Memory System Design

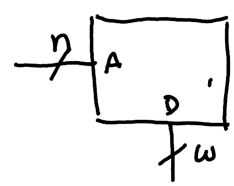
Exercise 1: Is t_{AW} a requirement or a guaranteed specification for this memory? How about the t_{AA}?

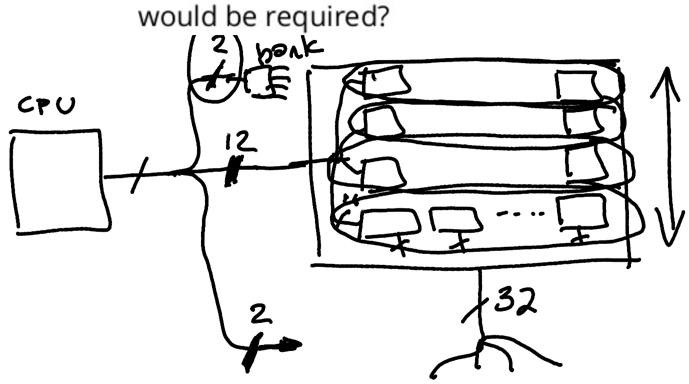
t_{AW} – requirement ($\approx$ setup time)

t_{AA} – guaranteed ($\approx$ prop. delay)

1FE00

Exercise 2: How many 4 k×4 memory IC's would be required to build a 16 k ×32 memory? What is the width of the data bus? How many address bus bits would be required? What address values could be placed on the address bus? How many chip-select lines would be required?





data bus is 32 bits

4 bytes

need 2 bits to select a byte

$4k = 2^{10} \cdot 2^2 = 2^{12}$

16 k x 32 bits

16 k x 4 bytes

= 64 kBytes

$\log_2(64k) = \log_2(2^6 \cdot 2^{10}) = 16$

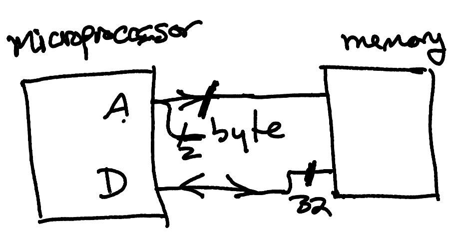
0000 → FFFF

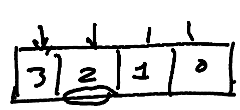
remember:

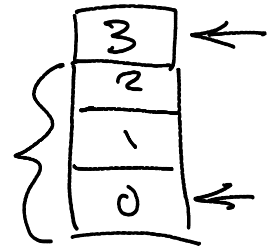
$2^{10} = 1k$

$2^{20} = 1M$

$2^{30} = 1G$







Exercise 3: How large are the two lowest memory regions in the memory map above?

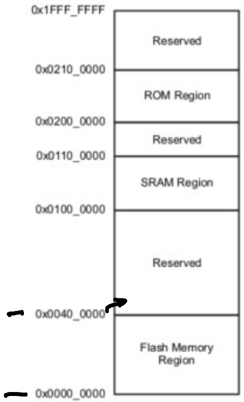


Figure 6-2. Code Zone Memory Map

100_0000 − 40_000
→ 0x 60_0000 = 12 MBytes.
→ 40_0000

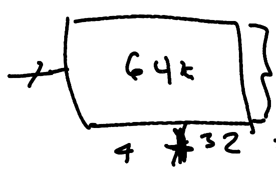
$2^{10} = 1k$
$2^{20} = 1M$
$2^{30} = 1G$

Exercise 4: If a CPU has a 32-bit address bus, now many bytes can it address? What range of addresses would correspond to the first 64 k Bytes? If this range of memory was to be implemented with 32-bit words, how many address bits would be required to select a byte within each word? How many bits would be required to select a 32-bit word within the 64 k range? How many bits are not directly connected to the memory ICs? What are they be used for?

N = 32 $\quad 2^{32} = 2^{30+2} = 2^{30} \cdot 2^2 = 4$ GB

0 → 64k − 1

32 bitwords → 2 bit for byte select



$\frac{64k}{4}$ = 16k words

4 * 32 = 4 bytes

2 bits for byte select.

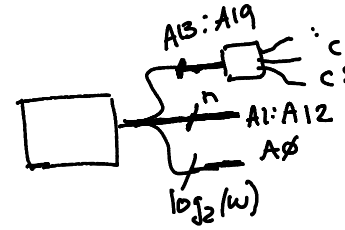
to select one word from 16k
$\log_2(16k) = 14$ bits

32 bits − (14 + 2) = 16 bits not used.
⇒ used to select other memory regions

word select (14); A_0, A_1 byte select (2).

64 k bytes = 16 k words × 4 bytes/word.

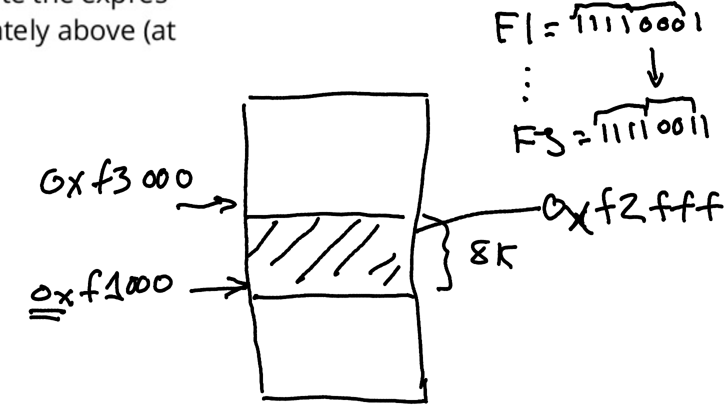
$\log_2(4k) = \log_2(2^{10} \cdot 2^2)$
$= \log_2(2^{12}) = 12$



Exercise 5: A 4k×16 memory is to be used in a system with a 20-bit address bus. This memory is to respond to addresses starting at 0xf1000. Draw the memory map. Assuming the address signal is defined as ~~signal A: std_logic_vector(19 downto 0);~~ and the chip-select as ~~signal CS0: std_logic~~;, write the ~~VHDL~~ Verilog that would implement the chip-select signals CS0. Write the expression for CS1 if there was a second ~~4~~ 8 kB bank immediately above (at a higher address than) the first.

logic [19:0] A;
logic cs0, cs1;

4k × 16 = 8 k bytes.



→ assign cs0 = A >= 20'hf1000 && A < 20'hf3000;
assign cs0 = A[19:13] > 7'b1111000
&& A[19:13] < 7'b1111001;

A19 A14
f1000 = 1111 0001 0000 0000 0000
↕
f2fff = 1111 0010 1111 1111 1111
f3

always 1 — always 0 — 01 or 10

assign cs0 = A[19:14] == 6'b111100
&& (A[13:12] == 2'b01
|| A[13:12] == 2'b10);

cs1 = f3000 → f5000
is the next 8k range after f3000

f3000 → 11110|011

f4fff → 11110|100

```
assign cs1 = A[19:15] == 5'b11110
        && (A[15:13] >= 3'b011 &&
            A[15:13] <= 3'b100);
```