F-831F

INTERFACE MANUAL

PDP-8

DIGITAL EQUIPMENT CORPORATION • MAYNARD, MASSACHUSETTS

The PDP-8 is a solid-state device designed for use as a small-scale general-purpose computer, an independent information handling facility in a larger computer system, or as the control element in a complex processing system.

This computer is a single-address, fixed word length, parallel processing computer using 12-bit, two's complement arithmetic. Cycle time of the 4096 word random access core memory is 1.5 microseconds. The 1.5 microsecond cycle time permits a computation rate of up to 333,333 additions per second. Other programming features include indirect addressing, microprogramming (combining instructions to occur within one machine cycle time of 1.5 μsec), and programmed monitoring of peripheral devices.

"Real-Time" features of this processor include program interrupt, In-Out Skip Facility (program flow modification as a function of status of selected peripheral device), high speed data channel (direct input of data to the computer at a rate up to 8 million bits per second), and program "pause". Over 100 peripheral devices may be attached to the computer, whether standard or non-standard devices. Simultaneity of these devices is limited by the basic speed of the computer and by the number of channels available (over 100). (For a figure of comparison, it takes less than 2% of the computer's time to read or write conventional IBM compatible tape.)

High-capacity, high-speed, I/O capabilities of the PDP-8 allow it to operate a variety of peripheral equipments. In addition to the standard Teletype and perforated tape equipment, the system is capable of operating in conjunction with a number of optional devices such as card equipment, line printers, magnetic tape transports, magnetic drums, analog-to-digital converters, digital-to-analog converters, CRT displays, and digital plotters.

Interfacing other special devices to the computer requires no modification to the processor and may be achieved in the field.

Eight auto-indexing registers are provided for program searching, multiple input/output list processing, and sorting operations.

A switch register is provided on the console for manual entry of data and instructions. This also provides a facility for the computer to "sense" conditions set in by the operator. Any one of 4096 possible conditions may be inserted into the switches.

All active registers are continuously displayed, including memory address register, memory buffer register, accumulator, Link Bit, instruction being performed, machine state, program counter and optional Multiply/Divide register.

The processor is composed of integrated silicon circuits whose temperature operating range exceeds the limits of 32°F to 130°F. The basic machine is self-contained in the table top version which includes the central processor, core memory, and power supplies. The machine runs off standard "house-power" of 115V, 60 cps. No air-conditioning or special sub-flooring is required. Height of the basic machine is approximately 32". Weight is approximately 250 lbs.

FLIP CHIP is a trademark of
DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS

Copyright © 1965 by
Digital Equipment Corporation

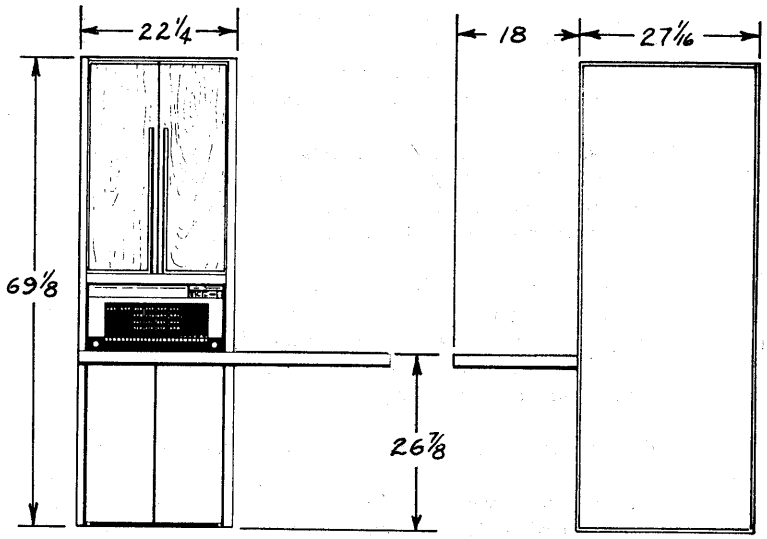
Digital Equipment Corporation makes no representation that the interconnection of its modular circuits in the manner described herein will not infringe on existing or future patent rights. Nor do the descriptions contained herein imply the granting of licenses to make, use, or sell equipment constructed in accordance therewith.

It is the policy of DEC to give assistance, where possible, in the design of special interfaces to the PDP-8. Should any questions arise relative to the data in this manual or subjects not sufficiently covered, the reader is invited to call the main plant in Maynard, Mass. or any of the engineering offices.

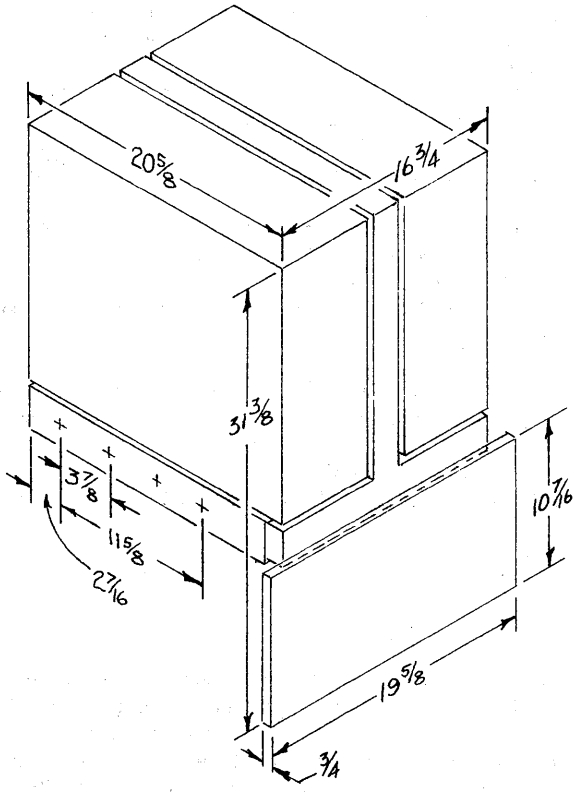
CONTENTS

DIMENSIONAL DATA

PDP-8 mounted in CAB-8B Cabinet

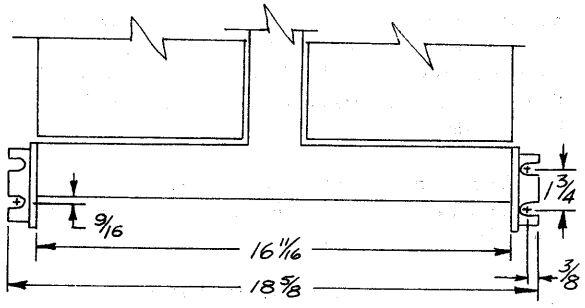


Components

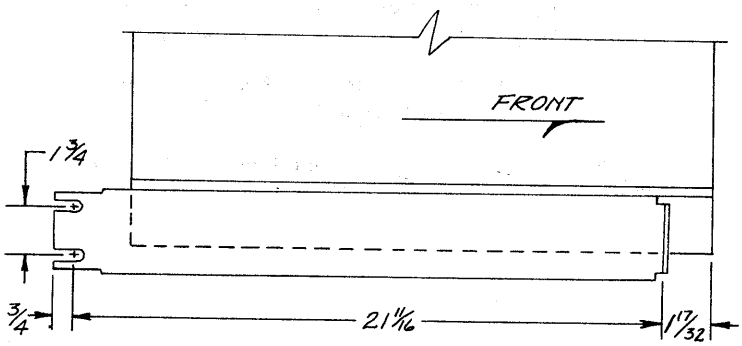


Processor and Memory

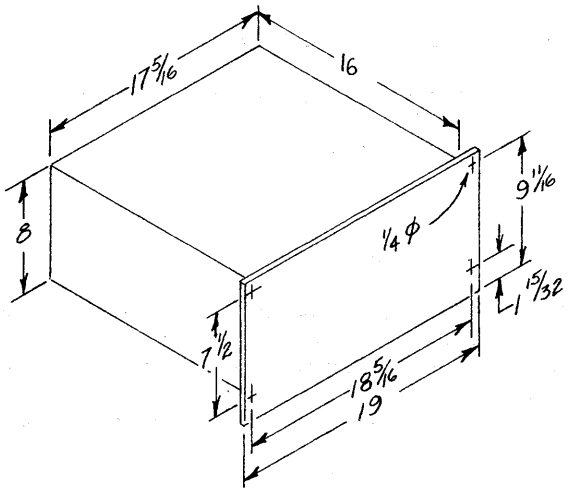
Mounting Hardware



Front View



Side View



Power Supply

INPUT-OUTPUT CONTROL

In order to realize a benefit from a computer system, some condition-description, data, information or other coded phenomenon must be inserted into the computer. (That is, the image or binary equivalent must be inserted.) The computer then, according to a previously arranged program or sequence of instructions, will perform logical and/or arithmetic operations internally either on the input data or as a function of the input data on other states, conditions or data.

As a result of this performance, the computer must then put data out in some form interpretable by the outside sensors (electrical transducers, human eyes, relay coils, etc.). These functions of inserting and extracting codes is commonly called input/output.

For most general digital computers, the proper interface exists to permit communication between the computer and the outside mechanical/electrical gadgetry, such as, typewriters, magnetic tape storage units, line printers, etc. These interfacing problems are solved. However, many more gadgets and devices do exist for which no interface has been designed. The user is often left with this design/development responsibility since economics prevents (strongly discourages) contract design work for a one-of-a-kind system.

Fortunately for the user, the I/O techniques utilized in the design of the PDP-8, permit inexpensive designs to be realized, designs which are born with comparative ease once the simple I/O techniques are understood.

Control of some sort is needed to determine when this exchange information is to take place and the location(s) in the computer's memory which will accept (yield) the data. This control may be exercised either by the computer or the device external to the computer.

Data transfers made at times controlled by the computer, hence under control of its stored program, are called programmed data transfers. Transfers made at times controlled by the external devices are called Data Break transfers.

Programmed Transfers

The great majority of I/O transfers take place under control of the computer, thereby taking advantage of the built-in control elements. Although programmed transfers take more of the computer's time and also take more actual time to get into the computer, it isn't often that these limitations come into play. Program transfers can handle data rates up to one million bits of information per second. Normally this is well beyond that required for typical laboratory or process control instrumentation.

Devices which typically are controlled through the programmed I/O section are as follows:

Analog-to-Digital Converters

Digital-to-Analog Converters

Digital Plotters

Line Printers

Relay Control Systems.

The simple process of transferring and storing of information under program control requires about six times as much computer time as the Data Break type transfer. This time, however, in terms of actual real time on a machine as fast as the PDP-8 is rather small. The user should, in most cases, use a programmed I/O transfer to realize the full benefits of the built-in control features of the PDP-8.

See page 6 for a more complete description of programmed transfers.

Data Break Transfers

Some external devices operate at very high speeds and are better handled through the Data Break facility. This facility permits the external device to almost arbitrarily insert or extract words from the computer's memory, bypassing the program control section. The disadvantage of this type of transfer is that the computer program has no cognizance of the transfer being made.

The Data Break is particularly well-suited for devices which transfer large amounts of information in block form. Some of these devices are as follows:

High Speed Magnetic Tape Drives

High Speed Drum Memories

CRT Display Systems.

Since the control exists externally, the interface cost is normally higher for a Data Break I/O interface.

The combined maximum transfer rate of the Data Break facility is eight million bits per second.

See page 12 for a more complete description of Data Break transfers.

PROGRAMMED DATA TRANSFERS

The PDP-8 is a parallel machine and collects and distributes data in 12 bit bites.

All programmed data transfers between the computer and peripheral devices connected to the computer take place through the Accumulator, the 12 bit arithmetic register.

Figure 1 illustrates the 12 bits of the accumulator plus the Link bit. The status of the Link bit is not available to be tested by peripheral devices. Coming from each flip flop in the Accumulator is a bus line which is continually representative of the status of that particular flip flop. Although figure 1 does not show it, the bus line is buffered from the actual accumulator flip flop by a bus driver. For sake of simplicity the bus driver, or amplifier, is not shown.

Since these bus lines continually track and represent the status of the flip flop, one needs only to sample these bus lines at some particular time to sense the value being held by the accumulator.

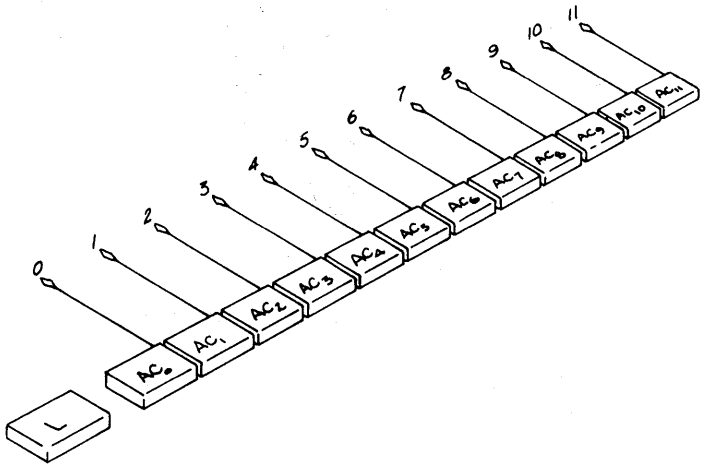


Fig. 1 12-bit Accumulator and Link Bit

In figure 2, 6 bits of the accumulator are sampled by a strobe pulse to conditionally set an external 6 bit data register. Since this is not a jam transfer it is necessary first to clear the external data register before setting ONES into it.

The read-in gates driving the external data register are part of the external device and are not supplied by the computer. Also the size of the data register may be any number of flip-flops up to a maximum of 12. (If more than 12 flip flops are involved, 2 or more transfers must take place.)

Obviously the Clear Pulse and the Strobe Pulse shown in figure 2 must come at the particular time when that data held in the accumulator is in truth the data that is to be placed in the external data register. These pulses therefore must be under computer control in order to by synchronized to the operation or program of the computer.

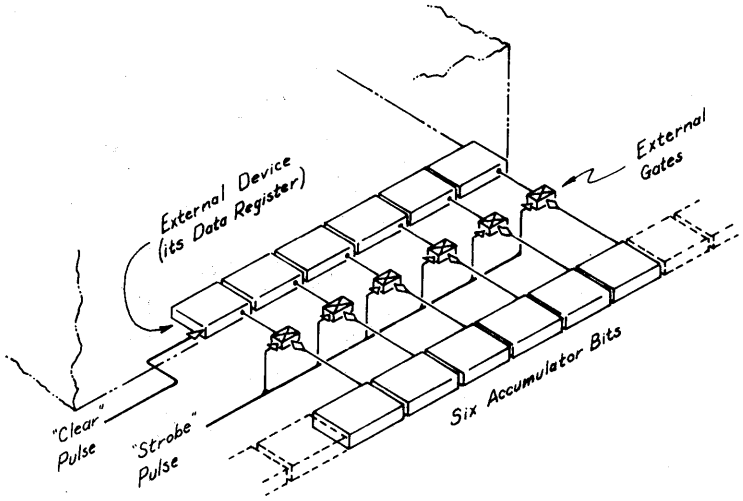


Fig. 2 Loading a 6-bit word into an External Device from the Accumulator

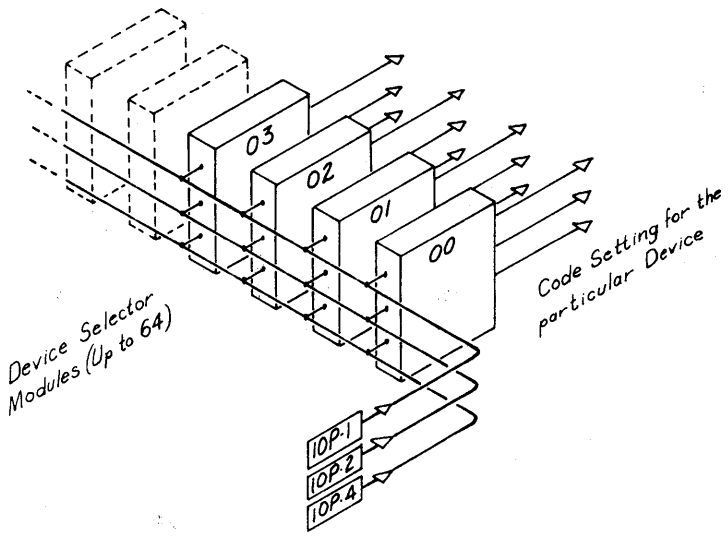


Fig. 3 Device Selectors and their Control Pulses

Figure 3 illustrates graphically the device selectors which may be plugged into the computer interface. These devices recognize the basic In/Out Transfer instruction (octal code 6) and activate IOP pulses 1, 2 and 4 which in turn are gated through the selected device selector module to go to the proper peripheral or external device.

Figure 4 illustrates the use of 2 of the pulses being gated by the device selector coded for "34". IOT pulse 1 (noted IOT 1) is used to clear the Data Register and IOT 4 is used to strobe the data from the Accumulator into the Data Register. Note that the pulses identified as IOP 1, IOP 2, and IOP 4 are internal processor pulses and are used only to activate the device selectors. The pulses put out by the device selectors are identified as IOT 1, IOT 2, and IOT 4.

As indicated in figure 4, 2 new instructions are now added to the repertoire of PDP-8 instructions for this particular system. One is at A and reads "Clear Data Register of Device Number 34". The other is at B and reads "Load Data Register of Device Number 34 with Contents of the Accumulator". (In actual use, of course, the instructions will be abbreviated).

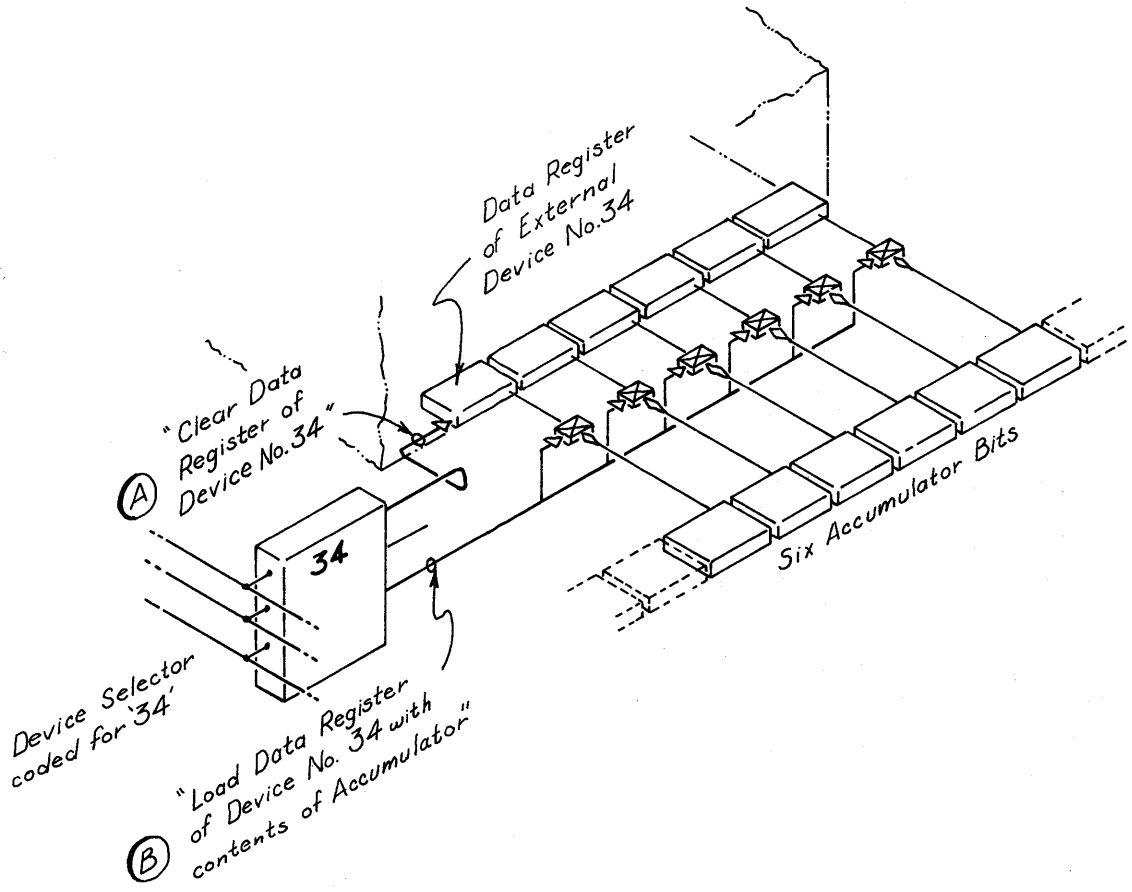


Fig. 4 Use of Device Selector for Activating and Controlling External Device

Figure 5 illustrates the decoding process for producing IOT pulses. The op code for the basic in/out group is 6. The 6 bits (or 2 octal digits) following the op code 6 identify the basic device number, in this case 34. The last 3 binary bits of the instruction (1 octal digit) identify which of the 3 IOT pulses are to be generated. In the example shown, the instruction 6341 will clear the data register in device number 34. The instruction 6344 will load the data register in device number 34 with the contents of the accumulator. IOT pulse 2 is not used and may be considered a spare.

Next it would be convenient if the computer could test the status of the device to determine if it is in truth ready to accept information. Since the computer is in most cases much faster than the external devices, it most often must wait a great deal before sending data to them. One such means for testing the status of external devices is the In/Out Skip Facility.

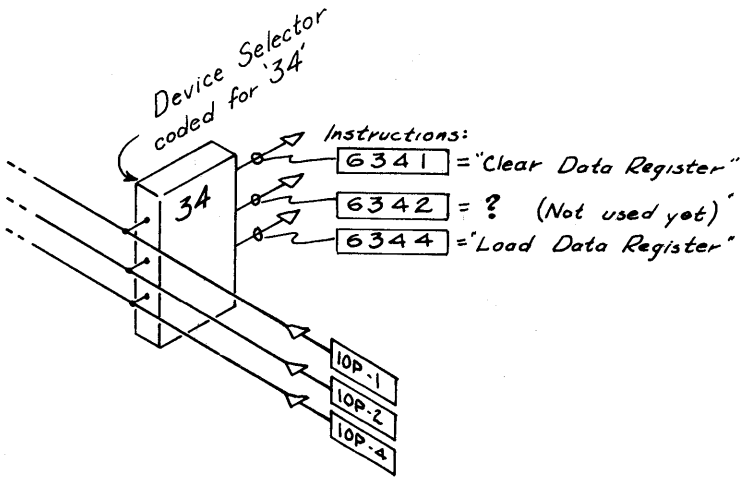


Fig. 5 Instruction Coding for Producing IOT Pulses

Figure 6 illustrates one use of the In/Out Skip Facility. In this case it is ANDing an IOT pulse with the status line of the external device (represented by a status flip flop) to skip the next sequential instruction should the device be ready to receive data. Here we may make use of IOT pulse 2 of the device selector coded 34.

The program sequence would read as follows:

6342	/skip if device 34 is ready
JMP .-1	/jump to the previous statement
6341	/clear data register of device 34
6344	/load data register of device 34

When the computer gets to this portion of the program, it will attempt to execute a skip with instruction 6342 provided that device 34 is ready. However, if 34 is not ready to accept data, the wrong signal will be ANDed with IOT pulse and it will not perform the skip. This causes the program to go to the next instruction in sequence which is a "jump" back to the skip instruction. The program will stay in this 2 instruction loop until the device is in reality ready, at which time the 2-input AND gate will be enabled, permitting a skip instruction to be performed. This will bring the program to the instruction 6341 which will clear the data register. Then instruction 6344 will be executed which will load the data register with the contents of the accumulator.

Assuming that the logic of the external device is fast enough, the instruction 6341 and 6344 may be inclusive ORed together to form the instruction 6345. This will both clear the data register and load it within one IOT instruction time. As a matter of fact, this is much more common than breaking the instructions apart, as shown in the above example.

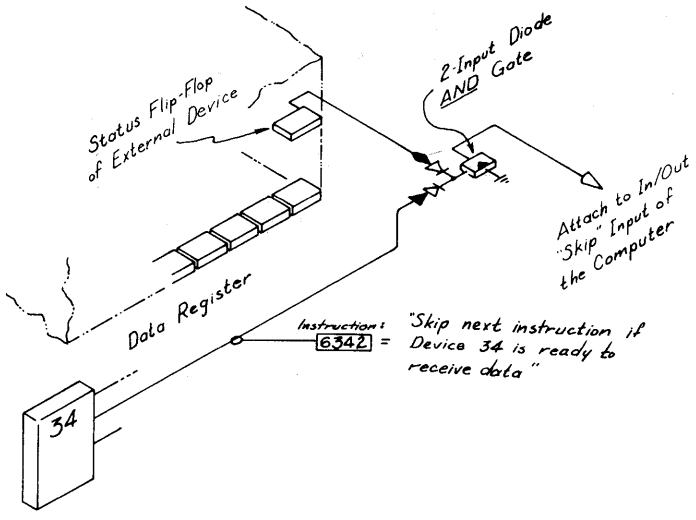


Fig. 6 Use of I/O Skip Facility to Test the Status of an External Device

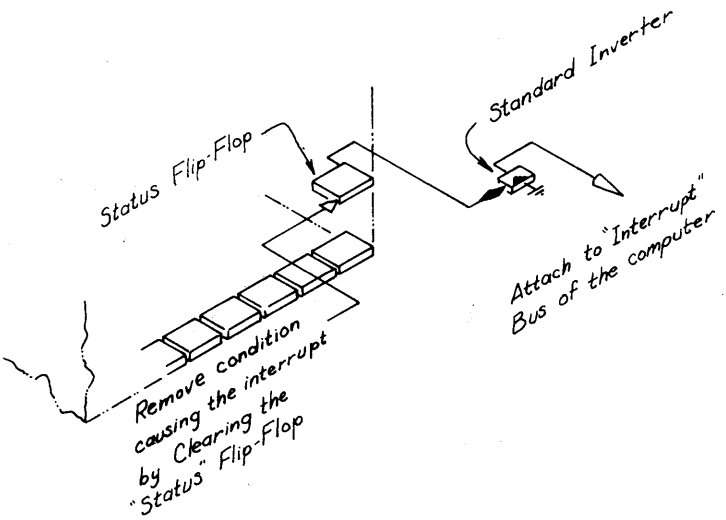


Fig. 7 Use of Program Interrupt

Quite often when there is a great deal of computing to be done, it is advantageous to permit the computer to do other things and not simply wait for some external device to permit it to transfer data. The Program Interrupt Facility permits the PDP-8 to busy itself with other things and branch to the in/out transfer only when signaled to do so by the external device.

In this case a status flip flop is connected to a standard inverter and causes a program interrupt when enabled. If this is the only device connected to the program interrupt facility, then the computer would be programmed to jump to a subroutine that would assemble more data to be transmitted to the device.

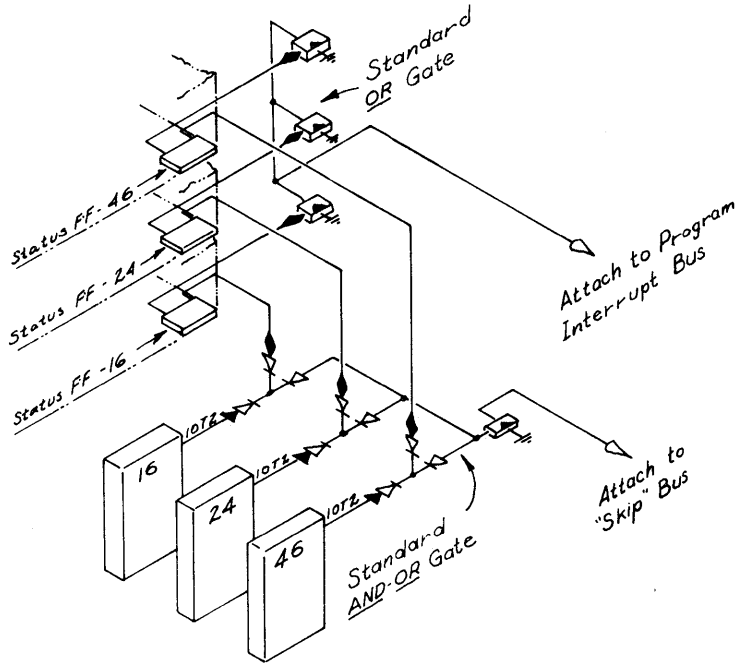


Fig. 8 Multiple Inputs to Interrupt and Skip Facilities

It is more common, however, to have numerous devices hooked to the program interrupt. In this case any one of the devices may cause a program interrupt and the computer must determine which of these devices interrupted its program. One way to do this is with the I/O Skip Facility.

Contrary to the approach taken in figure 6, here we will negate the condition of the skip, to skip if the device is not requesting service. Note in the figure that the status flip flops of the various devices are ORed together to permit program interrupt.

```
6162    /Skip if Device #16 is not requesting service
JMS X   /Jump to subroutine at "X" to service Device #16
6242    /Skip if Device #24 is not requesting service
JMS Y   /Jump to subroutine at "Y" to service Device #24
6462    /Skip if Device #36 is not requesting service
JMS Z   /Jump to subroutine at "Z" to service Device #46
6...    /Check other interrupt inputs (if any)
```

Fig. 9 Test Routine

IOT pulses again are used to AND with the status flip-flops to determine if a skip is permitted. In this case if a particular device is not requesting service the instruction caused to be skipped is that instruction to "jump to servicing routine for that device." The JMS instruction is the most convenient instruction to use for this. The skip routine in figure 9 illustrates that for each device not requiring service, one skip instruction is performed. Note that in determining the time required to service devices, the JMS instruction time should not be added since it is skipped.

So far in the examples above data transfers have been from the Accumulator to external devices. In order to transfer data from a device to the Accumulator, the above process is essentially reversed.

The illustration in figure 10 shows that the Accumulator has an input bus for each bit of the Accumulator. In order to set a 1 into a particular bit of the Accumulator, the bus must be brought to ground by the standard DEC inverter. In the illustration, the 2-input AND gates are used to set various bits of the Accumulator. In this case an IOT pulse is ANDed with the flip-flop states of the external device to conditionally set ONEs into the Accumulator. (The program must include a "Clear the Accumulator" instruction prior to loading in this manner, otherwise an inclusive OR function will take place between the previous contents of the Accumulator and the contents of the Data Register being read in). With devices loading data into the Accumulator, their status may be checked and tested as illustrated earlier.

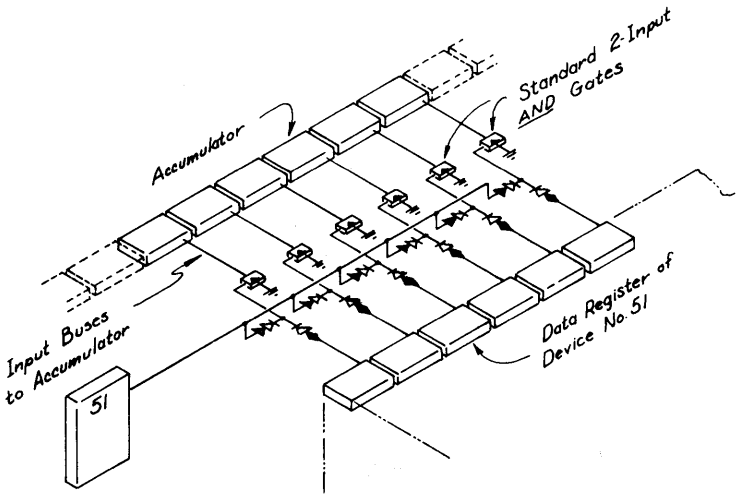


Fig. 10 Loading Data into the Accumulator from an External Device

The timing of the IOT cycle is shown in Figure 11. Note that the Accumulator Bus Drivers settle down as late as 400 ns prior to execution of IOP-1 pulse. Since Flip Chip type DCD gates require 400 ns set-up time, IOP-1 pulse should not be used to sample the state of the Accumulator through DCD gates. Rather it is suggested that IOP-1 be used to reset registers and/or clear flags. However, in those instances where it is desirable to gate out the contents of the Accumulator on the first IOP pulse, then the gates used should be diode gates.

IOP-2 and IOP-4 can be used to sample the state of the AC through DCD without any difficulty, providing the lead lengths of the AC lines are the same as those of the pulse lines providing the same transmission delays.

IOP-4 should not be used to activate the Skip facility. Use only IOP-1 or IOP-2.

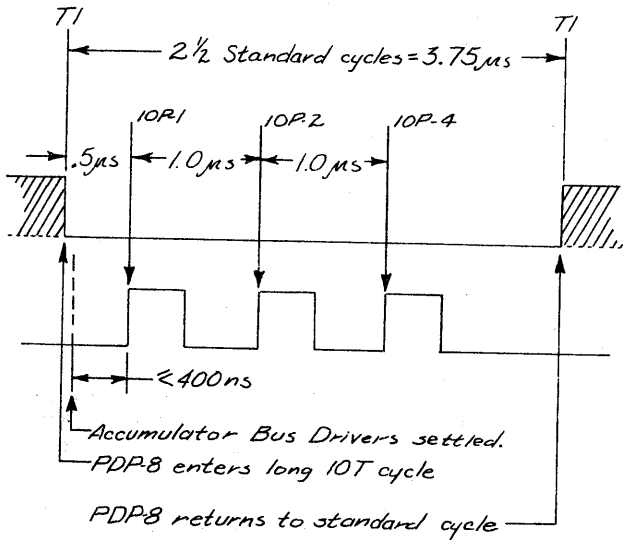


Fig. 11 IOT Timing Cycle

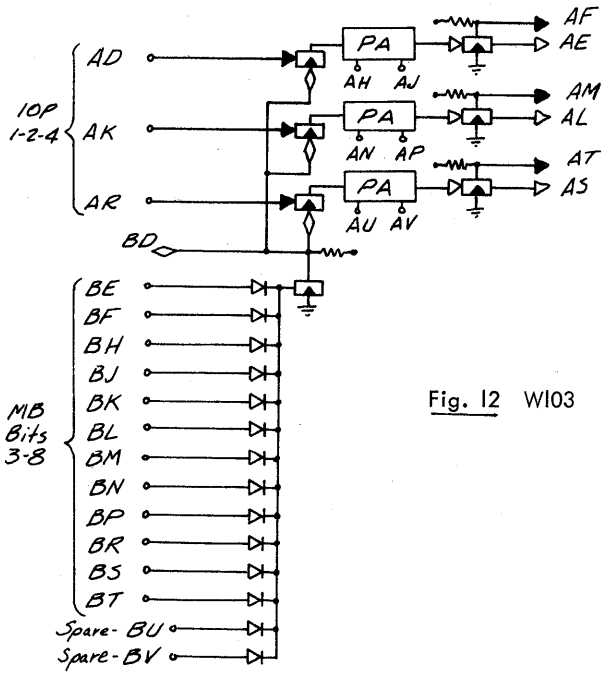


Fig. 12 WI03

Type WI03 Device Selector

The WI03 selects an input/output device according to the code in the instruction word (being held in the Memory Buffer during the IOT cycle.)

The 12 input diodes permit selection of any arbitrary 6-bit code and as used in this card, they decode the number held in MB bits 3 thru 8. When the proper enabling code is presented to the diode gate, the 3 input gates driving the 3 pulse amplifiers are enabled and will permit passage of the programmed IOP pulses. To establish a code on the card, the 6 unnecessary diodes are disabled by snipping one of their leads or removing them altogether. If MB bit 3 should be a ONE to set up the correct code, the diode going to the ZERO side of the MB bit 3 should be disabled. Two spare diodes are included for additional gating flexibility. If unnecessary, they should be disabled or clamped to -3v.

The 3 pulse amplifiers put out R Series type 100ns positive pulses. For those gates which require negative pulses, these are also provided as shown in the illustration.

Jumper terminals are provided on each amplifier for setting the output pulses to a 400ns width. It is recommended that the 400ns pulse be used when transmitting the pulse over distances. The 400ns pulse is also used to clear an R Series type up-counter whose carry gates are permanently enabled..

The positive pulse output of each pulse amplifier is rated at 65 ma of external load at ground, the negative output at 15 ma at ground.

The RI23 Bus Gate

This module provides 6 each 2-input AND gates for negative levels and is useful for reading data into the PDP-8 Accumulator.

Its inputs will accept standard negative levels or 100ns (400ns) negative pulses such as those generated by the WI03 Device Selector. Input load per gate is 1 ma shared among the inputs that are at ground.

Two RI23 modules provide sufficient gating to read one 12-bit word into the Accumulator. If more gates are needed, these gates may be ORed by shorting additional gate collectors together.

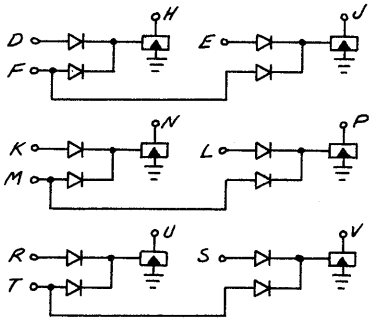


Fig. 13 RI23

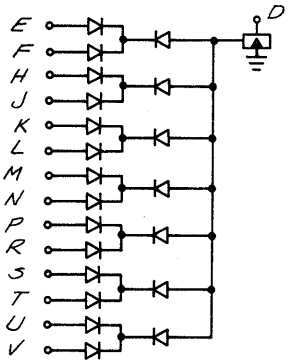


Fig. 14 RI41

Type RI41 Diode Gate

The RI41 AND/NOR gate provides facility for reading any one of 7 bits of information into one bit of the PDP-8 Accumulator. Seven 2-input AND gates are OR'ed together. Twelve RI41 Modules, each driving one bit of the Accumulator, would provide the gating sufficient to load any one of seven 12-bit words.

IOT pulses generated by the WI03, or similar, pulse amplifier may be ANDed with the particular bit status to conditionally set a ONE into that particular bit of the AC. Input load per pair is 1 ma shared by the grounded inputs.

A complete description may be found on page 43 of the Flip Chip Catalog C-105.

DATA BREAK TRANSFERS

The Data Break facility of the PDP-8, in contrast to programmed data transfers, permits control of transfers to be exercised by an external device. Since the external device now controls the computer, in a sense, it follows that more control logic is required in the interface.

In general terms, to initiate a Data Break type transfer of information, the interface control must do the following:

1. Specify the affected address in core memory.
2. Provide the data word by establishing the proper logic levels at the interface (assuming a "data in" transfer).

   OR

   Provide read-in gates and storage for the word (assuming a "data out" transfer).
3. Provide a logical signal to indicate direction of data word transfer.
4. Request a Data Break by supplying proper signal to the Data Break interface.

The address to be affected in core is normally provided in the interface in the form of a 12-bit flip-flop register (Data Break Address Register) which has been preset by the interface control (or sometimes by the computer itself).

Figure 15 shows the Address Register whose outputs drive directly into the Data Break interface. Also shown is the Data Break Request flip flop and the Direction flip flop.

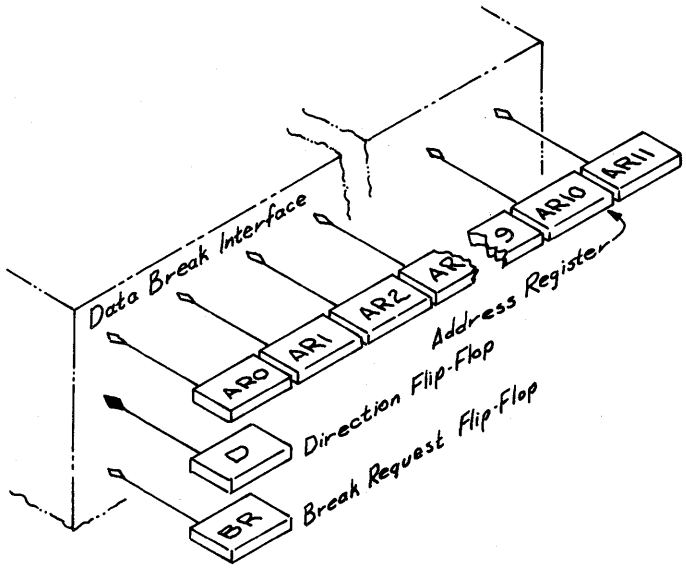


Fig. 15 Data Break Address Register and Control Flip Flops

The Input Buffer in Figure 16 holds the 12-bit word which will be written into that core location specified by the address held at the Address Register shown in Figure 15. Since the assertion level for a One at the interface is 0 volts and logical Zero is -3 volts, the negated or zero outputs of the registers are used to drive the Data Break interface.

Timing for a "data in" transfer is shown in Figure 17. A "data in" transfer would work as shown in the following paragraph.

Other portions of the interface control, not shown here, would establish a word to be written into core in the Data Break Input Buffer and also set the address value in the Data Break Address Register. The direction flip flop also would be set to indicate a "data in" transfer and when these conditions have been established, the Data Break Request flip flop would be set. (All of these conditions may be set with the same pulse but no transitions should occur after setting the Data Break request flip flop).

The computer will then complete the current instruction and acknowledge and perform a Data Break according to the conditions set at its interface. As indicated in the "data in" Timing Diagram, the interface will send back an Address Accepted Pulse which the interface then uses to clear the Data Break Request flip flop. (Notice also that if sequential locations in core are to be used for data storage in a "data in" or "data out" transfer, then the Address Accepted Pulse is useful to increment the value in the Address Register). After servicing the break request in 1.5 μsec, the computer will resume its program uninterrupted until the Data Break Request flip flop is once again set. Notice that the interface requires only that static levels be presented on a "data in" transfer. This minimizes the synchronizing logic necessary for the interface.

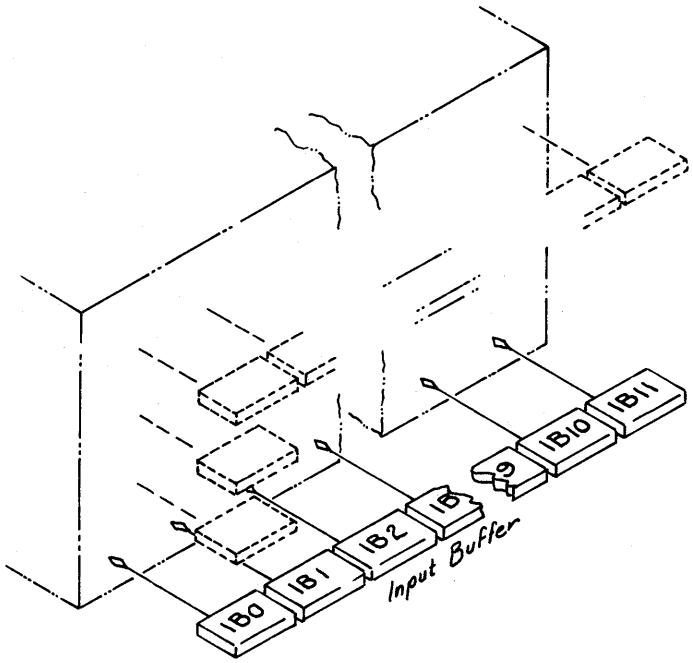


Fig. 16 Data Break Input Buffer

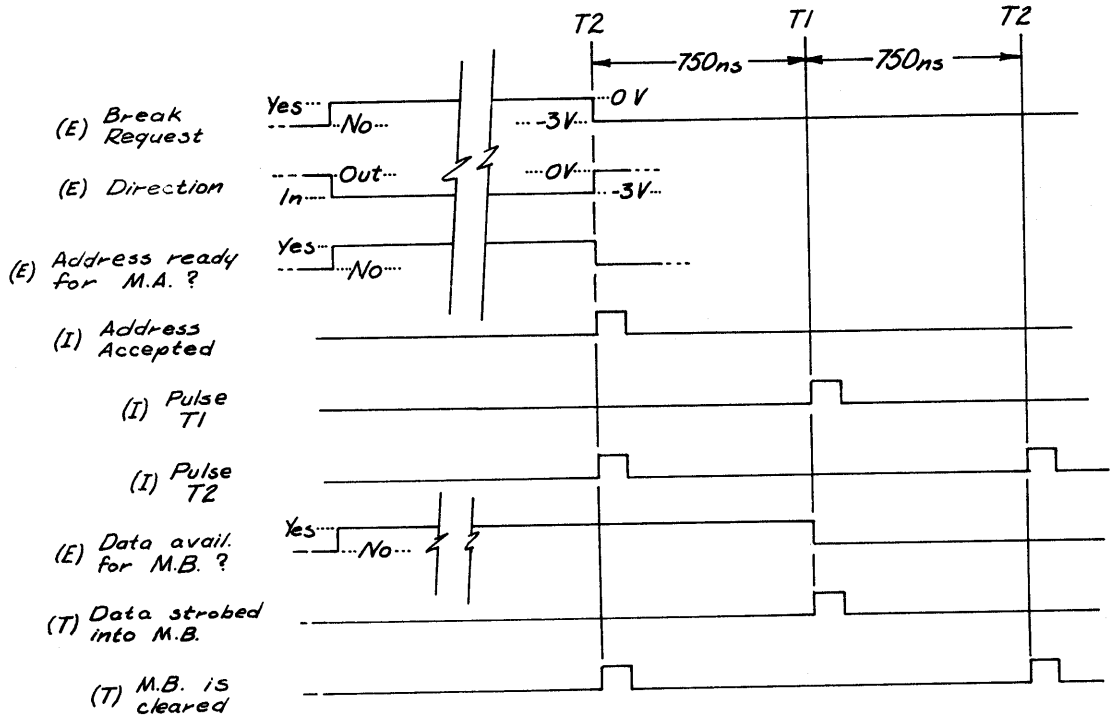


Fig. 17 Data Break Timing-Input

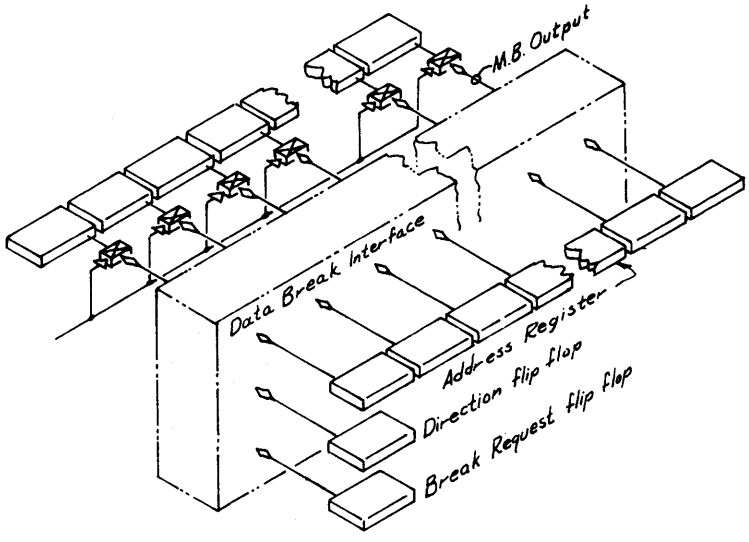


Fig. 18 Data Break Output Buffer

A "data out" transfer is much the same as the "data in" transfer.

The Address Register must be initialized to contain the address in core from which a word will be fetched. An Output Buffer must be provided with read-in gates to accept the word from the computer interface. (Note that buffering of some sort is necessary due to the fact that the information is available at the interface for only approximately 1.1 μsec.) The Direction flip flop must be set to indicate a "data out" transfer and the Data Break Request flip flop likewise must be set. (As before, all of these conditions may be set at the same time with the same clock pulse.)

Figure 18 illustrates the block diagram arrangement for a "data out" transfer.

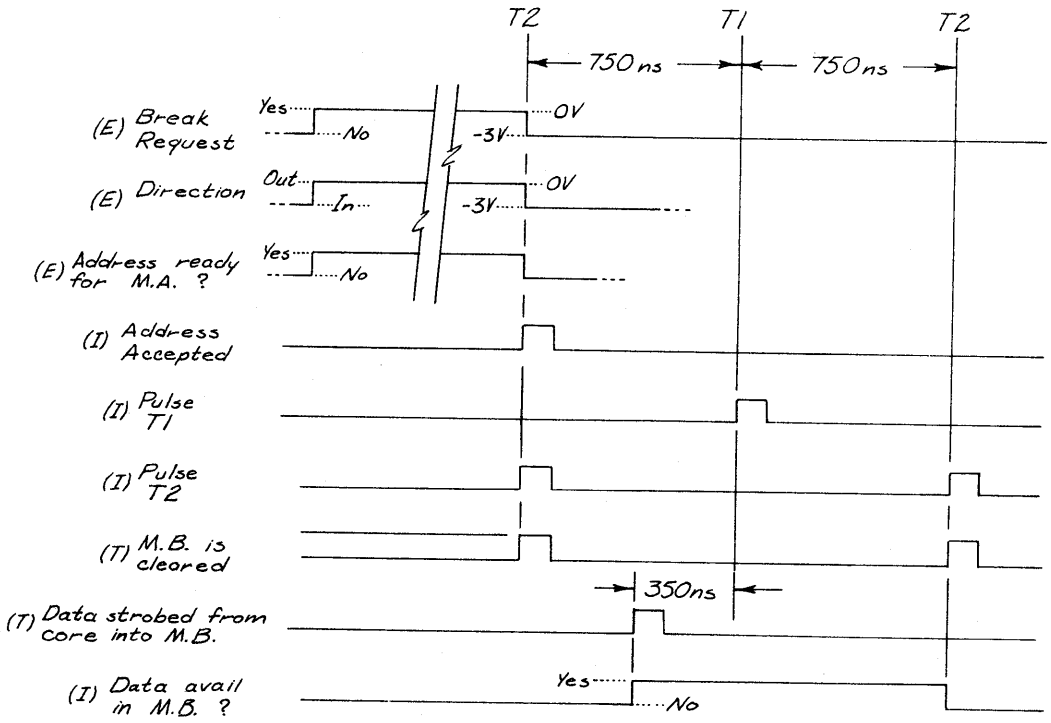


Fig. 19 Data Break Timing-Output

The timing for "data out" transfer is shown in Figure 19. Note that at time T2, the Memory Buffer (MB) is cleared. At approximately 400 ns later, the MB will be set to hold the contents of the specified memory cell. From this time until the next T2 pulse, some 1.100 μsec later, the Memory Buffer may be sampled to set the Output Buffer Register. In contrast to a "data in" transfer, the "data out" interface must provide the strobe or sampling pulse to gate the MB information into the buffer register. The most convenient times to gate the data is at either T1 or T2 within the break cycle, since these pulses are provided at the interface. It is convenient to AND the Break State signal, available at the interface, with T1 through a 2-input diode NAND gate to read the MB data into the Output Buffer. Note this arrangement in Figure 20. (Note that if the data is strobbed into the Output Buffer at T1, it must be gated through diode inverter gates. The conventional DCD gate requires a minimum set-up time of 400 ns and there is no assurance that the DCD gates will be completely set up at T1.)

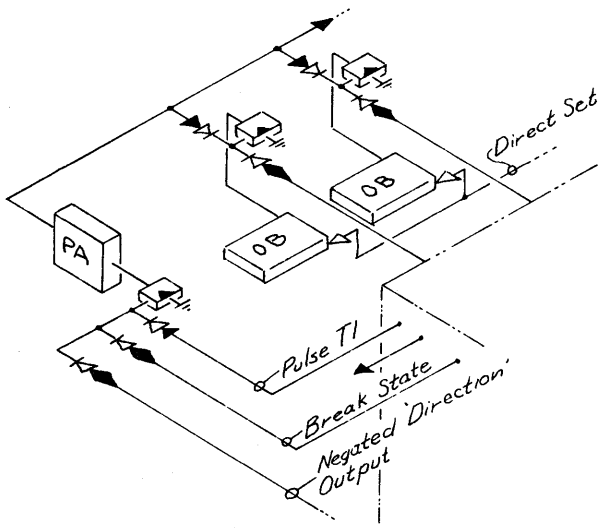


Fig. 20 Data Out Strobe Pulse

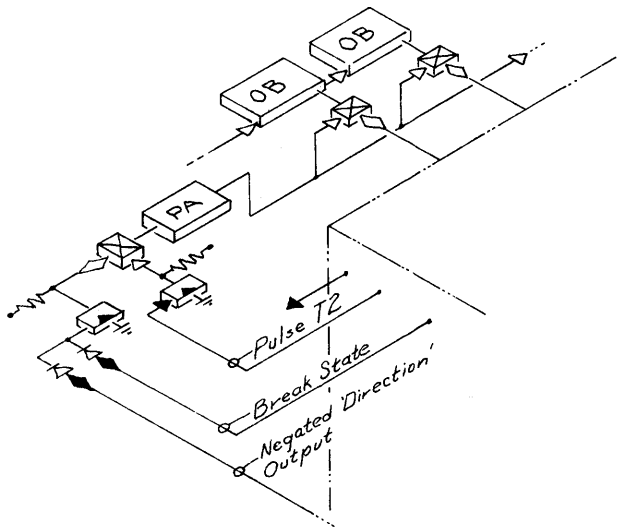


Fig. 21 Data-Out Strobe Pulse

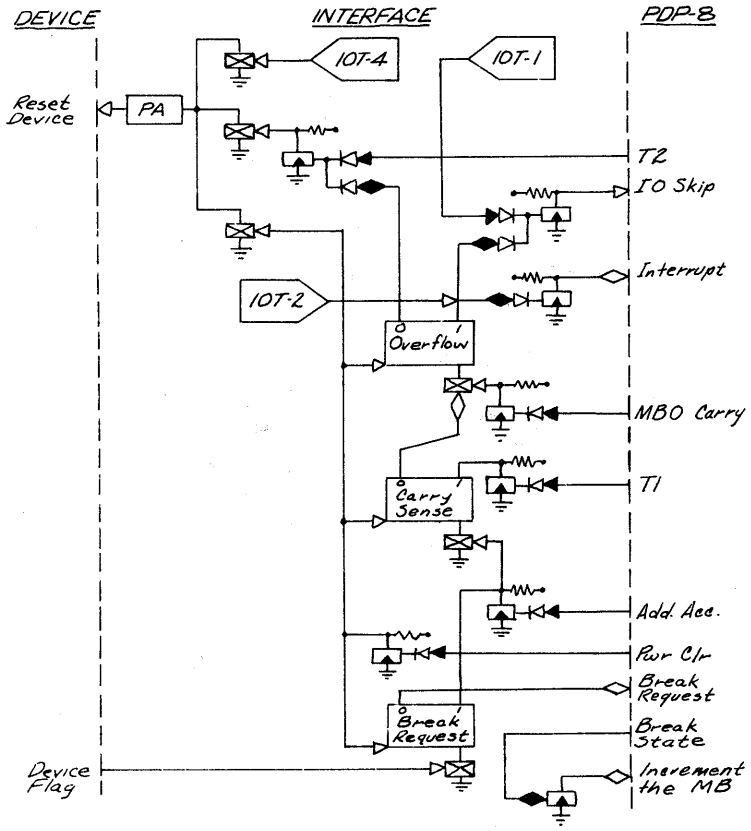
Figure 21 illustrates possibly a more satisfactory pulse for clocking the data from the MB into the Output Buffer. Here the Break State is ANDed with pulse T2 through a DCD gate which in turn triggers a pulse amplifier which clocks the data into the buffer. Timing wise, this is probably more satisfactory than the approach in Figure 20, but it does result in a longer lag (750 ns) from the time the Data Break Request is initiated to the time the data is available.

It should be fairly obvious that by selecting the proper logic arrangement, the Output Buffer Register could double as an Input Buffer and be used to present or receive data depending upon the state of the Direction flip flop. Indeed, in most interfaces, the same register buffers data for transfers in both directions providing a duplex mode is not required.

An additional input available at the Data Break interface is the "Increment the MB" input. By enabling this signal input, the data word in the memory cell specified by the Address Register is fetched and incremented by 1 and returned to the same memory cell within one cycle time of 1.5 μsec. This feature is particularly useful in histogram applications. A pulse height analysis application makes good use of this feature as shown in the following section.

"Increment the MB" request signal should be enabled only during the Break cycle.

Fig. 22 Using the "Increment the MB" Input



Use of "Increment the MB" Input:

It is often useful to increment a location in memory based upon an external number (e.g. a counter contains the number 4257, and you would like to increment location 4257 as a means of recording the event.) This technique permits building a "histogram" of a series of measurements in core memory. It is often used in pulse height analysis work.

The interface has the following properties:

1. It will "add one" to a location, using only one memory cycle.
2. A program interrupt occurs if the "add one" results in overflow of the highest order bit (bit zero).

The device which developes the digital number (such as an Analog to Digital Converter) has the following characteristics:

1. It has a flag (flip flop) which is raised when it is to be read.
2. It has a register which holds the digital number to be read. (This number is the address that will be incremented.)
3. Both flag and register are cleared by a computer command.

CABLING & HARDWARE NOTES

All interface connections to the PDP-8 are made at assigned module receptacle connectors in the left (memory-M) or right (processor-P) mounting frame (door). Horizontal rows of modules within a mounting frame are designated by capital letters from top to bottom. Module receptacles are numbered from left to right as viewed from the wiring side (right to left from the module side). Terminals of a connector or module are assigned capital letters from top to bottom, omitting G, I, O, and Q. Therefore, terminal PE2H is in the right mounting frame (P), the fifth row from the top (E), the second module from the left (2), and the seventh terminal from the top of the connector (H).

The module receptacles assigned for interface connection are: ME34, MF34, ME35, MF35, PE2, PF2, PE3, PF3, PE4, and PF4. Connection to these receptacles is made by means of coaxial cables terminated in DEC Type W011 Signal Cable Connectors.

Interface connection to the PDP-8 can be established for all peripheral equipment by making series cable connections between devices. In this manner only one set of cables is connected to the computer and two sets are connected to each device; one receiving the computer connection from the computer itself or the previous device, and one passing the connection to the next device.

Where physical location of equipment does not make series bus connections feasible, or when cable length becomes excessive, additional interface connectors can be provided at the computer.

All logic signals which pass between the PDP-8 and the input/output equipment are standard DEC levels or standard DEC pulses. Standard levels are either ground potential (0.0 to -0.3 volts), designated by an open diamond (—◇) or are -3 volts (-3.0 to -4.0 volts), designated by a solid diamond (—◆). Standard pulses in the positive direction are designated by an open triangle (—▷) and negative pulses are designated by a solid triangle (—▶). Pulses originating in R series modules are positive-going pulses which start at -3 volts, go to ground for 100 ns (or 400 ns), then return to -3 volts. Pulses originating in W series modules are referenced to ground, are 2.5 volts in amplitude (2.3 to 3.0 volts) with a 2-volt over shoot, and are of 400-ns (or 1000 ns) duration. The Device Selector Module W103 is an exception in that its pulses are similar to those of the R Series Pulse Amplifier Type R601.

Computer input signals that must drive the interface busses to ground (inputs to the AC, CLEAR AC, SKIP, and INTERRUPT REQUEST) must be connected to the base of a grounded-emitter transistor, and so can be considered as transitor-gated negative pulses (—▶▷) or levels (—◆◇).

LOADING AND DRIVING CONSIDERATIONS

All PDP-8 circuits providing output or receiving input interface signals are series R or W FLIP CHIP modules. Therefore the PDP-8 interface is defined entirely in terms of current driving or draining characteristics.

All R series modules are capable of driving currents in the direction from ground to -15 volts, assuming Ben Franklin's definition of current flow. R series modules are not designed to drive loads which are essentially base loads. If such loads are to be driven extra clamped load resistors must be added to make up the necessary differential in current. Therefore, R series loads are defined in terms of milliamperes at ground and zero mils at -3 volts. In general, the output of any R series inverter, including the flip-flop, can drive 20 milliamperes. However, a 2 milliampere load is included within most flip-flops and this current must be deducted from the available driving capability.

Inputs are also defined in terms of milliamperes. Level inputs to level gates are defined as 1 milliampere at ground. Level inputs to DCD gates are 2 milliamperes at ground, and pulse inputs to DCD gates are 3 milliamperes at ground. Where long lines are being driven extra clamped loads should be added to sufficiently discharge the cable capacitance. A good rule is to add an extra 2 mils of clamped load current for every foot of wire beyond 1-1/2 feet.

An exception to the preceding rules can be observed for the Bus Driver R650. This module is designed to drive coaxial cable of 100-ohms characteristic impedance through a series driving resistor. If coaxial cable is not used the direct input may be used provided the lines are short. If reflections occur on the line the resistive output of the bus driver may be used to correct the problem. Shunt termination is not recommended on the far end of the transmission line.

The R650 Bus Driver has two types of outputs, the fast and the slow (ramp) output. Using the fast output the bus driver operates merely as a fast amplifier. In using the ramp output an integrating capacitor is inserted between the input of the bus driver and the output stage, causing the output lines to move from ground to -3 volts or in the reverse direction in approximately 500-700 nanoseconds. This connection is used on the AC lines and is desirable to reduce crosstalk between the lines. All other R650 outputs are fast.

The W640 Pulse Amplifiers should be carefully terminated. If sufficient noise is generated at the output of the W640 it may cause the pulse amplifier to regenerate, hence it is also recommended that output lines of W640 modules be well shielded. The outputs of W640 modules may be either 400 nanoseconds or 1 microsecond in width. All connections on the standard PDP-8 use the 400 nanosecond pulse width.

Input signals to the PDP-8 are in many cases a clamped load resistor of 10 mils and a direct input to a flip-flop or pulse amplifier. The input load is therefore, 10 mils for the clamped load and 1 mil for the flip-flop or the pulse amplifier. Careful consideration should be given to capacitive loads on these lines since they must be a -3 volts before the machine tries to use the pulse amplifier or flip-flop for the next machine cycle. There are some exceptions to this statement. First, the data bits into the MB lines require two milliamperes at ground and no current at -3 volts. The transfer direction signal requires 1 mil at ground. Break request has a ten mil clamped load plus 2 mils for the internal circuitry or 12 mils at ground. The output of all DEC inverters in the system module series can drive 15 mils. In general, 10 mils are absorbed by the clamped load which is normally used. However, on the input signal interface no clamped load is used; hence, the above numbers may be used.

Timing is, in general, determined by the PDP-8 itself. However, a few blanket statements can be made about modules. First the R111 Diode Gate sets up in approximately 50 nanoseconds in either direction under normal load conditions. Fall times are faster with heavier loads, and indeed the best method to speed up an R111 Diode Gate is to connect an external load across the input to ground. The DCD gates set up in 400 nanoseconds is measured from the end of the preceding 100-nanosecond pulse. Both the level and pulse must return to -3 volts for 400 nanoseconds before the next pulse is applied. Series R pulses are 100 nanoseconds in width, measured from the 10% point of the leading edge to the 90% point of the trailing edge. Fall time is not critical on these pulses provided the pulse has returned to -3 volts in time to come up for the next pulse.

SYSTEM MODULE INTERFACING (INCLUDING B SERIES)

The following definitions and rules serve as a useful guide in determining the driving capability of output signals and the load presented to input signals by series B FLIP CHIP modules and all system modules as used in peripheral equipment connected to the PDP-8.

Base Load

Base load is the current which must be drawn from the base of a dc inverter to keep it saturated. In this condition the inverter circuit input terminal is at -3 volts, the emitter is at ground, and a nominal 1 milliampere of current flows through the 3000-ohm base resistor from ground. A 1500-ohm load resistor clamped at -3 volts can nominally accept 8 milliamperes, but tolerance considerations limit this number to 7 milliamperes. Thus, an inverter collector with a 1500-ohm clamped load can drive a maximum of 7 base loads.

Pulse Load

Pulse load is the load presented to the output of a pulse source by an inverter base in the same speed series, or by the direct set or clear input of a flip-flop. Pulse amplifiers are usually limited to driving 16 pulse loads. This number should be decreased if the bases are widely separated physically, and can be increased to 18 if the bases are all physically close together. The series inductance and shunt capacity of connecting wires can make pulses at the end of a series of bases either large or small. Consequently, when driving nearly the maximum number of bases, the pulse amplitude should be carefully checked after installation. A Terminating resistor in the 100-to-300 ohm range is desirable to reduce ringing on a heavily loaded pulse line. The loading on a pulse source is approximately the same when driving a base as a direct input to a flip-flop. One pulse source, of course, cannot drive both direct inputs of flip-flops and inverter bases because the direct inputs require DEC standard positive pulses and base inputs require DEC standard negative pulses. A pulse load is largely determined by the value of the speed-up capacitor connected in parallel with the 3000-ohm base resistor. In the 4000 Series 500 kc modules this capacitor is 680 pf; in 1000 Series 5 mc modules it is 82 pf; and in 6000 Series 10 kc modules it is 56 pf.

Pulsed Emitter Load

Pulsed emitter load is the load applied to the collector of an inverter which drives the pulse input to a flip-flop, pulse amplifier, or delay. The pulse current passes through all of the inverters in series with the pulse input, and it should be assumed to be the load on each of the series inverters.

DC Emitter Load

The load applied to the collector of an inverter driving a clamped load resistor is the dc emitter load. This load is also presented by the collector of an inverter which drives an emitter in an inverter network terminated by a clamped load resistor. Under these conditions, the collector of an inverter driving an emitter in a transistor gating network must also supply the base current leaving the succeeding inverters which are saturated. This current is small, but in complex networks it must be considered. An inverter in the DEC 1000 or 6000 Series modules can supply 15 milliamperes, and in the 4000 Series modules can supply 20 milliamperes.

An inverter network can always be analyzed by assuming:

a. that a short circuit exists between the emitter and collector when -3 volts is applied to the base.

b. that 1 milliampere of base current will flow if either the collector or emitter is held at ground potential.

c. that the maximum dc collector current through an inverter is 20 milliamperes for 4000 Series 500-kc modules and is 15 milliamperes for all other DEC series modules.

A capacitor-diode gate level input (4000 Series) does not present any dc load. A transient load occurs when the input level changes. Note that all capacitor-diode gates require that the level input precede the initiating pulse input by at least 1 microsecond.

DIODE GATE

The basic elements of digital logic used in the PDP-8 are the diode gate and diode-capacitor-diode gate. The diode gate is used in the R series to combine, amplify, invert and standardize the signals which represent various logic functions. Figure 23 is a schematic of a simple diode gate with one input.

When the input is negative, the node point is also negative and current flows from the transistor emitter through the biasing diodes and the biasing resistor to -15v. As a result, the PNP transistor is turned on forming a short circuit between the collector and the emitter. Thus, when the input voltage is negative, the output voltage is ground. Since the output is from a saturated transistor, it has a low output impedance and good driving power.

When the diode gate input voltage is ground, the biasing diodes and the resistor, which is connected to the +10v supply, hold the transistor base more positive than the emitter, and the transistor is turned off. The output is then an open circuit, and it will follow the voltage of any other circuit connected to it.

If the load resistor and clamp diode are attached to the transistor collector, they serve as a voltage source and hold the output at -3v while the transistor is off. When the transistor is on, the diode is cut off and the load resistor follows the output to ground.

The single-input diode gate therefore has three functions. First, it inverts the input signal. Second, it standardizes the output voltage to -3v or ground (if the clamped load diode and resistor are connected). Third, since the output current available from the transistor is much greater than the required input current, the diode gate amplifies.

A fourth function, gating, is obtained by adding more diode inputs to the node point, as shown in Figure 24.

The node terminal will be at approximately the same voltage as the most positive input. Thus, when any input terminal is grounded, the node terminal is also at ground and the circuit output is at -3v. If all of the inputs are negative, the node terminal will be negative and the circuit output will be at ground.

Gating functions may also be performed by wiring together two or more diode gate outputs and one load resistor, as shown in Figure 25. When any input is negative, it saturates the corresponding transistor and forces the output line to ground. If all inputs are at ground, all of the transistors are open circuits and the output voltage, determined by the clamped load resistor, is -3v.

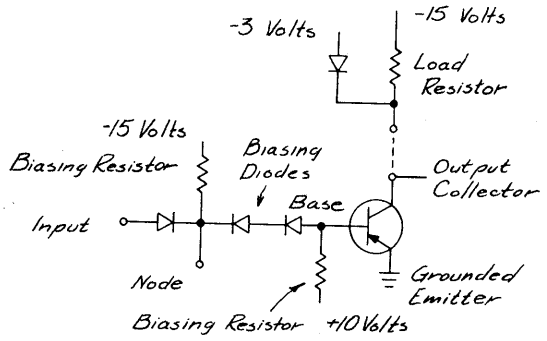


Fig. 23 Single Input Diode Gate

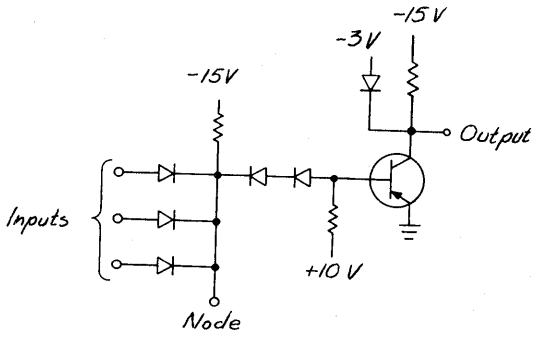


Fig. 24 Multiple Input Diode Gate

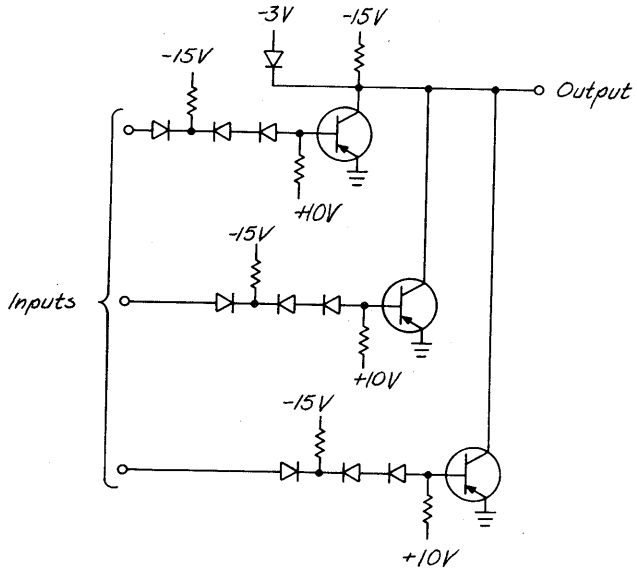


Fig. 25 Diode Gates in Parallel

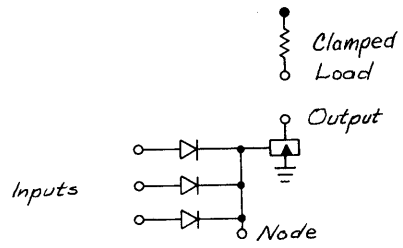


Fig. 26 Diode Gate Symbol

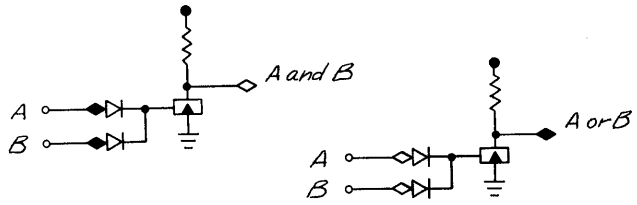


Fig. 27 Diamonds Indicate Voltage Levels

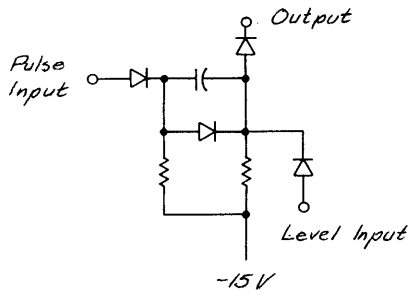


Fig. 28 Diode-Capacitor-Diode Gate Circuit

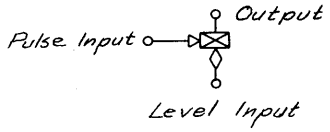


Fig. 29 Diode-Capacitor-Diode Gate Symbol

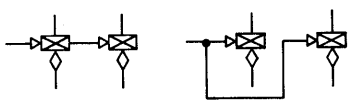


Fig. 30 Pulse Lines to Multiple Gates

The basic diode gate can be used to construct very complex logical functions. A drawing that showed all of the circuit components, however, would be tedious to draw and difficult to read. For this reason, logic diagrams use a shorthand notation which represents one or more components as a single functional unit. Referring to Figure 26, diodes are shown in the conventional way. The transistor circuit, including the biasing resistors and diodes, is shown as a simple rectangle with an arrowhead indicating the direction of the transistor emitter. This part of the circuit is called an inverter because of the function it performs. The load resistor is shown as a resistor with a large dot at the top indicating that it is diode clamped to -3v.

Assertion input and output voltage levels are shown by diamonds. A solid diamond indicates a -3v level, and an open diamond indicates a ground level. In the 2-input diode gate of Figure 27, for example, if input A and input B are both negative, the output will be at ground. If either A or B is at ground, the output will be negative.

DIODE-CAPACITOR-DIODE GATE

The diode-capacitor-diode (DCD) gate is used to standardize the input to various units such as flip-flops, delays, and pulse amplifiers. It provides logical isolation between pulse and level inputs and produces a logical delay which is essential for sampling flip-flops at the same time they are being changed. It also acts as a logical AND gate since both pulse and level inputs must meet certain requirements for a signal to appear at the output. Either positive pulses or positive-going level changes (both -3v to ground) may be used as the pulse input.

A schematic drawing of a DCD gate is shown in Figure 28. If the level input is held at ground and the pulse input is held at -3v, the capacitor will become charged after the set-up time has passed. If the pulse input then suddenly goes to ground, a positive-going pulse will appear at the output. There is delay at the level input, but the pulse input goes to the output without delay. Even if the level input changes simultaneously with a positive transition at the pulse input, the delay acts as a temporary memory: the pulse input is gated according to the level input that existed during the interval before the pulse.

The symbol for the DCD gate (Figure 29) is distinguished from the diode gate by an X in the rectangle. The output is at the top, the delayed (level) input is at the bottom, and the differentiating (level change or pulse) input is on the side. The input signal to be differentiated, whether a level change or a pulse, is indicated by an arrowhead, rather than a diamond. The pulse symbols are hollow when positive-going and solid when negative-going. In the DCD gate, the pulse input must be positive-going.

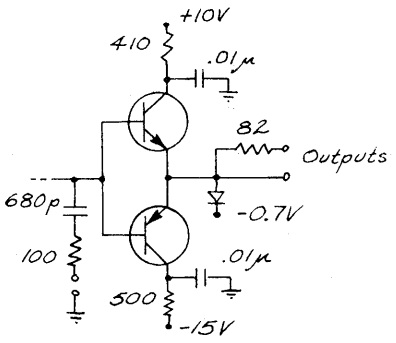
Since many DCD gates may be driven by the same pulse, the side of the rectangle opposite the pulse may be used to show a continuation of the same line, as in Figure 30. The illustration on the left below is a simplified version of the identical logical configuration on the right.

For a complete description of R Series logic used in PDP-8, please refer to DEC Flip Chip Catalog C-105, available from any DEC engineering office.

CIRCUITS

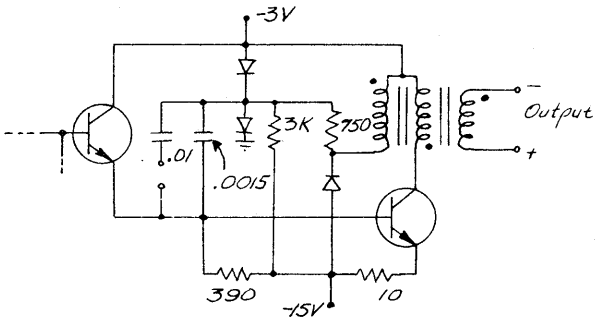
R650

DC Bus Driver with typical rise and fall time of 25ns. With slow-up capacitor shunted to ground output rise and fall time typically 750 ns. Resistor output provided for driving coaxial cable.



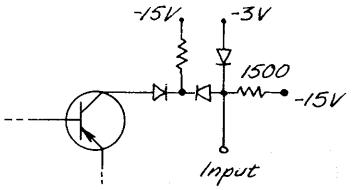
W640

Standardizing pulse amplifier for outputting either 400ns or 1000ns DEC pulses. Polarity of pulse will be opposite to polarity of output terminal grounded (pulse transformer output).



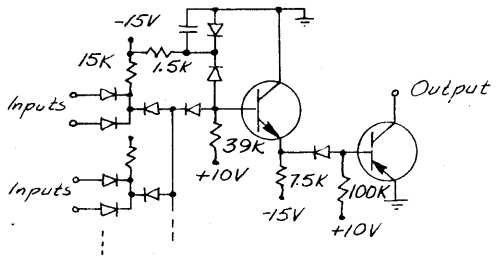
R210

Input mixer to Accumulator flip-flop. Pull the input to ground with collector of a pulsed inverter.



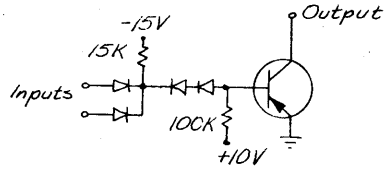
R141

Seven AND gates NORed together. Accepts DEC negative levels or DEC pulses of 70ns duration or longer.

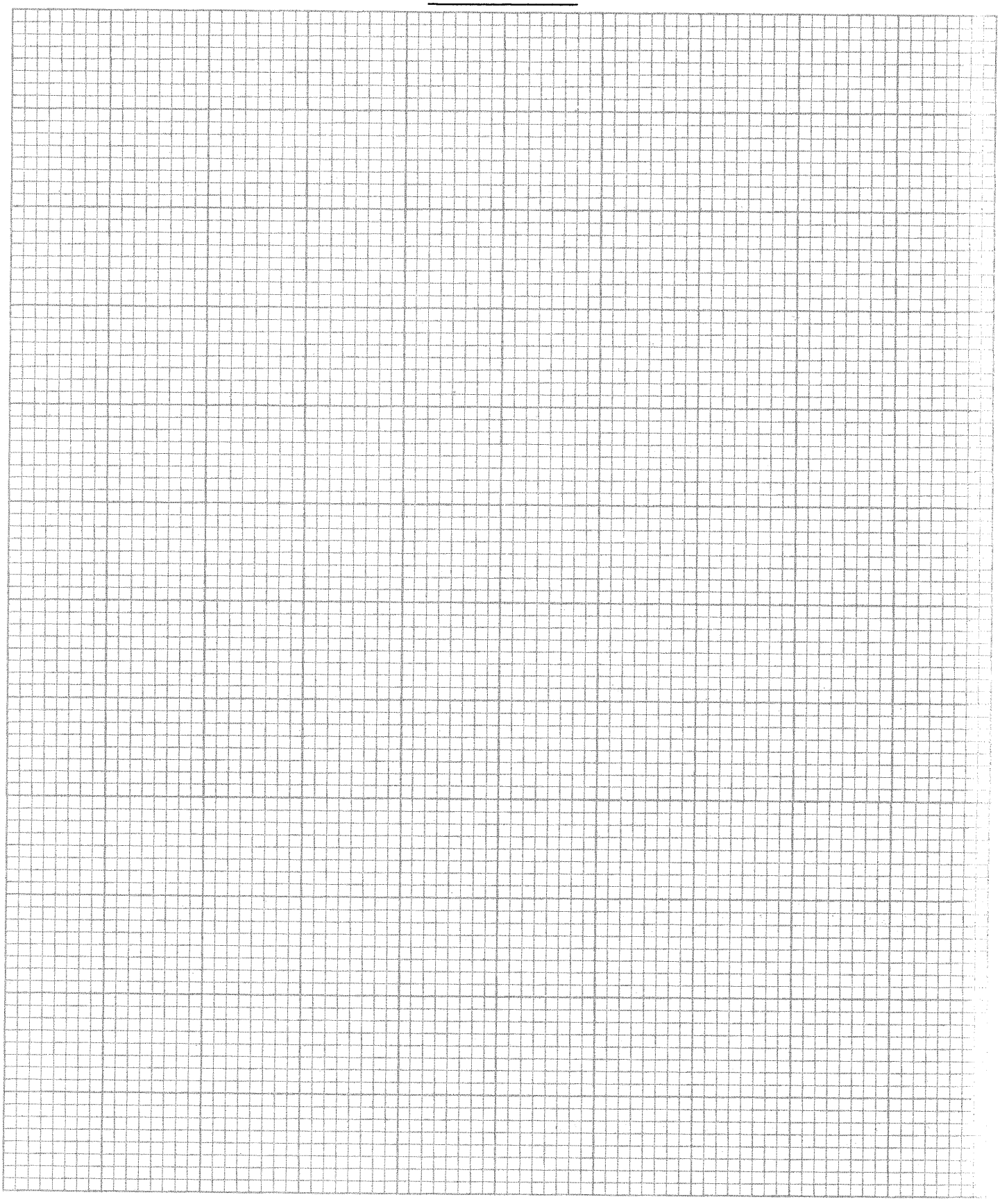


R123

Six 2-input NAND gates. Accepts DEC negative levels or DEC pulses of 70ns duration or longer.



INTERFACE SKETCH



PDP-8 INPUT SIGNAL INTERFACE

Signal	Symbol	Interface Connection	Signal Destination: Terminal	Signal Destination: Module Type
AC 0		PE2D	PA7E	R2I0
AC I		PE2E	PA8E	R2I0
AC 2		PE2H	PA9E	R2I0
AC 3		PE2K	PAI0E	R2I0
AC 4		PE2M	PAIIE	R2I0
AC 5		PE2P	PAI2E	R2I0
AC 6		PE2S	PAI3E	R2I0
AC 7		PE2T	PAI4E	R2I0
AC 8		PE2V	PAI5E	R2I0
AC 9		PF2D	PAI6E	R2I0
AC I0		PF2E	PAI7E	R2I0
AC II		PF2H	PAI8E	R2I0
CLEAR AC		PF2P	PAI9J	R603
INTERRUPT REQUEST		PF2M	PD36K	RIII
SKIP		PF2K	PD33V	R602
DATA ADD 0(I)		PE3D	PC7R	R2II
DATA ADD I(I)		PE3E	PC8R	R2II
DATA ADD 2(I)		PE3H	PC9R	R2II
DATA ADD 3(I)		PE3K	PCI0R	R2II
DATA ADD 4(I)		PE3M	PCIIR	R2II
DATA ADD 5(I)		PE3P	PCI2R	R2II
DATA ADD 6(I)		PE3S	PCI3R	R2II
DATA ADD 7(I)		PE3T	PCI4R	R2II
DATA ADD 8(I)		PE3V	PCI5R	R2II
DATA ADD 9(I)		PF3D	PCI6R	R2II
DATA ADD I0(I)		PF3E	PCI7R	R2II
DATA ADD II (I)		PF3H	PCI8R	R2II

Signal	Symbol	Interface Connection	Signal Destination: Terminal	Signal Destination: Module Type
DATA BIT 0(I)		PE4D	PD7M	R2II
DATA BIT I(I)		PE4E	PD8M	R2II
DATA BIT 2(I)		PE4H	PD9M	R2II
DATA BIT 3(I)		PE4K	PDI0M	R2II
DATA BIT 4(I)		PE4M	PDIIM	R2II
DATA BIT 5(I)		PE4P	PDI2M	R2II
DATA BIT 6(I)		PE4S	PDI3M	R2II
DATA BIT 7(I)		PE4T	PDI4M	R2II
DATA BIT 8(I)		PE4V	PDI5M	R2II
DATA BIT 9(I)		PF4D	PDI6M	R2II
DATA BIT I0 (I)		PF4E	PDI7M	R2II
DATA BIT II (I)		PF4H	PDI8M	R2II
TRANSFER DIRECTION	*	PF3M	PD23E	RIII
BREAK REQUEST		PF3K	PC32J	R203
†INC MB		PF3T	PD3IM	RI07

*Direction is into PDP-8 when signal is -3 volts, out of PDP-8 when ground potential.

†INC MB should be enabled only during the BREAK cycle.

PDP-8 OUTPUT SIGNAL INTERFACE

Signal	Symbol	Interface Connection	Signal Origin: Terminal	Signal Origin: Module Type
BAC 0(I)	◇	ME34D	ME26J	R650
BAC 1(I)	↓	ME34E	ME26T	R650
BAC 2(I)	↓	ME34H	ME27J	R650
BAC 3(I)	↓	ME34K	ME27T	R650
BAC 4(I)	↓	ME34M	ME28J	R650
BAC 5(I)	↓	ME34P	ME28T	R650
BAC 6(I)	↓	ME34S	MF26J	R650
BAC 7(I)	↓	ME34T	MF26T	R650
BAC 8(I)	↓	ME34V	MF27J	R650
BAC 9(I)	↓	MF34D	MF27T	R650
BAC 10(I)	↓	MF34E	MF28J	R650
BAC 11(I)	↓	MF34H	MF28T	R650
IOP 1	→	MF34K	MC31H	W640
IOP 2	↓	MF34M	MC31N	W640
IOP 4	↓	MF34P	MC31U	W640
B POWER CLEAR	↓	MF34V	MD30N	W640
BTI	↓	MF34S	MD30H	W640
BT2A	↓	MF34T	MD30U	W640
ADDRESS ACCEPTED	↓	PF3S	PF10H	W640
B RUN (I)	◆	PF2S	PE8J	R650
B (BREAK)	↓	PF3P	PE8T	R650

Signal	Symbol	Interface Connection	Signal Origin: Terminal	Signal Origin: Module Type
BMB 0(I)	◇	ME35D	MC26J	R650
BMB 1(I)	↓	ME35E	MC26T	R650
BMB 2(I)	↓	ME35H	MC27J	R650
BMB 3(0)	↓	ME35K	MC27T	R650
BMB 3(I)	↓	ME35M	MC28J	R650
BMB 4(0)	↓	ME35P	MC28T	R650
BMB 4(I)	↓	ME35S	MC29J	R650
BMB 5(0)	↓	ME35T	MC29T	R650
BMB 5 (I)	↓	ME35V	MD25J	R650
BMB 6(0)	↓	MF35D	MD25T	R650
BMB 6(I)	↓	MF35E	MD26J	R650
BMB 7(0)	↓	MF35H	MD26T	R650
BMB 7(I)	↓	MF35K	MD27J	R650
BMB 8(0)	↓	MF35M	MD27T	R650
BMB 8(I)	↓	MF35P	MD28J	R650
BMB 9(I)	↓	MF35S	MD28T	R650
BMB 10(I)	↓	MF35T	MD29J	R650
BMB 11(I)	↓	MF35V	MD29T	R650

Note: "B" Prefix means the signal is buffered either by an R650 Bus Driver (for levels) or by a W640 Pulse Amplifier (for pulses).