Fermilab

TM-772
0811.000

A

COMPUTER INTERFACE

Z80 S-100 BUS

to

PDP-11 UNIBUS

BY

Philip J. Lucas

February 27, 1978

TABLE OF CONTENTS

A Z80 S100 Bus to PDP-11 UNIBUS BIDIRECTIONAL INTERFACE

ABSTRACT

This paper describes a hardware module that allows the user to transfer data between the PDP-11 computer and a Z80 microprocessor. Any DEC CPU that uses the DEC Unibus is acceptable for the eleven side and any device that uses an S-100 bus is acceptable for the micro processor side.

I. INTRODUCTION

The availability of the DEC disk pac plus the need for mass storage for a Z80 microprocessor based data gathering system, led to the invention of an interface between the two. The system block diagram is shown in Figure 1. The operating system used on the DEC machine was RT-11 and was run on PDP-11/40. For all data transfers the DEC computer was chosen as master and Z80 CPU as slave. The interface can be configured to operate in two modes as far as the Z80 is concerned. The interface can communicate with the Z80 in an interrupt controlled mode or the Z80 can be made to wait until a transfer is completed. I personally prefer the interrupt mode since the Z80 handles this quite naturally and you can do input and output transfers simultaneously. That is you can send a byte and while the eleven is taking it you can read in data already set up by the eleven, and vice versa.

The interface has two eight bit buffers, one for input and one for output. An eight bit byte transfer was chosen, rather than a sixteen bit word, because the files originated as bytes and I preferred a simplistic approach because this was faster based on the cycle time of the Z80. What I mean here is that in order to transfer a word of sixteen bits several additional software instructions would be required in order to set it up, on the Z80 side. The Z80 initiates all transfers and manages all its memory allocations. The eleven manages its own memory also.

II. HARDWARE

A buffer module designed to plug directly into a DEC mainframe was used. It is an "MDB1710" made to fit the DECBB-11, DD-11A or DD11-B mounting panels. See Figure 2 for the block diagram of the MBD-1710. What the MBD-1710 can do is provide the interrupt protocol along with interrupt vectors and a device address. A system description is as follows: A byte is made available to the MBD-1710. Let's call this word "Word A". The user after making the word A available notifies Devint 1. Devint 1 will generate an interrupt to the PDP-11 using Vector 1. The eleven will now service the interrupt, by reading in "word A". Upon completion of the interrupt service routine the PDP-11 will remove the busy signal. This then completes a one byte transfer. There are two interrupts available as well as two vectors. The vectors are user strapped. Vector 1 is for input to the eleven and Vector 2 is for outputs from the eleven.

The hardware used to supply the 1710 is described below using the diagram in Figure 3. A standard Zilog Z80-PIO is the heart of the interface. A switch set decoder selects the PIO using any address from 00 thru FC. The least two significant address lines are used to control selection of the A/B port and the control of the C/D line. The A port is configured as an output port. The B port is configured as an input port. The A port handles all transfers to the eleven and the B port handles all transfers from the eleven. The A port will be discussed first. A byte that is to be transferred is written into the A port. When the A port output data lines become valid the A ready line goes high setting the D flip-flop. This flip-flop signals Devint 1 that there is a pending transfer. Also, if selected by closing switch 7 the wait line to the Z80 is activated. The eleven will service the device causing a strobe to go active. When service is completed the strobe goes to its inactive state allowing the wait line to go inactive. Now the Z80 can transfer another byte or fall thru.

The B port is very similar to A. The Z80 will initiate a file transfer by doing a dummy read of port B. This will activate the B ready line signaling an interrupt to the eleven. The eleven will respond, using Vector 2, by placing a byte into port B. Strobe B will strobe the data into the B port. The B ready line will hold the Z80 in a wait if it is desired by the User closing Switch section #7. Upon the rising edge of B strobe, B ready will go low releasing the Z80. Now the Z80 may repeat the process or fall thru.

If you use the PIO interrupt response you should open switch #7 and the A & B ports will interrupt when they require service, if programmed to do so. The tri-state buffer on the input side of the PIO is there to make this interface compatible with S-100 bus. The S-100 bus has two data buses. A "Bus In" and a "Bus out". The buffer is there to distinguish between the two. The inverting tri-state buffer on the PIO output side does two things. It inverts the data since the the PDP-11 is asserted low. I wanted to do this since the MBD 1710 automatically inverts coming back and it would be easier to find the data in the eleven. The second thing the buffer does is to hold off the data until the eleven is ready for it. Transfers from the eleven do not have the problem.

III. SOFTWARE

The following routines are given only as a means of a system checkout. The hardware places no restrictions on software which makes the interface flexible.

FOR Z80 OUTPUT USING WAIT

ADDRESS	OP-CODE	MNEMONIC	COMMENT
0000	3E 0F	LDA, #0F	:Select Port A as an output.
0002	D3 02	OUT (N), A	:Your an output.
0004	06 FF	LDB, #FF	:Some output.
0006	0E 00	LDC, #00	:Port A data adress.
0008	ED 41	OUT(C), B	:Output B to port.
000A	10 FC	DJNZ	:Loop until B = 0
000C	76	HALT	:Stop

FOR PDP-11 INPUT USING WAIT

ADDRESS	OP-CODE	MNEMONIC	COMMENT
170_8	00 1000	Data	:Address of INT routine
172_8	00 0300	Data	:Processor Status WD
1000	01 2700	MOV #300, RO	:Program start area
1002	00 3000		:Storage Area
00 1004	012737	MOV#1012,@#1000	:New Starting Address
00 1006	001012	Data	
00 1010	001000	Data	
00 1012	013730	MOV@#764016, (RO)+	:Store File
00 1014	764016		
00 1016	000002	RTI	:Return

What the above software does is to create a file in the Z80. The file is 256 bytes of descending value starting with FF_{hex}. The file after the transfer will appear starting at location 3000_8 in the eleven.

CONCLUSION

This interface should not be confused with a direct memory access. Originally DMA was thought of, but the hardware requirement was non trivial and I wanted to be able to go on line with this thing in just several weeks. This particular device if used in the interrupt mode can accomplish data transfers in background rather efficiently.

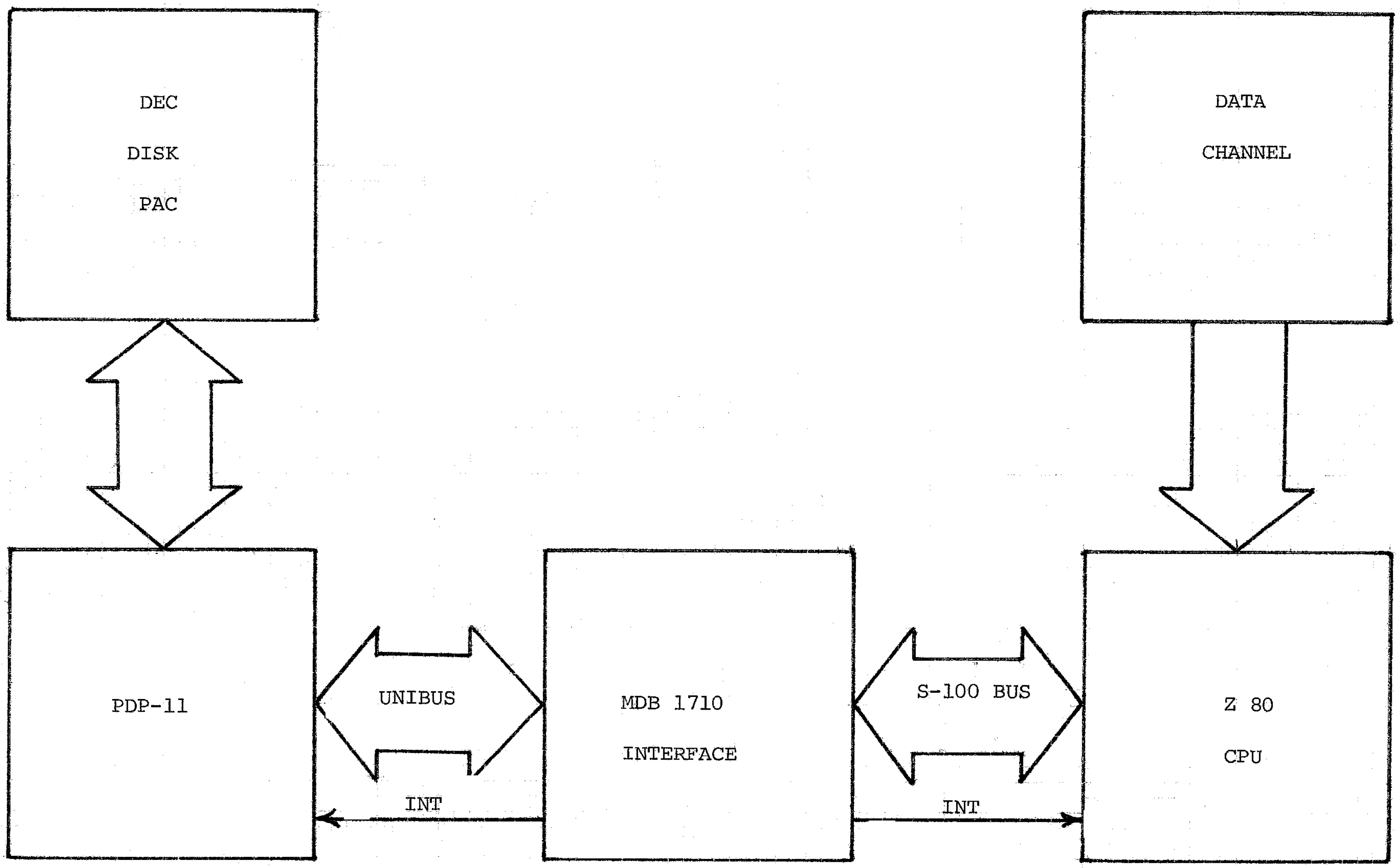
Each device, that is, the mocroprocessor and the DEC computer, manages its own memory. The Z80 side will manipulate and move files around and eleven need not know where the file is coming from (DMA cannot do this). The converse is also true. The interface originates interrupts to both the eleven side and the Z80 side. In theory both CPU's could be interrupted at the same time.

I found the S-100 bus somewhat mysterious at times. For example, the bus is supposed to represent one TTL load. I used an LSTTL because the S-100 criteria states that inputs should be no greater than one LSTTL input load. I found difficulty in pulling down the wait line with an LS05. I had to replace it with an 05. This led to a fan out problem because the Z80 PIO will only sink one TTL load. I think one would be safe in leaving off pull-ups when driving the S-100 bus to circumvent this problem, i.e., use the LS05 without the pull up Figure 3.

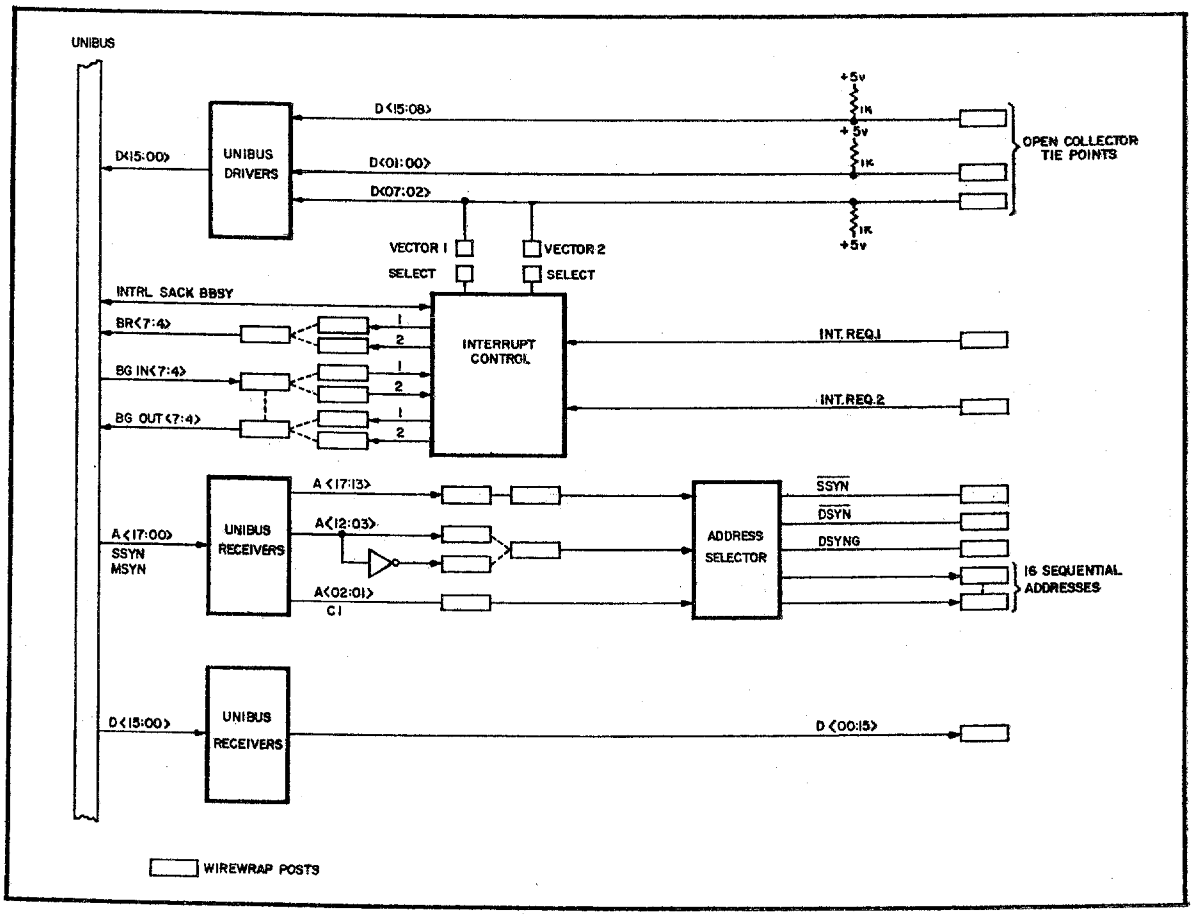
You will notice in Figure 3 that there is no reset line common to all devices. This is because the PIO resets internally with power on and if the state of the 74's become active on power on the eleven sequence will reset them.

The switch section #7 of course can be replaced by some sort of programmable device so one need not "fat finger" in this option. For this project doing that was optional equipment and has to be ordered special from the factory. (The first Model T did not have air conditioning).

FIGURE 1

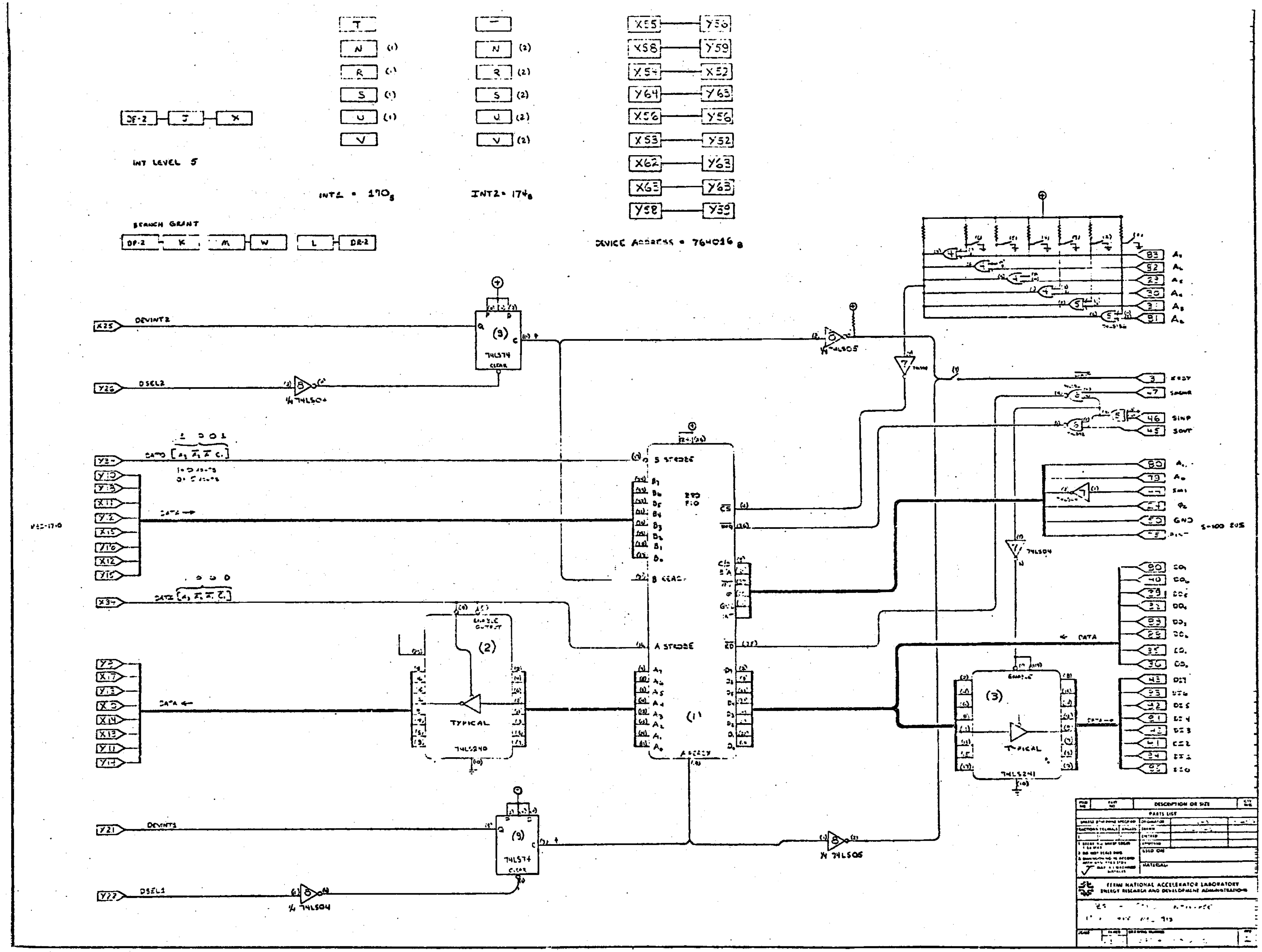


TM-772
0841.000



MDB-1710 BLOCK DIAGRAM

FIGURE 2



INT LEVEL 5
INT2 = 170₈
INT2 = 174₈
BRANCH GRANT
DEVICE ADDRESS = 764016₈
DEVINT2
DSEL2
74LS74
¼ 74LS04
¼ 74LS05
DATA
DEVINT1
DSEL1
74LS240
74LS241
TYPICAL
A STROBE
B READY
B STROBE
A READY
Z80 PIO
S-100 BUS
GND
SINP
SOUT
FERMI NATIONAL ACCELERATOR LABORATORY
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
PARTS LIST
DESCRIPTION OR SIZE
MATERIAL