PROGRAMMED DATA TRANSFERS

Movement of data to and from input/output devices under direct stored program control is called Programmed Data Transfers (PDT).

PDT CONTROL LOGIC

The signals involved in PDT's are:

BACØ through BAC19	(Buffered Accumulator)
SACØ- through SAC19-	(Set Accumulator not)
BIR3 through BIR7	(Buffered Instructor Register)
IOP	(Input/Output)
IOP. through IOP3	(Input/Output Pulse)
SKIP-	(Skip the next instruction)

SACØ- through SAC19- and SKIP- are wired-or signals (see Appendix 10 describing 1080 logic conventions). BAC's are usually considered to be data leaving the processor and SAC's are used to give data to the processor. The input/output pulses are generated by the processor and are individually programmable. They control the movement of data to and from the processor. The BIR's are decoded by an input/output device to determine what command the processor is giving. IOP is simply an indication that an input/output command is being given. The SKIP- line is usually used to indicate to the processor that an input/output device is busy.

Actively driven data and control signals on the input/output connector which originate in the 1080 and go from external device to external device must be terminated as shown in Fig. 1. A special terminating connector is provided. The same termination is required of data or control signals that are determined by external devices which are connected to the 1080 input/output connector. The principle of operation is shown in Fig. 2. This is called a wired "or." An MC858P is the only recommended integrated circuit in this application.

Some input/output devices are fast enough to keep up with a computer such as, for example, a CRT display. Any mechanical device, however, will require some sort of scheme for limiting the rate of data flow. Of course the opposite problem can occur. The computer may not be able to compute fast enough to keep up with an input/output device. The problem, then, is to design control logic that requires either the processor or the input/output device to wait for the other. Control logic that has this property is called an asynchronous interface.

Fig. 3 illustrates such an interface to an output device. The signal HYPF, which comes from a flip flop, is called a Flag and it is used to indicate when the output device is willing to accept more data. The processor can find out the state of the Flag by putting the proper number on the BIR lines and generating IOP1. If the Flag is up, an instruction will be skipped, because the interface logic brought SKIP- to ground. The processor clears the flag (causes HYPF to go low) with the same number on the BIR lines and IOP2. The same instruction presents data to the output device and commands it to respond to the data. When the hypothetical device is ready for new data, it causes the flag to go up. This, in principle, is how interface logic for Teletype or high speed reader works.

Fig. 4 shows the timing for the important signals. The BIR's are changing constantly when the processor is running and so commands must be gated with input/output pulses. The IOP pulses are separated in time and can be individually programmed. The only restriction on their use is that IOP3 may not cause a skip command.

SIGNAL

BACØ - BAC 19
Buffered Accumulator contents
Active pull-ups inside 1080
High = 1
Used to transfer data from 1080

BIR3 - BIR7
Buffered Instruction Register contents
High = 1
Gated with IOP for selecting I/O devices

BL
Buffered Link contents
Active pull-up inside 1080
High = 1
Used to transfer data from 1080

IOP
Input-Output indicates external I/O instruction
Active pull-up in 1080
High = 1

IOP1
Input-Output timing pulse 1
Occurs during I/O instruction when Bit 2 is set
Active pull-up
High = 1

IOP2, IOP3
Same as IOP1 except caused by Bits 1 and Ø

Active-
Asserted by the 1080 when it is in either hardwired or stored program operation. Low = 1 = Assertion

Skip-
Used to cause the 1080 to skip an instruction. Should be asserted only during an I/O instruction.
Low = 1 = Assertion

SACØ- to SAC19-
Set Accumulators. Used to transfer data to the 1080 accumulator. Should be asserted only during an I/O instruction.
Low = 1 = Assertion

XRAC-
External Reset Accumulator. Allows the accumulator to be cleared externally. Care must be exercised to prevent destruction of data or program.
Low = 1 = Assertion

XREAD-
External Read. Used to cause a 1080 Memory Read cycle. Should be asserted for ~ 750 nsec.
Low = 1 = Assertion

XWRITE-
External Write. Used to cause a 1080 Memory Write cycle. Should be asserted for ~ 750 nsec.
Low = 1 = Assertion

XLAC-
External Load Accumulator into Arithmetic unit.
Low = 1 = Assertion

XLMB-
External Load Memory Buffer into Arithmetic unit.
Low = 1 = Assertion

XTDBMB-
External Transfer Data Bus to Memory Buffer.
Low = 1 = Assertion

XTDBAC-
External Transfer Data Bus to Accumulator.
Low = 1 = Assertion

XADPC-
External Transfer Address Data to Program Counter.
Low = 1 = Assertion

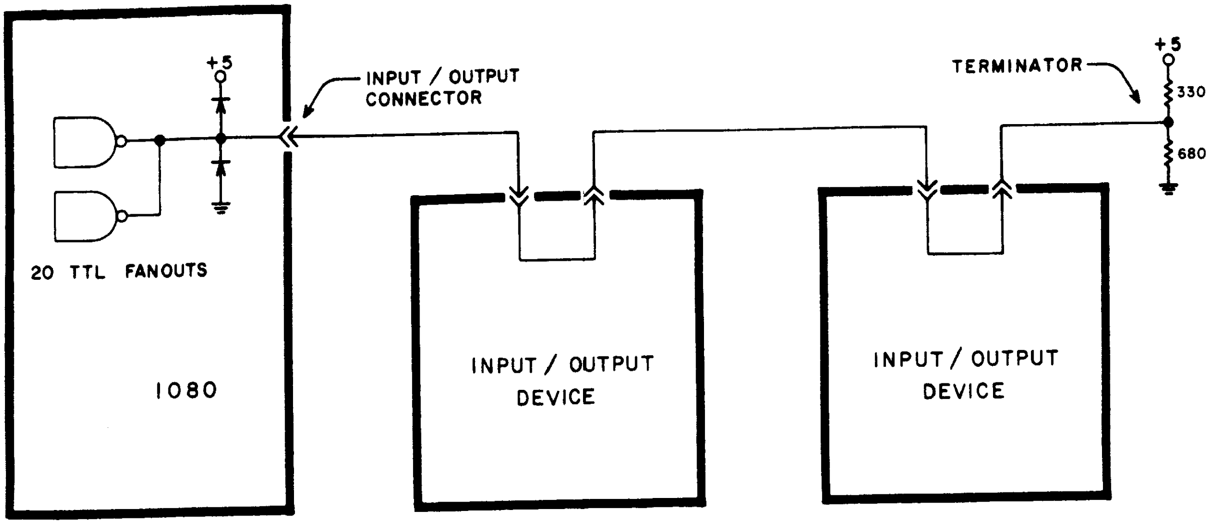
XIPC-
External Increment Program Counter. Used for sequential addressing.
Low = 1 = Assertion

XTMBPC-
External Transfer Memory Buffer to Program Counter.
Low = 1 = Assertion

XSTART-
External Start. Equivalent of "Continue-Execute." Starts stored program operation at the current contents of the Program Counter.
Low = 1 = Assertion

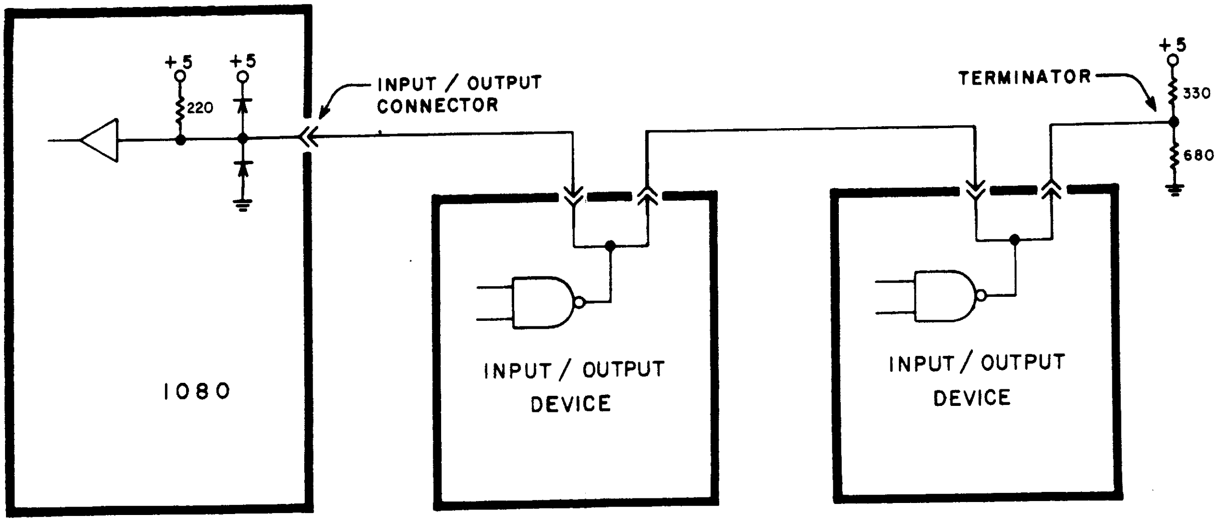
XSTOP-
External Stop. Equivalent to pushing Stored Program "STOP"
Low = 1 = Assertion

PSTART-
Program Start. Equivalent to pushing Stored Program "START"
Low = 1 = Assertion



INPUT / OUTPUT DEVICE ACTS AS RECEIVER.
THE LINE IS ACTIVELY DRIVEN BY THE 1080.

Fig. 1



WIRED—OR LINES.
USED WHEN INPUT / OUTPUT DEVICE MUST
BE ABLE TO CONTROL STATE OF LINES.

Fig. 2

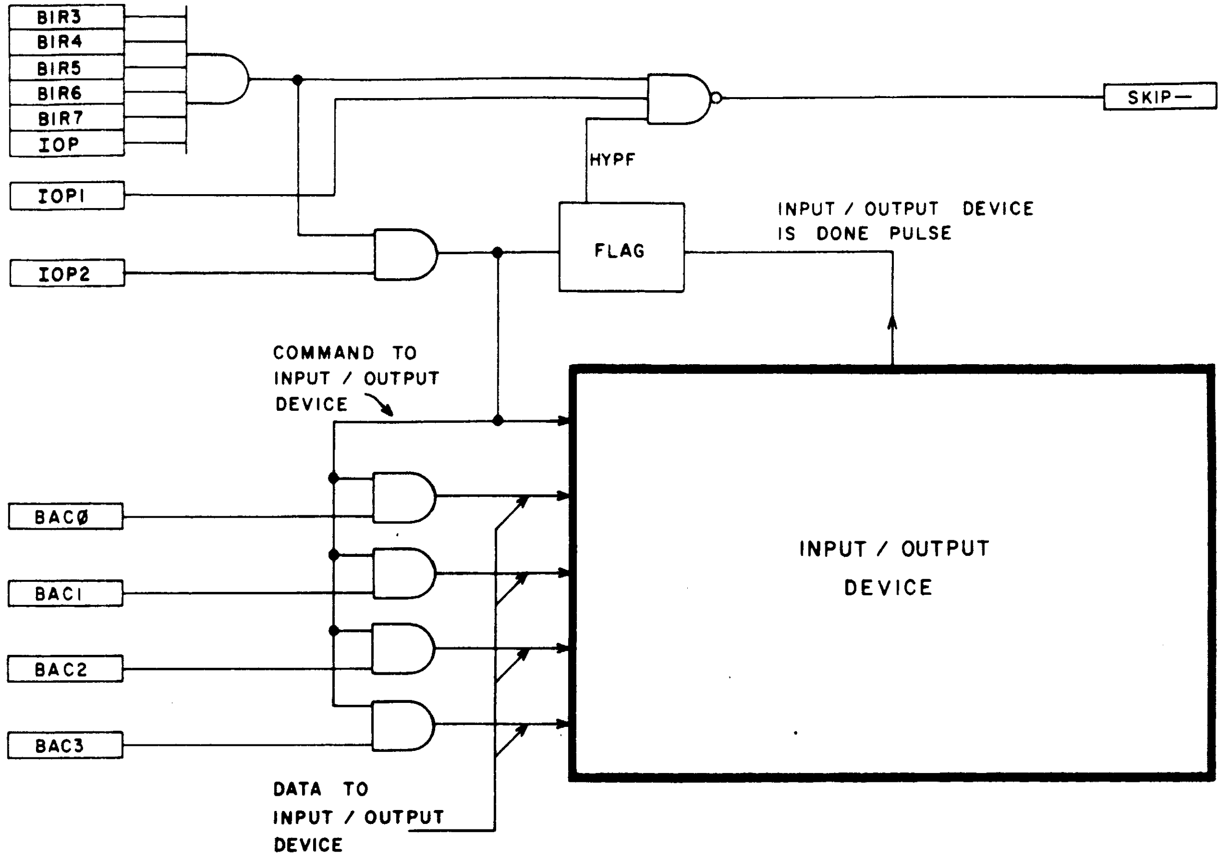
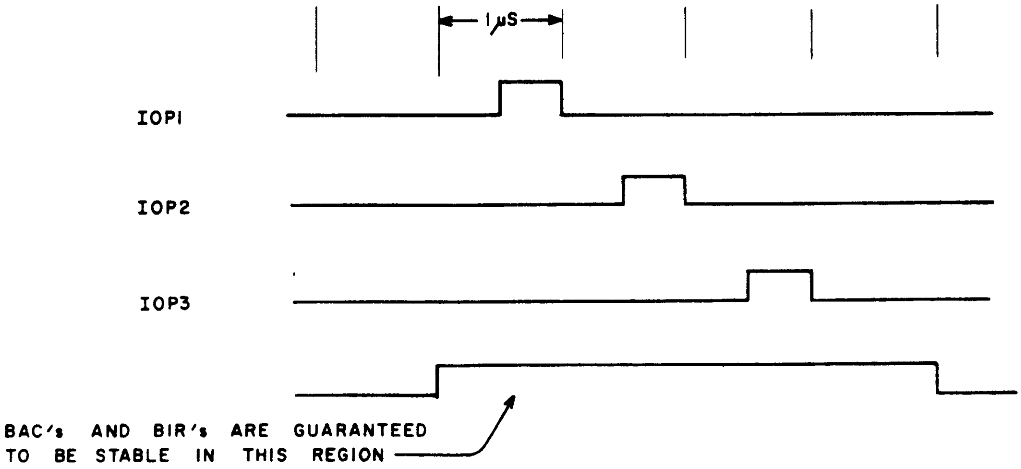


Fig. 3 **ASYNCHRONOUS INTERFACE TO HYPOTHETICAL DEVICE**



NOTE:
TIMING IS ACCURATE TO 5%

Fig. 4 **PROGRAMMED DATA TRANSFER TIMING**

INTERFACING THE 1080

The input output (I/O) connector at the rear of the 1080 provides a method for connecting digital logic to the 1080. This connector is designed so that one can connect standard computer I/O devices such as tape readers or punches. However, it is so versatile that it would be possible to connect an external computer and have it control the memory and arithmetic capabilities of the 1080.

Obviously, all intermediate capabilities are also present, making it possible to design logic modules for control of experiments. These modules can then be programmed to give information to the 1080 and to accept signals from it using the I/O connector. The list of available signals is shown on the next page.

INPUT/OUTPUT CONNECTOR

R3

EVEN PINS		ODD PINS	
2	BACØ	1	SACØ-
4	BAC1	3	SAC1-
6	BAC2	5	SAC2-
8	BAC3	7	SAC3-
1Ø	BAC4	9	SAC4-
12	BAC5	11	SAC5-
14	BAC6	13	SAC6-
16	BAC7	15	SAC7-
18	BAC8	17	SAC8-
2Ø	BAC9	19	SAC9-
22	BAC1Ø	21	SAC1Ø-
24	BAC11	23	SAC11-
26	BAC12	25	SAC12-
28	BAC13	27	SAC13-
3Ø	BAC14	29	SAC14-
32	BAC15	31	SAC15-
34	BAC16	33	SAC16-
36	BAC17	35	SAC17-
38	BAC18	37	SAC18-
4Ø	GND	39	GND
42	BAC19	41	SAC19-
44	BIR3	43	XRAC-
46	BIR4	45	XREAD-
48	BIR5	47	XWRITE-
5Ø	BIR6	49	XLAC-
52	BIR7	51	XLMB-
54	BL	53	XTDBMB-
56	IOP	55	XTDBAC-
58	IOP1	57	XADPC-
6Ø	IOP2	59	XIPC-
62	IOP3	61	XTMBPC-
64	ACTIVE-	63	XSTART-
66	SKIP-	65	XSTOP-
78	+5	67	PSTART-
8Ø	GND	77	+5
		79	GND