Memory System Design

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

t_{AW} – measured to input → requirement

t_{AA} – measured to output → guaranteed

Exercise 2: How many 256 k×8 memory IC's would be required to build a 1 M ×32 memory? What is the width of the data bus? How many address bus bits would be required from the CPU? Which of these would connect to the memory ICs? What address values could be placed on the address bus? How many chip-select lines would be required?

32

20 bits for word & bank select

18 bits for word select.

$0 - 2^{18} - 1$ different word addresses

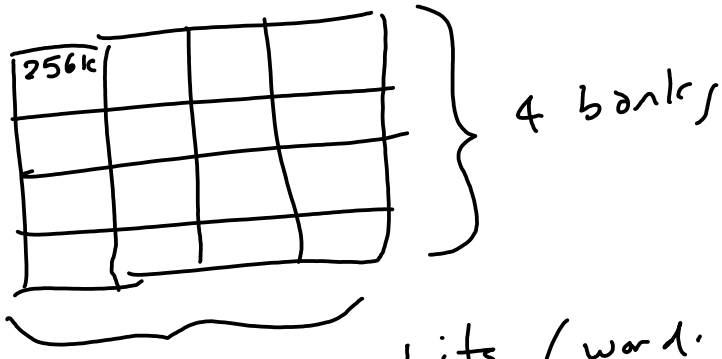
each 256k · 32 bank has 1 MByte

from CPU need

2 bits for bank select

18 bits for word select

2 bits for byte select



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

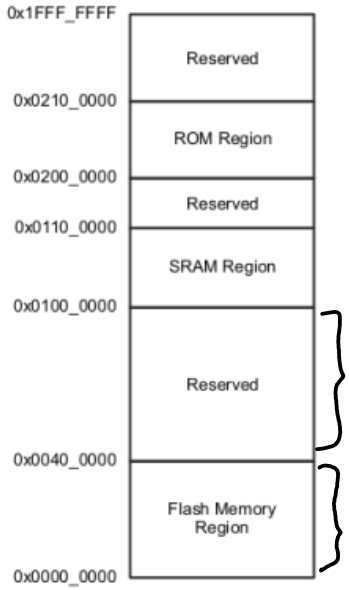


Figure 6-2. Code Zone Memory Map

100 0000
60 0000
A0 0000
$= 10 \times 2^{20} = 10$ MBytes.

40 0000
− 00 0000

$= 40\ 0000_{16} = 2^{22} = 4 \times 2^{16} \times 2^{10}$
$= 4$ MBytes.

0100 — 2 bits
4 · 5 = 20 bits

Exercise 4: If a CPU has a 32-bit address bus, how 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 would they be used for?

32 bits $\longrightarrow 2^{32}$ bytes can be addressed

64k Bytes = 0 to $(64 \times 1024 - 1)$

= 0 to 65535

32 bits is $\frac{32}{8} = 4$ bytes

need $\log_2 4 = 2$ bits to select one byte from each 32-bit word

64 kBytes = $\frac{64kB}{4 \text{ bytes/word}}$ = 16 k Words

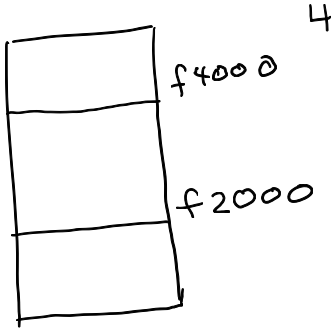
$\log_2 16k = 14$ bits needed to select a word

The least-significant 2 bits are not connected to the ICs:

They are used to select a byte from each 32-bit word.

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 20'hf2000. Draw the memory map. Assuming the address signal is defined as `logic [19:0] a;` and the chip-select as `logic cs0 ;`, write the Verilog that would implement the chip-select signals csO. Write the expression for a second chip-select, `cs1` that would enable a second 8 kBytes bank immediately above (at a higher address than) the first.

$4K \times 16 \text{ bits} = 4k \times \frac{16}{8} \text{ bytes} = 8K\text{Bytes}$
$= 8192_{10}$
$= 2000_{16}$




```
assign cs0 = a >= 20'hf2000 && a < 20'hf4000;
assign cs1 = a >= 20'hf4000 && a < 2'hf6000;
```

or, more simply, note that:

f2000 is 1111 0010 — 12 0s —
f3fff is 1111 0011 "
f4000 is 1111 0100 "
f5fff is 1111 0101 "

so this could also be written as:

```
assign cs0 = a[19:13] = 7'b1111_001;
assign cs1 = a[19:15] = 7'b1111_010;
```