EE 2004
Week 5 Tutorial
Solution

1. Loop: Summing 10 numbers in a sequence

Suppose a sequence is defined by the following recurrence relation:

$F_n = F_{n-1} + 0x02$

with seed value

$F_1 = 0x03$, where F_n is the n^{th} number in the sequence.

Write a loop that calculates the sum of the first 10 numbers in the sequence. Assume the sum can be stored in a byte (i.e., sum < = 255).

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

          cblock 0x00
          Count
          Numi
          Sum
          endc

          org 0x000000
;-----------------------------------------------------------------------
;Initialization: [Sum] = 0, [Numi] = 0x03, [Count] = d'10'

Initialization:  movlw d'10'
          movwf Count, A
          movlw 0x03
          movwf Numi, A
          clrf Sum, A

Here:     movf Numi, W, A;      [WREG] = [Numi]
          addwf Sum, F, A;      [Sum]  = [Sum] + [WREG]
          addlw 0x02;           [WREG] = [Numi] + 0x02
          movwf Numi, A
          decfsz Count, F, A
;    [Count] = [Count] - 1; skip if [Count] = 0 after decrement
          bra Here
          bra $
end
```

```
Program
Memory   Machine
Address  Code        LINE   SOURCE

                     00008                CBLOCK 0x00
                     00009                FirstReg
                     00010                SecondReg
                     00011                ThirdReg
                     00012                MaxReg
                     00013                endc
                     00014
000000 EF20 F000     00015                org 0x000000
000000 EF?? F???     00016                goto Main
                     00017 ;-----------------------------------------
-
000040               00018                org 0x000040
000040 0E2C          00019 Main:          movlw d'44'
000042 6E00          00020                movwf FirstReg, A
000044 0E3C          00021                movlw d'60'
000046 6E01          00022                movwf SecondReg, A
000048 0E37          00023                movlw d'55'
00004A 6E02          00024                movwf ThirdReg, A
00004C 5000          00025 Here:          movf FirstReg, W, A
00004E 6E03          00026                movwf MaxReg, A
000050 6401          00027                cpfsgt SecondReg, A
000052 D??? D003     00028                bra Continue
000054 D??? D000     00029                bra MaxEqSecond
000056 5001          00030 MaxEqSecond:   movf SecondReg, W, A
000058 6E03          00031                movwf MaxReg, A
00005A 5003          00032 Continue:      movf MaxReg, W, A
00005C 6402          00033                cpfsgt ThirdReg, A
00005E D??? D002     00034                bra Over
000060 5002          00035 MaxEqThird:    movf ThirdReg, W, A
000062 6E03          00036                movwf MaxReg, A
000064 D??? D7FF     00037 Over:          bra Over
                     00038                end
```

0 0000 0000 0000 0100 0000

0 0 0 2 0

$R = \frac{5A - 54}{2} = 3$

$R = \frac{56 - 56}{2} = 0$

$R = \frac{64 - 60}{2} = 2$

$R = \frac{64 - 66}{2} = -1$

$-1 =$ 000 0000 0001

111 1111 1110

\+ 1

111 1111 1111

7 F F

2. Branching: Relative addressing and absolute addressing

For the following two programs, calculate the relative/absolute addresses (marked by "??"). Demonstrate your calculations and verify the machine code you obtained by building the code and inspecting the .lst file.

For this question, answers can be obtained by looking at the .lst file generated by MPLAB.

Program Memory Address	Machine Code	LINE	SOURCE
		00022	CBLOCK 0x20
		00023	Binary
		00024	Tens
		00025	Units
		00026	ENDC
	EF10 F000	00027	
		00028	ORG 0x000000
000000	**EF?? F???**	**00029**	**goto Main**
		00030	ORG 0x000020
000020	0E4D	00031 Main:	movlw d'77'
000022	6E20	00032	movwf Binary, A
000024	6A21	00033 Bin_2_BCD:	clrf Tens, A
000026	6A22	00034	clrf Units, A
000028	5020	00035	movf Binary, W, A
00002A	0FF6	00036 Loop:	addlw -d'10'
00002C	**E3??** E302	**00037**	**bnc Next**
00002E	2A21	00038	incf Tens, F, A
000030	**D???** D7FC	**00039**	**bra Loop**
000032	0F0A	00040 Next:	addlw d'10'
000034	6E22	00041	movwf Units, A
		00042	END

0 0000 0000 0000 0010 0000

0 0 0 1 0

$R = \frac{32 - 2E}{2} = 2$

$R = \frac{2A - 32}{2} = -4$

$-4 = 000\ 0000\ 0100$

$111\ 1111\ 1011$

$+\ 1$

$111\ 1111\ 1100$

7 F C