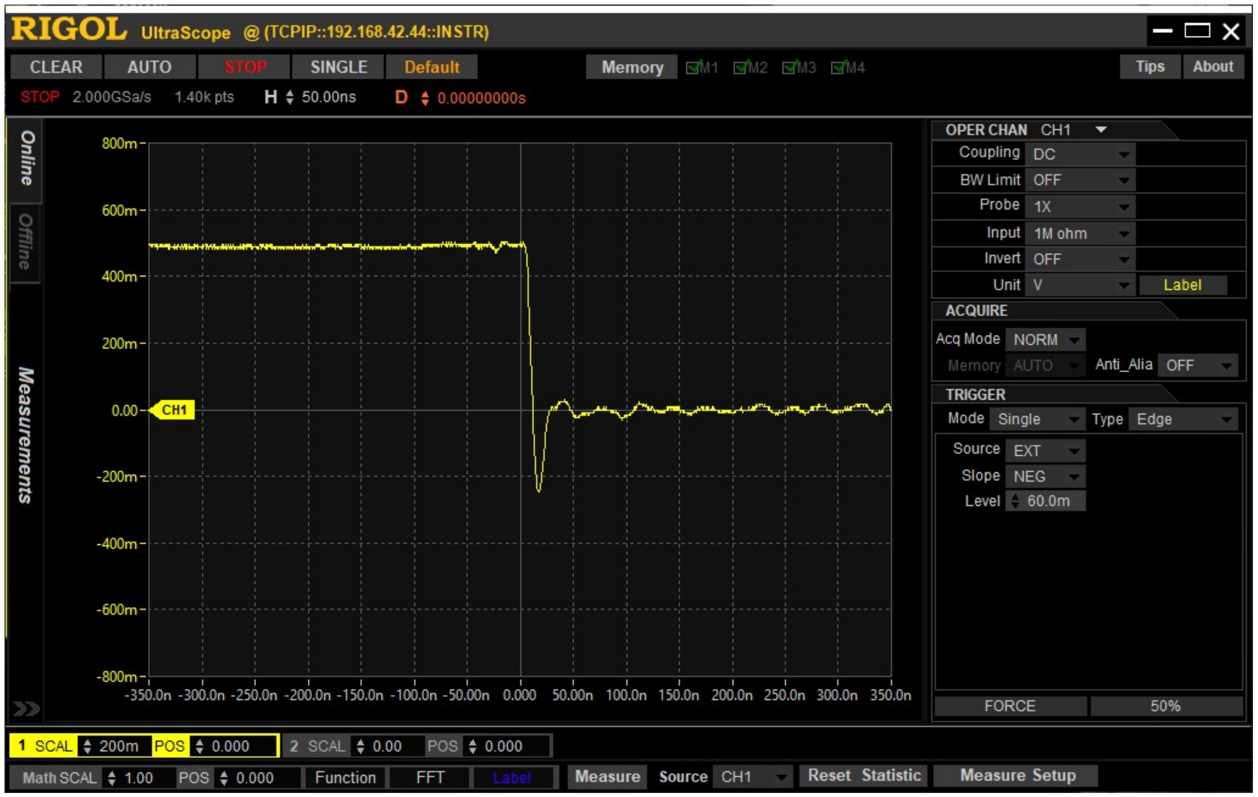
Setup: The jumper I normally have in place that connects TMAXPU to TMAx on the PDP11V2 card is removed – thus the card cannot actually request to be a bus master.

First, here is a screen shot of the TMAXPU signal going active (low) out of U25. (This signal should NOT have gone low – the machine was simply sitting in the console input poll loop). The trigger is the same signal, but connected to the external trigger input on the 'scope. (Generally in all of the traces, if the Trigger Source is EXT, it is connected to TMAXPU).



STOP 2.000GSa/s 1.40k pts H 50.00ns D 0.00000000s

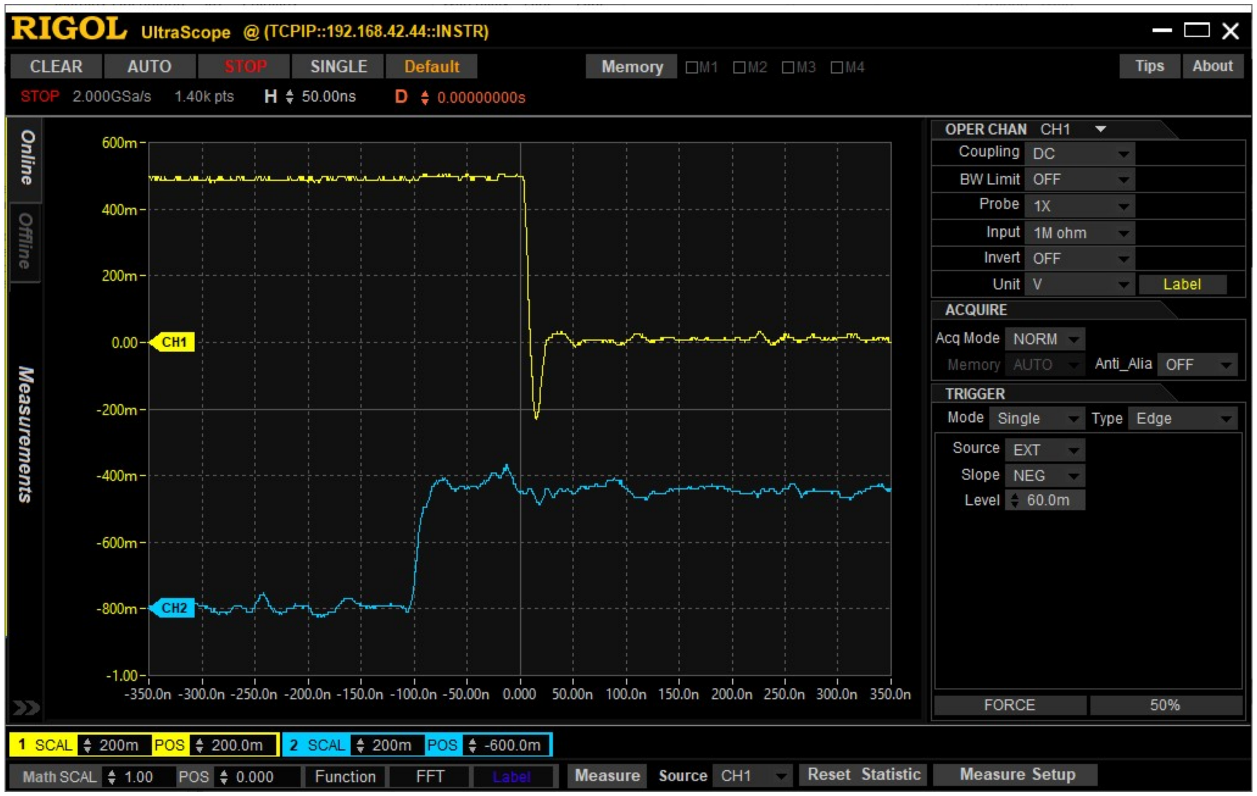
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 0.000 2 SCAL 0.00 POS 0.000
Math SCAL 1.00 POS 0.000 Source CH1

Next, I add signal pDBIN on channel 2 (pin 78 - the general read strobe) on channel 2.   Channel 1 is still TMAXPU and the external trigger is negative going on TMAXPU, as before. We can see that pDBIN goes active 100ns before TMAXPU:


RIGOL UltraScope @ (TCPIP::192.168.42.44::INSTR)

STOP 2.000GSa/s 1.40k pts H 50.00ns D 0.00000000s

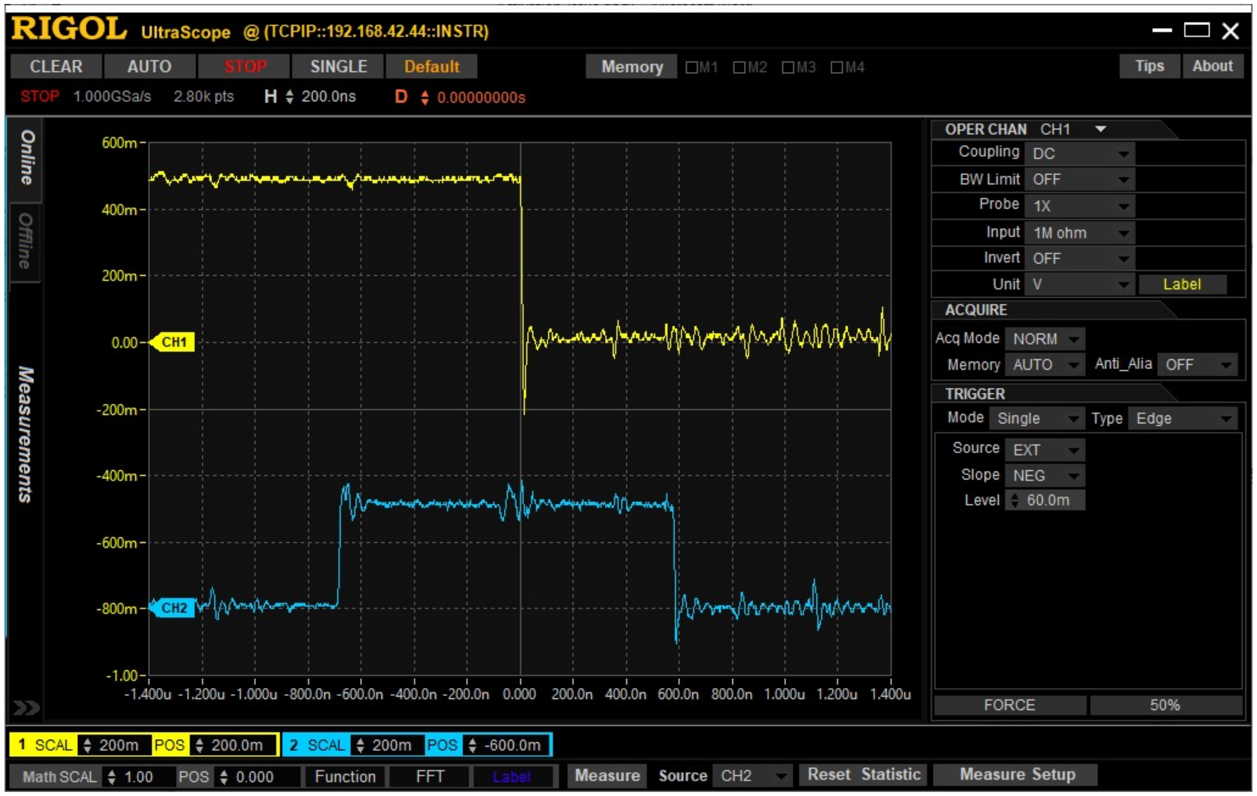
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Next, I replace pDBIN with sINP.   It is high – indicating an input I/O operation, as expected.   I also lengthened the timebase to provide more context – but the trigger point is unchanged. So, sINP goes active about 700ns before TMAXPU, and pDBIN went active about 100ns before as well.



STOP 1.000GSa/s 2.80k pts H 200.0ns D 0.00000000s

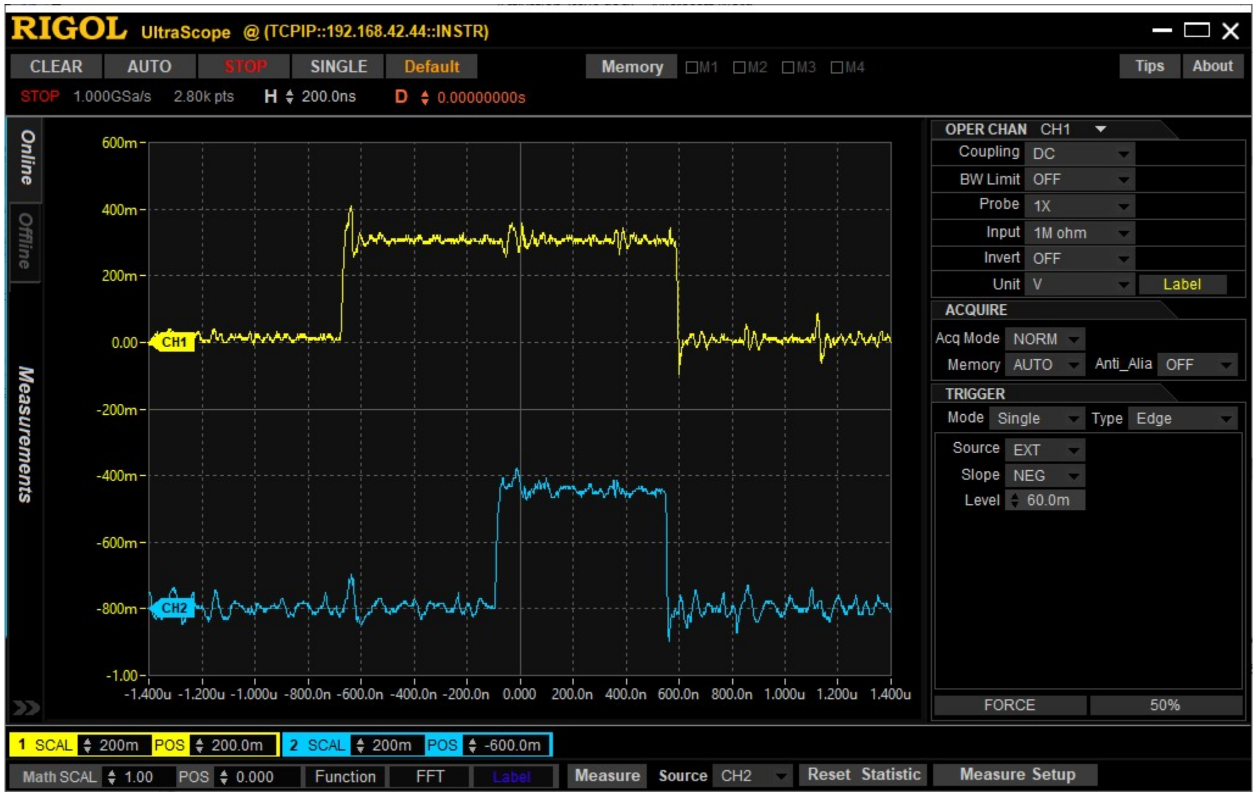
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V Label

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source EXT
Slope NEG
Level 60.0m

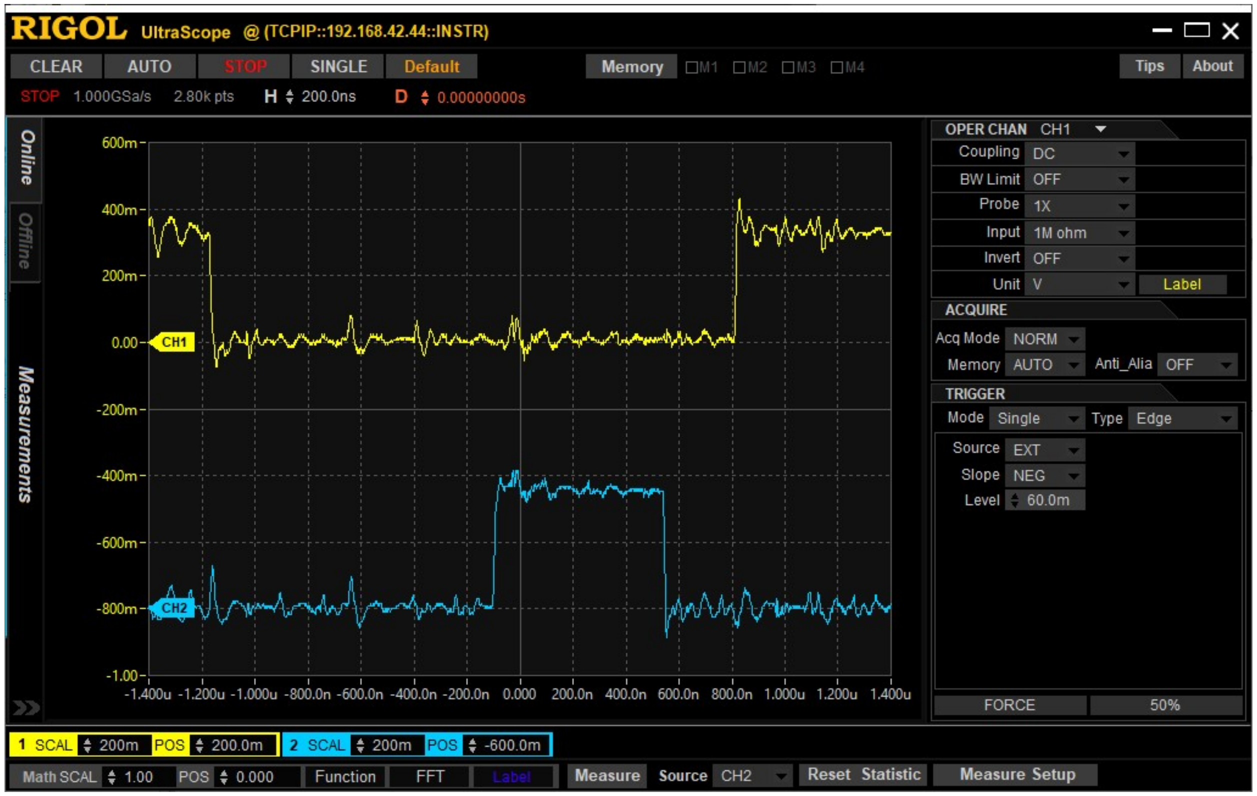
1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Source CH2

So next, I put up both sINP (bottom) and pDBIN (top) to show that relationship. The external trigger is still TMAXPU. Multiple captures demonstrated that TMAXPU is only triggering on an I/O operation (the console input loop, though sometimes it happens during the status polling for displaying the monitor sign-on message).



Since sINP is the later signal, I left that in place on channel 2, and picked an address line (A3) that it supposed to need to be a ‘1’ in U25. The trigger is TMAXPU.   The logic equation I use is this (because I have my Z80 set to mirror I/O addresses, I am only looking at the least significant 8 bits for 0xED):

```
!PULSE_TMA_H    =   (LA0 & !LA1 & LA2 & LA3 & !LA4 & LA5 & LA6 & LA7
                     & sINP & pDBIN);
reg1.d       =    'b'0;
reg1.ck      =    !PULSE_TMA_H;
reg1.ap      =     !MASTER_RESET # !PULSE_TMA_L;
!TMAXPU       =   reg1;
```



STOP 1.000GSa/s 2.80k pts H 200.0ns D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

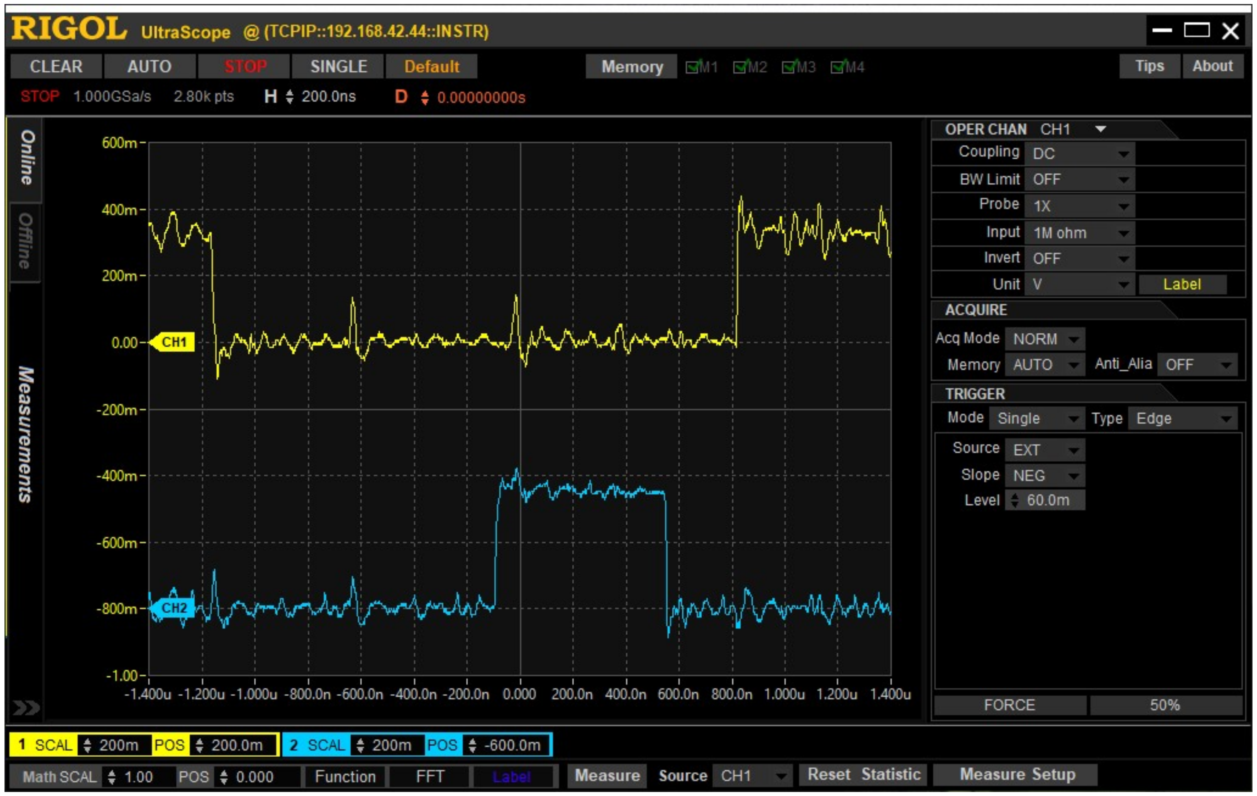
ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH2

WTF? A3 is NOT really a 1. But it is marginally close to 0.80v (x10 attenuation on the probe) to be in "no man's land" for logic levels. A3 is driven on the Cromemco ZPU by IC25, a 74367 (as are most of the bus lines), and appears on the PDP11V2 card at U30 pin 12, a 75LS244 (U30 - soldered in place underneath the J11). I tried swapping 74367's and even swapping in an 8T97 – no difference. It is also worth noting that these blips in A3 are happening every 200 to 400 ns.

Next was to look at the 74LS244. This is the same A3 signal, but taken at the 74LS244 instead of at the bus connector, triggered by TMAXPU. This signal is even more into the "gray area", which is interesting – that this noise is actually greater on the board than coming onto the board, which may explain why shielding helped – it would have added some capacitance and that may have tamped it down some.



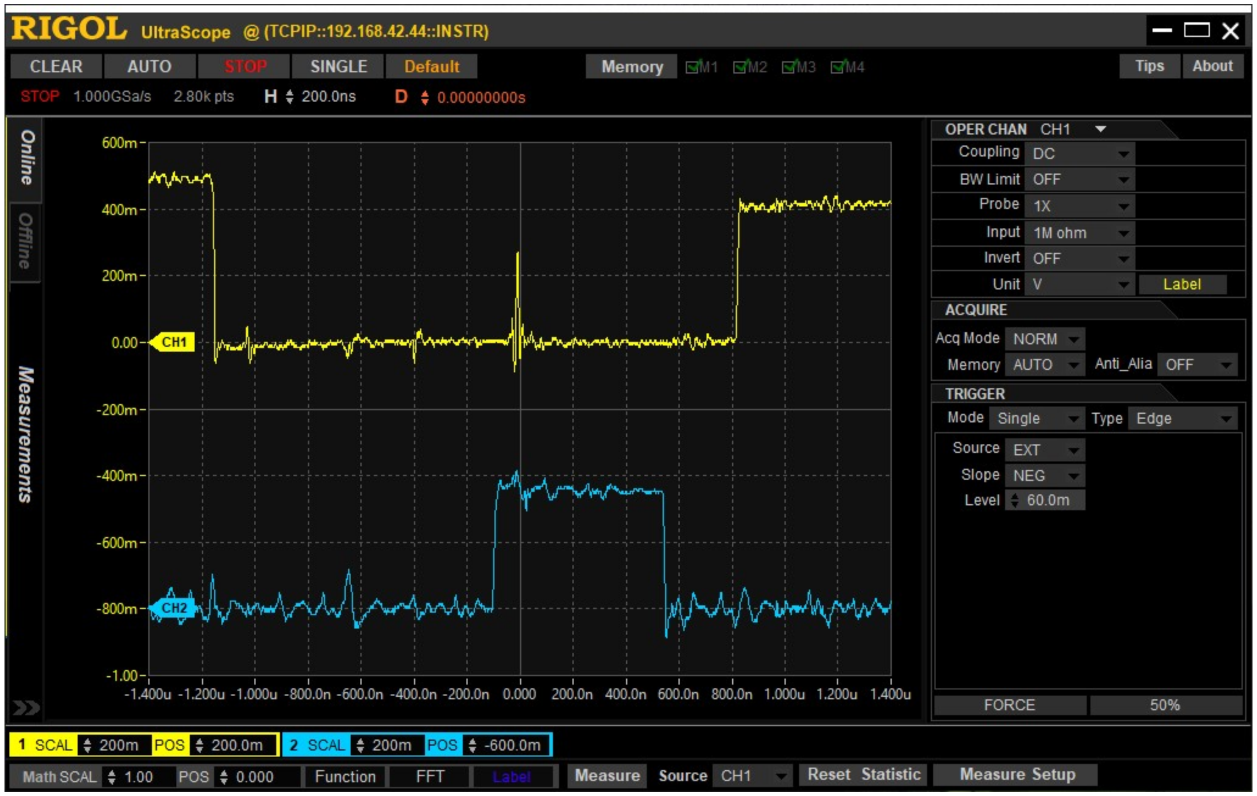
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V Label

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m

And here is A3 at the output of the 74LS244, triggered on TMAXPU, which shows a clear brief "glitch" on A3 in TTL "no man's land" that is enough for the CPLD to treat it as a 1. So, the closer I get to the CPLD input, the worse it looks.


RIGOL UltraScope @ (TCPIP::192.168.42.44::INSTR)
CLEAR
AUTO
STOP
SINGLE
Default
Memory
M1
M2
M3
M4
Tips
About
STOP 1.000GSa/s 2.80k pts H 200.0ns D 0.00000000s
Online
Offline
Measurements
CH1
CH2
600m
400m
200m
0.00
-200m
-400m
-600m
-800m
-1.00
-1.400u -1.200u -1.000u -800.0n -600.0n -400.0n -200.0n 0.000 200.0n 400.0n 600.0n 800.0n 1.000u 1.200u 1.400u
FORCE
50%
Function
FFT
Label
Measure
Reset Statistic
Measure Setup

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

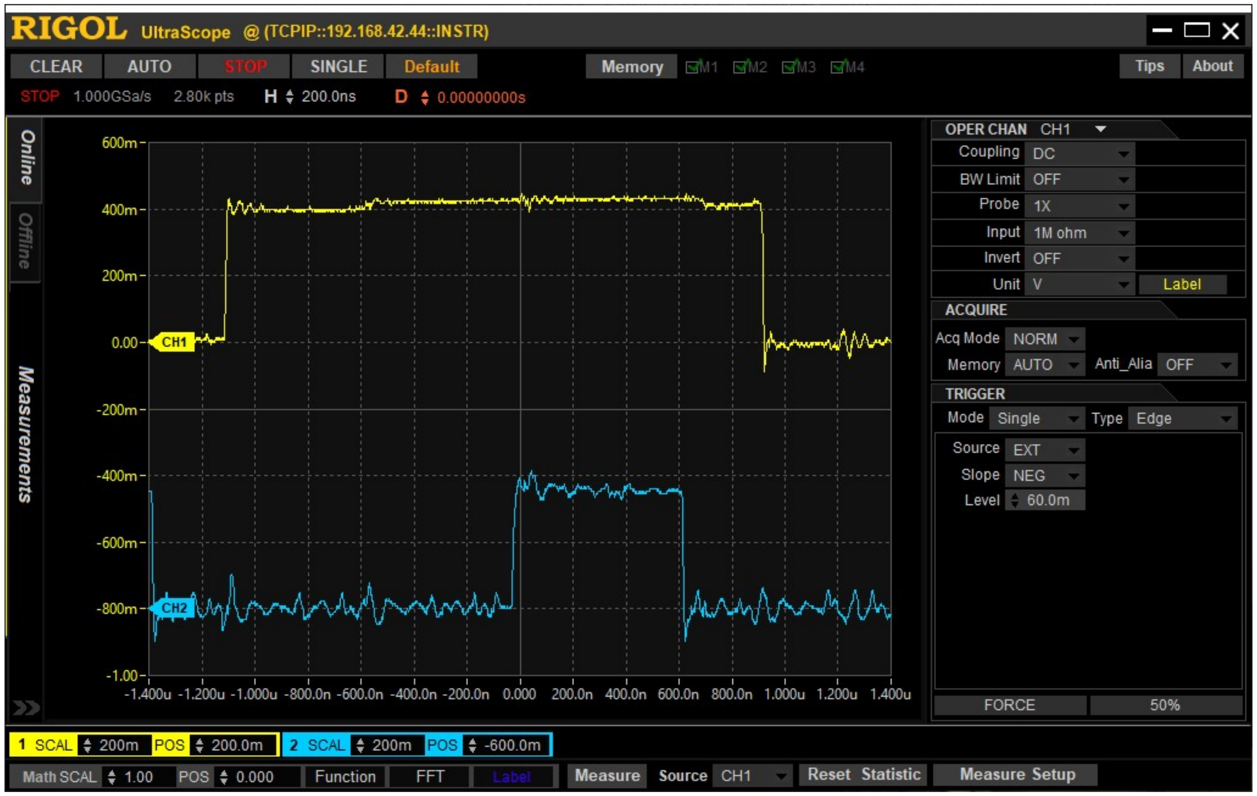
ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Note that this signal on A3 is not simply ringing – it is occurring when the signal is at 0 for an eternity (a full microsecond) before the problem. This is why adding termination did not help at all. It is happening roughly 100ns after pDBIN goes active, when nothing ought to really be going on anywhere.

And here is what it really ought to look like – with the address line going high well before the pDBIN signal goes active, rather than after. This was done with the J11 chip removed – in that configuration TMAPXPU never activates TMAXPU unless I use the O or W command. (This one is on the input of the 74LS244 U30, rather than the output. The top signal is A3. Trigger is TMAXPU).



STOP 1.000GSa/s 2.80k pts H 200.0ns D 0.00000000s

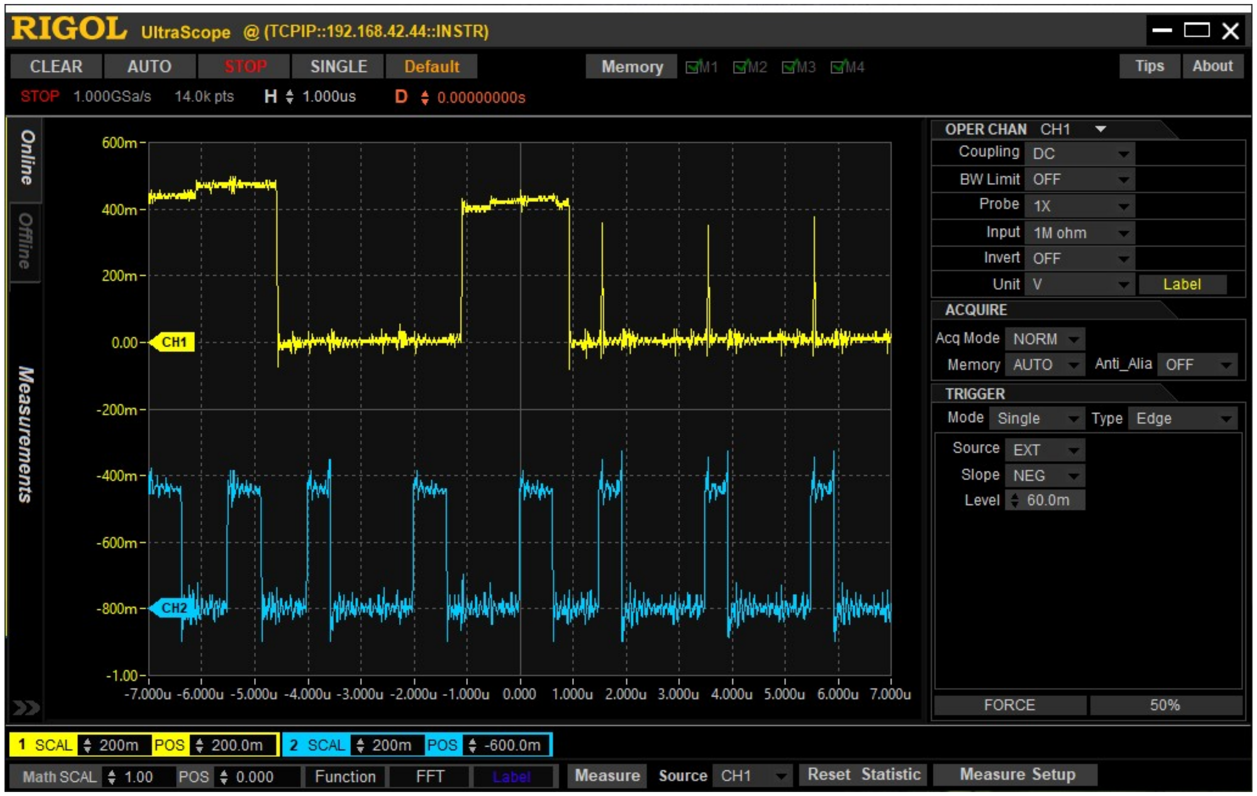
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Here is the same sequence, but with a longer timebase (again the J11 is removed). It shows A3 at the 74LS244 output spiking during some, but not all, pDBIN cycles – and, if one looks carefully, it is always about 80-100ns after pDBIN goes active (yes, it is hard to see at this scale). The trigger was the same (TMAXPU), but in this case that was manually generated by command.



STOP 1.000GSa/s 14.0k pts H 1.000us D 0.00000000s

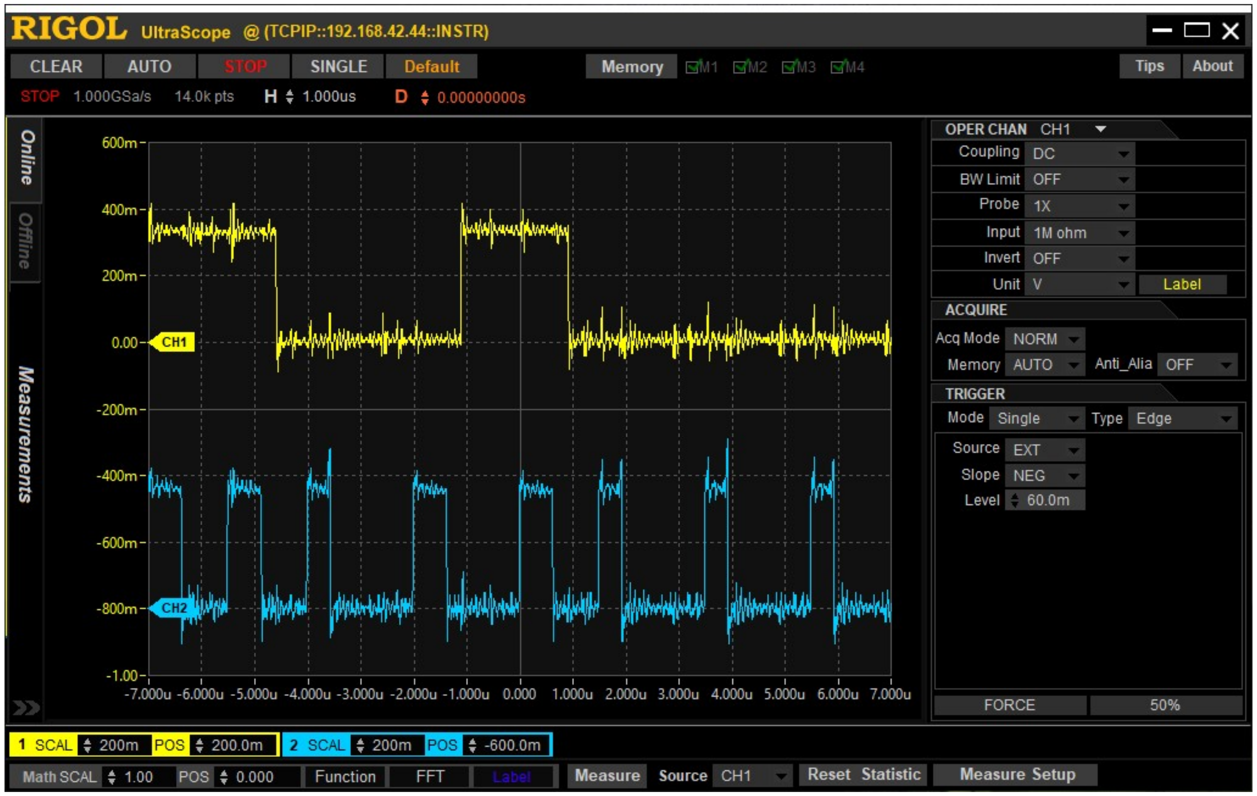
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

And here is the same sequence, but at the bus connector instead of at the output of the 74LS244 U30.



STOP 1.000GSa/s 14.0k pts H 1.000us D 0.00000000s

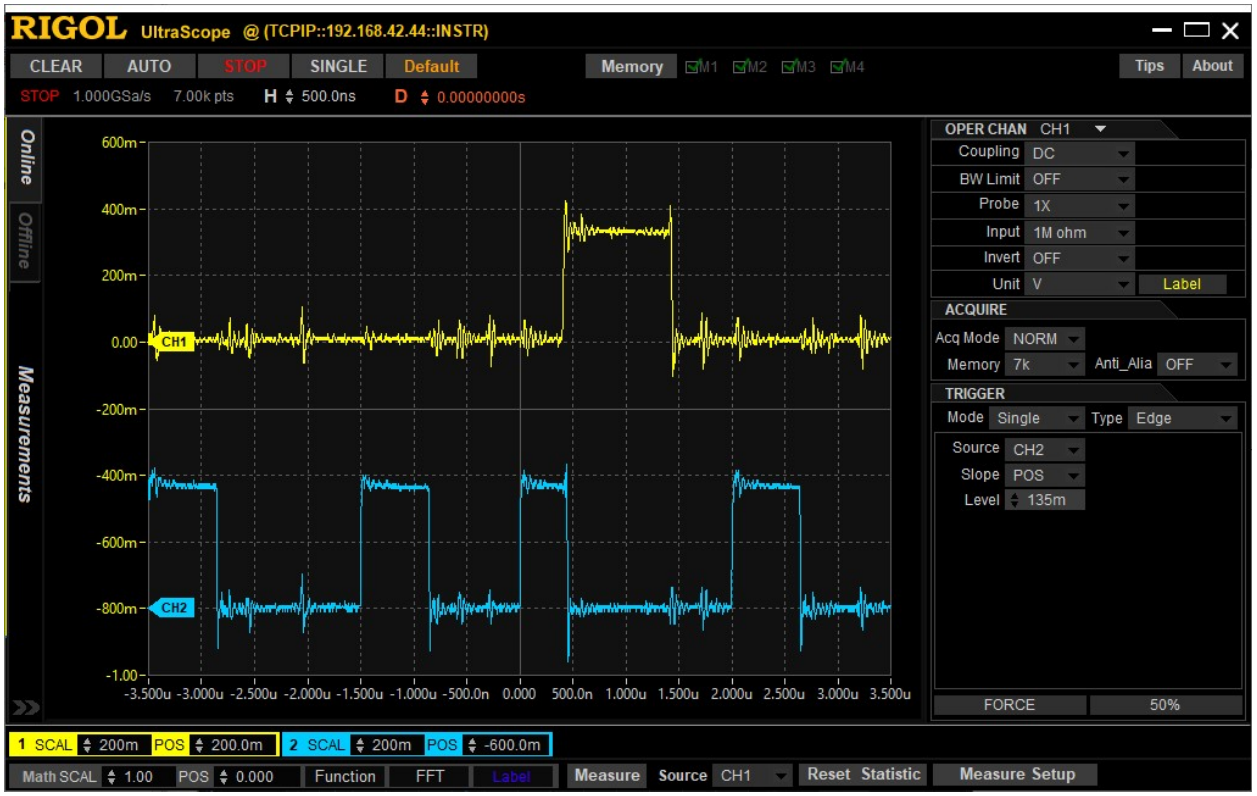
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V Label

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source EXT
Slope NEG
Level 60.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Here is a *typical* console input loop waveform (triggered on pDBIN) using the ZPU without the PDP11V2 board. As before, CH1 is A3, CH2 is pDBIN (which is also the trigger). This one was taken at 2MHz, as were the earlier waveforms:



UltraScope @ (TCPIP::192.168.42.44::INSTR)

STOP 1.000GSa/s 7.00k pts H 500.0ns D 0.00000000s

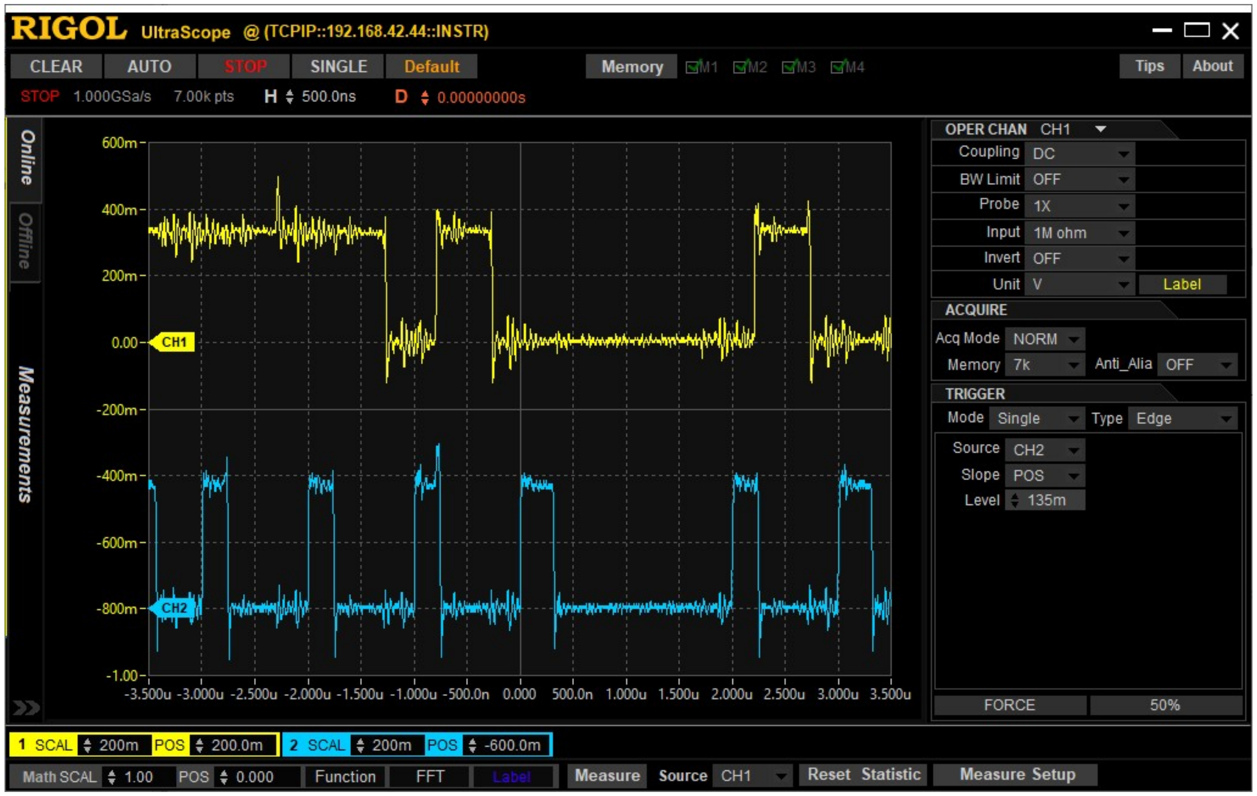
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory 7k
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH2
Slope POS
Level 135m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Here is a *typical* console input loop waveform using the ZPU without the PDP11V2 board. This waveform is taken at 4MHz to be consistent with the next one – but note that the earlier waveforms were taken at 2MHz. The ZPU is a good bit noisier than the newer Z80 board, below.



STOP 1.000GSa/s 7.00k pts H 500.0ns D 0.00000000s

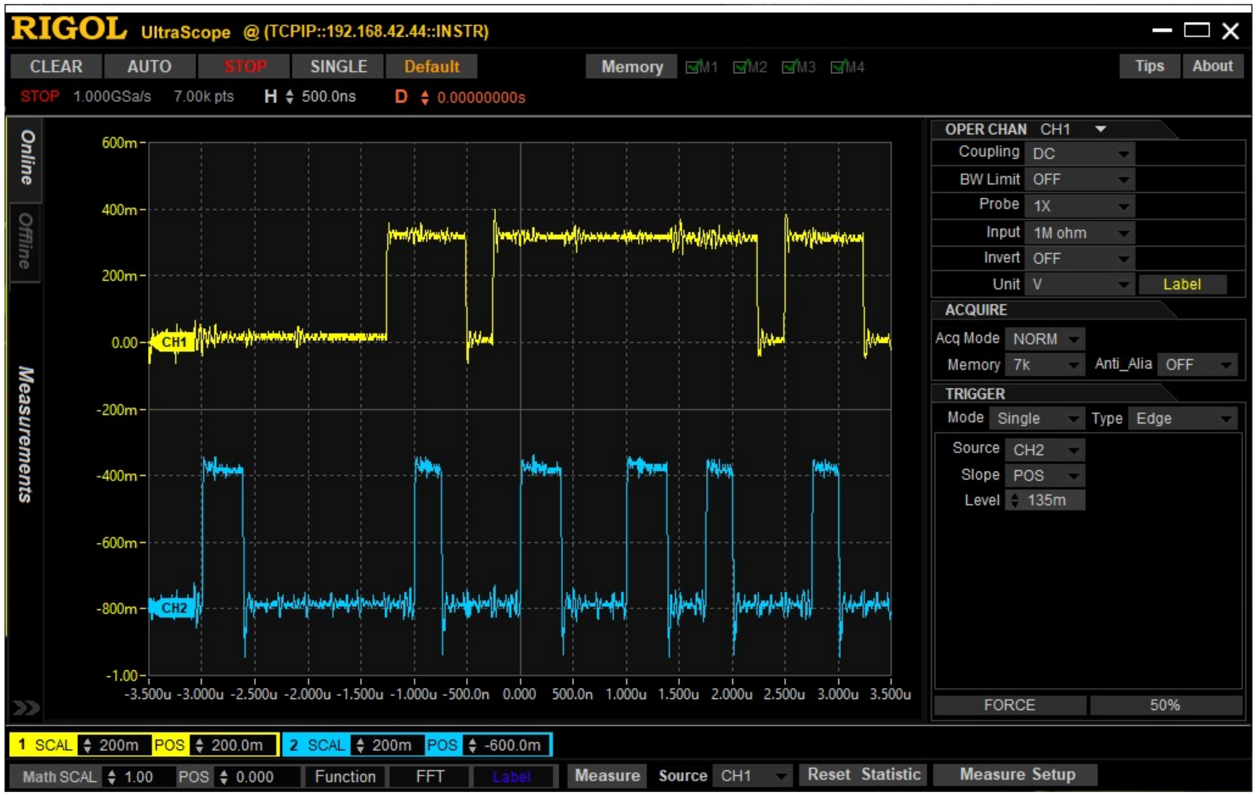
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory 7k
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH2
Slope POS
Level 135m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Here is a *typical* console input loop waveform using John Monohan's Z80 board, without the PDP11V2 board. Note that there is somewhat less noise on the address lines – but there is still some present, consistent with the observation that the false activation of the PDP11V2 board happens less often with this CPU board – but does still occur (though there is not any indication of spike on A3 sufficient to be a trigger.)



@ (TCPIP::192.168.42.44::INSTR)

STOP 1.000GSa/s 7.00k pts H 500.0ns D 0.00000000s

OPER CHAN	CH1
Coupling	DC
BW Limit	OFF
Probe	1X
Input	1M ohm
Invert	OFF
Unit	V

ACQUIRE

Acq Mode	NORM
Memory	7k
Anti_Alia	OFF

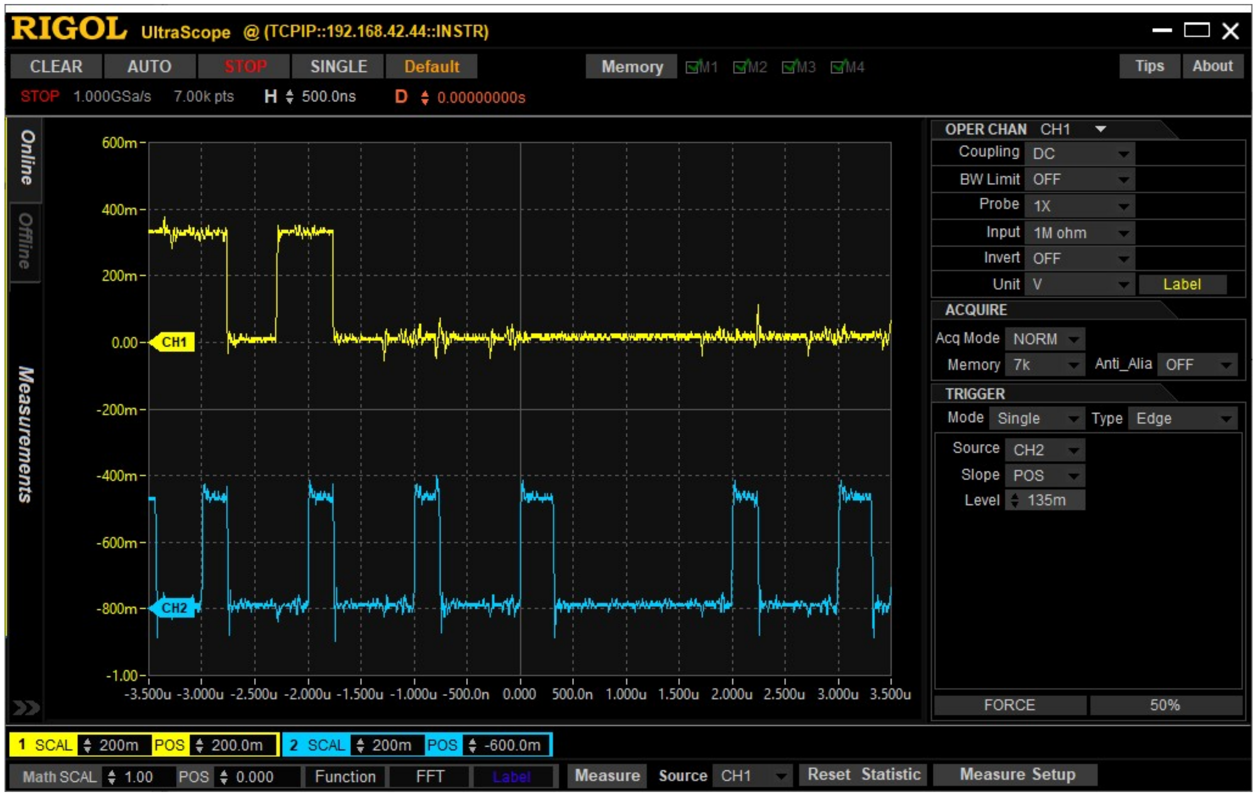
TRIGGER

Mode	Single
Type	Edge
Source	CH2
Slope	POS
Level	135m

1 SCAL 200m POS 200.0m 2 SCAL 200m POS -600.0m

Math SCAL 1.00 POS 0.000 Source CH1

This picture is the *typical* console wait loop with the ZPU back in the system, and a Godbout active bus terminator at the end of the bus. There is considerably less noise, but the address line spikes are still present. (Especially notable is the one towards the right of the trace). Trigger is Ch2 (pDBIN)



STOP 1.000GSa/s 7.00k pts H 500.0ns D 0.00000000s

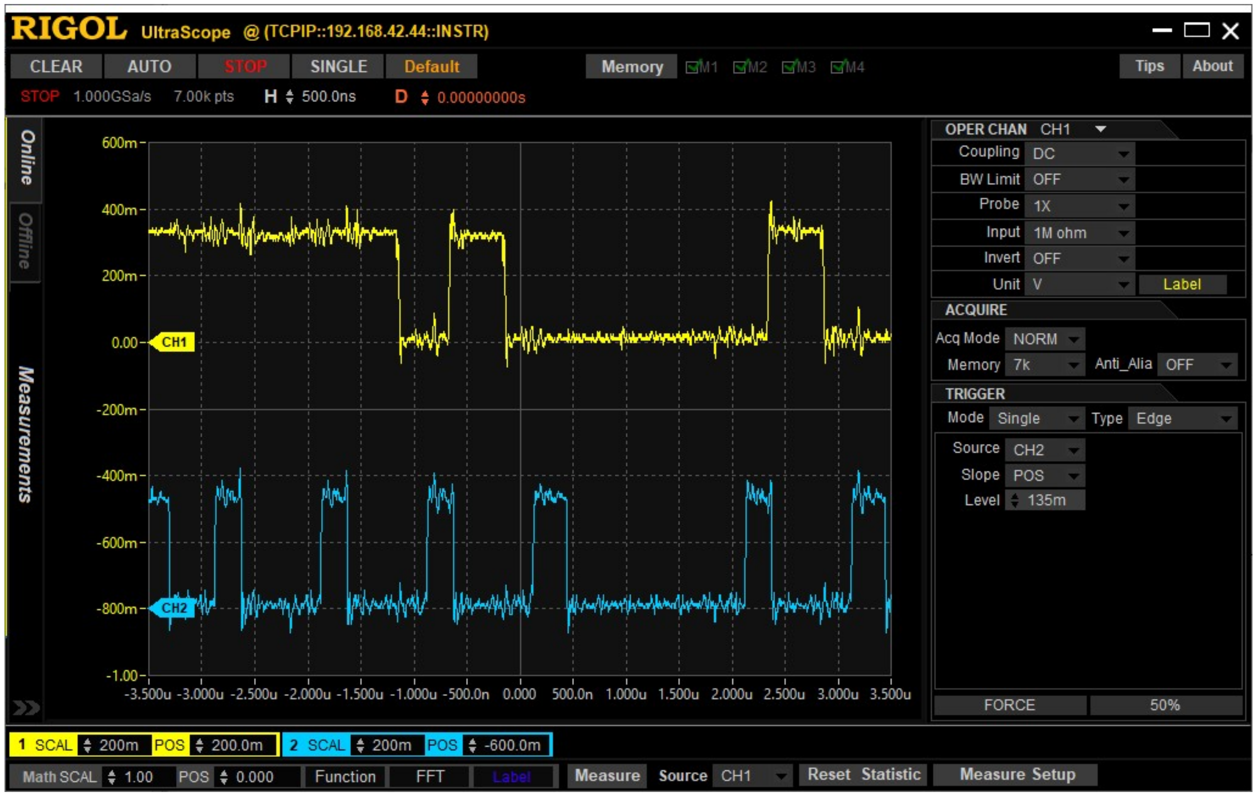
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V Label

ACQUIRE
Acq Mode NORM
Memory 7k Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source CH2
Slope POS
Level 135m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

This picture of a *typical* console wait loop was taken with the PDP11V2 board back in place, the ZPU, with active termination. The noise on the address line is still apparent (note particularly the right hand side of the screen, where the spike occurs during the input cycle.) Trigger is pDBIN



STOP 1.000GSa/s 7.00k pts H 500.0ns D 0.00000000s

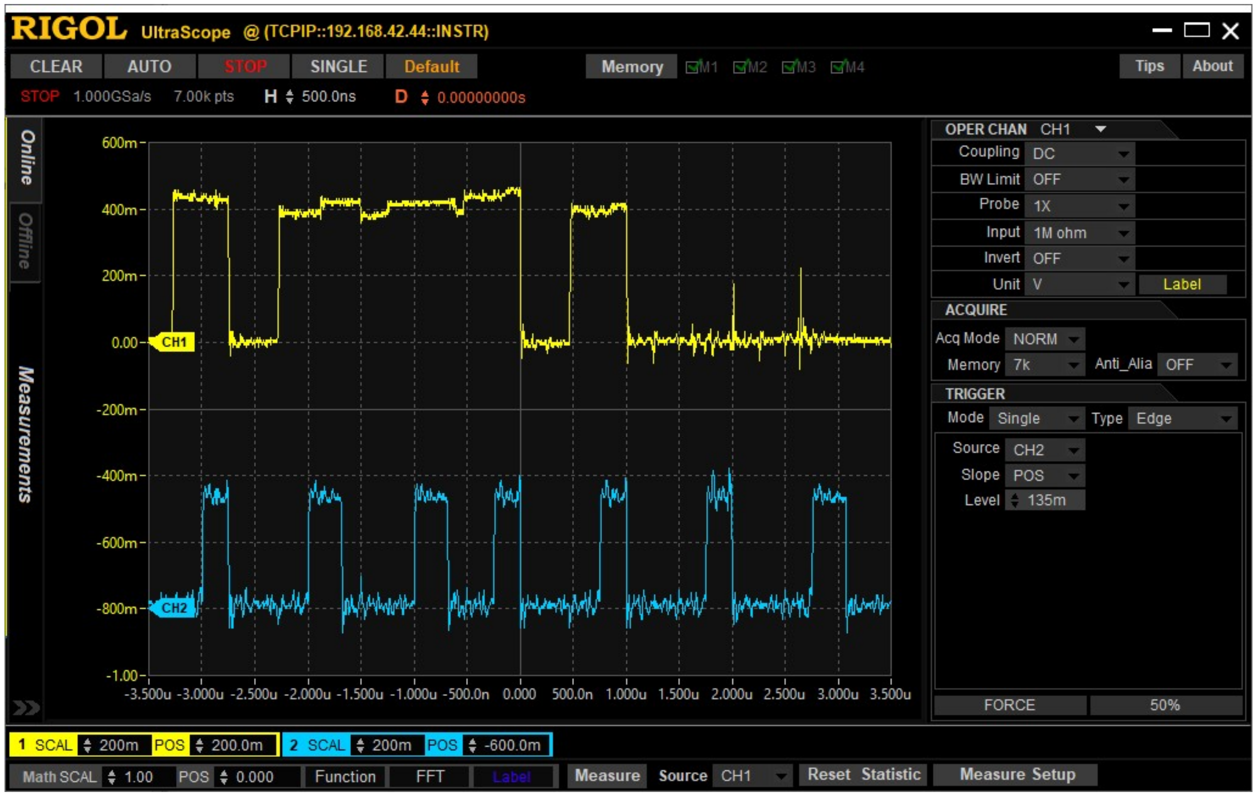
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory 7k Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source CH2
Slope POS
Level 135m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

And here it is with the active terminator in place, with the ZPU at the output of the 74LS244, U30 (but not the same timing as above, of course), with the 2V+ spikes present.   Trigger remains pDBIN.



STOP 1.000GSa/s 7.00k pts H 500.0ns D 0.00000000s

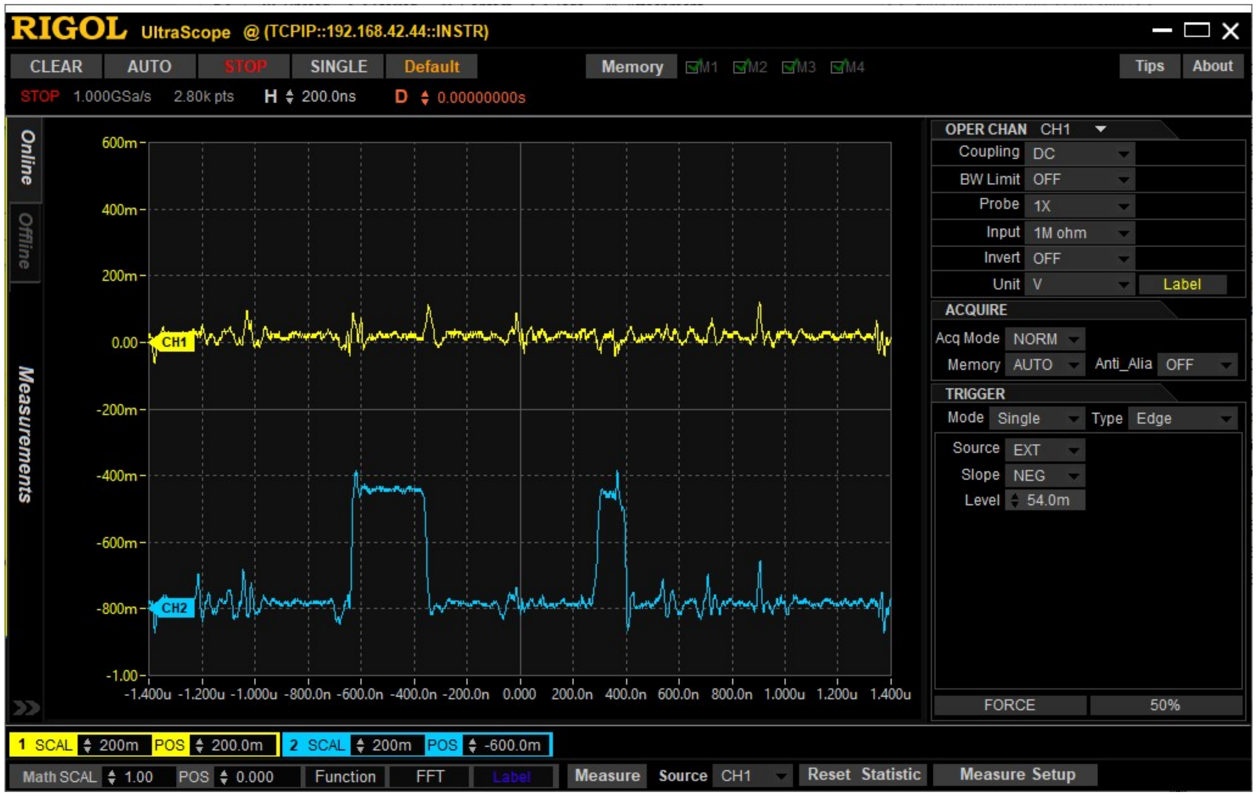
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory 7k Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source CH2
Slope POS
Level 135m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Finally, here is a trace with A3 and A11. My hope is that if I include some of the high order address lines in the equation (even though they should be the same as the low order address lines, as the ZPU board is jumpered to mirror I/O addresses) it will be less prone to the issue. (The trigger is back to TMAXPU).



STOP 1.000GSa/s 2.80k pts H 200.0ns D 0.00000000s

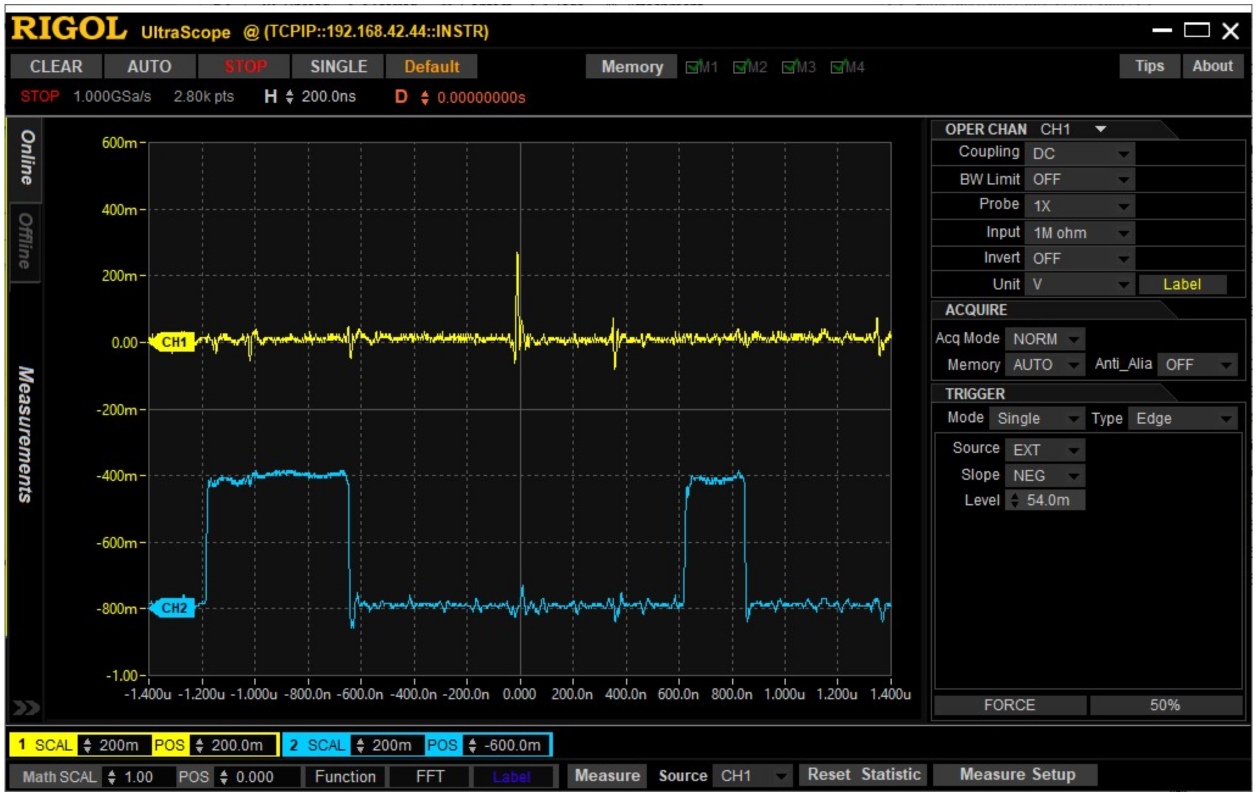
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope NEG
Level 54.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

This trace, with the output of U30 pin 12 on CH1 (LA3) and U37 pin 12 on CH2 (LA11) shows that such an approach has some hope – and adds to the understanding of why I am having the issue and few/nobody else is. If the SMB is being used to activate TMAx, then this issue is not involved – the activation signal TMAx is driven by TMA0* or … TMA3*. Secondly, if one is using all 16 bits (well, actually 13, since only A0 thru A12 are really in play), then this is much much less likely to occur. (Again, the trigger is TMAXPU). Had A11 been in the equation, this would not have triggered.



STOP 1.000GSa/s 2.80k pts H 200.0ns D 0.00000000s

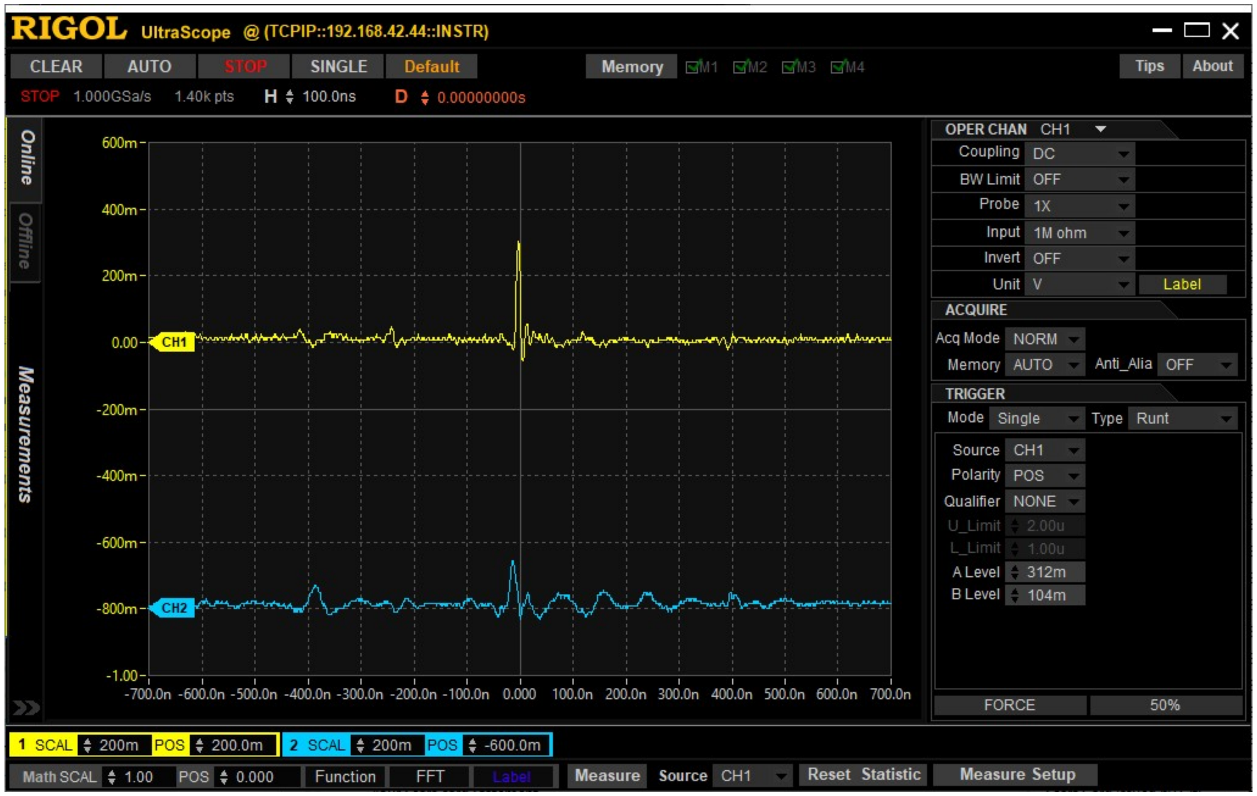
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope NEG
Level 54.0m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

As a final check, I connected CH1 to the output of A3 on the 'LS244, and used the 'scope's "runt" trigger to capture the pulses (independently of TMAXPU), and displayed the input on CH2. (I also separately checked A11, and there were certainly times when A11 spiked as well.)



STOP 1.000GSa/s 1.40k pts H 100.0ns D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

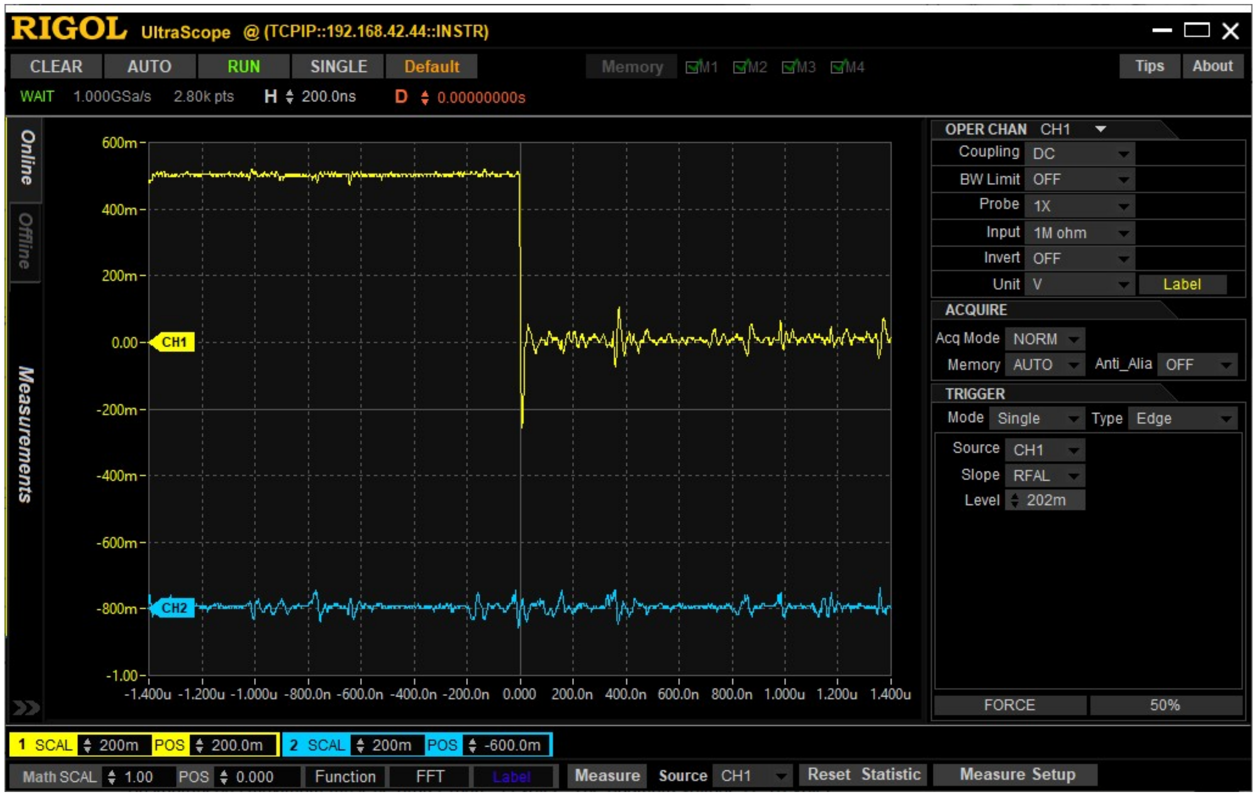
TRIGGER
Mode Single
Type Runt
Source CH1
Polarity POS
Qualifier NONE
U_Limit 2.00u
L_Limit 1.00u
A Level 312m
B Level 104m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

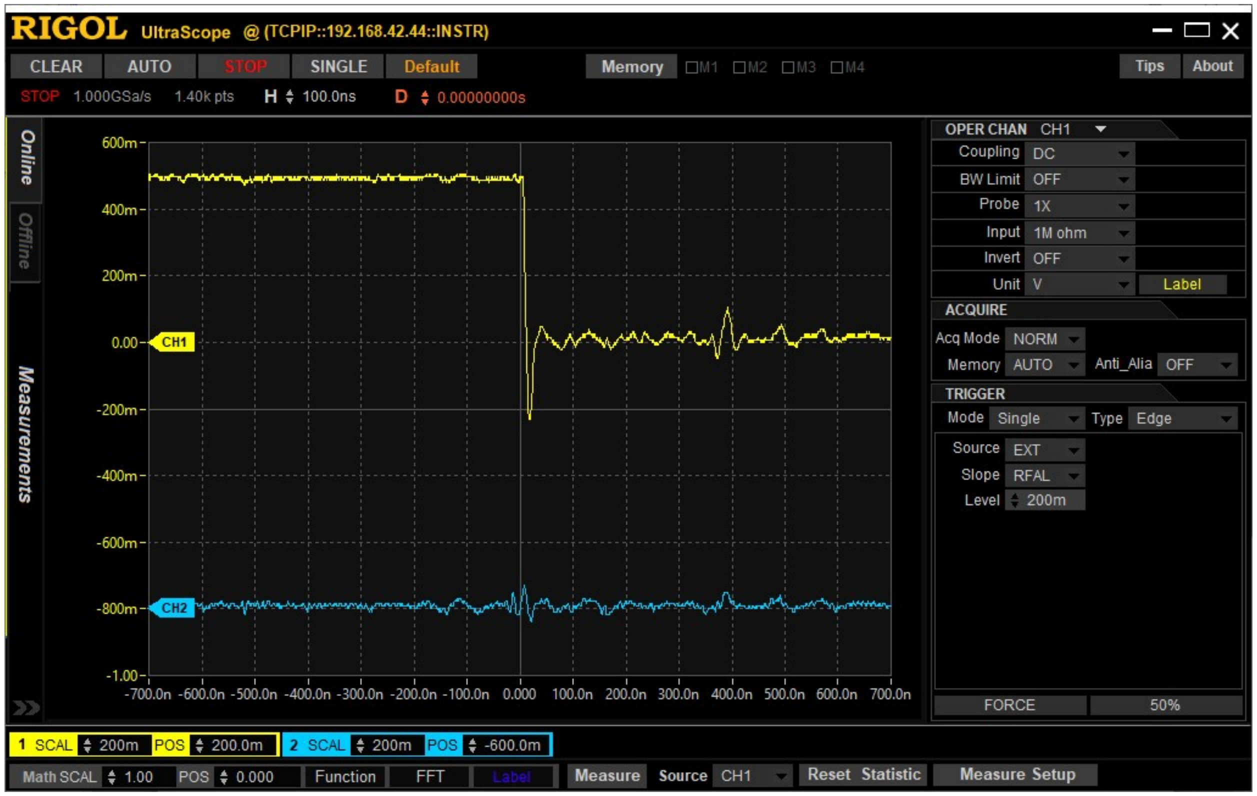
Observations:

- This isn't a case of a bad chip, mis-wiring, a solder bridge, or the like: this is occurring on all of the address lines. And, because OE is active on U30 (and U37) at all times, it isn't an enable issue.
- I am using address mirroring on the I/O port, so I am currently only testing 8 address bits to compare for the PDP11V2 board activation address. That makes it more susceptible. Others using 16 bits of I/O address, or using the SMB to activate the board are less likely to see this. At times I have seen this not trigger for hours (e.g., particularly when running the ZPU at 4MHz).
- If the 74LS244 were not present, this might not be occurring, but it might. At the very least, the 74LS244 (U30) is amplifying the signal and increasing the probability.
- The address lines are noisy, but it isn't ringing that is causing this. So while adding the active terminator does help some, it wasn't enough to make it go away.
- Moving boards around on the backplane affects this quite a bit. With the right arrangement and the active terminator, or placement of the shield, the issue goes away completely, or nearly so.
- It is sensitive to ZPU CPU speed, which is easily switchable. If it is happening often at 2MHz, then it happens less frequently at 4MHz (and probably vice versa.)
- While watching samples on my PC from the scope on the output of U30, these "runts" *only* occur:
  - When pDBIN is NOT active (which would not matter, of course)
  - About 100ns after the rising edge of pDBIN (perhaps the rising edge of the Z80 automatically inserted wait state, TW)
  - On the trailing edge of pDBIN.
  - Very rarely, at 4MHz, on the leading edge of pDBIN (but, as a guess, not all of the address lines at the same time, and maybe not during I/O requests)
  - They happen during both I/O cycles and non I/O read cycles
- Thus, this seems to be partly an artifact of the asynchronous design. If the address line equation were sampled on the leading edge of pDBIN, or during the rising edge of processor cycle T3 (when normal I/O actually occurs), this would likely not be a problem. I expect it ought to be possible (for me) to program the CPLD to behave that way (easiest will probably be to edge trigger the flip flop on sINP . pDBIN with the address line equation as the input.
- Update: Indeed, if I triggered on pDBIN's rising edge, the issue was dealt with …. Except
- I then found that when testing memory (j0,efff) that *that* was triggering TMAXPU. That ought not to be possible – sINP was not going high during that period (it was NOT occurring during the console I/O to check for an ESC character).

So, here is a trace of TMAXPU going active (low) during a “j0,efff” memory test. CH2 has sINP as it appears on the bus. This doesn’t even look like 0.8v to me (and it probably wasn’t).



So, I then added a “probe point” on pin 8 of U21, the 74LS07 that generates the signal fed to U25. It (CH2) is just *barely* reaching into the 0.8v “no man’s land”. The CPLD is being very very touchy. Interestingly, the j0,efff command *does* do I/O – it checks for the operator pressing ESC to terminate the test. Curiously, if I put a probe point right on U25’s pin, the problem disappears completely.



STOP 1.000GSa/s 1.40k pts H 100.0ns D 0.00000000s

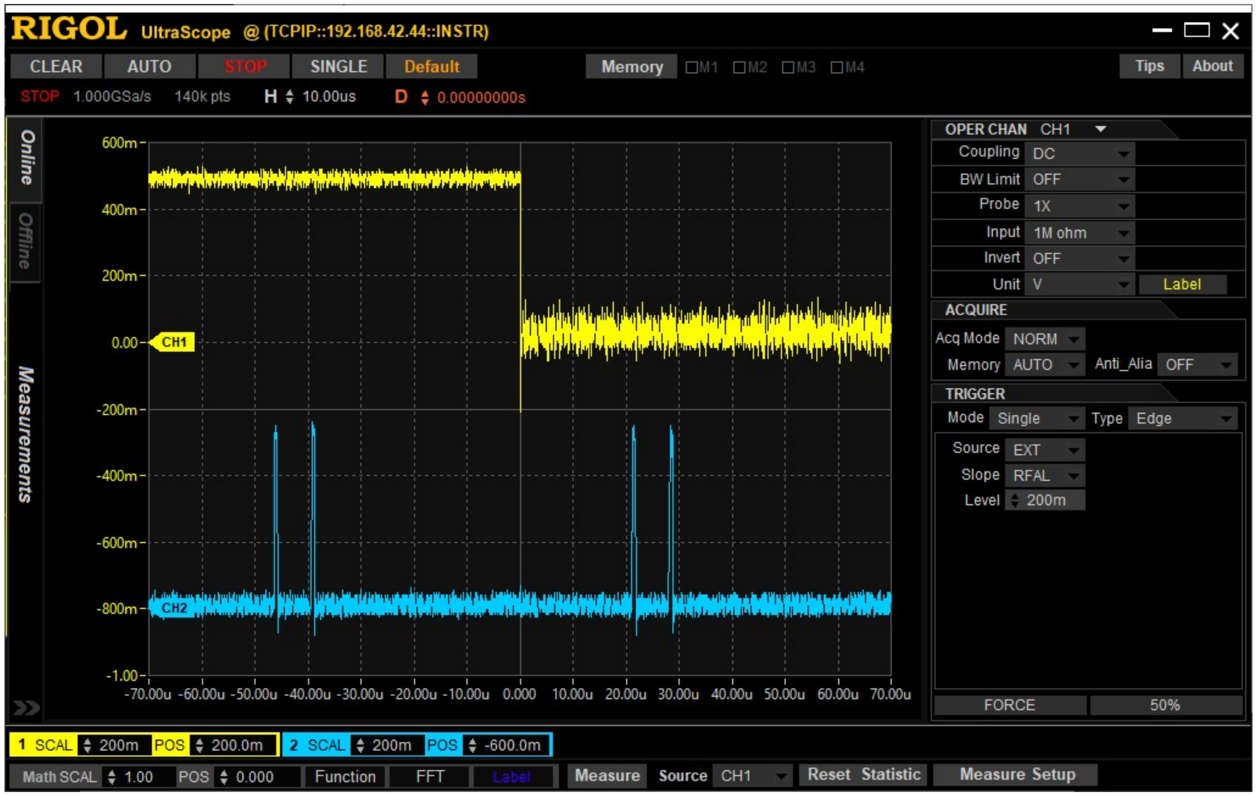
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source EXT
Slope RFAL
Level 200m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Interestingly, this issue seems to happen, when it happens, consistently a bit under 40us after the last real I/O (an eternity, really) (and not at any consistent point in the test itself)



STOP 1.000GSa/s 140k pts H 10.00us D 0.00000000s

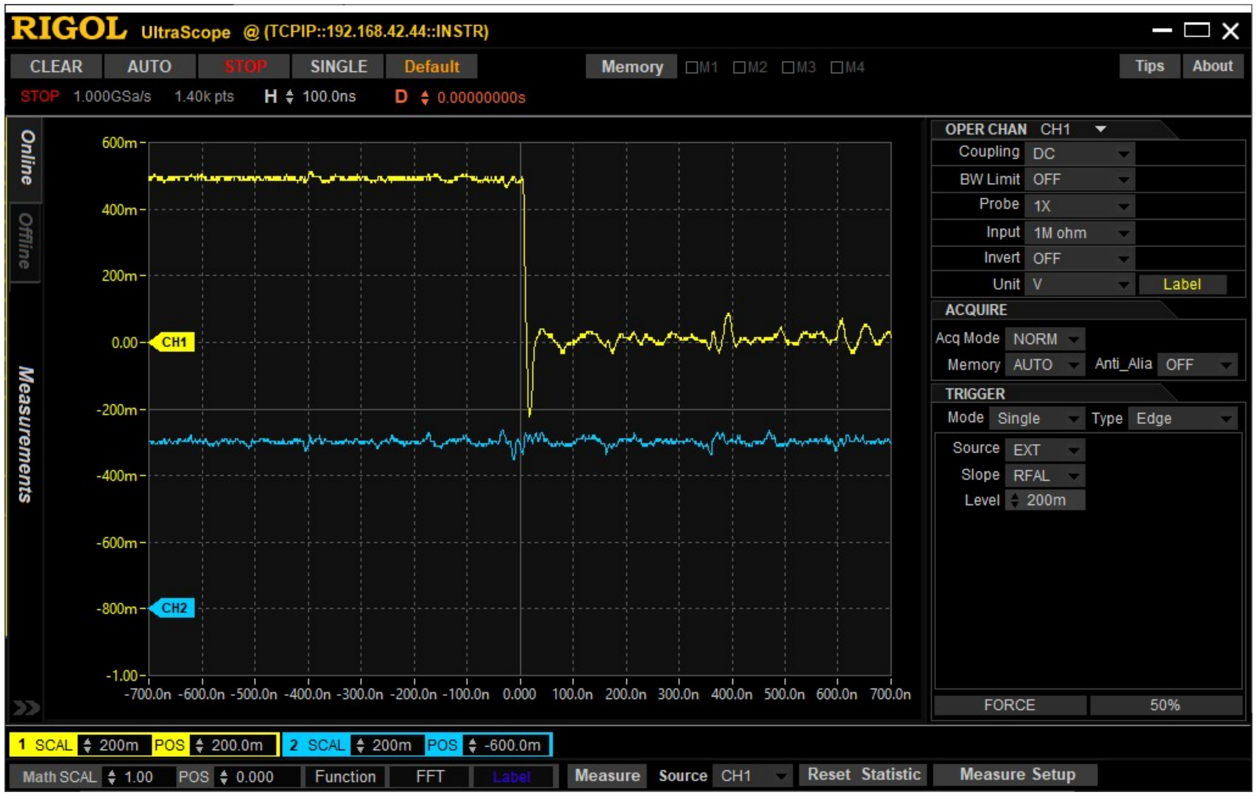
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope RFAL
Level 200m

1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

This made me a little suspicious. Could this be a noise immunity issue caused by noise on the +5V rail? So, I 'scoped the rail (with a probe point right at one of U25's bypass capacitors) during this new issue. Note the dip, of about 0.5v, right when this happens.


RIGOL UltraScope @ (TCPIP::192.168.42.44::INSTR)

STOP 1.000GSa/s 1.40k pts H 100.0ns D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope RFAL
Level 200m

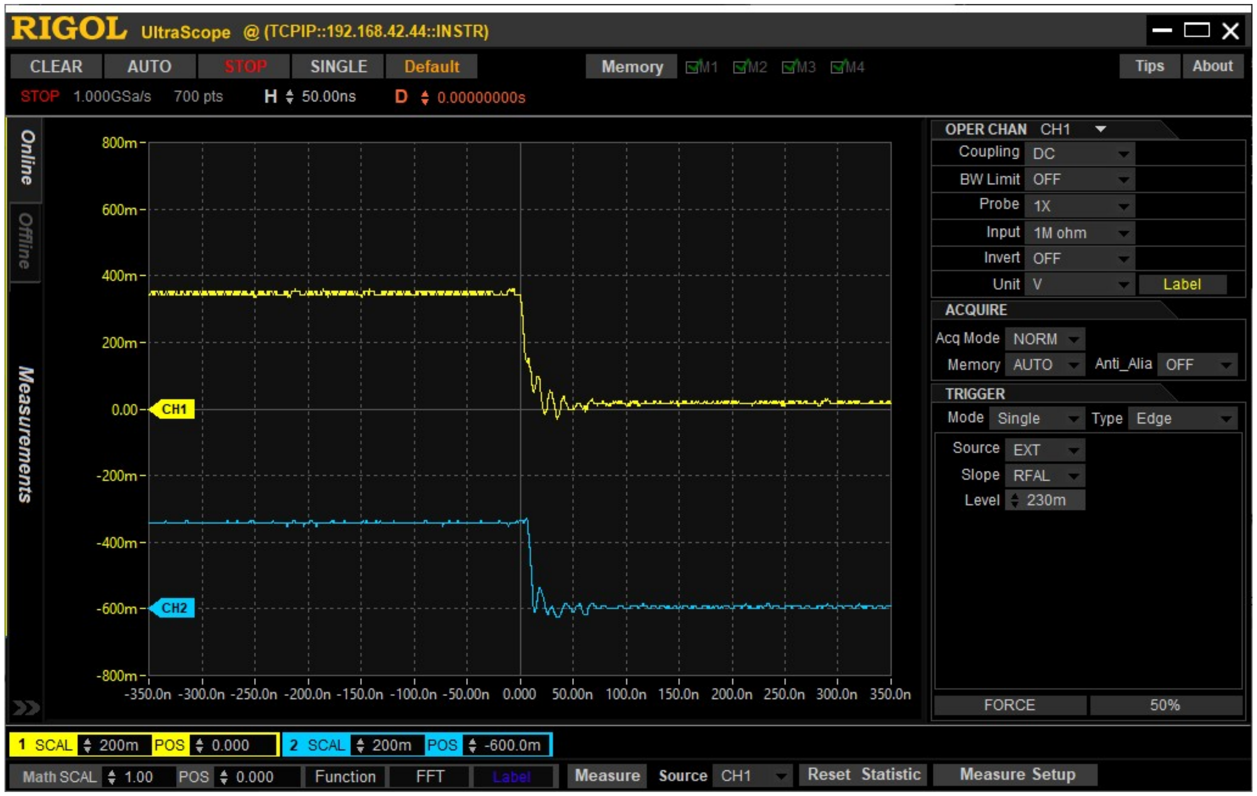
1 SCAL 200m POS 200.0m
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

So, I paralleled the 0.1uf caps on the back side, but that did help. Neither did 10uf. Nor did a .01uf. The 8v line also has a corresponding drop. The drop is about 10ns long (so, about 100Mhz), so no wonder adding 10uf did not make any difference. Later I tried adding .0047 and an additional 10uf at the regulator. That actually turned it into a little positive going bump – but the board still activated.

PostScript: This issue went away when I removed the Godbout active terminator.

There is a second issue associated with the active terminator – but I do not believe that is where the blame lies. With the active terminator in place, output from the PDP11V2 board to the Propeller board comes out blanks.

Here is a screenshot with the active terminator in place. The trigger is D* brought out to P26 on PDP11V2 U15 – so any odd byte write. I then entered a BOE001,41 command to the PDP11V2 monitor, running on the UART. The top signal, CH1, is pWR* on the S100 bus. The bottom signal is D2 sampled at PDP11V2 U20 pin 13. Note that the pWR* is going (or at least starting to go) active low well BEFORE the data line reaches its desired state. The horizontal is 50ns/div, so it is maybe 50ns to 60ns early. Note that the baseline for the data in high impedance state is on the order of 2.6v where it should be with the active terminator in place.



STOP 1.000GSa/s 700 pts H 50.00ns D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

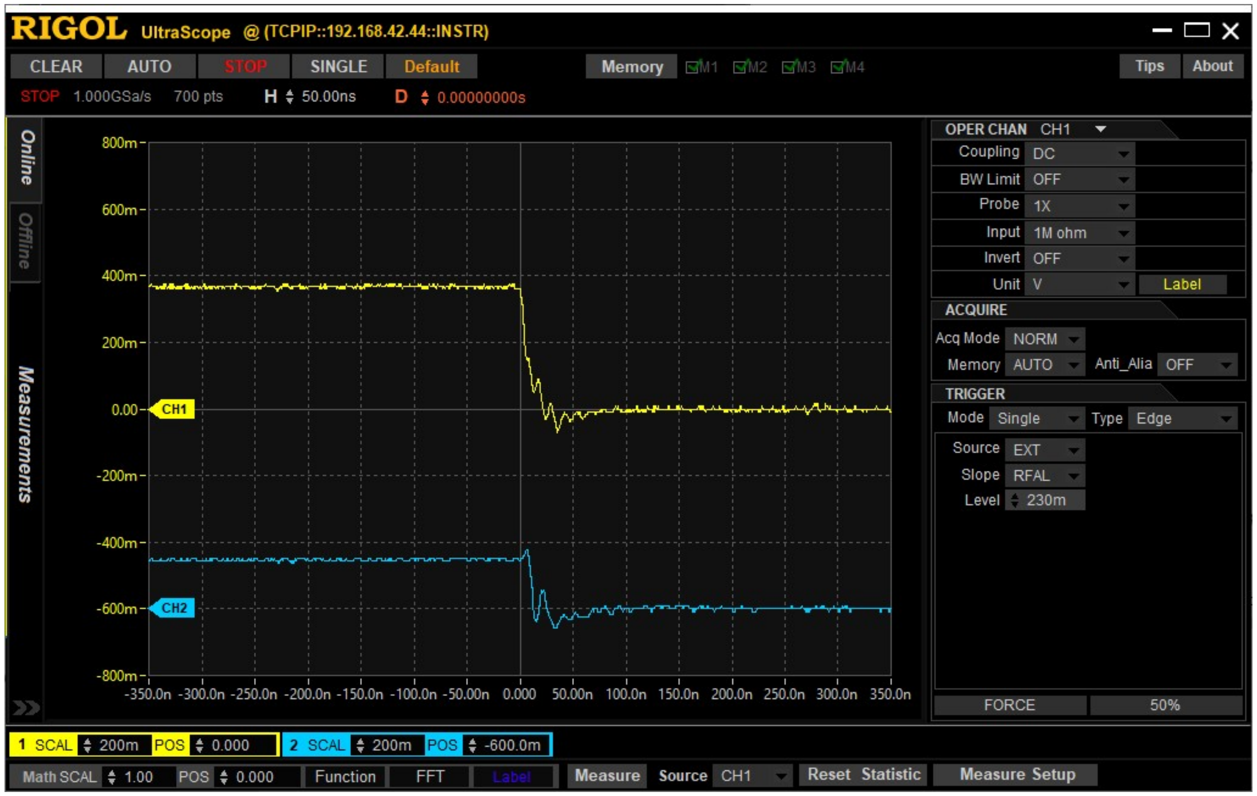
ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope RFAL
Level 230m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Here is the same setup, but with the active terminator removed. In this situation the correct output is observed, but that is likely simply because the data waveform has a baseline starting at 1.6v – closer to 0. pWR* still has not given the data time to stabilize. It probably works only because of a couple of gate delays on the Propeller board give it just enough time.

This is not conformal. The IEEE 696 spec for a write cycle specifies a tDWR time (setup time DO valid BEFORE pWR* goes low) of 0.1tCY. tCY is one system clock cycle. At 4Mhz that would mean that the data should be stable for at least 0.1 * 250ns, or 25ns, BEFORE pWR* goes low.



STOP 1.000GSa/s 700 pts H 50.00ns D 0.00000000s

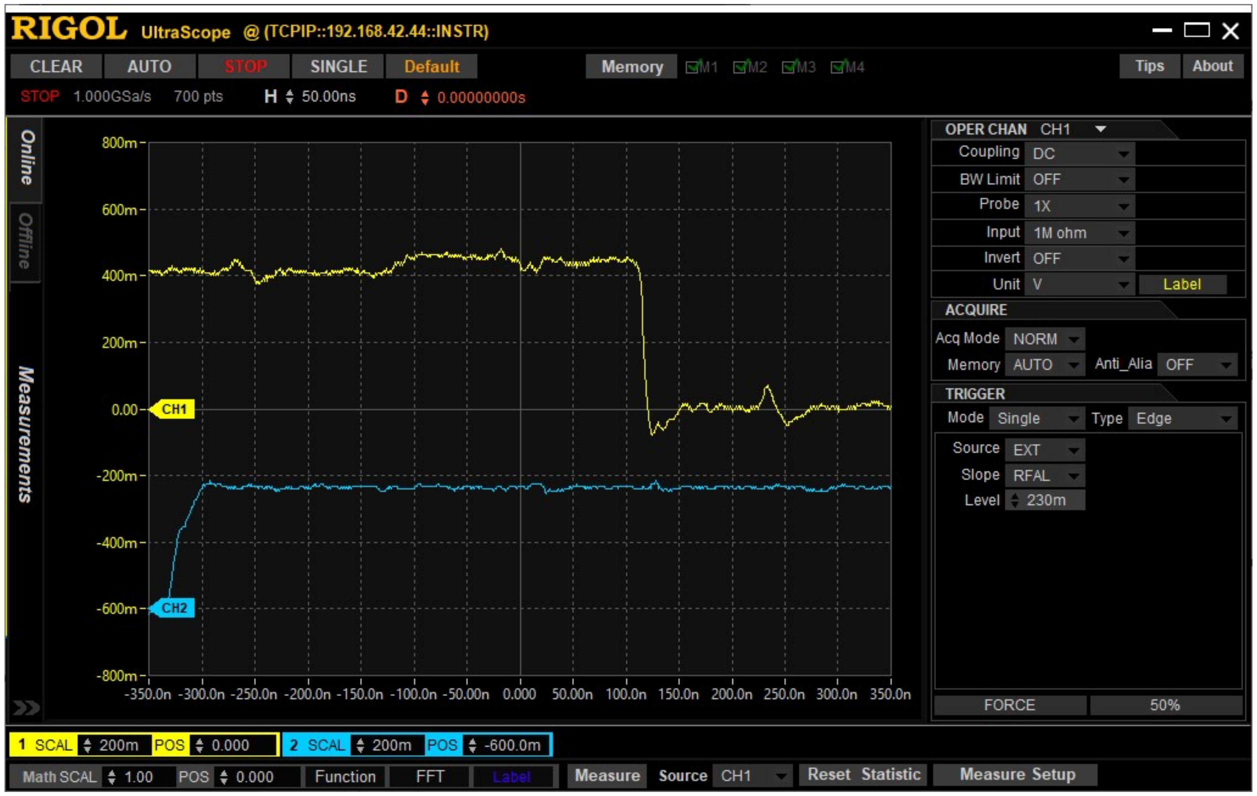
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source EXT
Slope RFAL
Level 230m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Finally, here is a picture of the Z80 outputting to the same port (the character is "A") (CH2). CH1 is the pWR* signal for the I/O (pin 77). It comes slightly after the trigger, which is sOUT (pin 45).



STOP 1.000GSa/s 700 pts H 50.00ns D 0.00000000s

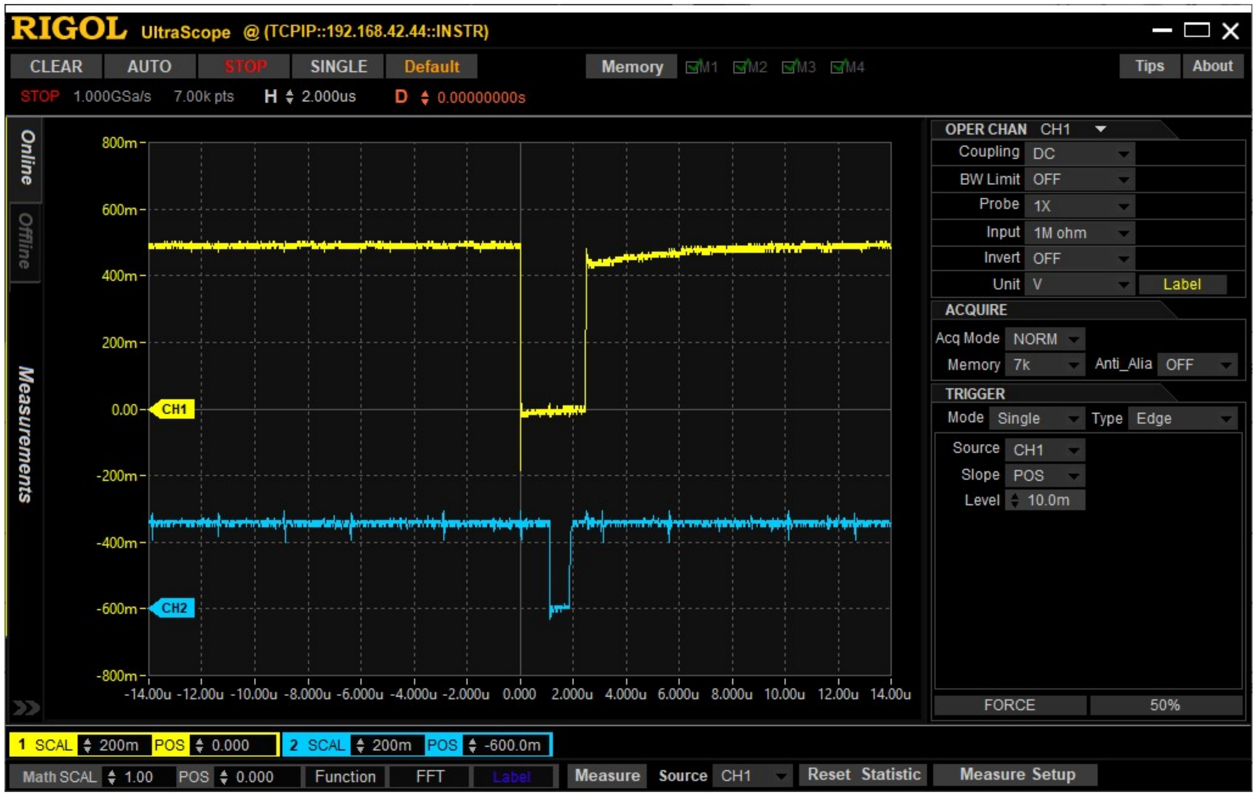
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source EXT
Slope RFAL
Level 230m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

The idea I had is to remove the signal bpWR* from the equation for OE_C, OE_D and OE_B.   Here is a trace of what that would look like. CH1 (and the trigger) is the proposed new signal. CH2 is the output data – plenty of setup time.



STOP 1.000GSa/s 7.00k pts H 2.000us D 0.00000000s

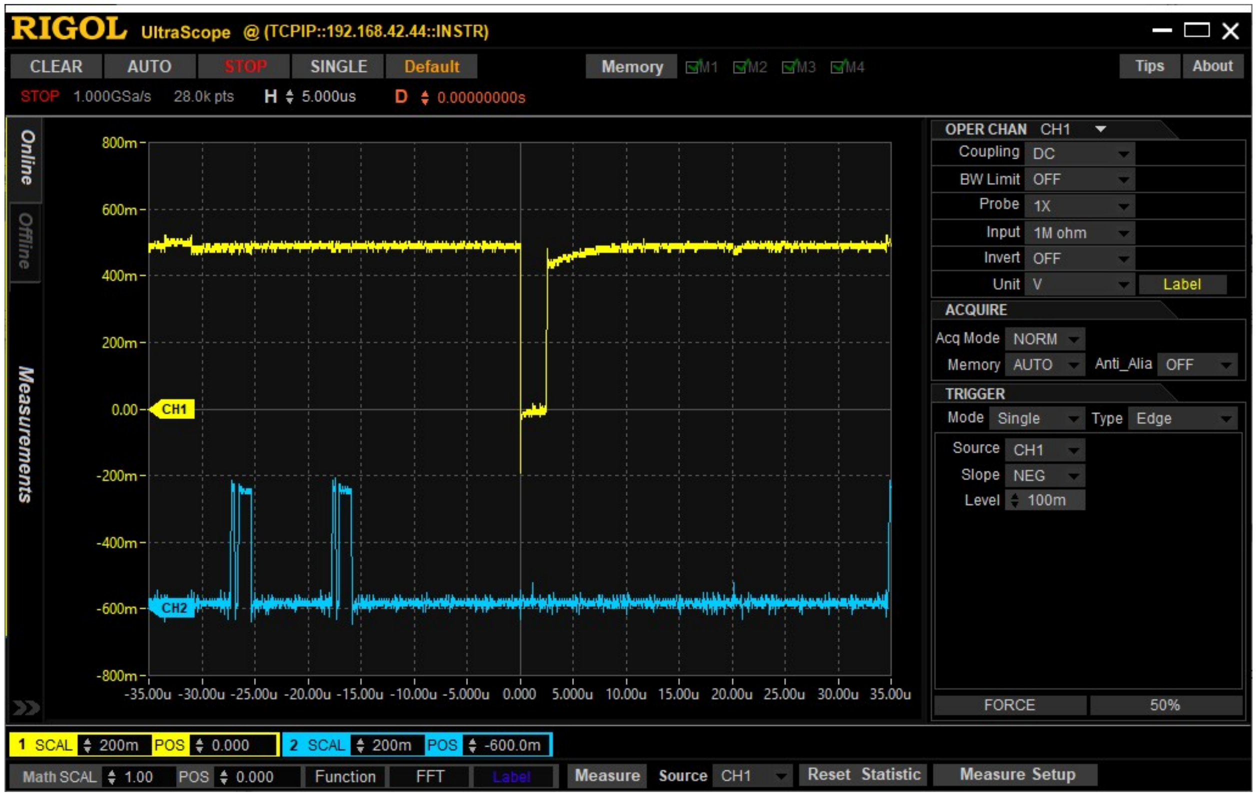
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory 7k
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH1
Slope POS
Level 10.0m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

And here is the new signal compared to pDBIN (CH2). It shows no danger at all of this new signal causing bus contention by overlapping with pDBIN.



STOP 1.000GSa/s 28.0k pts H 5.000us D 0.00000000s

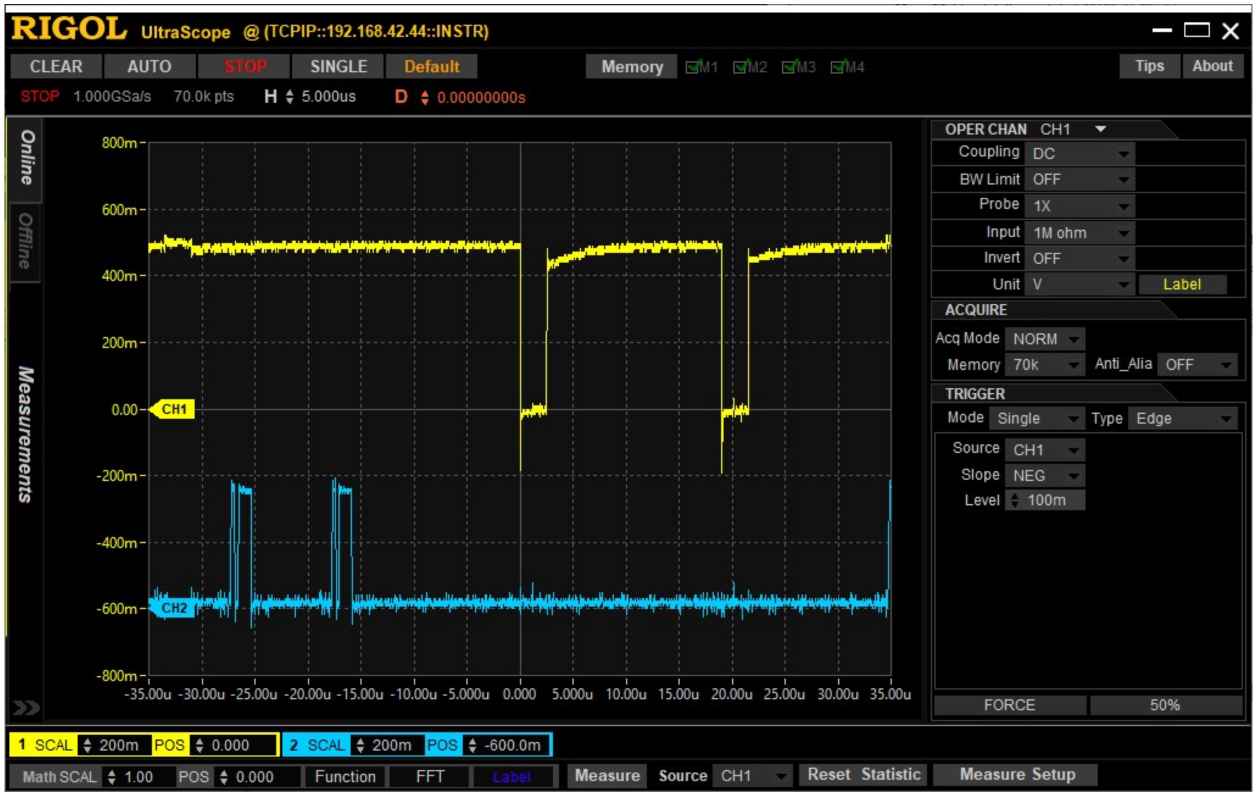
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH1
Slope NEG
Level 100m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

I then changed the test signal to enable the new scheme for both byte and word/even writes. This trace shows the new signal on CH1 compared to pDBIN:



STOP 1.000GSa/s 70.0k pts H 5.000us D 0.00000000s

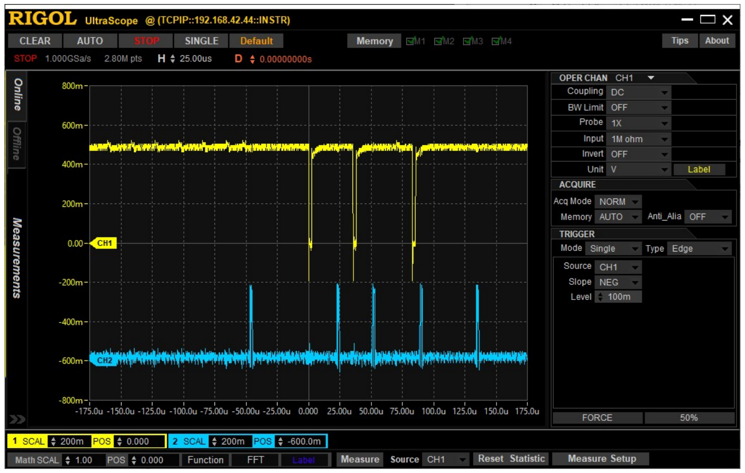
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory 70k
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH1
Slope NEG
Level 100m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

Here are a couple of traces on a much longer timebase (25us). The first was at a different spot in the code. The second is from the output to E001. There is clearly no danger of overlap between enabling the PDP11 output buffers (active low, CH1, also the trigger) and pDBIN (active high, CH2).



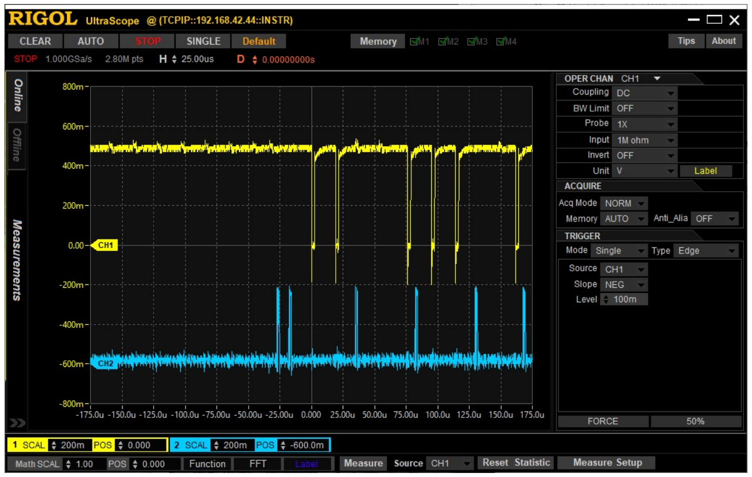
STOP 1.000GSa/s 2.80M pts H 25.00us D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source CH1
Slope NEG
Level 100m

1 SCAL 200m POS 0.000 2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000 Source CH1



STOP 1.000GSa/s 2.80M pts H 25.00us D 0.00000000s

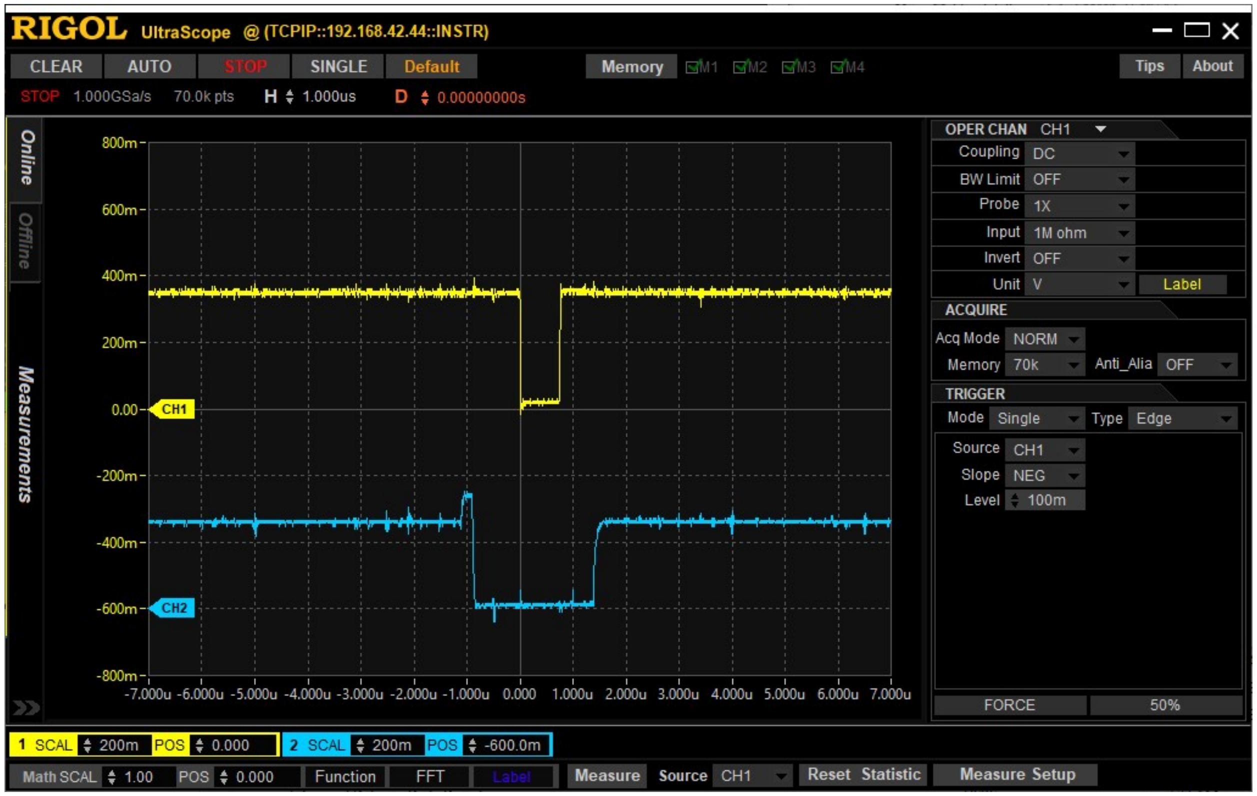
OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO Anti_Alia OFF

TRIGGER
Mode Single Type Edge
Source CH1
Slope NEG
Level 100m

1 SCAL 200m POS 0.000 2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000 Source CH1

I then removed the term !bpWR from the equations for !OE_C, !OE_D and !OE_B (it was not present for OE_A, which is a read cycle). I have not removed it from bsXTRQ. Here is what it looked like. CH1 is S100 pWR* and CH2 is the data on pin13 of U20. Note the little positive excursion that happens in the data initially. That matters not at all – it is during the time when the data has not yet been set up by the J11:


RIGOL UltraScope @ (TCPIP::192.168.42.44::INSTR)

STOP 1.000GSa/s 70.0k pts H 1.000us D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

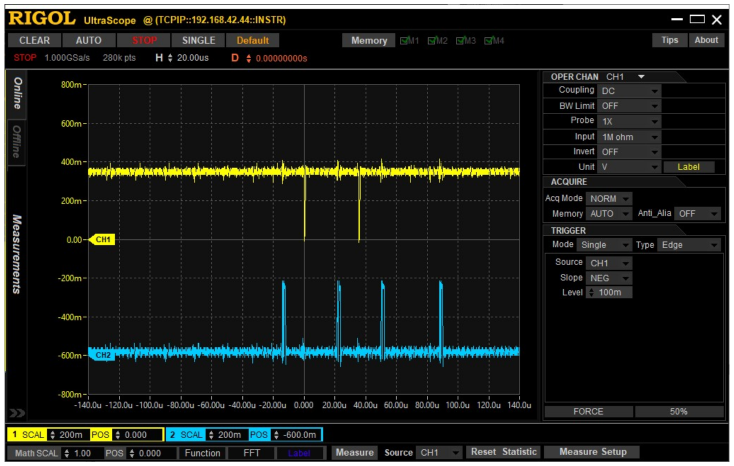
ACQUIRE
Acq Mode NORM
Memory 70k
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH1
Slope NEG
Level 100m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

And, indeed, the correct output is observed.

Here are pWRT* and pDBIN on a much longer time scale. As before, the first trace is while entering “BOE001,41” (but before hitting Enter), and the second is upon pressing Enter. Several machine cycles are show, clearly illustrating no danger of conflict on the bus for the data lines:


RIGOL UltraScope @ (TCPIP::192.168.42.44::INSTR)

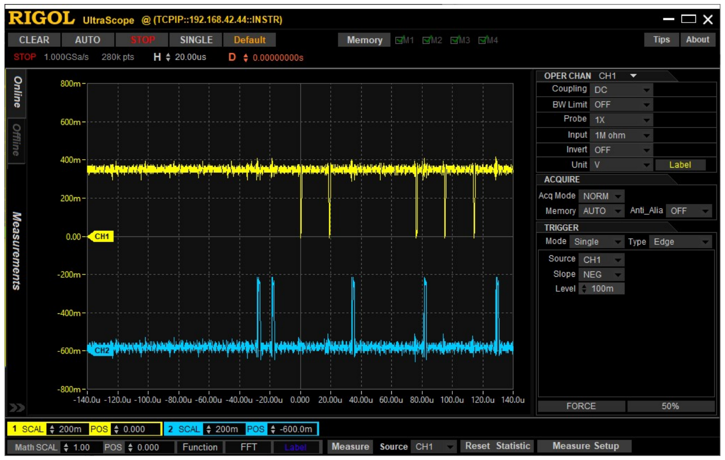
STOP 1.000GSa/s 280k pts H 20.00us D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH1
Slope NEG
Level 100m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1


RIGOL UltraScope @ (TCPIP::192.168.42.44::INSTR)

STOP 1.000GSa/s 280k pts H 20.00us D 0.00000000s

OPER CHAN CH1
Coupling DC
BW Limit OFF
Probe 1X
Input 1M ohm
Invert OFF
Unit V

ACQUIRE
Acq Mode NORM
Memory AUTO
Anti_Alia OFF

TRIGGER
Mode Single
Type Edge
Source CH1
Slope NEG
Level 100m

1 SCAL 200m POS 0.000
2 SCAL 200m POS -600.0m
Math SCAL 1.00 POS 0.000
Source CH1

The code changes were:

```
!OE_C          =  ((( /* !bpWR & */ !BUS_WORD_WRITE & !XFERII & !bsINTA) # (bpDBIN & !XFERII & !bsINTA))      /* U17,  16 Bit Rd or /Wr and 8 bit Rd, DAL 8-15 */

!OE_D          =  /* !bpWR & */ !BUS_BYTE_WRITE &  LA0  & !XFERII  & !bsINTA;                 /* U20, 8 Bit Write Odd address on DAL 8-15 */

!P26 = (!BUS_BYTE_WRITE # !BUS_WORD_WRITE) & !XFERII & !bsINTA;      /* TESTING data enable without !bpWR */

!OE_B          =  ((( /* !bpWR & */ !BUS_WORD_WRITE & !XFERII & !bsINTA)  # (bpDBIN & !XFERII & !bsINTA & !SIXTN))   /*   U23,  16 Bit Rd or Wr and 8 bit (RAM) Rd */

           # ( /* !bpWR & */ !BUS_BYTE_WRITE & !LA0  & !XFERII & !bsINTA));                    /*   8 Bit Write Even address (DAL 0-7) */
```