Homework 1

(Due date: September 25th @ 5:30 pm)
Presentation and clarity are very important! Show your procedure!

PROBLEM 1 (15 PTS)

- Complete the following table. We are representing positive integer numbers.

Decimal	BCD (bits)	Binary	Hexadecimal
			2FE
	0011 1001 0101		
		10101100	
249			

- Complete the following table. Use the fewest number of bits in each case:

REPRESENTATION			
Decimal	**Sign-and-magnitude**	**1's complement**	**2's complement**
-247			
			1001101
		10010000	
	110110100		
		01100101	
			0110111

PROBLEM 2 (20 PTS)

- Perform the following additions and subtractions of 8-bit unsigned integers. Indicate every carry (or borrow) from c_0 to c_8 (or b_0 to b_8). For the addition, determine whether there is an overflow. For the subtraction, determine whether we need to keep borrowing from a higher byte.

Example:

- 54 + 210

$c_8=1$ $c_7=1$ $c_6=1$ $c_5=1$ $c_4=0$ $c_3=1$ $c_2=1$ $c_1=0$ $c_0=0$

```
 54 = 0x36 =   0 0 1 1 0 1 1 0 +
210 = 0xD2 =   1 1 0 1 0 0 1 0
-------------------------------
Overflow! → 1  0 0 0 0 1 0 0 0
```

- 77 - 194

Borrow out! → $b_8=1$ $b_7=0$ $b_6=0$ $b_5=0$ $b_4=0$ $b_3=0$ $b_2=1$ $b_1=0$ $b_0=0$

```
 77 = 0x4D =  0 1 0 0 1 1 0 1 -
194 = 0xC2 =  1 1 0 0 0 0 1 0
------------------------------
              0 0 0 0 1 0 1 1
```

- 129 + 103
- 198 + 67
- 43 - 98
- 149 - 87

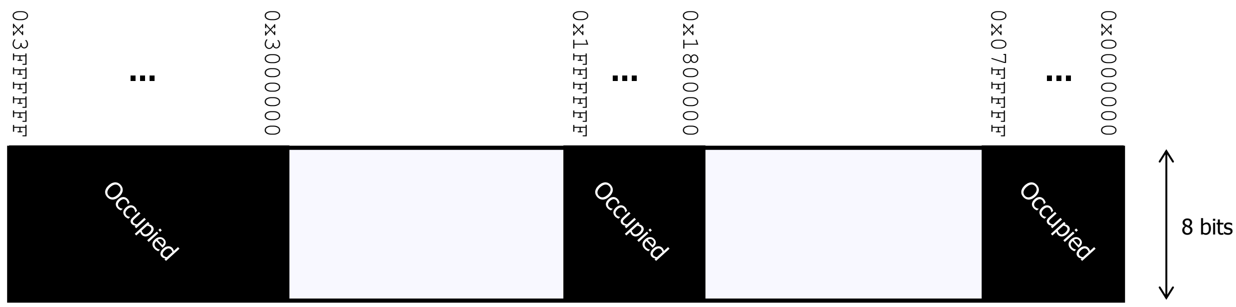
- Perform the following operations by representing the numbers in 2's complement representation using 8 bits. Indicate every carry (from c_0 to c_8). Determine whether the operation results in an overflow.
  - 101 + 23
  - 98 - 62
  - -90 - 82

- BCD Addition: We want to add these two decimal numbers using normal binary addition:
  - 6289 + 3098 = 9387
  - Convert the summation operands to their BCD representation.
  - Add the binary numbers as if they were unsigned numbers.
  - Specify which binary number we need to add to the previous summation so that we get the answer we are looking for (9387).

PROBLEM 3 (15 PTS)

- A microprocessor has a 32-bit address line. The size of the memory contents of each address is 8 bits.
  - What is the address range (lowest to highest, in hexadecimal) of the memory space for this microprocessor? What is the size (in bytes, KB, or MB) of the memory space?
  - A memory device is connected to the microprocessor. Based on the size of the memory, the microprocessor has assigned the addresses `0xA0400000` to `0xA07FFFFF` to this memory device. What is the size (in bytes, KB, or MB) of this memory device? What is the minimum number of bits required to represent the addresses on this memory device?

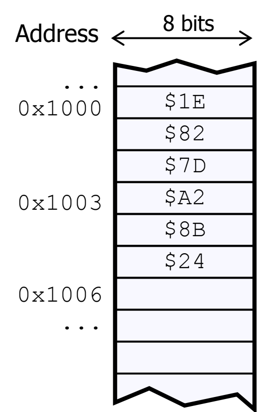
- The figure below depicts the complete memory space of a microprocessor:
  - What is the size (in bytes) of the memory space? What is the address bus size of the microprocessor?
  - If we have a memory chip of 8MB, how many bits do we require to address 8MB of memory (each memory address occupies 1 byte)?
  - If we want to connect the 8MB memory chip to the microprocessor, provide an address range (within the given memory space) that allows the 8MB of memory to be properly address. You are only allowed to use the non-occupied portions of the memory space as shown in the figure below.



PROBLEM 4 (15 PTS)

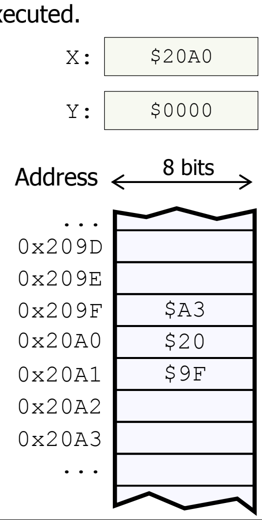
- Multi-precision addition: Write a set of instructions that properly perform the following unsigned addition: `$1E827D+$A28B24`.

  Your code first needs to store these constants in memory addresses `$1000` and `$1003` respectively (from higher byte to lower byte, as shown in the figure). The result (3 bytes) must be stored starting from memory address `$1006`. Complete the respective memory contents (in hexadecimal) in the figure.

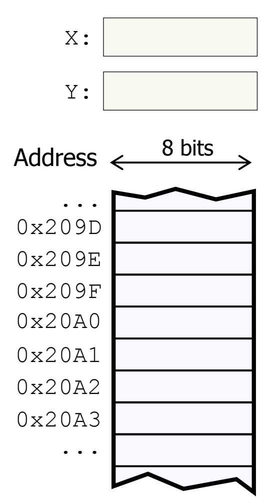


PROBLEM 5 (15 PTS)

- For the following set of instructions, provide the contents (in hexadecimal) of `X, Y` and relevant memory addresses after the last instruction has been executed. The figure on the left shows the values on `X, Y`, and memory before the first instruction is executed.




```
ldy 3,X-
movw [1,Y], 4,+X
clra
clrb
staa [D,Y]
```



PROBLEM 6 (20 PTS)

Given the following set of instructions, complete the following:

- Register values (in hexadecimal format) as the instructions are executed.
- The state of the memory contents (in hexadecimal format) after the last instruction has been executed. Also, specify the memory address at which the contents of `D` are stored (last instruction).
- The addressing mode of each instruction. Be specific, if for example the addressing mode is indexed, indicate which one in particular. Note that the `movw` instruction uses two addressing modes.

Addressing Mode	Instruction	A	B	X	Y
Inherent	clra clrb				
Immediate	ldx #$0F ldy #$00	$00	$00	$000F	$0000
	ldy #$1000				
	movw #$1F30,0,Y				
	ldab #149				
	sex b,d				
	nega				
	addd 0,y				
	iny				
	exg x,y				
	std [-1,X]