

Memory Interface

TramelBlaze

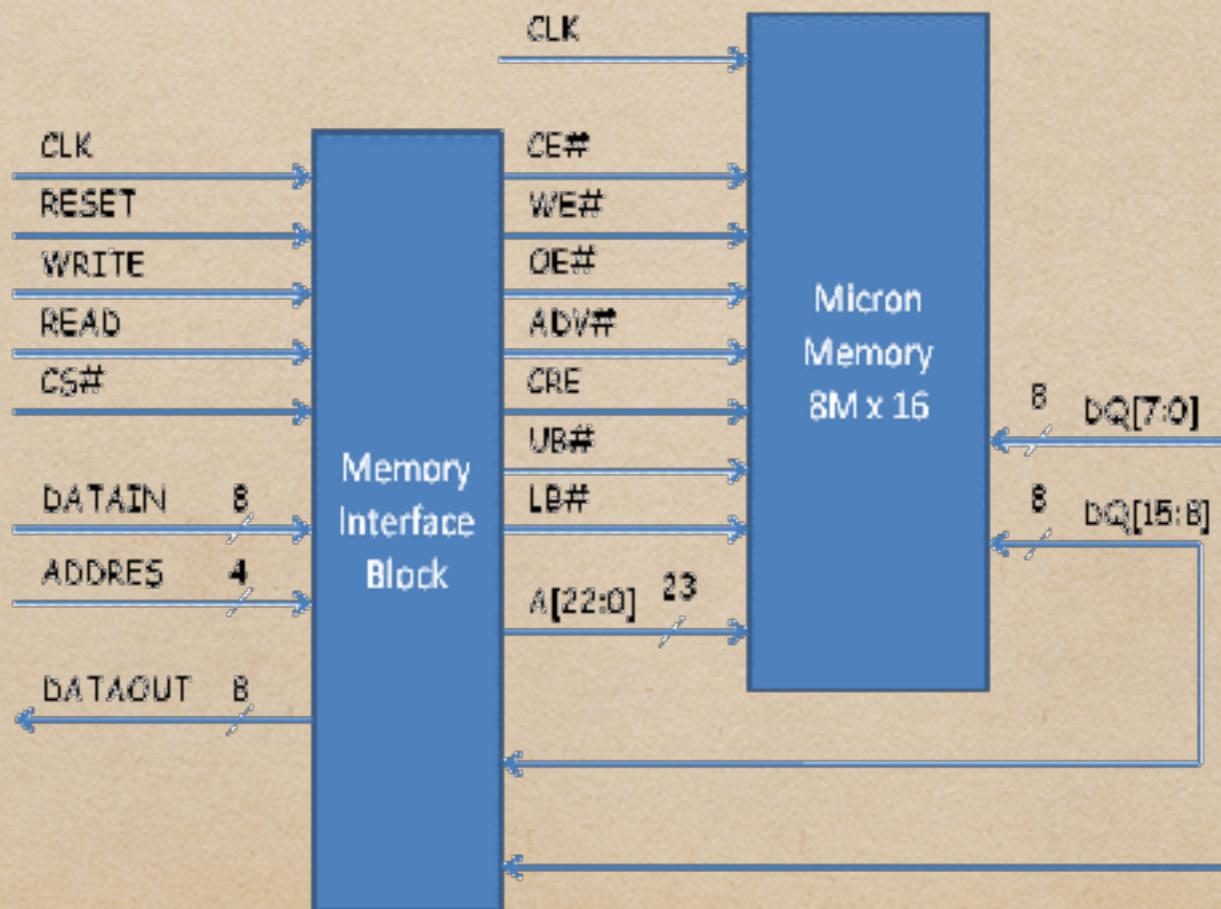
CECS 460

CSULB

John Tramel

Where we want to go

- ◆ The final implementation looks like the block diagram below. Our MIB will communicate with the Micron Memory located on our board
- ◆ Although the device is a synchronous device, when it is powered up it defaults to an asynchronous mode - that is what we will use



Memory Interface Signals

Signal	Width	In/Out	Description
CLK	1	Input	System Input
RESET	1	Input	System Input
WRITE	1	Input	Write Pulse to write to Memory Interface Block
READ	1	Input	Read Pulse to read from Memory Interface Block
CS#	1	Input	Chip Select to select Memory Interface Block
DATAIN	8	Input	Data to Memory Interface Block
ADDRESS	4	Input	Address to Memory Interface Block
DATAOUT	8	Output	Data Back to Processor
CE#	1	Output	Chip Select of Micron Memory
WE#	1	Output	Write Enable to Micron Memory
OE#	1	Output	Output Enable to Micron Memory
ADV#	1	Output	Always LOW
CRE	1	Output	Always LOW
UB#	1	Output	Upper Byte Enable
LB#	1	Output	Lower Byte Enable
A	23	Output	Address to Micron Memory
DQ	16	Inout	Bidirectional Data to/from Micron Memory

← Always '0'

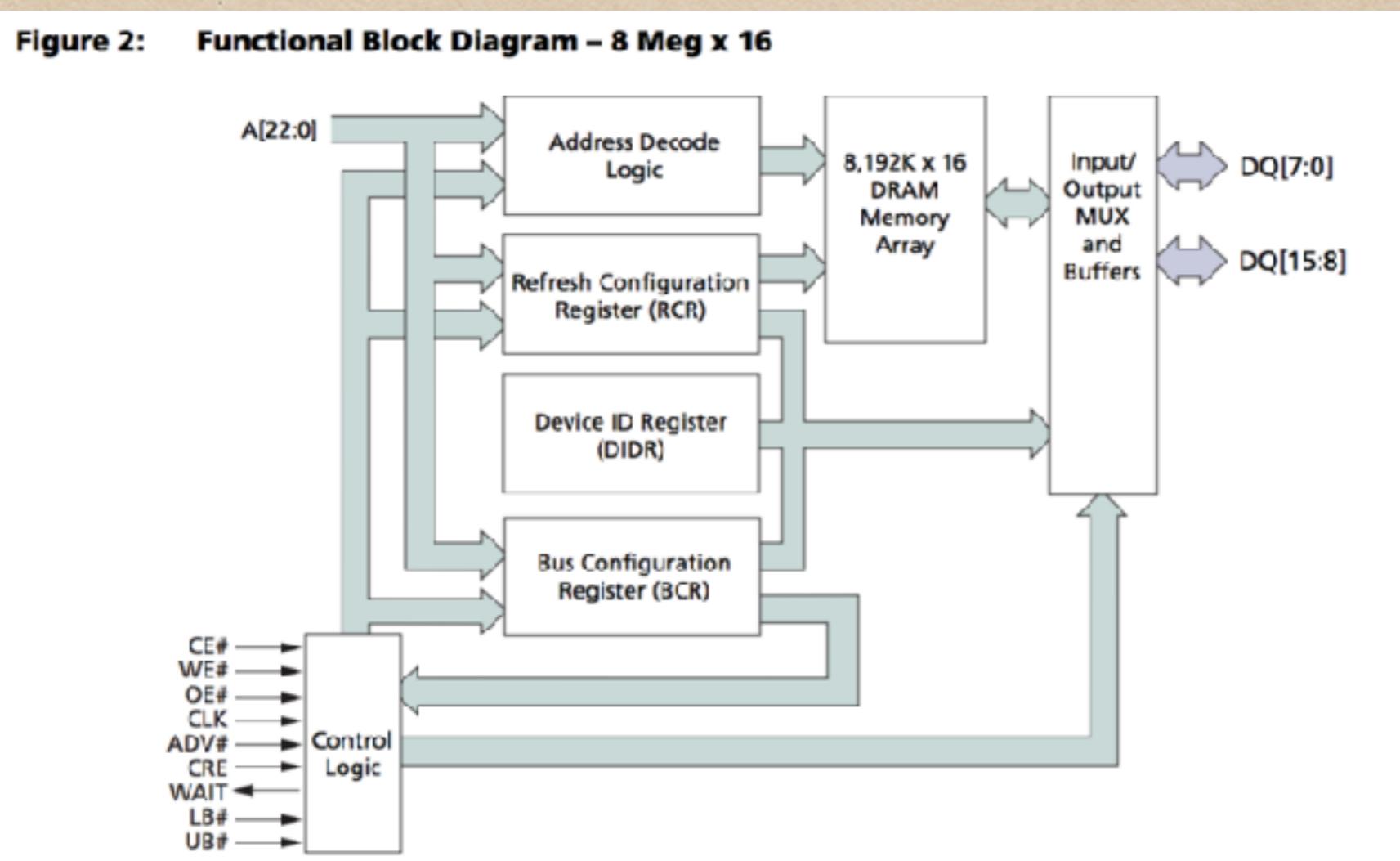
← Always '0'

← Always '0'

← Always '0'

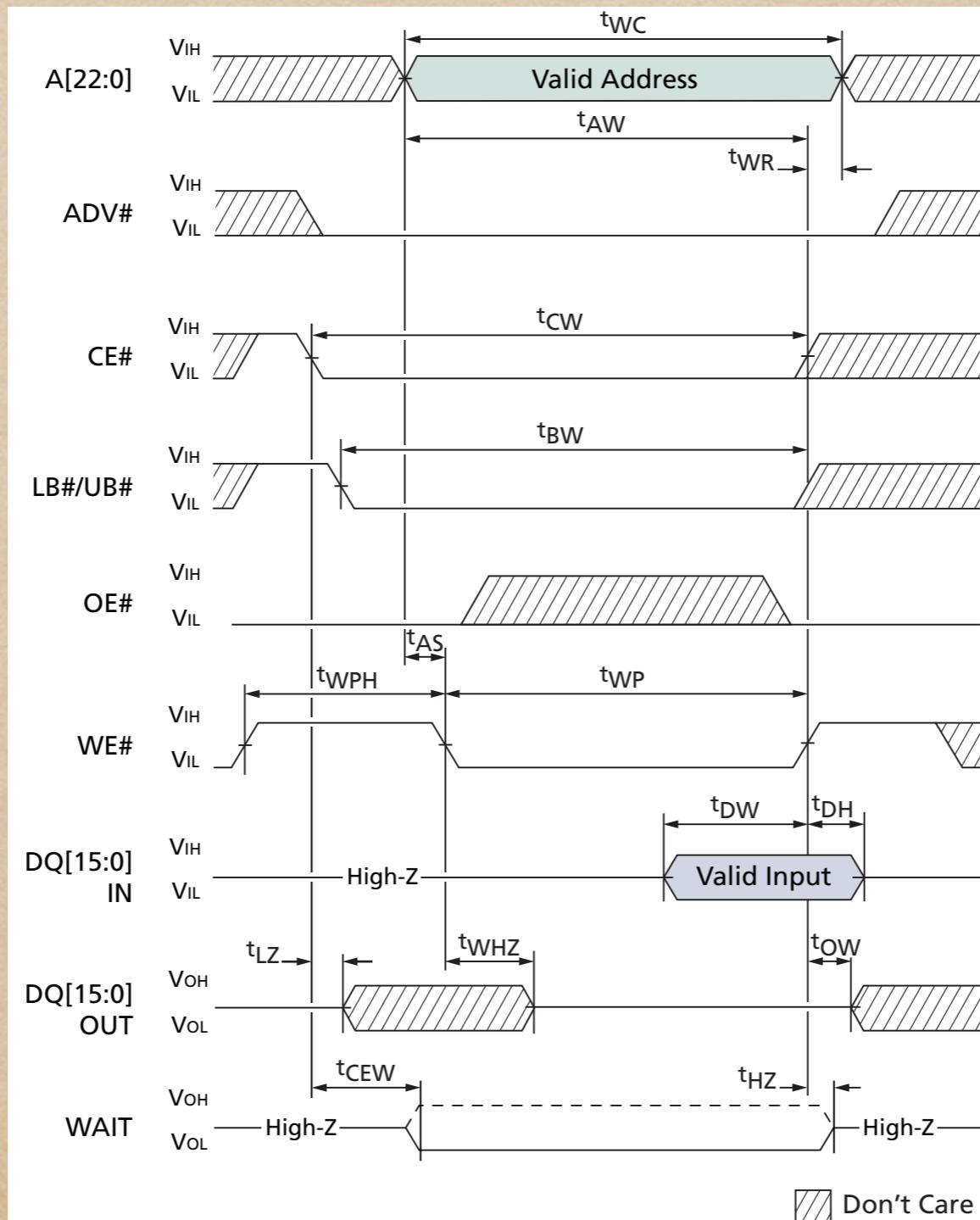
Memory Devices

- ◆ Both Nexys 2 and Nexus 3 Have the same memory device
- ◆ Nexys 3 M45W8MW16 (Micron)



Asynchronous Write

Figure 41: WE#-Controlled Asynchronous WRITE



Asynchronous Write Timing Requirements

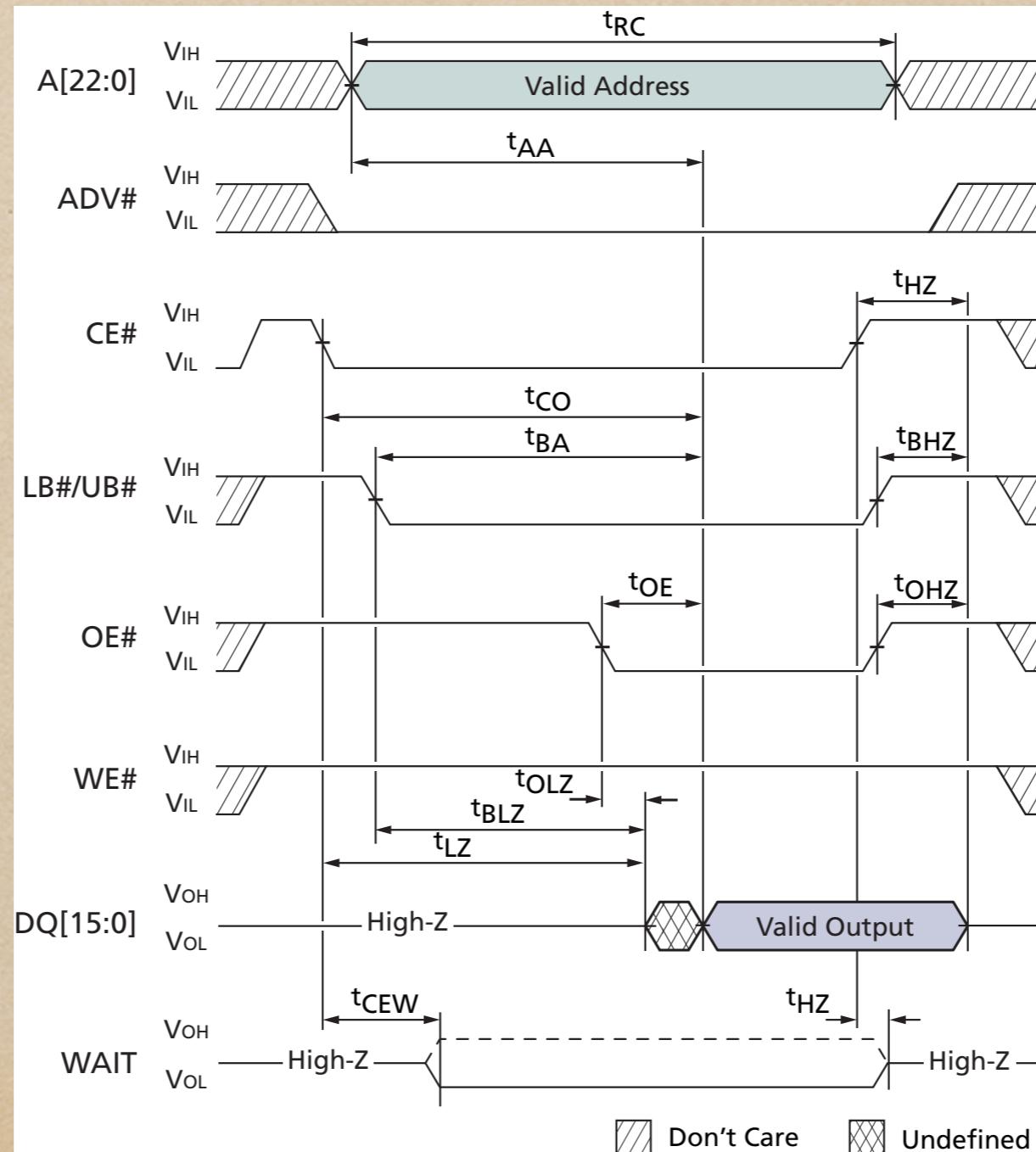
Table 16: Asynchronous WRITE Cycle Timing Requirements

Parameter	Symbol	70ns		85ns		Unit	Notes
		Min	Max	Min	Max		
Address and ADV# LOW setup time	t_{AS}	0		0		ns	
Address HOLD from ADV# going HIGH	t_{AVH}	2		2		ns	
Address setup to ADV# going HIGH	t_{AVS}	5		5		ns	
Address valid to end of WRITE	t_{AW}	70		85		ns	
LB#/UB# select to end of WRITE	t_{BW}	70		85		ns	
CE# LOW to WAIT valid	t_{CEW}	1	7.5	1	7.5	ns	
CE# HIGH between subsequent async operations	t_{CPH}	5		5		ns	
CE# LOW to ADV# HIGH	t_{CVS}	7		7		ns	
Chip enable to end of WRITE	t_{CW}	70		85		ns	
Data HOLD from WRITE time	t_{DH}	0		0		ns	
Data WRITE setup time	t_{DW}	20		20		ns	
Chip disable to WAIT High-Z output	t_{HZ}		8		8	ns	1
Chip enable to Low-Z output	t_{LZ}	10		10		ns	2
End WRITE to Low-Z output	t_{OW}	5		5		ns	2
ADV# pulse width	t_{VP}	5		7		ns	
ADV# setup to end of WRITE	t_{VS}	70		85		ns	
WRITE cycle time	t_{WC}	70		85		ns	
WRITE to DQ High-Z output	t_{WHZ}		8		8	ns	1
WRITE pulse width	t_{WP}	45		55		ns	3
WRITE pulse width HIGH	t_{WPH}	10		10		ns	
WRITE recovery time	t_{WR}	0		0		ns	

- Notes:
1. Low-Z to High-Z timings are tested with the circuit shown in Figure 27 on page 36. The High-Z timings measure a 100mV transition from either V_{OH} or V_{OL} toward $V_{CCQ}/2$.
 2. High-Z to Low-Z timings are tested with the circuit shown in Figure 27 on page 36. The Low-Z timings measure a 100mV transition away from the High-Z ($V_{CCQ}/2$) level toward either V_{OH} or V_{OL} .
 3. WE# LOW time must be limited to t_{CEM} (4μs).

Asynchronous Read

Figure 30: Asynchronous READ



Asynchronous Read Timing Requirements

Timing Requirements

Table 14: Asynchronous READ Cycle Timing Requirements

All tests performed with outputs configured for default setting of one-half drive strength, (BCR[5:4] = 01b)

Parameter	Symbol	70ns		85ns		Unit	Notes
		Min	Max	Min	Max		
Address access time	t_{AA}		70		85	ns	
ADV# access time	t_{AADV}		70		85	ns	
Page access time	t_{APA}		20		25	ns	
Address hold from ADV# HIGH	t_{AVH}	2		2		ns	
Address setup to ADV# HIGH	t_{AVS}	5		5		ns	
LB#/UB# access time	t_{BA}		70		85	ns	
LB#/UB# disable to DQ High-Z output	t_{BHZ}		8		8	ns	1
LB#/UB# enable to Low-Z output	t_{BLZ}	10		10		ns	2
Maximum CE# pulse width	t_{CEM}		4		4	μs	3
CE# LOW to WAIT valid	t_{CEW}	1	7.5	1	7.5	ns	
Chip select access time	t_{CO}		70		85	ns	
CE# LOW to ADV# HIGH	t_{CVS}	7		7		ns	
Chip disable to DQ and WAIT High-Z output	t_{HZ}		8		8	ns	1
Chip enable to Low-Z output	t_{LZ}	10		10		ns	2
Output enable to valid output	t_{OE}		20		20	ns	
Output hold from address change	t_{OH}	5		5		ns	
Output disable to DQ High-Z output	t_{OHZ}		8		8	ns	1
Output enable to Low-Z output	t_{OLZ}	3		3		ns	2
Page READ cycle time	t_{PC}	20		25		ns	
READ cycle time	t_{RC}	70		85		ns	
ADV# pulse width LOW	t_{VP}	5		7		ns	

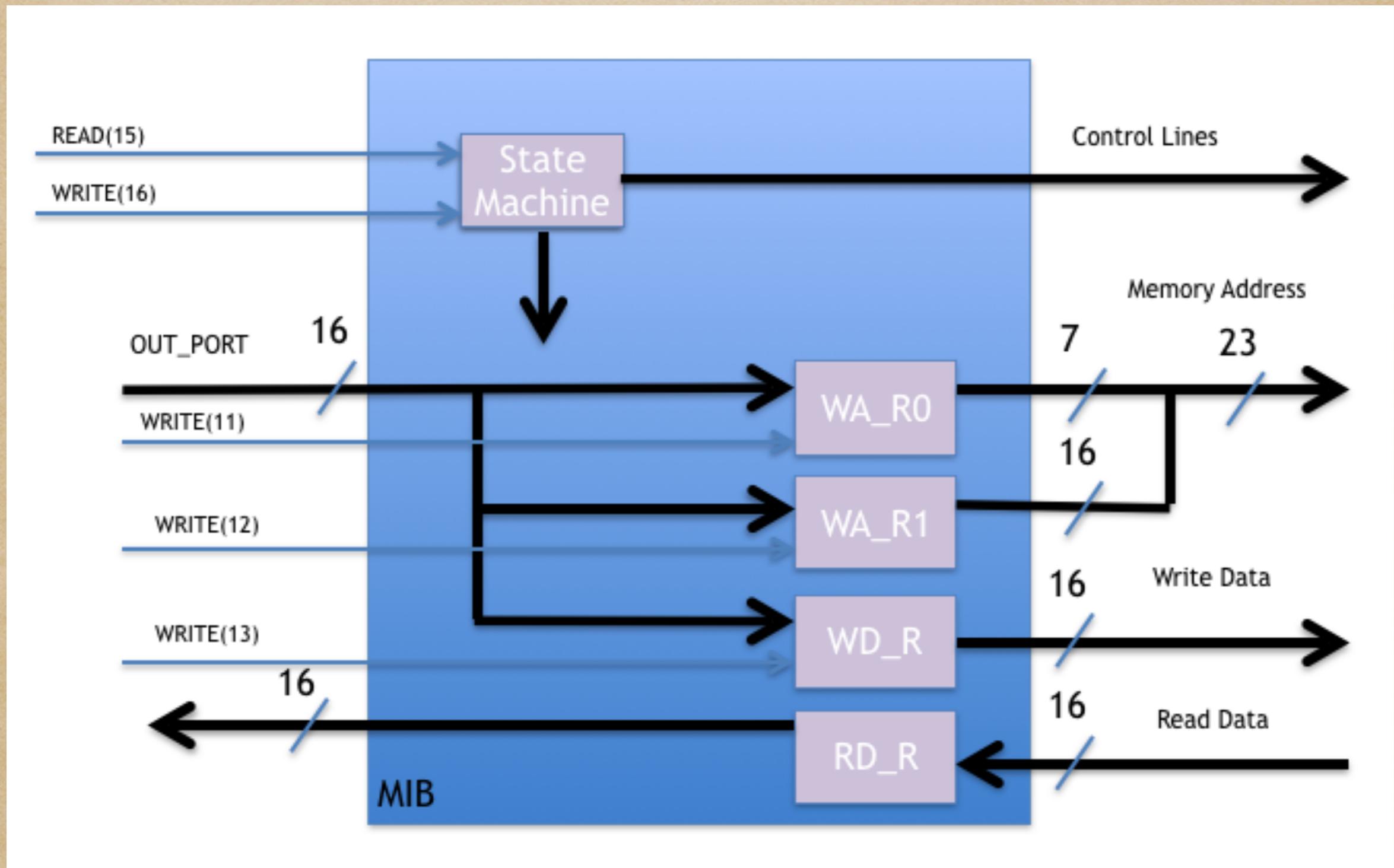
- Notes:
1. Low-Z to High-Z timings are tested with the circuit shown in Figure 27 on page 36. The High-Z timings measure a 100mV transition from either V_{OH} or V_{OL} toward $V_{CCQ}/2$.
 2. High-Z to Low-Z timings are tested with the circuit shown in Figure 27 on page 36. The Low-Z timings measure a 100mV transition away from the High-Z ($V_{CCQ}/2$) level toward either V_{OH} or V_{OL} .
 3. Page mode enabled only.

Implementation

- ◆ The interesting part of our project is to have an 16-bit interface communicate with a 16-bit memory, 8M deep
- ◆ Our MIB will need to have a number of internal registers that will need to be memory mapped to the processor
- ◆ Memory Map

Address	Operation
11	Write Address Register 0
12	Write Address Register 1
13	Write Data Out Register
14	Read Data In Register
15	Perform Memory Read
16	Perform Memory Write

Sample Block Diagram



Firmware Performance Description

- ◆ At startup you should print a banner and follow it with a newline and a prompt
- ◆ Every character that is received over the UART interface should still be echoed to the user. Now, starting at location 0, each byte should also be written to the Micron memory
- ◆ Remember the Micron memory interface should be written and read 16-bits at a time
- ◆ When an "*" is received from the user the data that has been written into the memory should be sent back to the user beginning with the first data byte received. Send a CR/LF first so that the dumped data begins at a new line
- ◆ After dumping the memory the controller should be ready to collect data again
- ◆ Every time you send a newline to the UART you should follow it with prompt
- ◆ When you detect a backspace (BS) you should automatically perform an auto delete to the display ensuring that you do not delete your prompt (no need to alter contents of memory)