Scientific Control Corporation

Interface Description
650

650 INTERFACE DESCRIPTION

© SCIENTIFIC CONTROL CORPORATION
14008 Distribution Way
Dallas, Texas 75234
CHapel 1-2111, AC 214

SCG 20-1266

TABLE OF CONTENTS

Table of Contents, continued

PREFACE

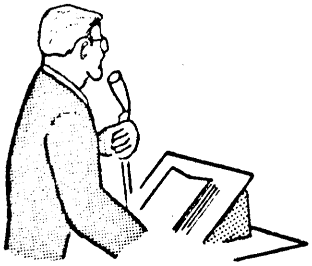
Over the past decade, the world has experienced an overwhelming trend to automate every phase of modern life.

With this trend came the realization that the greatest limiting factor in automation is the control and handling of large volumes of information from many sources. At first, this trend could be satisfied by small special purpose electronic controllers. As the volume of information to be handled increased, and the demands for greater flexibility of the control unit was expressed, the control task was passed to the digital computer.

The general purpose digital computer provides a versatile way to process large volumes of information efficiently. It is, however, a relatively useless machine unless there is a simple and understandable method for it to communicate with the operator and the devices it is to control.

With the large and ever-changing number and variety of input/output devices, i.e., card, tape, keyboard printer, visual display, analog converters, etc., the computer interface must be capable of keeping up with new and changing requirements of a modern processing system. To do this, it must require little or no change in the programming or hardware of the central processing unit. This versatility of machine requires an "open ended" design which will hold obsolesence to a minimum, yet present simplicity of control.

To this end, Scientific Control Corporation presents the SCC 650 General Purpose Digital Computer.



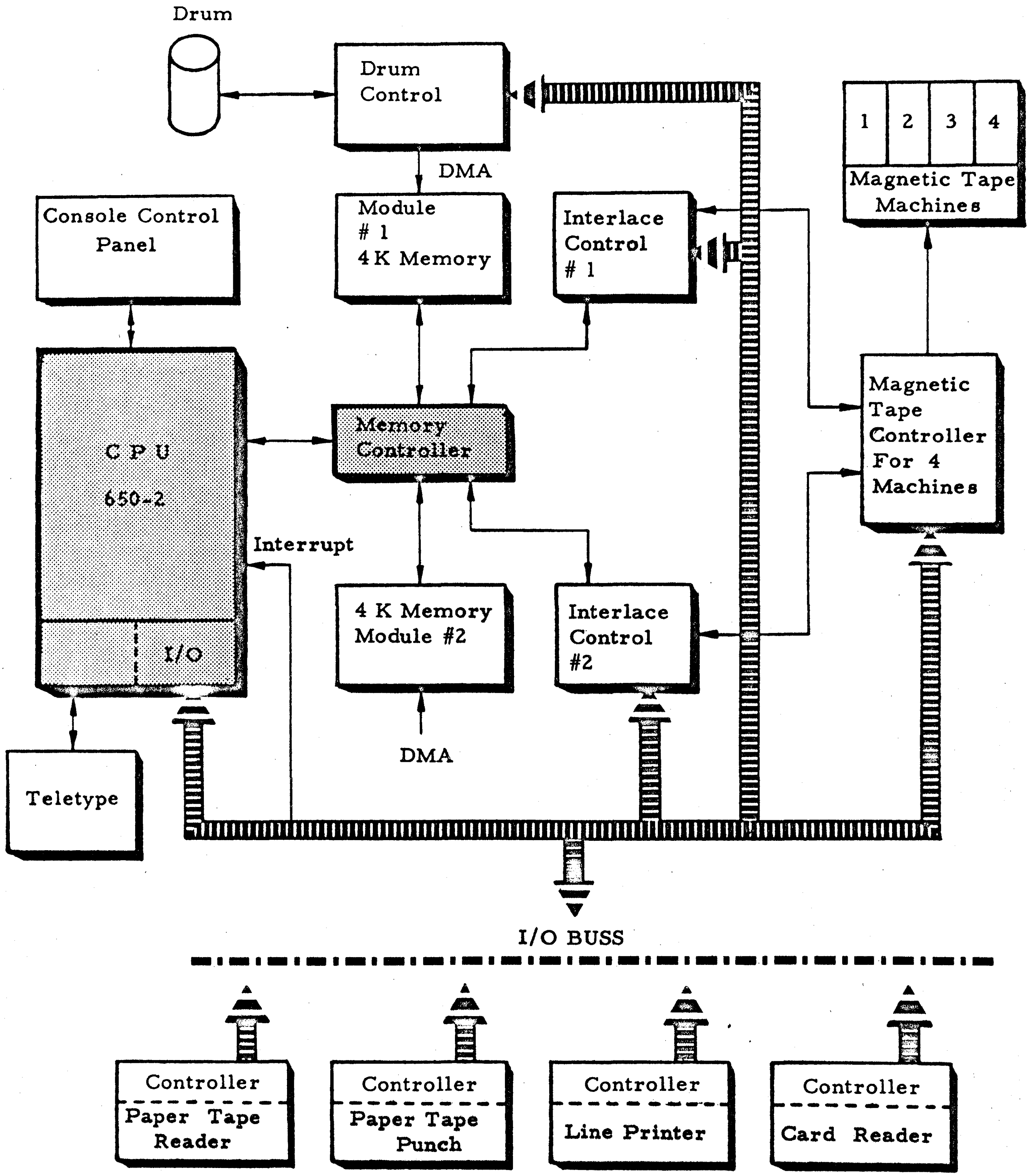
INTRODUCTION

The 650 is a silicon solid state general purpose digital computer with functional software and versatile hardware. It is designedto be used in a variety of applications and configurations. The basic computer includes a 4096 word, 12-bit, 1.75 microsecond cycle time random access core memory with direct external access. This memory is expandable in 4096 word modules to 32,768 words, with each module including direct memory access. Other outstanding features of the SCC 650 are memory protect, hardware index register, micro instructions (one machine cycle combinational instructions), priority interrupt, indirect addressing, and an operational control console.

High speed and efficient processing is achieved by a versatile input/output structure, fully parallel operation, modern addressing techniques, a comprehensive instruction list and flexible subroutine linkages.

The direct memory access feature with the control and storage element permits high speed data transfer between memory and external devices. This access channel operates on a cycle stealing basis; that is, it defers the central processor at the end of the current memory cycle. As a result, the external device is not required to wait until the present instruction is complete before control of the memory is transferred to the external device. Complete isolation between memory banks allows the computer to continue undisturbed if the external device requests a memory bank other than the one being used by the processor. Memory overlap may occur when two or more external devices request different memory modules simultaneously. Therefore, it is possible for each of eight different devices to transfer data simultaneously at a rate of 500,000 twelve-bit words per second into each independent bank for a total transfer rate of 4,000,000 words per second.

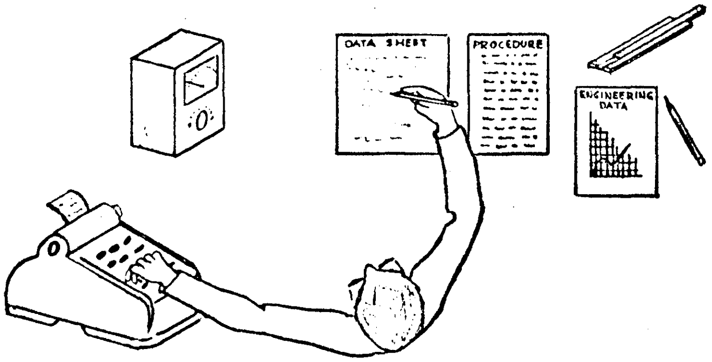
All arithmetic, logical operations, and data transfers within



TYPICAL SCC 650 COMPUTER SYSTEM

the CPU, memory and various registers, are performed in parallel, thus allowing large amounts of data to be transferred during each machine cycle. To add to this versatility, micro-program instructions are included to allow up to four different program operations to occur during one machine cycle time. All instructions which do not reference memory may be micro-programmed. This provides the programmer with a variety of bit-manipulation, shift, recovery, skip-test, logical and I/O instructions. The system programmer may also select four modes of addressing: indirect, direct, relative, or indexed. Each of the four possible addressing modes is based on the concept of a "primary address." The primary address of an instruction is the address which can be formed using the instruction address and the available address modification registers. Since the CPU may perform program operations upon every cell of the storage element, address modification may be extended all cells of the storage element as address modifiers.

Optional equipment offered for the SCC 650 computer includes: DMA channel controllers to allow block transfers between memory and high speed devices, paper tape punch and reader, card punch and reader, teletype consoles, magnetic tape units, incremental XY plotters, line printers and analog to digital, digital to analog converters, digital or analog multiplexers, real time clock, buffer controls and other devices. The SCC 650 may be rack mounted or, as an option, housed in a desk configuration.

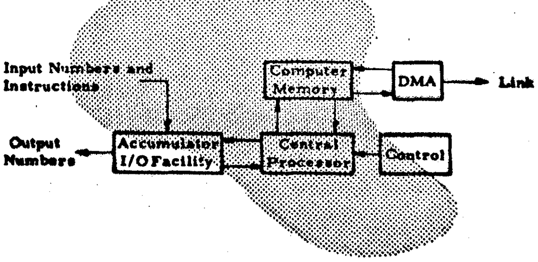


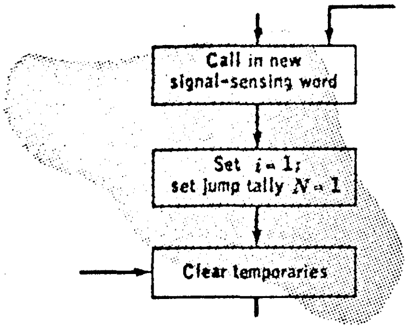
INPUT/OUTPUT FACILITY

There are two major methods to insert or extract information from the SCC 650 central control and storage.

The first and most frequently used is the data transfer under program control, or program transfer. This requires a resident program from the central processor to monitor the devices and (a) to call the device to be used and prepare it for data transfer, (b) determine when the device is ready to transfer data, (c) monitor control functions within the device, and (d) to terminate the device at completion of information transfer.

The second form of data exchange is the direct entry of information into the core storage element or direct memory access (DMA) transfer. This transfer enters or extracts information directly from the core storage on a high speed, direct-to-memory basis. This process proceeds on a priority cycle stealing mode if the DMA and CPU require the same memory bank at the same time. The DMA control will defer the CPU for one memory cycle if necessary. An external hardware control is required to enter data in this process to (a) provide the cell location of data transfer within the memory, (b) to initiate the data transfer, and (c) to determine data entry or exit conditions. This transfer process is used for most high speed and "block" data transfers. This form of transfer, along with the interrupt system, is sometimes used to command link two or more computers together for computer interchange operation.



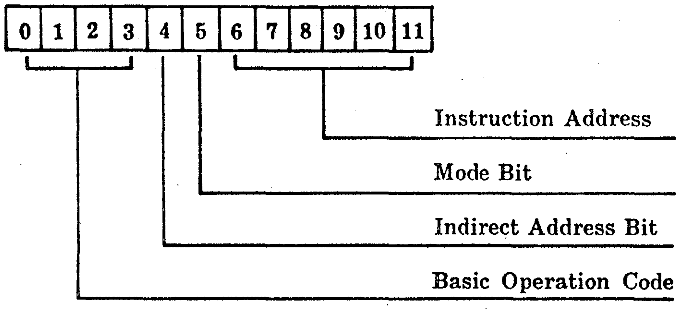

Call in new signal-sensing word
Set $i = 1$; set jump tally $N = 1$
Clear temporaries

PROGRAMMED DATA TRANSFER

This facility allows the computer program to control the input/output sequence by initiating proper I/O instructions and testing to see if the external device is in a ready state. If the external device is ready, then the computer will initiate a data transfer and execute the next instruction. Program data transfers can be initiated by the programmer or by the external device using a program interrupt. Upon receipt of an interrupt from the external device, the computer will trap to the particular subroutine that is programmed to service the interrupting device.

BASIC INSTRUCTION FORMAT

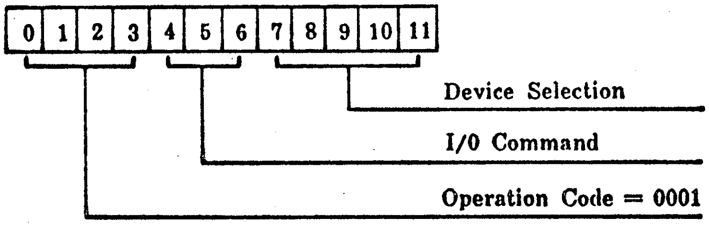
The basic instruction format is as follows:



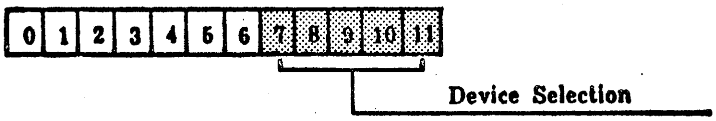
The SCC 650 basic 12-bit instruction format provides for a 4-bit machine operation code two control bits and six address bits.

The basic instruction format is presented to illustrate that this format is only interpretively modified to obtain the I/O format. Since one needs neither indirect addressing, nor mode control, during I/O operations, these bits (#4 and #5), along with bit #6 of the address, are combined to form the coding for the I/O commands. Bits 7, 8, 9, 10 and 11 may be interpreted as an I/O device addresses, or device type selection codes. The user may use this interpretation or may use these bits in combination with the A Register for special or unusual requirements.

Input-Output Instruction Format



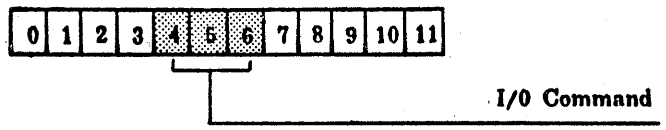
This interpretation of the input / output instruction of the computer allows for commanding up to 32 device types to execute eight functions. Within this basic structure, it is possible through the use of the EXU command to control 4, 096 of each of the 32 basic device types as explained under second order commands.



The codes for the device selection portion of the instruction are arbitrarily assigned a psuedo octal code consisting of two bits (#7 and #8) for the first octal number and three bits (#9, #10 and #11) for the second octal number. One should note that the most significant bit (#7) is arranged to denote an input (OV) or an output device (+8 V).

INPUT CODES		OUTPUT CODES	
00	CLOCK	20	INCREMENTAL PLOTTER
01	TTY KEBD #1	21	TTY PRINT #1
02	TTY READ #1	22	TTY PUNCH #1
03	TTY KEBD #2	23	TTY PRINT #2
04	TTY READ #2	24	TTY PUNCH #2
05	IBM KEBD	25	IBM PRINT
06	HS PAPER TAPE READER	26	HS PAPER TAPE PUNCH
07	CARD READER	27	CARD PUNCH
10	DISK FILE READ	30	DISC FILE WRITE
11	A/D CONVERTER AND BUFFER #1	31	LINE PRINTER
12	A/D BUFFER #2	32	DIGITAL CHANNEL
13	SYSTEM	33	SYSTEM
14	MAG. TAPE READ #1	34	MAG. TAPE WRITE #1
15	MAG. TAPE READ #2	35	MAG. TAPE WRITE #2
16	MAG. TAPE READ #3	36	MAG. TAPE WRITE #3
17	MAG. TAPE READ #4	37	MAG. TAPE WRITE #4

The I/O Command portion of the I/O instruction:



Synopsis of the I/O commands:

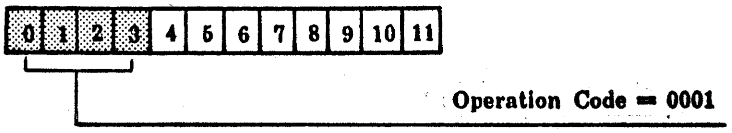
(Low Order Commands)

$R_4R_5R_6$	COMMANDS		MNEMONIC	CYCLES
000	Transmit to A & Skip (Selected device) → A (P) + 1 → P Otherwise P + 2 → P	If device is ready	TTA	1
001	Transmit from A & Skip (A) → Selected device (P) + 1 → P Otherwise P + 2 → P	If device is ready	TFA	1

(Synopsis of the I/O commands, continued) (Low Order Commands)

$R_4R_5R_6$	COMMANDS		MNEMONIC	CYCLES
010	Input Device Status (Device status) → A (P) + 1 → P Otherwise P + 2 → P	If device is ready	DST	1
011	Skip on device flag FLAG = 1 (P) + 1 → P FLAG = 0 (P) + 2 → P		SDF	1
(High Order Commands)				
100	Execute Command in A External device executes command in A		EXU	1
101	Terminate Inactivate device		TMR	1
110	Selected Activate device		SEL	1
111	Activate I/O Command		IOC	1

During the execution of any I/O command, the decode OP code (IOP) is routed via a driver inverter to the output connector to form the I/O operation command (IOPB-).



THE I/O OPERATION CODE

The three signals, R_4, R_5, and R_6, representing each bit of the I/O command, are also available on the I/O connector and are presented at the interface as (RO4B-), (RO5B-), and (RO6B-). These are decoded by the external device to determine the exact I/O command in order to simplify some interface hardware. The data and status control signals are decoded and presented to the interface as (TTAB-), (TFAB-), (DSTB-), and (SDFB-). All eight of the master CPU timing pulses are supplied to the I/O connector as (RTOB-), (RT1B-), (RT2B-), (RT3B-), (RT4B-), (RT5B-), (RT6B-), and (RT7B-). (See Timing Chart). The complete complement of timing is presented to allow the external device to manipulate data within its controller at any convenient time period with relation to the CPU timing requirement. The five bits of the device select code are decoded by the external device to determine the device which is to respond to the I/O command. These are presented as buffered data bits from the R Register (RO7B-), (RO8B-), (RO9B-), (R10B-), and (R11B-).

In order to carry out the SCC "open end" design philosophy, attention should be directed to the high order device commands EXU (100), TMR (101), SEL (110), and IOC (111). The EXU (execute command in A) requests the external device to look to the A Register for a command word. Since the CPU generates no function except the I/O op code decode, and requires no signal from the device, all high order commands may be interpreted one of two ways.

These high order commands may be used simply as four individual one instruction commands to the external device or may, according to system requirements, direct the device group addressed to look to the accumulator for further, or second order, instructions.

Example of a first order I/O instruction:

$(0607)_8$

Arm interrupt control for card reader. This allows the card reader to interrupt on reading a card segment and being ready for information transfer. This does not require a command in the A Register, so no prior load or modification of the A Register **is necessary in first order instructions.**

Example of second order I/O instruction:

$(0772)_8$ SEL, Digital Channel	Activate digital channel #$(4003)_8$ to accept next T F A, Digital Channel $(0072)_8$ data, upon accepting that data hold the value and return to "stand by off line" until commanded again. It should be noted that this instruction (SEL) normally does not look to the A Register for part of its control code, but as in this case, can be directed by hardware to do so.

I/O COMMANDS AND TIMING

The symbol "XX" means any or all of specific bits of a register, i.e., R Register bit 3 would be R03; any bit of the R Register would be R"XX".

Transmit to A

Time And Condition	Function	Performed By
T2·IOP	Set Device Ready Line (IDRDY) if ready and gate data to input Buss (IN"XX")	External Device
T4·IDRDY	Set internal ready condition DRDY	Computer
T5·DRDY	Clear Accumulator	Computer
T6·DRDY	Transfer input to Accumulator and set Skip F/F if device ready	Computer
T7·IOP	Clear (IDRDY) - Buss clear data buss	External Device

Transmit From A

T2·IOP	Data available for transfer of accumulator data to external device. Set (IDRDY) Buss low if device ready	External Device

I/O Commands & Timing

(Transmit From A, continued)

Time And Condition	Function	Performed By
T4· IDRDY	Set internal ready condition (DRDY)	Computer
T5· DRDY	Set Skip F/F if device ready	Computer
T6· IOP	Clear IDRDY - Buss	External Device

Input Device Status To A

T2·	Gate Status data to Input Lines (IN"XX")and set IDRDY - Buss low	External Device
T4· IDRDY-	Set internal ready condition (DRDY)	Computer
T5· DRDY	Clear Accumulator	Computer
T6· DRDY	Transfer input lines into Accumulator	Computer
T7· IOP	IDRDY - Buss clear	External Device

Skip On Device Flag

T2· IOP	Set Device Flag (DF-) low (0 volts)	External Device
T5· DF	Set Skip F/F if Device Flag (DF-) is low	Computer
T7· IOP	Clear Device Flag Buss (DF-)	External Device

Execute Command In A

T2· IOP	Clear Device Command Register	External Device
T6· IOP	Transfer Accumulator into Device Command Register	External Device
T7	Remove I/O Command	Computer

I/O Commands & Timing

Time And Condition	Function	Performed By
Terminate		
T2·IOP	Clear Device Activate F/F	External Device
T7	Remove I/O Command	Computer
Select		
T2·IOP	Set Device Activate F/F	External Device
T7	Remove I/O Command	Computer
I/O Control		
T2·IOP	Set I/O Control Activate F/F	External Device
T7	Remove I/O Command	Computer

Two signals are required by the computer from the external device. They are:

1. Device Flag. This signal is tested by the computer skip-on-device flag command. (SDF or I/O Command 011). Skip (P+2 → P) if flag (DF-) is +8 V; otherwise (DF- = O V) P +1 → P.

2. Device Ready (DRDY-). This signal conditions the computer transfer on:

   a) Transmit to Accumulator (TTA or I/O Command 000)
   b) Transmit from Accumulator (TFA or I/O Command 001)
   c) Input device status to Accumulator (DST or I/O Command 010).

These transfers are not completed and the skip will occur unless device-ready signal is present. That is, DRDY- = O V, P+1 → P; otherwise, DRDY- = +8 V, P+2 → P.

Both device flag and device ready must be set true at the CPU prior to (RT4B-) and cleared no earlier than (RT6B-). The (RT2B-) and (RT6B-) may be used to synchronize the turning on and off of device flag and device ready flip-flops.

A signal is provided DEVICE READY F/F BUFFERED (DRDYB-) to notify the external device controller that the Device Ready Flip-Flop within the computer is set, the CPU will not skip the next instruction and data transfer will occur. The TTA, TFA, SDF, DST instructions are organized so that they will skip the next instruction if the device is NOT ready (skip if not ready). This allows a test for the source of interrupts by TTA and jump instructions only. An example interrupt service routine would be as follows:

STA	Alpha	Save Accumulate
TTA	DV_1	Transmit to A Device #1
JMP	SER DV_1	Jump to Service Device #1
TTA	DV_2	Transmit to A Device #2
JMP	SER DV_2	Jump to Service Device #2
	ETC.	

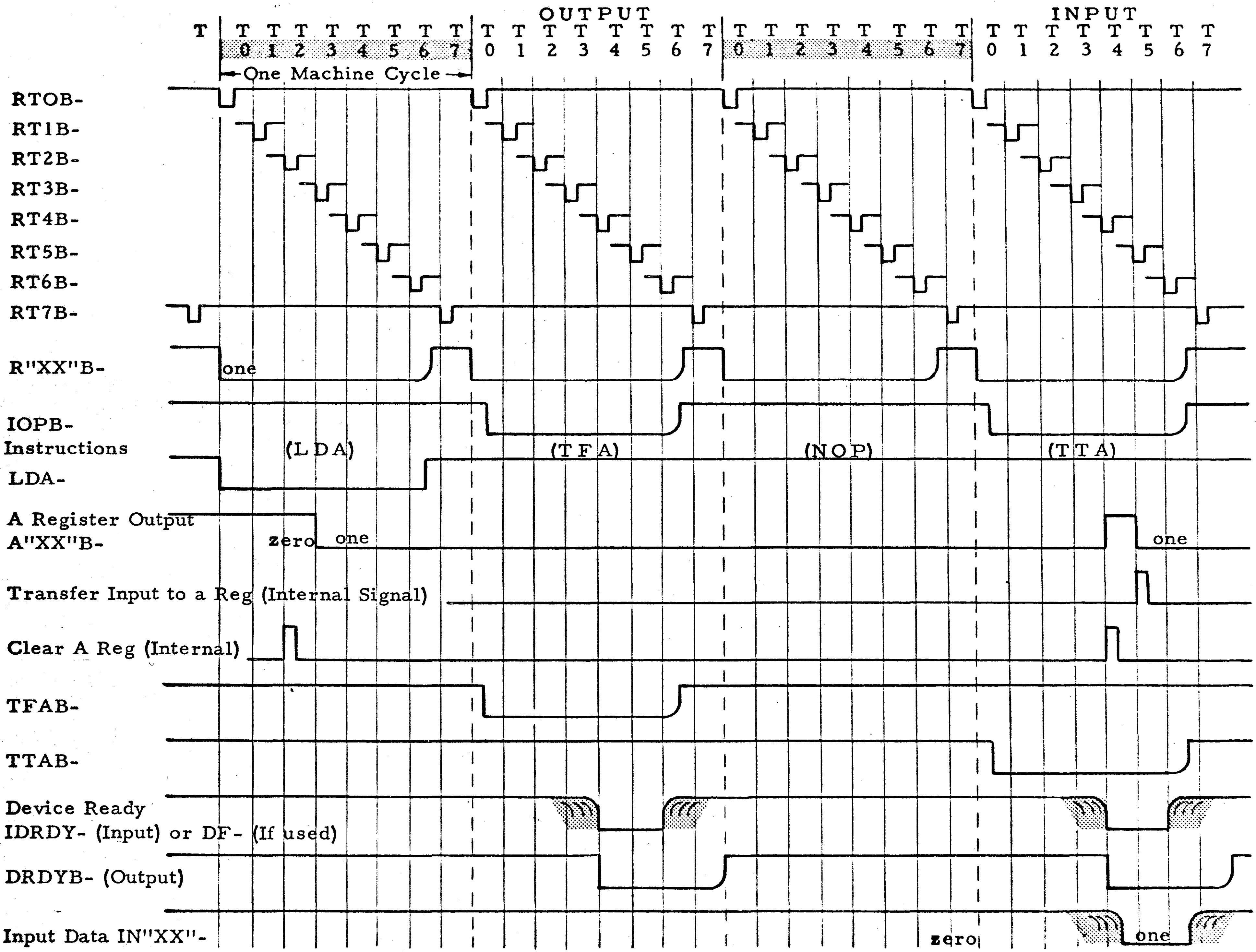

OUTPUT
INPUT
T
T 0 T 1 T 2 T 3 T 4 T 5 T 6 T 7
One Machine Cycle
RTOB-
RT1B-
RT2B-
RT3B-
RT4B-
RT5B-
RT6B-
RT7B-
R"XX"B-
one
IOPB-
Instructions
(LDA)
(TFA)
(NOP)
(TTA)
LDA-
A Register Output
A"XX"B-
zero
one
one
Transfer Input to a Reg (Internal Signal)
Clear A Reg (Internal)
TFAB-
TTAB-
Device Ready
IDRDY- (Input) or DF- (If used)
DRDYB- (Output)
Input Data IN"XX"-
zero
one

FIGURE NO. 3

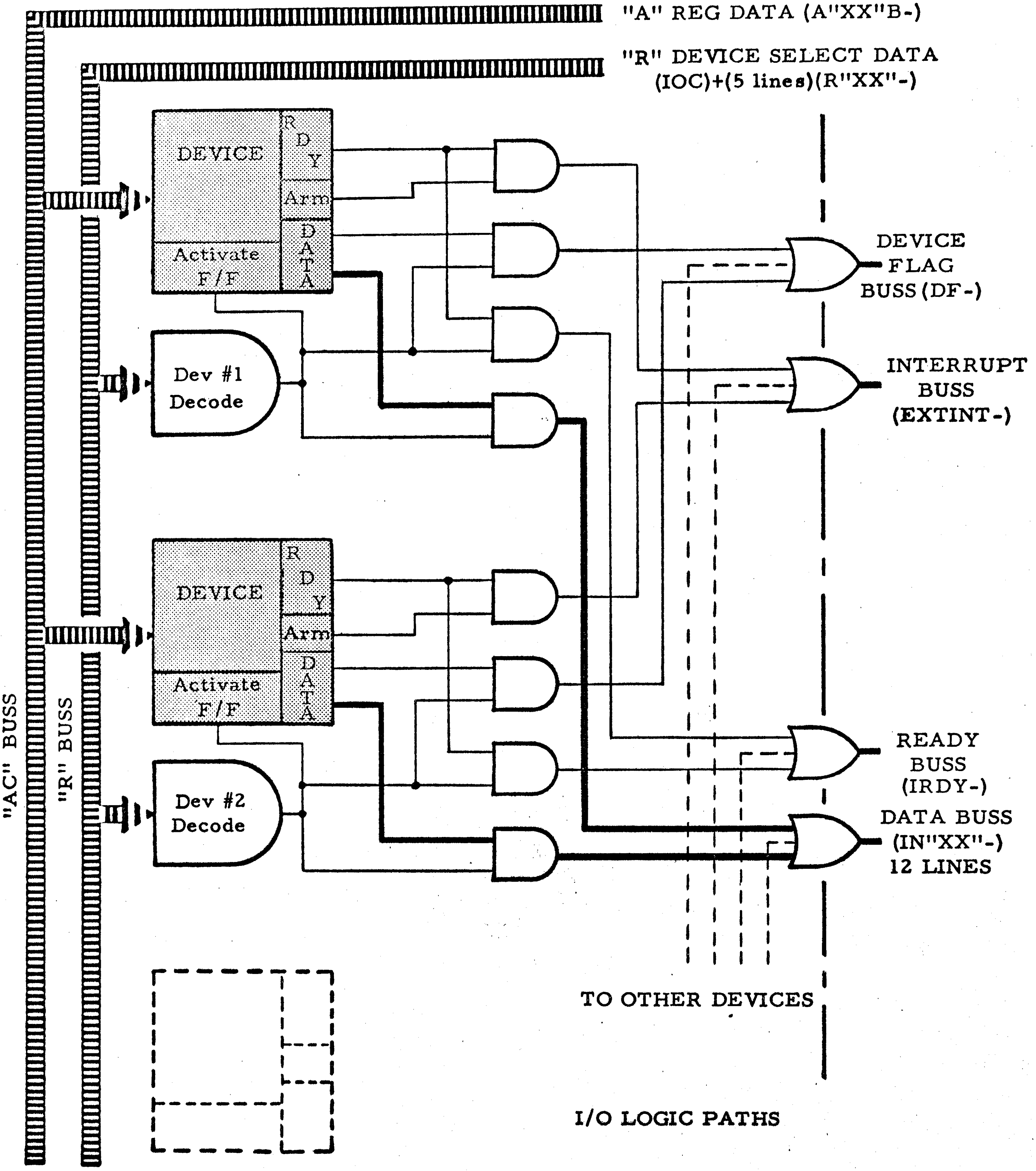

"A" REG DATA (A"XX"B-)
"R" DEVICE SELECT DATA (IOC)+(5 lines)(R"XX"-)
DEVICE
RDY
Arm
DATA
Activate F/F
Dev #1 Decode
DEVICE
RDY
Arm
DATA
Activate F/F
Dev #2 Decode
"AC" BUSS
"R" BUSS
DEVICE FLAG BUSS (DF-)
INTERRUPT BUSS (EXTINT-)
READY BUSS (IRDY-)
DATA BUSS (IN"XX"-) 12 LINES
TO OTHER DEVICES
I/O LOGIC PATHS

FIGURE NO. 4

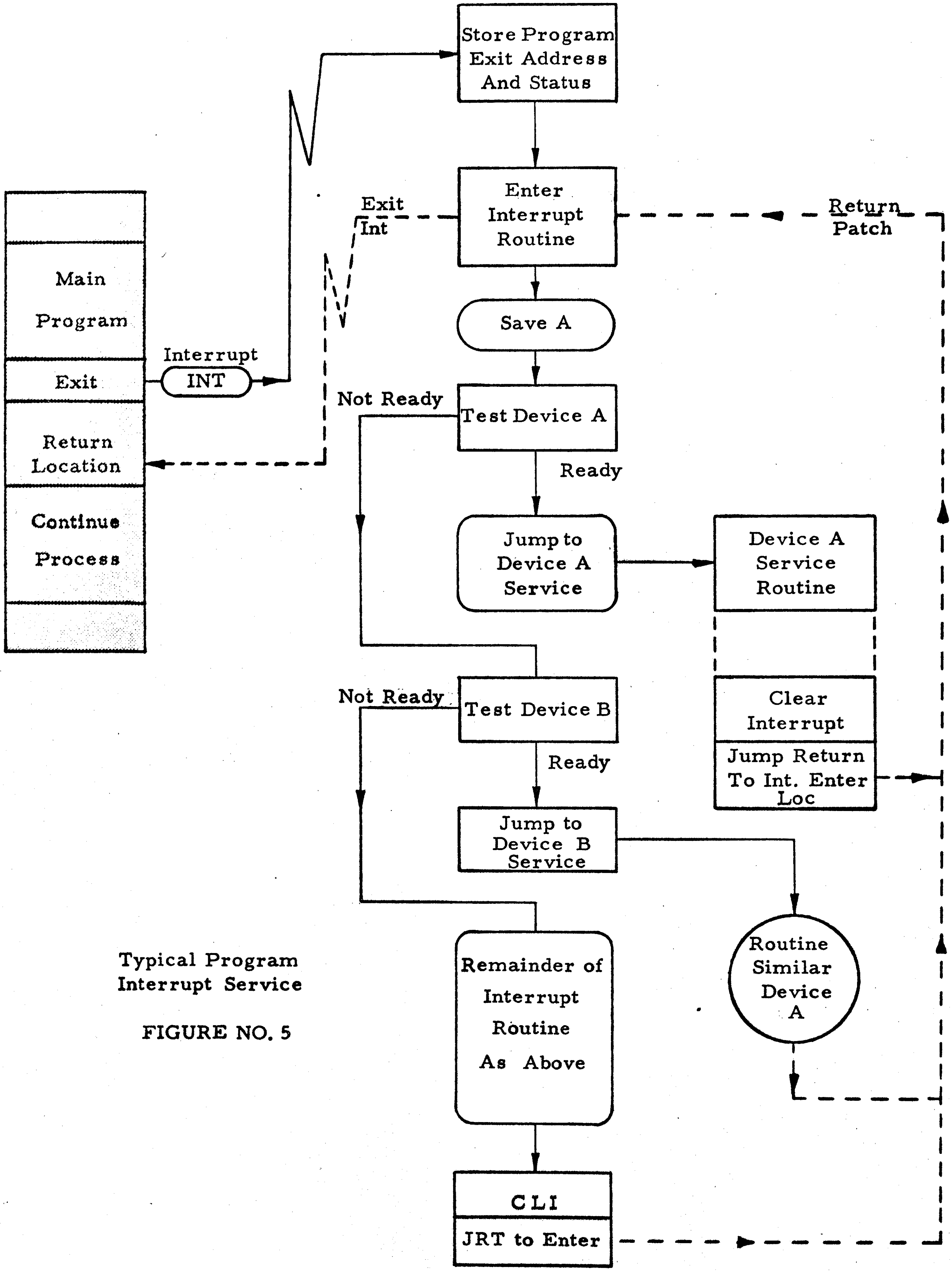
PRIORITY INTERRUPT

An input is provided (EXTINT-) for interrupt of the machine by the external devices. This input, when taken to 0 volts, will cause an indirect jump and store location to the interrupt trap location and from there to the interrupt service subroutine. This indirect trap allows an interrupt to variable locations during a program. A return to the main program from an interrupt is accomplished by a subroutine return jump JRT to the address located in the interrupt trap location. All exits from interrupt routines should be preceeded by a clear interrupt instruction CLI since the interrupt control will not allow another interrupt of the same level until the current interrupt is processed. If there is an interrupt awaiting the completion of an earlier interrupt to be processed, the interrupt control will exit from the current interrupt subroutine, and return to the main program for one instruction before servicing the waiting interrupt. This feature allows the main program to proceed at a reduced rate, even if it has continuous interrupts.

If the interrupt input line (EXTINT-) is activated (EXTINT=0 Volts) before T2 (RT3B-), the interrupt will be processed at the next T4 pulse (RT4B-) of the next instruction fetch phase if the interrupt subsystem is enabled.

The interrupt subsystem may be disabled or enabled under program control by SIN $(0322)_8$, set on; and SIF $(0222)_8$, set off. The interrupt control may also be overridden to an "on" condition by the interrupt enable button on the console. When the interrupt is enabled by the console, the program may not disable the interrupt subsystem.

The external control devices capable of interrupting the system may be individually armed or disarmed from the interrupt buss by program control and I/O instructions (i. e., arm printer interrupt, disarm tape #3 interrupt). This feature allows the programmer to control the devices on the interrupt buss at his discretion by programmed device command.



Typical Program
Interrupt Service

FIGURE NO. 5

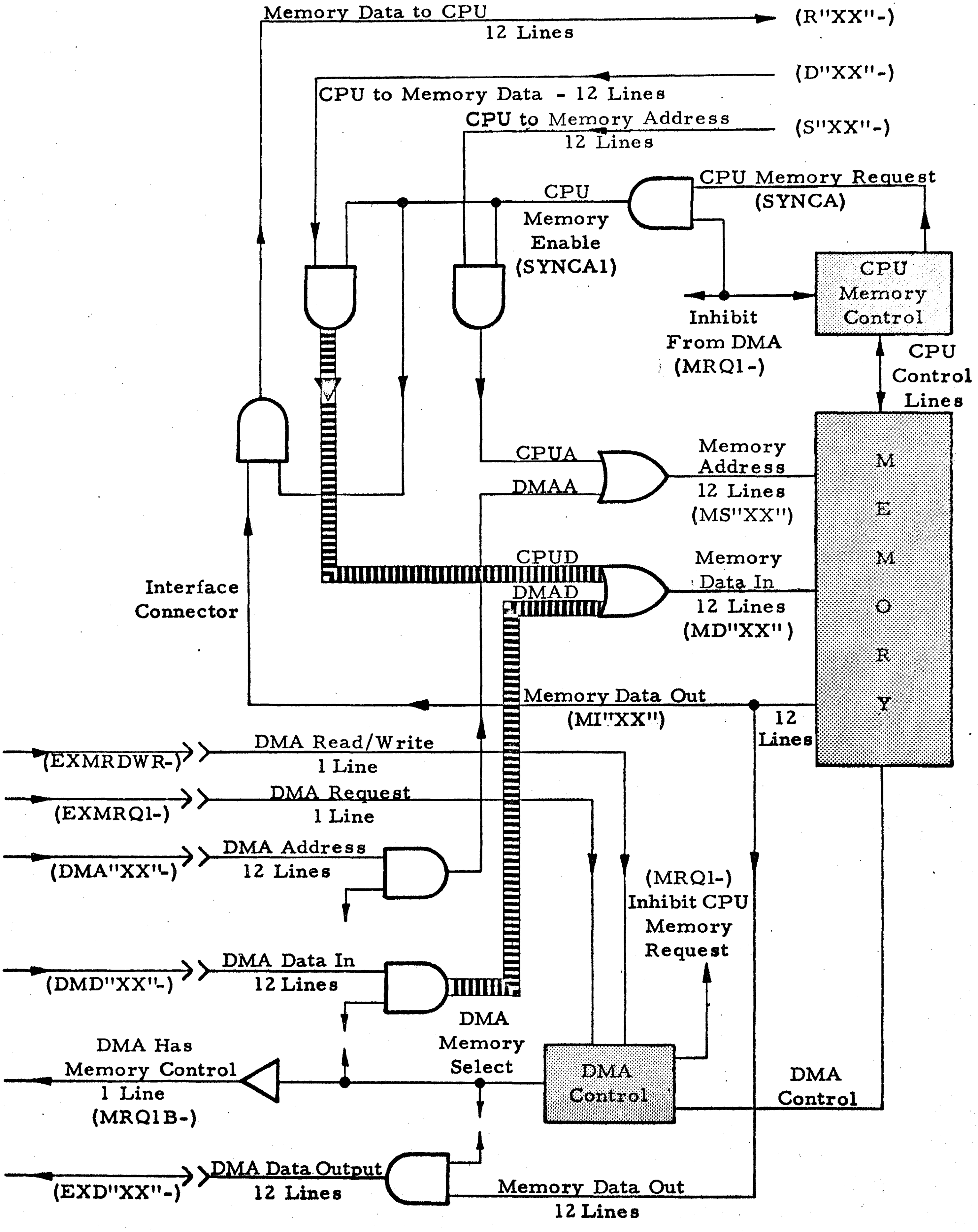
DIRECT MEMORY ACCESS

Direct Memory Access Feature. The direct memory access feature permits high speed data transfer between memory and external devices. This access channel operates on a "cycle stealing" basis. As a result, the external device is not required to wait until the present instruction is complete before control of the memory is transferred to the external device. The DMA does not necessarily defer the program in the CPU if the CPU is not requesting memory for that cycle. If the CPU is executing non-memory instructions (I/O, MICRO OPR, etc.), the CPU will not idle and wait for DMA release control. Complete isolation between memory banks allows the computer to continue undisturbed if the external device requests a memory bank other than the one being used by the processor. The CPU is deferred only if both it and the DMA require the same memory bank, in which case, the DMA has a priority over the CPU.

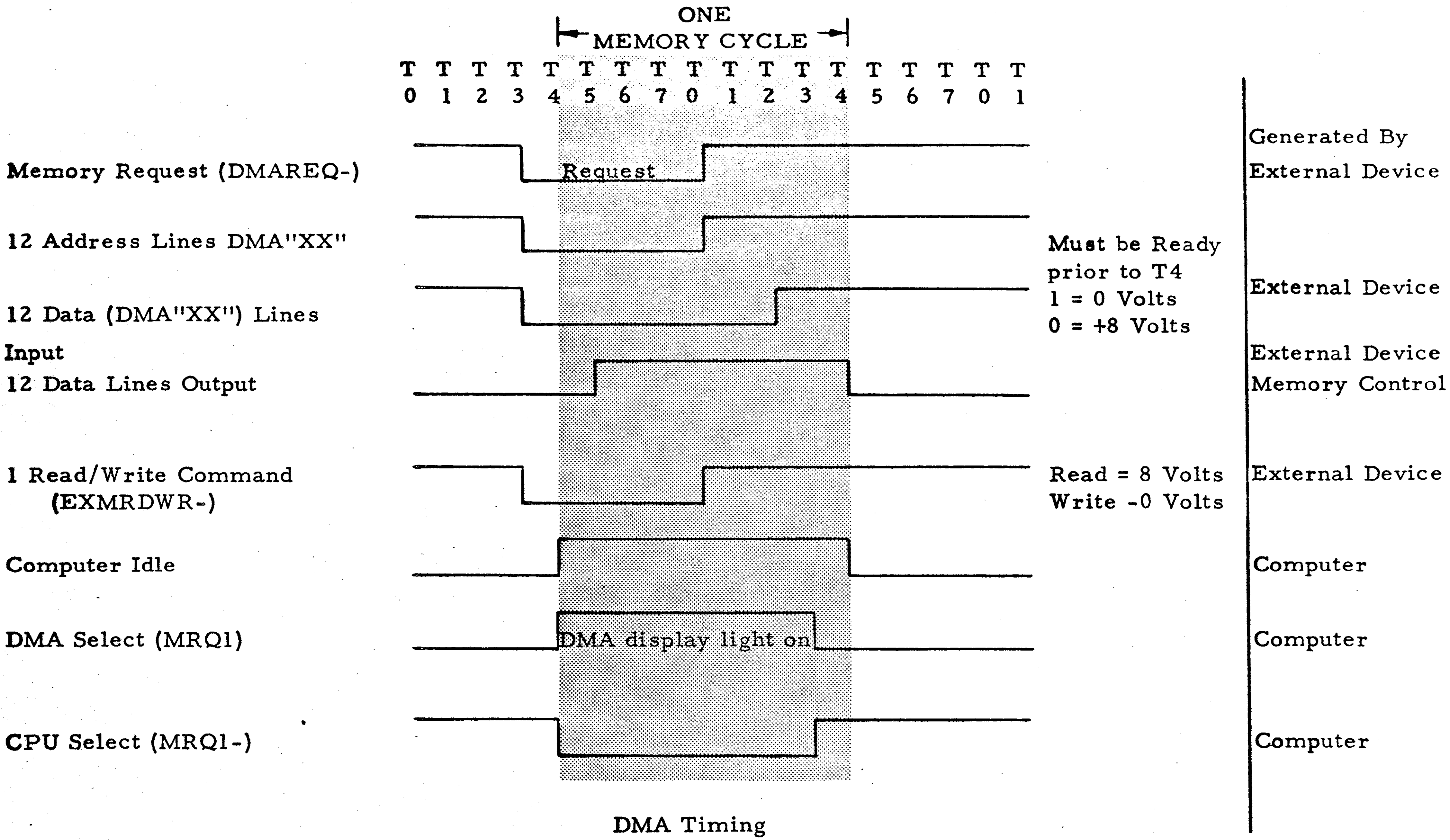
Direct Memory Access Transfers. External devices connected to the computer's direct memory access channel have priority over internal computer operations. Synchronization requires a maximum of one computer cycle. A direct memory access is initiated by the external device giving a memory request--setting the memory request line (DMAREQ-) to 0 Volts--to the computer. Each block or bank of 4,096 words of memory is requested separately through the DMA entry of that memory bank from which the data is to be entered or extracted. The external device must supply to the DMA channel 12 bits of address (DMA"XX"-), 12 bits of data (DMD"XX"-), a read/write line (EXMRDWR-) for +8V for read, 0 volts for write, and the memory request (DMAREQ-) +8V no request, 0 volts for access request. (See diagram of DMA entry).

All signals into the DMA connector should be settled by T4 of the computer cycle and will always be completely processed at the first T4 pulse following the DMA request. Data, address, and read/write signals should remain until the following TO.

A signal is returned to the interface from the DMA control logic to acknowledge that the DMA is active (MRQ1B-) and will remain at 0 volts as long as the DMA has control of the memory.



DMA SELECT GATING - FIGURE NO. 6



DMA Timing
Reference Only

FIGURE NO. 7

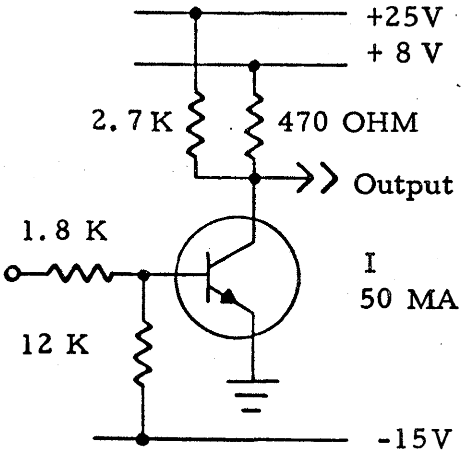
X	Y	S	C
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

LOGIC CIRCUITS

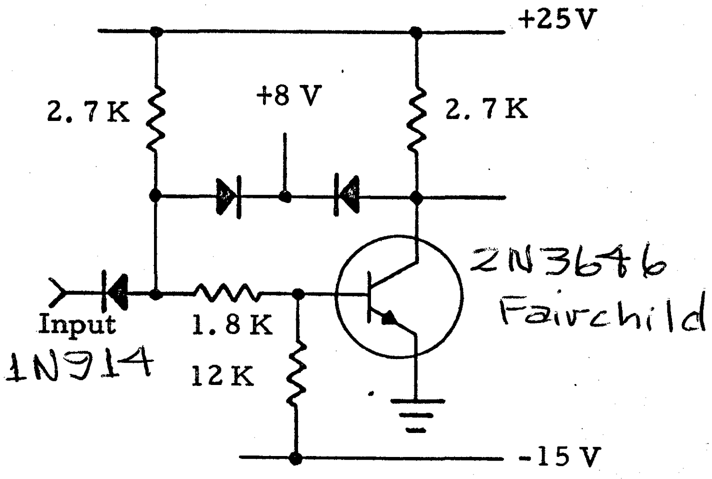
All signals from the computer to be used by external devices are isolated from the internal computer logic by the use of driver inverters. These drivers are included in all 650 - 2 standard logic. The internal CPU logic voltages are +8 volts DC true and 0 volts DC false for standard NAND gate logic. Signals from the external device to the computer at the interface are used by the computer as low true logic or "1". = 0V, "0" = +8V. The input (Type N) and output (Type DI) logic circuits are noted on the interface connector drawings, Figures 15 and 16, Appendix A.

Typical circuits for the input and output circuits are shown below.

Standard
Output Driver Inverter

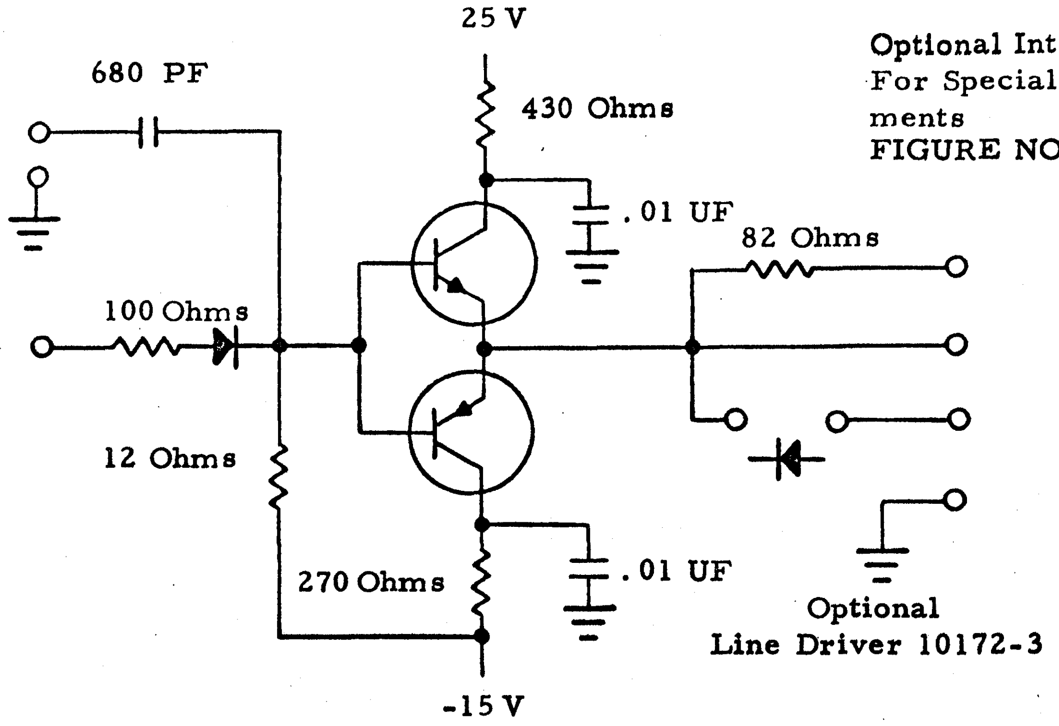


Standard
Input Logic

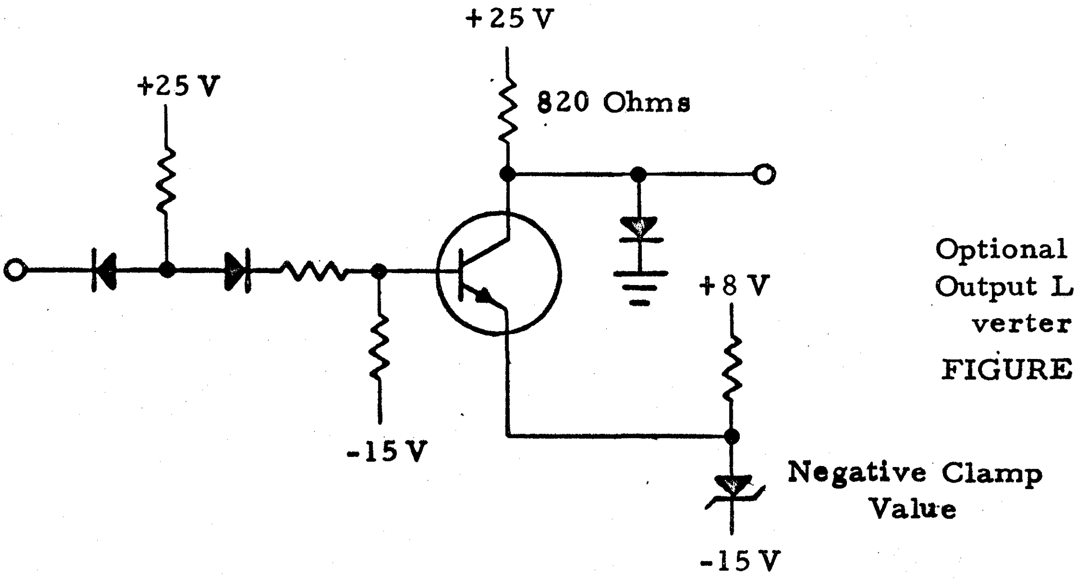


Inputs and outputs from the computer are low true signals. This signal logic was chosen for two reasons: The input circuits will rise to a +8 V level when disconnected from a driving source; therefore, low true logic will force a "zero" condition on all unused input lines. The second reason is to allow customers who wish to do the interface logic for their special applications to use descrete or integrated circuit logic which has an NPN transistor element on the output. By allowing low true, NPN output, logic can, through a saturated transistor, pull down input lines quickly and "sink" the line capacity.

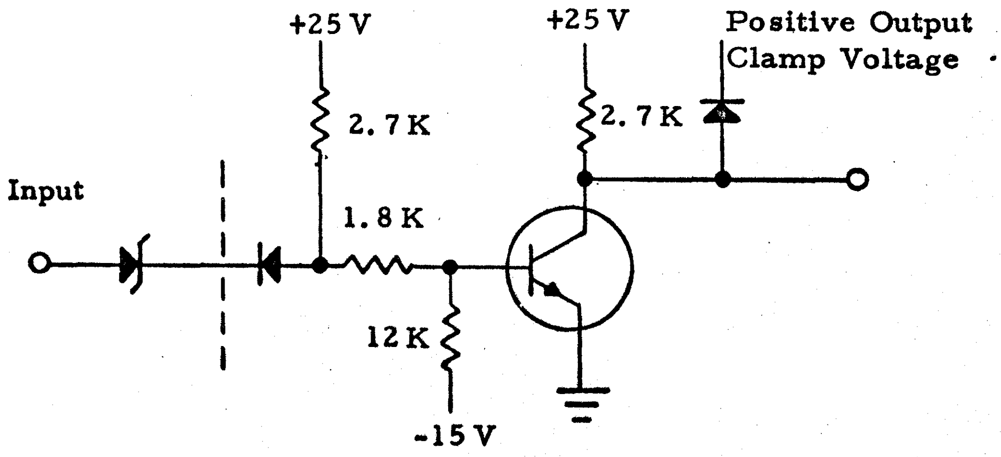
For special applications, the optional line driver 10172-3 is available to drive long lines or to match the output circuits to interface lines. See Figure 10. Also available for level conversion at the interface are the optional negative and positive level converters 10845 and LC10305. See Figures 11 and 12.



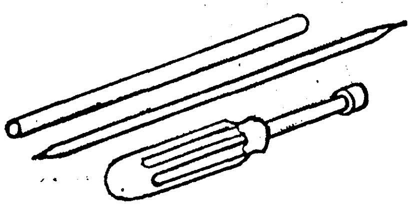
Optional Interface Cards
For Special I/O Requirements
FIGURE NO. 10



Optional Negative
Output Level Converter 10845
FIGURE NO. 11



Optional Output/Input
Converter LC 10305
(Inverter Card With
Zener Input)
FIGURE NO. 12



MECHANICAL CHARACTERISTICS

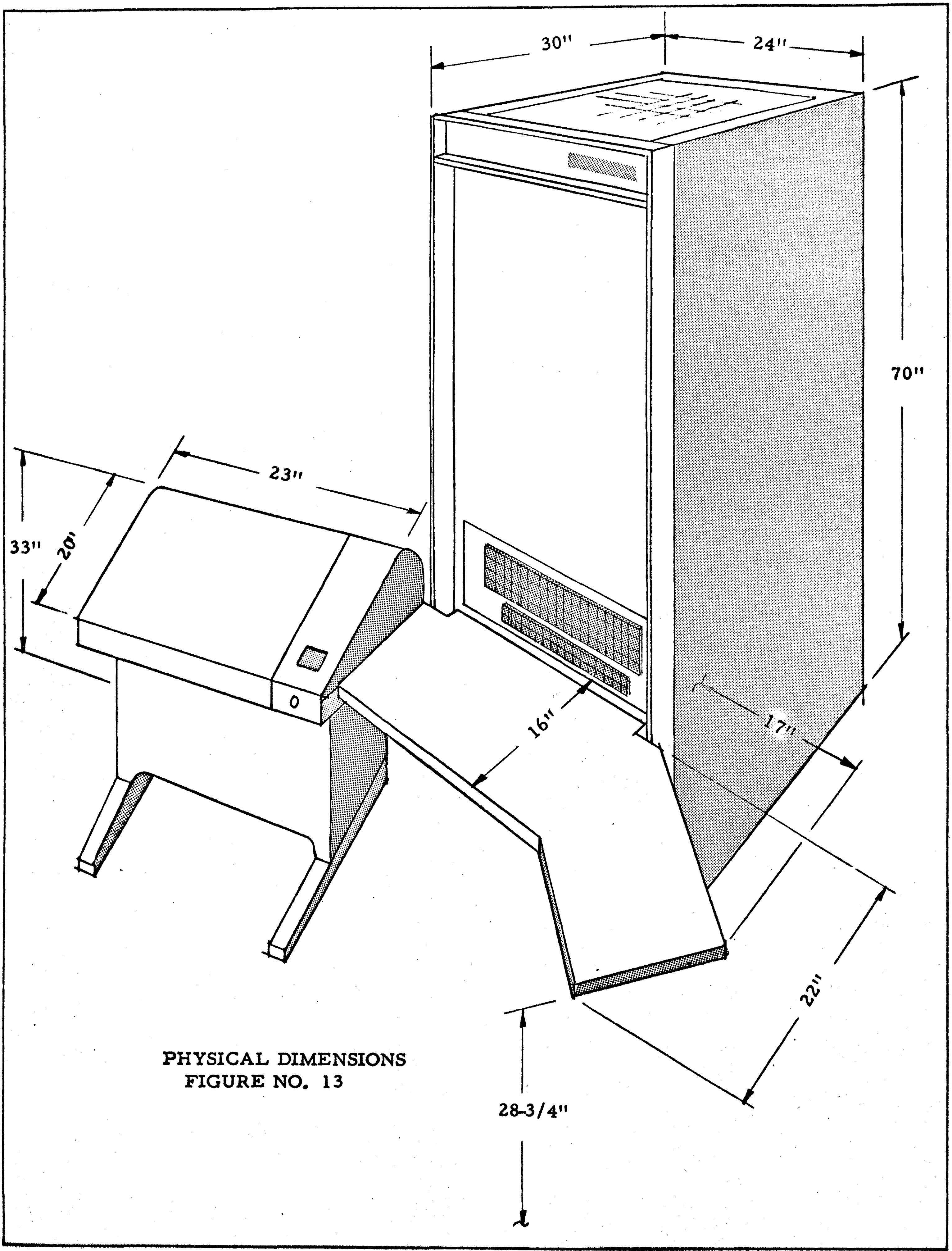
The basic SCC 650 is housed in a standard cabinet with a shadow box trim front and removable sides. The front opening space is 70 inches of panel space. For dimensions, see Figure 13. Other special optional configurations available for the SCC 650 are caster base for mobility, ruggedized slide mount for van applications, and desk configuration where modern formal dress is important. Other special configurations will be considered.

ELECTRICAL CHARACTERISTICS

The SCC 650 requires 115V AC, 60 cycle, 20 AMP, primary, single phase power. Any additional power required from the controlled AC power Buss within the CPU should be added for system requirements.

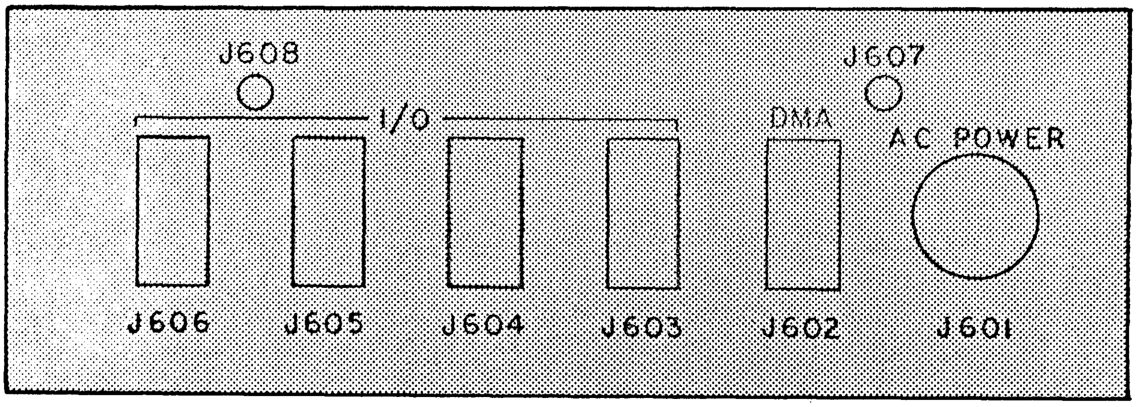
Internally, the CPU power supply furnishes +25V, +10V, +8V, and -15V. The low +10V is used for lamp and display power and the other supplies are used for the logic cards.

The memory supply produces +12V, +3.6V, and -12V for use by the memory system.



PHYSICAL DIMENSIONS
FIGURE NO. 13

All interface connections to the SCC 650 are made to the rear panel connector plate. This panel is located at the lower rear of the cabinet. On this panel are located the DMA, I/O, analog, and AC power connectors as shown below.

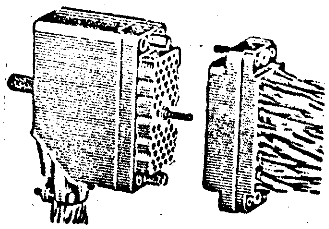


The interface connectors for the I/O and DMA are ELCO 56 pin connectors (ELCO #00-8016-056-000-007). The mating connector is ELCO #00-8016-056-000-003. The optional analog connectors are BNC panel mounts, type UG492 A/U bulkhead adapters and will accept any BNC cable mates.

The AC power is brought in through a Hubbell #7327 twist lock male base. The mating connector is a Hubbell #7313 female connector with #12 wire for line cord.

There are four parallel I/O connectors with pin configuration as shown in Figure 15, Appendix A.

There is one DMA connector connected as shown in Figure 16, Appendix A.



N = Input Logic
DI = Output Logic

INPUT/OUTPUT - APPENDIX A
PIN ASSIGNMENTS
FIGURE NO. 15

UNIT 650-2
CONN J-104
TYPE 8016-056

SIGNAL	FUNCTION	PIN	LOGIC CODE
IN00-	Data Input	A	N
IN01-	"	B	N
IN02-	"	C	N
IN03-	"	D	N
IN04-	"	E	N
IN05-	"	F	N
IN06-	"	H	N
IN07-	"	J	N
IN08-	"	K	N
IN09-	"	L	N
IN10-	"	M	N
IN11-	"	N	N
AC00B-	Data Output	P	DI
AC01B-	"	R	DI
AC02B-	"	S	DI
AC03B-	"	T	DI
AC04B-	"	U	DI
AC05B-	"	V	DI
AC06B-	"	W	DI
AC07B-	"	X	DI
AC08B-	"	Y	DI
AC09B-	"	Z	DI
AC10B-	"	a	DI
AC11B-	"	b	DI
ZEROB082	GND	c	-
ZEROB083	"	d	-
ZERO B081	"	e	
ZERO B081	"	f	
R04B-	Inst Reg Output	h	DI
R05B-	"	j	DI

SIGNAL	FUNCTION	PIN	LOGIC CODE
R06B-	Inst Reg Output	k	DI
R07B-	"	l	DI
R08B-	"	m	DI
R09B-	"	n	DI
R10B-	"	p	DI
R11B-	"	r	DI
IOE-	I/O Error	s	N
		t	
RT1B-	Timing	u	DI
RT3B-	"	v	DI
RT4B-	"	w	DI
RT5B-	"	x	DI
RT7B-	"	y	DI
STCLEAR-	Start	z	DI
EXTINT-	Interrupt	AA	N
SDFB-	I/O OPS	BB	DI
DSTB-	"	CC	DI
TFAB-	"	DD	DI
TTAB-	"	EE	DI
DRDYB- (Output)	Device Ready	FF	DI
RT0B-	Timing	HH	DI
RT2B-	"	JJ	DI
RT6B-	"	KK	DI
IOPB-	I/O Inst.	LL	DI
DF-	Device Flag	MM	N
IDRDY- (Input)	Device Ready	NN	N

DIRECT MEMORY ACCESS - APPENDIX A

PIN ASSIGNMENTS

FIGURE NO. 16

N = Input Logic
DI = Output Logic

UNIT 650-2
CONN J-105
TYPE 8016-056

SIGNAL	FUNCTION	PIN	LOGIC CODE
DMD00-	Data Input	A	N
DMD01-	"	B	N
DMD02-	"	C	N
DMD03-	"	D	N
DMD04-	"	E	N
DMD05-	"	F	N
DMD06-	"	H	N
DMD07-	"	J	N
DMD08-	"	K	N
DMD09-	"	L	N
DMD10-	"	M	N
DMD11-	"	N	N
DMA00-	Address Input	P	N
DMA01-	"	R	N
DMA02-	"	S	N
DMA03-	"	T	N
DMA04-	"	U	N
DMA05-	"	V	N
DMA06-	"	W	N
DMA07-	"	X	N
DMA08-	"	Y	N
DMA09-	"	Z	N
DMA10-	"	a	N
DMA11-	"	b	N
STCLEAR	Start	c	DI
ZERO C28	GND	d	-
ZERO C28	"	e	-
ZERO C28	"	f	-
ZERO C28	"	h	-

SIGNAL	FUNCTION	PIN	LOGIC CODE
EXD00-	Data Output	k	DI
EXD01-	"	l	DI
EXD02-	"	m	DI
EXD03-	"	n	DI
EXD04-	"	p	DI
EXD05-	"	r	DI
EXD06-	"	s	DI
EXD07-	"	t	DI
EXD08-	"	u	DI
EXD09-	"	v	DI
EXD10-	"	w	DI
E XD11-	"	x	DI
RT1B-	Timing	y	DI
RT3B-	"	z	DI
RT4B-	"	AA	DI
RT5B-	"	BB	DI
RT7B-	"	CC	DI
EXTINT-	Interrupt	DD	N
EXMRDWR-	Read/Write Control	EE	N
		FF	
		HH	
		JJ	
EXMRQ1-	DMA Request	KK	N
RT0B-	Timing	LL	DI
RT2B-	"	MM	DI
RT6B-	"	NN	DI

APPENDIX A
INTERFACE SIGNATURE DEFINITIONS

1. ACOOB through AC11B

   This signature is the I/O accumulator outputs. They are the buffered accumulator output data Buss lines and appear at the connector as low True Signals. These data buffers present the contents of the accumulator to the I/O connector at all times.

2. DF

   This is the I/O interface input to the CPU for the device flag Buss. For the I/O instruction, "skip no device flag" (SDF), this signature must be taken to 0 volts for the central processor to execute the next instruction. Otherwise, the central processor will skip the next instruction on an SDF instruction.

3. DMAOO through DMA11

   These are the external DMA address input lines provided by the external interface to provide the DMA control unit with addressing for the memory unit. These lines must be present during either DMA read or DMA write (Input).

4. DMDOO through DMD11

   These are the signatures provided by the external DMA to provide the DMA input control with data to be input into the memory (Input).

5. DRDYB

   This signature is a buffered internal device ready signal from the central processor to the I/O interface. It notifies the external I/O interface that the central processor acknowledges a device ready on the IDRDY Buss. The CPU will perform the specified operation and will execute the next ordered instruction (no skip) at this time.

6. DSTB

This is the decode of "input device status" (DST) I/O operation code. To completely provide the I/O DST, it is conditioned with IOP.

7. EXDOO through EXD11

These are the lines provided by the DMA control for the external DMA interface to receive the data from the memory during a read cycle (Output).

8. EXMRDWR

This signature is "external memory read/write" and is the single line input provided by the DMA for the direct memory access read / write control line. When this signature is taken to 0 volts by the external connector, the memory will be in a write condition (Input).

9. EXMRQ1

This is the signature provided by the external DMA interface to the DMA control to initiate a DMA cycle. If the external connector takes this signature to 0 volts, it will initiate a direct memory access cycle at the next T4 time interval.

10. EXTINT

This is the external interrupt signal line. If the external interface takes this signature to 0 volts, it will initiate an interrupt cycle at the end of the current instruction.

11. IDRDY

This is the device ready Buss. It is for the I / O interface to input to the central processor that the device is ready. If the interface takes this signature to 0 volts, it will notify the central processor during the I/O skip instructions that the external device is ready and to not skip.

12. INOO through IN11

This signature is the input data to the CPU and is provided for the external I / O interface to input data to the AC Register. This is a data Buss for program data transfers and the data will be input to the AC under central processor control.

13. IOE

This signature is for I/O error, and is used by the I / O interface to notify the central processor that there has been an I/O error (parity, fetch, overrun, etc.). If this signature is taken to 0 volts by the external connector, the I/O error flip-flop within the status register will be set and the CPU notified of the I/O error.

14. IOPB

This is the buffered decoded I/O instruction code. It is decoded from R0 through R 3 and is present only when the I / O instruction code $(0001)_2$ is present.

15. MRQ1B

This is the buffered signal notifying the external DMA interface that the central control unit acknowledges the DMA request and is executing a direct memory access cycle.

16. RTOB through RT7B

The signatures are provided to the interface by the CPU timing control unit and are the master control timing for the central processing unit. These are available to direct memory access and I/O for external synchronous timing and control use (Output).

17. RO4B through R11B

These signals are the buffered instruction register outputs to the I/O connector. Through this data Buss, the I/O interface may monitor the instruction register bits 4 through 11 at all times during processing.

18. SDFB

This is the buffered decode of skip no device flag and is a decode of R4 thru R 6 of the instruction register. For the interface to completely provide the skip on device flag OP code, this signal is conditioned with IOP.

19. TFAB

This is the buffered decode of the TFA instruction. It is decoded from R4 through R 6 and to completely provide the I / O TFA, it is conditioned with IOP in the CPU.

20. TTAB

This is the buffered decode of the "Transmit to A" (TFA) I/O instruction OP code. It is decoded from R4 thru R 6 and to be completely conditioned, it is gated with IOP in the central processor.

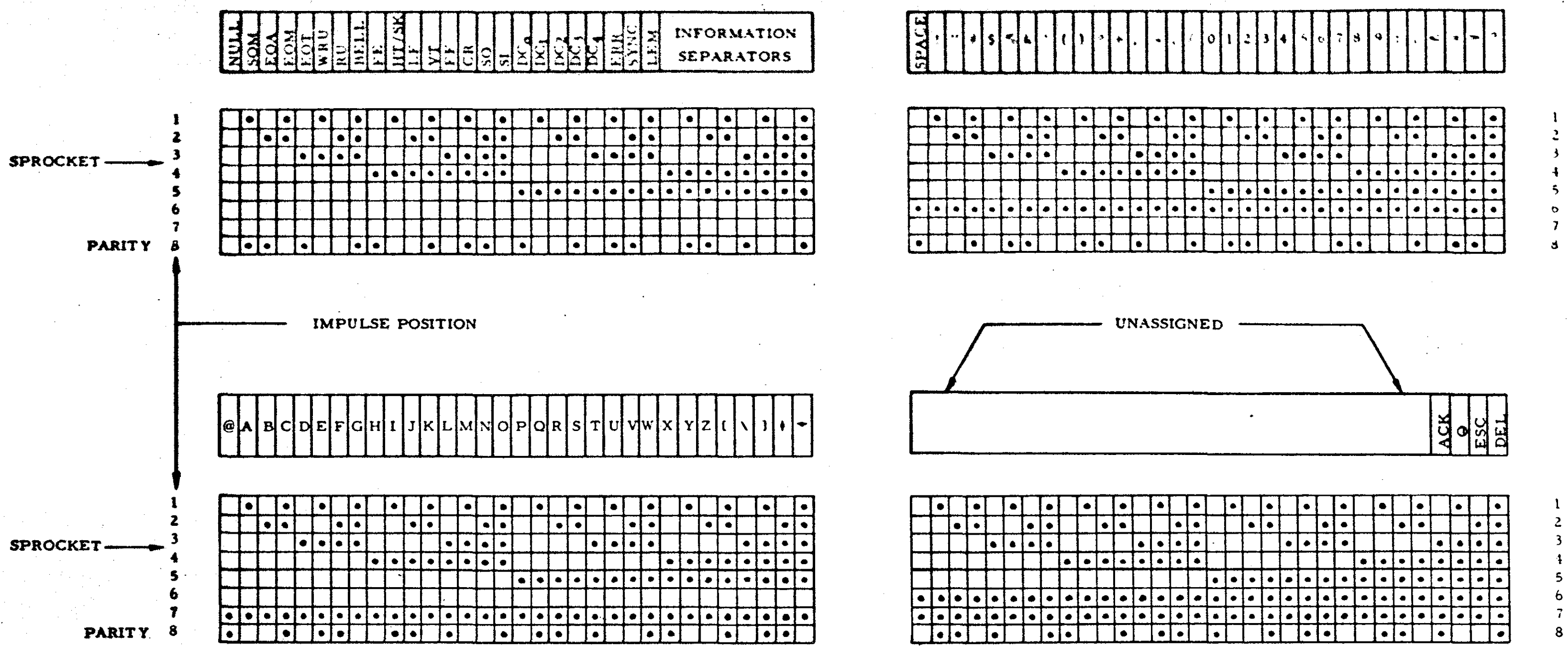
21. ZERO

This is a ground return or ground wire from the central processor power control unit. It is 0 volts and may be used for grounding external devices, or referencing the devices to central processor ground.

APPENDIX B

ASCII CODES AS THEY APPEAR IN "A" REGISTER

LOWER CASE	"A" REGISTER	LOWER CASE	"A" REGISTER	LOWER CASE	"A" REGISTER
A	0101	W	0327	[	0333
B	0102	X	0330	\	0134
C	0303	Y	0131	]	0335
D	0104	Z	0132	'	0336
E	0305	LF	0012	←	0137
F	0306	CR	0215	@	0300
G	0107	SP	0240	!	0041
H	0110	,	0254	"	0042
I	0311	-	0055	#	0243
J	0312	.	0056	$	0044
K	0113	/	0257	%	0245
L	0314	0	0060	&	0246
M	0115	1	0261	'	0047
N	0016	2	0262	(	0050
O	0317	3	0063	)	0251
P	0120	4	0264	+	0053
Q	0321	5	0065	<	0074
R	0322	6	0066	=	0275
S	0123	7	0267	>	0276
T	0324	8	0270	?	0077
U	0125	9	0071	BELL	0207
V	0226				



ASCII 8-LEVEL PAPER TAPE FORMAT

LEGEND

NULL	Null/Idle
SOM	Start of Message
EOA	End of Address
EOM	End of Message
EOT	End of Transmission
WRU	"Who are you?"
RU	"Are you. . . ?"
BELL	Audible Signal
FE_o	Format Effector
HT	Horizontal Tabulation
SK	Skip (punched card)
LF	Line Feed
V_{TAB}	Vertical Tabulation
FF	Form Feed
CR	Carriage Return
SO	Shift Out
SI	Shift In
DC_o	Device Control Reserved for Data Link Escape

DC_1-DC_2	Device Control
DC_1 (Stop)	Device Control (Stop)
ERR	Error
SYNC	Synchronous Idle
LEM	Logical End of Media
S_o-S_1	Separator (Information)
b	Word Separator (space, normally non-printing)
<	Less Than
>	Greater Than
↑	Up Arrow (Exponentiation)
←	Left Arrow (Implies/Replaced By)
\	Reverse Slant
ACK	Acknowledge
①	Unassigned Control
ESC	Escape
DEL	Delete/Idle

APPENDIX B

APPENDIX C

TABLE OF POWERS OF 2

2^n	n	2^{-n}
1	0	1.0
2	1	0.5
4	2	0.25
8	3	0.125
16	4	0.062 5
32	5	0.031 25
64	6	0.015 625
128	7	0.007 812 5
256	8	0.003 906 25
512	9	0.001 953 125
1 024	10	0.000 976 562 5
2 048	11	0.000 488 281 25
4 096	12	0.000 244 140 625
8 192	13	0.000 122 070 312 5
16 384	14	0.000 061 035 156 25
32 768	15	0.000 030 517 578 125
65 536	16	0.000 015 258 789 062 5
131 072	17	0.000 007 629 394 531 25
262 144	18	0.000 003 814 697 265 625
524 288	19	0.000 001 907 348 632 812 5
1 048 576	20	0.000 000 953 674 316 406 25
2 097 152	21	0.000 000 476 837 158 203 125
4 194 304	22	0.000 000 238 418 579 101 562 5
8 388 608	23	0.000 000 119 209 289 550 781 25
16 777 216	24	0.000 000 059 604 644 775 390 625
33 554 432	25	0.000 000 029 802 322 387 695 312 5
67 108 864	26	0.000 000 014 901 161 193 847 656 25
134 217 728	27	0.000 000 007 450 580 596 923 828 125
268 435 456	28	0.000 000 003 725 290 298 461 914 062 5
536 870 912	29	0.000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0.000 000 000 931 322 574 615 478 515 625

AREAS OF APPLICATION

I. Business and Manufacturing in General

1. Office
   - Cost analysis
   - Forecasting
   - Inventory control
   - Mailing list operations
   - Management strategy analysis
   - Performance evaluation
   - Price analysis
   - Purchase order writing
   - Sales forecasting
   - Wage and salary analysis
   - Work-in-process records

2. Plant and Production
   - Assembly line balancing
   - Lathe operations: automatic control
   - Procurement
   - Production scheduling
   - Quality control records
   - Route accounting (bakeries, bottling plants, dairies, etc.

3. Libraries
   - Information retrieval
   - Records and control

4. Magazine Publishing
   - Classified advertisement preparation
   - Mailing list maintenance
   - Subscription fulfillment

5. Oil Industry
   - Depletion accounting
   - Map construction
   - Oil purchase accounting
   - Operating records: logging
   - Remote control of oil production
   - Seismic data reduction
   - Well logs: corrections

6. Textile Industry
   - Fabric quality control
   - Material availability evaluation
   - Production planning
   - Sales analysis
   - Style reporting

7. Transportation
   - Air traffic control
   - Aircraft maintenance scheduling
   - Collision warning systems
   - Navigating systems
   - Preventive maintenance scheduling
   - Travel reservations

II. Science and Engineering

1. Aeronautics and Space Engineering
   - Curve fitting
   - Flight control for missiles and space vehicles
   - Flight simulation
   - Flight test data reduction
   - Heat transfer analysis
   - Vibration analysis
   - Wind tunnel data reduction

2. Chemical Engineering and Chemistry
   - Gas line calculation
   - Process control
   - Spectrum analysis

3. Civil Engineering
   - Abutment design
   - Beam design
   - Cut and fill calculations
   - Earthwork computations
   - Freeway assignment
   - Highway profiles
   - Monthly equipment summary
   - Stress analysis
   - Triangulation

4. Electrical Engineering
   - Component design
   - Computer logic circuits; design by numerical control
   - Feedback system, single loop, finding the root locus

5. Mechanical Engineering
   - Air conditioning calculations
   - Heat flow
   - Machine vibration analysis

6. Medicine and Physiology
   - Telemetering and analysis of Medical data
   - Probability in medical diagnosis

7. Military Engineering
   - Ballistic trajectories
   - Fire control

SCC maintains complete support activities for its users. Installation and maintenance services are available through SCC offices strategically located throughout the United States. For pre-procurement demonstration of hardware and programs in Dallas, contact local sales office or the Marketing Department in Dallas.

Arlington, Massachusetts
30 Park Avenue
617 — 648-2922
(Boston)

Skokie, Illinois
125 Old Orchard Arcade
312 — 675-6700
(Chicago)

Houston, Texas
7800 Westglen Drive
713 — 782-9851

Seattle, Washington
1806 South Bush Place
206 — 324-7911

Midland Park, New Jersey
36 Central Avenue
201 — 652-6750
(New York)

Crofton, Maryland
Village Green
301 — 647-6431
(Baltimore)

Orlando, Florida
2319 E. South Street
305 — 841-3556

Pasadena, California
180 East California Blvd.
213 — 681-2651
(Los Angeles)

Other SCC products include: telemetry systems and airborne signal conditioning equipment such as amplifiers, demodulators and converters.

Scientific Control Corporation
14008 Distribution Way • Dallas, Texas 75234 • 214 — 241-2111

102 6 66812

Printed in U.S.A.