GCH
7/62

THE

TRW-330

DIGITAL CONTROL COMPUTER

VOLUME I

PROGRAMMING MANUAL

Thompson Ramo Wooldridge Inc.
TRW Computers Company *(division)*
8433 FALLBROOK AVENUE • CANOGA PARK, CALIFORNIA

SUMMARY OF COMMANDS AND EXECUTION TIMES

Command	Word Times	Micro-seconds	Command	Word Times	Micro-seconds
Load A	2	266	Shift AB Left Arithmetic/Logical	28*	3724*
Load B	2	266	Shift AB Right Arithmetic/Logical	28	3724*
Load C	2	266	Float Shift A Left Arithmetic	28	3724
Load D	2	266	Float Shift A Left Logical	28	3724
Load Index	2	266	Float Shift A Left Logical Closed	28	3724
Load X	2	266	Float Shift AB Left Arithmetic	28	3724
Store A	2	266	Float Shift AB Left Arithmetic/Logical	28	3724
Store A Repeat	1(X+2)	133 (X+2)	Jump Unconditionally	(OPS-CIS)	(OPS-CIS) 133
Store B	2	266	Jump A Zero	2	266
Store D	2	266	Jump A Negative	2	266
Exchange A and B	2	266	Jump A Low Bit	2	266
Exchange A and C	2	266	Jump Index	2	266
Exchange A and D	2	266	Jump Overflow	2	266
Exchange A and I	2	266	Jump Parity Error	2	266
Replace A With I	2	266	Stop--Jump Resume	*	*
Replace I With A	2	266	Jump Record Address 0	(OPS-CIS)	(OPS-CIS) 133
Exchange A and X	2	266			
Exchange A and X Low Bits	2	266	Jump Record Address 1	(OPS-CIS)	(OPS-CIS) 133
Replace A With X	2	266			
Replace A With X Low Bits	2	266	Jump Record Address C	(OPS-CIS)	(OPS-CIS) 133
Replace X With A	2	266			
Exchange A and M	2	266	Block Transfer	*	*
Masked Exchange A and M	2	266	Scan Table Greater Than	*	*
Replace A With M	2	266	Scan Table Less Than	*	*
Replace M With A	2	266	Scan Table Equal	*	*
Exchange A and Q	2	266	Scan Table Not Equal	*	*
Replace A With Q	2	266	Scan Analog	*	*
Replace Q With A	2	266	Scan Digital Input	*	*
Clear A and B	2	266	Inhibit Interrupt On	2	266
Complement A	2	266	Inhibit Interrupt Off	2	266
Merge	2	266	Set Analog Sequence	2	266
Extract	2	266	Activate Control Signal	2	266
Reduce Index	2	266	Digital Input: Flexowriter	2	266
Add	2	266	Digital Input: Toggle Switches	2	266
Subtract	2	266	Digital Input: Teletype Reader	2	266
Multiply 27	30	3990	Digital Input: High-Speed Tape Reader	2	266
Multiply 21	24	3192	Digital Input: Console Switches	2	266
Multiply 14	21	2793	Digital Input: Digitran Switches S26-21	2	266
Multiply 7	10	1330	Digital Input: Digitran Switches S36-31	2	266
Divide 27	31	4123	Digital Input: Digital Clock	2	266
Divide 21	25	3325	Digital Input: Group Inputs	2	266
Divide 14	18	2394	Digital Output: Flexowriter	2	266
Divide 7	11	1463	Digital Output: Output Buffer	2	266
Square Root	16	2128	Digital Output: Single-Bit Outputs On	2	266
Shift A Left Arithmetic	28*	3724*	Digital Output: Single-Bit Outputs Off	2	266
Shift A Right Arithmetic	28*	3724*	Digital Output: High-Speed Tape Punch	2	266
Shift A Left Logical	28	3724	Digital Output: Multibit Output	2	266
Shift A Right Logical	28	3724	Digital Output: Logging Typewriter	2	266
Shift A Left Logical Closed	28*	3724*			
Shift AB Left Arithmetic	28*	3724*			

*See tabular listings

TRW-330 Digital Control Computer

VOLUME I

PROGRAMMING MANUAL

(Preliminary)

May 4, 1962

Thompson Ramo Wooldridge Inc.
TRW Computers Company *(division)*
8433 FALLBROOK AVENUE • CANOGA PARK, CALIFORNIA

TABLE OF CONTENTS

SECTION 1

INTRODUCTION

SCOPE OF THIS MANUAL

This manual provides information essential to a general understanding of the programming characteristics of the TRW-330 digital control computer, particularly those characteristics that are relevant to industrial control applications, for which the TRW-330 has been designed.

In order that this manual may be used as a basic programming handbook for the TRW-330 computer, a detailed breakdown has been included on the computer's flexible instruction repertoire and its various modes of operation.

In some cases, where more detailed information is available in separate manuals, the reader is referred to the place where the detailed information is available.

NOTE

It is the intent of this manual to describe every feature available with the TRW-330 computer. Therefore some of the material presented here will probably not be applicable to any given TRW-330 control installation.

SECTION 2

TRW-330 CHARACTERISTICS AND CAPABILITIES

GENERAL

The TRW-330 is a stored-program, digital control computer designed specifically for real-time industrial control applications. It is a 2's complement binary machine.

The computer is assembled in two or more vertical rack-and-panel cabinets bolted together to form a convenient package. Each cabinet is 84 inches high, 23 inches wide, and 24 inches deep.

The general design features of the TRW-330 computer include the following:

(a) The circulating registers are completely solid state.

(b) No vacuum tubes are used in the machine.

(c) The motor that turns the magnetic drum is an induction type, with excellent starting characteristics and no tendency to "drop out" during line fluctuations.

(d) The time-of-day clock uses mercury-wetted-contact relays.

(e) The guarded sections of memory are protected by toggle switches or, where program control of the guard feature is required, by mercury-wetted-contact relays.

(f) The analog input system can sample sixty points per second using a single amplifier, so that only one amplifier is required for each TRW-330 system. Furthermore, the amplifier used is a solid state type, including the choppers.

MEMORY

The memory of the TRW-330 computer is a magnetic drum that rotates at approximately 3540 revolutions per minute. The drum contains from 64 to 1024 tracks of 128 sectors, or words, per track.

Word time for the TRW-330 computer is approximately 133 microseconds. Each drum revolution takes approximately 17 milliseconds.

DATA AND INSTRUCTION WORDS

Each word in the TRW-330 consists of 28 binary digits, or bits.

The binary information contained in a TRW-330 word may represent numbers, alphabetic codes, control patterns, etc. Such words are called data words.

The binary contents of a word may represent a computer instruction such as add, subtract, jump, etc. These words are called instruction words.

The 28-bit data word consists of a sign bit and 27 magnitude bits.

The 28-bit instruction word contains two fields: a 10-bit command field and a 16-bit operand field; one of the bits are not used in an instruction word. The operand field may contain either an operand address or the operand itself.

The long word length provides certain inherent advantages: it allows direct addressing of the largest memory the machine can contain; by allowing an operand to be contained within the instruction, memory storage requirements are reduced and the need for accessing such constants is eliminated. This long word also allows rapid manipulation of messages and coded data.

PARITY CHECKING

Parity is checked on all operations involving memory access.

Each time a word (data or instruction) is stored in memory, a parity bit is generated and stored with the word. Whenever the word is read from memory, another parity bit is generated and compared with the previously stored parity bit. If the parity bits do not agree, a parity error signal is generated.

DIGITAL INPUT-OUTPUT

The TRW-330 computer can accept on-off signals from external sources, permitting digital inputs from such devices as switches, relays, paper-tape readers, time-of-day clocks, contact closures, etc.

Digital output signals can be applied to external equipment such as electric typewriters, paper-tape punches, relays, solenoids, controllers, etc.

ANALOG INPUT-OUTPUT

Variable voltages from measuring instruments and transducers can be automatically sampled and converted to digital form by the TRW-330. Up to 1024 of these analog input signals can be converted to their binary equivalents with a full-scale accuracy of 0.1%, and stored on the memory drum automatically.

The results of control calculations, expressed as binary numbers, can be automatically converted into voltages or currents to be applied to transducers or other control devices.

REGISTERS AND COMMANDS

Several registers, numerous command options, and a number of special commands have been included in the TRW-330 computer to aid programming, increase the effective speed of the machine, and to solve certain problems frequently encountered in industrial control. The standard computer has five registers (B, C, D, I, and X) (in addition to the accumulator A register) that are under the control of the programmer. All of these registers are switchable with the accumulator. Three of these registers (B, C, and D) are addressable, so that their contents can be used as operands. Two of these registers and the accumulator (A, B, and D) are directly storable in the memory. Some of the registers have special features: one is an index (I) register that can be incremented or decremented by any amount up to $2^{17}-1$. The X register functions in a float, or shift and count, command and in scan commands.

A repeat command permits a single word to be recorded in sequential sectors of any track.

Multiply and divide instructions can be performed in quarter-, half-, and three-quarter-length increments, as well as full-length; this makes it possible to match the command length to the word length for these commands, and thus substantially reduce program running time.

An automatic computer stop command is standard.

The ability to mark the current place in the program, branch to a subroutine, execute the subroutine, and subsequently return to the marked place, is standard on the TRW-330.

The block transfer feature is standard on the TRW-330; the block may be any length up to 128 words.

Analog and digital input scan commands and a table search command provide capabilities that are unusual in a drum computer. The analog scan, specially designed for industrial control, allows the program to compare 128 converted analog inputs against upper and lower limits in 34 milliseconds. Digital inputs can be scanned to detect a change from a previously stored image at the rate of one line every ten microseconds.

The table search command permits words recorded on the sectors of one track (comparison track) to be compared with the contents of corresponding sectors of a specified reference track. The search can be made for equality between the comparison word and the reference word, for inequality, for less than, and for greater than. The search is masked by a word loaded into one of the working registers; thus any single bit position, or any combination of bit positions, or entire words may be compared.

During execution of the table search command, the computer sequentially compares each comparison word (results of calculations, inputs, etc.) with its corresponding reference word. When the condition sought for is found, the computer identifies the word, and its location, that satisfies the condition.

For systems entailing heavy logging or tape punching duties, a 512-character output buffer is available to drive logging typewriters, Flexowriters, and/or paper tape punches.

The contents of all computer registers used by the programmer can be displayed on the computer control panel during program execution. Under program control, the contents of any register or memory cell may be typed out on the Flexowriter, or displayed on the operator's console display in modified form.

INTERRUPT

An interrupt feature of the TRW-330 allows the computer program to be interrupted by a switch closure occurring in circuits external to the computer. Upon receipt of an interrupt signal, the computer automatically stores the address of the next instruction as a re-entry address, and then transfers to the interrupt program. After determining the source and nature of the interrupt and taking responsive action, the computer returns to the master control program and continues at that point where it was interrupted. As many as 112 priority interrupts are available.

PROGRAMMING SOFTWARE AND MAINTENANCE CHECKING

A library of subroutines and interpretive routines is available to extend the capabilities of the TRW-330 computer to scientific and general-purpose applications.

Generally speaking, anything that helps the programmer also assists the maintenance man. Numerous features built into the TRW-330 for the benefit of the programmer will aid speedy localization of failures. For instance, the block transfer command and the table comparison command (one mode of the scan command), coupled with the parity check, make it possible to quickly localize a memory fault. A toggle switch allows the maintenance engineer to cause the machine to stop at any point, rather than merely record a parity error.

The flexible command structure and memory organization makes it easy to keep certain maintenance programs, such as the utility package, permanently stored in the machine.

Features that reduce human fatigue and assist in checking programs include: (a) the ability to remain seated while stepping through a program; (b) easily read neon bulb displays; (c) only one button need be depressed to step through an instruction; (d) the stop-on-jump switch makes it possible to stop at predetermined points in the program.

TYPICAL TRW-330 INSTALLATION

A process control installation by TRW Computers Company consists typically of the TRW-330 digital control computer, its analog input-output subsystem, and one or more optional subsystems depending upon the requirements of the specific installation. These subsystems may include expanded interrupts, priority interrupts, selective analog updating, programmed computer operations checks, digital output buffering to logging typewriters, and paper tape chromatograph peak integrators, etc.

SECTION 3

MEMORY ORGANIZATION

GENERAL

The TRW-330 can address directly as many as 1024 tracks and any one of 128 sectors of each track.

Tracks and sectors are numbered in octal notation. The first track on the memory drum is numbered 000. The last track on the memory drum is numbered 177 for a 128 track drum, 377 for a 256 track drum, 777 for a 512 track drum, and 1777 for a 1024 track drum. Sectors are numbered 000 through 177.

In the following paragraphs, the track numbers reserved for special purposes are listed. In a specific installation, all of these track reservations may not be required. In those cases, the remaining special-purpose tracks may be used for general storage.

GENERAL STORAGE

General storage is divided into tracks which are writable under program control, and tracks which are non-writable. Non-writable tracks can be written into by manually throwing a switch during program loading. In general, the control programs are entered into the non-writable tracks to preclude any possibility of these programs becoming inadvertently destroyed during normal operation. Writable tracks provide storage for intermediate solution of programmed computations. The ratio of writable to non-writable tracks will vary from one installation to another, depending upon special requirements. Normally, writable and non-writable tracks are arranged in groups of 16.

Addressable

ARITHMETIC REGISTERS

Track numbers 075, 076, and 077 are used to address the B, C, and D arithmetic registers, respectively. Therefore no tracks on the drum correspond to these track addresses. If these track addresses are used, the corresponding arithmetic registers will be addressed.

LOADER-VERIFIER

Tracks 000 and 001 contain a permanent Loader-Verifier routine. The programmer cannot write into this track. The Loader-Verifier is used for entering programs into other tracks, for verifying their accuracy, for entering scaled decimal information into the computer to obviate operator conversions, for entering routines at points designated by the computer control panel breakpoint switches, and for other utility purposes.

The use of these tracks is described in Section 13.

FAST-ACCESS TRACKS

Track addresses 060, 061, 062, and 063 are reserved for temporary storage of fast-access data. Data may be read from or written into these tracks under program control. This information can be accessed in one-fourth the time that data can be accessed on other tracks.

This set of fast-access tracks is actually a single track, of 128 words, with four heads mounted 90 degrees apart. Thus every word on the track can be accessed from any of the four heads, and thus from any of the four track addresses.

Installations requiring 256 words of fast-access data will also have track addresses 054, 055, 056, and 057 reserved for this purpose.

Since each fast-access track uses four track addresses for 128 words, each set reduces the total addressable general storage by three tracks, or 384 words.

DIGITAL OUTPUT TRACK

Track 072 is reserved for digital output buffering. The use of this track is described in Section 8.

ANALOG INPUT TRACKS

Tracks 024 and 025 are reserved for analog input data. Tracks 064 and 065 are reserved for analog input control. The use of these tracks is described in Section 9.

ANALOG OUTPUT TRACK

Track 074 is reserved for analog output buffering. The use of this track is described in Section 10.

INTERRUPT SYSTEM TRACKS

Tracks 037 and 073 are reserved for entry into and exit from priority interrupt subroutines. Track 037 contains the response routine entry address for each interrupt used in the system. When the computer responds to an interrupt, the address for re-entry into the main program is stored in track 073.

The sectors on these two tracks used for interrupts is equal to the number of interrupt lines designed into the specific system. The remaining sectors on both these tracks may be used for general storage.

The use of the interrupt tracks is described in Section 11.

SECTION 4

ARITHMETIC AND CONTROL UNIT

GENERAL

The TRW-330 has nine arithmetic and control registers. Six of the registers are available to the programmer; three registers are used for control purposes and are not directly accessible to the programmer.

Three of the program accessible registers, (A, B, and D), are full word length, each containing 28 bits. Two of the registers, (C and I), contain 16 bits each, and one register, (X), contains 8 bits.

Two of the control registers, (N and Y), are 16 bits in length, and one control register, (E), is 8 bits in length.

Associated with the control unit also are three static registers, two indicator flip-flops, and two to eight interrupt registers.

~~ARITHMETIC~~ REGISTERS

A REGISTER

The A register, 28 bits long, is the principal arithmetic register. The contents of A can be stored into or loaded from memory. A holds the augend, sum, dividend, quotient, multiplicand, and product of arithmetic operations. It can be shifted left or right, or its contents exchanged with other registers. The contents of A can be examined for zero, less than zero, for odd, or for overflow. Data can be merged into or extracted from A.

B REGISTER

The B register, with 28 binary digits, acts as a secondary arithmetic register. It holds the least significant half of the double length product in multiplications, and the remainder in divisions and square roots. The principal function of the B register is to provide temporary storage for intermediate program results. B can be loaded from or stored into memory, exchanged with A, and its contents can be shifted right or left in conjunction with the A register.

The B register is addressable. Its track address is 075.

C REGISTER

The C register contains 16 binary digits. It can be used as temporary storage. The C register is used in multiply and divide commands, which destroy any previous data in this register. It is also used in jump and search commands. The contents of A and C may be exchanged.

The C register is addressable. Its track address is 076.

D REGISTER

The D register is a 28-bit register used principally for temporary storage during calculation. D can be loaded from memory, or its contents exchanged with the contents of A.

The D register is addressable. Its track address is 077.

I REGISTER

The I, or index register, holds 16 binary digits. Its principal purpose is to modify the operand address during the execution of indexed instructions. The I register can be loaded from memory or switched with A. Its contents can be decremented or incremented by any magnitude.

X REGISTER

The X register is an 8-bit register. It can be loaded from memory or its contents exchanged with the 8 least significant bits of A. The X register is used in shift, scan, and store A repeat commands.

CONTROL REGISTERS

N REGISTER

The 16-bit N register is the principal control register of the TRW-330. It is not available to the programmer. In the RESUME-STEP mode of operation, the N register contains the address of the next instruction.

Y REGISTER

The 16-bit Y register is used for intermediate storage of the operand address, and in the RESUME-STEP mode of operation contains the non-indexed operand address in non-indexed instructions, or the modified operand address in indexed instructions, except in multiply and divide commands. It is not available to the programmer.

E REGISTER

The 8-bit E register controls the time required to perform commands which take more than one word time to execute.

STATIC REGISTERS

The three static control registers are those for track address, for command, and for computer state. These registers are not available to the programmer.

INDICATOR FLIP-FLOPS

The two indicator flip-flops associated with the control register are those for overflow and for parity error.

INTERRUPT REGISTERS

For those installations of the TRW-330 that include a priority interrupt system, registers for priority determination (masking) and for holding interrupts are included in the arithmetic and control unit. The masking registers are M registers; the holding registers are Q registers. The interrupt registers may be 8-bit, or 16-bit, or 24-bit, or 28-bit registers, as necessary. The minimum priority interrupt system may include only an 8-bit Q register and an 8-bit M register; the maximum interrupt system, to handle 112 interrupt lines, may include four 28-bit Q registers and four 28-bit M registers.

The interrupt registers are described in more detail in Section 11.

SECTION 5

WORD STRUCTURE AND MACHINE OPERATION

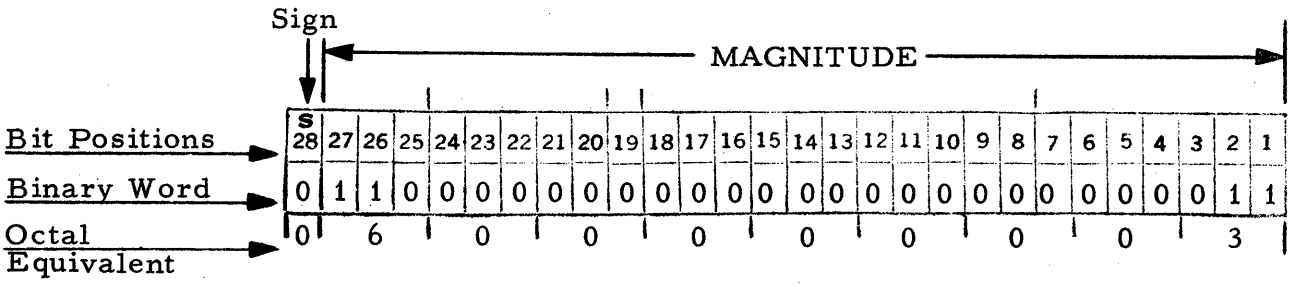
GENERAL

The binary contents of a TRW-330 word may represent numbers, alphabetic codes, control patterns, etc.; these are called data words.

The binary contents of a word may represent a computer instruction such as add, subtract, jump, etc.; these words are called instruction words.

DATA WORDS

A data word consists of 28 binary digits, or bits. The first 27 bits represent the magnitude of the number contained in the word. The 28th bit represents the sign of the magnitude. A binary zero in the sign position indicates that the magnitude is positive. A binary one in the sign position indicates that the magnitude is negative.



The relationship between the binary and octal number systems makes it convenient to group the binary contents of a data word into groups of three, starting at the least significant ~~end~~ bit of the word. In this manner it is possible to express the number octally. The binary and octal equivalents from zero through seven are tabulated as follows.

BINARY	OCTAL
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

From this tabulation the binary contents of the word illustrated above can be converted to the octal equivalent, 0600000003.

Negative quantities are expressed in 2's complement in the TRW-330 computer. To change the sign of any magnitude (positive to negative, or negative to positive), it is necessary merely to change all ones to zeros and all zeros to ones, (including the sign bit), and add one to the result. Thus, the magnitude of the binary number in the above example can be expressed as a negative binary number as:

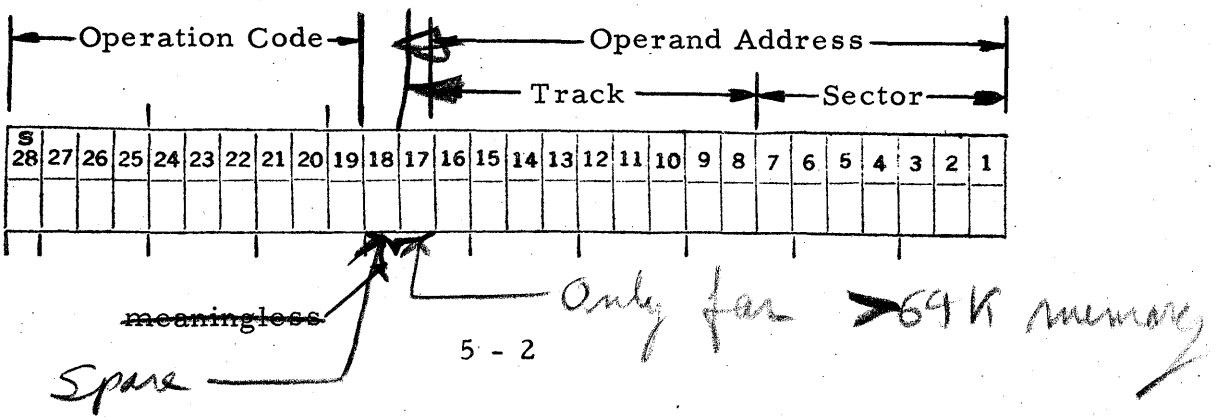
S	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
1	1		7		7		7		7		7		7		7		7		5									

The octal equivalent of this number is 1177777775.

The two numbers, 0600000003 and 1177777775, have equal magnitudes although their signs are opposite. Each is expressed as the complement of the other.

INSTRUCTION WORDS

An instruction word is made up of 28 binary digits divided into two general fields; the operation code (command) field, and the operand address field.



The operation code is contained in the ten high-order bits of the instruction word. The operand address is contained in the 16 low-order bits of the instruction word. The operand address consists of two subfields; the sector address and the track address. Bit position 18 is a spare bit and has no significance in the instruction word. Bit position 17 is used in TRW-330 computers having a memory capacity greater than 64 thousand words to expand the track addresses above 777_8.

It is convenient to indicate the contents of an instruction word in octal notation. The operation code, the operand track address, and the operand sector address should be converted from binary to octal separately according to their fields. The following example illustrates the binary contents of an instruction word and its octal equivalent.

BINARY	0	000	111	010	00	000	100	101	1	100	001	
		OP CODE				OPT			OPS			
						OPA						
OCTAL	0	0	7	2		0	4	5	1	4	1	

In some instructions the contents of the operand address field can be the operand itself rather than the address location of the operand. In this case the octal representation of the operand address is not divided into the sector and track groups as in the example, above, but all 16 bits of the operand address field are converted directly. The following example illustrates the binary and octal equivalent of this type of instruction.

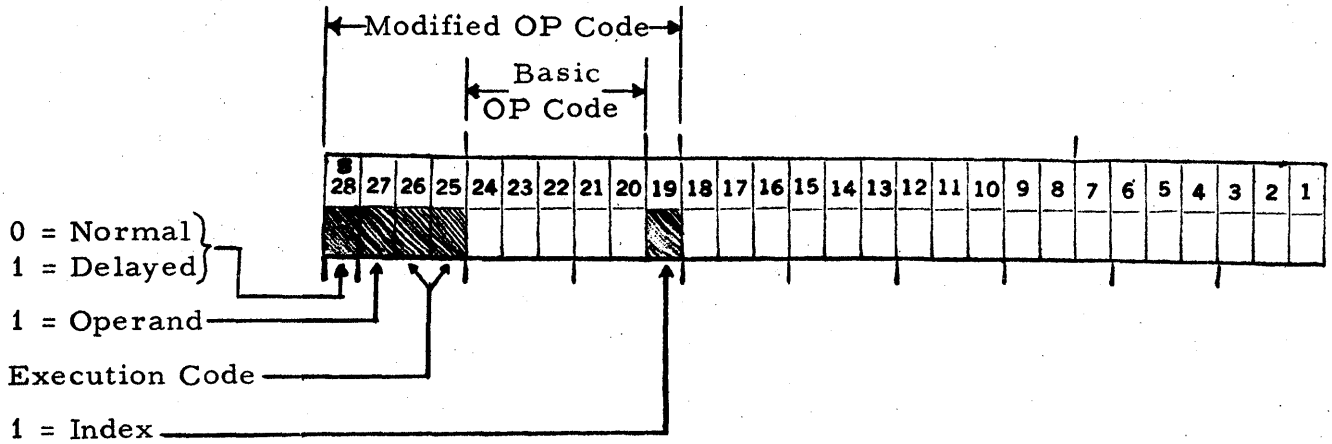
BINARY	0	100	111	010	00	0	001	001	011	100	001	
OCTAL	0	4	7	2	-	0	1	1	3	4	1	

The octal number 011341, in this case, is the actual operand rather than the address of the operand. The largest octal number which can be contained in the operand address field is 177777. This is equivalent to the decimal number 65,535. Operands which are included in the instruction word are treated as positive numbers by the computer.

OPERATION CODES

The TRW-330 computer reads instructions from memory and executes them in proper sequence. An instruction consists of an operation code and an operand address. The data in the 16 bit operand address field may be the address of the operand, or, in certain cases, it may be the actual operand. In cases where the command does not require an operand, (shifts, switches, and digital commands), the data in the operand field can modify the command.

The operation code is made up of the ten high-order bits of the instruction word. Bits 20 through 24 make up the basic operation code and the remaining bits act as command modifiers, or tag bits.



INDEX TAG

Bit 19 is the index tag bit which modifies the operand address of those commands requiring operands or the transfer address of jump commands. The basic operation code together with the index tag bit make up the two least significant octal digits of the operation code. For example, the basic add command is XX70. To index this command it is necessary only to add one to the basic command (XX70 + 1 = XX71). All even numbered operation codes are un-indexed. All odd numbered operation codes are indexed.

The true operand address of indexed instructions is the address located in the operand address field less the magnitude present in the index register: OPA-I. Since the index register contains 16 bits, it is capable of modifying either the track address, the sector address, or both. For example, if the apparent address is 042-036, and the contents of I correspond to 001-002, the true operand address of an indexed instruction would be 041-034. Note, however, how the contents of the index register would appear in binary.

	Binary	Octal
Apparent operand address	0001000100011110	042-036
Less contents of I	0000000010000010	001-002
True address of operand	0001000010011100	041-034

Because all 16 bits of I are subtracted from the apparent operand address of an indexed instruction, any number in the sector field of the index register can cause the apparent operand track address to be modified. For example:

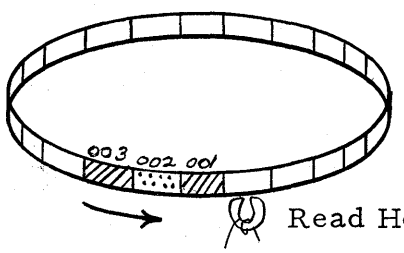
	Binary	Octal
Apparent operand address	01000100011010	042-032
Less contents of I	00000001110000	000-160
True address of operand	01000010101010	041-052

NORMAL OR DELAYED TAG

Tag bit 28 determines the address of the next instruction in relation to either the current instruction or the operand address of the current instruction. A zero in tag bit 28 designates that the instruction operates in the normal mode and that the next instruction address is a function of the current instruction location. A one in tag bit 28 designates that the instruction operates in the delayed mode and that the next instruction address is a function of the operand address. Instructions with the most significant octal digit of the operation code a zero (0XXX), are referred to as normal instructions. Instructions with the most significant octal digit of the operation code a one (1XXX), are referred to as delayed instructions.

NORMAL MODE OPERATION

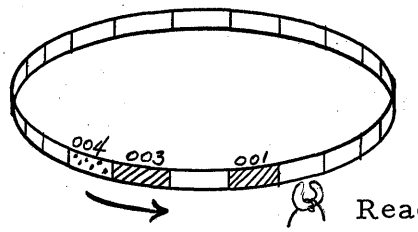
The next instruction address (NIA) of a normal mode instruction is always two greater than the current instruction sector (NIA = CIS + 2). If the operand address (OPA) is one greater than the current instruction address, maximum optimization is achieved. The following figure illustrates how normal mode instructions operate. The time required to read and execute any normal mode instruction where the OPA is one greater than CIA, is two word times, or 266 microseconds.



001 = CIA
002 = OPA for CIA
003 = NIA

The CIA is 001; therefore, the NIA is 003. If the OPA is located in 002, then at the end of word 002, the computer is ready to read the next instruction in 003.

The following figure illustrates the condition where OPA is not one greater than CIA.



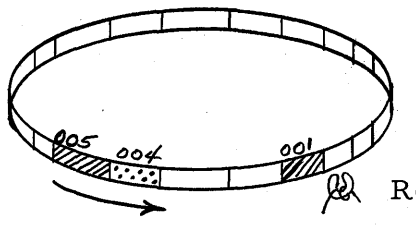
001 = CIA
004 = OPA for CIA
003 = NIA

After reading the CIA in 001, the computer must wait until sector 004 (OPA) comes under the read head before executing the instruction. Sector 003 (NIA) has already passed under the head and the computer must wait for almost one drum revolution before it will be able to read the next instruction. The interim between the time the instruction in 001 is read and the time the instruction in 003 is read is one drum revolution plus two word times.

In this case, use of a delayed mode instruction would save operating time.

DELAYED MODE OPERATION

The next instruction address (NIA) of a delayed mode instruction is always one greater than the operand address (NIA = OPA + 1). Thus, the use of a delayed instruction in the last example given under Normal Mode Operation would prevent the computer from waiting for almost one drum revolution before reading the next instruction. The following figure illustrates the manner in which a delayed instruction locates the next instruction address.



001 = CIA
004 = OPA
005 = NIA

The next instruction is immediately available to the computer as soon as the current instruction has been executed. Note that whenever the interval between CIA and NIA in either the normal or delayed mode of operation includes the regression of sector 177 to sector 000, the track address of NIA does not change. NIT is always equal to CIT except when the programmer purposely includes a jump instruction.

OPERAND MODE OPERATION

Instructions which require operands may be modified by tag bit 27. A one bit in bit position 27 uses the data in the operand address field as the operand itself rather than the address of the operand. For example, the instruction "Add, operand mode" appears as

BIN	0	100	110	000	XX	0	000	000	000	000	011
OCT	0	4	6	0		0	0	0	0	0	3

This instruction adds 000003 to the contents of the A register. It does not add the contents of track 000 sector 003.

Five instructions ~~which~~ can be executed in the ~~operand~~ mode (Load A, Load B, Load D, Merge, and Extract) ~~can be~~ tagg~~ed~~ with bit 26 of the operation code. This permits the 14 least-significant bits of the instruction word to operate on the 14 most-significant bits of the affected registers.

~~The third most significant octal digit of the operation code of operand instructions must be a 4, 5, 6, or 7.~~

The largest integer that may be used as an operand is $65,035_{10}$, or 177777_8.

Operand instructions cannot be indexed, nor is parity checked.

EXECUTION CODE TAG

Tag bits 25 and 26 are used as modifiers in multiply, divide, scan, shift, store A repeat, and digital commands, and are referred to as execution code bits. Refer to these commands for their specific applications.

ADDRESSING B, C, AND D REGISTERS

Any command which directly addresses a memory location may also address the B, C, and D registers. The track addresses 075, 076, and 077 are reserved for this purpose. Thus it is possible to use the contents of these three registers as operands. Track addresses 075, 076, and 077 address the B, C, and D registers respectively.

A few examples of the use of these registers are presented below indicating the contents of the pertinent registers before and after the instruction is carried out.

OP CODE	TTT-SSS	BEFORE	AFTER
LOAD A	075-XXX	A = 0770777777 B = 0013000000	A = 0013000000 B = 0013000000
LOAD B	076-072	B = 0770007777 C = 100032	B = 0000100032 C = 100032
ADD	076-XXX	A = 0770707777 C = 100100	A = 0771010077 C = 100100
LOAD X	076-XXX	X = 177 C = 100100	X = 100 C = 100100
REDUCE I	077-XXX	I = 100077 D = 0333000007	I = 100070 D = 0333000007
STORE A	076-XXX	A = 0333366666 C = 005701	A = 0333366666 C = 166666
MERGE	076-XXX	A = 1776000000 C = 000123	A = 1776000123 C = 000123

Note that when loading full length registers from shorter length registers, the remaining bits of the register being loaded are cleared.

The use of a delayed mode instruction, when addressing the B, C, or D register, permits the programmer to arbitrarily determine the location of the next instruction. The location of the next instruction is one greater than the operand sector address. For example, in the Load B instruction above, the next instruction sector address will be 073, (072 + 001 = 073).

SECTION 6

COMPUTER CONTROL AND MAINTENANCE PANEL

GENERAL

Operating controls and neon indicator lamps are mounted on the TRW-330 computer control panel. These controls and indicators are used primarily by the programmer for entering and debugging programs, and by the maintenance engineer for periodic maintenance tests.

TRW-330 computer control switches are used as an aid to the programmer in loading programs into the computer, and for checking programs for errors.

RUN-STOP SWITCH

The RUN-STOP switch is located at the bottom of the computer control panel. During normal computer operation this switch must be in the "RUN" position. When the switch is thrown to the "STOP" position, the computer completes the present instruction and stops its operation before locating the next instruction. When the computer is stopped, the programmer may inspect the contents of the registers and of the control flip-flops. The operation of the RUN-STOP switch is linked with the RESUME-STEP button operation.

RESUME-STEP BUTTON

The RESUME-STEP button is used whenever the computer has stopped. Depressing the RESUME-STEP switch causes the computer to locate and perform the next instruction.

JUN RUN-STOP SWITCH

During normal computer operation, the JUN RUN-STOP switch must be in the JUN RUN position. When the switch is thrown to the "STOP" position, the computer stops whenever it encounters an unconditional jump instruction. Pushing the RESUME-STEP button will then permit the computer to continue its program until the next unconditional jump is encountered. The programmer will find the use of this switch a convenience in debugging large areas of control programs.

BREAKPOINT SWITCHES

The 18 toggle switches located in the center of the computer control panel are breakpoint switches. Each switch represents a binary one or zero. A switch in the "up" position represents a binary one. A switch in the "down" position represents a binary zero. Thus, any number from zero to $2^{18}-1$ may be represented by the breakpoint switches.

The switches are grouped into threes to allow the switches to be set or read in octal notation conveniently. They are also grouped into track and sector fields when addresses are to be formed by the breakpoint switches.

The contents of the breakpoint switches can be entered into the 18 least significant bits of the A register with a digital-input instruction having a track address of 001.

START SWITCH

When the START button is depressed, the computer is immediately forced to take its next instruction from track 000, sector 000. This location is called the "origin". The permanently recorded jump instruction contained in the origin takes the computer into a short routine to inspect the contents of the breakpoint switches. The configuration of the breakpoint switches then determines the ultimate destination of the computer, whether it is to the load or verify routines, to one of the several utility routines, to the executive program, etc.

INTERRUPT BUTTON

The INTERRUPT button is an alternative interrupt source for test and maintenance purposes, or for program checkout.

INDICATORS

TRW-330 indicators are neon lamps located in the control panel and indicate the binary contents of the registers, or the on or off conditions of control flip-flops.

FLIP-FLOPS REGISTERS SWITCH

The FLIP-FLOPS REGISTERS switch is used to select the source of the information to be displayed by the 28 neon indicators at the top of the computer control panel.

When the switch is in the REGISTERS position, the binary contents of the Y, N, A, B, C, D, I, or X registers may be determined by throwing the REGISTERS switch to the corresponding register position. Binary ones are indicated by the corresponding neon lamp being on. Binary zeros are indicated by the corresponding neon lamp being off.

When the switch is in the FLIP-FLOPS position, each neon indicator corresponds to the state of the flip-flop with which it is associated. Not all indicators are of interest to the programmer, however. The STATE flip-flop indicators (FS1, FS2, and FS3) will be set to State 1 when the computer is stopped. State 1 is indicated by FS3 and FS2 neons off, and FS1 neon on.

The TRACK ADDRESS neons, FT1 through FT10, represent the Track Address of the next instruction when the computer is stopped in State 1. The sector address of the next instruction can be read in the first seven neon indicators marked Sector Address.

The operation code is indicated by the nine neon lamps marked Operation Code. In the STOP condition of State 1, the operation code will be that of the previous instruction.

The FOF neon lamp indicates whether or not an overflow has occurred in the A register, and FP2 neon lamp indicates a parity error has occurred.

The MASTER neon indicator is off during normal computer operation. It is on only when loading programs into those tracks which are otherwise protected.

The WRITE indicators 1, 2, 3, and 4, when lighted, indicate those track groups which are being protected.

SECTION 7

PROGRAMMING PARTICULARS

GENERAL

The general procedure when writing a program for a digital computer such as the TRW-330 is to consider the program as consisting of three parts: the statement of the problem, the flowgram and the program listing.

The statement of the problem should detail briefly the problem to be solved or the function to be performed.

The flowgram is a simple outline illustrating the functional steps the computer will follow in solving the problem. The flowgram may be general or detailed, but should provide enough of a transition between the statement of the problem and the actual listing of instructions to make clear the relationship of the one to the other.

A listing of the instructions in detail is made from the flowgram. The listing must be written in a format which is acceptable to the computer, and is usually written in a sequential manner, one instruction following the next in the same way the computer will perform the instructions.

The sample flow conversion program that follows demonstrates the procedure in writing a program as outlined above.

SAMPLE FLOW CONVERSION PROGRAM

STATEMENT OF PROBLEM

Solve flow calculation $F = P + R\sqrt{E\,\frac{S}{T}}$

where: $E = 100\text{-}1000$ at 2^{-27}

$S = 2\text{-}10$ at 2^{-4}

$T = \Delta 00\text{-}700$ at 2^{-12}

$R = 1000$ at 2^{-15}

$P = -100$ at 2^{-18}

FLOWGRAM

LAN	S → Req.
M14N	SE

D27N	$\frac{SE}{T}$
SQ	$\sqrt{\frac{SE}{T}}$
M27N	$R\sqrt{\frac{SE}{T}}$
ADN	$P + R\sqrt{\frac{SE}{T}}$

LISTING

LOCATION	OP CODE	OPERAND	REMARKS
030-000	LA-N 0070	030-001	S at $2^{-4} \rightarrow$(A)
030-002	M14-N 0102	020-003	(S)(E) at 2^{-12} = SE at $2^{-18} \rightarrow$(A)
030-023	D27-N 0312	030-024	(A)/T at 2^{-12} = $(\frac{SE}{T})$ at 2^{-6}
030-062	SA-N 0076	077-000	$\rightarrow$(D) at 2^{6}
030-064	SQRT-N 0006	077-000	$\sqrt{D} \rightarrow$(A) at 2^{-3}
030-104	M15-N 0302	030-105	(R at 2^{-15}) [(A)] $\rightarrow$(A) at 2^{-18}
030-142	AD-N 0060	030-143	P at 2^{-18} + $R\sqrt{SE}$/T at 2^{-18}
030-144	SA-N 0076	030-145	$\rightarrow$F at 2^{-18}
030-001	D + 2.0 - 04		S at 2^{-4} = 2.0
030-024	D + 400.0 - 12		T at 2^{-12} = 400.0
030-105	D + 1000 - 15		R at 2^{-15} = 1000.0
030-143	D - 100.0 - 18		P at 2^{-18} = -100.0

TRW-330 MACHINE LANGUAGE LOADER/VERIFIER

INTRODUCTION

The TRW-330 Machine Language Loader/Verifier is a permanently stored routine designed to perform the following functions:

LOAD a LISTABLE TRW-330 tape containing instructions and constants arranged in the format illustrated.

LOAD a BINARY tape containing 1 to 511 tracks of information.

VERIFY the information punched in either a Listable or Binary tape with the information stored on the drum. When two words do not agree, the memory word and its address are printed. Verifying does not stop when errors are detected.

Generate a CHECK SUM for either the Listable or Binary tape as it is loaded and compare this sum check against ones punched at the end of the Listable tape program or at end of each track on a Binary tape. Loading stops if the two check figures disagree.

NOTE: Whenever loading STOPS, whether it is to indicate check sum failure or simply a normal stop loading, the generated check sum is transferred into the A-Register before the STOP occurs.

The UNIVERSAL BREAKPOINT feature of the Loader/Verifier provides the programmer with easy access to any track and sector on the drum simply by setting the console toggle switches and pressing START.

TAPE FORMATS

Listed below are the tape formats acceptable to the Loader/Verifier.

Instruction Formats

(a) Normal or Delayed Mode

$$ATTTT\text{-}SSS\begin{matrix}T\\B\end{matrix}XXXX\begin{matrix}T\\B\end{matrix}TTTT\text{-}SSS\ {}^{C}/_{R}$$

(1) (2) (3)

Example:

A0005-004 0070 0041-037 Acceptable

A5-004 0070 41-037 Preferred

Note: Leading zeros in track address can be suppressed.

(b) Operand Mode = Octal Operand

$$ATTTT\text{-}SSS\begin{matrix}T\\B\end{matrix}XXXX\begin{matrix}T\\B\end{matrix}NNNNNN\ {}^{C}/_{R}$$

(1) (2) (4)

Example:

A132-176 0424 205777 $^{C}/_{R}$

(c) Operand Mode: Decimal Operand

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}$XXXX $\begin{matrix}T\\A\\B\end{matrix}$DIIIII. F -YY $^{C}/_{R}$

(1) (2) (5) (6)

Examples:

A177-124 0470 D. 95-16

A3-000 0424 D389. 75-25

A41-136 0420 D65535-27

(d) Upper Fill Mode

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}$XXXX$\begin{matrix}T\\A\\B\end{matrix}$ONNNNN $^{C}/_{R}$ Octal Operand

(1) (2) (7)

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}$XXXX$\begin{matrix}T\\A\\B\end{matrix}$DIIIII. F -YY $^{C}/_{R}$ Decimal Operand

(1) (2) (8)

Octal Constants

(a) Positive Number

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}$ONNNNNNNNN $^{C}/_{R}$

(1)

Example:

$(+147)_8$ Scaled 2^{-27} = 0000000147

A50-37 000147

Note: Up to four leading zeros can be suppressed. Octal constants must contain at least 6 digits.

(b) Negative Number

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}$-0NNNNNNNNN $^{C}/_{R}$

(1)

Example: $(-147)_8$

A50-37 -0000000147

Note: This format permits listing a negative number as a minus positive number. The loader converts it to the 2's complement and stores the number.

(c) Negative Number 2's Complement Form

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}$1NNNNNNNNN $^{C}/_{R}$

(1)

Example:

A50-37 1762000727

Decimal Constants

(a) Integers

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}D\pm$NNNNNNNNN. -YY $^{C}/_{R}$

(1) (9) (10)

Example:

A13-124 D+57983. -27

A53-000 D+0. -00

(b) Fractions

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}D\pm$. FFFFFFFFF $\pm$YY $^{C}/_{R}$

(1) (11) (12)

(c) Mixed Numbers

ATTTT-SSS$\begin{matrix}T\\A\\B\end{matrix}D\pm$IIIIII. FFFF-YY $^{C}/_{R}$

(1) (13) (10)

(d) Flex Constant

$$ATTTT\text{-}SSS\begin{matrix}T\\A\\B\end{matrix}FX_1X_2X_3X_4\,{}^{C}/_{R}$$

(1) (14) (15)

Example:

A7-007 F132A Mixed Alphanumeric

A65-177 FATSC All Alphabetic

A32-000 F1234 All Numeric

SYMBOLS USED

(1)	ATTTT-SSS	= $(\text{Location Address})_8$
(2)	XXXX	= $(\text{Operation Code})_8$
(3)	TTTT-SSS	= $(\text{Operand Address})_8$
(4)	NNNNNN	= $(\text{Operand})_8$
	Max. NNNNNN	= $(377777)_8$
(5)	DIIIII. F	= $(\text{Operand})_{10}$
	Max. IIIII. F	= $(65535)_{10}$
(6)	-YY	= $(\text{Operand Scale Factor})_{10}$
(7)	Max. ONNNNN	= $(037777)_8$
(8)	Max. IIIII. F	= $(16383)_{10}$
(9)	D±NNNNNNNNN.	= Decimal Integer
		Maximum Integer ±(268,435,455)
(10)	-YY	= $(\text{Scale Factor})_{10}$
		YY Range: 00 to 27
(11)	D±. FFFFFFF	= Decimal Fraction

NOTE: Minimum Fraction .00000001

(12) ±YY = $(\text{Scale Factor})_{10}$

(13) Positive Scale Factors are valid. However, the shift left occurs after number is converted. Therefore, the fraction 0.00000001 still represents the smallest fractional number which can be loaded.

(14) D±IIIII. FFFF = Signed Mixed Decimal Number

(15) F = Flex Constant Flag

(16) $X_1X_2X_3X_4$ = Any four flex codes

LOADING THE TAPE

Either Listable or Binary punched tape information can be loaded into the TRW-330 computer by carrying out the four following steps:

(1) Insert the punched tape into the Flexowriter reader.
(2) Turn the Flexowriter power on.
(3) Set the toggle switches on the control panel to 000000.
(4) Push the START button.

VERIFYING THE TAPE

After the tape has been loaded, the information entered into memory can be verified by repeating the steps used when loading the tape except that the toggle switches must be set to 000001. If the information in a word in memory does not compare with the information of the corresponding word on the punched tape, the address of the memory word and its contents are printed.

USING THE UNIVERSAL BREAKPOINT

The operator can force the computer to transfer to any sector of any track by setting the toggle switches to the desired address less 10_8, and then pushing the START button. For example, to transfer to Track 101 Sector 040, set the toggles to 101-030 and push the START button. The computer will take its next instruction from 101-040. The subtraction of 10_8 must be carried out modulo $>177_8$. Thus, to transfer to 120-000, the toggles must be set to 117-170.

TRW-330 UTILITY PACKAGE

The Utility Package is a set of general utility routines designed to aid the TRW-330 user in program checkout. The basic package consists of eight programs which can be selected and controlled from the TRW-330 Operator's Control Panel by settings of the 18 toggle switches. Each of the Utility Routines is described briefly below. For a detailed write-up on the Utility Package refer to TRW-330 Program Library routine number 0002.

(1) READ TOGGLE ADDRESS: transfers the contents of the memory address specified by the settings of the toggles switches, T_{17}-T1, to the D register. The memory address appears in the A register.

(2) STORE TOGGLE WORD: allows the operator to enter and store one instruction or constant in any selected memory location through toggle switch settings.

(3) BINARY DUMP: punches a tape in binary format of information from 1 to 511 consecutively addressed tracks. The tape so produced is acceptable to the TRW-330 Loader/Verifier.

(4) CHECK SUM: calculates and prints the octal sum of the contents of each of the tracks specified by the toggle switch settings. Each checksum is preceded by its corresponding track number.

(5) MEMORY WRITE: stores a toggle selected constant into any block of consecutively addressed sectors.

(6) OCTAL DUMP: prints the contents of the toggle selected memory locations as ten digit octal numbers.

(7) INSTRUCTION DUMP: prints in instruction format the contents of any consecutive group of memory track locations from a partial track up to 511 complete tracks.

(8) DECIMAL DUMP: prints as decimal numbers the contents of any consecutive group of memory locations from a single word up to 511 complete tracks. Each word is output as a signed mixed number scaled according to the positive or negative scale factor assigned. Print out of leading zeros is suppressed.

SUMMARY OF BASIC UTILITY ROUTINE ENTRIES

The operator can enter any of the above routines by making the appropriate toggle setting, then pressing START. In all cases the computer will HALT and the operator then will make several additional toggle switch settings to

define the various program parameters. The following table summarizes all the toggle settings necessary to use the Basic Utility Routines. The individual programs automatically select the necessary Flex punch/print conditions.

Toggle Setting / Routine Settings	(Entry Setting)$_8$ $T_{18} - T_1$	(Scale Factor)$_8$ $T_7 - T_1$	(1st Address)$_8$ $T_{18} - T_1$	(Last Address)$_8$ $T_{18} - T_1$	Instr/Constant $T_{10} - T_1$	Instr/Constant $T_{18} - T_1$
Read Toggle Address	000005	---	TTTT-SSS	---	---	---
Store Toggle Word	000007	---	TTTT-SSS	---	Op code, or 4 MSD	OPA, or 6 LSD
Binary Dump	000377	---	TTTT-000	TTTT-000	---	---
Check Sum	000200	---	TTTT-000	TTTT-000	---	---
Memory Write	000201	---	TTTT-SSS	TTTT-SSS	Op code, or 4 MSD	OPA, or 6 MSD
Octal Dump	000400	---	TTTT-SSS	TTTT-SSS	---	---
Instruction Dump	000401	---	TTTT-SSS	TTTT-SSS	---	---
Decimal Dump	000600	± XX	TTTT-SSS	TTTT-SSS	---	---

NOTE: To use this table, make each toggle setting indicated in the order listed. Columns showing dashes are to be ignored.

Example:

To print out the contents, in decimal format, of the 10_9 consecutive words located from 125-000 to 125-011, each word scaled 2^{-9}, the following steps are required:

(1) Set toggles to 000600
(2) Push START button
(3) Set toggles to 000111 (2^{-9})
(4) Push RESUME button
(5) Set toggles to 125-000
(6) Push RESUME button
(7) Set toggles to 125-011
(8) Push RESUME button

Note that bits 7-1 read, in binary, 1001001. Bit 7 designates "negative"; bits 6-1 designate the decimal number "9".

FAST ACCESS TRACK CORRESPONDENCE CHART

TRACKS				
	060	061	062	063
	061	062	063	060
	062	063	060	061
	063	060	061	062
SECTORS	000	040	100	140
	001	041	101	141
	002	042	102	142
	003	043	103	143
	004	044	104	144
	005	045	105	145
	006	046	106	146
	007	047	107	147
	010	050	110	150
	011	051	111	151
	012	052	112	152
	013	053	113	153
	014	054	114	154
	015	055	115	155
	016	056	116	156
	017	057	117	157
	020	060	120	160
	021	061	121	161
	022	062	122	162
	023	063	123	163
	024	064	124	164
	025	065	125	165
	026	066	126	166
	027	067	127	167
	030	070	130	170
	031	071	131	171
	032	072	132	172
	033	073	133	173
	034	074	134	174
	035	075	135	175
	036	076	136	176
	037	077	137	177

Card is Letter

Table of Powers of 2

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

Octal-Decimal Integer Conversion Table

0000 (Octal)	0000 (Decimal)
to	to
0777 (Octal)	0511 (Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
0000	0000	0001	0002	0003	0004	0005	0006	0007
0010	0008	0009	0010	0011	0012	0013	0014	0015
0020	0016	0017	0018	0019	0020	0021	0022	0023
0030	0024	0025	0026	0027	0028	0029	0030	0031
0040	0032	0033	0034	0035	0036	0037	0038	0039
0050	0040	0041	0042	0043	0044	0045	0046	0047
0060	0048	0049	0050	0051	0052	0053	0054	0055
0070	0056	0057	0058	0059	0060	0061	0062	0063
0100	0064	0065	0066	0067	0068	0069	0070	0071
0110	0072	0073	0074	0075	0076	0077	0078	0079
0120	0080	0081	0082	0083	0084	0085	0086	0087
0130	0088	0089	0090	0091	0092	0093	0094	0095
0140	0096	0097	0098	0099	0100	0101	0102	0103
0150	0104	0105	0106	0107	0108	0109	0110	0111
0160	0112	0113	0114	0115	0116	0117	0118	0119
0170	0120	0121	0122	0123	0124	0125	0126	0127
0200	0128	0129	0130	0131	0132	0133	0134	0135
0210	0136	0137	0138	0139	0140	0141	0142	0143
0220	0144	0145	0146	0147	0148	0149	0150	0151
0230	0152	0153	0154	0155	0156	0157	0158	0159
0240	0160	0161	0162	0163	0164	0165	0166	0167
0250	0168	0169	0170	0171	0172	0173	0174	0175
0260	0176	0177	0178	0179	0180	0181	0182	0183
0270	0184	0185	0186	0187	0188	0189	0190	0191
0300	0192	0193	0194	0195	0196	0197	0198	0199
0310	0200	0201	0202	0203	0204	0205	0206	0207
0320	0208	0209	0210	0211	0212	0213	0214	0215
0330	0216	0217	0218	0219	0220	0221	0222	0223
0340	0224	0225	0226	0227	0228	0229	0230	0231
0350	0232	0233	0234	0235	0236	0237	0238	0239
0360	0240	0241	0242	0243	0244	0245	0246	0247
0370	0248	0249	0250	0251	0252	0253	0254	0255

	0	1	2	3	4	5	6	7
0400	0256	0257	0258	0259	0260	0261	0262	0263
0410	0264	0265	0266	0267	0268	0269	0270	0271
0420	0272	0273	0274	0275	0276	0277	0278	0279
0430	0280	0281	0282	0283	0284	0285	0286	0287
0440	0288	0289	0290	0291	0292	0293	0294	0295
0450	0296	0297	0298	0299	0300	0301	0302	0303
0460	0304	0305	0306	0307	0308	0309	0310	0311
0470	0312	0313	0314	0315	0316	0317	0318	0319
0500	0320	0321	0322	0323	0324	0325	0326	0327
0510	0328	0329	0330	0331	0332	0333	0334	0335
0520	0336	0337	0338	0339	0340	0341	0342	0343
0530	0344	0345	0346	0347	0348	0349	0350	0351
0540	0352	0353	0354	0355	0356	0357	0358	0359
0550	0360	0361	0362	0363	0364	0365	0366	0367
0560	0368	0369	0370	0371	0372	0373	0374	0375
0570	0376	0377	0378	0379	0380	0381	0382	0383
0600	0384	0385	0386	0387	0388	0389	0390	0391
0610	0392	0393	0394	0395	0396	0397	0398	0399
0620	0400	0401	0402	0403	0404	0405	0406	0407
0630	0408	0409	0410	0411	0412	0413	0414	0415
0640	0416	0417	0418	0419	0420	0421	0422	0423
0650	0424	0425	0426	0427	0428	0429	0430	0431
0660	0432	0433	0434	0435	0436	0437	0438	0439
0670	0440	0441	0442	0443	0444	0445	0446	0447
0700	0448	0449	0450	0451	0452	0453	0454	0455
0710	0456	0457	0458	0459	0460	0461	0462	0463
0720	0464	0465	0466	0467	0468	0469	0470	0471
0730	0472	0473	0474	0475	0476	0477	0478	0479
0740	0480	0481	0482	0483	0484	0485	0486	0487
0750	0488	0489	0490	0491	0492	0493	0494	0495
0760	0496	0497	0498	0499	0500	0501	0502	0503
0770	0504	0505	0506	0507	0508	0509	0510	0511

1000 (Octal)	0512 (Decimal)
to	to
1777 (Octal)	1023 (Decimal)

	0	1	2	3	4	5	6	7
1000	0512	0513	0514	0515	0516	0517	0518	0519
1010	0520	0521	0522	0523	0524	0525	0526	0527
1020	0528	0529	0530	0531	0532	0533	0534	0535
1030	0536	0537	0538	0539	0540	0541	0542	0543
1040	0544	0545	0546	0547	0548	0549	0550	0551
1050	0552	0553	0554	0555	0556	0557	0558	0559
1060	0560	0561	0562	0563	0564	0565	0566	0567
1070	0568	0569	0570	0571	0572	0573	0574	0575
1100	0576	0577	0578	0579	0580	0581	0582	0583
1110	0584	0585	0586	0587	0588	0589	0590	0591
1120	0592	0593	0594	0595	0596	0597	0598	0599
1130	0600	0601	0602	0603	0604	0605	0606	0607
1140	0608	0609	0610	0611	0612	0613	0614	0615
1150	0616	0617	0618	0619	0620	0621	0622	0623
1160	0624	0625	0626	0627	0628	0629	0630	0631
1170	0632	0633	0634	0635	0636	0637	0638	0639
1200	0640	0641	0642	0643	0644	0645	0646	0647
1210	0648	0649	0650	0651	0652	0653	0654	0655
1220	0656	0657	0658	0659	0660	0661	0662	0663
1230	0664	0665	0666	0667	0668	0669	0670	0671
1240	0672	0673	0674	0675	0676	0677	0678	0679
1250	0680	0681	0682	0683	0684	0685	0686	0687
1260	0688	0689	0690	0691	0692	0693	0694	0695
1270	0696	0697	0698	0699	0700	0701	0702	0703
1300	0704	0705	0706	0707	0708	0709	0710	0711
1310	0712	0713	0714	0715	0716	0717	0718	0719
1320	0720	0721	0722	0723	0724	0725	0726	0727
1330	0728	0729	0730	0731	0732	0733	0734	0735
1340	0736	0737	0738	0739	0740	0741	0742	0743
1350	0744	0745	0746	0747	0748	0749	0750	0751
1360	0752	0753	0754	0755	0756	0757	0758	0759
1370	0760	0761	0762	0763	0764	0765	0766	0767

	0	1	2	3	4	5	6	7
1400	0768	0769	0770	0771	0772	0773	0774	0775
1410	0776	0777	0778	0779	0780	0781	0782	0783
1420	0784	0785	0786	0787	0788	0789	0790	0791
1430	0792	0793	0794	0795	0796	0797	0798	0799
1440	0800	0801	0802	0803	0804	0805	0806	0807
1450	0808	0809	0810	0811	0812	0813	0814	0815
1460	0816	0817	0818	0819	0820	0821	0822	0823
1470	0824	0825	0826	0827	0828	0829	0830	0831
1500	0832	0833	0834	0835	0836	0837	0838	0839
1510	0840	0841	0842	0843	0844	0845	0846	0847
1520	0848	0849	0850	0851	0852	0853	0854	0855
1530	0856	0857	0858	0859	0860	0861	0862	0863
1540	0864	0865	0866	0867	0868	0869	0870	0871
1550	0872	0873	0874	0875	0876	0877	0878	0879
1560	0880	0881	0882	0883	0884	0885	0886	0887
1570	0888	0889	0890	0891	0892	0893	0894	0895
1600	0896	0897	0898	0899	0900	0901	0902	0903
1610	0904	0905	0906	0907	0908	0909	0910	0911
1620	0912	0913	0914	0915	0916	0917	0918	0919
1630	0920	0921	0922	0923	0924	0925	0926	0927
1640	0928	0929	0930	0931	0932	0933	0934	0935
1650	0936	0937	0938	0939	0940	0941	0942	0943
1660	0944	0945	0946	0947	0948	0949	0950	0951
1670	0952	0953	0954	0955	0956	0957	0958	0959
1700	0960	0961	0962	0963	0964	0965	0966	0967
1710	0968	0969	0970	0971	0972	0973	0974	0975
1720	0976	0977	0978	0979	0980	0981	0982	0983
1730	0984	0985	0986	0987	0988	0989	0990	0991
1740	0992	0993	0994	0995	0996	0997	0998	0999
1750	1000	1001	1002	1003	1004	1005	1006	1007
1760	1008	1009	1010	1011	1012	1013	1014	1015
1770	1016	1017	1018	1019	1020	1021	1022	1023

Octal-Decimal Integer Conversion Table

	0	1	2	3	4	5	6	7
2000	1024	1025	1026	1027	1028	1029	1030	1031
2010	1032	1033	1034	1035	1036	1037	1038	1039
2020	1040	1041	1042	1043	1044	1045	1046	1047
2030	1048	1049	1050	1051	1052	1053	1054	1055
2040	1056	1057	1058	1059	1060	1061	1062	1063
2050	1064	1065	1066	1067	1068	1069	1070	1071
2060	1072	1073	1074	1075	1076	1077	1078	1079
2070	1080	1081	1082	1083	1084	1085	1086	1087
2100	1088	1089	1090	1091	1092	1093	1094	1095
2110	1096	1097	1098	1099	1100	1101	1102	1103
2120	1104	1105	1106	1107	1108	1109	1110	1111
2130	1112	1113	1114	1115	1116	1117	1118	1119
2140	1120	1121	1122	1123	1124	1125	1126	1127
2150	1128	1129	1130	1131	1132	1133	1134	1135
2160	1136	1137	1138	1139	1140	1141	1142	1143
2170	1144	1145	1146	1147	1148	1149	1150	1151
2200	1152	1153	1154	1155	1156	1157	1158	1159
2210	1160	1161	1162	1163	1164	1165	1166	1167
2220	1168	1169	1170	1171	1172	1173	1174	1175
2230	1176	1177	1178	1179	1180	1181	1182	1183
2240	1184	1185	1186	1187	1188	1189	1190	1191
2250	1192	1193	1194	1195	1196	1197	1198	1199
2260	1200	1201	1202	1203	1204	1205	1206	1207
2270	1208	1209	1210	1211	1212	1213	1214	1215
2300	1216	1217	1218	1219	1220	1221	1222	1223
2310	1224	1225	1226	1227	1228	1229	1230	1231
2320	1232	1233	1234	1235	1236	1237	1238	1239
2330	1240	1241	1242	1243	1244	1245	1246	1247
2340	1248	1249	1250	1251	1252	1253	1254	1255
2350	1256	1257	1258	1259	1260	1261	1262	1263
2360	1264	1265	1266	1267	1268	1269	1270	1271
2370	1272	1273	1274	1275	1276	1277	1278	1279

	0	1	2	3	4	5	6	7
2400	1280	1281	1282	1283	1284	1285	1286	1287
2410	1288	1289	1290	1291	1292	1293	1294	1295
2420	1296	1297	1298	1299	1300	1301	1302	1303
2430	1304	1305	1306	1307	1308	1309	1310	1311
2440	1312	1313	1314	1315	1316	1317	1318	1319
2450	1320	1321	1322	1323	1324	1325	1326	1327
2460	1328	1329	1330	1331	1332	1333	1334	1335
2470	1336	1337	1338	1339	1340	1341	1342	1343
2500	1344	1345	1346	1347	1348	1349	1350	1351
2510	1352	1353	1354	1355	1356	1357	1358	1359
2520	1360	1361	1362	1363	1364	1365	1366	1367
2530	1368	1369	1370	1371	1372	1373	1374	1375
2540	1376	1377	1378	1379	1380	1381	1382	1383
2550	1384	1385	1386	1387	1388	1389	1390	1391
2560	1392	1393	1394	1395	1396	1397	1398	1399
2570	1400	1401	1402	1403	1404	1405	1406	1407
2600	1408	1409	1410	1411	1412	1413	1414	1415
2610	1416	1417	1418	1419	1420	1421	1422	1423
2620	1424	1425	1426	1427	1428	1429	1430	1431
2630	1432	1433	1434	1435	1436	1437	1438	1439
2640	1440	1441	1442	1443	1444	1445	1446	1447
2650	1448	1449	1450	1451	1452	1453	1454	1455
2660	1456	1457	1458	1459	1460	1461	1462	1463
2670	1464	1465	1466	1467	1468	1469	1470	1471
2700	1472	1473	1474	1475	1476	1477	1478	1479
2710	1480	1481	1482	1483	1484	1485	1486	1487
2720	1488	1489	1490	1491	1492	1493	1494	1495
2730	1496	1497	1498	1499	1500	1501	1502	1503
2740	1504	1505	1506	1507	1508	1509	1510	1511
2750	1512	1513	1514	1515	1516	1517	1518	1519
2760	1520	1521	1522	1523	1524	1525	1526	1527
2770	1528	1529	1530	1531	1532	1533	1534	1535

Octal	Decimal
2000 to 2777	1024 to 1535

(Octal) (Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
3000	1536	1537	1538	1539	1540	1541	1542	1543
3010	1544	1545	1546	1547	1548	1549	1550	1551
3020	1552	1553	1554	1555	1556	1557	1558	1559
3030	1560	1561	1562	1563	1564	1565	1566	1567
3040	1568	1569	1570	1571	1572	1573	1574	1575
3050	1576	1577	1578	1579	1580	1581	1582	1583
3060	1584	1585	1586	1587	1588	1589	1590	1591
3070	1592	1593	1594	1595	1596	1597	1598	1599
3100	1600	1601	1602	1603	1604	1605	1606	1607
3110	1608	1609	1610	1611	1612	1613	1614	1615
3120	1616	1617	1618	1619	1620	1621	1622	1623
3130	1624	1625	1626	1627	1628	1629	1630	1631
3140	1632	1633	1634	1635	1636	1637	1638	1639
3150	1640	1641	1642	1643	1644	1645	1646	1647
3160	1648	1649	1650	1651	1652	1653	1654	1655
3170	1656	1657	1658	1659	1660	1661	1662	1663
3200	1664	1665	1666	1667	1668	1669	1670	1671
3210	1672	1673	1674	1675	1676	1677	1678	1679
3220	1680	1681	1682	1683	1684	1685	1686	1687
3230	1688	1689	1690	1691	1692	1693	1694	1695
3240	1696	1697	1698	1699	1700	1701	1702	1703
3250	1704	1705	1706	1707	1708	1709	1710	1711
3260	1712	1713	1714	1715	1716	1717	1718	1719
3270	1720	1721	1722	1723	1724	1725	1726	1727
3300	1728	1729	1730	1731	1732	1733	1734	1735
3310	1736	1737	1738	1739	1740	1741	1742	1743
3320	1744	1745	1746	1747	1748	1749	1750	1751
3330	1752	1753	1754	1755	1756	1757	1758	1759
3340	1760	1761	1762	1763	1764	1765	1766	1767
3350	1768	1769	1770	1771	1772	1773	1774	1775
3360	1776	1777	1778	1779	1780	1781	1782	1783
3370	1784	1785	1786	1787	1788	1789	1790	1791

	0	1	2	3	4	5	6	7
3400	1792	1793	1794	1795	1796	1797	1798	1799
3410	1800	1801	1802	1803	1804	1805	1806	1807
3420	1808	1809	1810	1811	1812	1813	1814	1815
3430	1816	1817	1818	1819	1820	1821	1822	1823
3440	1824	1825	1826	1827	1828	1829	1830	1831
3450	1832	1833	1834	1835	1836	1837	1838	1839
3460	1840	1841	1842	1843	1844	1845	1846	1847
3470	1848	1849	1850	1851	1852	1853	1854	1855
3500	1856	1857	1858	1859	1860	1861	1862	1863
3510	1864	1865	1866	1867	1868	1869	1870	1871
3520	1872	1873	1874	1875	1876	1877	1878	1879
3530	1880	1881	1882	1883	1884	1885	1886	1887
3540	1888	1889	1890	1891	1892	1893	1894	1895
3550	1896	1897	1898	1899	1900	1901	1902	1903
3560	1904	1905	1906	1907	1908	1909	1910	1911
3570	1912	1913	1914	1915	1916	1917	1918	1919
3600	1920	1921	1922	1923	1924	1925	1926	1927
3610	1928	1929	1930	1931	1932	1933	1934	1935
3620	1936	1937	1938	1939	1940	1941	1942	1943
3630	1944	1945	1946	1947	1948	1949	1950	1951
3640	1952	1953	1954	1955	1956	1957	1958	1959
3650	1960	1961	1962	1963	1964	1965	1966	1967
3660	1968	1969	1970	1971	1972	1973	1974	1975
3670	1976	1977	1978	1979	1980	1981	1982	1983
3700	1984	1985	1986	1987	1988	1989	1990	1991
3710	1992	1993	1994	1995	1996	1997	1998	1999
3720	2000	2001	2002	2003	2004	2005	2006	2007
3730	2008	2009	2010	2011	2012	2013	2014	2015
3740	2016	2017	2018	2019	2020	2021	2022	2023
3750	2024	2025	2026	2027	2028	2029	2030	2031
3760	2032	2033	2034	2035	2036	2037	2038	2039
3770	2040	2041	2042	2043	2044	2045	2046	2047

Octal	Decimal
3000 to 3777	1536 to 2047

(Octal) (Decimal)

Octal-Decimal Integer Conversion Table

4000	2048
to	to
4777	2559
(Octal)	(Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
4000	2048	2049	2050	2051	2052	2053	2054	2055
4010	2056	2057	2058	2059	2060	2061	2062	2063
4020	2064	2065	2066	2067	2068	2069	2070	2071
4030	2072	2073	2074	2075	2076	2077	2078	2079
4040	2080	2081	2082	2083	2084	2085	2086	2087
4050	2088	2089	2090	2091	2092	2093	2094	2095
4060	2096	2097	2098	2099	2100	2101	2102	2103
4070	2104	2105	2106	2107	2108	2109	2110	2111
4100	2112	2113	2114	2115	2116	2117	2118	2119
4110	2120	2121	2122	2123	2124	2125	2126	2127
4120	2128	2129	2130	2131	2132	2133	2134	2135
4130	2136	2137	2138	2139	2140	2141	2142	2143
4140	2144	2145	2146	2147	2148	2149	2150	2151
4150	2152	2153	2154	2155	2156	2157	2158	2159
4160	2160	2161	2162	2163	2164	2165	2166	2167
4170	2168	2169	2170	2171	2172	2173	2174	2175
4200	2176	2177	2178	2179	2180	2181	2182	2183
4210	2184	2185	2186	2187	2188	2189	2190	2191
4220	2192	2193	2194	2195	2196	2197	2198	2199
4230	2200	2201	2202	2203	2204	2205	2206	2207
4240	2208	2209	2210	2211	2212	2213	2214	2215
4250	2216	2217	2218	2219	2220	2221	2222	2223
4260	2224	2225	2226	2227	2228	2229	2230	2231
4270	2232	2233	2234	2235	2236	2237	2238	2239
4300	2240	2241	2242	2243	2244	2245	2246	2247
4310	2248	2249	2250	2251	2252	2253	2254	2255
4320	2256	2257	2258	2259	2260	2261	2262	2263
4330	2264	2265	2266	2267	2268	2269	2270	2271
4340	2272	2273	2274	2275	2276	2277	2278	2279
4350	2280	2281	2282	2283	2284	2285	2286	2287
4360	2288	2289	2290	2291	2292	2293	2294	2295
4370	2296	2297	2298	2299	2300	2301	2302	2303

	0	1	2	3	4	5	6	7
4400	2304	2305	2306	2307	2308	2309	2310	2311
4410	2312	2313	2314	2315	2316	2317	2318	2319
4420	2320	2321	2322	2323	2324	2325	2326	2327
4430	2328	2329	2330	2331	2332	2333	2334	2335
4440	2336	2337	2338	2339	2340	2341	2342	2343
4450	2344	2345	2346	2347	2348	2349	2350	2351
4460	2352	2353	2354	2355	2356	2357	2358	2359
4470	2360	2361	2362	2363	2364	2365	2366	2367
4500	2368	2369	2370	2371	2372	2373	2374	2375
4510	2376	2377	2378	2379	2380	2381	2382	2383
4520	2384	2385	2386	2387	2388	2389	2390	2391
4530	2392	2393	2394	2395	2396	2397	2398	2399
4540	2400	2401	2402	2403	2404	2405	2406	2407
4550	2408	2409	2410	2411	2412	2413	2414	2415
4560	2416	2417	2418	2419	2420	2421	2422	2423
4570	2424	2425	2426	2427	2428	2429	2430	2431
4600	2432	2433	2434	2435	2436	2437	2438	2439
4610	2440	2441	2442	2443	2444	2445	2446	2447
4620	2448	2449	2450	2451	2452	2453	2454	2455
4630	2456	2457	2458	2459	2460	2461	2462	2463
4640	2464	2465	2466	2467	2468	2469	2470	2471
4650	2472	2473	2474	2475	2476	2477	2478	2479
4660	2480	2481	2482	2483	2484	2485	2486	2487
4670	2488	2489	2490	2491	2492	2493	2494	2495
4700	2496	2497	2498	2499	2500	2501	2502	2503
4710	2504	2505	2506	2507	2508	2509	2510	2511
4720	2512	2513	2514	2515	2516	2517	2518	2519
4730	2520	2521	2522	2523	2524	2525	2526	2527
4740	2528	2529	2530	2531	2532	2533	2534	2535
4750	2536	2537	2538	2539	2540	2541	2542	2543
4760	2544	2545	2546	2547	2548	2549	2550	2551
4770	2552	2553	2554	2555	2556	2557	2558	2559

5000	2560
to	to
5777	3071
(Octal)	(Decimal)

	0	1	2	3	4	5	6	7
5000	2560	2561	2562	2563	2564	2565	2566	2567
5010	2568	2569	2570	2571	2572	2573	2574	2575
5020	2576	2577	2578	2579	2580	2581	2582	2583
5030	2584	2585	2586	2587	2588	2589	2590	2591
5040	2592	2593	2594	2595	2596	2597	2598	2599
5050	2600	2601	2602	2603	2604	2605	2606	2607
5060	2608	2609	2610	2611	2612	2613	2614	2615
5070	2616	2617	2618	2619	2620	2621	2622	2623
5100	2624	2625	2626	2627	2628	2629	2630	2631
5110	2632	2633	2634	2635	2636	2637	2638	2639
5120	2640	2641	2642	2643	2644	2645	2646	2647
5130	2648	2649	2650	2651	2652	2653	2654	2655
5140	2656	2657	2658	2659	2660	2661	2662	2663
5150	2664	2665	2666	2667	2668	2669	2670	2671
5160	2672	2673	2674	2675	2676	2677	2678	2679
5170	2680	2681	2682	2683	2684	2685	2686	2687
5200	2688	2689	2690	2691	2692	2693	2694	2695
5210	2696	2697	2698	2699	2700	2701	2702	2703
5220	2704	2705	2706	2707	2708	2709	2710	2711
5230	2712	2713	2714	2715	2716	2717	2718	2719
5240	2720	2721	2722	2723	2724	2725	2726	2727
5250	2728	2729	2730	2731	2732	2733	2734	2735
5260	2736	2737	2738	2739	2740	2741	2742	2743
5270	2744	2745	2746	2747	2748	2749	2750	2751
5300	2752	2753	2754	2755	2756	2757	2758	2759
5310	2760	2761	2762	2763	2764	2765	2766	2767
5320	2768	2769	2770	2771	2772	2773	2774	2775
5330	2776	2777	2778	2779	2780	2781	2782	2783
5340	2784	2785	2786	2787	2788	2789	2790	2791
5350	2792	2793	2794	2795	2796	2797	2798	2799
5360	2800	2801	2802	2803	2804	2805	2806	2807
5370	2808	2809	2810	2811	2812	2813	2814	2815

	0	1	2	3	4	5	6	7
5400	2816	2817	2818	2819	2820	2821	2822	2823
5410	2824	2825	2826	2827	2828	2829	2830	2831
5420	2832	2833	2834	2835	2836	2837	2838	2839
5430	2840	2841	2842	2843	2844	2845	2846	2847
5440	2848	2849	2850	2851	2852	2853	2854	2855
5450	2856	2857	2858	2859	2860	2861	2862	2863
5460	2864	2865	2866	2867	2868	2869	2870	2871
5470	2872	2873	2874	2875	2876	2877	2878	2879
5500	2880	2881	2882	2883	2884	2885	2886	2887
5510	2888	2889	2890	2891	2892	2893	2894	2895
5520	2896	2897	2898	2899	2900	2901	2902	2903
5530	2904	2905	2906	2907	2908	2909	2910	2911
5540	2912	2913	2914	2915	2916	2917	2918	2919
5550	2920	2921	2922	2923	2924	2925	2926	2927
5560	2928	2929	2930	2931	2932	2933	2934	2935
5570	2936	2937	2938	2939	2940	2941	2942	2943
5600	2944	2945	2946	2947	2948	2949	2950	2951
5610	2952	2953	2954	2955	2956	2957	2958	2959
5620	2960	2961	2962	2963	2964	2965	2966	2967
5630	2968	2969	2970	2971	2972	2973	2974	2975
5640	2976	2977	2978	2979	2980	2981	2982	2983
5650	2984	2985	2986	2987	2988	2989	2990	2991
5660	2992	2993	2994	2995	2996	2997	2998	2999
5670	3000	3001	3002	3003	3004	3005	3006	3007
5700	3008	3009	3010	3011	3012	3013	3014	3015
5710	3016	3017	3018	3019	3020	3021	3022	3023
5720	3024	3025	3026	3027	3028	3029	3030	3031
5730	3032	3033	3034	3035	3036	3037	3038	3039
5740	3040	3041	3042	3043	3044	3045	3046	3047
5750	3048	3049	3050	3051	3052	3053	3054	3055
5760	3056	3057	3058	3059	3060	3061	3062	3063
5770	3064	3065	3066	3067	3068	3069	3070	3071

Octal-Decimal Integer Conversion Table

	0	1	2	3	4	5	6	7
6000	3072	3073	3074	3075	3076	3077	3078	3079
6010	3080	3081	3082	3083	3084	3085	3086	3087
6020	3088	3089	3090	3091	3092	3093	3094	3095
6030	3096	3097	3098	3099	3100	3101	3102	3103
6040	3104	3105	3106	3107	3108	3109	3110	3111
6050	3112	3113	3114	3115	3116	3117	3118	3119
6060	3120	3121	3122	3123	3124	3125	3126	3127
6070	3128	3129	3130	3131	3132	3133	3134	3135
6100	3136	3137	3138	3139	3140	3141	3142	3143
6110	3144	3145	3146	3147	3148	3149	3150	3151
6120	3152	3153	3154	3155	3156	3157	3158	3159
6130	3160	3161	3162	3163	3164	3165	3166	3167
6140	3168	3169	3170	3171	3172	3173	3174	3175
6150	3176	3177	3178	3179	3180	3181	3182	3183
6160	3184	3185	3186	3187	3188	3189	3190	3191
6170	3192	3193	3194	3195	3196	3197	3198	3199
6200	3200	3201	3202	3203	3204	3205	3206	3207
6210	3208	3209	3210	3211	3212	3213	3214	3215
6220	3216	3217	3218	3219	3220	3221	3222	3223
6230	3224	3225	3226	3227	3228	3229	3230	3231
6240	3232	3233	3234	3235	3236	3237	3238	3239
6250	3240	3241	3242	3243	3244	3245	3246	3247
6260	3248	3249	3250	3251	3252	3253	3254	3255
6270	3256	3257	3258	3259	3260	3261	3262	3263
6300	3264	3265	3266	3267	3268	3269	3270	3271
6310	3272	3273	3274	3275	3276	3277	3278	3279
6320	3280	3281	3282	3283	3284	3285	3286	3287
6330	3288	3289	3290	3291	3292	3293	3294	3295
6340	3296	3297	3298	3299	3300	3301	3302	3303
6350	3304	3305	3306	3307	3308	3309	3310	3311
6360	3312	3313	3314	3315	3316	3317	3318	3319
6370	3320	3321	3322	3323	3324	3325	3326	3327

	0	1	2	3	4	5	6	7
6400	3328	3329	3330	3331	3332	3333	3334	3335
6410	3336	3337	3338	3339	3340	3341	3342	3343
6420	3344	3345	3346	3347	3348	3349	3350	3351
6430	3352	3353	3354	3355	3356	3357	3358	3359
6440	3360	3361	3362	3363	3364	3365	3366	3367
6450	3368	3369	3370	3371	3372	3373	3374	3375
6460	3376	3377	3378	3379	3380	3381	3382	3383
6470	3384	3385	3386	3387	3388	3389	3390	3391
6500	3392	3393	3394	3395	3396	3397	3398	3399
6510	3400	3401	3402	3403	3404	3405	3406	3407
6520	3408	3409	3410	3411	3412	3413	3414	3415
6530	3416	3417	3418	3419	3420	3421	3422	3423
6540	3424	3425	3426	3427	3428	3429	3430	3431
6550	3432	3433	3434	3435	3436	3437	3438	3439
6560	3440	3441	3442	3443	3444	3445	3446	3447
6570	3448	3449	3450	3451	3452	3453	3454	3455
6600	3456	3457	3458	3459	3460	3461	3462	3463
6610	3464	3465	3466	3467	3468	3469	3470	3471
6620	3472	3473	3474	3475	3476	3477	3478	3479
6630	3480	3481	3482	3483	3484	3485	3486	3487
6640	3488	3489	3490	3491	3492	3493	3494	3495
6650	3496	3497	3498	3499	3500	3501	3502	3503
6660	3504	3505	3506	3507	3508	3509	3510	3511
6670	3512	3513	3514	3515	3516	3517	3518	3519
6700	3520	3521	3522	3523	3524	3525	3526	3527
6710	3528	3529	3530	3531	3532	3533	3534	3535
6720	3536	3537	3538	3539	3540	3541	3542	3543
6730	3544	3545	3546	3547	3548	3549	3550	3551
6740	3552	3553	3554	3555	3556	3557	3558	3559
6750	3560	3561	3562	3563	3564	3565	3566	3567
6760	3568	3569	3570	3571	3572	3573	3574	3575
6770	3576	3577	3578	3579	3580	3581	3582	3583

6000 to 6777 (Octal)	3072 to 3583 (Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
7000	3584	3585	3586	3587	3588	3589	3590	3591
7010	3592	3593	3594	3595	3596	3597	3598	3599
7020	3600	3601	3602	3603	3604	3605	3606	3607
7030	3608	3609	3610	3611	3612	3613	3614	3615
7040	3616	3617	3618	3619	3620	3621	3622	3623
7050	3624	3625	3626	3627	3628	3629	3630	3631
7060	3632	3633	3634	3635	3636	3637	3638	3639
7070	3640	3641	3642	3643	3644	3645	3646	3647
7100	3648	3649	3650	3651	3652	3653	3654	3655
7110	3656	3657	3658	3659	3660	3661	3662	3663
7120	3664	3665	3666	3667	3668	3669	3670	3671
7130	3672	3673	3674	3675	3676	3677	3678	3679
7140	3680	3681	3682	3683	3684	3685	3686	3687
7150	3688	3689	3690	3691	3692	3693	3694	3695
7160	3696	3697	3698	3699	3700	3701	3702	3703
7170	3704	3705	3706	3707	3708	3709	3710	3711
7200	3712	3713	3714	3715	3716	3717	3718	3719
7210	3720	3721	3722	3723	3724	3725	3726	3727
7220	3728	3729	3730	3731	3732	3733	3734	3735
7230	3736	3737	3738	3739	3740	3741	3742	3743
7240	3744	3745	3746	3747	3748	3749	3750	3751
7250	3752	3753	3754	3755	3756	3757	3758	3759
7260	3760	3761	3762	3763	3764	3765	3766	3767
7270	3768	3769	3770	3771	3772	3773	3774	3775
7300	3776	3777	3778	3779	3780	3781	3782	3783
7310	3784	3785	3786	3787	3788	3789	3790	3791
7320	3792	3793	3794	3795	3796	3797	3798	3799
7330	3800	3801	3802	3803	3804	3805	3806	3807
7340	3808	3809	3810	3811	3812	3813	3814	3815
7350	3816	3817	3818	3819	3820	3821	3822	3823
7360	3824	3825	3826	3827	3828	3829	3830	3831
7370	3832	3833	3834	3835	3836	3837	3838	3839

	0	1	2	3	4	5	6	7
7400	3840	3841	3842	3843	3844	3845	3846	3847
7410	3848	3849	3850	3851	3852	3853	3854	3855
7420	3856	3857	3858	3859	3860	3861	3862	3863
7430	3864	3865	3866	3867	3868	3869	3870	3871
7440	3872	3873	3874	3875	3876	3877	3878	3879
7450	3880	3881	3882	3883	3884	3885	3886	3887
7460	3888	3889	3890	3891	3892	3893	3894	3895
7470	3896	3897	3898	3899	3900	3901	3902	3903
7500	3904	3905	3906	3907	3908	3909	3910	3911
7510	3912	3913	3914	3915	3916	3917	3918	3919
7520	3920	3921	3922	3923	3924	3925	3926	3927
7530	3928	3929	3930	3931	3932	3933	3934	3935
7540	3936	3937	3938	3939	3940	3941	3942	3943
7550	3944	3945	3946	3947	3948	3949	3950	3951
7560	3952	3953	3954	3955	3956	3957	3958	3959
7570	3960	3961	3962	3963	3964	3965	3966	3967
7600	3968	3969	3970	3971	3972	3973	3974	3975
7610	3976	3977	3978	3979	3980	3981	3982	3983
7620	3984	3985	3986	3987	3988	3989	3990	3991
7630	3992	3993	3994	3995	3996	3997	3998	3999
7640	4000	4001	4002	4003	4004	4005	4006	4007
7650	4008	4009	4010	4011	4012	4013	4014	4015
7660	4016	4017	4018	4019	4020	4021	4022	4023
7670	4024	4025	4026	4027	4028	4029	4030	4031
7700	4032	4033	4034	4035	4036	4037	4038	4039
7710	4040	4041	4042	4043	4044	4045	4046	4047
7720	4048	4049	4050	4051	4052	4053	4054	4055
7730	4056	4057	4058	4059	4060	4061	4062	4063
7740	4064	4065	4066	4067	4068	4069	4070	4071
7750	4072	4073	4074	4075	4076	4077	4078	4079
7760	4080	4081	4082	4083	4084	4085	4086	4087
7770	4088	4089	4090	4091	4092	4093	4094	4095

7000 to 7777 (Octal)	3584 to 4095 (Decimal)

Octal-Decimal Fraction Conversion Table

OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000	.000000	.100	.125000	.200	.250000	.300	.375000
.001	.001953	.101	.126953	.201	.251953	.301	.376953
.002	.003906	.102	.128906	.202	.253906	.302	.378906
.003	.005859	.103	.130859	.203	.255859	.303	.380859
.004	.007812	.104	.132812	.204	.257812	.304	.382812
.005	.009765	.105	.134765	.205	.259765	.305	.384765
.006	.011718	.106	.136718	.206	.261718	.306	.386718
.007	.013671	.107	.138671	.207	.263671	.307	.388671
.010	.015625	.110	.140625	.210	.265625	.310	.390625
.011	.017578	.111	.142578	.211	.267578	.311	.392578
.012	.019531	.112	.144531	.212	.269531	.312	.394531
.013	.021484	.113	.146484	.213	.271484	.313	.396484
.014	.023437	.114	.148437	.214	.273437	.314	.398437
.015	.025390	.115	.150390	.215	.275390	.315	.400390
.016	.027343	.116	.152343	.216	.277343	.316	.402343
.017	.029296	.117	.154296	.217	.279296	.317	.404296
.020	.031250	.120	.156250	.220	.281250	.320	.406250
.021	.033203	.121	.158203	.221	.283203	.321	.408203
.022	.035156	.122	.160156	.222	.285156	.322	.410156
.023	.037109	.123	.162109	.223	.287109	.323	.412109
.024	.039062	.124	.164062	.224	.289062	.324	.414062
.025	.041015	.125	.166015	.225	.291015	.325	.416015
.026	.042968	.126	.167968	.226	.292968	.326	.417968
.027	.044921	.127	.169921	.227	.294921	.327	.419921
.030	.046875	.130	.171875	.230	.296875	.330	.421875
.031	.048828	.131	.173828	.231	.298828	.331	.423828
.032	.050781	.132	.175781	.232	.300781	.332	.425781
.033	.052734	.133	.177734	.233	.302734	.333	.427734
.034	.054687	.134	.179687	.234	.304687	.334	.429687
.035	.056640	.135	.181640	.235	.306640	.335	.431640
.036	.058593	.136	.183593	.236	.308593	.336	.433593
.037	.060546	.137	.185546	.237	.310546	.337	.435546
.040	.062500	.140	.187500	.240	.312500	.340	.437500
.041	.064453	.141	.189453	.241	.314453	.341	.439453
.042	.066406	.142	.191406	.242	.316406	.342	.441406
.043	.068359	.143	.193359	.243	.318359	.343	.443359
.044	.070312	.144	.195312	.244	.320312	.344	.445312
.045	.072265	.145	.197265	.245	.322265	.345	.447265
.046	.074218	.146	.199218	.246	.324218	.346	.449218
.047	.076171	.147	.201171	.247	.326171	.347	.451171
.050	.078125	.150	.203125	.250	.328125	.350	.453125
.051	.080078	.151	.205078	.251	.330078	.351	.455078
.052	.082031	.152	.207031	.252	.332031	.352	.457031
.053	.083984	.153	.208984	.253	.333984	.353	.458984
.054	.085937	.154	.210937	.254	.335937	.354	.460937
.055	.087890	.155	.212890	.255	.337890	.355	.462890
.056	.089843	.156	.214843	.256	.339843	.356	.464843
.057	.091796	.157	.216796	.257	.341796	.357	.466796
.060	.093750	.160	.218750	.260	.343750	.360	.468750
.061	.095703	.161	.220703	.261	.345703	.361	.470703
.062	.097656	.162	.222656	.262	.347656	.362	.472656
.063	.099609	.163	.224609	.263	.349609	.363	.474609
.064	.101562	.164	.226562	.264	.351562	.364	.476562
.065	.103515	.165	.228515	.265	.353515	.365	.478515
.066	.105468	.166	.230468	.266	.355468	.366	.480468
.067	.107421	.167	.232421	.267	.357421	.367	.482421
.070	.109375	.170	.234375	.270	.359375	.370	.484375
.071	.111328	.171	.236328	.271	.361328	.371	.486328
.072	.113281	.172	.238281	.272	.363281	.372	.488281
.073	.115234	.173	.240234	.273	.365234	.373	.490234
.074	.117187	.174	.242187	.274	.367187	.374	.492187
.075	.119140	.175	.244140	.275	.369140	.375	.494140
.076	.121093	.176	.246093	.276	.371093	.376	.496093
.077	.123046	.177	.248046	.277	.373046	.377	.498046

Octal-Decimal Fraction Conversion Table

OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000000	.000000	.000100	.000244	.000200	.000488	.000300	.000732
.000001	.000003	.000101	.000247	.000201	.000492	.000301	.000736
.000002	.000007	.000102	.000251	.000202	.000495	.000302	.000740
.000003	.000011	.000103	.000255	.000203	.000499	.000303	.000743
.000004	.000015	.000104	.000259	.000204	.000503	.000304	.000747
.000005	.000019	.000105	.000263	.000205	.000507	.000305	.000751
.000006	.000022	.000106	.000267	.000206	.000511	.000306	.000755
.000007	.000026	.000107	.000270	.000207	.000514	.000307	.000759
.000010	.000030	.000110	.000274	.000210	.000518	.000310	.000762
.000011	.000034	.000111	.000278	.000211	.000522	.000311	.000766
.000012	.000038	.000112	.000282	.000212	.000526	.000312	.000770
.000013	.000041	.000113	.000286	.000213	.000530	.000313	.000774
.000014	.000045	.000114	.000289	.000214	.000534	.000314	.000778
.000015	.000049	.000115	.000293	.000215	.000537	.000315	.000782
.000016	.000053	.000116	.000297	.000216	.000541	.000316	.000785
.000017	.000057	.000117	.000301	.000217	.000545	.000317	.000789
.000020	.000061	.000120	.000305	.000220	.000549	.000320	.000793
.000021	.000064	.000121	.000308	.000221	.000553	.000321	.000797
.000022	.000068	.000122	.000312	.000222	.000556	.000322	.000801
.000023	.000072	.000123	.000316	.000223	.000560	.000323	.000805
.000024	.000076	.000124	.000320	.000224	.000564	.000324	.000808
.000025	.000080	.000125	.000324	.000225	.000568	.000325	.000812
.000026	.000083	.000126	.000328	.000226	.000572	.000326	.000816
.000027	.000087	.000127	.000331	.000227	.000576	.000327	.000820
.000030	.000091	.000130	.000335	.000230	.000579	.000330	.000823
.000031	.000095	.000131	.000339	.000231	.000583	.000331	.000827
.000032	.000099	.000132	.000343	.000232	.000587	.000332	.000831
.000033	.000102	.000133	.000347	.000233	.000591	.000333	.000835
.000034	.000106	.000134	.000350	.000234	.000595	.000334	.000839
.000035	.000110	.000135	.000354	.000235	.000598	.000335	.000843
.000036	.000114	.000136	.000358	.000236	.000602	.000336	.000846
.000037	.000118	.000137	.000362	.000237	.000606	.000337	.000850
.000040	.000122	.000140	.000366	.000240	.000610	.000340	.000854
.000041	.000125	.000141	.000370	.000241	.000614	.000341	.000858
.000042	.000129	.000142	.000373	.000242	.000617	.000342	.000862
.000043	.000133	.000143	.000377	.000243	.000621	.000343	.000865
.000044	.000137	.000144	.000381	.000244	.000625	.000344	.000869
.000045	.000141	.000145	.000385	.000245	.000629	.000345	.000873
.000046	.000144	.000146	.000389	.000246	.000633	.000346	.000877
.000047	.000148	.000147	.000392	.000247	.000637	.000347	.000881
.000050	.000152	.000150	.000396	.000250	.000640	.000350	.000885
.000051	.000156	.000151	.000400	.000251	.000644	.000351	.000888
.000052	.000160	.000152	.000404	.000252	.000648	.000352	.000892
.000053	.000164	.000153	.000408	.000253	.000652	.000353	.000896
.000054	.000167	.000154	.000411	.000254	.000656	.000354	.000900
.000055	.000171	.000155	.000415	.000255	.000659	.000355	.000904
.000056	.000175	.000156	.000419	.000256	.000663	.000356	.000907
.000057	.000179	.000157	.000423	.000257	.000667	.000357	.000911
.000060	.000183	.000160	.000427	.000260	.000671	.000360	.000915
.000061	.000186	.000161	.000431	.000261	.000675	.000361	.000919
.000062	.000190	.000162	.000434	.000262	.000679	.000362	.000923
.000063	.000194	.000163	.000438	.000263	.000682	.000363	.000926
.000064	.000198	.000164	.000442	.000264	.000686	.000364	.000930
.000065	.000202	.000165	.000446	.000265	.000690	.000365	.000934
.000066	.000205	.000166	.000450	.000266	.000694	.000366	.000938
.000067	.000209	.000167	.000453	.000267	.000698	.000367	.000942
.000070	.000213	.000170	.000457	.000270	.000701	.000370	.000946
.000071	.000217	.000171	.000461	.000271	.000705	.000371	.000949
.000072	.000221	.000172	.000465	.000272	.000709	.000372	.000953
.000073	.000225	.000173	.000469	.000273	.000713	.000373	.000957
.000074	.000228	.000174	.000473	.000274	.000717	.000374	.000961
.000075	.000232	.000175	.000476	.000275	.000720	.000375	.000965
.000076	.000236	.000176	.000480	.000276	.000724	.000376	.000968
.000077	.000240	.000177	.000484	.000277	.000728	.000377	.000972

Octal-Decimal Fraction Conversion Table

OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000400	.000976	.000500	.001220	.000600	.001464	.000700	.001708
.000401	.000980	.000501	.001224	.000601	.001468	.000701	.001712
.000402	.000984	.000502	.001228	.000602	.001472	.000702	.001716
.000403	.000988	.000503	.001232	.000603	.001476	.000703	.001720
.000404	.000991	.000504	.001235	.000604	.001480	.000704	.001724
.000405	.000995	.000505	.001239	.000605	.001483	.000705	.001728
.000406	.000999	.000506	.001243	.000606	.001487	.000706	.001731
.000407	.001003	.000507	.001247	.000607	.001491	.000707	.001735
.000410	.001007	.000510	.001251	.000610	.001495	.000710	.001739
.000411	.001010	.000511	.001255	.000611	.001499	.000711	.001743
.000412	.001014	.000512	.001258	.000612	.001502	.000712	.001747
.000413	.001018	.000513	.001262	.000613	.001506	.000713	.001750
.000414	.001022	.000514	.001266	.000614	.001510	.000714	.001754
.000415	.001026	.000515	.001270	.000615	.001514	.000715	.001758
.000416	.001029	.000516	.001274	.000616	.001518	.000716	.001762
.000417	.001033	.000517	.001277	.000617	.001522	.000717	.001766
.000420	.001037	.000520	.001281	.000620	.001525	.000720	.001770
.000421	.001041	.000521	.001285	.000621	.001529	.000721	.001773
.000422	.001045	.000522	.001289	.000622	.001533	.000722	.001777
.000423	.001049	.000523	.001293	.000623	.001537	.000723	.001781
.000424	.001052	.000524	.001296	.000624	.001541	.000724	.001785
.000425	.001056	.000525	.001300	.000625	.001544	.000725	.001789
.000426	.001060	.000526	.001304	.000626	.001548	.000726	.001792
.000427	.001064	.000527	.001308	.000627	.001552	.000727	.001796
.000430	.001068	.000530	.001312	.000630	.001556	.000730	.001800
.000431	.001071	.000531	.001316	.000631	.001560	.000731	.001804
.000432	.001075	.000532	.001319	.000632	.001564	.000732	.001808
.000433	.001079	.000533	.001323	.000633	.001567	.000733	.001811
.000434	.001083	.000534	.001327	.000634	.001571	.000734	.001815
.000435	.001087	.000535	.001331	.000635	.001575	.000735	.001819
.000436	.001091	.000536	.001335	.000636	.001579	.000736	.001823
.000437	.001094	.000537	.001338	.000637	.001583	.000737	.001827
.000440	.001098	.000540	.001342	.000640	.001586	.000740	.001831
.000441	.001102	.000541	.001346	.000641	.001590	.000741	.001834
.000442	.001106	.000542	.001350	.000642	.001594	.000742	.001838
.000443	.001110	.000543	.001354	.000643	.001598	.000743	.001842
.000444	.001113	.000544	.001358	.000644	.001602	.000744	.001846
.000445	.001117	.000545	.001361	.000645	.001605	.000745	.001850
.000446	.001121	.000546	.001365	.000646	.001609	.000746	.001853
.000447	.001125	.000547	.001369	.000647	.001613	.000747	.001857
.000450	.001129	.000550	.001373	.000650	.001617	.000750	.001861
.000451	.001132	.000551	.001377	.000651	.001621	.000751	.001865
.000452	.001136	.000552	.001380	.000652	.001625	.000752	.001869
.000453	.001140	.000553	.001384	.000653	.001628	.000753	.001873
.000454	.001144	.000554	.001388	.000654	.001632	.000754	.001876
.000455	.001148	.000555	.001392	.000655	.001636	.000755	.001880
.000456	.001152	.000556	.001396	.000656	.001640	.000756	.001884
.000457	.001155	.000557	.001399	.000657	.001644	.000757	.001888
.000460	.001159	.000560	.001403	.000660	.001647	.000760	.001892
.000461	.001163	.000561	.001407	.000661	.001651	.000761	.001895
.000462	.001167	.000562	.001411	.000662	.001655	.000762	.001899
.000463	.001171	.000563	.001415	.000663	.001659	.000763	.001903
.000464	.001174	.000564	.001419	.000664	.001663	.000764	.001907
.000465	.001178	.000565	.001422	.000665	.001667	.000765	.001911
.000466	.001182	.000566	.001426	.000666	.001670	.000766	.001914
.000467	.001186	.000567	.001430	.000667	.001674	.000767	.001918
.000470	.001190	.000570	.001434	.000670	.001678	.000770	.001922
.000471	.001194	.000571	.001438	.000671	.001682	.000771	.001926
.000472	.001197	.000572	.001441	.000672	.001686	.000772	.001930
.000473	.001201	.000573	.001445	.000673	.001689	.000773	.001934
.000474	.001205	.000574	.001449	.000674	.001693	.000774	.001937
.000475	.001209	.000575	.001453	.000675	.001697	.000775	.001941
.000476	.001213	.000576	.001457	.000676	.001701	.000776	.001945
.000477	.001216	.000577	.001461	.000677	.001705	.000777	.001949

SECTION 8

COMMANDS

LOAD A

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Load (OPA) into A.	NORMAL	LA -N LA	00 70	OPA	CIT-CIS+2	No	B, C, D	(OPA) → A			
	DELAYED	LA -D	10 70	OPA	CIT-OPS+1				A (OPA)	1777777777 0000001234	0000001234 N.C.
Load (OPA-(I)) into A.	NORMAL INDEXED	LA -NI LA -I	0071	OPA	CIT-CIS+2	No	B, C, D	(OPA-(I)) → A			
	DELAYED INDEXED	LA -DI	1071	OPA	CIT-OPS+1				A (OPA-(I))	1777777777 0000001234	0000001234 N.C.
Load $OPRND_{16-1}$ into A_{16-1}.	OPERAND	LA -O	0470	OPRND	CIT-CIS+2	No	-	$OPRND_{16-1} \rightarrow A_{16-1}$ $0 \rightarrow A_{S,27-17}$	A OPRND	1777777777 123456	0000123456 N.C.
Load $OPRND_{14-1}$ into $A_{S,27-15}$.	UPPER FILL	LA -U	0670	OPRND	CIT-CIS+2	No	-	$OPRND_{14-1} \rightarrow A_{S,27-15}$ $A_{14-1} \rightarrow A_{14-1}$	A OPRND	1077006452 01740	0076006452 N.C.

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) If C register addressed: (C) ⟶ A_{16-1}; 0 ⟶ $A_{S,27-17}$.

LOAD B

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Load (OPA) into B.	NORMAL	LB -N LB	00 32	OPA	CIT-CIS+2	No	C, D	(OPA) → B			
	DELAYED	LB -D	1032	OPA	CIT-OPS+1				B (OPA)	1777777777 0000001234	0000001234 N.C.
Load (OPA-(I)) into B.	NORMAL INDEXED	LB -NI LB -I	00 33	OPA	CIT-CIS+2	No	C, D	(OPA-(I)) → B			
	DELAYED INDEXED	LB -DI	1033	OPA	CIT-OPS+1				B (OPA-(I))	1777777777 0000001234	0000001234 N.C.
Load $OPRND_{16-1}$ into B_{16-1}.	OPERAND	LB -O	0432	OPRND	CIT-CIS+2	No	-	$OPRND_{16-1} \rightarrow B_{16-1}$ $0 \rightarrow B_{S,27-17}$	B OPRND	1777777777 123456	0000123456 N.C.
Load $OPRND_{14-1}$ into $B_{S,27-15}$.	UPPER FILL	LB -U	0632	OPRND	CIT-CIS+2	No	-	$OPRND_{14-1} \rightarrow B_{S,27-15}$ $B_{14-1} \rightarrow B_{14-1}$	B OPRND	1777777777 12342	0516137777 N.C.

NOTES: (1) Registers: A, C, D, I, and X unchanged. (2) If C register addressed: (C) ⟶ B_{16-1}; 0 ⟶ $B_{S,27-17}$.

LOAD C

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Load $(OPA)_{16-1}$ into C.	NORMAL	LC -N LC	0034	OPA	CIT-CIS+2	No	B, D	$(OPA)_{16-1} \rightarrow C$			
	DELAYED	LC -D	1034	OPA	CIT-OPS+1				C (OPA)	377777 012345	012345 N.C.
Load $(OPA-(I))_{16-1}$ into C.	NORMAL INDEXED	LC -NI LC -I	0035	OPA	CIT-CIS+2	No	B, D	$(OPA-(I))_{16-1} \rightarrow C$			
	DELAYED INDEXED	LC -DI	1035	OPA	CIT-OPS+1				C (OPA-(I))	377777 012345	012345 N.C.
Load $OPRND_{16-1}$ into C.	OPERAND	LC -O	0434	OPRND	CIT-CIS+2	No	-	$OPRND_{16-1} \rightarrow C$	C OPRND	377777 123456	123456 N.C.

NOTES: (1) Registers: A, B, D, I, and X unchanged.

LOAD D

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Load (OPA) into D.	NORMAL	LD -N LD	0024	OPA	CIT-CIS+2	No	B, C	$(OPA) \rightarrow D$			
	DELAYED	LD -D	1024	OPA	CIT-OPS+1				D (OPA)	1777777777 0000001234	0000001234 N.C.
Load (OPA-(I)) into D.	NORMAL INDEXED	LD -NI LD -I	0025	OPA	CIT-CIS+2	No	B, C	$(OPA-(I)) \rightarrow D$			
	DELAYED INDEXED	LD -DI	1025	OPA	CIT-OPS+1				D (OPA-(I))	1777777777 0000001234	0000001234 N.C.
Load $OPRND_{16-1}$ into D_{16-1}.	OPERAND	LD -O	0424	OPRND	CIT-CIS+2	No	-	$OPRND_{16-1} \rightarrow D_{16-1}$ $0 \rightarrow D_{S,27-17}$	D OPRND	1066042571 123456	0000123456 N.C.
Load $OPRND_{14-1}$ into $D_{S,27-15}$.	UPPER FILL	LD -U	0624	OPRND	CIT-CIS+2	No	-	$OPRND_{14-1} \rightarrow D_{S,27-15}$ $D_{14-1} \rightarrow D_{14-1}$	D OPRND	1777777777 12342	0516137777 N.C.

NOTES: (1) Registers: A, B, C, I, and X unchanged. (2) If C register addressed: (C) $\longrightarrow$ D_{16-1}; 0 $\longrightarrow$ $D_{S,27-17}$.

LOAD INDEX

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Load $(OPA)_{16-1}$ into I.	NORMAL	LI -N LI	0036	OPA	CIT-CIS+2	No	B,C,D	$(OPA)_{16-1} \rightarrow I$			
	DELAYED	LI -D	1036	OPA	CIT-OPS+1				I (OPA)	177777 0000000005	000005 N.C.
Load $(OPA-(I))_{16-1}$ into I.	NORMAL INDEXED	LI -NI LI -I	0037	OPA	CIT-CIS+2	No	B,C,D	$(OPA-(I))_{16-1} \rightarrow I$			
	DELAYED INDEXED	LI -DI	1037	OPA	CIT-OPS+1				I (OPA-(I))	177777 0000000005	000005 N.C
Load $OPRND_{16-1}$ into I.	OPERAND	LI -O	0436	OPRND	CIT-CIS+2	No	-	$OPRND_{16-1} \rightarrow I$			
									I OPRND	177777 000005	000005 N.C

NOTES: (1) Registers: A, B, C, D, and X unchanged.

LOAD X

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Load $(OPA)_{8-1}$ into X.	NORMAL	LX -N LX	0022	OPA	CIT-CIS+2	No	B,C,D	$(OPA)_{8-1} \rightarrow X$			
	DELAYED	LX -D	1022	OPA	CIT-OPS+1				X (OPA)	377 0000001234	234 N.C.
Load $(OPA-(I))_{8-1}$ into X.	NORMAL INDEXED	LX -NI LX -I	0023	OPA	CIT-CIS+2	No	B,C,D	$(OPA-I)_{8-1} \rightarrow X$			
	DELAYED INDEXED	LX -DI	1023	OPA	CIT-OPS+1				X (OPA-(I))	377 0000000234	234 N.C.
Load $OPRND_{8-1}$ into X.	OPERAND	LX -O	0422	OPRND	CIT-CIS+2	No	-	$OPRND_{8-1} \rightarrow X$			
									X OPRND	377 123456	256 N.C

NOTES: (1) Registers: A, B, C, D, and I unchanged. (2) If C register addressed: $(C)_{8-1} \longrightarrow I$.

STORE A

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Store (A) in OPA.	NORMAL	SA -N SA	0076	OPA	CIT-CIS+2	No	B, C, D	(A) → OPA			
	DELAYED	SA -D	1076	OPA	CIT-OPS+1				A	0000004050	N.C.
									(OPA)	0000001234	0000004050
Store (A) in OPA-(I).	NORMAL INDEXED	SA -NI SA -I	0077	OPA	CIT-CIS+2	No	B, C, D	(A) → OPA-(I)			
	DELAYED INDEXED	SA -DI	1077	OPA	CIT-OPS+1				A	0000004050	N.C.
									(OPA-(I))	0000123456	0000004050

NOTES: (1) Registers: A, I, and X unchanged; B, C, and D unchanged, unless addressed.

(2) If C register addressed: $(A)_{16-1} \rightarrow C$.

STORE A REPEAT

EXECUTION TIME— 2+(X) WORD TIMES 266+133(X) MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Store (A) in OPA and in all consecutive sectors up to OPA+(X).	NORMAL	SAR -N SAR	0176	OPA	CIT-CIS+(X)+2	No	B, C, D	(A) → OPA, OPA+1 OPA+(X) (X) → X	A	0000000000	N.C.
									X	177	N.C.
	DELAYED	SAR -D	1176	OPA	CIT-OPS+(X)+1				(OPA+177)	1456701234	0000000000
									(OPA)	1234567012	0000000000
Store (A) in OPA-(I) and in all consecutive sectors up to OPA-(I)+(X).	NORMAL INDEXED	SAR -NI SAR -I	0177	OPA	CIT-CIS+(X)+2	No	B, C, D	(A) → OPA-(I), OPA-(I)+1, OPA-(I)+(X) (X) → X	A	0000000000	N.C.
									X	012	N.C.
	DELAYED INDEXED	SAR -DI	1177	OPA	CIT-OPS+(X)+1				(OPA-(I)+12)	0000001234	0000000000
									(OPA-(I))	1777777777	0000000000

NOTES: (1) Registers: A, B, C, D, I, and X unchanged; although B, C, and D registers are addressable, this command would not be used to address them.

STORE B

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Store (B) in OPA.	NORMAL	SB -N SB	0066	OPA	CIT-CIS+2	No	C, D	(B) → OPA			
	DELAYED	SB -D	1066	OPA	CIT-OPS+1				B (OPA)	0000004050 0000001234	N.C. 0000004050
Store (B) in OPA-(I).	NORMAL INDEXED	SB -NI SB -I	0067	OPA	CIT-CIS+2	No	C, D	(B) → OPA-(I)			
	DELAYED INDEXED	SB -DI	1067	OPA	CIT-OPS+1				B (OPA-(I))	0000004050 0000001234	N.C. 0000004050

NOTES: (1) Registers: A, B, I, and X unchanged; C and D unchanged, unless addressed.

(2) If C register addressed: $(B)_{16-1} \rightarrow C$.

STORE D

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Store (D) in OPA.	NORMAL	SD -N SD	0072	OPA	CIT-CIS+2	No	B, C	(D) → OPA			
	DELAYED	SD -D	1072	OPA	CIT-OPS+1				D (OPA)	0000004050 0000001234	N.C. 0000004050
Store (D) in OPA-(I).	NORMAL INDEXED	SD -NI SD -I	0073	OPA	CIT-CIS+2	No	B, C	(D) → OPA-(I)			
	DELAYED INDEXED	SD -DI	1073	OPA	CIT-OPS+1				D (OPA-(I))	0004050000 0000001234	N.C. 0004050000

NOTES: (1) Registers: A, D, I, and X unchanged; B and C unchanged, unless addressed.

(2) If C register addressed: $(D)_{16-1} \rightarrow C$.

EXCHANGE AB

EXECUTION TIME— 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Exchange (A) with (B).	NORMAL	EAB -N EAB	0042	035-NNN	CIT-CIS+2	No	-	(A) → B (B) → A	A	0000001234	1777777777
									B	1777777777	0000001234
	DELAYED	EAB -D	1042	035-OPS	CIT-OPS+1				(OPA)	-	-

NOTES: (1) Registers: C, D, I, and X unchanged.

EXCHANGE AC

EXECUTION TIME— 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Exchange $(A)_{16-1}$ with (C).	NORMAL	EAC -N EAC	0042	031-NNN	CIT-CIS+	No	-	$(A)_{16-1} \rightarrow C$ $(C) \rightarrow A_{16-1}$ $0 \rightarrow A_{5,27-17}$	A	1777000177	0000123456
									C	123456	000177
	DELAYED	EAC -D	1042	031-OPS	CIT-OPS+				(OPA)	-	-

NOTES: (1) Registers: B, D, I, and X unchanged.

EXCHANGE AD

EXECUTION TIME— 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Exchange (A) with (D).	NORMAL	EAD -N EAD	0042	030-NNN	CIT-CIS+2	No	-	(A) → D (D) → A	A	0000001234	1777777777
									D	1777777777	0000001234
	DELAYED	EAD -D	1042	030-OPS	CIT-OPS+1				(OPA)		-

NOTES: (1) Registers: B, C, I, and X unchanged.

EXCHANGE AI

EXECUTION TIME— 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Exchange $(A)_{16-1}$ with (I).	NORMAL	EAI -N EAI	0042	025-NNN	CIT-CIS+2	No	-	$(A)_{16-1} \rightarrow I$ $(I) \rightarrow A_{16-1}$ $0 \rightarrow A_{S,27-17}$	A	1777000177	0000123456
									I	123456	000177
	DELAYED	EAI -D	1042	025-OPS	CIT-OPS+1						
									(OPA)	—	—

NOTES: (1) Registers B, C, D, and X unchanged.

REPLACE A WITH I

EXECUTION TIME— 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer (I) into A_{16-1}.	NORMAL	RAI -N RAI	0142	025-NNN	CIT-CIS+2	No	-	$(I) \rightarrow A_{16-1}$ $0 \rightarrow A_{S,27-17}$ $(I) \rightarrow I$	A	1777777777	0000 000277
									I	000277	N.C.
	DELAYED	RAI -D	1142	025-OPS	CIT-OPS+1						
									(OPA)	—	—

NOTES: (1) Registers: B, C, D, I, and X unchanged.

REPLACE I WITH A

EXECUTION TIME— 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer $(A)_{16-1}$ into I.	NORMAL	RIA -N RIA	0242	025-NNN	CIT-CIS+2	No	-	$(A)_{16-1} \rightarrow I$ $(A) \rightarrow A$	A	0000000350	N.C.
									I	123456	000350
	DELAYED	RIA -D	1242	025-OPS	CIT-OPS+1						
									(OPA)	—	—

NOTES: (1) Registers: A, B, C, D, and X unchanged.

EXCHANGE AX

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
/ Exchange $(A)_{8-1}$ with (X).	NORMAL	EAX -N EAX	0042	024-NNN	CIT-CIS+2	No	-	$(A)_{8-1} \rightarrow X$ $(X) \rightarrow A_{8-1}$ $0 \rightarrow A_{5,27-9}$	A	0000007220	0000000377
									X	377	220
	DELAYED	EAX -D	1042	024-OPS	CIT-OPS+1						
									(OPA)	-	-

NOTES: (1) Registers: B, C, D, and I unchanged.

EXCHANGE AX LOW BITS

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer $(A)_{8-1}$ to X, and $(X)_{7-1}$ to A_{7-1}.	NORMAL	EAXL -N EAXL	0442	024-NNN	CIT-CIS+2	No	-	$(X)_{7-1} \rightarrow A_{7-1}$ $(A)_{5,27-8} \rightarrow A_{5,27-8}$ $(A)_{8-1} \rightarrow X$	A	0212006660	0212006620
									X	020	260
	DELAYED	EAXL -D	1442	024-PPP	CIT-OPS+1						
									(OPA)		

REPLACE A WITH X

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer (X) into A_{8-1}.	NORMAL	RACX -N RACX	0142	024-NNN	CIT-CIS+2	No	-	$(X) \rightarrow A_{8-1}$ $0 \rightarrow A_{5,27-9}$ $(X) \rightarrow X$	A	1777777777	0000000120
									X	120	N.C.
	DELAYED	RACX -D	1142	024-OPS	CIT-OPS+1						
									(OPA)	-	-

NOTES: (1) Registers: B, C, D, I, and X unchanged.

REPLACE A WITH X LOW BITS

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer $(X)_{7-1}$ into A_{7-1}.	NORMAL	RAHX -N RAHX	0542	024-NNN	CIT-CIS+2	No	-	$(X)_{7-1} \rightarrow A_{7-1}$ $(A)_{S,27-8} \rightarrow A_{S,27-8}$ $(X) \rightarrow X$	A	1000555055	1000555177
									X	177	N.C
	DELAYED	RAHX -D	1542	024-OPS	CIT-OPS+1				(OPA)	-	-

NOTES: (1) Registers: B, C, D, I, and X unchanged.

REPLACE X WITH A

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer $(A)_{8-1}$ into X.	NORMAL	RXA -N RXA	0242	024-NNN	CIT-CIS+2	No	-	$(A)_{8-1} \rightarrow X$ $(A) \rightarrow A$	A	000000277	N.C
									X	120	277
	DELAYED	RXA -D	1242	024-OPS	CIT-OPS+1				(OPA)	-	-

NOTES: (1) Registers: A, B, C, D, and I unchanged.

EXCHANGE AM

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Exchange (A) with (M).	NORMAL	EAM -N EAM	0042	020-NNN	CIT-CIS+2	No	-	$(A) \rightarrow M$ $(M) \rightarrow A$	A	1673726045	1776024473
									M	1776024473	1673726045
	DELAYED	EAM -D	1042	020-OPS	CIT-OPS+1				(OPA)	-	-

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) M register may be 8, 16, or 28 bits long. If M is less than 28 bits, only corresponding bits are exchanged; upper bits of A recirculate. (3) Up to four M registers may be used (up to 112 priority interrupt lines):

M Register Bits	Register Designations	Operand Track Addresses
1-28	M1	020
29-56	M2	060
57-84	M3	120
85-112	M4	160

MASKED EXCHANGE AM

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Exchange corresponding bits between A and M for which there is a "1" in the corresponding bit position in the B register.	NORMAL	EABM-N EABM	0442	020-NNN	CIT-CIS+2	No	-	SEE NOTES	A	1777777777	1775777777
									B	0002000000	N.C.
	DELAYED	EABM-D	1442	020-OPS	CIT-OPS+1				M	1755776777	1757776777
									(OPA)		

NOTES: (1) See "Exchange AM". (2) B register used as a mask: 1-bits in B register permit corresponding bit positions in A to be written into corresponding bit positions in M, and vice versa; 0-bits in B register mask corresponding bit positions in A and M registers from being written into, and those bit positions in A and M are unchanged.

REPLACE A WITH M

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer (M) into A.	NORMAL	RAM-N RAM	0142	020-NNN	CIT-CIS+2	No	-	(M) → A (M) → M	A	0000004050	1777676467
									M	1777676467	N.C.
	DELAYED	RAM-D	1142	020-OPS	CIT-OPS+1						
									(OPA)	-	-

NOTES: (1) See "Exchange AM".

REPLACE M WITH A

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer (A) into M.	NORMAL	RAM-N RAM	0242	020-NNN	CIT-CIS+2	No	-	(A) → M (A) → A	A	1777777777	N.C.
									M	0177777777	1777777777
	DELAYED	RAM-D	1242	020-OPS	CIT-OPS+1						
									(OPA)		

NOTES: (1) See "Exchange AM".

MASKED REPLACE M WITH A

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Replace corresponding bits in M from A, for which there is a "1" in the corresponding bit position in the B register.	NORMAL	RMAB -N RMAB	0642	020-NNN	CIT-CIS+2	No	-	SEE NOTES	A	1777777777	N.C.
									B	0002000000	N.C
	DELAYED	RMAB -D	1642	020-OPS	CIT-OPS+1				M	1755776777	1757776777
									(OPA)		

NOTES: (1) See "Exchange AM". (2) B register used as a mask: 1-bits in B register permit corresponding bit positions in A to be written into corresponding bit positions in M; 0-bits in B register mask corresponding bit positions in M register from being written into, and those bit positions in M are unchanged. (A and B are unchanged.)

EXCHANGE AQ

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Exchange (A) with (Q).	NORMAL	EAQ -N EAQ	0042	021-NNN	CIT-CIS+2	No	-	(A) → Q (Q) → A	A	0000000000	0000001001
									Q	0000001001	0000000000
	DELAYED	EAQ -D	1042	021-OPS	CIT-OPS+1						
									(OPA)	-	-

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) Q register may be 8, 16, or 28 bits long. If Q is less than 28 bits, only corresponding bits are exchanged; upper bits of A recirculate. (3) Up to four Q registers may be used (up to 112 priority interrupt lines):

Q Register Bits	Register Designations	Operand Track Addresses
1-28	Q1	021
29-56	Q2	061
57-84	Q3	121
85-112	Q4	161

REPLACE A WITH Q

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer (Q) into A.	NORMAL	RAQ -N RAQ	0142	021-NNN	CIT-CIS+2	No	-	(Q) → A (Q) → Q	A	0000123456	0000002000
									Q	0000002000	N.C
	DELAYED	RAQ -D	1142	021-OPS	CIT-OPS+1						
									(OPA)		

NOTES: (1) See "Exchange AQ".

REPLACE Q WITH A

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Transfer (A) into Q.	NORMAL	RQA -N RQA	0242	021-NNN	CIT-CIS+2	No	-	(A) → Q (A) → A	A	0000000000	M.C.
									Q	0000020000	0000000000
	DELAYED	RQA -D	1242	021-OPS	CIT-OPS+1						
									(OPA)	-	-

NOTES:
(1) See "Exchange AQ".

CLEAR AB

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Set (A) and (B) to zero.	NORMAL	ZAB -N ZAB	0042	036-NNN	CIT-CIS+2	No	-	0 → A 0 → B			
									A	1777777777	0000000000
	DELAYED	ZAB -D	1042	036-OPS	CIT-OPS+1				B	1333333337	0000000000
									(OPA)	-	-

NOTES: (1) Registers: C, D, I, and X unchanged.

COMPLEMENT A

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Replace (A) with its 2's complement.	NORMAL	CMP -N CMP	0042	026-NNN	CIT-CIS+2	Yes	-	2's COMPLEMENT of (A) → A	A	0000001234	1777776544
	DELAYED	CMP -D	1042	026-OPS	CIT-OPS+1				(OPA)	-	-

NOTES: (1) Registers: B, C, D, I, and X unchanged.

MERGE

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Take logical sum of (A) and (OPA).	NORMAL	MG -N MG	0064	OPA	CIT-CIS+2	No	B, C, D	(A) ⊕ (OPA) → A			
	DELAYED	MG -D	1064	OPA	CIT-OPS+1				A	0000001234	0000005335
									(OPA)	0000004321	N.C
Take logical sum of (A) and (OPA-(I)).	NORMAL INDEXED	MG -NI MG -I	0065	OPA	CIT-CIS+2	No	B, C, D	(A) ⊕ (OPA-(I)) → A			
	DELAYED INDEXED	MG -DI	1065	OPA	CIT-OPS+1				A	0000001234	0000005335
									(OPA-(I))	0000004321	N C
Take logical sum of (A) and $OPRND_{16-1}$.	OPERAND	MG -O	0464	OPRND	CIT-CIS+2	No	-	$(A)_{16-1} \oplus OPRND_{16-1} \rightarrow A_{16-1}$ $(A)_{S,27-17} \rightarrow A_{S,27-17}$			
									A	1111611234	1111715335
									OPRND	114321	N C
Take logical sum of $(A)_{S,\ 27-15}$ and $OPRND_{14-1}$.	UPPER FILL	MG -U	0664	OPRND	CIT-CIS+2	No	-	$(A)_{S,27-15} \oplus OPRND_{14-1} \rightarrow A_{S,27-15}$ $(A)_{14-1} \rightarrow A_{14-1}$			
										1011161234	1217147234
									OPRND	24321	N C

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) If C register addressed: $(A)_{16-1} \oplus (C) \longrightarrow A_{16-1}$; $(A)_{S,\ 27-17} \longrightarrow A_{S,\ 27-17}$.

EXTRACT

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Take logical product of (A) and (OPA).	NORMAL	Ex -N Ex	0020	OPA	CIT-CIS+2	No	B,C,D	(A) ⊗ (OPA) → A			
	DELAYED	Ex -D	1020	OPA	CIT-OPS+1				A (OPA)	0000001234 0000004321	0000000220 N.C.
Take logical product of (A) and (OPA-(I)).	NORMAL INDEXED	Ex -NI Ex -I	0021	OPA	CIT-CIS+2	No	B,C,D	(A) ⊗ (OPA-(I)) → A			
	DELAYED INDEXED	Ex -DI	1021	OPA	CIT-OPS+1				A (OPA-(I))	0000001234 0000004321	0000000220 N.C.
Take logical product of (A) and $OPRND_{16-1}$.	OPERAND	Ex -O	0420	OPRND	CIT-CIS+2	No	-	$(A)_{16-1} \otimes OPRND_{16-1} \rightarrow A_{16-1}$ $0 \rightarrow A_{S,27-17}$			
									A OPRND	1111001234 014321	0000000220 N.C.
Take logical product of $(A)_{S,27-15}$ and $OPRND_{14-1}$.	UPPER FILL	Ex -U	0620	OPRND	CIT-CIS+2	No	-	$(A)_{S,27-15} \otimes OPRND_{14-1} \rightarrow A_{S,27-15}$ $(A)_{14-1} \rightarrow A_{14-1}$			
									A OPRND	1777777777 12344	0516231777 N.C.

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) If C register addressed: $(A)_{16-1} \otimes (C) \longrightarrow A_{16-1}$; $0 \longrightarrow A_{S,27-17}$.

REDUCE INDEX

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Subtract $(OPA)_{16-1}$ from (I).	NORMAL	RI -N RI	0026	OPA	CIT-CIS+2	No	B,C,D	$(I) - (OPA)_{16-1} \rightarrow I$			
	DELAYED	RI -D	1026	OPA	CIT-OPS+1				I (OPA)	000010 0000000001	000007 N.C.
Subtract $(OPA-(I))_{16-1}$ from (I).	NORMAL INDEXED	RI -NI RI -I	0027	OPA	CIT-CIS+2	No	B,C,D	$(I) - (OPA-(I)) \rightarrow I$			
	DELAYED INDEXED	RI -DI	1027	OPA	CIT-OPS+1				I (OPA-(I))	000010 0000000001	000007 N.C.
Subtract $OPRND_{16-1}$ from (I).	OPERAND	RI -O	0426	OPRND	CIT-CIS+2	No	-	$(I) - OPRND_{16-1} \rightarrow I$			
									OPRND	000010 0000001	000007 N.C.

NOTES: (1) Registers: A, B, C, D, and X unchanged.

ADD

EXECUTION TIME-- 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Add (OPA) to (A).	NORMAL	AD -N AD	CC 60	OPA	CIT-CIS+2	YES	B, C, D	(A) + (OPA) → A			
	DELAYED	AD -D	1C 60	OPA	CIT-OPS+1				A (OPA)	000000100 000000500	000000600 N.C.
Add (OPA-(I)) to (A).	NORMAL INDEXED	AD -NI AD -I	CC 61	OPA	CIT-CIS+2	YES	B, C, D	(A) + (OPA-I) → A			
	DELAYED INDEXED	AD -DI	1C 61	OPA	CIT-OPS+1				A (OPA-I)	000001234 1777777775	000001201 N.C.
Add $OPRND_{16-1}$ to (A).	OPERAND	AD -O	0460	OPRND	CIT-CIS+2	YES	-	(A) + $OPRND_{16-1}$ → A			
									A OPRND	1777777001 000055	1777777056 N.C.

NOTES: (1) Registers: B, C, D, I, and X unchanged.

SUBTRACT

EXECUTION TIME-- 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Subtract (OPA) from (A).	NORMAL	SU -N SU	CC 30	OPA	CIT-CIS+2	YES	B, C, D	(A) - (OPA) → A			
	DELAYED	SU -D	1C 30	OPA	CIT-OPS+1				A (OPA)	000000500 000000100	000000400 N.C.
Subtract (OPA-(I)) from (A).	NORMAL INDEXED	SU -NI SU -I	CC 31	OPA	CIT-CIS+2	YES	B, C, D	(A) - (OPA-(I)) → A			
	DELAYED INDEXED	SU -DI	1C 31	OPA	CIT-OPS+1				A (OPA-(I))	000000500 1777777600	000000700 N.C.
Subtract $OPRND_{16-1}$ from $(A)_{16-1}$.	OPERAND	SU -O		OPRND	CIT-CIS+2	YES	-	(A) - $OPRND_{16-1}$ → A			
									A OPRND	1111111111 000333	1777674444 N.C.

NOTES: (1) Registers: B, C, D, I, and X unchanged.

MULTIPLY 27

EXECUTION TIME— 30 WORD TIMES 3990 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Multiply (A) x (OPA).	NORMAL	M27 -N M27	0302	OPA	CIT-CIS+36	No	B, C, D	$(A)_{27-1} \times (OPA)_{21-1} \rightarrow A_{27-1} B_{27-1}$ $(A)_S \times (OPA)_S \rightarrow A_S \neq B_S$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000000
									B	–	0001201720
									C	–	001234
	DELAYED	M27 -D	1302	OPA	CIT-OPS+35				(OPA)	0000000754	N.C.
Multiply (A) x (OPA-(I)).	NORMAL INDEXED	M27 -NI M27 -I	0303	OPA	CIT-CIS+36	No	B, C, D	$(A)_{27-1} \times (OPA-(I))_{27-1} \rightarrow A_{27-1} B_{27-1}$ $(A)_S \times (OPA-(I))_S \rightarrow A_S \neq B_S$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000000
									B	–	0000133250
	DELAYED INDEXED	M27 -DI	1303	OPA	CIT-OPS+35				C	–	001234
									(OPA-(I))	0000000106	N.C.
Multiply (A) x $OPRND_{16-1}$.	OPERAND	M27 -O	0703	OPRND	CIT-CIS+36	No	–	$(A)_{27-1} \times OPRND_{16-1} \rightarrow A_{16-1} B_{27-1}$ $(A)_S \rightarrow A_S \neq B_S$ $(A)_S \rightarrow A_{27-17}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000000
									B	–	0000030624
									C		001234
									OPRND	000023	N.C.

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling: (A) at 2^R x (OPA) at 2^S = (A) (B) at 2^{R+S}.

MULTIPLY 21

EXECUTION TIME— 24 WORD TIMES 3192 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Multiply (A) x $(OPA)_{S, 21-1}$.	NORMAL	M21 -N M21	0202	OPA	CIT-CIS+30	No	B, C, D	$(A)_{27-1} \times (OPA)_{21-1} \rightarrow A_{27-1} B_{27-7}$ $(A)_S \times (OPA)_S \rightarrow A_S \neq B_S$ $(OPA)_{27-22} \rightarrow B_{6-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000000
									B	0000000000	0120172000
	DELAYED	M21 -D	1202	OPA	CIT-OPS+27				C	–	001234
									(OPA)	0000000754	N.C.
Multiply (A) x $(OPA-(I))_{S, 21-1}$.	NORMAL INDEXED	M21 -NI M21 -I	0203	OPA	CIT-CIS+30	No	B, C, D	$(A)_{27-1} \times (OPA-(I))_{21-1} \rightarrow A_{27-1} B_{27-7}$ $(A)_S \times (OPA-(I))_S \rightarrow A_S \neq B_S$ $(OPA-(I))_{27-22} \rightarrow B_{6-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000000
									B	0000012012	0013325000
	DELAYED INDEXED	M21 -DI	1203	OPA	CIT-OPS+27				C	–	001234
									(OPA-(I))	0000000106	N.C.
Multiply (A) x $OPRND_{16-1}$.	OPERAND	M21 -O	0602	OPRND	CIT-CIS+30	No	–	$(A)_{27-1} \times OPRND_{16-1} \rightarrow A_{16-1} B_{27-1}$ $(A)_S \rightarrow A_S \neq B_S$ $(A)_S \rightarrow A_{27-17}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000000
									B	0000007600	0003064400
									C	–	001234
									OPRND	000023	N.C.

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling: (A) at 2^R x (OPA) at 2^S = (A) $(B)_{27-7}$ at 2^{R+S+6}.

MULTIPLY 14

EXECUTION TIME— 17 WORD TIMES 2261 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Multiply (A) x $(OPA)_{S,\ 14-1}$	NORMAL	M14 -N M14	0102	OPA	CIT-CIS+21	No	B, C, D	$(A)_{27-1} \times (OPA)_{14-1} \rightarrow A_{27-1} B_{27-14}$ $(A)_S \times (OPA)_S \rightarrow A_S \neq B_S$ $(OPA)_{27-15} \rightarrow B_{13-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000024
									B	0000000000	0036400000
	DELAYED	M14 -D	1102	OPA	CIT-OPS+20				C	—	001234
									(OPA)	0000000754	N.C.
Multiply (A) x $(OPA\text{-}(I))_{S,\ 14-1}$	NORMAL INDEXED	M14 -NI M14 -I	0103	OPA	CIT-CIS+21	No	B, C, D	$(A)_{27-1} \times (OPA\text{-}(I))_{14-1} \rightarrow A_{27-1} B_{27-14}$ $(A)_S \times (OPA\text{-}(I))_S \rightarrow A_S \neq B_S$ $(OPA\text{-}(I))_{27-15} \rightarrow B_{13-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000002
									B	0000000000	0662500000
	DELAYED INDEXED	M14 -DI	1103	OPA	CIT-OPS+20				C	—	001234
									(OPA-(I))	0000000106	N.C.
Multiply (A) x $OPRND_{14-1}$	OPERAND	M14 -O	0502	OPRND	CIT-CIS+21	No	-	$(A)_{27-1} \times OPRND_{14-1} \rightarrow A_{14-1} B_{27-14}$ $(A)_S \rightarrow A_S \neq B_S$ $(A)_S \rightarrow A_{27-15}$ $OPRND_{15-15} \rightarrow B_{2-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000000
									B	0000000000	0614500000
									C	—	001234
									OPRND	000023	N.C.

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling: (A) at 2^R x (OPA) at 2^S = (A) $(B)_{27-14}$ at 2^{R+S+13}.

MULTIPLY 7

EXECUTION TIME— 10 WORD TIMES 1330 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Multiply (A) x $(OPA)_{S,\ 7-1}$	NORMAL	M7 -N M7	0002	OPA	CIT-CIS+12	No	B, C, D	$(A)_{27-1} \times (OPA)_{7-1} \rightarrow A_{27-1} B_{27-21}$ $(A)_S \times (OPA)_S \rightarrow A_S \neq B_S$ $(OPA)_{27-8} \rightarrow B_{20-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000001063
									B	0000000000	0500000000
	DELAYED	M7 -D	1002	OPA	CIT-OPS+11				C	—	001234
									(OPA)	0000000754	N.C.
Multiply (A) x $(OPA\text{-}(I))_{S,\ 7-1}$	NORMAL INDEXED	M7 -NI M7 -I	0003	OPA	CIT-CIS+12	No	B, C, D	$(A)_{27-1} \times (OPA\text{-}(I))_{7-1} \rightarrow A_{27-1} B_{27-21}$ $(A)_S \times (OPA\text{-}(I))_S \rightarrow A_S \neq B_S$ $(OPA\text{-}(I))_{27-8} \rightarrow B_{20-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000555
									B	0000000000	0240000000
	DELAYED INDEXED	M7 -DI	1003	OPA	CIT-OPS+11				C	—	001234
									(OPA-(I))	0000000106	N.C.
Multiply (A) x $OPRND_{7-1}$	OPERAND	M7 -O	0402	OPRND	CIT-CIS+12	No	-	$(A)_{27-1} \times OPRND_{7-1} \rightarrow A_{7-1} B_{27-21}$ $(A)_S \rightarrow A_S \neq B_S \rightarrow A_{27-8}$ $OPRND_{16-8} \rightarrow B_{7-1}$ $(A)_{16-1} \rightarrow C$	A	0000001234	0000000143
									B	0000000000	0120000000
									C	—	001234
									OPRND	000023	N.C.

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling: (A) at 2^R x (OPA) at 2^S = (A) $(B)_{27-21}$ at 2^{R+S+20}.

DIVIDE 27

EXECUTION TIME— 31 WORD TIMES 4123 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Divide (A) by (OPA).	NORMAL	D27 -N D27	0312	OPA	CIT-CIS+37	YES	B, C, D	(A)/(OPA) → A REMAINDER → B (OPA)$_{16-1}$ → C	A	0000000133	0443146314
									B	-	0000000200
									C	-	000240
									(OPA)	0000000240	N.C.
	DELAYED	D27 -D	1312	OPA	CIT-OPS+36						
Divide (A) by (OPA-(I)).	NORMAL INDEXED	D27 -NI D27 -I	0313	OPA	CIT-CIS+37	YES	B, C, D	(A)/(OPA-(I)) → A REMAINDER → B (OPA-(I))$_{16-1}$ → C	A	0000000144	0000000355
									B	-	001 0000000
									C	-	000000
									(OPA-(I))	0330000000	N.C.
	DELAYED INDEXED	D27 -DI	1313	OPA	CIT-OPS+36						
Divide (A) by OPRND$_{16-1}$.	OPERAND	D27 -O	0712	OPRND	CIT-CIS+37	YES	-	(A)/OPRND$_{16-1}$ → A REMAINDER → B OPRND$_{16-1}$ → C	A	0000000163	0056000000
									B	-	0000000000
									C	-	002400
									OPRND	002400	N.C.

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling: (A) at 2^R/(OPA) at 2^S = (A) at 2^{R-S}; Remainder = (B) at 2^S (scaled from B_{27}). (3) Correctness of quotient depends on ratio between absolute magnitudes of dividend and divisor:

(a) If (A)/(OPA < 1, quotient is correct as generated.

(b) Overflow is indicated whenever (A)/(OPA) $\geq$ 1.

(c) If 1 $\leq$ (A)/(OPA) < 2, quotient as generated is missing its most significant bit. Correct quotient can be reconstructed in this case, however, by shifting right one place (changes scaling) and replacing the missing bit in A_{27}: a "1" if A_S is "0"; a "0" if A_S is "1".

(d) If (A)/(OPA) $\geq$ 2, quotient as generated is meaningless and cannot be reconstructed.

DIVIDE 21

EXECUTION TIME— 25 WORD TIMES 3325 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Divide (A) by (OPA).	NORMAL	D21 -N D21	0212	OPA	CIT-CIS+31	YES	B, C, D	(A)/(OPA) → $A_{3,21-1}$ A_S → A_{27-22} REMAINDER → B (OPA)$_{16-1}$ → C	A	00000000133	0004431463
									B	-	0000000040
									C	-	000240
									(OPA)	0000000240	N.C.
	DELAYED	D21 -D	1212	OPA	CIT-OPS+30						
Divide (A) by (OPA-(I)).	NORMAL INDEXED	D21 -NI D21 -I	0213	OPA	CIT-CIS+31	YES	B, C, D	(A)/(OPA-(I)) → $A_{5,21-1}$ A_S → A_{27-22} REMAINDER → B (OPA-(I))$_{16-1}$ → C	A	0000007100	0000000133
									B	-	0100000000
									C	-	000 00
									(OPA-(I))	0500000000	N.C.
	DELAYED INDEXED	D21 -DI	1213	OPA	CIT-OPS+30						
Divide (A) by OPRND$_{16-1}$.	OPERAND	D21 -O	0612	OPRND	CIT-CIS+31	YES	-	(A)/OPRND$_{16-1}$ → $A_{5,21-1}$ A_S → A_{27-22} REMAINDER → B OPRND$_{16-1}$ → C	A	0000000710	0002663143
									B	-	0000001000
									C	-	002400
									OPRND	002400	N.C.

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling: (A) at 2^R/(OPA) at 2^S = (A) at 2^{R-S-6}; Remainder = (B) at 2^S (scaled from B_{27}). (3) Correctness of quotient depends on ratio between absolute magnitudes of dividend and divisor:

(a) If (A)/(OPA) < 1, quotient is correct as generated.

(b) Overflow is indicated whenever (A)/(OPA) $\geq$ 1.

(c) If 1 $\leq$ (A)/(OPA) < 2, quotient as generated is missing its most significant bit. Correct quotient can be reconstructed in this case, however, by replacing the missing bit in A_{22}: a "1" if A_S is "0"; a "0" if A_S is "1".

(d) If (A)/(OPA) $\geq$ 2, quotient as generated is meaningless and cannot be reconstructed.

DIVIDE 14

EXECUTION TIME— 18 WORD TIMES 2394 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Divide (A) by (OPA).	NORMAL	D14 -N D14	0112	OPA	CIT-CIS+22	YES	B, C, D	(A)/(OPA) → $A_{S,14-1}$ A_S → A_{27-15} REMAINDER → B $(OPA)_{16-1}$ → C	A	0000000133	0000022146
									B	–	0000000100
									C	–	000240
	DELAYED	D14 -D	1112	OPA	CIT-OPS+21				(OPA)	0000000240	N.C
Divide (A) by (OPA-(I)).	NORMAL INDEXED	D14 -NI D14 -I	0113	OPA	CIT-CIS+22	YES	B, C, D	(A)/(OPA-(I)) → $A_{S,14-1}$ A_S → A_{27-15} REMAINDER → B $(OPA-(I))_{16-1}$ → C	A	0000001770	0000000052
									B	–	0000400000
									C	–	000000
	DELAYED INDEXED	D14 -DI	1113	OPA	CIT-OPS+21				(OPA-(I))	0001400000	N.C.
Divide (A) by $OPRND_{16-1}$.	OPERAND	D14 -O	0512	OPRND	CIT-CIS+22	YES	-	(A)/$OPRND_{16-1}$ → $A_{S,14-1}$ A_S → A_{27-15} REMAINDER → B $OPRND_{16-1}$ → C	A	0000000133	0000022146
									B	–	0000000100
									C	–	000240
									OPRND	000240	N.C.

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling: (A) at 2^R/(OPA) at 2^S = (A) at 2^{R-S-13}; Remainder = (B) at 2^S (scaled from B_{27}). (3) Correctness of quotient depends on ratio between absolute magnitudes of dividend and divisor:

(a) If (A)/(OPA) < 1, quotient is correct as generated.

(b) Overflow is indicated whenever (A)/(OPA) ≥ 1.

(c) If 1 ≤ (A)/(OPA) < 2, quotient as generated is missing its most significant bit. Correct quotient can be reconstructed in this case, however, by replacing the missing bit in A_{15}: a "1" if A_S is "0"; a "0" if A_S is "1".

(d) If (A)/(OPA) ≥ 2, quotient as generated is meaningless and cannot be reconstructed.

DIVIDE 7

EXECUTION TIME— 11 WORD TIMES 1463 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Divide (A) by (OPA).	NORMAL	D7 -N D7	0012	OPA	CIT-CIS+13	YES	B, C, D	(A)/(OPA) → $A_{S,7-1}$ A_S → A_{27-8} REMAINDER → B $(OPA)_{16-1}$ → C	A	0000000133	0000000110
									B	–	0000000200
									C	–	000240
	DELAYED	D7 -D	1012	OPA	CIT-OPS+12				(OPA)	0000000240	N.C.
Divide (A) by (OPA-(I)).	NORMAL INDEXED	D7 -NI D7 -I	0013	OPA	CIT-CIS+13	YES	B, C, D	(A)/(OPA-(I)) → $A_{S,7-1}$ A_S → A_{27-8} REMAINDER → B $(OPA-(I))_{16-1}$ → C	A	1777777645	1777777670
									B	–	1777777600
									C	–	000240
	DELAYED INDEXED	D7 -DI	1013	OPA	CIT-OPS+12				(OPA-(I))	0000000240	N.C.
	OPERAND	D7 -O	0412	OPRND	CIT-CIS+13	YES	-	(A)/$OPRND_{16-1}$ → $A_{S,7-1}$ A_S → A_{27-8} REMAINDER → B $OPRND_{16-1}$ → C	A	0000000344	0000000026
									B	–	0000002000
									C	–	002400
									OPRND	002400	N.C

NOTES: (1) Registers: D, I, and X unchanged. (2) Scaling (A) at 2^R/(OPA) at 2^S = (A) at 2^{R-S-20}; remainder = (B) at 2^S scaled from B_{27}). (3) Correctness of quotient depends on ratio between absolute magnitudes of dividend and divisor:

(a) If (A)/(OPA) < 1, quotient is correct as generated.
(b) Overflow is indicated whenever (A)/(OPA) ≥ 1.
(c) If 1 ≤ (A)/(OPA) < 2, quotient as generated is missing its most significant bit. Correct quotient can be reconstructed in this case, however, by replacing the missing bit in A_8: a "1" if A_S is "0"; a "0" if A_S is "1".
(d) If (A)/(OPA) ≥ 2, quotient as generated is meaningless and cannot be reconstructed.

SQUARE ROOT

EXECUTION TIME— 16 WORD TIMES 2128 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Take square root of $(OPA)_{27-2}$.	NORMAL	SQRT -N SQRT	0006	OPA	CIT-CIS+20	No	B, C, D	$\sqrt{(OPA)_{27-2}} \rightarrow A_{27-15} \rightarrow C_{13-1}$ REMAINDER $\rightarrow B_{27-2}$ $0 \rightarrow A_{S,14-1} \rightarrow B_{S,1} \rightarrow C_{16-14}$	A	—	0000740000
									B	—	0000000000
									C	—	017000
									(OPA)	0000000341	N.C.
	DELAYED	SQRT -D	1006	OPA	CIT-OPS+17						
Take square root of $(OPA-(I))_{27-2}$.	NORMAL INDEXED	SQRT -NI SQRT -I	0007	OPA	CIT-CIS+20	No	B, C, D	$\sqrt{(OPA-(I))_{27-2}} \rightarrow A_{27-15} \rightarrow C_{13-1}$ REMAINDER $\rightarrow B_{27-2}$ $0 \rightarrow A_{S,14-1} \rightarrow B_{S,1} \rightarrow C_{16-14}$	A	—	0746210000
									B	—	0000054116
									C	—	017145
									(OPA-(I))	0716000000	N.C.
	DELAYED INDEXED	SQRT -DI	1007	OPA	CIT-OPS+17						
Take square root of $OPRND_{16-2}$.	OPERAND	SQRT -O	0406	OPRND	CIT-CIS+20	No	-	$\sqrt{OPRND_{16-2}} \rightarrow A_{22-15} \rightarrow C_{8-1}$ REMAINDER $\rightarrow B_{27-2}$ $0 \rightarrow A_{S,27-23,14-1} \rightarrow B_{S,1} \rightarrow C_{16-9}$	A	—	000 5500000
									B	—	0000000266
									C	—	000132
									OPRND	177776	N.C.

NOTES:

(1) Registers: D, I, and X unchanged.

(2) Scaling: If (OPA) at w^{2R}, then $\sqrt{(OPA)}$ = (A) at 2^{R} = (C) at 2^{R-14} (assuming C, scaled at 2^{-27} Remainder = (B) at 2^{2R}.

SHIFT A LEFT ARITHMETIC

EXECUTION TIME— WORD TIMES: SEE NOTE MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ left PPP places.	NORMAL	SLA -N SLA	0042	011-PPP	CIT-CIS+35	YES	-	$(A)_{27-1}$ ← SHIFT PPP PLACES 0's → ENTER A_1	A	0012300001	0230000100
										PPP = 006	0'S ON
	DELAYED	SLA -D	1042	011-PPP	CIT-CIS+PPP+1						

NOTES: (1) Registers: C, D, I, and X unchanged; for "Shift A" instructions, B unchanged. (2) Maximum PPP = 034. (3) In normal mode, execution time = 28 word times, or 3724 microseconds; in delayed mode, execution time= PPP + 1 word times, or 133 (PPP + 1) microseconds. (4) In delayed mode, if PPP = 0, NIA = CIT-CIS + 2; execution time = 2 word times, or 266 microseconds. (5) For left arithmetic shifts, overflow is indicated if any bit shifted out of $A_{27} \neq A_S$.

SHIFT A RIGHT ARITHMETIC

EXECUTION TIME— WORD TIMES: SEE NOTE MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ right PPP places.	NORMAL	SRA -N SRA	0042	010-PPP	CIT-CIS+35	NO	-	$(A)_{27-1}$ → SHIFT PPP PLACES A_S's → ENTER A_{27}	A	0123000177	0001230001
										PPP= 006	
	DELAYED	SRA -D	1042	010-PPP	CIT-CIS+PPP+1						

NOTES: (1) See "Shift A Left Arithmetic".

SHIFT A LEFT LOGICAL

EXECUTION TIME— WORD TIMES: SEE NOTE MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS REG.	BEFORE	AFTER
Shift (A) left PPP places.	NORMAL	SLL -N SLL	0042	007-PPP	CIT-CIS+35	NO	-	(A) ← SHIFT PPP PLACES 0's → ENTER A_1	A	1234500077	0450007700
										PPP=006	
	DELAYED	SLL -D	1042	007-PPP	CIT-CIS+PPP+1						

NOTES: (1) See "Shift A Left Arithmetic".

SHIFT A RIGHT LOGICAL

EXECUTION TIME— SEE NOTE WORD TIMES MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift (A) right PPP places.	NORMAL	SRL -N SRL	0042	006-PPP	CIT-CIS+35	No	-	(A) SHIFT → PPP PLACES 0's ENTER → A_5	A	1224500077	0012245000
										PPP = 006	
	DELAYED	SRL -D	1042	006-PPP	CIT-CIS+PPP+1						

NOTES: (1) See "Shift A Left Arithmetic".

SHIFT A LEFT LOGICAL CLOSED

EXECUTION TIME— SEE NOTE WORD TIMES MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift (A) left end-around PPP places.	NORMAL	SLC-N SLC	0042	005-PPP	CIT-CIS+35	No	-	(A) SHIFT ← PPP PLACES A_5's ENTER → A_1	A	0123000456	1000456051
										PPP = 011	
	DELAYED	SLC-D	1042	005-PPP	CIT-CIS+PPP+1						

NOTES: (1) See "Shift A Left Arithmetic".

SHIFT AB LEFT ARITHMETIC

EXECUTION TIME— SEE NOTE WORD TIMES MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ and $(B)_{27-1}$ left PPP places.	NORMAL	SBA -N SBA	0042	017-PPP	CIT-CIS+35	Yes	-	$(A)_{27-1}(B)_{27-1}$ SHIFT ← PPP PLACES 0's ENTER → B_1	A	0000012345	0012345234
									B	1234500777	1500777000
	DELAYED	SBA -D	1042	017-PPP	CIT-CIS-PPP+1					PPP = 011	

NOTES: See "Shift A Left Arithmetic"

SHIFT AB LEFT ARITHMETIC/LOGICAL

EXECUTION TIME— SEE NOTE WORD TIMES MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ and (B) left PPP places.	NORMAL	SBL -N SBL	0042	015-PPP	CIT-CIS+35	YES	-	$(A)_{27-1}$ (B) SHIFT PPP PLACES 0's ENTER B_1	A	00000 12345	0012345516
									B	12345 00001	0500001000
	DELAYED	SBL -D	1042	015-PPP	CIT-CIS+PPP+1					PPP = 011	

NOTES: (1) See "Shift A Left Arithmetic"

SHIFT AB RIGHT ARITHMETIC/LOGICAL

EXECUTION TIME— SEE NOTE WORD TIMES MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ and (B) right PPP places.	NORMAL	SBR -N SBR	0042	014-PPP	CIT-CIS+35	No	-	$(A)_{27-1}$ (B) SHIFT PPP PLACES A_S's ENTER A_{27}	A	1000001234	1777000001
									B	0123000777	0470123000
	DELAYED	SBR -D	1042	014-PPP	CIT-CIS+PPP+1					PPP = 011	

NOTES: (1) See "Shift A Left Arithmetic".

FLOAT SHIFT A LEFT ARITHMETIC

EXECUTION TIME— 28 WORD TIMES 3724 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ left until $A_{27} \neq A_S$, and count number of shifts.	NORMAL	SFLA	0442	011-NNN	CIT-CIS+35	No	-	$(A)_{27-1}$ SHIFT UNTIL $A_{27} \neq A_S$ $0_S \rightarrow A_1$ (X) - (SHIFTS) → X	A	0002340000	0470001000
									X	000	003

NOTES: (1) Registers: B, C, D, and I unchanged. (2) No shift if: $(A)_{27-1} = 0$; or $(A)_{27} \neq (A)_S$.

FLOAT SHIFT A LEFT LOGICAL

EXECUTION TIME— 28 WORD TIMES 3724 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift (A) left until $A_S = 1$, and count number of shifts.	NORMAL	SFLL	0442	007-NNN	CIT-CIS+35	No	—	(A) $\overset{\text{SHIFT UNTIL}}{\leftarrow}$ $A_S = 1$ 0's → A, (X) - (SHIFTS) → X	A	0010000000	1000000000
									X	000	372

NOTES: (1) Registers: B, C, D, and I unchanged. (2) No shift if: $(A)_S = 1$; or (A) = 0.

FLOAT SHIFT A LEFT LOGICAL CLOSED

EXECUTION TIME— 28 WORD TIMES 3724 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift (A) left end-around until $A_S = 1$ and count number of shifts.	NORMAL	SFLC	0442	005-NNN	CIT-CIS+35	No	—	(A) $\overset{\text{SHIFT UNTIL}}{\leftarrow}$ $A_S = 1$ $(A)_S \rightarrow A_1$ (X) - (SHIFTS) → X	A	0002340000	1160000000
									X	007	377

NOTES: (1) Registers: B, C, D, and I unchanged. (2) No shift if: $(A)_S = 1$; or (A) = 0.

FLOAT SHIFT AB LEFT ARITHMETIC

EXECUTION TIME— 28 WORD TIMES 3724 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ and $(B)_{27-1}$ left until $A_{27} \neq A_S$, and count number of shifts.	NORMAL	SFBA	0442	017-NNN	CIT-CIS+35	No	—	$(A)_{27-1}(B)_{27-1}$ $\overset{\text{SHIFT UNTIL}}{\leftarrow}$ $A_{27} \neq A_S$ 0's → B, (X) - (SHIFTS) → X	A	0002340000	0470000174
									B	1760000000	1000000000
									X	015	006

NOTES: (1) Registers: C, D, and I unchanged. (2) No shift if: $(A)_{27-1} = 0$; or $(A)_{27} \neq (A)_S$.

FLOAT SHIFT AB LEFT ARITHMETIC/LOGICAL

EXECUTION TIME— 28 WORD TIMES 3724 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	BEFORE	AFTER
Shift $(A)_{27-1}$ and (B) left until $A_{27} \neq A_S$, and count numbers of shifts	NORMAL	SFBL	0442	015-NNN	CIT-CIS+35	No	—	$(A)_{27-1}$ (B) $\overset{\text{SHIFT UNTIL}}{\leftarrow}$ $A_{27} \neq A_S$ 0's → B, (X) - (SHIFTS) → X	A	0002340000	0470000176
									B	1760000000	0000000000
									X	015	006

NOTES: (1) Registers: C, D, and I unchanged. (2) No shift if: $(A)_{27-1} = 0$; or $(A)_{27} \neq (A)_S$.

JUMP UNCONDITIONALLY

EXECUTION TIME— OPS-C1S WORD TIMES 133 (OPS-C1S) MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Jump to OPA.	NORMAL	JUN -N JUN	0000	OPA	OPA
Jump to OPA-(I).	NORMAL INDEXED	JUN -NI JUN -I	0001	OPA	OPA-(I)
Jump to address shown in C register.	OPERAND	JUN -O	0400	OPA	$C_{TTT-SSS}$

NOTES: (1) Registers: No change.

JUMP A ZERO

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
If (A) = 0, jump to OPA.	NORMAL	JZE -N JZE	0044	OPA	IF (A) ≠ 0, CIT-CIS+2 IF (A) = 0, OPA
If (A) = 0, jump to OPA-(I).	NORMAL INDEXED	JZE -NI JZE -I	0045	OPA	IF (A) ≠ 0, CIT-CIS+2 IF (A) = 0, OPA-(I)
If (A) = 0, jump to address shown in C register.	OPERAND	JZE -O	0444	OPA	IF (A) ≠ 0, CIT-CIS+2 IF (A) = 0, $C_{TTT-SSS}$

NOTES: (1) Registers: No change.

JUMP A NEGATIVE

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND ADDRESS	NEXT INSTRUCTION ADDRESS
If (A) < 0, jump to OPA.	NORMAL	JNG -N JNG	0050	OPA	IF (A) ≥ 0, CIT-CIS+2 IF (A) < 0, OPA
If (A) < 0, jump to OPA-(I).	NORMAL INDEXED	JNG -NI JNG -I	0051	OPA	IF (A) ≥ 0, CIT-CIS+2 IF (A) < 0, OPA-(I)
If (A) < 0, jump to address shown in C register.	OPERAND	JNG -O	0450	OPA	IF (A) ≥ 0, CIT-CIS+2 IF (A) < 0, C$_{TTT-SSS}$

NOTES: (1) Registers: No change.

JUMP A LOW BIT

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND ADDRESS	NEXT INSTRUCTION ADDRESS
If $(A)_1$ = 1, jump to OPA.	NORMAL	JLB -N JLB	1000	OPA	IF $(A)_1$ = 0, CIT-CIS+2 IF $(A)_1$ = 1, OPA
If $(A)_1$ = 1, jump to OPA-(I).	NORMAL INDEXED	JLB -NI JLB -I	1001	OPA	IF $(A)_1$ = 0, CIT-CIS+2 IF $(A)_1$ = 1, OPA-(I)
If $(A)_1$ = 1, jump to address shown in C register.	OPERAND	JLB -O	1400	OPA	IF $(A)_1$ = 0, CIT-CIS+2 IF $(A)_1$ = 1, C$_{TTT-SSS}$

NOTES: (1) Registers: No change.

JUMP INDEX

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND ADDRESS	NEXT INSTRUCTION ADDRESS
If (I) ≠ 0, jump to OPA.	NORMAL	JIN -N JIN	0014	OPA	IF (I) = 0, CIT-CIS+2 IF (I) ≠ 0, OPA
If (I) ≠ 0, jump to OPA-(I).	NORMAL INDEXED	JIN -NI JIN -I	0015	OPA	IF (I) = 0, CIT-CIS+2 IF (I) ≠ 0, OPA-(I)
If (I) ≠ 0, jump to address shown in C register.	OPERAND	JIN -O	0414	OPA	IF (I) = 0, CIT-CIS+2 IF (I) ≠ 0, C TTT-SSS

NOTES: (1) Registers: No change.

JUMP OVERFLOW

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND ADDRESS	NEXT INSTRUCTION ADDRESS
If overflow indicator is on, jump to OPA.	NORMAL	JOF -N JOF	0054	OPA	IF O'F OFF, CIT-CIS+2 IF O'F ON, OPA
If overflow indicator is on, jump to OPA-(I).	NORMAL INDEXED	JOF -NI JOF -I	0055	OPA	IF O'F OFF, CIT-CIS+2 IF O'F ON, OPA-(I)
If overflow indicator is on, jump to address shown in C register.	OPERAND	JOF -O	0454	OPA	IF O'F OFF, CIT-CIS+2 IF O'F ON, C TTT-SSS

NOTES: (1) Registers: No change.

(2) Overflow indicator turned off on jump.

JUMP PARITY ERROR

EXECUTION TIME-- 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND ADDRESS	NEXT INSTRUCTION ADDRESS
If parity error indicator is on, jump to OPA.	NORMAL	JPE -N JPE	0010	OPA	IF NO P.E., CIT-CIS+2 IF P.E., OPA
If parity error indicator is on, jump to OPA-(I).	NORMAL INDEXED	JPE -NI JPE -I	0011	OPA	IF NO P.E., CIT-CIS+2 IF P.E., OPA-(I)
If parity error indicator is on, jump to address shown in C register.	OPERAND	JPE-O	0410	OPA	IF NO P.E., CIT-CIS+2 IF P.E., $C_{TTT-SSS}$

NOTES: (1) Registers: No change.

(2) Parity error indicator turned off on jump.

STOP --JUMP RESUME

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND ADDRESS	NEXT INSTRUCTION ADDRESS
Stop. When RESUME button is pressed, jump to OPA.	NORMAL	STP -N STP	0040	OPA	OPA
Stop. When RESUME button is pressed, jump to OPA-(I).	NORMAL INDEXED	STP -NI STP -I	0041	OPA	OPA-(I)
Stop. When RESUME button is pressed, jump to address shown in C register.	OPERAND	STP -O	0440	OPA	$C_{TTT-SSS}$

NOTES: (1) Registers: No change.

JUMP RECORD ADDRESS 0

EXECUTION TIME— [redacted] WORD TIMES [redacted] MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Store re-entry address in "0" sector of fast-access track, and jump to OPA.	NORMAL	JRA0 -N JRA0	0256	OPA	OPA
Store re-entry address in "0" sector of fast-access track, and jump to OPA-(I).	NORMAL INDEXED	JRA0 -NI JRA0 -I	0257	OPA	OPA-(I)

NOTES: (1) Registers: No change. (2) Re-entry address -- CIT-CIS + 2 -- is automatically stored in one of the fast-access tracks; track and sector selected is determined automatically by sector in which current instruction appears. Jump can be made into any sector of any track, but for each fast-access track there is an optimum sector for machine running time.

If CIS is	CIT-CIS+2 stored in	Optimum jump sector
177-036	060-040	041
037-076	061-100	101
077-136	062-140	141
137-176	063-000	001

(3) Note that operation code stored with re-entry address is 0000; thus the instruction stored is an unconditional jump.

JUMP RECORD ADDRESS 1

EXECUTION TIME— [redacted] WORD TIMES [redacted] MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Store re-entry address in "1" sector of fast-access track, and jump to OPA.	NORMAL	JRA1 -N JRA1	0356	OPA	OPA
Store re-entry address in "1" sector of fast-access track, and jump to OPA-(I).	NORMAL INDEXED	JRA1 -NI JRA1 -I	0357	OPA	OPA-(I)

NOTES: (1) Registers: No change. (2) Re-entry address -- CIT-CIS+2 -- is automatically stored in one of the fast access tracks; track and sector selected is determined automatically by sector in which current instruction appears. Jump can be made into any sector of any track, but for each fast-access track there is an optimum sector for machine running time.

If CIS is	CIT-CIS+2 stored in	Optimum jump sector
177-036	060-041	042
037-076	061-101	102
077-136	062-141	142
137-176	063-001	002

(3) Note that operation code stored with re-entry address is 0000; thus the instruction stored is an unconditional jump.

JUMP RECORD ADDRESS C

EXECUTION TIME— [redacted] WORD TIMES [redacted] MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Store re-entry address in C register, and jump to OPA.	NORMAL	JRAC -N JRAC	0056	OPA	OPA
Store re-entry address in C register, and jump to (OPA-(I)).	NORMAL INDEXED	JRAC -NI JRAC -I	0057	OPA	OPA-(I)

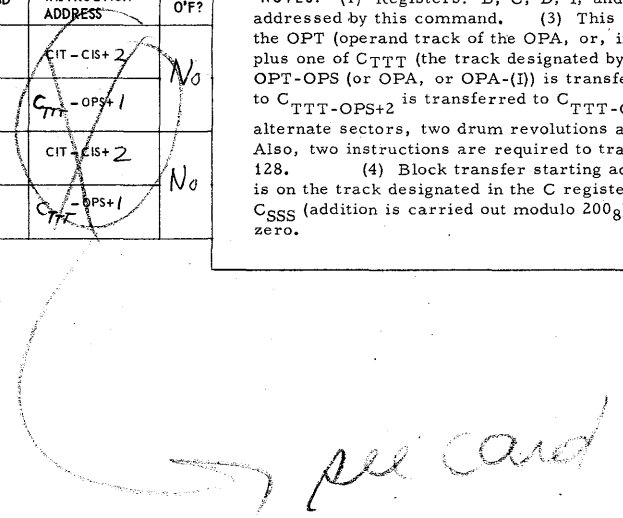
NOTES: (1) Registers: No change.

(2) Note that operation code for an instruction stored in C register (in which operation code bits are missing) is read as 0000; thus the instruction stored is an unconditional jump.

BLOCK TRANSFER

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS	CAN O'F?
			OP CODE	OPERAND		
Transfer information to specified track.	NORMAL	BT -N BT	0074	OPA	CIT - CIS+2	No
	DELAYED	BT -D	1074	OPA	C_{TTT} - OPS+1	
Transfer information to specified track.	NORMAL INDEXED	BT -NI BT -I	0075	OPA	CIT - CIS+2	No
	DELAYED INDEXED	BT -DI	1075	OPA	C_{TTT} - OPS+1	

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) The D register may be addressed by this command. (3) This instruction transfers the contents of sectors of the OPT (operand track of the OPA, or, if indexed, of OPA-(I)) with corresponding sectors plus one of C_{TTT} (the track designated by the track field of the C register). Thus, OPT-OPS (or OPA, or OPA-(I)) is transferred to $C_{TTT-OPS+1}$, OPT-OPS+2 is transferred to $C_{TTT-OPS+2}$ is transferred to $C_{TTT-OPS+3}$, and so on. Since transferring is by alternate sectors, two drum revolutions are required for transfer of an entire track. Also, two instructions are required to transfer a group of consecutive sectors less than 128. (4) Block transfer starting address is OPA (or OPA-(I). The stopping address is on the track designated in the C register, at the sector equal to the sum of OPS and C_{SSS} (addition is carried out modulo 200_8). A full track is transferred if C_{SSS} equals zero.



LESS

SCAN TABLE ~~GREATER~~ THAN

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Scan for masked bits on OPT ~~greater~~ less than reference bits.	NORMAL	SCNG	0162	OPA	UNSUCCESSFUL, CIT-000 SUCCESSFUL, CIT-001

GREATER

SCAN TABLE ~~LESS~~ THAN

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Scan for masked bits on OPT ~~less~~ greater than reference bits.	NORMAL	SCNL	0262	OPA	UNSUCCESSFUL, CIT-000 SUCCESSFUL, CIT-001

SCAN TABLE EQUAL

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Scan for masked bits on OPT equal to reference bits.	NORMAL	SCNE	0062	OPA	UNSUCCESSFUL, CIT-000 SUCCESSFUL, CIT-001

SCAN TABLE NOT EQUAL

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Scan for masked bits on OPT not equal to reference bits.	NORMAL	SCNN	0362	OPA	UNSUCCESSFUL, CIT-000 SUCCESSFUL, CIT-001

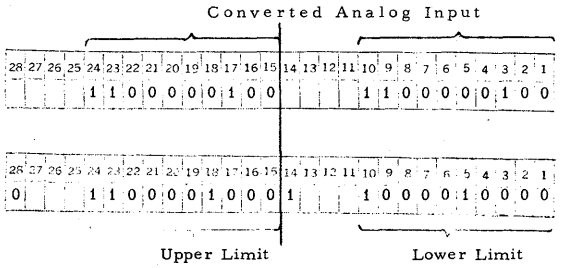
NOTES: (1) Registers: B, C, D, and I unchanged. (2) The D register may be addressed by this command. (3) These instructions compare the contents of sectors of the OPT (operand track of the OPA) with corresponding sectors plus one of C_{TTT} (the track designated by the track field of the C register). Thus, OPT-OPS (or OPA) is compared with $C_{TTT\text{-}OPS+1}$, OPT-OPS+2 is compared with $C_{TTT\text{-}OPS+3}$, and so on. Since comparison is by alternate sectors, two drum revolutions are required for comparison of an entire track. Also, two instructions are required to compare a group of consecutive sectors less than 128. (4) The contents of the B register are used as a mask. Only those bit positions are compared in which there is a 1-bit in the corresponding bit position in B. All bits will be compared if B contains 1777777777. (5) Scan starting address is OPA. The stopping address, for an "unsuccessful" scan, is on the track designated in the C register, at the sector equal to the sum of OPS and C_{SSS} (addition is carried out modulo 200_8). A full track is compared if C_{SSS} equals zero.

(6) Scan starts at starting address and continues until: (a) an "unsuccessful" scan is completed, i.e., the stopping address is reached without finding the condition scanned for; or (b) a "successful" scan is completed, i.e., the condition scanned for is found. (7) At the end of an "unsuccessful" scan, the A register contains the contents of the stopping sector on the track designated in the C register. NIA of an "unsuccessful" scan is CIT-000. (8) At the end of a "successful" scan, the A register contains the contents of the last compared sector on the track designated in the C register; the X register contains the address of the last compared sector. NIA of a "successful" scan is CIT-001.

SCAN ANALOG

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Scan analog inputs for hi-low limits.	NORMAL	SCNA	0462	OPA	UNSUCCESSFUL, CIT-000 SUCCESSFUL, CIT-001

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) The B or D registers may be addressed by this command. (3) This instruction compares the contents of sectors of the OPT (operand track of the OPA) with corresponding sectors plus one of C_{TTT} (the track designated by the track field of the C register). Thus, OPT-OPS (or OPA) is compared with $C_{TTT\text{-}OPS+1}$, OPT-OPS+2 is compared with $C_{TTT\text{-}OPS+3}$, and so on. Since comparison is by alternate sectors, two drum revolutions are required for comparison of an entire track. Also, two instructions are required to compare a group of consecutive sectors less than 128. (4) When analog signals are converted to digital form and recorded on the drum, the binary data is written twice in the appropriate analog input sector, first in bits 1 through 10, then again in bits 15 through 24. In the corresponding reference sector, the upper and lower limits of this analog input value are written in corresponding bit positions; a 1-bit in bit position 14 or 28 specifies that half of the reference word is the lower limit, while a 0-bit in 14 or 28 specifies that half as the upper limit. In the example shown here, Input Lower Limit Reference, and Input Upper Limit Reference is an "unsuccessful" scan, which means the input is within limits. (5) Scan starting address is OPA. The stopping address, for an "unsuccessful" scan, is on the track designated in the C register, at the sector equal to the sum of OPS and C_{SSS} (addition is carried out modulo 200_8). A full track is compared if C_{SSS} equals zero. (6) Scan starts at starting address and continues until: (a) an "unsuccessful" scan is completed, i.e., the stopping address is reached without finding an input that has exceeded limits; or (b) a "successful" scan is completed, i.e., an input outside limits has been found. (7) At the end of an "unsuccessful" scan, the A register contains the contents of the stopping sector on the track designated in the C register. NIA of an "unsuccessful" scan is CIT-000. (8) At the end of a "successful" scan, the A register contains the contents of the last compared sector on the track designated in the C register; A_{28} specifies whether "successful" comparison was made on lower or upper half of word, with a 1-bit in A_{28} indicating upper half. NIA of a "successful" scan is CIT-001.



SCAN DIGITAL INPUT

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION		NEXT INSTRUCTION ADDRESS
			OP CODE	OPERAND ADDRESS	
Scan digital input group bits for inequality with reference bits.	NORMAL	SCND	1062	OPA	UNSUCCESSFUL, CIT-000 SUCCESSFUL, CIT-001

NOTES: (1) Registers: C, D, and I unchanged. (2) The D register may be addressed by this command. (3) This instruction compares the contents of digital input groups (up to 28 input lines per group) with reference sectors of C_{TTT} (the track designated by the track field of the C register). Digital input groups are taken successively, starting with that group having the same group number as OPT (operand track address of the OPA); the sectors of C_{TTT} are alternate sectors starting with OPS+1 (where OPS is the sector address of the OPA). Thus, digital input group OPT is compared with $C_{TTT\text{-}OPS+1}$, digital input group OPT+1 is compared with $C_{TTT\text{-}OPS+3}$, and so on. (4) Scan starting address is digital input group OPT. The stopping address, for an "unsuccessful" scan, is on the track designated in the C register, at the sector equal to the sum of OPS and C_{SSS} (addition is carried out modulo 200_8). (5) Scan starts at starting address and continues until: (a) an "unsuccessful" scan is completed, i.e., an inequality is found. (6) At the end of an "unsuccessful" scan, the A register contains the contents of the stopping sector on the track designated in the C register. NIA of an "unsuccessful" scan is CIT-000. (7) At the end of a "successful" scan, the A register contains the contents of the last compared sector on the track designated in the C register; the X register contains the address of the last compared sector; the B register contains 1-bits in those bit positions where the compared digital inputs and the bits in the comparing sector are not equal. NIA of a "successful" scan is CIT-001.

INHIBIT INTERRUPT ON

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	REGISTER STATUS: BEFORE	REGISTER STATUS: AFTER
Turn on interrupt-inhibit flip-flop.	NORMAL	ION	0100	OPA	OPA	No	-	Set "Interrupt Inhibit" flip-flop on--preventing interrupts--and jump to OPA.	(OPA)	N. C.	

INHIBIT INTERRUPT OFF

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS	CAN O'F?	ADDRESSABLE REG'S.	EFFECT OF OPERATION	REGISTER STATUS: REG.	REGISTER STATUS: BEFORE	REGISTER STATUS: AFTER
Turn off interrupt-inhibit flip-flop.	NORMAL	IOF	0200	OPA	OPA	No	-	Set "Interrupt-Inhibit" flip-flop off--enabling interrupts--and jump to OPA.	(OPA)	N. C.	

SET ANALOG SEQUENCE

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Reset sequence counter.	NORMAL	SAS -N SAS	0042	032-NNN	CIT-CIS+2
	DELAYED	SAS -D	1042	032-NNN	CIT-OPS+1

NOTES: (1) Registers: No Change. (2) This command resets sequence counter to sector NNN which directs the analog input system to interrupt its reading sequence and begin reading analog inputs according to control track locations 064-NNN, 064-NNN+2, etc. (where track 064 is the analog input control track).

ACTIVATE CONTROL SIGNAL

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Emit a 130-microsecond pulse.	NORMAL	ACS -N ACS	0042	022-NNN	CIT-CIS+2
	DELAYED	ACS -D	1042	022-OPS	CIT-OPS+1

NOTES: (1) Registers: No Change.

DIGITAL INPUT: FLEXOWRITER

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Read in one tape frame from Flexowriter.	NORMAL	DIF -N DIF	0052	000-NNN	CIT-CIS+2
	DELAYED	DIF -D	1052	000-OPS	CIT-OPS+1
	OPERAND	DIF -O	0452	000-NNN	CIT-CIS+2

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) Each tape frame contains six bits, or channels, of information. Channels containing punches are read as 1-bits. No-punch is read as a 0-bit. (3) Tape frame reads into A_{6-1}. In Normal and Delayed, 0's read into unused bits of A; in Operand, unused bits of A recirculate. (4) Flex punch and print must be off during DIF command.

DIGITAL INPUT: TOGGLE SWITCHES

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Read in toggle switch settings from computer control panel.	NORMAL	DIT -N DIT	0052	001-NNN	CIT-CIS+2
	DELAYED	DIT -D	1052	001-OPS	CIT-OPS+1
	OPERAND	DIT -O	0452	001-NNN	CIT-CIS+2

NOTES: (1) Registers: B, D, I, and X unchanged. (2) Toggles in the "up" position are read as 1-bits. (3) Toggle switches T18-T1 read into A_{18-1}. In Normal and Delayed, 0's read into unused bits of A; in Operand, unused bits of A recirculate.

DIGITAL INPUT: TELETYPE READER

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Read in one character from teletype reader.	NORMAL	DIX -N DIX	0052	002-NNN	CIT-CIS+2
	DELAYED	DIX -D	1052	002-OPS	CIT-OPS+1
	OPERAND	DIX -O	0452	002-NNN	CIT-CIS+2

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) 5-bit teletype reader character reads into A_{5-1}. In Normal and Delayed, 0's read into unused bits of A; in Operand, unused bits of A recirculate.

DIGITAL INPUT: HIGH-SPEED TAPE READER

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Read in one tape frame from high-speed tape reader.	NORMAL	DIR -N DIR	0052	003-NNN	CIT-CIS+2
	DELAYED	DIR -D	1052	003-OPS	CIT-OPS+1
	OPERAND	DIR -O	0452	003-NNN	CIT-CIS+2

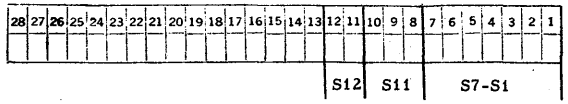
NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) Each tape frame contains six bits, or channels, of information. Channels containing punches are read as 1-bits. No-punch is read as a 0-bit. (3) 6-bit tape frame reads into A_{6-1}. In Normal and Delayed, 0's read into unused bits of A; in Operand, unused bits of A recirculate.

DIGITAL INPUT: CONSOLE SWITCHES

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Read in console switch positions.	NORMAL	DIS -N DIS	0052	004-NNN	CIT-CIS+2
	DELAYED	DIS -D	1052	004-OPS	CIT-OPS+1
	OPERAND	DIS -O	0452	004-NNN	CIT-CIS+2

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) These switches are the seven one-bit selector switches, S1-S7, and the two octal function select switches, S11 and S12, used to reference the function matrix. (3) Switch S12 reads into A_{13-11}, S11 into A_{10-8}, and S7-S1 into A_{7-1}. In Normal and Delayed, 0's enter $A_{S,\ 27-14}$; in Operand, $(A)_{S,\ 27-14}$ recirculates.



DIGITAL INPUT: DIGITRAN SWITCHES

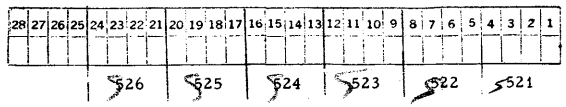
EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Read in BCD information from six digitran switches.	NORMAL	DID -N DID	0052	006-NNN	CIT-CIS+2
	DELAYED	DID -D	1052	006-OPS	CIT-OPS+1
	OPERAND	DID -O	0452	006-NNN	CIT-CIS+2

NOTES:

(1) Registers: B, C, D, I, and X unchanged.

(2) BCD information from digitran switches S26-S21 is read into A_{24-1}. In Normal and Delayed, 0's enter the unused bits of A; in Operand mode, the unused bits of A recirculate.

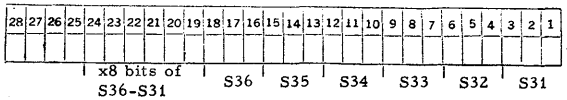


DIGITAL INPUT: DIGITRAN SWITCHES

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Read in split BCD information from digitran switches.	NORMAL	DID -N DID	0052	005-NNN	CIT-CIS+2
	DELAYED	DID -D	1052	005-OPS	CIT-OPS+1
	OPERAND	DID -O	0452	005-NNN	CIT-CIS+2

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) Switches S36-S31 are BCD switches. In reading into the A register, however, the X8 bit of each switch is read in separately from the X4, X2, and X1 bits. The latter three bits are read into A as octal numbers. (3) The X8 bits of S36-S31 read into A_{24-19}; the X4-X2-X1 bits are read in octally as follows: S36 into A_{18-16}, S35 into A_{15-13}, S34 into A_{12-10}, S33 into A_{9-7}, S32 into A_{6-4}, and S31 into A_{3-1}. In Normal and Delayed, 0's enter the unused bits of A; in Operand, the unused bits of A recirculate.

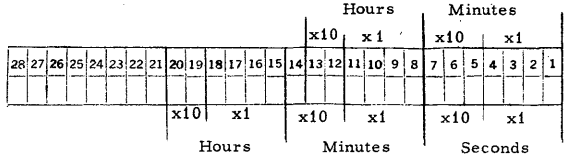


DIGITAL INPUT: DIGITAL CLOCK

EXECUTION TIME— 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Read in BCD information from digital clock.	NORMAL	DIC -N DIC	0052	007-NNN	CIT-CIS+2
	DELAYED	DIC -D	1052	007-OPS	CIT-OPS+1
	OPERAND	DIC -O	0452	007-NNN	CIT-CIS+2

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) Clock records in Navy time notation: 0000-2359. It can read in hours/minutes or hours/minutes/seconds. (3) BCD information from clock is read into A_{13-1} (hours/minutes clock) or into A_{20-1} (hours/minutes/seconds clock). In Normal and Delayed, 0's enter unused bits of A; in Operand, unused bits of A recirculate.



DIGITAL INPUT: GROUP INPUTS

EXECUTION TIME-- 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Read in selected digital input group.	NORMAL	DIN -N DIN	0052	010-NNN	CIT-CIS+ 2
	DELAYED	DIN -D	1052	010-OPS	CIT-OPS+ 1

NOTES: (1) Registers: B, C, D, I, and X unchanged. (2) Operand track addresses from 010 up designate specific groups of 28 digital input lines. The numbers and functions of these inputs are determined by the particular installation. (3) The 28-bit group reads into the A register.

DIGITAL OUTPUT: FLEXOWRITER

EXECUTION TIME-- 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Output one frame to Flexowriter.	NORMAL	OUF -N OUF	0046	000-NNN	CIT-CIS+ 2
	DELAYED	OUF -D	1046	000-OPS	CIT-OPS+ 1

NOTES: (1) Registers: No Change. (2) Output to Flex printer or punch from $(A)_{6-1}$. (3) An automatic 110-millisecond delay occurs between successive OUF commands.

DIGITAL OUTPUT: OUTPUT BUFFER

EXECUTION TIME-- 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Start digital output buffer.	NORMAL	OUB -N OUB	0046	001-OPS	CIT-CIS+ 2
	DELAYED	OUB -D	1046	001-OPS	CIT-OPS+ 1

NOTES: (1) Registers: No Change. (2) This command directs the digital output buffer to begin reading out from consecutive sectors beginning at sector OPS. (3) The output device itself must previously have been selected by an OUSN command; if not, the buffer will attempt an output. (4) After the OUB command has been given, the computer is free to continue executing its program. (5) Characters in the buffer track are contained in bits 24 through 1. (6) Output buffer information is decoded as either four 6-bit characters or six 4-bit characters, depending on the output device selected. (Flexowriter and IBM logging typewriter are 6-bit character devices.) (7) Words are unpacked and read out from high-order bits first. (8) A full or partial track can be read out; the end-of-message is flagged for the output buffer by setting a 1-bit in the sign position of the final message word. (9) More than one of the same type of output device may be read to simultaneously.

DIGITAL OUTPUT: SINGLE-BIT OUTPUTS ON

EXECUTION TIME-- 2 WORD TIMES 2.66 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION: OP CODE	INSTRUCTION: OPERAND	NEXT INSTRUCTION ADDRESS
Turn on designated devices.	NORMAL	OUSN -N OUSN	0246	002-NNN	CIT-CIS+ 2
	DELAYED	OUSN -D	1246	002-OPS	CIT-OPS+ 1

NOTES: (1) Registers: No Change. (2) Simultaneously, each 1-bit in A turns on selected device(s): A_1, Flex motor; A_2, Flex punch; A_3, Flex printer; A_4, Select Flex for OUB; etc.

DIGITAL OUTPUT: SINGLE-BIT OUTPUTS OFF

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Turn off designated devices.	NORMAL	OUSF-N OUSF	0046	002-NNN	CIT-CIS+2
	DELAYED	OUSF-D	1046	002-OPS	CIT-OPS+1

NOTES: (1) Registers: No Change. (2) Simultaneously, each 1-bit in A turns off designated device(s): A_1, Flex motor; A_2, Flex punch; A_3, Flex print; A_4, Select Flex for OUB; etc.

DIGITAL OUTPUT: HIGH-SPEED TAPE PUNCH

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Output one frame to high-speed punch.	NORMAL	OUP-N OUP	0046	003-NNN	CIT-CIS+2
	DELAYED	OUP-D	1046	003-OPS	CIT-OPS+1

NOTES: (1) Registers: No Change. (2) 6-channel frame directed to high-speed punch from $(A)_{6-1}$.

DIGITAL OUTPUT: MULTIBIT OUTPUT

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Output BCD characters to display.	NORMAL	OUM-N OUM	0046	004-NNN	CIT-CIS+2
	DELAYED	OUM-D	1046	004-OPS	CIT-OPS+1

NOTES: (1) Registers: No Change. (2) This command, OUM track 4, directs output to the visual display on the operator's console. (3) BCD information from A_{20-1} directed to display: A_{20-17} to character D5; A_{16-13} to D4; A_{12-9} to D3; A_{8-5} to D2; and A_{4-1} to D1. (4) Additional OUM tracks are available as options; when other tracks are used, 1-bits from the A register turn devices on and 0-bits turn devices off.

DIGITAL OUTPUT: LOGGING TYPEWRITER

EXECUTION TIME-- 2 WORD TIMES 266 MICROSECONDS

OPERATION	MODE	SYMBOLIC OPERATION CODE	INSTRUCTION OP CODE	INSTRUCTION OPERAND	NEXT INSTRUCTION ADDRESS
Output one character to logging typewriter.	NORMAL	OUL-N OUL	0046	005-NNN	CIT-CIS+2
	DELAYED	OUL-D	1046	005-OPS	CIT-OPS+1

NOTES: (1) Registers: No Change. (2) A_{6-1} is decoded and printed on logging typewriter.

CONVENTIONS, ABBREVIATIONS, AND SYMBOLS

A	Working Register A, or Accumulator
B	Working Register B
C	Working Register C
D	Working Register D
I	Working Register I, or Index
X	Working Register X
M	Interrupt Register M (M1, M2, M3, and M4)
Q	Interrupt Register Q (Q1, Q2, Q3, and Q4)
-N	Normal Mode
-NI	Normal Mode, Indexed
-D	Delayed Mode
-DI	Delayed Mode, Indexed
-O	Operand Mode
-U	Upper Fill, version of Operand Mode
SYMB	Symbolic
NNN	Numbers not effective in instruction, except in Set Analog Sequence

A_7-1	Bits 7-1 of A register
A_S	Sign bit of A register, or A_{28}
$B_{S, 14-1}$	Sign bit (B_{28}) and bits 14-1 of B register
NIA	Next Instruction Address
CIT	Current Instruction Track
CIS	Current Instruction Sector
CIA	Current Instruction Address, equals CIT-CIS
OPT	Operand Track
OPS	Operand Sector
OPA	Operand Address, equals OPT-OPS
C_{TTT}	Track Field of C Register, equals C_{16-8}
C_{SSS}	Sector Field of C Register, equals C_{7-1}
PPP	Number of Places Shifted
OPRND	Number contained in bits 16-1 (or 14-1) of an instruction

NOTE that the track address for the next instruction can be changed only by Jump Instructions, or by Inhibit Interrupt On or Off Instructions.

NOTE that operation codes, operand addresses, and next instruction addresses are given in octal notation.

EXAMPLES: If NIA equals CIT-CIS+2, computer will find its next instruction in the current instruction track, at the sector numbered two greater than the current sector; thus, if CIA (CIT-CIS) equals 042-016, NIA equals 042-020. If NIA equals CIT-OPS+1, and CIT equals 063 and OPS equals 177, then NIA equals 063-000 (note that track address is not incremented; sectors are added modulo 200_8).