Homework 1 | James Hicles | EENG 260

• Problem 1 •

-If a microprocessor has a 32 bit address bus and a 16 bit data bus, also its registers are 16 bit registers

a) What is the size of the address space this microprocessor can address?

• Because we have a 32 bit address bus we can access 2^{32} different addresses. ∴ The address space is 2^{32} 32 bit addresses.

b) This is a 32-bit microprocessor because the width of the address bus is 32 bits.

• Problem 2.

	End	Start		
- a)	1FFFFFFF -	00040000 =	1FFBFFFF →	536608767
	21FFFFFF -	2000 8000 =	1FF7FFF →	33521663
	3FFFFFFF -	2210 0000 =	1DEFFFF →	502267903
	40003FFF -	4000 2000 =	1FFF →	8191
	4001FFFF -	4001 4000 =	BFFF →	4915
	40027FFF -	4002 6000 =	1FFF →	8191
	4002BFFF -	4002A000 =	1FFF →	8191
	4002FFFF -	4002E000	TFFF →	8191
	4003BFFF -	4003A000 =	1FFF →	8191
	4003FFFF -	4003D000 =	2FFF →	12287
	4004BFFF -	40042000 =	9FFF →	40959
	40057FFF -	40051000 =	6FFF →	28671
	400AEFFF -	4005EC00 =	50FFF →	331775
	400F8FFF -	400B0000 =	48FFF →	299007
	400FBFFF -	400FA000 =	1FFF →	8191
	DFFFFFFF -	44000000 =	9BFFFFFF →	2617245695
	E000DFFF -	E0003000 =	AFFF →	45055
	E003FFFF -	5000F000 =	30FFF →	200703
	FFFFFFFF -	E0047000 =	1FFBDFFF →	536600575

$\log_2(4.227 \cdot 10^9) = 31.977$

$\approx 2^{31.977}$ Addresses are reserved.

$4.227301357 \cdot 10^9$ Addresses are reserved

Only 67,665,939 are not reserved

pg 41

b) -To be backward compatible with future products?
-Reserved for future use

-To use for constants because it's easier to just grab a memory address than it is to store a constant into memory?

-Allow for compatability with other products with more peripherals using the same chip.

c) 0x400A.F000 – 0x400A.F7FF or 0x400A.FFFF
START — 2KiB — END

On memory map this is the end address of EEPROM and key locker but it is 4KB from the base not 2KB

D) 16,384 Bits of memory

E) 512 32-bit words

F) 32 Blocks

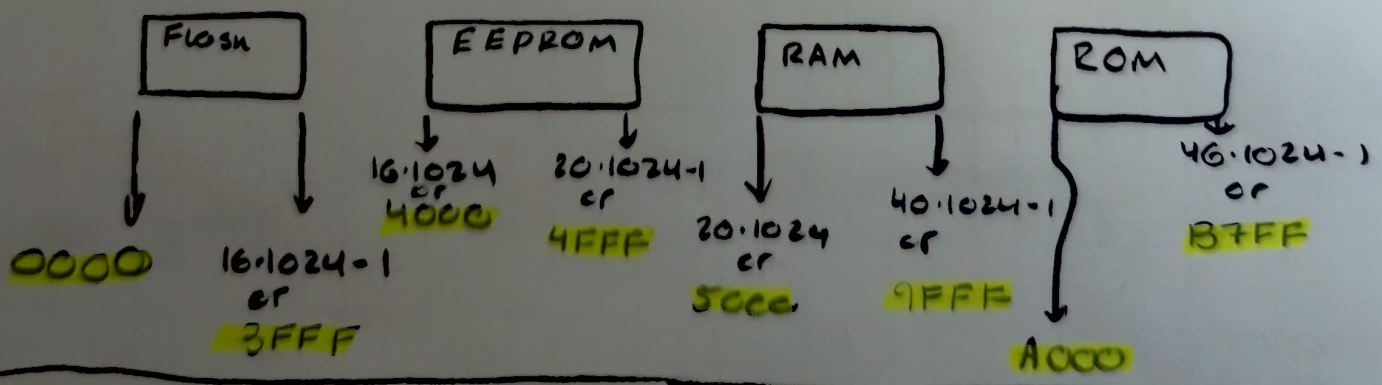
• Problem 3 •

a) ROM is non volatile

b) The bus interface unit writes data onto the bus during a write cycle.

[Pg 129 5th edition]

• Problem 4 •



• The start and end address of ROM is A000 and BFFF

- Problem 5 -

```
LDR   R0, Base
LDR   R1, [R0,#Offset]
ORR   R1, #0x020
STR   R1, [R0, #Offset]
```

- How on earth do we do this in 3 lines?!