALSO
                    *SHIFT NEXT BIT OF MULTIPLIER TO SHIFT FF
                    *THEN LOOP UNTIL COMPLETION
0140 0030022E       BFC  S        0  RNI                    EXIT IF POS
0142 BC55FFFF       XORI Q   Q      FFFF$                    COMP RESULT
0144 BC44FFFF       XORI A   A      FFFF$
0146 CC550001       ADDI Q   Q      1$

0148 4C440002       ADD  A   A  0           CI
014A 0000022E       JMP              RNI
```

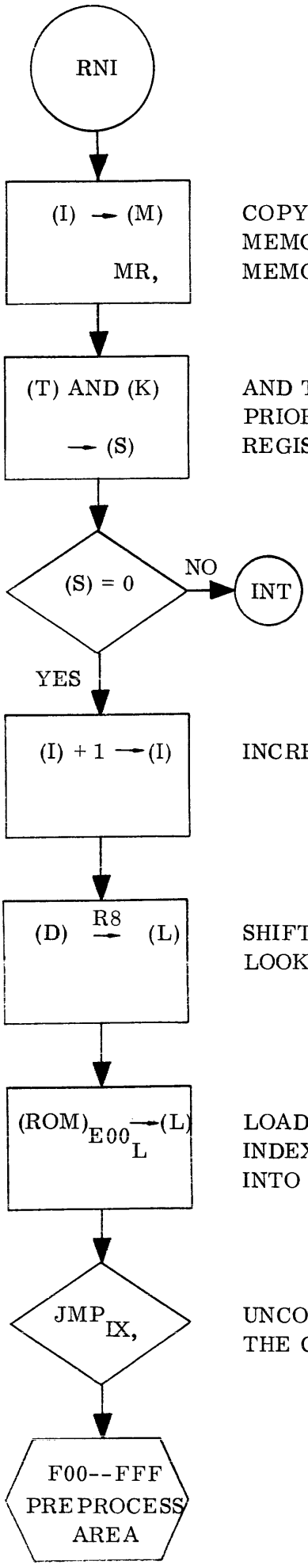
DIVIDE

```
LOC.  INST.   LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS

0084 A0010010 D     LDI      C      10$                    LOAD 16 INTO COUNTER
0086 A0020164       LDI      L      DTOP                   TOP OF LOOP TO LINK
0088 0848014C       BTC  A        0  DNEG                   BR IF DIVIDEND NEG
008A A0030000       LDI      S      0$                     SET SIGN IF QUOT&DIV
008C 24560000       COPY D   U                             SAVE MEMORY DATA
008E 08680162       BNZ  U        0  DN                     BRANCH IF DIVISOR NEG
0090 086C015C       BNZ  U           DNZ     W            BR IF DIVISOR NOT ZERO
0092 AC880001 OVFL  ORI  X    X      1$                     SET OVFL BIT
0094 0000022E       JMP              RNI
                   *DNEG IS A CONTINUATION OF THE DIVIDE
                   *ROUTINE IN P2 AREA
014C 00540092 DNEG  BFC  D           OVFL    W            BR IF DIVISOR ZERO
014E BC44FFFF       XORI A    A      FFFF$                  COMP DIVIDEND
0150 BC55FFFF       XORI Q    Q      FFFF$
0152 CC550001       ADDI Q    Q      1$
0154 4C440002       ADD  A    A   0          CI
0156 24560000       COPY D    U                            SAVE MEMORY DATA
0158 A0030003       LDI      S      3$                     SET SIGN OF QUOT&DIV
015A 08680162       BNZ  U        0  DN                     BR IF DIVISOR NEG
015C BC66FFFF DNZ   XORI U    U      FFFF$                  COMPLEMENT DIVISOR TO
015E CC660001       ADDI U    U      1$                     PERFORM SUBTRACT
0160 00000164       JMP              DTOP
0162 B0330001 DN    XORI S    S      1$                     FLIP SIGN OF QUOTIENT
0164 6C644280 DTOP  DIV  U    A   A          SO,            TRIAL SUBTRACT OR
                   *                  SUBTRACT
0166 2C550320       COPY Q    Q              L1,SO,SI       SHIFT DIVIDEND
                   *BIT IN AND SHIFT DIVIDEND BIT OUT
0168 2C440D20       COPY A    A              L1,SI,D,J,     SHIFT DIVIDEND
                   *AND LOOP TO COMPLETION
016A 6C644280       DIV  U    A   A          SO,            LAST CYCLE REM NOW OK
016C 2C560320       COPY Q    U              L1,SO,SI       COMPLETE QUOT TO L
016E 00102092       BFC  C        2  OVFL                   SHIFT FF =0 = OVFL
0170 0030E182       BFC  S        F  RPOS                   BR IF REMAINDER PLUS
0172 BC45FFFF       XORI A    Q      FFFF$                  COMP REMAINDER
0174 CC550001       ADDI Q    Q      1$                     REMAINDER
0176 08600186 QUOT  BFC  U        0  UNEG                   QUOT IS NEG OK FOR OV
0178 0038F17E       BTC  S        F  QNEG                   BR,SIGN OF QUOT NEG
017A BC64FFFF       XORI U    A      FFFF$                  QUOTIENT TO A
017C 0000022E       JMP              RNI                    EXIT
017E CC640001 QNEG  ADDI U    A      1$                     CONVERT 1'S COMP QUOT
                   *                  TO 2'S COMP QUOTIENT
0180 0000022E       JMP              RNI                    EXIT
0182 2C450000 RPOS  COPY A    Q                             REMAINDER TO (Q)
0184 08680178       BTC  U        0  QUOT+1                 QUOT IS POS
0186 0030F092 UNEG  BFC  S        F  OVFL                   IF FORMED QUOTIENT IS
                   *NEGATIVE AND THE SIGN OF THE QUOTIENT IS
                   *POSITIVE,AN OVERFLOW IS INDICATED
0188 CC698001       SUBI U    O      7FFF$                  IF THE FORMED QUOT
                   *IS NEGATIVE AND THE SIGN OF THE QUOTIENT
                   *IS NEGATIVE AN OVERFLOW IS INDICATED
                   *EXCEPT FOR -2TO THE 15TH
018A 0894017E       BFC  O           QNEG    W,             TEST FOR -2TO 15TH
018C 00000092       JMP              OVFL
```

SAMPLE FLOWCHARTS

ARE INCLUDED ON THE FOLLOWING PAGES



1. COPY INSTRUCTION ADDRESS REGISTER INTO MEMORY ADDRESS REGISTER AND INITIATE MEMORY READ.
2. AND THE INTERRUPT REGISTER (T) WITH THE PRIORITY MASK REGISTER (K). USING TEMPORARY REGISTER, PLACE RESULT IN SCRATCH REGISTER.
3. TEST FOR VALID INTERRUPTS.
4. INCREMENT INSTRUCTION ADDRESS REGISTER.
5. SHIFT 8-BIT OP CODE INTO LINK FOR TABLE LOOK-UP.
6. LOAD THE CONTENTS OF THE TABLE AT $E00_{16}$, INDEXED BY THE CONTENTS OF THE LINK, INTO THE LINK.
7. UNCONDITIONALLY JUMP TO $F00_{16}$ INDEXED BY THE CONTENTS OF THE LINK.

FIGURE 4-1. SAMPLE FLOWCHART, READ NEXT INSTRUCTION (Sheet 1 of 2)

1. Copy the Instruction Address register, I (500), into the Memory Address register, M. Initiate a Read from location 500 in core storage.

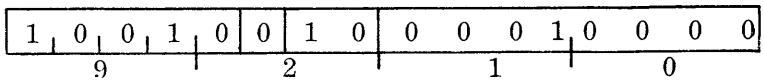
2. During the time required for memory to react, interrupts may be tested without time penalty. In this case, assume that the priority mask in the K-register when logically AND'ed with the raw interrupts in the T-register produce a zero, which is stored in the S-register for subsequent testing.

3. The S-register is tested for zero; if S is zero, any interrupts which may be in T are of a lower priority than the one being serviced and are, therefore, deferred.

4. The I-register is also incremented without time penalty:

   I (500) +1 → I (501).

5. When the Memory Data register, D, becomes available it contains the instruction. In this case, a short format is chosen and is as follows:

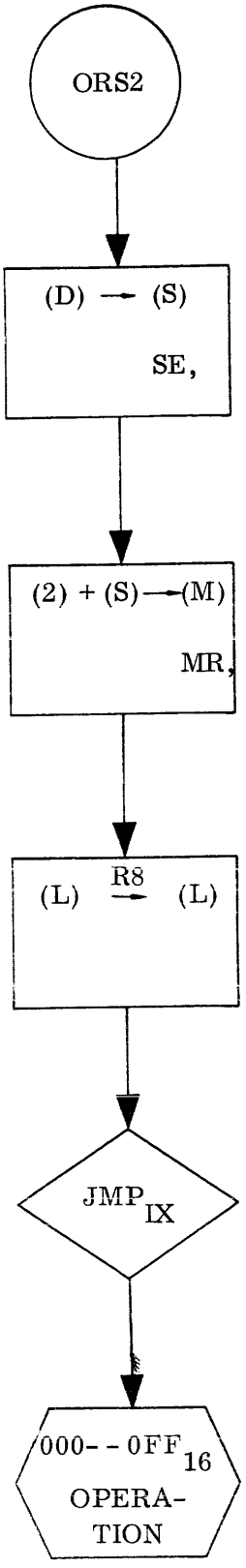


   $D = 9210_{16}$. D is shifted right eight places and stored in L. $L = 92_{16}$ at end of instruction.

6. A Load instruction with the address field set to $E00_{16}$ is used to read the contents of the table starting at E00. The Load instruction is indexed logically by the Link register giving an effective Read-Only Memory (ROM) address of $E92_{16}$. In this example, the contents of ROM location E92 are placed in the Link register. The Link register, L, now contains 4410_{16}. Load is a two-cycle instruction.

7. A Jump-to-F00 indexed by the Link causes program execution in the ROM to continue at location $F10_{16}$. (4410_{16} OR $F00_{16} = 4F10_{16}$. The ROM Address register is 12 bits; therefore, the effective address is $F10_{16}$.)

FIGURE 4-1. SAMPLE FLOWCHART,
READ NEXT INSTRUCTION (Sheet 2 of 2)



8. COPY INSTRUCTION (MEMORY DATA REGISTER) INTO SCRATCH WITH BIT 8 EXTENDED THROUGH BITS 7 - - 0.

9. ADD INDEX REGISTER 2 TO THE EXPANDED DISPLACEMENT TO FORM THE EFFECTIVE ADDRESS IN THE MEMORY ADDRESS REGISTER; INITIATE MEMORY READ.

10. SHIFT THE LINK RIGHT EIGHT PLACES TO FORM THE OPERATION ADDRESS.

11. UNCONDITIONALLY JUMP TO 000 INDEXED BY THE CONTENTS OF THE LINK.

FIGURE 4-2. SAMPLE FLOWCHART, OPERAND READ, SHORT FORMAT, RELATIVE TO INDEX REGISTER 2 (Sheet 1 of 2)

8. The instruction in the Memory Data register, D , is now copied into a Scratch register, S , with sign extension specified.

$D = 9210_{16}$ $S = 0010_{16}$. S now contains only the displacement.

9. Index Register 2 = 510_{16} is added to $S = 10_{16}$ giving 520_{16} as the effective address of the operand. This address is placed in the Memory Address register and a Read of the operand is initiated.

10. During the time required to fetch the operand, the exit to the operation subroutine may be prepared. The Link register, $L = 4410_{16}$, is shifted right eight places.

$L = 0044_{16}$, which is the address of the Subtract subroutine.

11. A Jump-to-LOC 000 indexed by the contents of the Link register results in an effective Jump-to-LOC 044 of ROM for execution of the Subtract when the operand becomes available in the Memory Data register, D.

FIGURE 4-2. SAMPLE FLOWCHART, OPERAND READ, SHORT FORMAT, RELATIVE TO INDEX REGISTER 2 (Sheet 2 of 2)

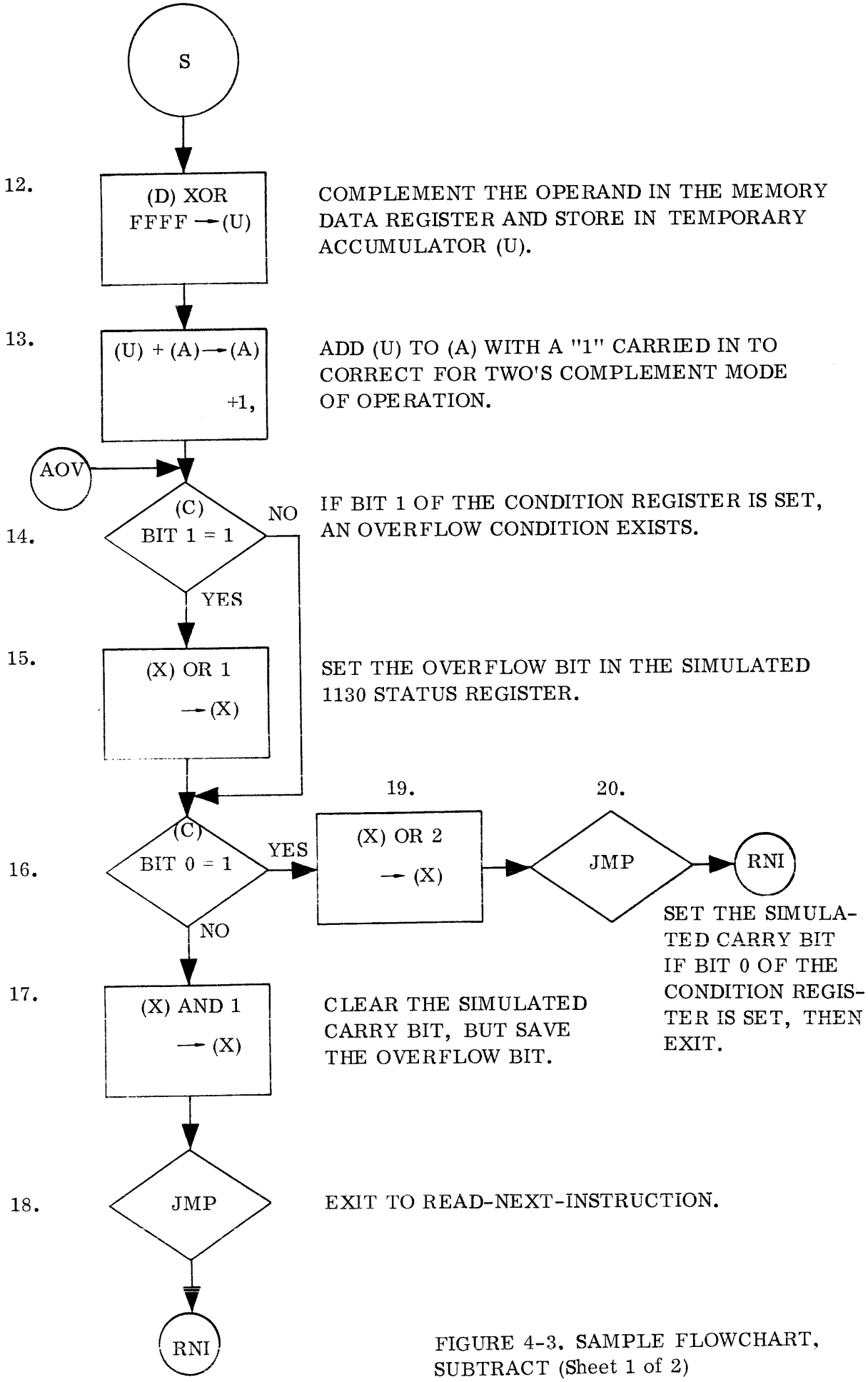


FIGURE 4-3. SAMPLE FLOWCHART, SUBTRACT (Sheet 1 of 2)

12. The operand in D (150_{16}) is complemented by an Exclusive OR with all 1 bits. The result is placed in the U-register:

$$150_{16} \text{ XOR } FFFF_{16} = FEAF_{16}.$$

13. The contents of the U-register ($FEAF_{16}$) are added to the contents of the accumulator (300_{16}) and a 1 bit is forced into the carry-in position for the two's complement mode of operation: $FEAF_{16} + 150_{16} + 1 = 1B0_{16}$ with a carry of 1.

	1111	1110	1010	1111	FEAF
	0000	0011	0000	0000	300
				1	1
1	0000	0001	1011	0000	

14. Bit 1 of the Condition/Counter register, C , contains the overflow status. This condition is tested. For this example, it is zero and the next instruction executed is 16.

15. If the overflow bit is set. a 1 bit is forced into the simulated 1130 status register.

16. Bit 0 of the C-register contains carry status. In this example, the carry is a 1 and control transfers to statement 19.

17. If the carry is zero, the carry bit (bit 14) and all the unused bits are cleared with an AND. The overflow bit (bit 15) is unchanged.

18. An Unconditional Jump to read the next instruction terminates the program.

19. The carry bit of the simulated Status register is forced to a 1 and the overflow bit (bit 15) is unchanged.

20. An Exit-to-RNI is performed.

FIGURE 4-3. SAMPLE FLOWCHART, SUBTRACT (Sheet 2 of 2)

APPENDIX A

POWERS OF TWO

APPENDIX A

POWERS OF TWO

2^n	n	2^{-n}
1	0	1.0
2	1	0.5
4	2	0.25
8	3	0.125
16	4	0.062 5
32	5	0.031 25
64	6	0.015 625
128	7	0.007 812 5
256	8	0.003 906 25
512	9	0.001 953 125
1 024	10	0.000 976 562 5
2 048	11	0.000 488 281 25
4 096	12	0.000 244 140 625
8 192	13	0.000 122 070 312 5
16 384	14	0.000 061 035 156 25
32 768	15	0.000 030 517 578 125
65 536	16	0.000 015 258 789 062 5
131 072	17	0.000 007 629 394 531 25
262 144	18	0.000 003 814 697 265 625
524 288	19	0.000 001 907 348 632 812 5
1 048 576	20	0.000 000 953 674 316 406 25
2 097 152	21	0.000 000 476 837 158 203 125
4 194 304	22	0.000 000 238 418 579 101 562 5
8 388 608	23	0.000 000 119 209 289 550 781 25
16 777 216	24	0.000 000 059 604 644 775 390 625
33 554 432	25	0.000 000 029 802 322 387 695 312 5
67 108 864	26	0.000 000 014 901 161 193 847 656 25
134 217 728	27	0.000 000 007 450 580 596 923 828 125
268 435 456	28	0.000 000 003 725 290 298 461 914 062 5
536 870 912	29	0.000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0.000 000 000 931 322 574 615 478 515 625
2 147 483 648	31	0.000 000 000 465 661 287 307 739 257 812 5
4 294 967 296	32	0.000 000 000 232 830 643 653 869 628 906 25
8 589 934 592	33	0.000 000 000 116 415 321 826 934 814 453 125
17 179 869 184	34	0.000 000 000 058 207 660 913 467 407 226 562 5
34 359 738 368	35	0.000 000 000 029 103 830 456 733 703 613 281 25
68 719 476 736	36	0.000 000 000 014 551 915 228 366 851 806 640 625
137 438 953 472	37	0.000 000 000 007 275 957 614 183 425 903 320 312 5
274 877 906 944	38	0.000 000 000 003 637 978 807 091 712 951 660 156 25
549 755 813 888	39	0.000 000 000 001 818 989 403 545 856 475 830 078 125
1 099 511 627 776	40	0.000 000 000 000 909 494 701 772 928 237 915 039 062 5
2 199 023 255 552	41	0.000 000 000 000 454 747 350 886 464 118 957 519 531 25
4 398 046 511 104	42	0.000 000 000 000 227 373 675 443 232 059 478 759 765 625
8 796 093 022 208	43	0.000 000 000 000 113 686 837 721 616 029 739 379 882 812 5
17 592 186 044 416	44	0.000 000 000 000 056 843 418 860 808 014 869 689 941 406 25
35 184 372 088 832	45	0.000 000 000 000 028 421 709 430 404 007 434 844 970 703 125
70 368 744 177 664	46	0.000 000 000 000 014 210 854 715 202 003 717 422 485 351 562 5
140 737 488 355 328	47	0.000 000 000 000 007 105 427 357 601 001 858 711 242 675 781 25
281 474 976 710 656	48	0.000 000 000 000 003 552 713 678 800 500 929 355 621 337 890 625
562 949 953 421 312	49	0.000 000 000 000 001 776 356 839 400 250 464 677 810 668 945 312 5
1 125 899 906 842 624	50	0.000 000 000 000 000 888 178 419 700 125 232 338 905 334 472 656 25
2 251 799 813 685 248	51	0.000 000 000 000 000 444 089 209 850 062 616 169 452 667 236 328 125
4 503 599 627 370 496	52	0.000 000 000 000 000 222 044 604 925 031 308 084 726 333 618 164 062 5
9 007 199 254 740 992	53	0.000 000 000 000 000 111 022 302 462 515 654 042 363 166 809 082 031 25
18 014 398 509 481 984	54	0.000 000 000 000 000 055 511 151 231 257 827 021 181 583 404 541 015 625
36 028 797 018 963 968	55	0.000 000 000 000 000 027 755 575 615 628 913 510 590 791 702 270 507 812 5
72 057 594 037 927 936	56	0.000 000 000 000 000 013 877 787 807 814 456 755 295 395 851 135 253 906 25
144 115 188 075 855 872	57	0.000 000 000 000 000 006 938 893 903 907 228 377 647 697 925 567 626 953 125
288 230 376 151 711 744	58	0.000 000 000 000 000 003 469 446 951 953 614 188 823 848 962 783 813 476 562 5
576 460 752 303 423 488	59	0.000 000 000 000 000 001 734 723 475 976 807 094 411 924 481 391 906 738 281 25
1 152 921 504 606 846 976	60	0.000 000 000 000 000 000 867 361 737 988 403 547 205 962 240 695 953 369 140 625
2 305 843 009 213 693 952	61	0.000 000 000 000 000 000 433 680 868 994 201 773 602 981 120 347 976 684 570 312 5
4 611 686 018 427 387 904	62	0.000 000 000 000 000 000 216 840 434 497 100 886 801 490 560 173 988 342 285 156 25

APPENDIX B

ASCII, TELETYPE, AND HOLLERITH CODES

APPENDIX B

ASCII, TELETYPE, AND HOLLERITH CODES

Hex. Code	Character Name	ASCII Character	Teletype Character	Key or Key Combinations	Hollerith Punch Code
80	Null/Idle	NULL	---	CTRL @	---
81	Start of Message	SOM	---	CTRL A	---
82	End of Address	EOA	---	CTRL B	---
83	End of Message	EOM	---	CTRL C	---
84	End of Transmission	EOT	---	CTRL D	---
85	Who Are You	WRU	---	CTRL E	---
86	Are You	RU	---	CTRL F	---
87	Audible Signal	BELL	---	CTRL G	---
88	Format Effector	FE	---	CTRL H	---
89	Horizontal Tabulation	H TAB	---	CTRL I	---
8A	Line Feed	LF	---	CTRL J	---
8B	Vertical Tabulation	V TAB	---	CTRL K	---
8C	Form Feed	FF	---	CTRL L	---
8D	Carriage Return	CR	---	CTRL M	---
8E	Shift Out	SO	---	CTRL N	---
8F	Shift In	SI	---	CTRL O	---
90	Device Control Reversed for Data Line Escape	DC0	---	CTRL P	---
91	Device Control ON	DC1	---	CTRL Q	---
92	Device Control (TAPE)	DC2	---	CTRL R	---
93	Device Control OFF	DC3	---	CTRL S	---
94	Device Control	DC4	---	CTRL T	---
95	Error	ERR	---	CTRL U	---
96	Synchronous Idle	SYNC	---	CTRL V	---
97	Logical End of Media	LEM	---	CTRL W	---
98	Separator, Information	S0	---	CTRL X	---
99	Separator, Data Delimiters	S1	---	CTRL Y	---
9A	Separator, Words	S2	---	CTRL Z	---
9B	Separator, Groups	S3	---	SHIFT CTRL K	---
9C	Separator, Records	S4	---	SHIFT CTRL L	---
9D	Separator, Files	S5	---	SHIFT CTRL M	---
9E	Separator, Misc.	S6	---	SHIFT CTRL N	---
9F	Separator, Misc.	S7	---	SHIFT CTRL O	---
A0	Space	SP	Space	Space Bar	**

** Space - No character is printed.

Hex. Code	Character Name	ASCII Character	Teletype Character	Key or Key Combinations	Hollerith Punch Code
A1	Exclamation Point	!	!	SHIFT !	***
A2	Quotation Marks	"	"	SHIFT "	12-8-7
A3	Number Sign	#	#	SHIFT #	11-8-6
A4	Dollar Sign	$	$	SHIFT $	11-8-3
A5	Percent Sign	%	%	SHIFT %	12-8-5
A6	Ampersand	&	&	SHIFT &	8-6
A7	Apostrophe	'	'	SHIFT '	8-7
A8	Parenthesis, Beginning	(	(	SHIFT (	0-8-4
A9	Parenthesis, Ending	)	)	SHIFT)	12-8-4
AA	Asterisk	*	*	SHIFT *	11-8-4
AB	Plus Sign	+	+	SHIFT +	12
AC	Comma	,	,	,	0-8-3
AD	Hyphen	-	-	-	11
AE	Period	.	.	.	12-8-3
AF	Virgule	/	/	/	0-1
B0	Numeral 0	0	0	0	0
B1	Numeral 1	1	1	1	1
B2	Numeral 2	2	2	2	2
B3	Numeral 3	3	3	3	3
B4	Numeral 4	4	4	4	4
B5	Numeral 5	5	5	5	5
B6	Numeral 6	6	6	6	6
B7	Numeral 7	7	7	7	7
B8	Numeral 8	8	8	8	8
B9	Numeral 9	9	9	9	9
BA	Colon	:	:	:	0-8-6
BB	Semicolon	;	;	;	8-4
BC	Less Than	<	<	SHIFT <	11-8-5
BD	Equals	=	=	SHIFT =	8-3
BE	Greater Than	>	>	SHIFT >	0-8-5
BF	Interrogation Point	?	?	SHIFT ?	12-8-6
C0	At	@	@	SHIFT @	—
C1	Letter A	A	A	A	12-1

***No Hollerith Punch or Card Code available for this character.

Hex. Code	Character Name	ASCII Character	Teletype Character	Key or Key Combinations	Hollerith Punch Code
C2	Letter B	B	B	B	12-2
C3	Letter C	C	C	C	12-3
C4	Letter D	D	D	D	12-4
C5	Letter E	E	E	E	12-5
C6	Letter F	F	F	F	12-6
C7	Letter G	G	G	G	12-7
C8	Letter H	H	H	H	12-8
C9	Letter I	I	I	I	12-9
CA	Letter J	J	J	J	11-1
CB	Letter K	K	K	K	11-2
CC	Letter L	L	L	L	11-3
CD	Letter M	M	M	M	11-4
CE	Letter N	N	N	N	11-5
CF	Letter O	O	O	O	11-6
D0	Letter P	P	P	P	11-7
D1	Letter Q	Q	Q	Q	11-8
D2	Letter R	R	R	R	11-9
D3	Letter S	S	S	S	0-2
D4	Letter T	T	T	T	0-3
D5	Letter U	U	U	U	0-4
D6	Letter V	V	V	V	0-5
D7	Letter W	W	W	W	0-6
D8	Letter X	X	X	X	0-7
D9	Letter Y	Y	Y	Y	0-8
DA	Letter Z	Z	Z	Z	0-9
DB	Bracket, Left	[	[	SHIFT K	***
DC	Reverse Virgule	\	\	SHIFT L	***
DD	Bracket, Right	]	]	SHIFT M	***
DE	Up Arrow (exponentation)	↑	↑	SHIFT	---
DF	Left Arrow	←	←	SHIFT	---
E1 through FC are not available					
FD	Unassigned Control	1	---	ALT MODE	---
FE	Not Available				
FF	Delete/Idle/Rub Out	DEL	---	RUB OUT	---

*** No Hollerith Punch or Card Code available for this character.

APPENDIX C

TABLE OF BINARY-TO-HEXADECIMAL CONVERSION

APPENDIX C

TABLE OF BINARY-TO-HEXADECIMAL CONVERSION

Binary	Hexdecimal
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

APPENDIX D

HEXADECIMAL-TO-DECIMAL CONVERSION TABLE

APPENDIX D

HEXADECIMAL-TO-DECIMAL CONVERSION TABLE

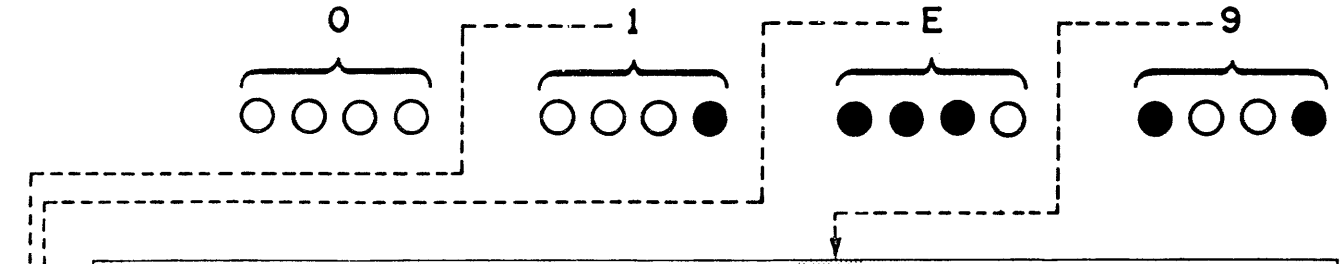
The table in this appendix provides for direct conversion of decimal and hexadecimal numbers in these ranges:

Hexadecimal	Decimal
000 to FFF	0000 to 4095

For numbers outside the range of the table, add the following values to the table figures:

Hexadecimal	Decimal
1000	4096
2000	8192
3000	12288

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



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	0010	0011	0012	0013	0014	0015
010	0016	0017	0018	0019	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	0030	0031
020	0032	0033	0034	0035	0036	0037	0038	0039	0040	0041	0042	0043	0044	0045	0046	0047
030	0048	0049	0050	0051	0052	0053	0054	0055	0056	0057	0058	0059	0060	0061	0062	0063
040	0064	0065	0066	0067	0068	0069	0070	0071	0072	0073	0074	0075	0076	0077	0078	0079
050	0080	0081	0082	0083	0084	0085	0086	0087	0088	0089	0090	0091	0092	0093	0094	0095
060	0096	0097	0098	0099	0100	0101	0102	0103	0104	0105	0106	0107	0108	0109	0110	0111
070	0112	0113	0114	0115	0116	0117	0118	0119	0120	0121	0122	0123	0124	0125	0126	0127
080	0128	0129	0130	0131	0132	0133	0134	0135	0136	0137	0138	0139	0140	0141	0142	0143
090	0144	0145	0146	0147	0148	0149	0150	0151	0152	0153	0154	0155	0156	0157	0158	0159
0A0	0160	0161	0162	0163	0164	0165	0166	0167	0168	0169	0170	0171	0172	0173	0174	0175
0B0	0176	0177	0178	0179	0180	0181	0182	0183	0184	0185	0186	0187	0188	0189	0190	0191
0C0	0192	0193	0194	0195	0196	0197	0198	0199	0200	0201	0202	0203	0204	0205	0206	0207
0D0	0208	0209	0210	0211	0212	0213	0214	0215	0216	0217	0218	0219	0220	0221	0222	0223
0E0	0224	0225	0226	0227	0228	0229	0230	0231	0232	0233	0234	0235	0236	0237	0238	0239
0F0	0240	0241	0242	0243	0244	0245	0246	0247	0248	0249	0250	0251	0252	0253	0254	0255
100	0256	0257	0258	0259	0260	0261	0262	0263	0264	0265	0266	0267	0268	0269	0270	0271
110	0272	0273	0274	0275	0276	0277	0278	0279	0280	0281	0282	0283	0284	0285	0286	0287
120	0288	0289	0290	0291	0292	0293	0294	0295	0296	0297	0298	0299	0300	0301	0302	0303
130	0304	0305	0306	0307	0308	0309	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319
140	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	0330	0331	0332	0333	0334	0335
150	0336	0337	0338	0339	0340	0341	0342	0343	0344	0345	0346	0347	0348	0349	0350	0351
160	0352	0353	0354	0355	0356	0357	0358	0359	0360	0361	0362	0363	0364	0365	0366	0367
170	0368	0369	0370	0371	0372	0373	0374	0375	0376	0377	0378	0379	0380	0381	0382	0383
180	0384	0385	0386	0387	0388	0389	0390	0391	0392	0393	0394	0395	0396	0397	0398	0399
190	0400	0401	0402	0403	0404	0405	0406	0407	0408	0409	0410	0411	0412	0413	0414	0415
1A0	0416	0417	0418	0419	0420	0421	0422	0423	0424	0425	0426	0427	0428	0429	0430	0431
1B0	0432	0433	0434	0435	0436	0437	0438	0439	0440	0441	0442	0443	0444	0445	0446	0447
1C0	0448	0449	0450	0451	0452	0453	0454	0455	0456	0457	0458	0459	0460	0461	0462	0463
1D0	0464	0465	0466	0467	0468	0469	0470	0471	0472	0473	0474	0475	0476	0477	0478	0479
1E0	0480	0481	0482	0483	0484	0485	0486	0487	0488	0489	0490	0491	0492	0493	0494	0495
1F0	0496	0497	0498	0499	0500	0501	0502	0503	0504	0505	0506	0507	0508	0509	0510	0511

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
200	0512	0513	0514	0515	0516	0517	0518	0519	0520	0521	0522	0523	0524	0525	0526	0527
210	0528	0529	0530	0531	0532	0533	0534	0535	0536	0537	0538	0539	0540	0541	0542	0543
220	0544	0545	0546	0547	0548	0549	0550	0551	0552	0553	0554	0555	0556	0557	0558	0559
230	0560	0561	0562	0563	0564	0565	0566	0567	0568	0569	0570	0571	0572	0573	0574	0575
240	0576	0577	0578	0579	0580	0581	0582	0583	0584	0585	0586	0587	0588	0589	0590	0591
250	0592	0593	0594	0595	0596	0597	0598	0599	0600	0601	0602	0603	0604	0605	0606	0607
260	0608	0609	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	0620	0621	0622	0623
270	0624	0625	0626	0627	0628	0629	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639
280	0640	0641	0642	0643	0644	0645	0646	0647	0648	0649	0650	0651	0652	0653	0654	0655
290	0656	0657	0658	0659	0660	0661	0662	0663	0664	0665	0666	0667	0668	0669	0670	0671
2A0	0672	0673	0674	0675	0676	0677	0678	0679	0680	0681	0682	0683	0684	0685	0686	0687
2B0	0688	0689	0690	0691	0692	0693	0694	0695	0696	0697	0698	0699	0700	0701	0702	0703
2C0	0704	0705	0706	0707	0708	0709	0710	0711	0712	0713	0714	0715	0716	0717	0718	0719
2D0	0720	0721	0722	0723	0724	0725	0726	0727	0728	0729	0730	0731	0732	0733	0734	0735
2E0	0736	0737	0738	0739	0740	0741	0742	0743	0744	0745	0746	0747	0748	0749	0750	0751
2F0	0752	0753	0754	0755	0756	0757	0758	0759	0760	0761	0762	0763	0764	0765	0766	0767
300	0768	0769	0770	0771	0772	0773	0774	0775	0776	0777	0778	0779	0780	0781	0782	0783
310	0784	0785	0786	0787	0788	0789	0790	0791	0792	0793	0794	0795	0796	0797	0798	0799
320	0800	0801	0802	0803	0804	0805	0806	0807	0808	0809	0810	0811	0812	0813	0814	0815
330	0816	0817	0818	0819	0820	0821	0822	0823	0824	0825	0826	0827	0828	0829	0830	0831
340	0832	0833	0834	0835	0836	0837	0838	0839	0840	0841	0842	0843	0844	0845	0846	0847
350	0848	0849	0850	0851	0852	0853	0854	0855	0856	0857	0858	0859	0860	0861	0862	0863
360	0864	0865	0866	0867	0868	0869	0870	0871	0872	0873	0874	0875	0876	0877	0878	0879
370	0880	0881	0882	0883	0884	0885	0886	0887	0888	0889	0890	0891	0892	0893	0894	0895
380	0896	0897	0898	0899	0900	0901	0902	0903	0904	0905	0906	0907	0908	0909	0910	0911
390	0912	0913	0914	0915	0916	0917	0918	0919	0920	0921	0922	0923	0924	0925	0926	0927
3A0	0928	0929	0930	0931	0932	0933	0934	0935	0936	0937	0938	0939	0940	0941	0942	0943
3B0	0944	0945	0946	0947	0948	0949	0950	0951	0952	0953	0954	0955	0956	0957	0958	0959
3C0	0960	0961	0962	0963	0964	0965	0966	0967	0968	0969	0970	0971	0972	0973	0974	0975
3D0	0976	0977	0978	0979	0980	0981	0982	0983	0984	0985	0986	0987	0988	0989	0990	0991
3E0	0992	0993	0994	0995	0996	0997	0998	0999	1000	1001	1002	1003	1004	1005	1006	1007
3F0	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023

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

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

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

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

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

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
E00	3584	3585	3586	3587	3583	3589	3590	3591	3592	3593	3594	3595	3596	3597	3598	3599
E10	3600	3601	3602	3603	3604	3605	3606	3607	3608	3609	3610	3611	3612	3613	3614	3615
E20	3616	3617	3618	3619	3620	3621	3622	3623	3624	3625	3626	3627	3628	3629	3630	3631
E30	3632	3633	3634	3635	3636	3637	3638	3639	3640	3641	3642	3643	3644	3645	3646	3647
E40	3648	3649	3650	3651	3652	3653	3654	3655	3656	3657	3658	3659	3660	3661	3662	3663
E50	3664	3665	3666	3667	3668	3669	3670	3671	3672	3673	3674	3675	3676	3677	3678	3679
E60	3680	3681	3682	3683	3684	3685	3686	3687	3688	3689	3690	3691	3692	3693	3694	3695
E70	3696	3697	3698	3699	3700	3701	3702	3703	3704	3705	3706	3707	3708	3709	3710	3711
E80	3712	3713	3714	3715	3716	3717	3718	3719	3720	3721	3722	3723	3724	3725	3726	3727
E90	3728	3729	3730	3731	3732	3733	3734	3735	3736	3737	3738	3739	3740	3741	3742	3743
EA0	3744	3745	3746	3747	3748	3749	3750	3751	3752	3753	3754	3755	3756	3757	3758	3759
EB0	3760	3761	3762	3763	3764	3765	3766	3767	3768	3769	3770	3771	3772	3773	3774	3775
EC0	3776	3777	3778	3779	3780	3781	3782	3783	3784	3785	3786	3787	3788	3789	3790	3791
ED0	3792	3793	3794	3795	3796	3797	3798	3799	3800	3801	3802	3803	3804	3805	3806	3807
EE0	3808	3809	3810	3811	3812	3813	3814	3815	3816	3817	3818	3819	3820	3821	3822	3823
EF0	3824	3825	3826	3827	3828	3829	3830	3831	3832	3833	3834	3835	3836	3837	3838	3839
F00	3840	3841	3842	3843	3844	3845	3846	3847	3848	3849	3850	3851	3852	3853	3854	3855
F10	3856	3857	3858	3859	3860	3861	3862	3863	3864	3865	3866	3867	3868	3869	3870	3871
F20	3872	3873	3874	3875	3876	3877	3878	3879	3880	3881	3882	3883	3884	3885	3886	3887
F30	3888	3889	3890	3891	3892	3893	3894	3895	3896	3897	3898	3899	3900	3901	3902	3903
F40	3904	3905	3906	3907	3908	3909	3910	3911	3912	3913	3914	3915	3916	3917	3918	3919
F50	3920	3921	3922	3923	3924	3925	3926	3927	3928	3929	3930	3931	3932	3933	3934	3935
F60	3936	3937	3938	3939	3940	3941	3942	3943	3944	3945	3946	3947	3948	3949	3950	3951
F70	3952	3953	3954	3955	3956	3957	3958	3959	3960	3961	3962	3963	3964	3965	3966	3967
F80	3968	3969	3970	3971	3972	3973	3974	3975	3976	3977	3978	3979	3980	3981	3982	3983
F90	3984	3985	3986	3987	3988	3989	3990	3991	3992	3993	3994	3995	3996	3997	3998	3999
FA0	4000	4001	4002	4003	4004	4005	4006	4007	4008	4009	4010	4011	4012	4013	4014	4015
FB0	4016	4017	4018	4019	4020	4021	4022	4023	4024	4025	4026	4027	4028	4029	4030	4031
FC0	4032	4033	4034	4035	4036	4037	4038	4039	4040	4041	4042	4043	4044	4045	4046	4047
FD0	4048	4049	4050	4051	4052	4053	4054	4055	4056	4057	4058	4059	4060	4061	4062	4063
FE0	4064	4065	4066	4067	4068	4069	4070	4071	4072	4073	4074	4075	4076	4077	4078	4079
FF0	4080	4081	4082	4083	4084	4085	4086	4087	4088	4089	4090	4091	4092	4093	4094	4095

COMMENT SHEET

THE DIGITAL SCIENTIFIC META 4™ SERIES 16 COMPUTER SYSTEM

PRELIMINARY SYSTEM MANUAL

Publication No. 7006MO
(Revision E)

FROM: Name: _____

Business Address: _____

COMMENTS: (Describe errors, suggested additions or deletions, etc.; please include page number.)

Cut Along Line

Form No. DSC7

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.
FOLD ON DOTTED LINES AND STAPLE

Staple

Staple

Fold

Fold

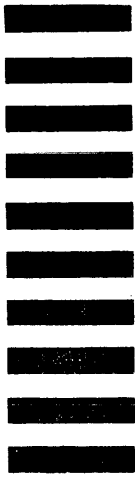
First Class
Permit No. 6225
San Diego, Calif.

BUSINESS REPLY MAIL
No Postage Stamp Necessary If Mailed in the United States

— POSTAGE WILL BE PAID BY —

DIGITAL SCIENTIFIC CORPORATION
11455 Sorrento Valley Road
San Diego, California 92121

Attention: Marketing Department, Publications Group



Cut Along Line

Fold

Fold

Staple

Staple