Grant Saviers

4-0030-9

digital INTEROFFICE MEMORANDUM

DATE: 18 April 1969

SUBJECT: PDP-11 I/O Bus

Jerry

TO: R. Cady
H. McFarland
J. O'Loughlin
B. Young
P. Janson

FROM: Jerry Butler

CC: PDP-11 Design Review Committee

Ref: ES:0002 2 April 1969

I have read your paper on I/O Bus rather thoroughly. I am not certain I understand all implications as well as I should, however, I would like to throw out a few comments based on my understandings to date. Before doing so I should state some assumptions: (MINE).

1. The new PDP-11 "Family" will go from small ($<$ 8I) to large ($\geq$ 10) configurations some day.

2. Planning for the larger configurations <u>now</u> is is a necessity in order to guarantee getting there from here.

3. I assume the bus sequences behave approximately as I have shown in attached flow charts.

4. In both small and large machines high bus transfer rates, long bus lengths, and low latencies for I/O are characteristics necessary to a good portion of our users.

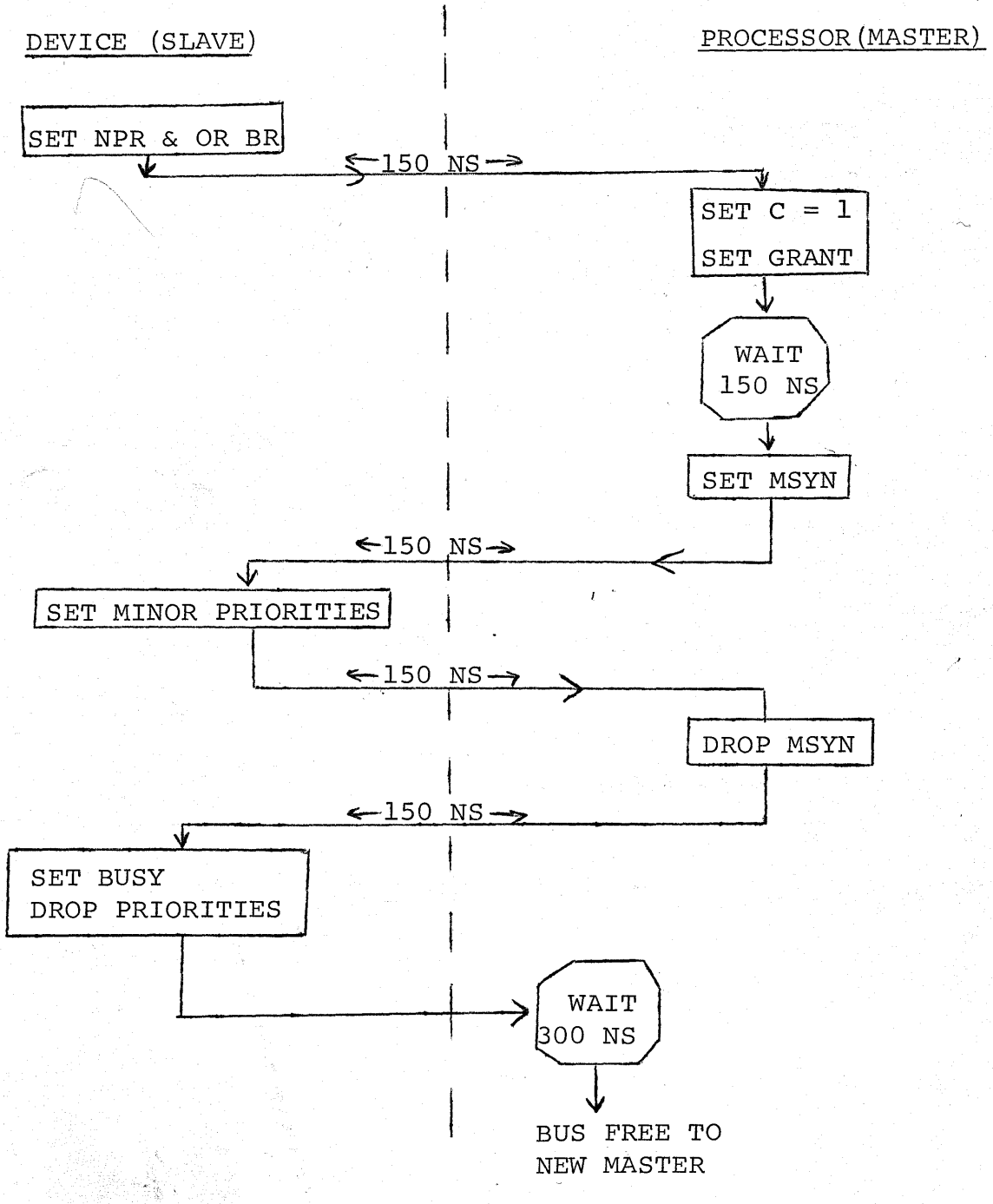
5. Memories with cycle time $<$1 usec may be desirable on these machines some day.

6. Goodies like multi-processor configurations, multiple memory buses, and memory interleaving are reasonable topics for consideration in medium (15) to large (10) machines.

APR 23 1969

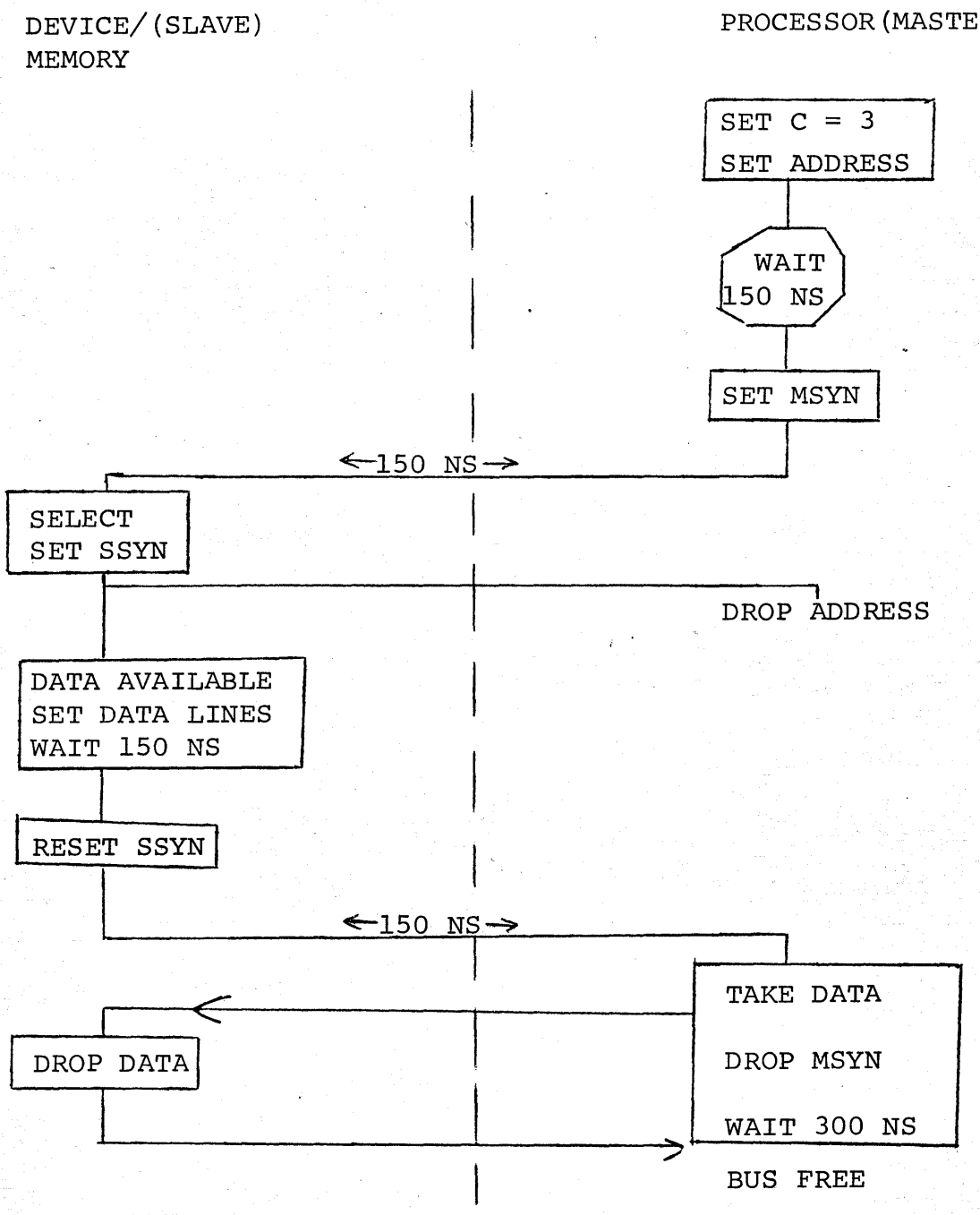
PRIORITY BUS TRANSFERS

Bus = 75 feet



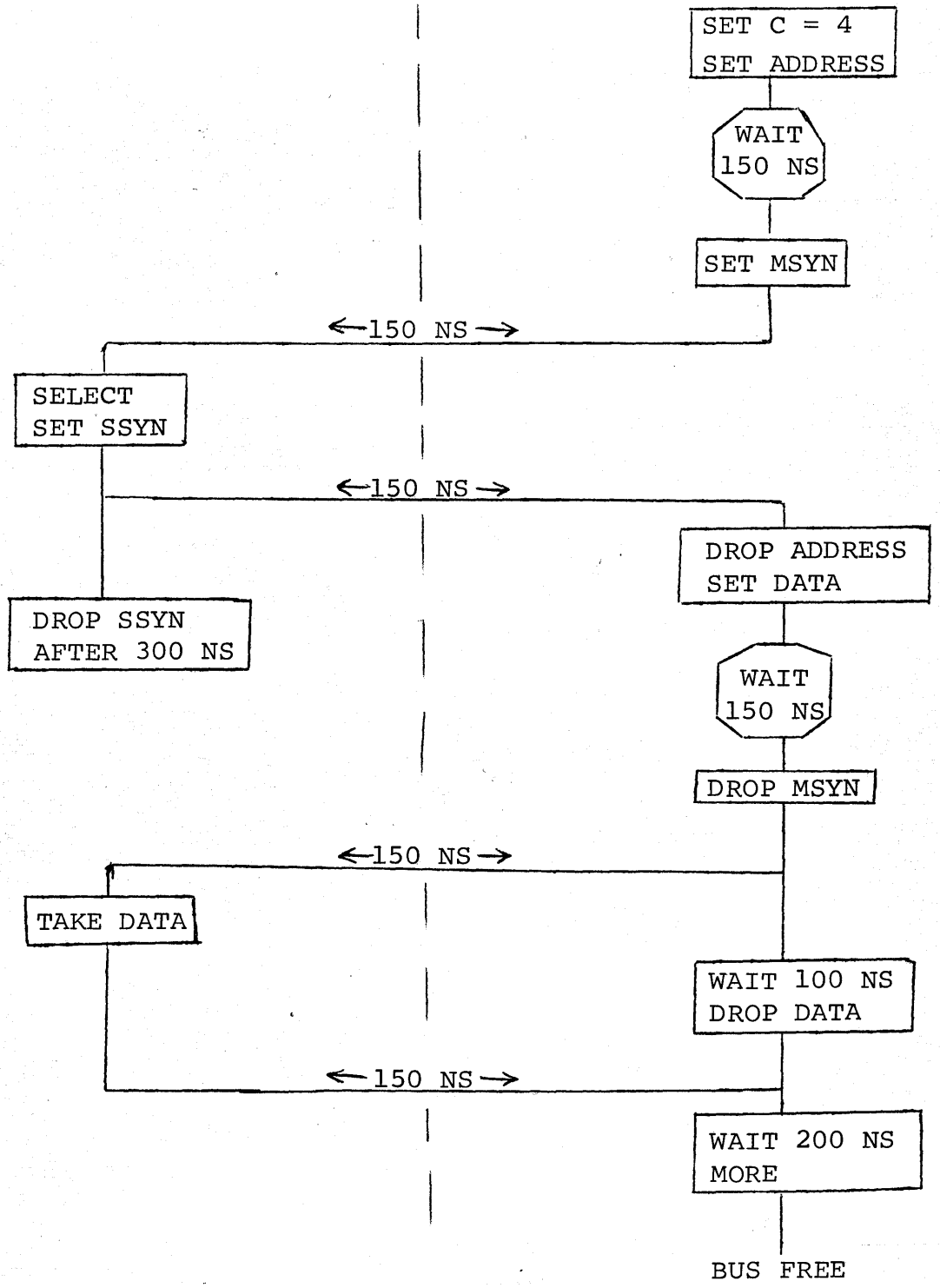
PRIORITY BUS TRANSFER = (5 x 150 NS) + 300 NS = 1.05 usec

DATA IN TRANSFER



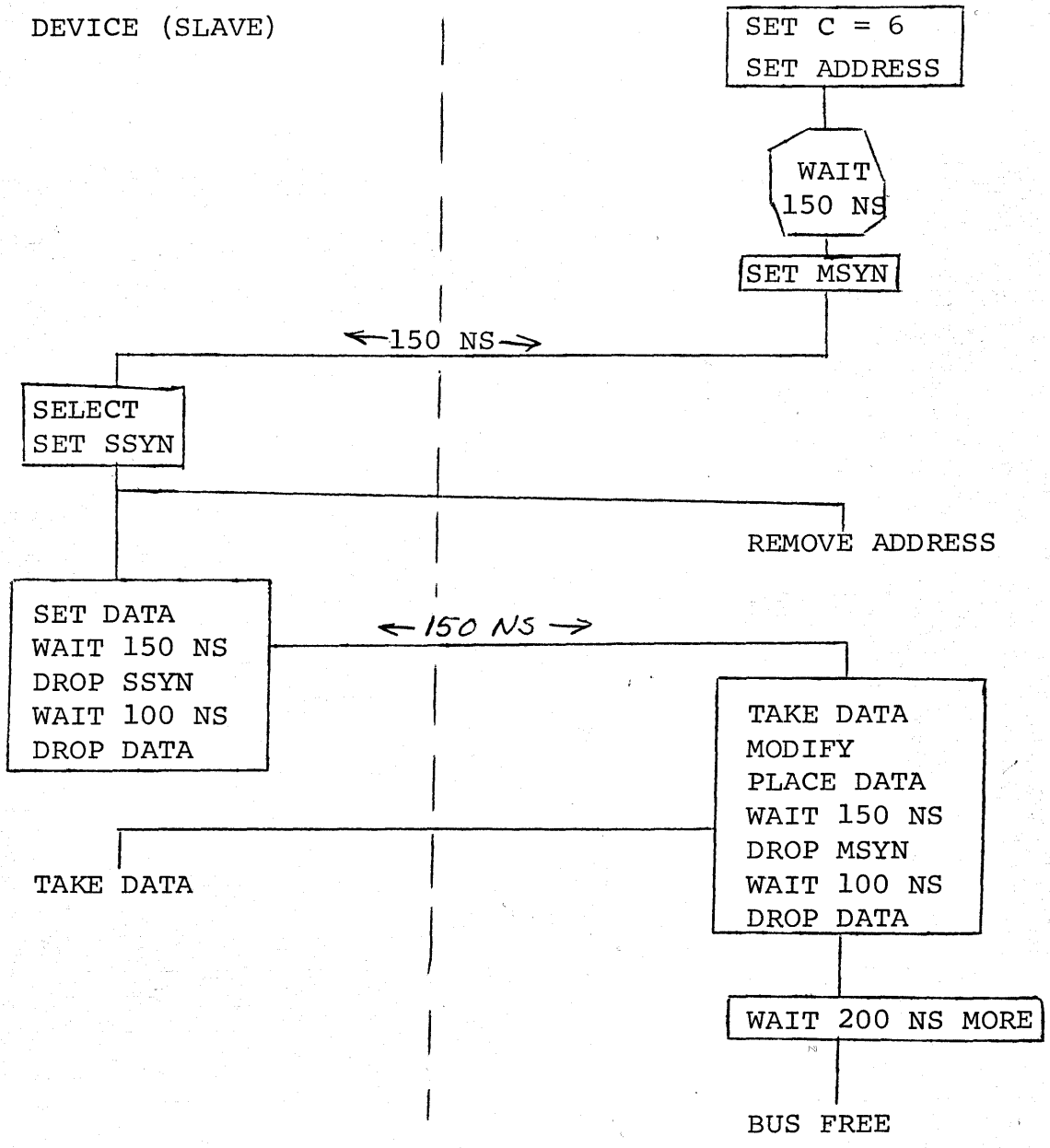
DATA IN TRANSFER 900 NS with Zero Access Time
DATA IN TRANSFER 1.4 usec with 1 usec Memory

DATA OUT TRANSFER



DATA OUT = 900 NS

DATA MODIFY TRANSFER



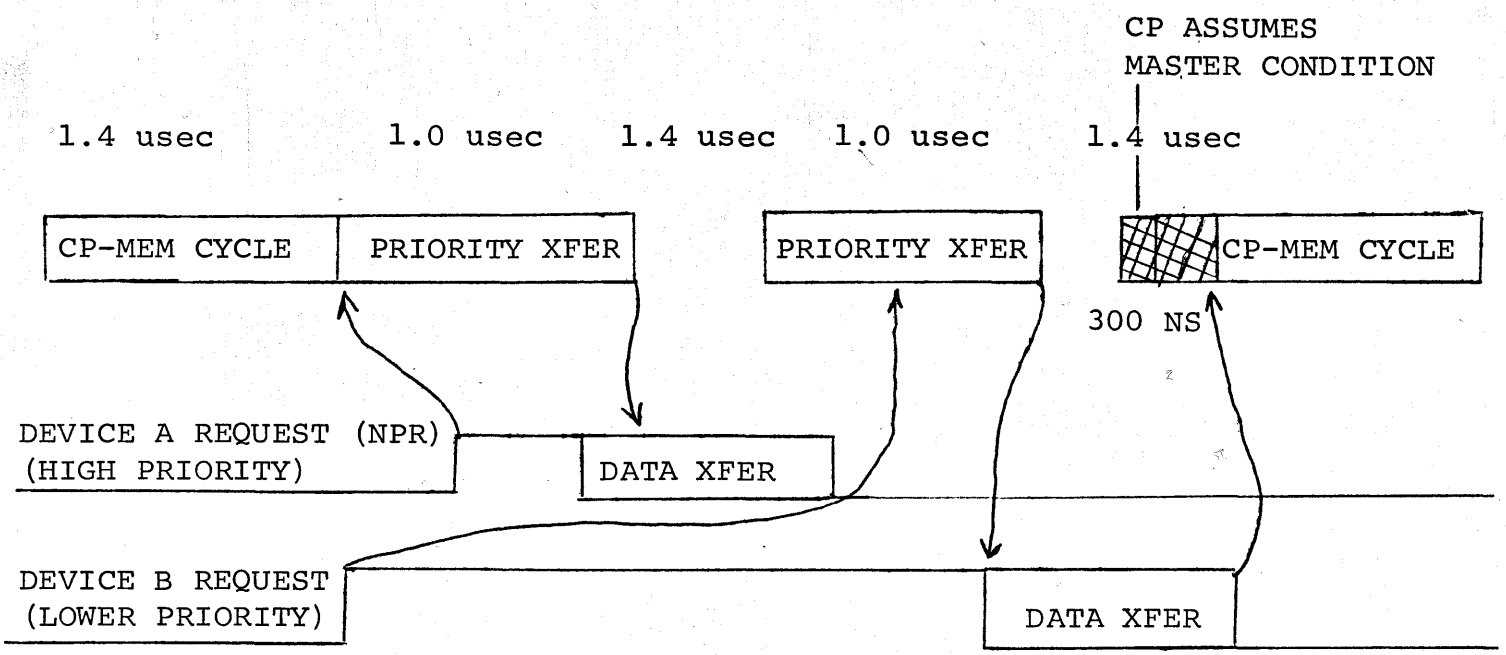
DATM = 5 x 150 NS + 300 NS = 1050 NS

A cursory examination of the bus sequences (see attached charts) shows typical bus sequence times of .9 to 1.1 usec. Furthermore, these sequences will affect the minimum instruction cycle time achievable in the system. Based on guesstimation it looks as if this instruction cycle time will be limited to approximately 1.3 to 1.5 usec. This may be a serious limitation on larger systems (or even smaller ones) a few years from now. For example, look at competition, (SDS Sigma 2,5,7 750 NS - 900 NS, Lockheed Mac 1 usec, PDP-15 800 NS, SEL 810 B 790 NS, etc). Using this type of interlocked bus sequence bus times can be reduced, by changing things around a bit, however, not by very much. With a 20 foot limitation on bus length the PDP-15 memory achieves 700 NS bus sequence times. with this method.

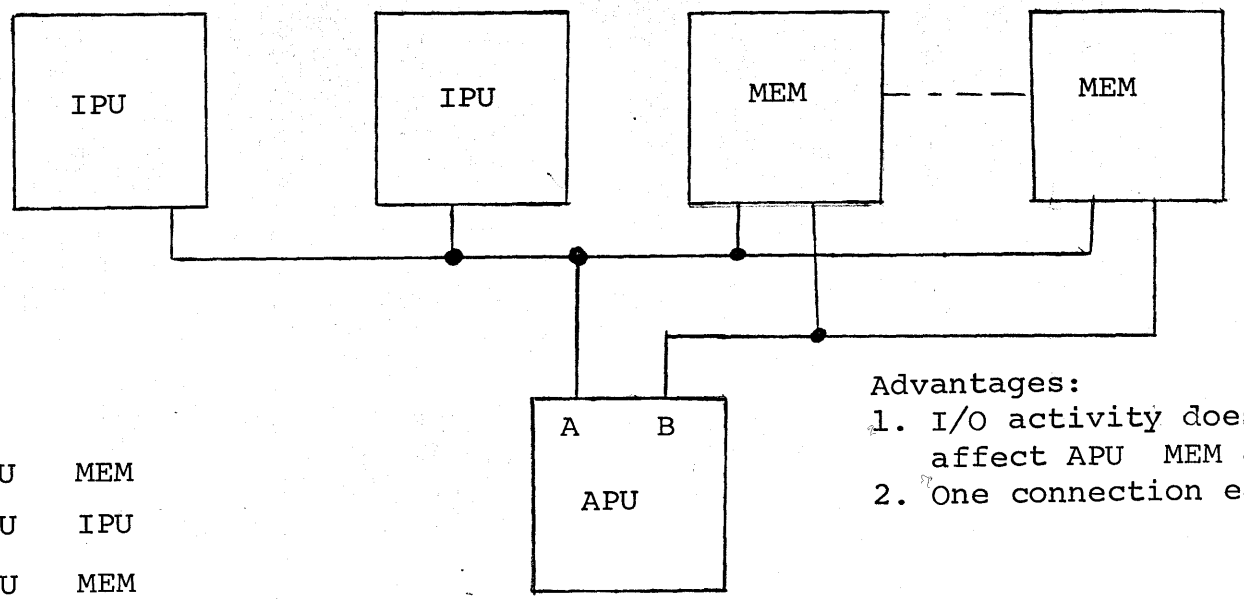
Also these sequences do not achieve total interlock. If you want and need total bus interlock considerable modification is required of your sequences. For example if your bus length lengthens,delays would have to be changed in the master devices. With a total interlock scheme masters and slaves close to the processor would run fast and ones further out on the bus more slowly without any delay adjustments. It appears to me that this bus scheme falls short of full interlock and thus a lot of advantages are lost. At the same time you are paying for all the disadvantages of an interlocked scheme (slow communication time, more complex control logic, etc).

I would recommend taking a serious look at the bus sequence times required now and 2 to 3 years in the future to attain competetive speeds. I would also suggest that you look at the PDP-10 type of memory bus to see how they do it. (The 10 does not use DC interlock. They do achieve very fast bus time however). You might also look at the PDP-15 memory bus for an example of <u>total</u> interlock.

In addition to the basic memory sequencing there are some I/O limitations - See drawing. In the drawing it is shown that the fastest back-to-back transfer of data to/from memory on different devices is $\backsim$ one word/byte every 2.4 usec. While doing this the bus is 100% busy, the cpu is computing but totally locked out of memory and memory is being used to $\backsim$ 50% efficiency. The latency of the system appears to be the length of a pause (DATM) reference to memory plus a priority transfer plus half the data transfer

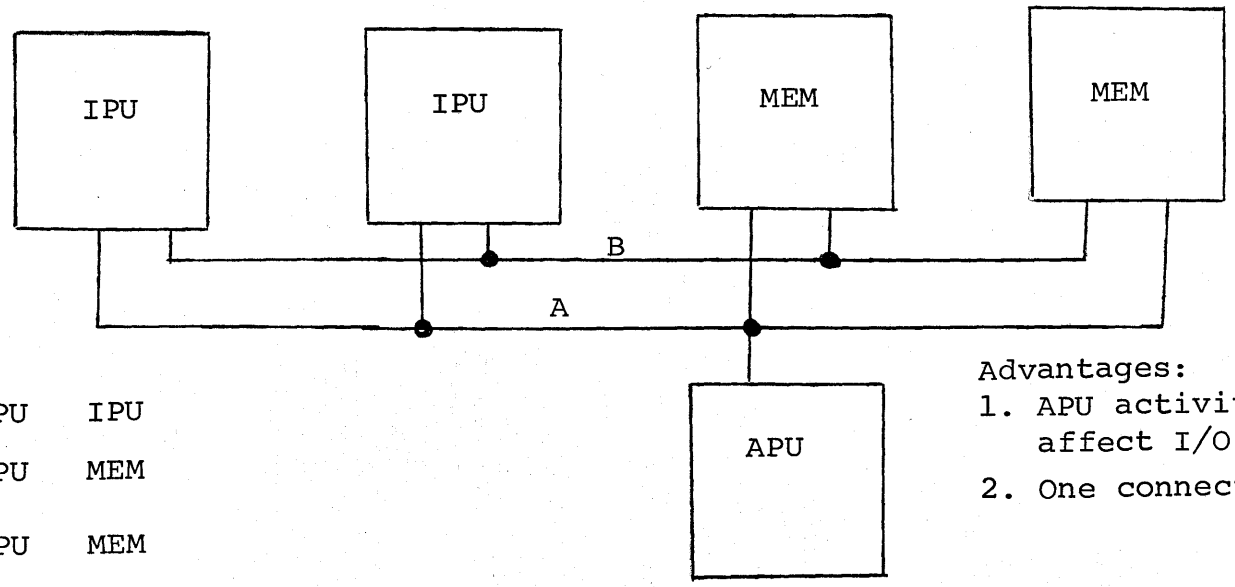


CONDITIONS - Two Devices request just after CP-MEM Transfer starts,Device A does transfer, then Device B, then Control is returned to CP.



Bus A. IPU MEM
APU IPU
Bus B. APU MEM

Advantages:
1. I/O activity does not affect APU MEM activity
2. One connection each device



Bus A. APU IPU
APU MEM
Bus B. IPU MEM

Advantages:
1. APU activity does not affect I/O MEM activity
2. One connection APU

or 1.6 usec + 1 usec + .7 us or about 3.3 usec. This is very good I think. However, the throughput rate and memory efficiency leaves something to be desired.

On larger system with lots of I/O gear, multiple memory buses are used to get up to twice the efficiency of use of the memories by allowing high density transfers into one area of memory and calculation programs in another area of memory. I haven't given much thought to adapting your bus to this approach. There are some loose ends though. (Consider who is the master and when on each bus). Two drawings attached show two possible schemes without regard to your particular bus problems.

To summarize what I've said see the list below:

1. The DC Interlock bus scheme is somewhat slow.

2. The PDP-11 bus doesn't achieve total interlock.

3. Therefore your paying the price (disadvantages) of the interlock scheme without getting the value (advantages) of the interlock scheme.

4. Present and future speed requirements should be carefully considered.

5. Look at (if you haven't already) the PDP-10 memory bus and the PDP-15 memory bus.

6. It appears that high I/O rates (stacked up requests) do not use memory at 100% efficiency.

7. The I/O bus limits back-to-back transfer rates from different devices to ≈2.4 usec between words (this is better than average but we could do better I think).

8. For larger systems multiple bus configurations should be considered.

I hope that you will utilize these comments as they are intended. i.e. constructive criticism. Please consider that I may not be completely in tune with your paper on this and therefore may have missed points which make the comments invalid. I hope you will straighten me out if I have any serious misunderstandings about your bus. My extension is 2215.

GVB;nfw