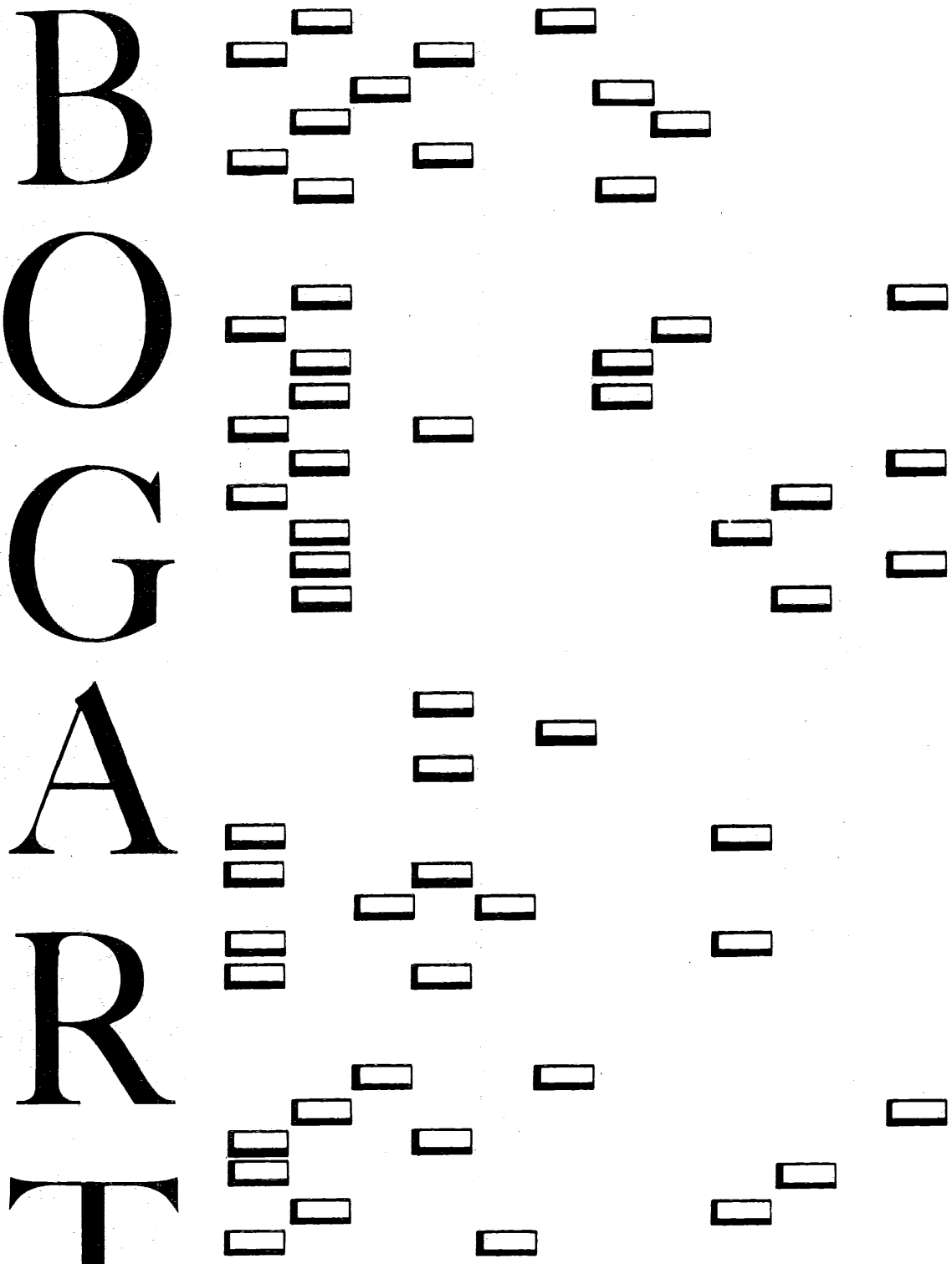
BOGART



PROGRAMMER MANUAL

PROGRAMMER MANUAL

BOGART - AFSAF D131

Task Number 352-7011

NATIONAL SECURITY AGENCY
Washington, 25, D. C.
July 1957

PROGRAMMER MANUAL

BOGART - AFSAF D131

Task Number 352-7011

Compiled by: Roy A. Carter
Technical Writer ENGR-31

Reviewed by: James A. Pederson
Project Analyst ANEQ-1

Vera S. Russell
Project Analyst ENGR-13

Approved by: Harry Lofink
Chief Publications Branch, ENGR-31

S-87,579

CONTENTS

CONTENTS

CONTENTS

BOGART

PROGRAMMERS MANUAL

SECTION I

INTRODUCTION

A. Scope of Manual

This manual is a guide for BOGART programmers and operators. It may also be used as a training supplement for new persons in the computer field.

The properties and description of internal registers are presented to give the reader a basic knowledge of the logical design of BOGART. The relationship between the computer and peripheral equipment is detailed to demonstrate the technical aspects of computer applications. The last section describes the operating procedures of BOGART.

B. General Description

BOGART is a general-purpose computer using a modified one-address instruction code. It has a 4096-word magnetic-core storage. Special commands enable BOGART to operate on a 24-bit word, as a whole, or any 8-bit third of a word. The functions of this computer include manipulating data, differentiating various forms of data, performing analytical operations, counting operations, and arithmetic operations. The specific operations are determined by a program.

The computation speed of BOGART is comparable to that of ATLAS and ABNER. Physically, however, BOGART is much smaller than either of these computers. The main cabinet is 8 feet long, 3 feet wide, and 5 feet high. External equipment provides input and output. The inputs are punched paper tape, Flexowriter, punched cards, and magnetic tape. Output is available through the same media as input with the addition of a line printer. BOGART also has an external motor-alternator unit which supplies 208-volt, 3-phase, 400-cycle alternating current. This is the main power source for the computer.

BOGART

PROGRAMMERS MANUAL

SECTION II

INTERNAL REGISTERS

A. Storage System

The basic unit of information in BOGART is the 24-bit binary word. It may be interpreted by the various registers of the machine as a positive or negative number, an instruction, a constant, or as three units of 8-bit data. This information may be stored in the storage section for any length of time and recalled or replaced when desired by the program. The storage section consists of the core storage, the Z-register, and the S-register.

1. Core Storage

This storage can retain 4096 24-bit words, a total of 98,304 bits of information.

The magnetic-core storage contains 24 planes. Each 10-inch plane is a 64 by 64 square array of 4096 cores. Three of the four wires running through each core write information into the core and the fourth wire senses whether a 1 or a 0 is stored in the core.

If the core is magnetized in one direction, a 1 is stored in it; and if it is magnetized in the reverse direction, a 0 is stored in it. Information read out by applying a current to the two read-out wires drives the core into its zero state. If a 1 is stored in the core, the magnetic direction-change induces a voltage in the read-out wire. By sensing the voltage on the read-out wire, it is possible to determine the previous magnetic state of the core.

Since the read-out wire sets a core to the zero state, it is necessary for the memory circuits to restore the core to a one-state. If it contains a 1, this is the regeneration portion of the storage cycle. The total storage cycle of read-out and information regeneration takes 20 microseconds per word. The information

stored in any location is available to the Z-register in approximately 8 microseconds of the cycle. While the storage control is restoring the information in the core, the computer takes this information and processes it in the machine register.

Each plane in the magnetic-core storage corresponds to one bit of a word in the memory; and the 24 planes store a complete word. Each core in the plane corresponds to one of the 4096 storage locations. To select a word in the storage, the storage control accepts the address stored in the S-register. The interpreted address supplies one of the 64 horizontal read-out windings and one of the 64 vertical read-out windings of the storage matrix with half of the current necessary to change the magnetic state of a core. Only the core at the intersection of the two windings receiving full current, switches. This action induces a read-out voltage on the read-out winding of each core-storage plane. All other cores receive a half current. These cores do not switch and are not read out.

Information is stored in the core storage by the same procedure as in read-out. When the information in a set of cores has been read out, the information is not used for regeneration, but the core is used for new information.

2. Z-Register

This register is the communication link between the magnetic-core storage and the internal registers of the computer. The Z-register, a 24-bit register, receives information that is read from the magnetic cores and holds the information for regeneration. The Z-register also receives information from the X- and P-registers and holds it to be stored in the magnetic cores.

3. S-Register

This register is the storage address register for the core storage. The S-register retains the address of the storage location that is to be referred to by the program. It specifies which core-storage location is to be read into the Z-register and at which location the information contained in the Z-register is to be stored in the core storage.

The S-register receives address information from the P-register and the K-portion of the U-register.

B. Arithmetic Unit

This unit consists of the exchange or X-register, the accumulator or A-register, and the auxiliary arithmetic or Q-register. By the appropriate operation of these registers it is possible to perform addition, subtraction, multiplication, and division. It is also possible for the control to make decisions based on the contents of the A- or the Q-registers. The bit positions of all registers are numbered 23 to 0, from left to right. Arithmetic is performed on the one's-complement arithmetic system. Negative numbers are represented by the one's complement of the positive number. Each digit of the number is reversed to find its negative. Thus, for an eight-bit word, plus five is 00000101, minus five is 11111010.

1. X-Register

This register is a 24-bit register with several special exchange functions. It transfers information between the A-register, Q-register, U-register, I-register, and O-register. The X-register, the major exchange unit of the arithmetic unit, has the following properties:

a. The complement of the contents of the X-register may be transferred to A or Q. In complementing, the value of each bit of the register is reversed. A one becomes a zero and a zero becomes a one. This form of complementing enables the X-register to change the sign of a number, therefore subtraction is essentially a case of complementing and adding.

b. The X-register may receive the logical product of the Q- and X-registers. In the logical product, the X-register contains a one in each bit position of the Q-register which contained a one.

c. An 8-bit number may be extended to become a signed 24-bit number. This is a function of the transmission from Z to X. The repositioning of a number while it is being transmitted from Z to X causes an 8-bit number to be stored in bits 7 to 0 of the X-register. During the transmission of the 8-bit number, the left-most bit is examined and if it is a one, bits 23 to 8 of X are made all ones. If the left-most bit is a zero, bits 23 to 8 of X are left in the zero state.

d. The X-register receives a full 24-bit word from the A-, the Q-, or the Z-registers. The X-register can transmit a full 24-bit word to the A-, the Q-, or the Z-registers.

e. The X-register may receive a 7-bit character from the I-register (input) for storage in the core storage. The X-register may send 7-bit characters (6 to 0) to the O-register (output) for operation of the output equipment. A portion of a word may be transmitted to or from bits 7 - 0, or 14 - 0 of the X-register as follows:

X-Register Bits	To or From Register	
7 - 0	7 - 0	Z
7 - 0	15 - 8	Z
7 - 0	23 - 16	Z
14 - 0	14 - 0	Z
14 - 0	14 - 0	U

2. A-Register

This 24-bit register is the accumulator of the BOGART arithmetic unit. The contents of the X-register are added to the A-register. It can be cleared to all zeros on command. It shifts circularly. In the multiply and divide step, the shifting functions are different and are covered in the instruction code.

3. Q-Register

This register is a 24-bit auxiliary arithmetic register. It is used with the A-register in the MULTIPLY and DIVIDE STEP instructions and may be used to form a logical product. Several other functions of the Q-register are defined in the instruction code. The Q-register transmits or receives information from the X-register. It has left circular shifting properties and may be combined with the A-register as a 48-bit register.

The Q-register acts as an assembly or buffer register for various types of input and output as defined in the instruction code.

C. Communication Registers

These registers of BOGART translate internal computer signals into a form suitable for activating the external device. The standard

communications signal between the computer and the external equipment has sufficient voltage to saturate a direct-coupled transistor. The output register and the external-function register change the timed pulses from the computer to a direct-current signal. The input register changes a direct-current signal to the clocked pulses necessary for internal use in the computer.

1. I-Register

The input register is the information source for the computer. If a high-speed information source is selected, the input register requests information from that source whenever an input instruction appears in the control unit. If a slow-speed source is selected (punched paper tape unit), one frame is read in anticipation of the input instruction that follows. When an input instruction appears in control, the information read into the I-register is transmitted to the memory unit of the computer. After information is taken from the I-register, the computer cannot use the I-register until new information has been read into it.

2. O-Register

The output register is the principal information output of the X-register. When information is placed in the O-register, control sends signals to the external equipment causing it to accept this information. After the external equipment has accepted the information and is ready for more, it returns a signal to the BOGART control which then may cause more information to be transmitted from the X-register to the O-register.

3. F-Register

This register is the external-function register of the computer. When an external-function instruction appears in control, the F-register receives a 15-bit coding from the low-order 15 bits of the U-register.

All of the external units to the computer monitor the F-register. After information enters the F-register, BOGART control emits a signal on the external function-enable line. This signal causes all external control lines to examine the F-register content.

When an external unit finds a pertinent function in the F-register, this unit performs the function and sends back an external function resume-signal to inform the computer that the external function has been performed. The return-time of the function-resume signal depends on the external function performed.[1]

D. Internal Controls

This control unit obtains instructions from the memory, interprets them, and operates the other BOGART registers and units in correct sequence to initiate the specified instructions. It consists of a series of registers with functions as defined below:

1. P-Register

This 12-bit register always specifies the memory location which contains the next instruction. In the normal operation, successive instructions are obtained from successive memory locations.

To allow for the use of the NORMAL JUMP instruction, P may also receive a 12-bit number from bits 11 to 0 of the U-register. P transmits information to the S-register and also transmits its contents to the Z-register when the RETURN JUMP instruction is activated.

2. U-Register

The U-register is the principal control register of the computer. It consists of 24-bits of storage which may be numbered 23 to 0, from left to right. Various portions of the U-register have different functions. Bits 23 to 18 are interpreted as the operation code and the binary pattern of these bits identifies the instruction to be performed by the computer. In the block diagram, these six bits are called "E". They may be transmitted to the instruction-code or "E"-register.

Bits 17 to 15, called "b", specify which of the seven B-registers will be added to bits 14 to 0 of the U-register before the

[1] See SectionV, EXTERNAL COMMUNICATION SYSTEM, for details of the external functions and codes.

instruction is executed. Bits 17 to 15 of U are transmitted to the three-bit T-register. This register performs the translation and decides which of the B-registers is referred to.

Bits 14 to 12 of U, called "m", specify the modification of the portion of a word to be handled in transmissions from X to Z and Z to X. It is also used to specify variations in the Stop and Jump type instructions.

Bits 11 to 0 of U, called "k", specify a memory reference. These bits may be transmitted to the S-register in the storage control unit.

Bits 14 to 0 of U may be transmitted to bits 14 to 0 of the X-register. This is used in certain orders which use only constants up to 77777.

The U-register receives 24-bits of information from the Z-register in the storage unit. Since the U-register does not depend on the X-register in the Arithmetic unit, they can operate simultaneously.

The low-order 15 bits of U may also receive information from the specified B-boxes.

The low-order 12 bits of U are transmitted to the S-register to specify a memory reference in executing a NORMAL instruction. These same 12 bits may be transmitted to the P-register for executing a JUMP instruction.

The right-most 15 bits of U may be transmitted to the F-register for communication with external equipment. The 15 bits may be transmitted to the X-register where they are treated as a 15-bit positive quantity in the type of instruction that uses the lower 15-bits as a constant.

3. B-Registers

The control unit of BOGART has seven 15-bit B-registers. Each may receive information from the low-order 15 bits of the U-register. The contents of any B-register may be added to the low-order 15 bits of an instruction in the U-register before the instruction is executed by control. The B-register to be used is

decided by the value of the T-register which receives its information from bits 17 to 15 of the U-register.

Instructions are available that add a constant to the B-register specified, compare the B-register with a specified constant, and allow the program to take different courses, depending on the relative magnitude of the B-register.

Addition of the B-register contents to the lower 15 bits of the U-register occurs modulo 2^{15}. The carry that may leave the left-most bit of the U addition is ignored. The B-registers may be used to subtract a constant from the specified address of an instruction. By placing the two's complement (one more than the one's complement) of the minuend in the B - register.

The B_7-register is available as the repeat counter in addition to its normal use as a B-register. In the REPEAT instructions, the repeat count is stored in B_7 and counted down by one each time the repeated instruction is performed. If the repeated instruction calls for a skip or a jump of program address, the content of the B_7-register is left in B_7 and the jump or skip occurs. This feature is discussed under programming.

4. Sub-control Registers

The sub-control units of BOGART are the N-, E-, and K-registers. They are discussed briefly because they may be of interest to the programmer.

a. N-Main Control

This unit controls normal information recirculation and transmission throughout the computer. All computer subcommands are controlled transmissions of information from one register to another. Any instruction may be performed by setting up the correct pattern of sub-commands.

b. E-Instruction Code Translator

The left six bits of U are transmitted to the E-cores which decode the instruction code (c) to one of 57 possible machine operations. The E-cores interpret the command code and set up the correct sequence of sub-commands in Main Control that enables the computer to execute the designated instructions.

c. K-Register

This six-bit register is a counter. It obtains its data from the low-order six bits of U which is the shift count. As a shift is executed, the K-register is counted down to zero. At zero the shift is completed and the next instruction is initiated.

BOGART

PROGRAMMERS MANUAL

SECTION III

INSTRUCTIONS

A. Definitions

1. Instruction Code Format

The program control register interprets a 24-bit instruction for program operation. The bits are numbered from left to right in descending order, 23, 22, 21 . . . . 2, 1, 0. This 24-bit sequence is divided into designator sections (c, b, m, and k), each with its specific purpose.

Designator Section	c	b	m	k
Number of Bits	6	3	3	12

Figure 1

a. c-Designator

This 6-bit function code, represented by bits 23 to 18 of the instruction, specified one of 57 possible instructions to be executed by the program control section. The instruction code is listed on page III-16.

b. b-Designator

The program control section contains seven 15-bit B-boxes, also called index registers. Each B-box stores a constant which may be used to modify the quantities m and k (the 15 low-order bits of an instruction; bits 14 to 0). After the instruction appears in the control unit, m and k are modified by the specified B-box prior to the execution of the instruction. The modified instruction appears only in the U-register, but the original instruction in the storage is unchanged. Though not physically present, B-box zero may be considered to be a 15-bit register whose content is always zero.

The 3-bit index designator, b, is represented by bits 17 to 15 of the instruction. These bits select the appropriate B-box according to the following assignments.

b-Designator	Assignment
0	Make no modification of m, k
1	Add to m, k the contents of B_1
2	Add to m, k the contents of B_2
3	Add to m, k the contents of B_3
4	Add to m, k the contents of B_4
5	Add to m, k the contents of B_5
6	Add to m, k the contents of B_6
7	Add to m, k the contents of B_7

Although B_7 is available as a modifier, it is permanently wired as a counter for the REPEAT instruction. When a REPEAT instruction occurs, the previous content of B_7 is eradicated. B_7 then assumes the value of the address-portion m+k in the REPEAT instruction.

The same B-box may be used to modify any number of instructions in a program cycle.

c. m-Designator

The modifier, m, is represented by bits 14 to 12 of every instruction. The modifier enables BOGART to operate on a 24-bit word, an 8-bit third of a word, or the low-order 15 bits of a word. An 8-bit third of a word is limited to low-order, center, or high-order segments. The low-order 8 bits are bits 7 to 0; the center 8 bits are 15 to 8; and the high-order 8 bits are 23 to 16.

Modifier	Corresponding Bits
0	23 to 0
1, 5	7 to 0
2, 6	15 to 8
3, 7	23 to 16
4	14 to 0

The instructions given by these values are:

m - 0 Operate on the entire 24-bit word as stored.

m - 1 Operate on a 24-bit word which has the low-order 8 bits corresponding to the low-order 8 bits of the word in storage, and which has the high-order 16 bits equal to the eighth bit.

m - 2 Operate on a 24-bit word which has the low-order 8 bits corresponding to the center 8 bits of the word stored, and which has the high-order 16 bits equal to the eighth bit.

m - 3 Operate on a 24-bit word which has the low-order 8 bits corresponding to the high-order 8 bits of the word in storage, and which has the high-order 16 bits equal to the eighth bit.

m - 4 Operate on a 24-bit word which has the low-order 15 bits corresponding to the low-order 15-bits of storage, and which has the high-order 9 bits equal to zero.

m - 5 Operate on a 24-bit word which has the low-order 8 bits corresponding to the low-order 8 bits of the word in storage, and which has the high-order 16 bits equal to zero.

m - 6 Operate on a 24-bit word which has the low-order 8 bits corresponding to the center 8 bits of the word in storage, and which has the high-order 16 bits equal to zero.

m - 7 Operate on a 24-bit word which has the low-order 8 bits corresponding to the high-order 8 bits of the word in storage, and the high-order 16 bits equal to zero.

When an 8-bit third of a word is used in the 24-bit arithmetic registers of BOGART, the 8 bits are relocated to occupy bit-positions 7 to 0 of the register. The high-order 16 bits of the register assume the value specified by m. In storing with m = 1, 2, 3, 5, 6, and 7, BOGART replaces the specified third of a word with bits 7 to 0 of the arithmetic register.

When m = 4, bits 14 to 0 of the original word occupy the same bit-positions in the arithmetic registers. The remaining positions in the register, bits 23 to 15, are zeros. In storing with m = 4, BOGART replaces bits 14 to 0 of the stored word with the corresponding bits of the arithmetic register. The remaining bits of the word are unchanged.

d. k-Base Execution Address

The 12-bit base execution address, k, as modified by B, specifies one of the 4,096 storage addresses as the location of the operand used in the execution of the instruction. Designator k is represented by bits 11 to 0 of the instruction.

2. Arithmetic

BOGART uses one's complement arithmetic. The one's complement is formed by replacing each binary bit with its opposite value. In this system, a negative number is represented by complementing each bit of the positive number. For example, plus five is 000000000000000000000101; minus five is 111111111111111111111010.

Minus zero, or all ones, causes some programming trouble in the one's complement system. Minus zero is considered a negative number for POSITIVE and NEGATIVE JUMP, but for ZERO and NONZERO JUMP it is considered not zero. Usually BOGART will not generate a minus zero with normal arithmetical operations, but in certain cases a minus zero may be generated in the accumulator. Orders LPR, RLP, SCO, RSC, SBT, and LSH may or may not generate a minus zero in the accumulator.

Examples of one's complement arithmetic with word size = 12 are to be found on pages III - 5 and III - 6.

ADDITION:

Decimal	Octal	One's Complement		
+7	+7		000000000111	
(+) -5	(+) -5	carry	111111111010	
+2	+2	(1)	000000000001	
			1	(add carry to RT most bit))
			000000000010	(answer +2)
-7	-7		111111111000	
(+) -6	(+) -6	carry	111111111001	
-13	-15	(1)	111111110001	
			1	(add carry)
			111111110010	(answer -15)
+3	+3	no	000000000011	
(+) -4	(+) -4	carry	111111111011	
-1	-1		111111111110	(answer -1)
-2	-2	carry	111111111101	
(+) +5	(+) +5	(+)	000000000101	
+3	+3	(1)	000000000010	
			1	(add carry)
			000000000011	(answer +3)

SUBTRACTION: The subtrahend is complemented, then added to the minuend.

Decimal	Octal	One's Complement	
+7	+7	000000000111	
(-) -5	(-) -5	000000000101	(complement of -5)
+12	+14	000000001100	(answer +14)
-7	-7	111111111000	
(-) -6	(-) -6	000000000110	(complement of -6)
-1	-1	111111111110	(answer -1)
+3	+3	000000000011	
(-) -4	(-) -4	000000000100	(complement of -4)
+7	+7	000000000111	(answer +7)
-2	-2	carry 111111111101	
(-) +5	(-) +5	(-) 111111111010	(complement of +5)
-7	-7	(1) 111111110111	
		1	(add carry)
		111111111000	(answer -7)

3. Symbols

→ - (arrow) transmit

( ) - (parenthesis) content of

' - (prime) complement of

⊕ - (circular plus) add without carry

> - greater than, A > B (A is greater than B.)

< - less than, A < B (A is less than B.)

∴ - therefore

↩ - return jump

↦ - normal jump

⊖ - (on) status of switch; control light corresponding to switch . . ON

⊙ - (off) status of switch; control light corresponding to switch . . OFF

Y - low-order 15 bits of instruction after modification by B

y - low-order 15 bits of instruction without modification by B

(Y) - content of storage location specified by Y

JS - jump switch

SS - stop switch

Q_{23} - (subscript 23) the 24th or left-most bit position of the register

A_0 - (subscript 0) the right-most bit of the accumulator

A_n - (subscript $_n$) the corresponding bit position (0 through 23) of the accumulator

NI - next instruction

(A_s) - (subscript $_s$) the shifted content of the accumulator

(A_i) - (subscript $_i$) the initial content of the accumulator

(A_f) - (subscript $_f$) the final content of the accumulator

L(Q)(Y) - logical (bit by bit) product of (Q) and (Y)

4. Mnemonic Code

Mnemonic is defined as an aid to memory. The following pages list the octal code, mnemonic code, name of instructions, and the symbolic description for each instruction in BOGART. The mnemonic code has been chosen as a standard notation in refering to BOGART instructions.

Octal Code	Mnemonic Code	Instruction	Symbolic Description of the Instruction
00			
01	ENB	Enter B	$y \rightarrow B$
02	LDB	Load B	$(Y) \rightarrow B$
03	STB	Store B	$(B) \rightarrow Y$
04	INB	Increase B	$(B)_i + y \rightarrow B_f$
05	BSK	Index skip	If $(B) > y$, skip and clear B; if $(B) < y$, $(B) + 1 \rightarrow B_f$
06	TSK	Threshold skip	If $(B) > y$, skip
07	RPT	Repeat	Do next instruction Y times
10		Unassigned	
11	ENA	Enter A	$Y \rightarrow A$
12	LDA	Load A	$(Y) \rightarrow A$
13	STA	Store A	$(A) \rightarrow Y$
14	COA	Complement A	$(A') \rightarrow A$
15	SHA	Shift A	$(A)_s \rightarrow (A)_f$
16	LSH	Long shift	$(A, Q)_s \rightarrow (A, Q)_f$
17	AJP	A jump	(See symbolic expression for each option on page III-30.)

Octal Code	Mnemonic Code	Instruction	Symbolic Description of the Instruction
20		Unassigned	
21	ENQ	Enter Q	$Y \rightarrow Q$
22	LDQ	Load Q	$(Y) \rightarrow Q$
23	STQ	Store Q	$(Q) \rightarrow Y$
24	COQ	Complement Q	$(Q') \rightarrow Q$
25	SHQ	Shift Q	$(Q_s) \rightarrow Q_f$
26	TAQ	Transmit A to Q	$(A) \rightarrow Q$
27	QJP	Q jump	(See symbolic expression for each option on page III-38.)
30		Unassigned	
31	ADC	Add constant	$Y + (A)_i \rightarrow A_f$
32	ADD	Add	$(Y) + (A)_i \rightarrow A_f$
33	ADR	Add replace	$(Y) + (A)_i \rightarrow A_f$ and Y
34	RAO	Replace add one	$(Y) + 1 \rightarrow A$ and Y
35	LAQ	Load A, add Q	$(Y) + (Q) \rightarrow A$
36	AQS	Add Q and store	$(A)_i + (Q) \rightarrow A_f$ and Y
37	RAQ	Replace add Q	$(Y) + (Q) \rightarrow A$ and Y
40		Unassigned	
41	SBC	Subtract constant	$(A)_i - Y \rightarrow A_f$
42	SUB	Subtract	$(A)_i - (Y) \rightarrow A_f$
43	SBR	Subtract replace	$(A)_i - (Y) \rightarrow A_f$ and Y
44	RSO	Replace subtract one	$(Y) - 1 \rightarrow A$ and Y
45	LSQ	Load A, subtract Q	$(Y) - (Q) \rightarrow A$
46	SQS	Subtract Q and store	$(A)_i - (Q) \rightarrow A_f$ and Y
47	RSQ	Replace subtract Q	$(Y) - (Q) \rightarrow A$ and Y
50	LPR	Logical product	$L(Q)(Y) \rightarrow A$
51	ALP	Add logical product	$(A)_i + L(Q)(Y) \rightarrow A_f$
52	SCO	Selective complement	If $(Y)_n = 1$, complement $(A)_n$
53	SBT	Substitute bits	$L(Y_n)(Q_n) + L(A_n)(Q_n') \rightarrow A_f$
54	RLP	Replace logical product	$L(Q)(Y) \rightarrow A$ and Y

Octal Code	Mnemonic Code	Instruction	Symbolic Description of the Instruction
55	RAL	Replace add logical product	$(A)_i + L(Q)(Y) \rightarrow A$ and Y
56	RSC	Replace selective complement	If $(Y)_n = 1$, complement (A_n), $A_n \rightarrow Y_n$
57	RSB	Replace substitute bits	$L(Y_n)(Q_n) + L(A_n)(Q'_n) \rightarrow A_n$ and Y_n
60	MUL	Multiply step	Product $\rightarrow$ Q
61	DIV	Divide step	Quotient $\rightarrow$ Q, remainder $\rightarrow$ A
62	SCL	Selective clear	$(A)_i + L(A)(Y) \rightarrow A_f$
63		Unassigned	
64	ESK	Equal skip	If (A) = (Y), skip
65	USK	Unequal skip	If $(A) \neq (Y)$, skip
66	GSK	Greater skip	If (A) > (Y), skip
67	LSK	Less skip	If (A) < (Y), skip
70	EXF	External function	Y F
71	INF	Input	If m = 1, 2, 3, 5, 6 or 7; I $\rightarrow$ Y
72	OUT	Output	If m = 1, 2, 3, 5, 6 or 7; Y $\rightarrow$ O
73	TRJ	Trace jump	(See section III-C for description and symbolic expression for each option.)
74	SNJ	Sense jump	
75	SLJ	Selective jump	
76	SLS	Selective stop	
77		Unassigned	

The format for the following Instruction Codes and Command Steps was selected as the most practical adaptation of this material.

5. Instruction Terms

a. RETURN JUMP

Transmit the address of the next sequential instruction to the low-order 15 bits of the storage location specified by the low-order 12 bits of the RETURN JUMP instruction. Then change the program address to obtain the next instruction from the storage location specified plus one.

The "b"-modifier may be used.

The third-of-a-word feature may not be used.

The repeat may be used. The repeat sequence is terminated after the execution of a Jump instruction when the jump occurs, storing the number of unperformed repetitions in B_7. If the Jump instruction does not occur, repetition of the instruction continues.

The RETURN JUMP is very useful for sub-routines.

Example:

Main Program

ML	OP	B	M	Address
X	SLJ		4, 5, 6 or 7	Y
X+1				

Sub-Routine

ML	OP	B	M	Address
Y				X+1
Y+1				

X - Location of the RETURN JUMP instruction.
X+1 - Location of the instruction following the RETURN JUMP instruction.
Y - Specified location, where number X+1 is transmitted.
Y+1 - Location of the next instruction to be executed.

(Example continued on next page)

Step (1)	Step (2)	Step (3)
Execute X	Transmit X+1 to low-order 15 bits of (Y)	Execute Y+1 Continue with Y-sequence of instruction until programmed otherwise.

b. NORMAL JUMP (Set PAK=Y)

Change the program address to obtain the next instruction from the memory location specified by the lower 12 bits of the NORMAL JUMP instruction.

The "b"-modifier may be used.

The third-of-a-word feature may not be used.

The repeat may be used. The repeat sequence is terminated if the Jump occurs, storing the number of unperformed repetions in B_7. If the Jump does not occur, repetition of the instruction continues.

This instruction is used to change series of addresses in a program.

Example:

Series A

ML	OP	B	M	Address
X	SLJ		1, 2, 3, or $\emptyset$	Y

Series B

ML	OP	B	M	Address
Y				

X - Location of Jump instruction
Y - Location of next instruction to be executed

B. Execution Time of Instructions

Octal Code	Mnemonic Code	Time in Microseconds Normal	Repeat
00			
01	ENB	20	
02	LDB	40	
03	STB	40	
04	INB	20	
05	BSK	32	
06	TSK	32	
07	RPT	20	
10			
11	ENA	32	12
12	LDA	40	20
13	STA	40	20
14	COA	28	8
15	SHA	Timing depends on amount of shift;	
16	LSH	see pages III-28 and III-29.	
17	AJP	Variable timing depending on JUMP or NO JUMP conditions; see page III-30.	
20			
21	ENQ	32	12
22	LDQ	40	20
23	STQ	40	20
24	COQ	36	16
25	SHQ	Timing depends on amount of shift; see page III-36.	
26	TAQ	32	12
27	QJP	Variable timing depending on JUMP or NO JUMP conditions; see page III-38.	
30			
31	ADC	32	12
32	ADD	40	20
33	ADR	60	40
34	RAO	60	40
35	LAQ	48	28
36	AQS	56	36
37	RAQ	68	48

Octal Code	Mnemonic Code	Time in Microseconds Normal	Repeat
40			
41	SBC	32	12
42	SUB	40	20
43	SBR	60	40
44	RSO	60	40
45	LSQ	48	28
46	SQS	56	36
47	RSQ	68	48
50	LPR	44	24
51	ALP	44	24
52	SCO	40	20
53	SBT	44	24
54	RLP	60	40
55	RAL	64	44
56	RSC	60	40
57	RSB	60	40
60	MUL	Variable timing; see page	III-62.
61	DIV	- - - - - do - - - - - - - -	III-66.
62	SCL	40	20
63			
64	ESK	64	44
65	USK	64	44
66	GSK	60	40
67	LSK	60	40
70	EXF	36	16
71	INP	Variable timing; see page	III-76.
72	OUT	- - - - - do - - - - - - - -	III-78.
73	TRJ	- - - - - do - - - - - - - -	III-80.
74	SNJ	- - - - - do - - - - - - - -	III-81.
75	SLJ	- - - - - do - - - - - - - -	III-83.
76	SLS	- - - - - do - - - - - - - -	III-85.
77			

Note that instructions 00 through 07 are not timed under REPEAT CONTROL.

The method of timing an instruction under REPEAT CONTROL is as follows:

Time of RPT Instr.	plus	number of repetitions minus one	times	timing of instr. under RPT	plus	normal time of instr.

Example:

Repeat the ENA (11) instruction 15 times:

20 + (14)(12) + 32 = 220 microseconds

C. Instruction Code

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ENTER B	01	(ENB)	y → B

a. INSTRUCTION DESCRIPTION - Replace the content of the specified B-register with the low-order 15 bits of the instruction.

b. SPECIAL FUNCTIONS

1. The execution of this instruction causes the repeat sequence to be terminated.

2. The B-designator, "b", is used to designate a specific B-box, but the instruction is not modified by the specified B-register before execution.

3. The INSTRUCTION MODIFIER, "m", is a part of the constant entering B.

c. TIMING - Normal operation time is 20 microseconds.

4. If the B-designator is zero this instruction functions as a "no operation" instruction.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LOAD B	02	(LDB)	(Y) → B

a. INSTRUCTION DESCRIPTION - Add the content of B-box specified to the low-order 15 bits of this instruction, then replace the content of the B-box with the content of the storage location specified by this modified address.

b. SPECIAL FUNCTIONS - (B) is added to m, k and transferred to S before B is loaded from storage. The bits from the storage location may comprise any 8-bit third of a word or the low-order 15 bits of a word.

1. Execution of this instruction causes the repeat sequence to be terminated.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", as a word-portion designator, is used with this instruction. If m = 0, the specified B-register receives the low-order 15 bits from the storage location.

c. TIMING - Normal operation time is 40 microseconds.

4. If the B-designator is zero, this instruction functions as a "no operation" instruction.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
STORE B	03	(STB)	(B)→y

a. INSTRUCTION DESCRIPTION - Replace the low-order 15 bits of the content of the storage location with the content of the specified B-register.

b. SPECIAL FUNCTIONS - The high-order 9 bits of the storage location will remain unchanged. All arithmetic registers remain unchanged.

1. The execution of this instruction causes the repeat sequence to be terminated.

2. The B-designator, "b", is used to designate a specific B-box, but the instruction is not modified by the specified B-register before execution. The B-register specified is not changed.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction. (Only the low-order 15 bits of the storage location change, regardless of the value of "m".)

c. TIMING - Normal operation time is 40 microseconds.

4. If the B-designator is zero, the low-order fifteen bits of the storage location specified will be set to zero.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
INCREASE B	04	(INB)	(B) + y → B

a. INSTRUCTION DESCRIPTION - Add the low-order 15 bits of this instruction to the specified B-register.

b. SPECIAL FUNCTIONS - This instruction enables B to advance its address position by any constant.

1. Execution of this instruction causes the repeat sequence to be terminated.

2. The B-designator, "b", is used to designate a specific B-box, but the instruction is not modified by the B-register before execution.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operation time is 20 micro-seconds.

4. If the B-designator is zero, this instruction functions as a "no operation" instruction.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
INDEX SKIP	05	(BSK)	If (B)$\geq$y, skip NI then clear B; If (B)$<$y, then (B)+1$\rightarrow$B

a. INSTRUCTION DESCRIPTION - If the content of the specified B-register is greater than or equal to the low-order 15 bits of this instruction, skip the next sequential instruction and clear B. If the content of the specified B-register is less than the low-order 15 bits of this instruction, add one to the specified B-register and execute the next sequential instruction.

b. SPECIAL FUNCTIONS

1. Execution of this instruction causes the repeat sequence to be terminated.

2. The B-designator, "b", is used to designate a specific B-box, but the instruction is not modified by the B-register before execution.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operation time is 32 microseconds.

4. For the purposes of this instruction, numbers which contain a 1 in the fifteenth bit are regarded as negative.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
THRESHOLD SKIP	06	(TSK)	If (B) $\geq$ y, skip NI

a. INSTRUCTION DESCRIPTION - If the content of the specified B-register is greater than or equal to the low-order 15 bits of this instruction, skip the next sequential instruction. If the content of the specified B-register is less than the low-order 15 bits of this instruction, continue with the present sequence.

b. SPECIAL FUNCTIONS - This instruction checks the value of (B) without changing its content.

1. The execution of this instruction causes the repeat sequence to be terminated.

2. The B-designator, "b", is used to designate a specific B-box, but the instruction is not modified by the B-register before execution.

3. The INSTRUCTION MODIFIER, "m", has no meaning.

c. TIMING - Normal operation time is 32 microseconds.

4. For the purposes of this instruction, numbers which contain a 1 in the fifteenth bit are regarded as negative.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPEAT	07	(RPT)	Execute NI Y times.

a. INSTRUCTION DESCRIPTION - Execute the next instruction the number of times indicated by the low-order 12 bits of this instruction, as modified by specified B-box.

If m = 0, 2, 4, 6 after modification -- No change in the repeated instruction.

If m = 1 or 3 after modification -- Increase the low-order 15 bits of the repeated instruction by one each time the instruction is executed.

If m = 5 or 7 after modification -- Decrease the low-order 15 bits of the repeated instruction by one each time the instruction is executed.

B_7 (bits 00 through 11) is used as the repeat counter.

The repeat is terminated under any one of the following conditions with B_7 containing the count of unperformed repetitions:

1. When the instruction being repeated calls for a change of program address, such as jump or skip.

2. Upon receipt of an "External Disconnect" signal from external equipment.

3. Upon depletion of the repeat count.

4. Execution of instructions 01, 02, 03, 04, 05, 06, 07.

b. SPECIAL FUNCTIONS

1. B_7 is decreased by one each time the repeated instruction is executed.

2. B-designator, "b", is used.

3. The INSTRUCTION MODIFIER, "m", is used with this instruction to specify the address modification of the repeated instruction.

4. Since the repeat instruction is an auxiliary instruction, one or more of the registers of the computer may change during the execution of the repeated instruction.

5. The repeated instruction is modified by "b" in the initial execution. Thereafter, the "m" of the repeat instruction designates the address modification.

6. Repeating an instruction zero times causes that instruction to be skipped.

c. TIMING - Normal operation time is 40 microseconds if Repeat count = 0; if Repeat count $\neq$ 0, operation time is 20 microseconds.

d. SEQUENCE OF OPERATIONS FOR A REPEATED INSTRUCTION - The modifier (specifing the modification of the repeated instruction) is stored in B_7 with the repeat count. Thus, a modification by B_7 includes the modifier. The modifier is cleared from B_7 upon termination of the repeated instruction.

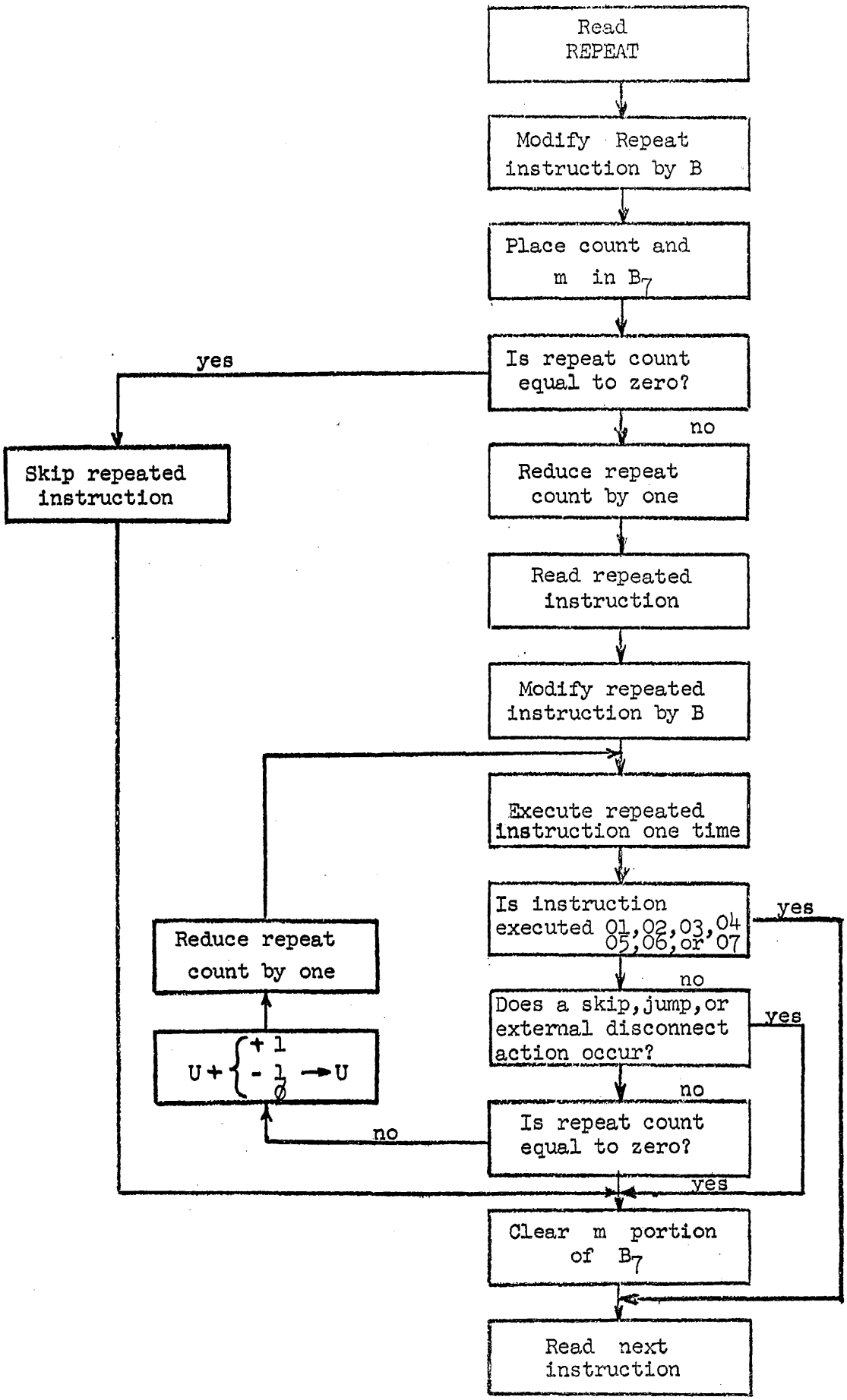


Figure 2 - Sequence of Operations for a Repeated Instruction

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ENTER A	11	(ENA)	Y → A

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the low-order 15 bits of this instruction as modified by specified B.

b. SPECIAL FUNCTIONS - If "b" is zero, the low-order 15 bits of the instruction enter A as a positive 15-bit number. If "b" is not zero, the low-order 15 bits, plus the content of the designated B-box, enter A as a positive number. In any case the high order 9 bits of A, bits 15 - 23, will be zero.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", as a modifier, has no meaning in this instruction.

c. TIMING - Normal operation time is 32 microseconds; repeat, 12 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LOAD A	12	(LDA)	(Y) → A

a. INSTRUCTION DESCRIPTION - Replace the content of A with the content of the storage location specified by the low-order 15 bits of this instruction as modified by B.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", used as a word-portion designator, is available for use with this instruction.

c. TIMING - Normal operation time is 40 microseconds; repeat operation time, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
STORE A	13	(STA)	(A) → Y

a. INSTRUCTION DESCRIPTION - Replace the content of the storage location, specified by the low-order 15 bits of this instruction, as modified by B, with the content of the Accumulator.

b. SPECIAL FUNCTIONS - The Accumulator is unchanged by this operation.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator in this instruction. Only the portion of the stored information as specified by "m" is altered. Bits 0 to 7 of A may be stored in any designated third of a word.

c. TIMING - Normal operation time is 40 microseconds; repeat, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
COMPLEMENT A	14	(COA)	$(A^1) \rightarrow A$

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with its one's complement.

b. SPECIAL FUNCTIONS - The instruction may be classified as a "no address" instruction since the low-order 18 bits of the instruction are not used. Complementing of positive zero (all 0's) results in positive zero.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", has no meaning in this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operation time is 28 microseconds; repeat operation, 8 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SHIFT A	15	(SHA)	$(A)_s \rightarrow (A)_f$

a. INSTRUCTION DESCRIPTION - Shift the Accumulator, circularly to the left, the number of bit positions specified by the low-order 6 bits of this instruction as modified by specified B.

b. SPECIAL FUNCTIONS - The maximum shift is 63 bit positions.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Operation time, if 1, 2, 4, 5, 6, 8, 9, or 10 shifts, is 32 microseconds. Operation time if 3, 7, 11, 12, 13, 14, or 15 shifts, is 36 microseconds. Operation time is increased 4 microseconds with each additional four shifts. Operation time for repeat is variable, depending upon number of shifts.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LONG SHIFT	16	(LSH)	$(AQ)_s \rightarrow (AQ)_f$

a. INSTRUCTION DESCRIPTION - Shift (A) and (Q), as one 48-bit register, circularly to the left the number of bit positions as specified by the low-order 6 bits of this instruction as modified by B.

b. SPECIAL FUNCTIONS - The maximum shift is 63 bit positions.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Operation time, if 1, 2, 4, 5, 6, 8, 9, or 10 shifts, is 32 microseconds. Operation time, if 3, 7, 11, 12, 13, 14, or 15 shifts, is 36 microseconds. Operation time is increased by 4 microseconds with each additional four shifts. Operation time for repeat is variable depending upon the number of shifts.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
A JUMP	17	(AJP)	

a. INSTRUCTION DESCRIPTION - The Accumulator Jump instruction has eight options, depending on the value of "m" and the present status of the Accumulator.

If m = 0 and if the content of the Accumulator is zero, do a normal jump; if A is not zero, continue the present sequence.

If m = 1 and if the content of the Accumulator is not zero, do a normal jump; if the content of the Accumulator is equal to zero, continue the present sequence.

If m = 2 and if the content of the Accumulator is positive (A_{23}=0), do a normal jump; if A is not positive, continue the present sequence.

If m = 3 and if the content of the Accumulator is negative (A_{23}=1), do a normal jump; if A is not negative, continue the present sequence.

If m = 4 and if the content of the Accumulator is zero do a return jump; if A is not zero, continue the present sequence.

If m = 5 and if the content of the Accumulator is not zero, do a return jump. If the content of the Accumulator is equal to zero, continue the present sequence.

If m = 6 and if the content of the Accumulator is positive (A_{23}=0), do a return jump; if A is not positive, continue the present sequence.

If m = 7 and if the content of the Accumulator is negative (A_{23}=1), do a return jump; if A is not negative, continue the present sequence.

b. SPECIAL FUNCTIONS - If the trace jump control is on, a return jump to zero sequence will replace either the normal jump or return jump sequence. During this instruction the Accumulator does not change.

1. The REPEAT feature is available for use with this instruction, being terminated if the jump condition exists.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used to designate the jump condition.

c. TIMING - For NO JUMP, the normal operation time is 40 microseconds; repeat operation, 20 microseconds. For JUMP, the normal operation time for no trace jump and m = 0, 1, 2, or 3, is 40 microseconds; for trace jump or m = 4, 5, 6, or 7, is 56 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ENTER Q	21	(ENQ)	Y→Q

a. INSTRUCTION DESCRIPTION - Replace the content of the Q-register with the low-order 15 bits of this instruction as modified by B.

b. SPECIAL FUNCTIONS - If "b" is zero, the low-order 15 bits of the instruction enter Q as a positive 15-bit number. If "b" is not zero, the low-order 15 bits, plus the content of the designated B-box, enter Q as a positive number. In any case the high order 9 bits of Q, bits 15 - 23, will be zero.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operation time is 32 microseconds; repeat, 12 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LOAD Q	22	(LDQ)	(Y) → Q

a. INSTRUCTION DESCRIPTION - replace the content of the Q-register with the content of the storage location specified by the low-order 15 bits of this instruction, as modified by B.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with the instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 40 microseconds; repeat, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
STORE Q	23	(STQ)	(Q) → Y

a. INSTRUCTION DESCRIPTION - Replace the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, with the content of the Q-register.

b. SPECIAL FUNCTIONS - The Q-register remains unchanged by this operation.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction. Only the portion of the stored information as specified by "m" is altered. Bits 0 to 7 of Q may be stored in any designated third of a word.

c. TIMING - Normal operating time is 40 microseconds; repeat, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
COMPLEMENT Q	24	(COQ)	$(Q^{1}) \rightarrow Q$

a. INSTRUCTION DESCRIPTION - Replace the content of the Q-register with its one's complement.

b. SPECIAL FEATURES - The instruction is a "no-address" instruction since the low-order 18 bits of the instruction are not used. Complementing of positive zero (all 0's) results in a negative zero (all 1's).

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", has no meaning in this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operating time is 36 microseconds; repeat, 16 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SHIFT Q	25	(SHQ)	$(Q)_s \rightarrow Q_f$

a. INSTRUCTION DESCRIPTION - Shift Q circularly to the left the number of bit positions specified by the low-order 6 bits of this instruction as modified by B.

b. SPECIAL FUNCTIONS - The maximum shift is 63 bit positions.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Operating time for 1, 2, 4, 5, 6, 8, 9, or 10 shifts is 32 microseconds. Operation time, if 3, 7, 11, 12, 13, 14, or 15 shifts, is 36 microseconds. Operation time is increased 4 microseconds with each additional four shifts.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
TRANSFER A TO Q	26	(TAQ)	(A)→Q

a. INSTRUCTION DESCRIPTION - Replace the content of the Q-register with the content of the Accumulator.

b. SPECIAL FUNCTIONS - This instruction is a "no-address" instruction, since the low-order 18 bits are not used. The A-register remains unchanged and the Q-register is equal to the Accumulator after executing the instruction.

1. The REPEAT feature has no meaning in this instruction.

2. The B-designator, "b", has no meaning in this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operating time is 32 microseconds; repeat, 12 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
Q JUMP	27	(QJP)	

a. INSTRUCTION DESCRIPTION - This instruction has eight options, depending upon the value of "m" and the present status of the Q-register.

If m = 0 and if Q is positive (Q_{23}=0), do a normal jump; if Q is not positive, continue the present sequence. Then in either case, shift Q circularly left one bit position.

If m = 1 and if Q is negative (Q_{23}=1), do a normal jump; if Q is not negative, continue the present sequence. Then in either case, shift Q circularly left one bit position.

If m = 2 and if Q is positive (Q_{23}=0), do a normal jump; if Q is not positive, continue the present sequence. Do not shift Q.

If m = 3 and if Q is negative (Q_{23}=1), do a normal jump if Q is not negative, continue the present sequence. Do not shift Q.

If m = 4 and if Q is positive (A_{23}=0), do a return jump; if Q is not positive, continue the present sequence. Then in either case, shift Q circularly left one bit position.

If m = 5 and if Q is negative (Q_{23}=1), do a return jump; if Q is not negative, continue the present sequence. Then in either case, shift Q circularly left one bit position.

If m = 6 and if Q is positive (Q_{23}=0), do a return jump; if Q is not positive, continue the present sequence. Do not shift Q.

If m = 7 and if Q is negative (Q_{23}=1), do a return jump; if Q is not negative, continue the present sequence. Do not shift Q.

b. SPECIAL FUNCTIONS - If the trace jump control is on, a return jump to zero sequence will replace either the normal jump or return jump sequence. The Q-register changes when "m" = 0, 1, 4, or 5, but does not change when "m" = 2, 3, 6, or 7.

1. The REPEAT feature is available for use with this instruction, and the repeat is terminated when the jump occurs.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used to designate the jump condition.

c. TIMING - For NO JUMP, the normal operation time is 40 microseconds; repeat, 20 microseconds. For JUMP, the normal operation time for no trace jump and m=0, 1, 2, or 3, is 40 microseconds; for trace jump or m=4, 5, 6, or 7, it is 56 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ADD CONSTANT	31	(ADC)	$Y + (A)_i \rightarrow A_f$

a. INSTRUCTION DESCRIPTION - Add to the content of the Accumulator the low-order 15 bits of this instruction as modified by B.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operation time is 32 microseconds; repeat, 12 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ADD	32	(ADD)	$(Y) + (A)_i \rightarrow A_f$

a. INSTRUCTION DESCRIPTION - Add the content of the storage location specified by the low-order 15 bits of this instruction, as modified by B, to the content of the Accumulator.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 40 microseconds; repeat, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ADD REPLACE	33	(ADR)	$(A)_i + (Y) \rightarrow A$, then $(A)_f \rightarrow Y$

a. INSTRUCTION DESCRIPTION - Add the content of the storage location specified by the low-order 15 bits of this instruction to the Accumulator, then replace the content of that storage location with the content of the Accumulator.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE ADD ONE	34	(RAO)	(Y) + 1→A, then (A)→Y

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with one plus the content of the storage location, then replace the content of that storage location with the content of the Accumulator.

b. SPECIAL FUNCTION - After Execution, the Accumulator will have the same content as that of the specified storage location.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LOAD A, ADD Q	35	(LAQ)	(Y) + (Q) → A

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the content of the Q-register, plus the content of the storage location as specified by the low-order 15 bits of this instruction as modified by B.

b. SPECIAL FUNCTIONS - After the execution of this instruction the Q-register remains unchanged.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 48 microseconds; repeat, 28 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ADD Q AND STORE	36	(AQS)	$(A)_i + (Q) \rightarrow A_f$ then $(A)_f \rightarrow Y$

a. INSTRUCTION DESCRIPTION - Add the content of the Q-register to the content of the Accumulator, then replace the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, with the content of the Accumulator.

b. SPECIAL FUNCTIONS - After execution of this instruction the Q-register remains unchanged.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 56 microseconds; repeat, 36 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE ADD Q	37	(RAQ)	(Y) + (Q) → A, then (A) → Y

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the content of the storage location specified by the low-order 15 bits of this instruction; add the content of the Q-register, then replace the content of that storage location with the content of the Accumulator.

b. SPECIAL FUNCTIONS - After execution of this instruction the Q-register remains unchanged. The content of the specified storage location is equal to the content of A.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 68 microseconds; repeat, 48 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SUBTRACT CONSTANT	41	(SBC)	$(A)_i - Y \rightarrow A_f$

a. INSTRUCTION DESCRIPTION - Subtract the low-order 15 bits of this instruction, as modified by B, from the content of the Accumulator.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operation time is 32 micro-seconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SUBTRACT	42	(SUB)	$(A)_i - (Y) \rightarrow A_f$

a. INSTRUCTION DESCRIPTION - Subtract the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, from the content of the Accumulator.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", may be used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 40 microseconds; repeat, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SUBTRACT REPLACE	43	(SBR)	$(A)_i - (Y) \rightarrow A_f$ then $(A)_f \rightarrow Y$

a. INSTRUCTION DESCRIPTION - Subtract the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, from the content of the Accumulator, then replace the content of that same storage location with the content of the Accumulator.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", may be used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE SUBTRACT ONE	44	(RSO)	(Y) - 1 → A, then (A) → Y

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B; subtract one: and then replace the content of the specified storage location with the content of the Accumulator.

b. SPECIAL FUNCTIONS - After execution, the content of the Accumulator is equal to the content of the specified storage location.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LOAD A, SUBTRACT Q	45	(LSQ)	(Y) - (Q) → A

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, minus the content of the Q-register.

b. SPECIAL FUNCTIONS - The content of the Q-register remains unchanged.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", may be used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 48 microseconds; repeat, 28 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SUBTRACT Q AND STORE	46	(SQS)	$(A)_i - (Q) \rightarrow A_f$ then $(A)_f \rightarrow Y$

a. INSTRUCTION DESCRIPTION - Subtract the content of the Q-register from the content of the Accumulator, then replace the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, with the final content of the Accumulator.

b. SPECIAL FUNCTIONS - The Q-register remains unchanged.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", may be used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 56 microseconds; repeat, 36 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE; SUBTRACT Q	47	(RSQ)	(Y) - (Q) → A, then (A) → Y

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B; subtract the content of the Q-register, then replace the content of the specified storage location with the content of the Accumulator.

b. SPECIAL FUNCTIONS - The content of the storage location is equal to the content of A. The Q-register remains unchanged.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", may be used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 68 microseconds; repeat, 48 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LOGICAL PRODUCT	50	(LPR)	L(Q)(Y) → A

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the logical product of the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, and the content of the Q-register.

b. SPECIAL FUNCTIONS - The Q-register remains unchanged. Logical product means bit-by-bit multiplication without carry. (1 x 1 = 1, 0 x 1 = 0, 1 x 0 = 0, 0 x 0 = 0)

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 44 microseconds; repeat, 24 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
ADD LOGICAL PRODUCT	51	(ALP)	$(A)_i + L(Y)(Q) \longrightarrow A_f$

a. INSTRUCTION DESCRIPTION - Add to the content of the Accumulator the Logical Product of the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, and the content of the Q-register.

b. SPECIAL FUNCTIONS - The Q-register remains unchanged.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 44 microseconds; repeat, 24 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SELECTIVE COMPLEMENT	52	(SCO)	If $(Y)_n = 1$, complement $(A)_n$ or $A_I \oplus (Y) \rightarrow A_F$

a. INSTRUCTION DESCRIPTION - Complement bits of the Accumulator corresponding to the 1's in the storage location, specified by the low-order 15 bits of this instruction as modified by B. This accomplishes the "vector-add" or "add-without carry" function between (A) and (Y).

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is available for use with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 40 microseconds; repeat, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SUBSTITUTE BITS	53	(SBT)	$L(Y)_n(Q)_n + L(A)_n(Q')_n \rightarrow A_f$

a. INSTRUCTION DESCRIPTION - Replace bits of the Accumulator with bits of the storage location, specified by the low-order 15 bits of this instruction as modified by B, corresponding to the 1's in the Q-register.

b. SPECIAL FUNCTIONS - Bits of the Accumulator are replaced with bits from the storage location where Q-register contains a 1 in the relative position. (This instruction may vary from the normal idea of substitute bits.) The Q-register is unchanged during this instruction.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 44 microseconds; repeat, 24 microseconds.

4. If the contents of Q and of the storage location specified are the same, this instruction places the boolean sum (logical "or") of A and Q into A.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE LOGICAL PRODUCT	54	(RLP)	L(Q)(Y)→A, then (A)→Y

a. INSTRUCTION DESCRIPTION - Replace the content of the Accumulator with the logical product of the storage location, specified by the low-order 15 bits of this instruction, and the content of the Q-register; then replace the content of the specified storage location with the final content of the Accumulator.

b. SPECIAL FUNCTIONS - The Q-register is unchanged during the instruction.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE ADD LOGICAL PRODUCT	55	(RAL)	$(A)_i + L(Y)(Q) \longrightarrow A_f$, then $(A)_f \longrightarrow Y$

a. INSTRUCTION DESCRIPTION - Add to the content of the Accumulator the logical product of the content of the storage location, specified by the low-order 15 bits of this instruction as modified by B, and the content of the Q-register; then replace the content of the specified storage location with the final content of the Accumulator.

b. SPECIAL FUNCTIONS - The Q-register remains unchanged during this instruction.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 64 microseconds; repeat, 44 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE SELECTIVE COMPLEMENT	56	(RSC)	If $(Y)_n = 1$, complement $(A)_n$; then $A_n \rightarrow Y_n$ or $A_I \oplus (Y) \longrightarrow A_F$ and Y

a. INSTRUCTION DESCRIPTION - Complement bits of the Accumulator corresponding to the 1's in the storage location, specified by the low-order 15 bits of this instruction as modified by B, then replace the content of the specified storage location with the final content of the Accumulator.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
REPLACE SUBSTITUTE BITS	57	(RSB)	$L(Y)_n(Q)_n + L(A)_n(Q)_n \rightarrow A_n$, then $(A) \rightarrow Y$

a. INSTRUCTION DESCRIPTION - Replace bits of the Accumulator with bits of the storage location, specified by the low-order 15 bits of this instruction as modified by B, corresponding to the 1's in the Q-register; then replace the content of the specified storage location with the content of the Accumulator.

b. SPECIAL FUNCTIONS - The Q-register remains unchanged during the instruction.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
MULTIPLY STEP	60	(MUL)	$(Q)(Y) \rightarrow Q$

a. INSTRUCTION DESCRIPTION - The multiplicand is in the storage location specified by the low-order 15 bits of the MUL instruction. If the content of Q_0 equals 1 (odd), the content of the specified storage location is added to the content of the Accumulator. If the content of Q_0 equals 0 (even), then the Accumulator remains unchanged. In either of these cases, A and Q are shifted right one bit position as one open-ended register.

This instruction is only a phase of true multiplication and therefore is called multiply step. To complete a multiplication one must program as follows:

1. Clear Accumulator. (Multiply add occurs if a number other than zero is in A.)

2. Place the number representing the multiplier in the Q-register.

3. Execute the multiply step 24 times. (Note: If the multiplier is known in advance, to save machine time the multiply step may be repeated as many times as there are significant bits in the multiplier.)

b. SPECIAL FUNCTIONS - If the multiply step is repeated 24 times, the answer will be found in the Q-register with the high-order bits extended in the A-register.

This instruction is valid only for positive quantities.

1. The REPEAT feature is used as noted on the preceding page.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Operation time varies according to the problem, depending upon the following conditions:

If $Q_0 = 0$, the operation time is 28 microseconds; repeat, 8 microseconds.

If $Q_0 = 1$, the operation time is 52 microseconds; repeat, 32 microseconds.

MULTIPLY STEP

EXAMPLE 1 (If the multiplier is unknown.)

Accumulator	Q-Register	
	The value of R	Before MUL executed
$A_{23}, A_{22}, \ldots$ etc. $\ldots A_2, A_1, A_0$	$Q_{23}, Q_{22} \ldots$ etc. $\ldots Q_1, Q_0$	

PROBLEM: R times S equals P

Multiplier: R

Multiplicand: S

If $Q_0=1$, add Y to A; then shift AQ right 1 bit position.
If $Q_0=0$, no change in A; then shift AQ right 1 bit position. (This applies to each of the 24 steps.)

Program Format

M.L.	OP.	B	M	Addr.	
0010	ENA	0	0	0000	Clear A
0011	LDQ	0	0	0020	R (Q)
0012	RPT	0	0	0030	Since the multiplier is unknown, step the MUL instruction 24 times.
0013	MUL	0	0	0021	Answer will be in "Q".
0020	..... R				(Multiplier)
0021	..... S				(Multiplicand)

Q-Register	After the last step of MUL is executed
the value of P	
$Q_{23}, Q_{22} \ldots$ etc. $\ldots Q_1, Q_0$	

MULTIPLY STEP

EXAMPLE 2 (If the multiplier is known.)

Accumulator	Q-Register
0 0 ... zeros 0 0 0	0 0... zeros ..0 0 1 0 1
$A_{23}, A_{22}, \ldots$ etc... A_1, A_0	$Q_{23}, Q_{22}, \ldots$ etc... Q_1, Q_0

Before MUL is executed

PROBLEM: 5 times 4

Multiplier: 5

Multiplicand: 4

If $Q_0=1$, (A) + (Y) → A, then shift AQ right one bit position.

Q-Register	
1 0	0 0 ..zeros.. 0 1 0
A_1, A_0	$Q_{23}, Q_{22}, .$ etc. $.Q_1, Q_0$

After MUL STEP 1

Program Format

M.L.	OP.	B	M	Addr.	
10	ENA	0	0	0000	Clear A
11	LDQ	0	0	0020	Multiplier → Q
12	RPT	0	0	0003	Step the MUL instruction 3 times.
13	MUL	0	0	0021	
14	LSH	0	0	0003	Long shift left 3 bit positions to put ANSWER in A.
20	00	0	0	0005	(Multiplier)
21	00	0	0	0004	(Multiplicand)

If $Q_0=0$, no change in A; then shift AQ right one bit position.

Q-Register	
0 1	0 ..zeros ... 0 0 1
A_0	Q_{23}.. etc..... Q_1, Q_0

After MUL Step 2

If $Q_0=1$, (A) + (Y) → A, then shift AQ right one bit position.

Q-Register	
1 0	1 0 ...zeros..0 0 0
A_0	Q_{23}... etc.... Q_1, Q_0

After MUL STEP 3

(AQ) is as follows: After LSH is executed.

(Answer is in the Accumulator.)

Accumulator	Q-Register
0..zeros..... 0 1 0 1 0 0	0 0..zeros..0 0 0
A_{23}.. etc.. A_4, A_3, A_2, A_1, A_0	Q_{23}.. etc... Q_1, Q_0

Instruction	Operation Code	Mnemonic Code	Symbolic Description
DIVIDE STEP	61	(DIV)	Quotient → Q Remainder → A

a. INSTRUCTION DESCRIPTION - The divisor is in the storage location specified by the low-order 15 bits of the DIV instruction. Shift A and Q (as one open-ended 48 bit register) to the left by one bit position. Then subtract the content of the specified storage location from the content of the Accumulator.

If the content of the Accumulator is positive ($A_{23} = 0$), then replace the content of the lower bit of the Q-register (Q_0) with a one.

If the content of the Accumulator is negative ($A_{23} = 1$), then replace the content of the lower bit of the Q-register (Q_0) with a zero, and add the content of the specified storage location to the content of the Accumulator.

This instruction is only a phase of true division and therefore is called division step. To complete a division, program as follows:

1. Clear the Accumulator.
2. Place the number representing the dividend in the Q-register.
3. Repeat the divide step 24 times.

To divide a 48-bit number by a 24-bit number program as follows:

1. Place the high-order bits of the dividend in the accumulator. NOTE- the divisor must be greater than the contents of the accumulator or an erroneous result will be obtained.

2. Place the low-order bits of the dividend in the Q register. Q_{23} may contain a one in this case.

3. Repeat the divide step 24 times.

b. SPECIAL FUNCTIONS - The quotient will be found in the Q-register with the remainder being located in the Accumulator. <u>This instruction is valid only for positive quantities.</u>

1. The REPEAT feature is used as noted on the following page.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 64 microseconds; repeat, 44 microseconds.

DIVIDE STEP
EXAMPLE 1 (If the dividend is unknown.)

Accumulator	Q-Register
zeros	the value of S
A_{23}.....etc....A_0	Q_{23}...etc....Q_0

Before DIV executed

PROBLEM: $\frac{S}{R} = Q$

Program Format

M.L.	OP.	B	M	Addr.	
0010	ENA	0	0	0000	Clear A
0011	LDQ	0	0	0020	Dividend Q
0012	RPT	0	0	0030	Since the dividend is unknown, step the DIV 24 times.
0013	DIV	0	0	0021	
0020 ..... S					(Dividend)
0021 ..... R					(Divisor)

Shift (AQ) left one bit position, then (A)-(Y) $\rightarrow A_f$; if $A_{23} = 1$, add (Y) to (A), put a zero in the low-order bit of Q; if $A_{23} = 0$, put a 1 in the low-order bit of Q.

(The above applies to each of the 24 steps.)

(AQ) is as follows: (Answer is in the Q-register.)

Accumulator	Q-Register
remainder	quotient
A_{23}.....etc.....A_0	Q_{23}.....etc.....Q_0

DIVIDE STEP

EXAMPLE 2 (If the dividend is known.)

Accumulator	Q-Register	After
all zeros	1 1 1 0 0 ..zeros... 0 0	LSH
A_{23}.....etc....A_0	Q_{23}.....etc..... Q_0	executed

PROBLEM: $\frac{7}{3} = 2$

Program Format

M.L.	OP.	B	M	Addr.	
0010	ENQ	0	0	0000	Clear Q
0011	LDA	0	0	0020	Dividend A
0012	LSH	0	0	0055	Dividend Q, A is cleared
0013	RPT	0	0	0003	Since the dividend is known to be 3 significant bits, step the DIV 3 times
0014	DIV	0	0	0021	(divisor in Y)
0020	00	0	0	0007	(dividend)
0021	00	0	0	0003	(divisor)

Q-Register

...1	1 1 0 ..zeros..0 0
A_0	Q_{23}.....etc..... Q_0

(After DIV step 1)

Q-Register

... 0	1 0 ..zeros... 0 1
A_0	Q_{23}....etc..... Q_0

(After DIV step 2)

Q-Register

...0 1	0 0 ..zeros..0 1 0
A_1, A_0	Q_{23}...etc.... Q_1, Q_0

(After DIV step 3)

(AQ) is as follows:

Accumulator	Q-Register
0 0 ..zeros..0 0 1	0 0 ..zeros.. 0 1 0
A_{23}....etc.....A_0	Q_{23}....etc.... Q_1, Q_0

(remainder in A) (quotient in Q)

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SELECTIVE CLEAR	62	(SCL)	$(A)_i \oplus L(A)(Y) \longrightarrow A_f$
			or
			If $(Y)_n = 1$, clear A_n

a. INSTRUCTION DESCRIPTION - Clear bits of the Accumulator corresponding to the 1's in the content of the storage location specified by the low-order 15 bits of this instruction.

b. SPECIAL FUNCTIONS

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Normal operation time is 40 microseconds; repeat, 20 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
EQUAL SKIP	64	(ESK)	If (A) = (Y), then SKIP

a. INSTRUCTION DESCRIPTION - Compare the content of the Accumulator with the content of the storage location, specified by the low-order 15 bits of this instruction; if equal, skip the next sequential instruction; if unequal, go to the next sequential instruction.

b. SPECIAL FUNCTIONS - The Accumulator is unchanged.

1. The REPEAT feature is available for use with this instruction until the skip condition is present.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word portion designator with this instruction.

c. TIMING - Normal operation time is 64 microseconds; repeat, 44 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
UNEQUAL SKIP	65	(USK)	If (A) $\neq$ (Y), then SKIP

a. INSTRUCTION DESCRIPTION - Compare the content of the Accumulator with the content of the storage location, specified by the low-order 15 bits of this instruction; if unequal, skip the next sequential instruction; if equal, go to the next sequential instruction.

b. SPECIAL FUNCTIONS - The Accumulator is unchanged.

1. The REPEAT feature is available for use with this instruction until the skip condition is present.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 64-microseconds; repeat, 44 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
GREATER SKIP	66	(GSK)	If (A) $\geq$ (Y), then SKIP

a. INSTRUCTION DESCRIPTION - Compare the content of the Accumulator with the storage location as specified by the low-order 15 bits of this instruction. If the content of the Accumulator is greater than or equal to the content of the specified storage location, skip the next sequential instruction. If the content of the Accumulator is less than the content of the specified storage location, go on to the next instruction.

b. SPECIAL FUNCTIONS - The Accumulator is unchanged.

1. The REPEAT feature is available for use with this instruction until the skip condition is present.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
LESS SKIP	67	(LSK)	If (A) < (Y), then SKIP

a. INSTRUCTION DESCRIPTION - Compare the content of the Accumulator with the content of the storage location as specified by the low-order 15 bits of this instruction. If the content of the Accumulator is less than the content of the specified storage location, skip the next sequential instruction. If the content of the Accumulator is greater than or equal to the content of the specified storage location, go to the next instruction.

b. SPECIAL FUNCTIONS - The Accumulator is unclanged.

1. The REPEAT feature is available for use with this instruction until the skip condition is present.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator with this instruction.

c. TIMING - Normal operation time is 60 microseconds; repeat, 40 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
EXTERNAL FUNCTION	70	(EXF)	Y→F

a. INSTRUCTION DESCRIPTION - Replace the content of the External Function Register (F) with the low-order 15 bits of this instruction as modified by B and initiate the next external function.

b. SPECIAL FUNCTIONS - This instruction provides a 15-bit signal to the External Function Register. The circuits of the external devices interpret the content of the register as one of the External Function Commands.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", has no meaning in this instruction.

c. TIMING - Normal operation time is 36 microseconds; repeat, 16 microseconds.

d. NOTE - If the external function code given is not a legitimate code, the computer will "hang up" on the next external function instruction.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
INPUT	71	(INP)	If m=1, 2, 3, 5, 6 or 7: I→Y If m=0 or 4: I→Q, then Q→Y

a. INSTRUCTION DESCRIPTION - Replace the content of the storage location, specified by the low-order 15 bits of this instruction, with the content of data now being read into the computer.

If m =0, clear Q -
--shift content of Q-register circularly to the left by 6 bit positions; then replace the low-order 6 bits of the Q-register with the low-order 6 bits of the content of the I-register. Repeat this sequence four times then replace the content of the specified storage location with the final content of the Q-register.

If m = 1, 2, 3, 5, 6, or 7, replace the low-order 7 bits of the specified third of the storage location with the content of the 7-bit Input Register.

If m = 4, clear Q -
--shift the content of the Q-register circularly to the left by 6 bit positions, then replace the low-order 6 bits of the Q-register with the low-order 6 bits of the Input Register. Repeat this sequence four times, then replace the low-order 15 bits of the specified storage location with the low-order 15 bits of the Q-register.

b. SPECIAL FUNCTIONS - When m=4 the high-order bits of the assembled Q-register (Q_{15} thru Q_{23}) are not stored in the storage location. The Q-register is always used as a buffer assembly storage for input when m=0 or 4.

This instruction cannot be executed on Command Step because an Input Fault will occur after the first step.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as explained on the preceding page.

c. TIMING - Operation time when m≠0 or 4 is 52 microseconds; repeat, 32 microseconds. When m=0 or 4 the operation time is 120 microseconds; repeat, 100 microseconds. These are minimum times. Actual timing depends on the speed of the input device.

4. If an external disconnect, such as an end-of-record condition on magnetic tape, occurs while characters are stored in the Q-register, the content of the specified storage location is not replaced with the final content of the Q-register.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
OUTPUT	72	(OUT)	If m=1, 2, 3, 5, 6, or 7: Y→O If m=0 or 4: Y→Q, then Q→O

a. INSTRUCTION DESCRIPTION - Write the content of the storage location specified by the low-order 15 bits of this instruction, with the selected external equipment.

If m = 0, replace the content of the Q-register with the content of the specified storage location; circularly-shift Q-register to the left by 6 bit positions, then replace the content of the output register with the low-order 6 bits of the Q-register; initiate output (repeat the second step four times).

If m = 1, 2, 3, 5, 6, or 7, replace the content of the Output register with the low-order 7 bits of the specified 8 bit section of the storage location and initiate output.

If m = 4, assemble the low-order 15 bits of the specified storage location in the Q-register and proceed similarly to m=0.

b. SPECIAL FUNCTIONS - Q-register is used as a buffer storage for output when m=0 or 4. This instruction cannot be used under command step because an output fault will occur after the first step.

1. The REPEAT feature is available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a word-portion designator.

c. TIMING - Operation time, if m≠0 or 4 is 40 microseconds; repeat, 20 microseconds. If m=0 or 4, operation time is 116 microseconds; repeat, 96 microseconds. These are minimum times. Actual timing depends on the speed of the output device.

4. To avoid a loss of one or more characters of output, it is necessary to have a sense jump loop to cause a sufficient time delay BETWEEN:

(1) output at slow speed (FLEXOWRITER) and output at high speed (MAGNETIC TAPE).

or BETWEEN:

(2) output on the FLEXOWRITER and selection of the FLEXOWRITER for input.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
TRACE JUMP	73	(TRJ)	

a. INSTRUCTION DESCRIPTION - Trace jump control is a bit register within the main control of the computer. If trace jump control is on, the next jump instruction which would cause a change in the normal sequence will become a return jump to zero and will turn off trace jump control.

Trace jump control is turned on once when the set trace jump switch is turned on. Thereafter, it is turned on by the Trace Jump instruction if the trace jump switch is on.

In addition to turning on trace jump control, this instruction performs a jump operation depending on the value of "m".

If m = 0, 1, 2, or 3, do a normal jump. If Set Trace Jump switch is on, turn on trace jump control.

If m = 4, 5, 6, or 7, do a return jump. If Set Trace Jump switch is on, turn on trace jump control.

b. SPECIAL FUNCTIONS - This is the only instruction that allows the computer to monitor all jumps. The monitoring routine starts at storage location one. If trace jump control is on, the first jump in the main program which would cause a change of normal instruction sequence will be a return jump to zero. This jump to zero stores the address of the next sequential instruction at storage location zero and transfers control to the monitoring routine which starts at location one.

The monitoring routine returns control to the main program by constructing the correct address for a trace jump instruction. This instruction is executed, trace jump control is turned on, and the main program continues as if it had not been interrupted.

1. The REPEAT feature is not available for use with this instruction.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used as a jump designator.

4. Trace mode exists when set trace jump switch, on the control panel, is manually selected.

c. TIMING - If m = 0, 1, 2, or 3, operation time is 40 microseconds. If m = 4, 5, 6, or 7, or if the TJC is already on, operation time is 56 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SENSE JUMP	74	(SNJ)	

a. INSTRUCTION DESCRIPTION - The Sense Jump instruction has eight options. Two options depend upon "m" and the condition specified by a previous External Function. Six options depend upon value of "m" and the present status of the input register, output register, or external function register.

If m = 0 and if the condition specified by a previous External Function Instruction exists, do a normal jump; if not, go to the next sequential instruction.

If m = 1 and if the Input Register has been loaded and it has not been read out, do a normal jump; if not, go to the next sequential instruction.

If m = 2 and if the Output Register is cleared (available for output), do a normal jump; If not, go to the next sequential instruction.

If m = 3 and if the External Function Register is available, do a normal jump; if not, go to the next sequential instruction.

If m = 4 and if the condition specified by a previous external function exists, do a return jump; if not, go on with the next sequential instruction.

If m = 5 and if the Input Register has been loaded and has not been read out, do a return jump; if not, go to the next sequential instruction.

If m = 6 and if the Output Register is clear (available for output), do a return jump; if not, go to the next sequential instruction.

If m = 7 and if the External Function Register is available, do a return jump; if not, go to the next sequential instruction.

b. SPECIAL FUNCTIONS - There are no register changes during this instruction. If the trace jump control is on, a return jump to zero sequence will replace either the normal jump or return jump sequence.

1. The REPEAT feature is available with this instruction, being terminated when the jump condition exists.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used to designate the jump conditions.

c. TIMING - Operation time for NO JUMP is 40 microseconds; when m=0, 1, 2, 3, and no trace jump, 40 microseconds; when m=4, 5, 6, 7, and trace jump, 56 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SELECTIVE JUMP	75	(SLJ)	

a. INSTRUCTION DESCRIPTION - This instruction has eight options depending upon the value of "m" and the present status of the three jump switches on the console. If any of the jump conditions listed below do not occur, continue with the present sequence. If the trace jump control is on, a return jump to zero sequence will replace either the normal jump or return jump sequence.

If m = 0, do a normal jump.

If m = 1 and jump switch 1 is on, do a normal jump.

If m = 2 and jump switch 2 is on, do a normal jump.

If m = 3 and jump switch 3 is on, do a normal jump.

If m = 4, do a return jump.

If m = 5 and jump switch 1 is on, do a return jump.

If m = 6 and jump switch 2 is on, do a return jump.

If m = 7 and jump switch 3 is on, do a return jump.

b. SPECIAL FUNCTIONS - When m=0 or 4, the jump switches are not effective. On all other options the switch must be on to produce the jump. If the switch is not on, the program will continue in the present sequence.

1. The REPEAT sequence is available for use with this instruction but is terminated if a jump condition exists.

2. The B-designator, "b", may be used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used to designate the jump conditions.

c. TIMING - Operation time for NO JUMP is 40 microseconds; repeat, 20 microseconds. Operation time for JUMP is 40 microseconds; for a return jump or trace jump the time is increased to 56 microseconds.

Instruction	Operation Code	Mnemonic Code	Symbolic Description
SELECTIVE STOP	76	(SLS)	

a. INSTRUCTION DESCRIPTION - This instruction has eight options depending upon the value of "m" and the present status of the three stop switches on the console.

If m = 0, do a normal jump,* then stop computation and indicate a stop.

If m = 1, do a normal jump,* then if optional stop switch 1 is on, stop computation and indicate a stop. If switch is off, do normal jump.

If m = 2, do normal jump,* then if optional stop switch 2 is on, stop computation and indicate a stop. If switch 2 is off, do normal jump.

If m = 3, do normal jump,* then if optional stop switch 3 is on, stop computation and indicate a stop. If switch 3 is off, do normal jump.

If m = 4, do return jump,* then stop computation and indicate a stop.

If m = 5, do return jump,* then if optional stop switch 1 is on, stop computation and indicate a stop. If switch 1 is off, do return jump.

If m = 6, do return jump,* then if optional stop switch 2 is on, stop computation and indicate a stop. If switch 2 is off, do return jump.

If m = 7, do return jump,* then if optional stop switch 3 is off, do return jump.

b. SPECIAL FUNCTIONS - Stop switches are located on the console switch panel and are manually operated. When m=0 or 4 the stop switches are not effective. On all other options the switch must be on to produce a stop.

* See d NOTE, page III-86.

If switch is not on, the computer will perform a normal or return jump but will not produce a stop. If the trace jump control is on, a return jump to zero sequence will replace either the normal jump or return jump sequence.

1. The REPEAT feature is available but is terminated whenever a jump occurs.

2. The B-designator, "b", is used with this instruction.

3. The INSTRUCTION MODIFIER, "m", is used to designate the stop conditions. (See preceding page.)

c. TIMING - If stop switch is not set, operation time for m=0, 1, 2, and 3 is 36 microseconds; for m=4, 5, 6, or 7 the time is 52 microseconds. If Stop switch is set, the time for m=1, 2, 3 is 29 microseconds; for m=4, 5, 6, or 7 the time is 41 microseconds.

d. NOTE - The term "do a normal jump" means set P to obtain next instruction from Y. At the stop (if any) the Selective Stop instruction will be in U and the jump address will be in P.

The term "do a return jump" means store the return address at the specified location. Set P to obtain the next instruction at the specified location plus one.

BOGART

PROGRAMMERS MANUAL

SECTION IV

EXTERNAL FUNCTION CODES

External function codes, originated by external function instructions, control the peripheral equipment of BOGART. When an external function instruction is interpreted in the main control, the low-order 15 bits of the instruction is sent to the external function register where it becomes the external function code.

The external function register transmits an external function signal to all peripheral equipments. The coded direction, as listed below, is accepted by the control unit of the equipment for which it is intended. Upon acceptance, the control unit returns a function resume signal which allows the computer to transmit or process more information. If an unlisted external function code is programmed, it will not be accepted, a function resume signal will not be returned, and the computer will stop.

While the external function code is being transmitted to the external equipment, the external function register is locked out from the computer. If a subsequent external function instruction is programmed while the register is locked out, the execution of that instruction will be delayed until the lock-out condition is removed. Some functions remain locked out until an input or output instruction is executed. Thus, if an external function is attempted during lockout and before the input or output instruction is performed, the computer will "hang up." The lock-out time of each external function is presented in the description of external function codes.

A. List of EXTERNAL FUNCTION CODES

0XXXX EXTERNAL CONTROL

01XXX External relay control

01101 Energize relay one
01102 Energize relay two

01103	Energize relay three
01201	Release relay one
01202	Release relay two
01203	Release relay three
02XXX	Flexowriter control
02100	Start motor
02200	Stop motor
03XXX	Paper tape control
03101	Start Ferranti motor
03102	Start Teletype reader motor
03104	Start Teletype punch motor
03107	Start all motors
03201	Stop Ferranti motor
03202	Stop Teletype reader motor
03204	Stop Teletype punch motor
03207	Stop all motors
04XXX	Converter control
04000	Turn off I/O indicator of last selected unit
04001	Write tape mark on last selected tape
04002	Rewind last selected tape
04003	Turn on I/O indicator of last selected unit
04004	Backspace last selected tape
04100	Select card reader
04200	Select tape unit 0 For control only
04201	Select tape unit 1 For control only
04202	Select tape unit 2 For control only
04203	Select tape unit 3 For control only
04204	Select tape unit 4 For control only
04205	Select tape unit 5 For control only
04206	Select tape unit 6 For control only
04207	Select tape unit 7 For control only
04210	Select tape unit 8 For control only
04211	Select tape unit 9 For control only
04300	Select card punch For control only
04400	Select printer For control only

1XXXX	INPUT SELECTION
12XXX	Flexowriter input
12000	Slect Flexowriter input
13XXX	Paper tape input
13001	Select Ferranti reader input
13002	Select Teletype reader input
14XXX	Converter input
14100	Read cards
14200	Read tape unit 0 (coded)
14201	Read tape unit 1 (coded)
14202	Read tape unit 2 (coded)
14203	Read tape unit 3 (coded)
14204	Read tape unit 4 (coded)
14205	Read tape unit 5 (coded)
14206	Read tape unit 6 (coded)
14207	Read tape unit 7 (coded)
14210	Read tape unit 8 (coded)
14211	Read tape unit 9 (coded)
14240	Read tape unit 0 (binary)
14241	Read tape unit 1 (binary)
14242	Read tape unit 2 (binary)
14243	Read tape unit 3 (binary)
14244	Read tape unit 4 (binary)
14245	Read tape unit 5 (binary)
14246	Read tape unit 6 (binary)
14247	Read tape unit 7 (binary)
14250	Read tape unit 8 (binary)
14251	Read tape unit 9 (binary)
2XXXX	OUTPUT SELECTION
22XXX	Flexowriter output
22000	Select Flexowriter output
23XXX	Paper tape output
23000	Select Teletype punch output

24XXX	Converter output
24200	Write tape unit 0 (coded)
24201	Write tape unit 1 (coded)
24202	Write tape unit 2 (coded)
24203	Write tape unit 3 (coded)
24204	Write tape unit 4 (coded)
24205	Write tape unit 5 (coded)
24206	Write tape unit 6 (coded)
24207	Write tape unit 7 (coded)
24210	Write tape unit 8 (coded)
24211	Write tape unit 9 (coded)
24240	Write tape unit 0 (binary)
24241	Write tape unit 1 (binary)
24242	Write tape unit 2 (binary)
24243	Write tape unit 3 (binary)
24244	Write tape unit 4 (binary)
24245	Write tape unit 5 (binary)
24246	Write tape unit 6 (binary)
24247	Write tape unit 7 (binary)
24250	Write tape unit 8 (binary)
24251	Write tape unit 9 (binary)
24300	Punch cards
24400	Print on tabulator
3XXXX	EXTERNAL FAULT
4XXXX	SENSE SELECTION
41XXX	Relay control
41001	Sense relay one
41002	Sence relay two
41003	Sense relay three
41011	Sense switch one
41012	Sense switch two
41013	Sense switch three
43XXX	Paper tape control
43001	Sense no-tape in Ferranti reader
43002	Sense no-tape in Teletype reader

44XXX	Converter control
44000	Sense any of the following
44001	Sense Rd/Wr check
44002	Sense I/O indicator
44003	Sense short transfer
44004	Sense long transfer
5XXXX	EXTERNAL FAULT
6XXXX	EXTERNAL FAULT
7XXXX	EXTERNAL FAULT

B. Description of EXTERNAL FUNCTION CODES

0XXXX EXTERNAL CONTROL

These external functions transmit power signals and turn on and off certain devices. The motors on all external equipments may be turned on simultaneously but only one input and one output may be operated at the same time.

01XXX EXTERNAL RELAY CONTROL

The three relays included in BOGART allow for expansion of the external control system. The computer may energize or release these relays separately or collectively. The relay contacts available at a connector on BOGART, may be plugged to perform desired functions.

01101	Energize relay one
01102	Energize relay two
01103	Energize relay three

Energize the specified relay to make contact. This condition remains until a release-relay external function is given.

The F-register is locked out for 30 microseconds.

01201	Release relay one
01202	Release relay two
01203	Release relay three

Deenergize the specified relay. If the specified relay is not energized, the function is ignored.

The F-register is locked out for 30 microseconds.

02XXX FLEXOWRITER CONTROL

02100 Start Flexowriter motor

Turn on the Flexowriter motor.
The F-register is locked out for 30 microseconds.

02200 Stop Flexowriter motor

Turn off the Flexowriter motor.
The F-register is locked out for 30 microseconds.

03XXX PAPER TAPE CONTROL

03101 Start Ferranti motor
03102 Start Teletype reader motor
03103 Start Teletype punch motor

Turn on the motor of the specified paper-tape equipment.
The F-register is locked out for 30 microseconds.

04XXX CONVERTER CONTROL

The converter control functions enable BOGART to generate the required control signals which specify the operation of the IBM equipment.

A unit connected to the converter is selected for control whenever it is input from, output to, as well as when one of the external function codes below is given. A unit once selected remains selected for control until another unit connected to the converter is selected for input, for output, or for control.

04000 Turn off I/O indicator of last selected unit

Turn off the I/O indicator of the last selected IBM unit which was chosen by the select external function. Only one IBM unit may be connected to the converter for control purposes at any time.

The F-register is locked out for 80 microseconds.

04001 Write tape mark on last selected tape

This function records an end-of-file mark on the last selected tape unit. If card punch, card reader, or printer has been selected since the last tape unit was selected, no EXF resume signal occurs. The computer will hang up when the next external function instruction is given.

The F-register is locked out until the end-of-file mark has been written (50 milliseconds).

04002 Rewind last selected tape

Rewind to the starting point and prepare for reading or writing.

If card reader, punch, or printer have been selected since the last tape device was selected, no EXF resume signal occurs. The computer will hang up when the next external function instruction is given.

The F-register is locked out for 80 microseconds, if the tape is already rewound. Otherwise, the F-register is locked out for 40 milliseconds.

04003 Turn on I/O indicator of last selected unit

Turn on the I/O indicator of the unit which is now connected to converter for control purposes.

The F-register is locked out for 80 microseconds.

04004 Backspace last selected tape

Backspace last selected tape one unit record and prepare to continue operation in original read or write mode. If card punch, card reader, or printer has been selected since the last tape unit was selected, no EXF resume signal occurs. The computer will hang up when the next external function instruction is given.

The F-register is locked out until the backspace action is completed. Estimated lock-out time is 60 + .067 N milliseconds, where N = number of characters in the unit record.

04100 Select card reader

Connect the card reader to the converter for control purposes only. Drop any other selection.

The F-register is locked out for 40 microseconds.

04200 Select tape unit 0
04201 Select tape unit 1
04211 Select tape unit 9

Connect the tape unit specified to the converter for control purposes only. Drop any other selection.

The F-register is locked out for 40 microseconds.

04300 Select card punch

Connect the card punch to the converter for control purposes only. Drop any other selection.

The F-register is locked out for 40 microseconds.

04400 Select printer

Connect the printer to the converter for control purposes only. Drop any other selection.

The F-register is locked out for 40 microseconds.

1XXXX INPUT SELECTION

An external function of this group indicates that a new input device is to be connected to the seven-bit input register. Drop the previously connected input device from the input register connection.

12XXX FLEXOWRITER INPUT

This external function permits automatic keyboard entry of information. If the flexowriter tape reader is turned on, information is read, the keyboard is actuated, and an automatic input is provided. A flexowriter input selection drops the typewriter output selection as well as other input selections. The Flexowriter will not punch tape when it is being used for input.

12000 Select keyboard input

Connect the Flexowriter to the input register. This instruction effects a "pseudo input enable" which calls for one character of information by turning on the Input light on the Flexowriter. If the tape reader has been manually selected, the first valid Flexowriter character is read from tape to the input register. The F-register is locked out for 30 microseconds while the Flexowriter selection is stored by typewriter control. Note that the input does NOT go through the Flexowriter plug-board.

13XXX PAPER TAPE INPUT

This external function transfers information from paper tape to internal storage. The paper tape specified by the function is set in motion by the function code. The F-register is locked out for 30 microseconds. When the first character is read into the input register, the tape is halted and the register returns an input resume signal to the tape reader. The next input instruction reads one

character from the input register and then the tape moves to supply the other necessary input characters. If only one character is called for by the input instruction, the tape is not placed in motion.

If a "no tape" condition is detected, an external disconnect signal is sent to the main control.

13001 Select Ferranti reader input

Connect the Ferranti reader to the input register and read one character to the input register in anticipation of the following input instruction. Drop any other input selection.

The F-register is locked out for 30 microseconds.

13002 Select Teletype input

Connect the Teletype high-speed reader to the input register. Read one character to the input register in anticipation of the following input instruction. Drop any other input selection.

The F-register is locked out for 30 microseconds.

14XXX CONVERTER INPUT

Connect a particular IBM input device to the converter and designate whether the characters read are to be routed via the substitution plugboard for coded input information or via the transposition plugs for binary information. The next input instruction actuates the selected (connected) input device. The low-order six bits of the characters read by the input device are transmitted to the input register through the correct plugboard. An even or odd parity check is made on each character as required by the input device. If an error is discovered in the horizontal or vertical redundancy check, the "read-write" (Rd/Wr) error register is set.

Because of the high rate of information transfer, a repeated input instruction must be used for records of more than one word. The word is one character when "m" of the input instruction is 1, 2, 3, 5, 6, or 7, and four characters when "m" is 0 or 4.

An external function instruction selecting the appropriate unit for input must be given before each input or repeated input instruction involving a unit connected to the converter. An exception is consecutive input (or output) instructions with an m-designator of 1, 2, 3, 5, 6, or 7.

An external function given between a converter input select and its associated input order will cause the computer to "hang up".

When the number of characters called for by the repeated INPUT instruction exceeds the number of characters in the record, the end-of-record signal from the input device terminates the repeat sequence and/or the input. The long-transfer register is set and the external function register is released from interlock. If the

number of characters called for is less than the number in the unit record, the occurance of another character from the record sets the short-transfer register. The end-of-record signal releases the external function register from interlock.

When an input or output unit is connected to the converter, the I/O indicator of that unit is recognized by the sense jump instruction. The read-write error, long transfer, short transfer, and I/O indicator may be recognized individually or collectively by the sense jump instruction.

The read-write error register is cleared by the next read or write external function signal to the converter. The long and short transfer registers are cleared by the next read external function to the converter. The I/O indicator is turned ON by an external function signal. The indicator does not turn OFF when the external unit is disconnected. It it is turned OFF manually at the external unit or by an external function signal.

A converter input selection drops any other input selection from the input register.

All inputs from IBM devices are selected by both an external function instruction and an input or repeated input instruction for each unit record read.

14100 Read cards

Connect the IBM 714 card reader to the converter and connect the converter to the BOGART input register. The next input instruction reads a unit record from the 714 record storage unit into the BOGART storage. Without any signals from BOGART, the 714 then reads a card to refill its record storage unit. Regardless of the length of the input called for, only one record is read from the record storage unit to BOGART storage. Approximately 240 milliseconds are requared to reload the record storage unit and any subsequent input instruction is delayed until the record storage unit is refilled.

The F-register is locked out until end of subsequent input instruction.

The first card is read into storage when the "start" button on the card reader is pushed twice.

14200	Read tape unit 0 (coded)
14201	Read tape unit 1 (coded)
14211	Read tape unit 9 (coded)

Connect the specified tape unit to the converter and connect the converter to the input register. Prepare the converter to transmit information to the input register via the substitution plugboard. The tape unit is not started until the subsequent input instruction is given. Time of the input instruction is 10 milliseconds for the tape to reach normal speed and 67 microseconds for each character read.

The F-register is locked out until the end of the record is read.

14240	Read tape unit 0 (binary)
14241	Read tape unit 1 (binary)
14251	Read tape unit 9 (binary)

Connect the specified tape unit to the converter and connect the converter to the input register. Prepare the converter to transmit information to the input register via the transposition plugs on the substitution plugboard. The tape unit is not started until the subsequent input instruction is given. Time of the input instruction is 10 milliseconds for the tape to reach normal speed and 67 microseconds for each character read.

The F-register is locked out until the end of record is read.

2XXXX OUTPUT SELECTION

Connect an output device to the output register and drop the previously selected output connection.

22XXX SELECT TYPEWRITER OUTPUT

Connect a typewriter to the output register and drop the previously selected output device. When this type of external function is followed by an output instruction, the typewriter prints the character recorded in the output register as translated by the typewriter substitution plugboard. Special characters may be typed or deleted by plugging unique code combinations. An output resume signal is generated at the end of each typewriter operation. Information typed out may also be recorded in Flexowriter code by the Flexowriter punch.

22000	Select typewriter printer output

Connect the Flexowriter equipment to the output register. If the Flexowriter was connected to the input register, drop that

selection. The next output instruction transmits information from the output register to the Flexowriter substitution plugboard and operates the keys of the Flexowriter. The maximum rate of output is one character every 102 milliseconds.

The F-register is locked out for 30 microseconds.

23XXX SELECT PAPER TAPE OUTPUT

This type of external function transfers information from internal storage to perforated paper tape.

23000 Select Teletype punch output

Connect the Teletype high-speed paper tape punch to the output register.

The F-register is locked out for approximately 30 microseconds.

The next output instruction punches information on the paper tape. If m of the output instruction is 1, 2, 3, 5, 6, or 7, all seven levels of the tape are perforated according to the pattern of the low-order seven bits of the output register. The computer will hang up when the next external function instruction is given.

24XXX CONVERTER OUTPUT

Connect an IBM output device to the converter and designate whether the characters to be written are to be routed via the substitution plugboard for coded output or via the transposition plugs for binary information. The low-order six bits from the output register are written. The next output instruction starts the write sequence of the selected output unit. The completion of the output instruction (repeated or single) signals an end-of-record to the output equipment. For magnetic tape, an end-of-record gap is recorded. Refer to IBM manuals for information on the card punch and the printer.

Because of the high rate of information transfer necessary for the IBM equipment, and the automatic end-of-record feature of the output instruction, records of more than one BOGART word in length must be transmitted by a repeated output instruction. The word is one character when m of the output instruction is 1, 2, 3, 5, 6, or 7 and four characters when m equals 0 or 4. For m = 4, the information recorded consists of nine zeros plus the low-order 15 bits of the word.

If the output device recognizes a writing error, the read-write error register is set.

Each output operation with IBM equipment must consist of an output selection plus an output instruction repeated output instruction or several consecutive output instructions with an M-field of 1, 2, 3, 5, 6, or 7.

24200	Write tape unit 0 (coded)
24201	Write tape unit 1 (coded)
24211	Write tape unit 9 (coded)

Connect the specified tape unit to the converter in write status and connect the converter to the output register. Prepare the converter to accept information from the output register via the substitution plugboard. The tape unit is not started in motion until the subsequent output instruction is given. When the tape reaches operating speed, it accepts information from the computer. The end of the output instruction causes an end-of-record gap to be written on the tape. Records may be of any length. Time of the output instruction is 10 milliseconds for the tape to reach normal speed and 67 microseconds for each character written.

The F-register is locked out until the output instruction is accomplished.

24240	Write tape unit 0 (binary)
24241	Write tape unit 1 (binary)
24251	Write tape unit 9 (binary)

Connect the specified tape unit to the converter in write status and the converter to the output register. Prepare the converter to accept information from the output register via the transposition plug for binary output. The tape unit is not started in motion until the subsequent output instruction is given. When the tape reaches operating speed, it accepts information from the computer. The end of the output instruction causes an end-of-record gap to be written on the tape. Records may be of any length. Time of the output instruction is 10 milliseconds for the tape to reach normal speed and 67 microseconds for each character written.

The F-register is locked out until the output instruction is accomplished.

24300 Punch card

Connect the card punch to the converter and the converter to the output register. The converter accepts information from the output register via the substitution plugboard. The subsequent output instruction causes a unit record to be stored in the record storage unit of the card punch. Then the information is punched in the IBM card.

The F-register is locked out until the end of the output instruction.

24400 Print on tabulator

Connect the tabulator to the converter and the converter to the output register. The converter accepts information from the output register via the substitution plugboard. The subsequent output instruction causes a unit record to be stored in the tabulator record storage unit. Then the tabulator prints the information.

The F-register is locked out until the output instruction is accomplished.

3XXXX EXTERNAL FAULT

4XXXX SENSE SELECTION

An external function of this group indicates that a new external unit is to be connected to the external sense line. Thus the function allows the sense jump instruction to act according to the condition of any external unit. A sense selection remains in effect until replaced by a different sense selection. To monitor only one external condition, one sense selection will suffice for the entire program. The combination of a sense selection followed by a sense jump instruction enables the computer to act on the condition of the external equipment. One sense selection suffices for any number of sense jump instructions if only one condition is being examined.

F-register locked out for 30 microseconds.

41XXX SENSE RELAY AND SWITCH CONTROL

These sense selections enable the sense jump to interrogate the state of any one of three switches and/or three relays. The computer controls the relay operations through external control functions. The switches may be set by external equipment.

41001 Sense relay one
41002 Sense relay two
41003 Sense relay three

If the specified relay is actuated, perform the jump operation. Otherwise, continue with the present sequence of instructions.

41011 Sense switch one
41012 Sense switch two
41013 Sense switch three

If the specified switch is actuated, perform the jump operation. Otherwise, continue with the present sequence of instructions.

43XXX SENSE PAPER TAPE CONTROL

These sense selections enable the computer to examine the paper tape supply in the paper tape readers.

43001 Sense no-tape in Ferranti reader

If the No-tape switch is up, indicating that there is no paper tape in the Ferranti reader, perform the jump operation. Otherwise, continue with the present sequence of instruction.

43002 Sense no-tape in Teletype reader

If the No-tape switch is up, indicating that there is no paper tape in the Teletype paper tape reader, perform the jump operation. Otherwise, continue with the present sequence of instructions.

44XXX SENSE CONVERTER CONTROL

These sense selections enable the computer to examine the converter bit registers and the I/O indicators of the various IBM input/output equipments.

44000 Sense any condition (converter sense selection only)

If any of the converter control conditions (Rd/Wr check, I/O indicator, short transfer, or long transfer) would cause a jump operation, perform the operation. Otherwise, continue with the present sequence of instructions.

This sense selection enables the program to check if there is any condition existing in the converter which requires program attention. Usually the jump then leads to a sequence of program tests to find the conditions which caused the jump to occur.

44001 Sense Rd/Wr check

If an error occurred in the last reading or writing operation via the converter unit, as indicated by the Rd/Wr check register, perform the jump operation. Otherwise, continue with the present sequence of instructions.

44002 Sense of I/O indicator

If the I/O indicator of the particular IBM equipment connected to the converter is on, perform the jump operation. Otherwise, continue with the present sequence of instructions.

44003 Sense short transfer

If the short transfer register was set by the last input operation, perform the jump operation. Otherwise, continue with the present sequence of instructions.

The short transfer register is set if there are more characters in the unit record than are called for by the input instruction.

44004 Sense long transfer

If the long transfer register was set by the last input operation, perform the jump operation. Otherwise, continue with the present sequence of instructions.

The long transfer register is set if there are fewer characters in the unit record than are called for by the input instruction.

BOGART

PROGRAMMERS MANUAL

SECTION V

OPERATING PROCEDURES

A. Control Panel

The BOGART control panel, located on the console below the indicator display and maintenance panel, contains 2 toggle switches, a low-speed oscillator, 10 lever switches, 12 indicator lights and a running-time meter.

1. Toggle Switches

These switches are for maintenance purposes only and should be in the down position at all times.

2. Low-Speed Oscillator

This knob controls the clock-pulse rate when the machine is in low-speed operation. The speed is increased by turning the knob clockwise.

3. Set Trace Jump

This lever switch sets the trace jump circulation bit represented by TJC on the indicator display panel. This control may be used to monitor jumps within a given program.

4. Running-Time Meter

This meter, calibrated in tenths of a second, records machine operating time up to 9,999 seconds. It may be manually reset to zero.

5. Jump Switches

These three lever switches are locked ON in the up position, momentarily ON in the down position, and OFF in the normal or middle position. Above each jump switch is an indicator lamp.

When a Jump switch is ON, the corresponding indicator lamp lights. A SELECTIVE JUMP instruction is executed if one of the following conditions is met:

a. Select Jump switch 3 is ON, and "m" of the SELECTIVE JUMP instruction is 3 or 7.

b. Select Jump switch 2 is ON, and "m" of the SELECTIVE JUMP instruction is 2 or 6.

c. Select Jump switch 1 is ON, and "m" of the SELECTIVE JUMP instruction is 1 or 5.

If the above conditions do not occur, the program continues in its normal sequence. The programmer decides whether or not the operator is to turn on one or more Jump switches (see page III-83).

6. Fault Light

When any fault is sensed by the computer, it stops and the Fault Lamp lights.

7. Stop Switches

These three lever switches are ON in the up position, momentarily ON in the down position, and OFF in the normal or middle position. Above each switch is an indicator lamp. The computer will stop and the corresponding indicator lamp will light if one of the following conditions is met:

a. Select Stop switch 3 is ON, and "m" of the SELECTIVE STOP instruction is 3 or 7.

b. Select Stop switch 2 is ON, and "m" of the SELECTIVE STOP instruction is 2 or 6.

c. Select Stop switch 1 is ON, and "m" of the SELECTIVE STOP instruction is 1 or 5.

If the above conditions do not occur, the program continues in its normal sequence. The programmer determines when the operator is to turn on one or more Stop switches. To deactivate

This Selective Stop operation, the Selective Stop switch must be returned to the normal or middle position.

8. Stop Light

This light has no corresponding lever switch. It lights whenever the computer stops in an Unconditional Stop situation.

9. Mode of Operation Switches

There are three of these switches: P T Load/Master Clear switch, Instruction/Command Step switch, and Run/Step switch.

a. P T Load/Master Clear switch - This switch is locked ON in the P T Load or up position, momentarily ON in the Master Clear or down position, and OFF in the normal position.

(1) P T Load - When the switch is in this position the computer is prepared to accept coded paper tape input. The P T Load selection automatically issues a master clear signal. The indicator corresponding to this switch lights only when the switch is ON.

(2) Master Clear - When the switch is in this position all computer registers are reset to zero. Master Clear does not affect the magnetic core memory. It is only active when the computer is stopped. If the Run switch is activated after a Master Clear, the computer obtains its first instruction from memory location (0000) and proceeds to operate.

(3) Normal - This position indicates the normal operating condition of the computer.

b. Instruction/Command Step switch - The indicator corresponding to this switch lights whenever the mode of operation is other than high speed.

(1) Instruction Step - This mode of operation is activated when the switch is in the UP position. During this mode, if the Step switch is activated, the computer executes the instruction, reads the next instruction and stops. If the Run switch is

activated, the computer executes the instruction, reads the next instruction, and continues at a rate determined by the low-speed oscillator setting.

(2) High Speed - This mode of operation is activated when the switch is in the normal position. During this mode, if the Run switch is activated, the computer proceeds at high speed until a programmed stop comes into control or until the Step switch is activated. When the Step switch is activated, the computer stops after the execution of the instruction in control.

(3) Command Step - This mode of operation is activated when the switch is in the down position. During this mode the computer executes each command step of an instruction as the Step switch is activated. If the Run switch is activated, the computer executes commands according to the Low-Speed Oscillator setting.

c. Run/Step switch - Both UP and Down positions of this switch are momentary.

(1) Run - In this position, the switch starts the machine. All push-buttons on the indicator display panel are inoperative when the computer is running. If the High-Speed mode of operation has been selected and the Run switch activated, the computer procedes at high speed. With the Instruction Step or Command Step mode of operation, the instructions are stepped according to the low-speed oscillator setting. If Paper-Tape Load operation is in effect, activate the Run switch to load paper tape onto the computer.

(2) Step - When the switch is in this position, regardless of mode of operation, the computer stops after completing the next instruction or command. If the Step switch is pressed again, the computer executes one more unit (whether the unit be an instruction or a command) and then stops. This process is repeated every time the Step switch is activated. If the mode of operation.

is Run and the Step switch is activated, the computer stops after the execution of the instruction in control. Whatever condition is set on the control panel becomes effective when the Run is reset.

B. Indicator Display and Maintenance Panel

There are 19 internal registers. The content of each register on the indicator display panel has a binary representation, shown by two neon lights per bit. The upper lighted neon represents "1", the lower, represents "0". Whenever the computer is "master cleared", these registers are set to zero.

1. Maintenance Registers

These registers indicate command operations.

2. Load Mode Operation Registers

During Load Mode operation several of the registers are used for special purposes, as follows:

a. P-register - During the Load Mode operation, the P-register is the Load Check Counter. The P-register steps once each time a check address is correct.

b. During the Load Mode operation, Q-register becomes a buffer register. It is cleared each time the U-register advances once.

c. The Accumulator is used as a compare register for the Check Address and Load Address.

d. During the Load Mode operation, the U-register is set up as the storage address register.

C. Program Load Mode

Seven-level paper tape is used as input to BOGART. Levels 6, 1, 2, 3, 4, and 5 are information levels. Level-7 coding controls tape input to the computer. The 24-bit machine word is represented by four 6-bit frames of tape. A seventh-level punch is always above the fourth frame of a machine word.

The first of a group of information words, an insert address, specifies the storage location at which the next word is to be stored. The computer identifies this address by a seventh-level code designation. The seventh-level hole positions are designated a, b, c, and d, reading from the first or lead column.

Levels	Numeric Value	Frames																													
		d	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d	a	b
7		0	0			0				0				0				0				0	0			0						
6	(4)					0				0	0			0												0						
1	(2)				0				0					0												0						
2	(1)				0			0	0	0	0			0								0				0	0					
Feed holes		o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
3	(4)				0	0			0	0																0						
4	(2)					0	0			0																	0					
5	(1)				0	0																				0						

Figure 2. Sample of Punched Paper Tape.

The insert address is indicated by an a and d punched in the seventh level. When the d is read, the computer interprets the information in that group of four frames as the storage address for the information in the next four frames.

The word following the insert-address word is an enter-data word which may be followed by many similar words. It is represented by a d punched in the seventh level which orders the computer to write the word in the storage location previously specified by the U-register. The U-register then advances one count.

The last word in an information series is a check-address word, designated by a, b, and d punched in the seventh level. This word verifies that the preceding sequence of data has been loaded into the correct storage locations.

Notice that all words have the d punched in the seventh level. The tape reader reads the holes punched in the command level of any word plus the d hole of the previous word. This previous d hole is the d prime (d') hole. The insert address coding, therefore, is d'ad, the enter-data coding is d'd, and the check-address coding is d'bd.

The insert address in this particular example is 3547. The first word of information is 12345670, the second is 00000040, and the last enter-data word is 00001000. The check address is 00003552. When the tape is read by the computer, the information is placed in the storage in the following order.

Storage location	Content
3547	12345670
3550	00000040
3551	00001000
3552	previous content in storage

The check address is one greater than the last information address. (see Page V -7a)

D. Starting Operation

1. Turn on photo reader.
2. Set tape on the reader.
3. Select PER on tape console.
4. Place the P T Load switch in the UP position.
5. Place the Run/Step switch in the momentary UP position. The light above the P T Load switch lights. To stop the process, place the Run/Step switch in the DOWN position.
6. After the tape has been loaded place the Master Clear switch in the momentary down position.
7. Manually set the P-register to the address where the first instruction of the program to be executed is found.
8. If Select Stop switches and Select Jump switches are to be set, place them in the UP position, otherwise, DO NOT TOUCH.
9. Put the machine in High-speed mode of operation.
10. Place the Run switch in the momentary UP position.

NOTE - If storage location 7777 is loaded into the photo-electric reader from paper tape, the computer will then write the next frame of the paper tape into the low order third of storage location ØØØØ (clearing bit 7 of that storage location) and then "hang up".

MECCANO SUPPLEMENT TO THE BOGART PROGRAMMER'S MANUAL

Compiled by:
W. H. Martin
REMP 13
1 March 1958

Source:
Meccano Instruction Book
and Equation File,
Sperry-Rand Corporation

ABSTRACT

Meccano was constructed to provide a flexible means of analyzing analog information on a digital computer. The apparatus consists essentially of an analog-to-digital converter and a clock counter with a variable counting rate, both of which may be used as auxiliary inputs to the Bogart Computer. A provision for sensing the states of relays is also provided.

This supplement contains operating and programming instructions for Meccano.

CONTENTS

1. General Description.

The Meccano unit provides an auxiliary input to the Bogart computer. Such input information may be of three types:

(a) Analog - from an analog-to-digital converter incorporated in Meccano,

(b) Mechanical - from some externally keyed source,

(c) Real-Time - from the free-running counter within Meccano.

Meccano contains two seven-bit registers, the Text and Clock registers, both of which can communicate directly with the Input Register of Bogart. Inputs to the Text Register may be from either the analog-to-digital converter or a set of external relay contacts. The Clock Register obtains its input from a free-running counter with a variable counting rate.

The maximum rate at which information can be transferred from Meccano to Bogart is one transmission every 64 microseconds. (Repeated Input Instruction).

Changes which occur in the bits of the Text Register are reflected, under one of two switch-controlled options, in the one-bit Change Register of Meccano.

2. Clock Register.

The Clock Register is a seven-bit register-counter. The rate at which the counter is advanced (Advance Clock pulse) is determined by the position of the Advance Clock switch, S03, which selects increments of powers of two from 4 to 32,768 microseconds. These increments are the outputs from a fourteen-stage, free-running binary counter, called the variable time input. The rate at which Clock overflows occur varies with the position of S03 from a minimum of .0005 seconds to a maximum of approximately 4.2 seconds.

3. Text Register.

Overflow Bit. The high-order bit of the Text Register, has a special function. It is set whenever an overflow occurs in the Clock Register; that is, whenever the Clock advances from 177 to 000. It is

cleared only after a Text-to-Bogart input instruction has been completed, thereby insuring that every such overflow is duly recorded.

The clock operates continuously, so that bit 6 of the Text Register will be set, and will remain set, after each Clock overflow even if neither register has been selected. If it is desired to separate the overflow bit from the ~~five~~ six low-order data bits, this must be done in Bogart under program control.

Input Options. Input to the Text Register is controlled by the Text Input Switch (S01) on the Meccano control panel. With the switch in KEY position, the inputs to bits 0 through 5 come from the phone jacks (labeled J00, J01, ... , J05) on the back of the Meccano cabinet. When a relay contact or telegraph key is connected to Meccano through one of these jacks, closing of the contacts will set the corresponding bit in the Text Register, and opening the contacts will return that bit to zero. The six bits numbered 0 through 5 are completely independent of one another, each following its own input, thereby providing six separate mechanical inputs to the Text Register.

When the Text Input Switch is in the PCM position, the inputs to bits 0 through 5 of the Text Register come from the analog-to-digital converter (Datrac) in Meccano. This converter has only one input; however, its output consists of ten data bits plus a sign bit. Since the Text Register has but six inputs, while Datrac has eleven outputs, a patch cord arrangement has been provided to give the programmer a choice of those bits which he feels are most significant for his purpose.

The patch cords and jacks are located inside the back of Meccano just above the rear of the PCM unit. The inputs to bits 0 through 5 of the Text Register are labeled D00, D01, ... , D05 . The patch cords are labeled +, 9, 8, ... , 1, 0, and correspond to the neon indicators on the front panel of Datrac. Normally the six most significant bits of the PCM unit (+, 9, 8, 7, 6, 5) will be plugged into D05, D04, ... , D00, respectively. The remaining patch cords should be plugged into the

extra jacks provided. Since there is a small voltage (20 v) on these patch cords, it is important that none of them are left dangling, thus allowing them to short out on the chassis frame.

The input to the PCM unit may be of several voltage ranges. In order to handle most cases, four different full-scale input ranges are provided: ± 1 volt, ± 10 volts, ± 100 volts. To select the proper input in the last three ranges a switch is provided on the second chassis from the left in the back of the PCM unit. The positions are labeled 1, 10, and 100 corresponding to the ± 1, ± 10, ±100 volts full-scale input. The ± 100 millivolt input requires a special chassis in the PCM unit. The first two chassis (one is a dummy) on the left are removed and the special input chassis is installed. The input jack on this range is in the extra chassis itself. For the normal input ranges either the input jack on the front panel of Datrac or the input jack on the rear (next to the patch cords) may be used.

For more detailed information on the operation of the PCM unit, refer to the Datrac Instruction Manual furnished with the device.

4. Change Register.

As mentioned earlier, the Change Register has a capacity of one bit. When the Sense Change switch, S02, is in the TEXT psoition, a change in state of any of the seven bits of the Text Register sets the Change bit. When S02 is in the CLOCK position, however, only a change in state of the overflow bit of the Text Register will set the Change bit. The change bit is cleared when, and only when, the clock overflow bit of the Text Register is cleared; that is, only after a succeeding Text Register-to-Bogart instruction has been executed.

5. Programming.

The following external function codes have been set up to allow Meccano to operate under program control:

15000 - Select Text Register as input.

15001 - Select Clock Register as input.

15002 - Select Alternate Mode as input.
In this mode of operation the first input instruction will read the Text Register, the second input instruction will read the Clock Register, and so forth alternately.

45000 - Select Sense Meccano.
This external function selects the Change Register for Control. This instruction is usually followed by the Sense-Jump instruction (SNJ).

Repeated input instructions may be used.

The programmer may wish to indicate those options of Meccano not under program control which he employs in a specific program. Thus, he may indicate the proper positions of the Input, Sense-Change, Advance Clock, and Scale-Range-Switches, as well as whether or not the overflow input bit to the Text Register is to be disabled. He may also wish to indicate which positions of the Datrac Output Register are to be connected to the Text Register.

6. Operation.

The following steps should be observed when turning on the equipment:

(1) See that all connecting cables are properly seated in their respective jacks. Also make certain that the output leads from the Datrac unit are plugged into the corresponding positions in the jack strip above the rear of that unit.

(2) Be sure that the Power toggle switch on the front of the Datrac unit is in the ON position and the Trigger Mode switch is on the EXTERNAL position, if the device is to be used.

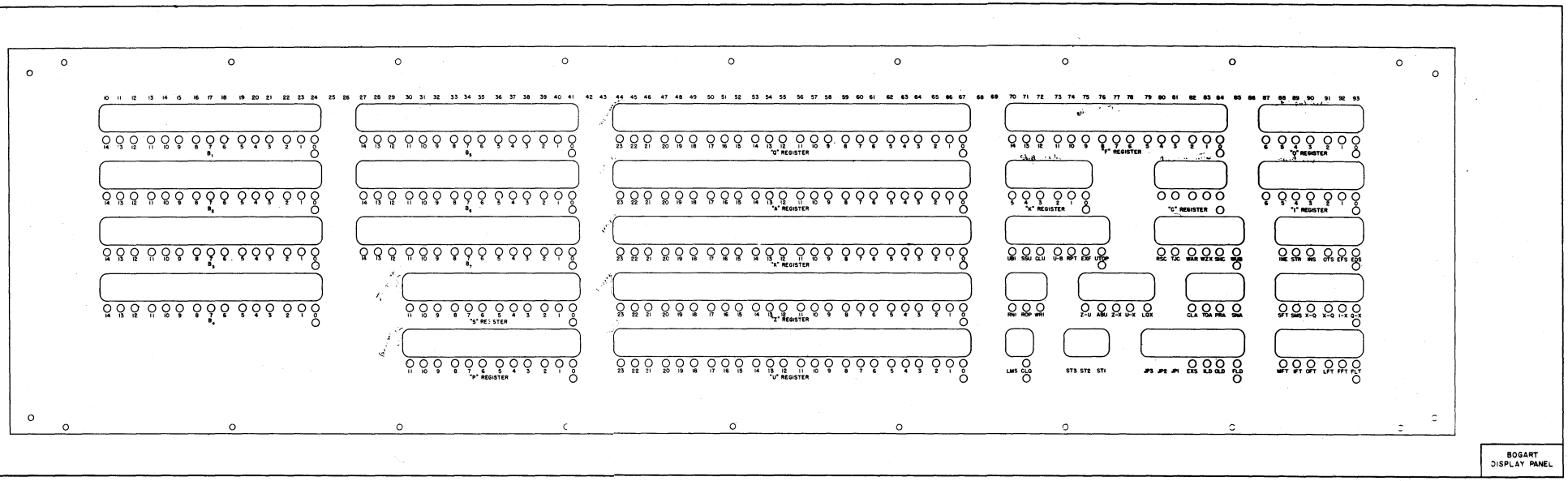
(3) See that the individual circuit-breakers (CB01, CB02, CB04) controlling 400-cycle power to their respective chassis are turned on. The Clock circuit-breaker (CB04) is found on the Clock Chassis subassembly suspended from Standard Chassis number 2. The other breakers are located on the indicator-control panel.

(4) Turn on the 110-volt, 60-cycle switch (CB03) located on the center bulkhead inside the rear cabinet door. This distributes power to the Datrac unit and starting relays, and turns on the blower motor.

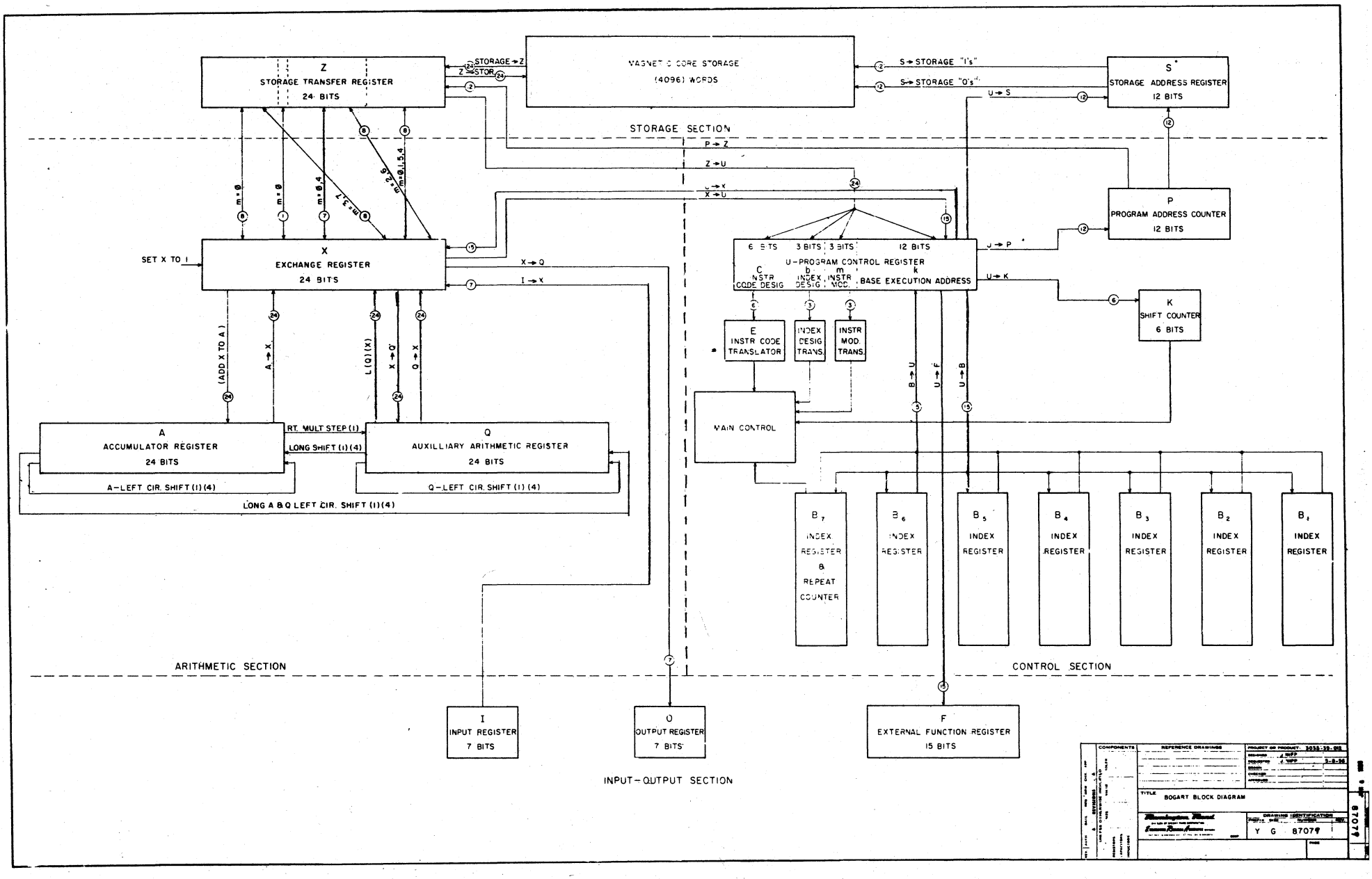
(5) Press the Power-On Button on the indicator-control panel. This energizes the 60-cycle power relay (K01), which is held in by its own contacts. When K01 pulls in, K02 and the +50 and -100 volt power supplies are energized. With the closing of the K02 contacts, the Main Power indicator (I01) and the AC On light (I02) will glow.

NOTE: Failure of the AC On light will indicate a burned-out lamp or a blown fuse in the -14 volt supply. These fuses are found on the inside center panel above the 60-cycle power switch (CB03).

The unit should be allowed to warm up for five minutes before being used.



BOGART
DISPLAY PANEL



COMPONENTS	REFERENCE DRAWINGS
PROJECT OR PRODUCT	3035-39-019
TITLE	BOGART BLOCK DIAGRAM
DRAWING IDENTIFICATION	Y G 8707 9