COMPUTATIONAL QUESTIONS:

1. What is the two's complement representation of the following decimal numbers? Show your hand calculations to convert these to binary and what method you choose to use. You may use a calculator to check your answers, but you still must show/describe your work!

   (a) -26

   (b) -89

a.) for -26, start by converting to +26 & finding 2's complement

$\Rightarrow \frac{26}{2} = \frac{13}{2}^{0} = \frac{6}{2}^{1} = \frac{3}{2}^{0} = \frac{1}{1}^{1} = 0^{1}$ = 00011010

inverting/finding complement => 11100101, adding 1

=> **1110 0110 = -26**

b) -89 => $\frac{+89}{2} = \frac{44}{2}^{1} = \frac{22}{2}^{0} = \frac{11}{2}^{0} = \frac{5}{2}^{1} = \frac{2}{2}^{1} = \frac{1}{2}^{0} = 0^{1}$

=> 0101 1001, invert => 1010 0110, add 1 => **1010 0111**

=> **-89**

2. Compute the sum of the following pairs of numbers using 8-bit two's complement arithmetic just like the uC does (convert to binary, do the addition then convert the result back to decimal). Show your hand calculations in binary, you may check your answers with calculator.

   (a) $36, -10$

   (b) $16, -77$

a.) 36, -10

first using 2's complement on -10 => $\frac{10}{2} = \frac{5}{2}^{0} = \frac{2}{2}^{1} = \frac{1}{2}^{0} = 0^{1}$

=> 0000 1010, invert => 1111 0101, add 1 => 1111 0110 = -10

$\frac{36}{2} = \frac{18}{2}^{0} = \frac{9}{2}^{0} = \frac{4}{2}^{1} = \frac{2}{2}^{0} = \frac{1}{2}^{0} = 0^{1}$ => 0010 0100 = 36

=> 36 + -10 => 0010 0100
+ 1111 0110
10001 1010 => 0001 1010 = 36 + -10 = 26

b.) 16, -77, -77 => 77/2 = 38/2 = 19/2 = 9/2 = 4/2 = 2/2 = 1/2 = 0 => 0100 1101,
invert => 1011 0010, add 1 => 1011 0011 = -11

16/2 = 8/2 = 4/2 = 2/2 = 1/2 = 0 => 0001 0000

=> 16 + -77 => 0001 0000
+ 1011 0011
1100 0011 => 16 + -77 = -61

3. Answer the following questions from Gaonkar: 2.14, 3.4, 3.10, 4.11, 4.13, 4.1SE, 4.2SE, 5.16 (SE questions are to be done by hand and submitted the same as the regular questions. You may choose to use the MPlab simulator to check your answers.)

2.11.) The size of the address bus for the data memory in the PIC18F MCU is 12 bits and can can address 4MB of registers

3.4.) If the BSR register holds byte 01, then the command MOVWF 0x7F, 0 copies the contents of the W register to the register 0x7F in the access bank since the dp code is set to 0

3.10.) For the first instruction, MOVLW 0XFA, the w register now is filled with the value 0XFA (1111 1010), and N OV Z DC are set to 1, 0, 0, & 0 respectively

for the next instruction, ADDLW 0x38, the w register now reads 0XFA + 0x38 = 0x32 = 0011 0010 because the W register can only accomodate 8 bits, when the answer should really be 0001 0011 0010 because of this, N OV Z DC will be 0, 1, 0, & 1 respectively, due to the overflow & carry necessary in for the operation

4.11) Given the 3 lines of instructions:

Start: MOVLW 0X67

ADDLW 0X33

SLEEP

a.) W register should be 0X33 + 0X67 = 0XCC

b.) Flags set after addition => N = 1, OV = 1, Z = 0, DC = 0, C = 0

c.) Based on the flags set, the status byte would be 0X18

4.13) calculating sum of 2 bytes

=> 0X92 = 1001 0010, 0XA7 = 1010 0111

=> 0X92 + 0XA7 =>
$$\begin{array}{r} {\scriptstyle (1)\ \ 1\ \ \ 1\ \ \ \ \ \ } \\ 1001\ 0010 \\ +\ 1010\ 0111 \\ \hline \end{array}$$

0011 1001 = 57

Based on the instructions/operations set the overflow & carry flags

4.15E)

a.) sum of Byte1 & Byte 2 => 0X34 + 0X56 =>
$$\begin{array}{r} 34 \\ +\ 56 \\ \hline 8A \end{array}$$

b.) Status register is 0X18 (N + OV = 1)

c.) Reglo contains contents 0X00 since there is overflow

d.) ✓

4.25E

a.) sum of the bytes =>
$$\begin{array}{r} 35 \\ +\ 2A \\ \hline 5F \end{array}$$

b) Status byte is 0X00CU since there is no carry, overflow, it is not negative, it is positive, & there is no carry-out

c.) Reg 10 contains the sum of byte1 + byte2 = 35+2A = 5F because there was no overflow

d.) V

5.16) Instructions to add 2 bytes 78_H & $F2_H$

=> MOVLW 0x78
AddLW 0xF2

=> ① 78
+ F2
6A

=> carry bit set in status

=> N=0, OV=0, Z=0, DC=, C=1

for unsigned bytes, the overflow bit will no longer be set, but the carry bit still would be set

QUESTIONS (5067 ONLY)

1. What is the two's complement representation of the following decimal numbers? Show your work in binary.

   (a) −348

   (b) −130

a.) -348 => start w/ +348

=> $\frac{348}{2}$ = $\frac{174}{2}$ = $\frac{87}{2}$ = $\frac{43}{2}$ = $\frac{21}{2}$ = $\frac{10}{2}$ = $\frac{5}{2}$ = $\frac{2}{2}$ = $\frac{1}{2}$ = 0

(0 0 1 1 1 1 0 0)

=> 0000 0001 0101 1100, invert =>

1111 1110 1010 0011, add 1 => 1111 1110 1010 0100 = -348

b.) -130 => +130 = $\frac{65}{2}$ = $\frac{32}{2}$ = $\frac{16}{2}$ = $\frac{8}{2}$ = $\frac{4}{2}$ = $\frac{2}{2}$ = $\frac{1}{2}$ = 0

(0 1 0 0 0 0 1 0)

=> 0000 0000 1000 0010, invert =>

1111 1111 0111 1101, add 1 => 1111 1111 0111 1110 = -130

2. Compute the sum of the following pairs of numbers using 8-bit two's complement arithmetic (convert to binary, do the addition then convert the result back to decimal). Show your work in binary.

   (a) $-63, -17$

   (b) $-54, 28$

a.) -63 + -17 => starting w/ 2's complement

=> -63 => +63 = 31 = 15 = 7 = 3 = 1 = 0 => 0011 1111, invert => 1100 0000.

add 1 => 1100 0001 = -63

same for -17 => +17 = 8 = 4 = 2 = 1 = 0 => 0001 0001. invert => 1110 1110.

add 1 => 1110 1111 = -17

=> -63 + -17 => 1 1 1 1 1
1100 0001
+ 1110 1111
1011 0000

=> 1011 0000 = -63 -17 = -80 w/ carry

b.) -54, 28,

same process => -54 => +54 = 27 = 13 = 6 = 3 = 1 = 0 => 0011 0110, invert

=> 1100 1001, add 1 => 1100 1010 = -54

28 = 14 = 7 = 3 = 1 = 0 => 0001 1100 = 28

=> -54 + 28 => 1 1
1100 1010
+ 0001 1100
1110 0110 = -54 + 28 = -26

3. Gaonkar 3.19, 4.4SE, 4.5SE, 5.17 (SE questions are to be done by hand and submitted the same as the regular questions. You may choose to use the MPlab simulator to check your answers.)

3.19) The statement can be explained in the way that each 1-word instruction cycle really executes in 2 parts of a cycle, because 1 full cycle is made up of 2 instruction cycles, which consist of fetch + execute

4.4SE) Using same program as 4.1SE, changing byte1 to $F7_H$ + byte 2 to 88_H =>

a.) sum of new data bytes =>
$$\begin{array}{r} \textcircled{1}F7 \\ +\ 88 \\ \hline 17F \end{array}$$

b.) Status register contains overflow + carry bits because the sum exceeds the usual 8-bit representation of the working register

=> Status = 0x09

c.) Because an overflow occurred, URIE is set to 0

=> reg 10 = 0

d.) ✓

4.5SE)

After building the given script, the sum of the numbers comes out to be 0x75 in reg11

5.17) For signed numbers, the OV flag & carry flags are both going to be needed, as with signed numbers, negative values are now possible, thus both the overflow & carry bits are necessary / relevant since overflow signals when there is a sign change