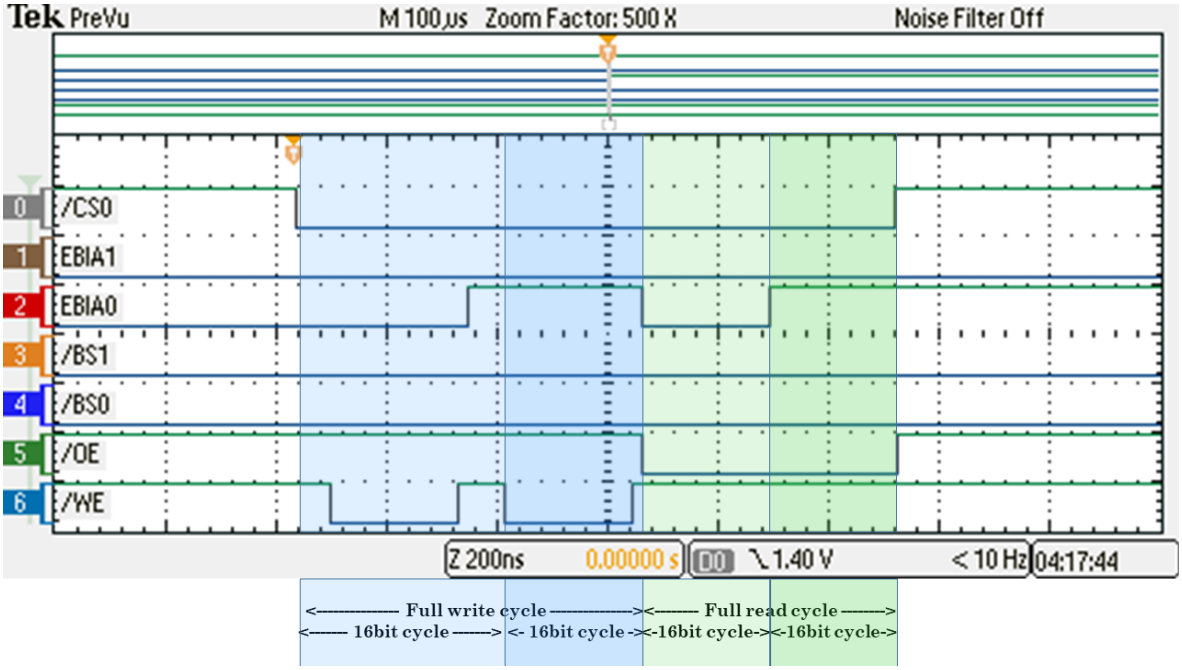
Zak Rowland
CST331
Lab 3 – EBI Intro
Conclusions

1.



The first 16-bit cycle writes the half word at A1, A0 = 0X. The second 16-bit cycle writes the half word at A1, A0 = 1X. The third 16-bit cycle reads the half word at A1, A0 = 0X. The final 16-bit cycle reads the half word at A1, A0 = 1X. Address line A0 is not outputted in this configuration, so A1 and A2 are tied to EBIA0 and EBIA1 respectively.

2. a.

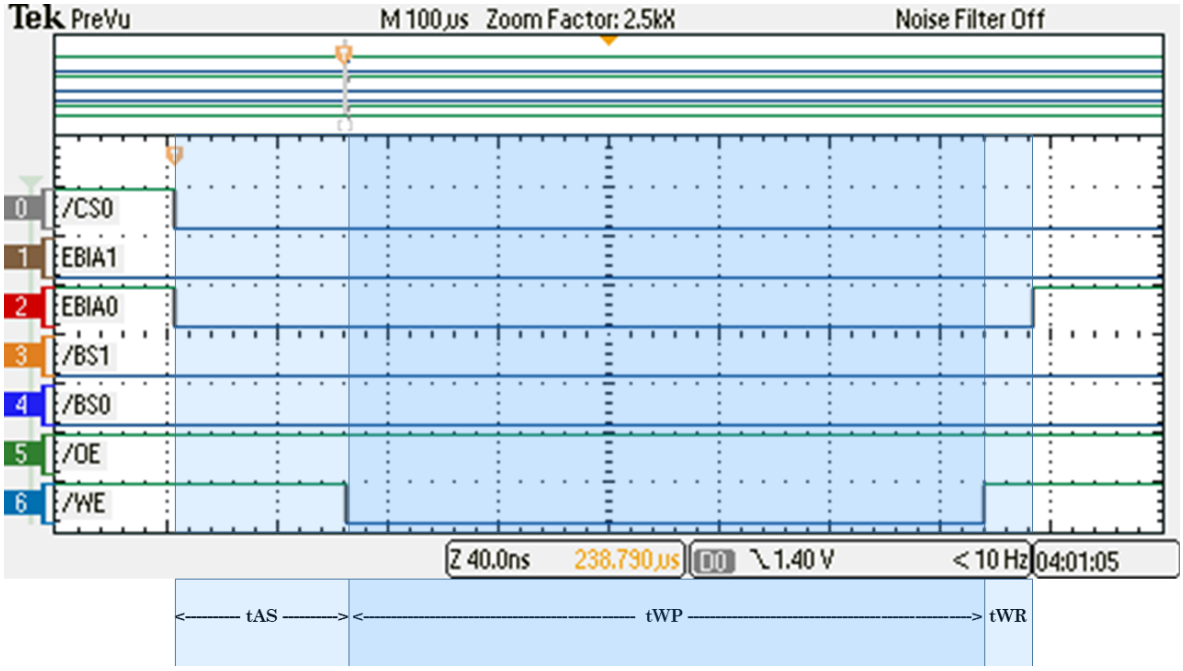
Parameter:	tRC	tAS	tWP	tWR
Value:	11 cycles	3 cycles	11 cycles	1 cycle

b.

Write cycle 1:

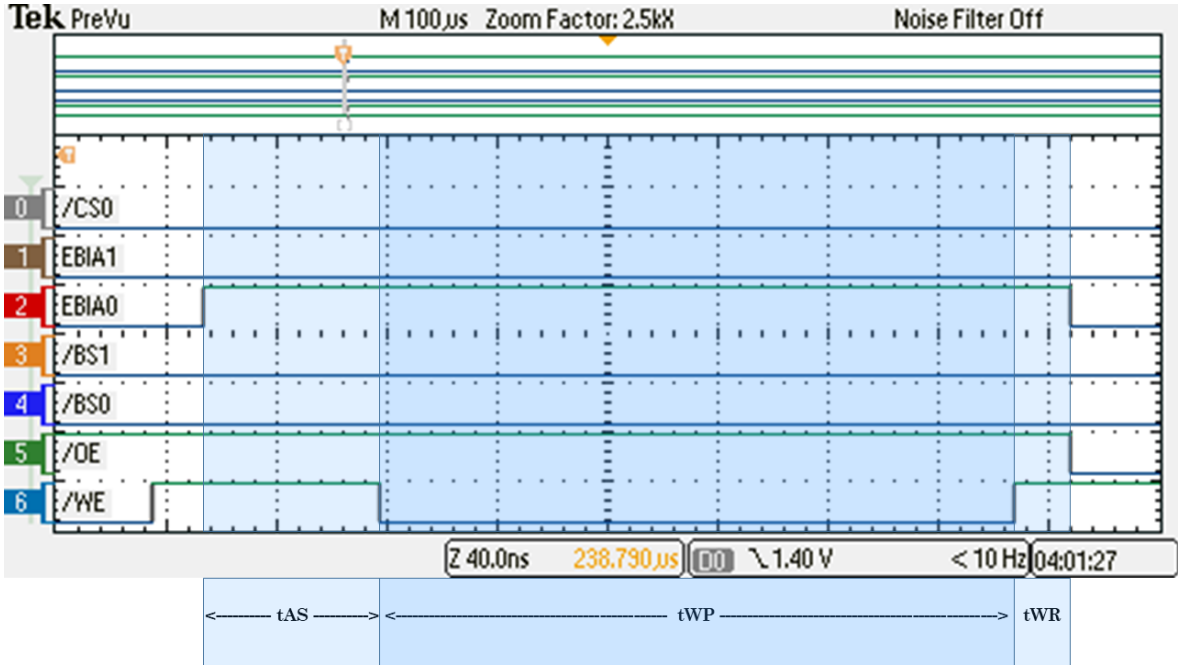
Parameter:	tRC	tAS	tWP	tWR
Act. Time:	N/A	62ns	230ns	18ns

Zak Rowland
CST331
Lab 3 – EBI Intro
Conclusions



Write cycle 2:

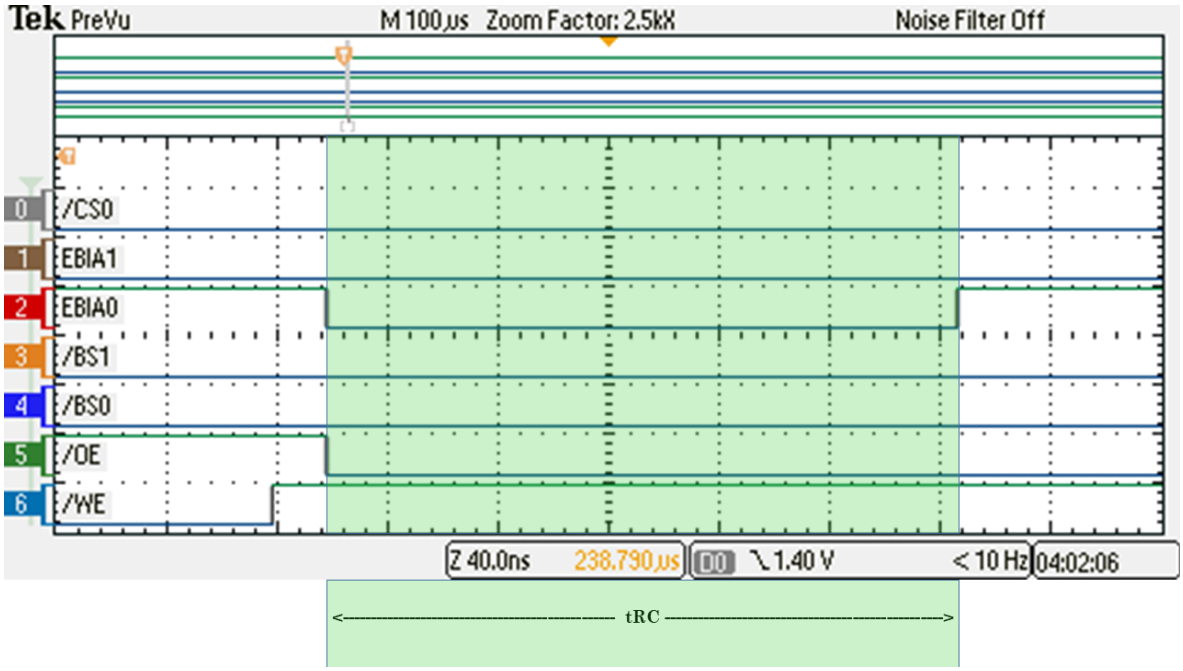
Parameter:	tRC	tAS	tWP	tWR
Act. Time:	N/A	64ns	230ns	20ns



Zak Rowland
CST331
Lab 3 – EBI Intro
Conclusions

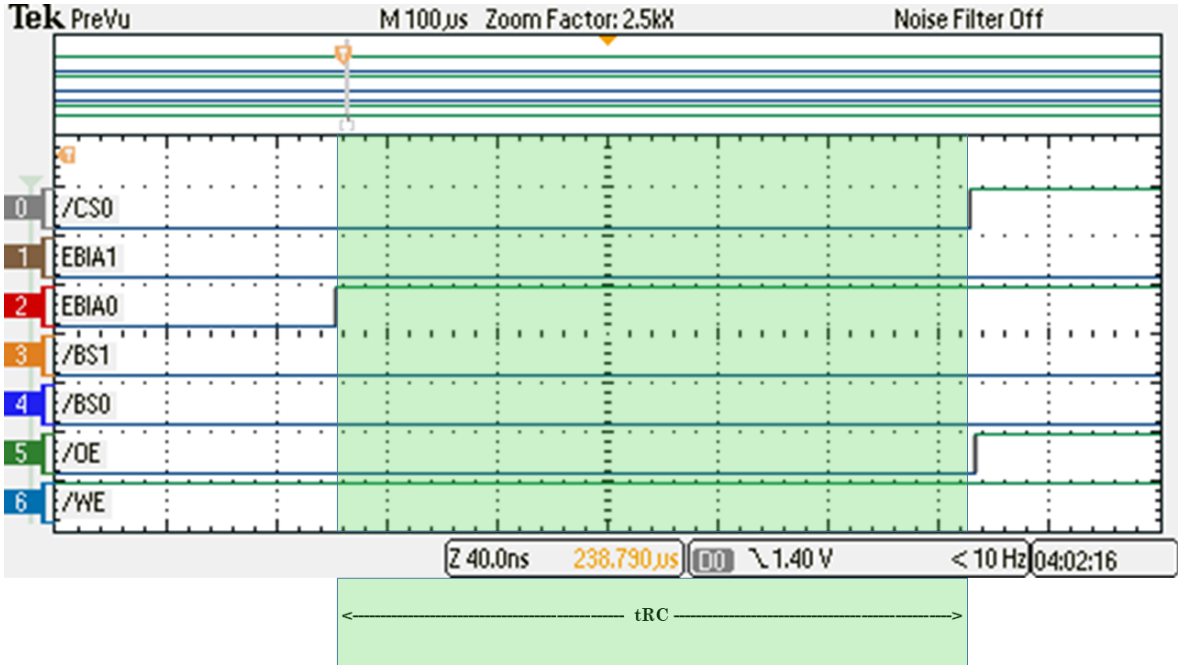
Read cycle 1:

Parameter:	tRC	tAS	tWP	tWR
Act. Time:	228ns	N/A	N/A	N/A



Read cycle 2:

Parameter:	tRC	tAS	tWP	tWR
Act. Time:	230ns	N/A	N/A	N/A

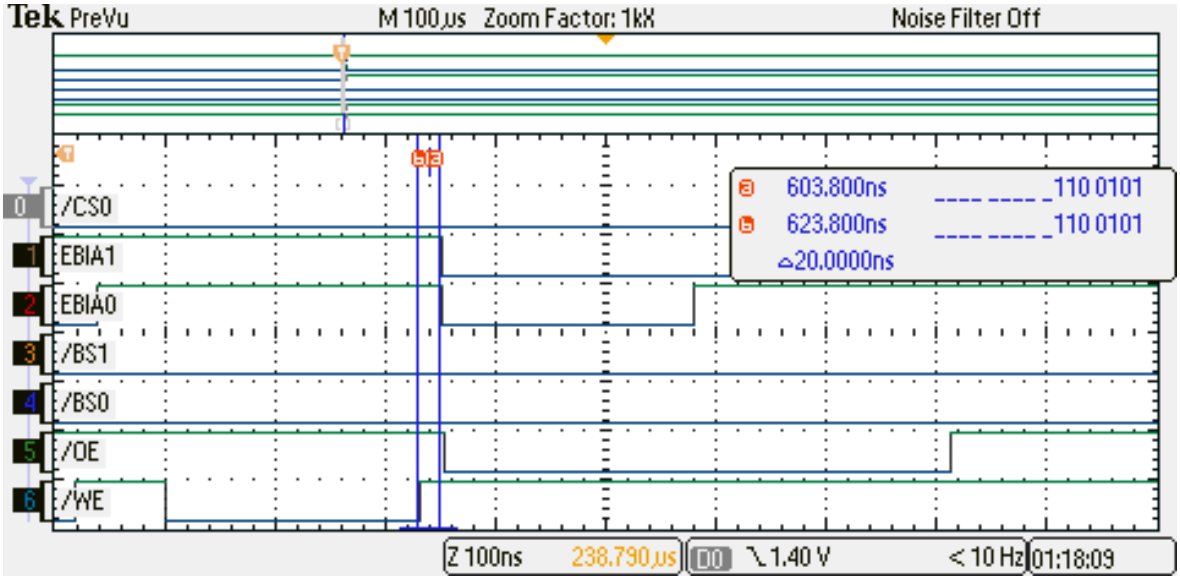


c. I will use the tRC parameter to determine the clock. The device is configured for 11 cycles for tRC and this parameter was measured to be 228ns. 228ns / 11 cycles is equal to 20.72ns per cycle. 20.72ns is equivalent to a 48.26MHz clock. Since the system clock was configured to operate at 96MHz and the PBCLK8 divider was left at the default of 2, the EBI is operating from PBCLK8 which is 48MHz.

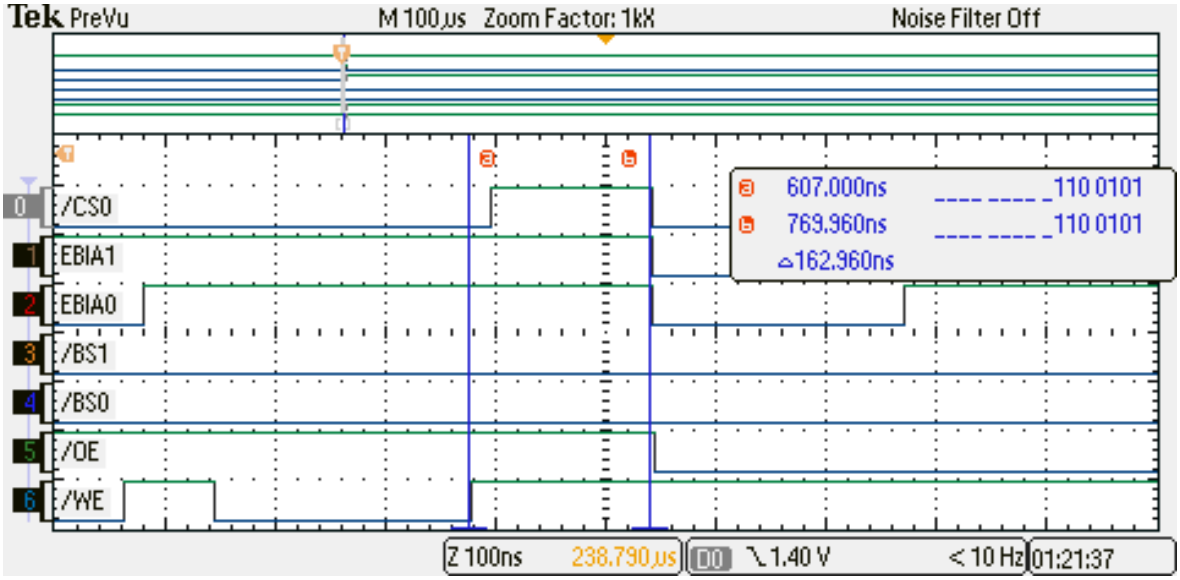
3. The time between the end of the write cycle and the start of the read cycle is 20ns when tBTA is set to 0. When tBTA is set to 7, the time between the end of the write cycle and the start of the read cycle is 162.96ns. The time between the write and read cycles increased by 142.96ns. It appears that the cycles before were back to back (besides tWR) because CS0 never has to negate then assert again. After comparing the screenshots, it appears that the time between the two 16-bit write cycles doesn't change. This is expected because control of the bus isn't changing at this time.

Zak Rowland
CST331
Lab 3 – EBI Intro
Conclusions

tBTA = 0:

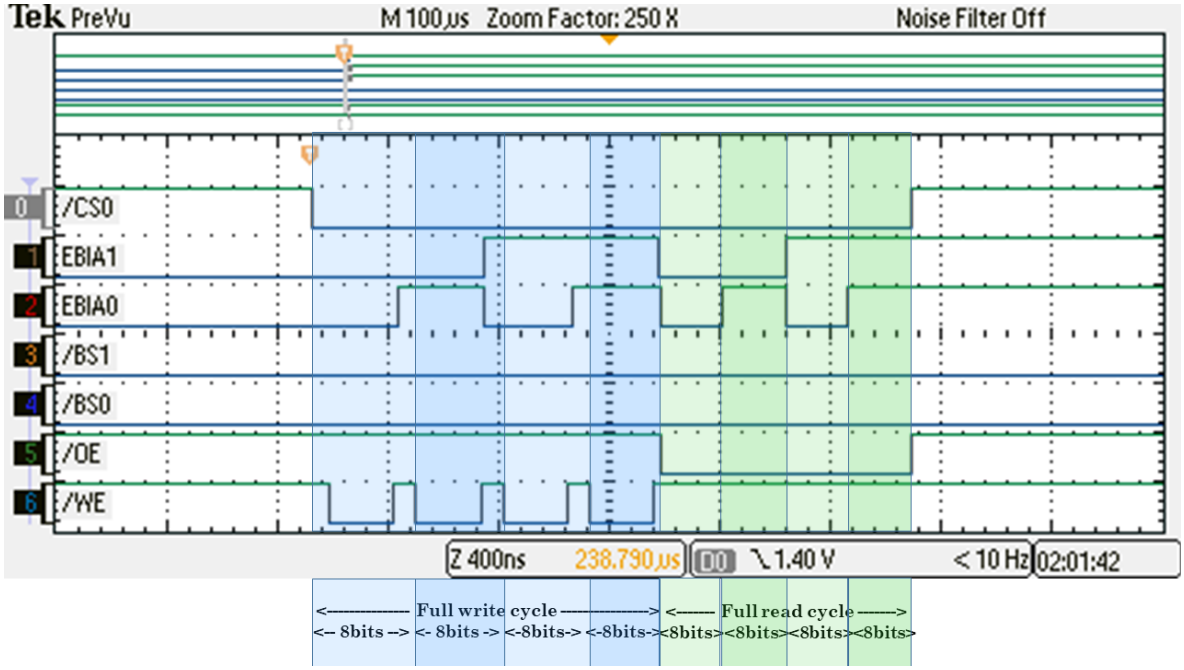


tBTA = 7:



Zak Rowland
CST331
Lab 3 – EBI Intro
Conclusions

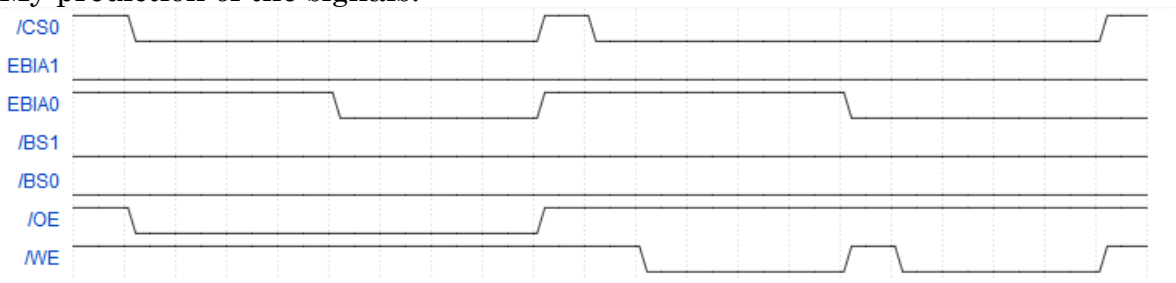
4.



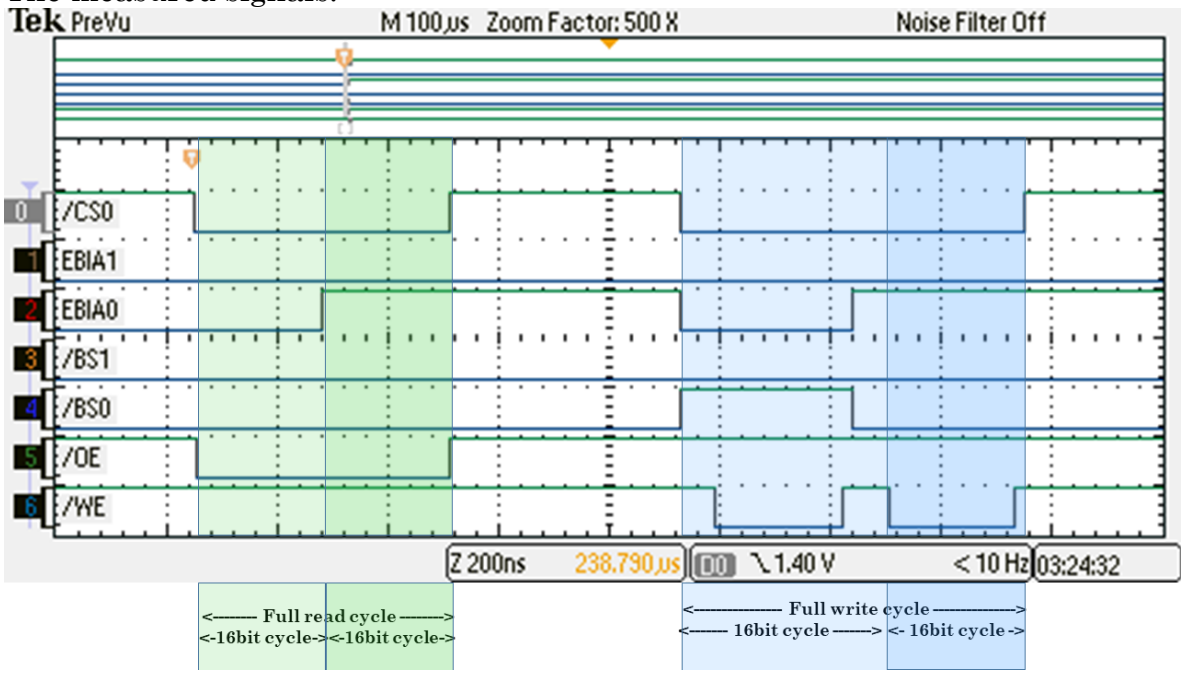
The first 8-bit cycle writes the byte at A1, A0 = 00. The second 8-bit cycle writes the byte at A1, A0 = 01. The third 8-bit cycle writes the byte at A1, A0 = 10. The fourth 8-bit cycle writes the byte at A1, A0 = 11. The fifth 8-bit cycle reads the byte at A1, A0 = 00. The sixth 8-bit cycle reads the byte at A1, A0 = 01. The seventh 8-bit cycle reads the byte at A1, A0 = 10. The final 8-bit cycle reads the byte at A1, A0 = 11. In this configuration, the EBI treats A1 and A0 as byte selection which means bank select isn't used.

5.

My prediction of the signals:



The measured signals:

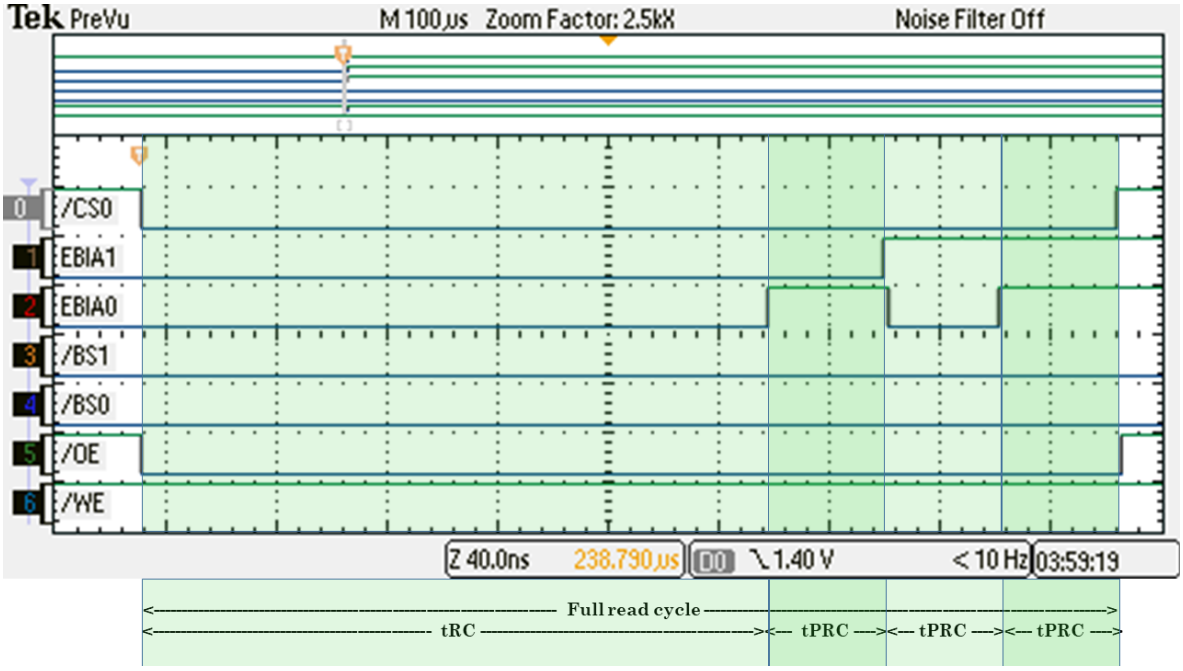


The first 16-bit cycle reads the half word at A1, A0 = 0X. The second 16-bit cycle reads the half word at A1, A0 = 1X. The third 16-bit cycle writes the byte at A1, A0 = 0X. The final 16-bit cycle writes the half word at A1, A0 = 1X. The inconsistency I noticed is that it appears the read cycle still reads a full word (doesn't use bank select) unlike the write cycle which begins by reading a single byte followed by a half word.

6.

Read cycle:

Parameter:	tRC	tPRC1	tPRC2	tPRC3
Time:	226ns	42.36ns	42.04ns	42.08ns

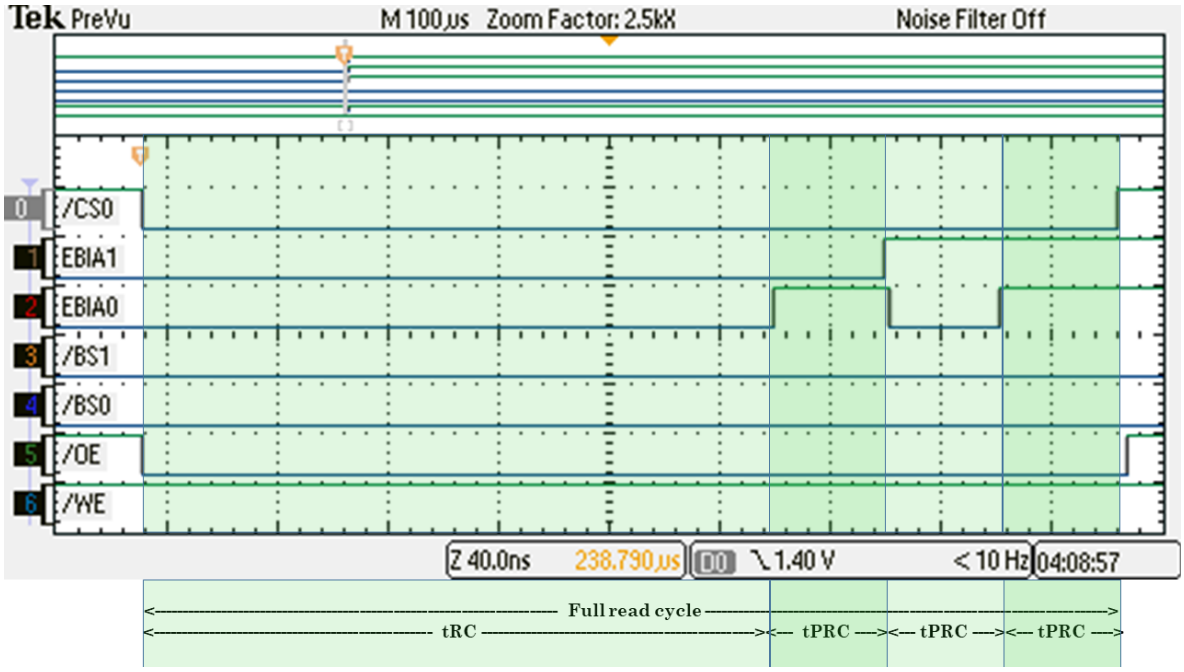


Settings:

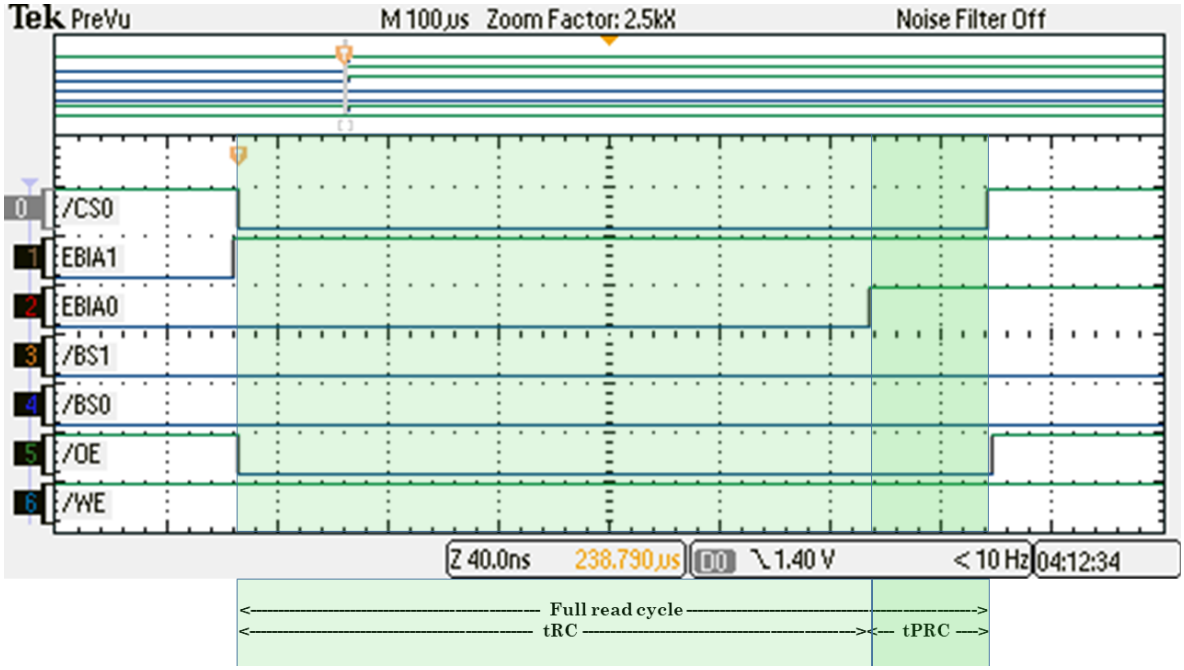
Parameter:	tRC	tPRC
Value:	11 cycles	2 cycles

7.

asm("lwr %0, 0xE0000001":"=r"(value));



asm("lwr %0, 0xE0000002":"=r"(value));



A pagemode access (after an initial tRC) was started for both A1, A0 = 01 and A1, A0 = 10. The first configuration reads a byte after tRC has passed, then reads three more bytes after three more tPRC periods. The second configuration reads a byte after tRC has passed, then reads the final byte after one tPRC period. The first configuration should only read 3 bytes, like step 5, however it still reads a full word. I believe this happens because the EBI isn't actually connected to anything.