

EE 2004
2016-2017: Semester B
Assignment 1
Due: Feb 24, 2017

Solution

Instructions:

The .asm and .lst files for the programs required for Questions 1 and 2 should be submitted. The answers for Questions 3 and 4 should be provided in a Microsoft Word file. **The Microsoft Word file and all the requested .asm/.lst files should be zipped as a single zip file for submission.**

Students must submit the assignment through Canvas. Click on the item "Assignment" on the left panel. You should see a row with title "Assignment 1". Click on the "Assignment 1" label and find the "Submit Assignment" label on the right panel. Click on it and upload the requested zip file.

Question 1 (35 marks)

Loop: Summing 9 numbers in a sequence

The Lucas sequence is defined by the following recurrence relation:

$$Q_n = Q_{n-1} + Q_{n-2}$$

with seed value

$Q_1 = 1$ and $Q_2 = 3$ where Q_n is the n^{th} number in the sequence.

Write a loop in assembly language that calculates the sum of the first 9 numbers in the sequence (i.e., $Q_1 + Q_2 + \dots + Q_9$). Store the sum in the file register with address 0x000. Save your program in a file named Loop.asm. Assemble and verify your code using MPLAB. **You should submit the Loop.asm and Loop.lst files.**

Solution:

```
LIST P=18F4520 ;directive to define processor
#include <P18F4520.INC> ;processor specific
                        ;variable definitions

CBLOCK 0x00
Sum
Qn1
Qn2
Count
endc

Iteration equ d'7'

;-----
                        ORG      0x0000
                        goto Main ;go to start of main code
;-----
;Start of main program

Main:      movlw Iteration
            movwf Count, A
            movlw d'3'
            movwf Qn1, A; this variable stores F_{n-1}
            movlw d'1'
            movwf Qn2, A; F_{n-2}
            movlw d'4'; initialized Sum as F_1 + F_2 = 4;
            movwf Sum, A

Loop:      movf Qn1, W, A
            addwf Qn2, W, A; [WREG] = [Qn1] + [Qn2]
                        ;Note: WREG stores Qn
            addwf Sum, F, A; [Sum] = [Sum] + [WREG]
            movff Qn1, Qn2
            movwf Qn1, A

            decfsz Count, F, A
            bra Loop

            bra $
;*****
;End of program
;
                        END
```

Question 2 (25 marks)

BCD addition of two 32-bit numbers

Write a program to add two decimal numbers 45524199 and 33499999. Note that the result should be stored in four file registers. Save your program in a file named BCD.asm. Assemble and verify your code using MPLAB. You should submit the BCD.asm and BCD.lst files.

```
LIST P=18F4520;directive to define processor
#include <P18F4520.INC> ;processor specific
                        ;variable definitions

CBLOCK 0x10
SumLow
SumHigh
SumUpper
SumUpper2
endc

;-----
ORG      0x0000
goto Main ;go to start of main code
;-----

Main:      movlw 0x99
           addlw 0x99
           daw
           movwf SumLow, A

           movlw 0x41
           movwf SumHigh, A
           movlw 0x99
           addwfc SumHigh, W, A
           daw
           movwf SumHigh, A

           movlw 0x52
           movwf SumUpper, A
           movlw 0x49
           addwfc SumUpper, W, A
           daw
           movwf SumUpper, A

           movlw 0x45
           movwf SumUpper2, A
           movlw 0x33
           addwfc SumUpper2, W, A
           daw
           movwf SumUpper2, A

           bra $

END
```

Question 3 (20 marks)

Use the program below to answer the following questions. Explanation must be given to all your answers.

Line Number

```
9          cblock 0x04
10          BSR_Set
11          MyReg
12          endc
13
14          ;-----
15          ORG      0x0000
16
17          Main:    movlb BSR_Set
18                  clrf MyReg, A
19                  clrf MyReg, BANKED
20
21                  movlw 0xF2
22                  movwf MyReg, A
23                  movlw 0x37
24                  addwf MyReg, F, A
25
26                  movlw 0x33
27                  movwf MyReg, BANKED
28                  movlw 0x59
29                  addwfc MyReg, F, BANKED
30
31          ;*****
32          ;End of program
33          ;
34          END
```

- (a) Write down the 12-bit address of the memory location in which the result of the addition operation in Line 24 of the above program is stored.

005

- (b) What are the statuses of the five flags in the STATUS register immediately following the execution of Line 24? You must demonstrate how you come up with the statuses of the five flags. Otherwise, you will obtain 0 mark in this question.

C = 1

DC = Z = OV = N = 0

- (c) Write down the 12-bit address of the memory location in which the result of the addition operation in Line 29 of the above program is stored.

405

- (d) What are the statuses of the five flags in the STATUS register immediately following the execution of Line 29? You must demonstrate how you come up with the statuses of the five flags. Otherwise, you will obtain 0 mark in this question.

DC = C = Z = 0

OV = N = 1

- (e) Identify the value stored in the register you identified in (c) immediately after Line 29 has executed. Use hex representation.

8D

Question 4 (20 marks)

For the following program, calculate the relative/absolute addresses (marked by “??”). You must show detailed calculations. Otherwise, you will obtain 0 mark in this question.

Program Memory Address	Machine Code	LINE	SOURCE
		00008	CBLOCK 0x00
		00009	FirstReg
		00010	SecondReg
		00011	ThirdReg
		00012	MinReg
		00013	endc
		00014	
		00015	org 0x000000
000000	<u>EF?? F???</u>	00016	goto Main
		00017	;-----
		00018	org 0x000040
000040	0E37	00019	Main:
000042	6E00	00020	movlw d'55'
000044	0E3C	00021	movwf FirstReg, A
000046	6E01	00022	movlw d'60'
000048	0E21	00023	movwf SecondReg, A
00004A	6E02	00024	movlw d'33'
		00025	movwf ThirdReg, A
00004C	5000	00026	Here:
00004E	6E03	00027	movf FirstReg, W, A
000050	5C01	00028	movwf MinReg, A
000052	<u>E7??</u>	00029	subwf SecondReg, W, A
000054	5001	00030	bnn Continue;
000056	6E03	00031	MinEqSecond:
000058	5003	00032	movf SecondReg, W, A
00005A	6002	00033	movwf MinReg, A
00005C	<u>D???</u>	00034	cpfslt ThirdReg, A
00005E	5002	00035	bra Over
000060	6E03	00036	MinEqThird:
000062	<u>D???</u>	00037	movf ThirdReg, W, A
		00038	movwf MinReg, A
		00039	Over:
			bra Over
			end

Solution:

```

                                00008          CBLOCK 0x00
00000000          00009          FirstReg
00000001          00010          SecondReg
00000002          00011          ThirdReg
00000003          00012          MinReg
                                00013          endc
                                00014
000000          00015          org 0x000000
000000 EF20 F000  00016          goto Main
                                00017 ;-----
000040          00018          org 0x000040
000040 0E37      00019 Main:    movlw d'55'
000042 6E00      00020          movwf FirstReg, A
000044 0E3C      00021          movlw d'60'
000046 6E01      00022          movwf SecondReg, A
000048 0E21      00023          movlw d'33'
00004A 6E02      00024          movwf ThirdReg, A
                                00025
00004C 5000      00026 Here:    movf FirstReg, W, A
00004E 6E03      00027          movwf MinReg, A
000050 5C01      00028          subwf SecondReg, W, A; [WREG]
= [SecondReg]-[FirstReg]
000052 E702      00029          bnn Continue; if [SecondReg]
>= [FirstReg], jump to Continue
000054 5001      00030 MinEqSecond: movf SecondReg, W, A
000056 6E03      00031          movwf MinReg,
A ;if [SecondReg]<[MinReg] then [MinReg] = [SecondReg]
000058 5003      00032 Continue: movf MinReg, W, A
00005A 6002      00033          cpfslt ThirdReg, A
00005C D002      00034          bra Over
00005E 5002      00035 MinEqThird: movf ThirdReg, W, A
000060 6E03      00036          movwf MinReg, A ;if
[ThirdReg]<[MinReg] then [MinReg] = [ThirdReg]
000062 D7FF      00037 Over:    bra Over
                                00038
                                00039          end
```

- Checklist:**
- (1) Loop.asm and Loop.lst
 - (2) BCD.asm and BCD.lst
 - (3) Word file providing the answers for Questions 3 and 4.

These files should be zipped and submitted as a single zip file.