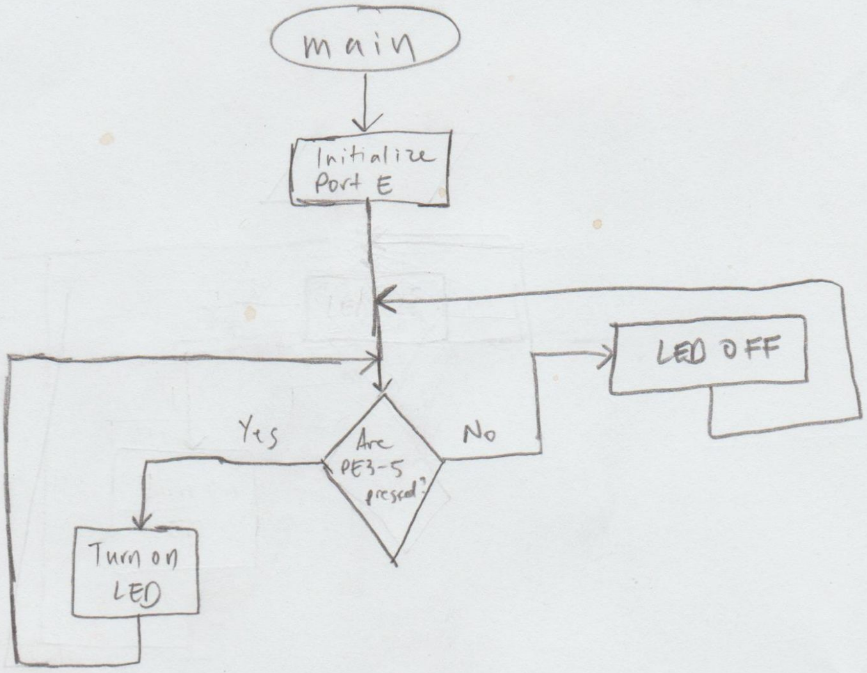
Lab 1 Flowchart + Psuedocode

Elvin Galarza ejg2298
Instructor: VJR

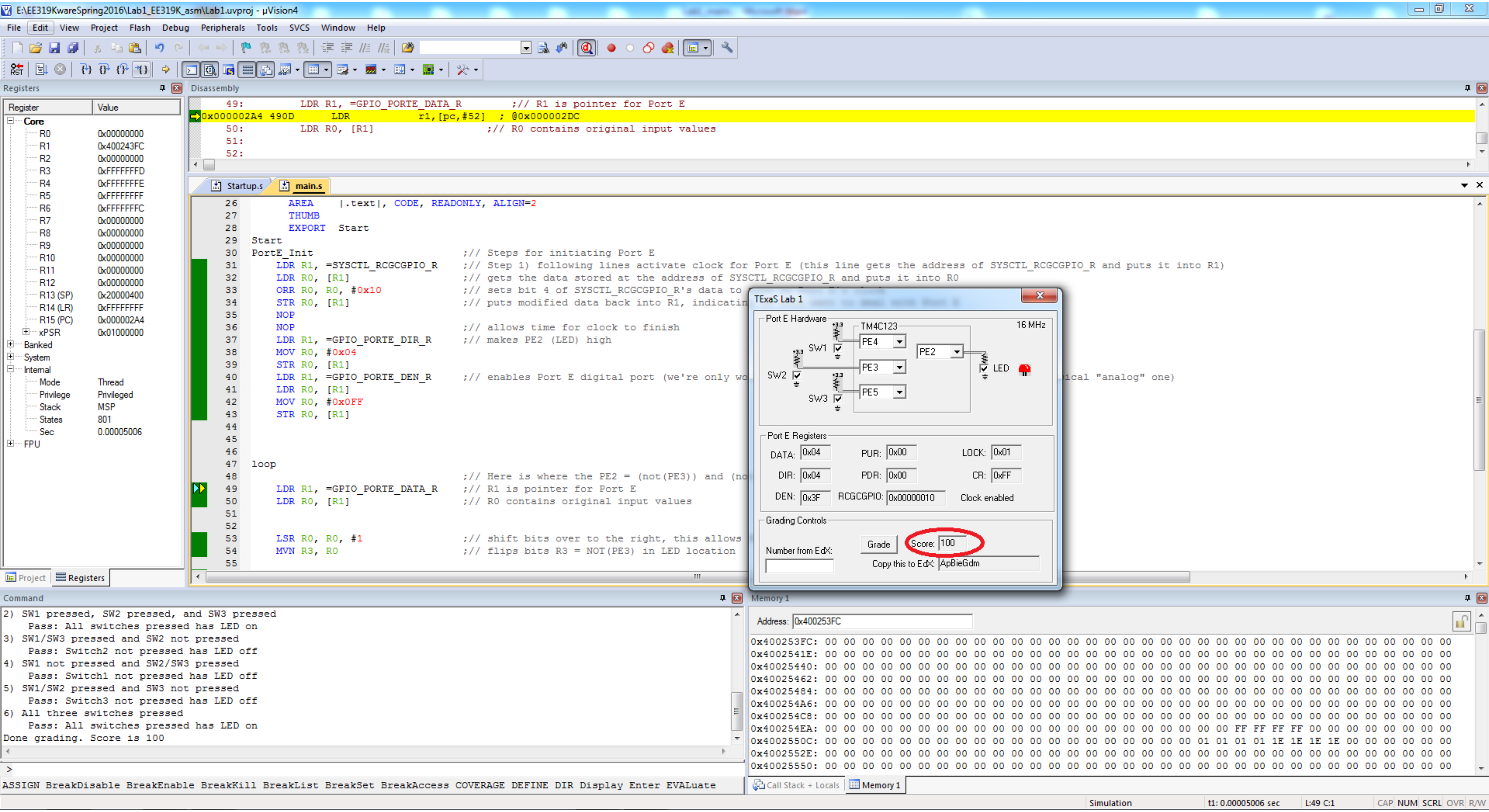
Flowchart:



Psuedo Code:

```
Initialize clock for PortE
Allow for some time to pass by, so that clock can activate

check: If  not all of PE3-5 are pressed (1 indicating at least one not pressed)
           Turn LED off (0 indicating off)
           Return to check if PE3-5 are all pressed
       If  all PE3-5 are pressed (0 indicating pressed)
           Turn LED on (1 indicating on)
           Return to check if PE3-5 are all pressed
```



E:\EE319KwareSpring2016\Lab1_EE319K_asm\Lab1.uvproj - µVision4

Register	Value
Core	
R0	0x00000000
R1	0x400243FC
R2	0x00000000
R3	0xFFFFFFFD
R4	0xFFFFFFFE
R5	0xFFFFFFFF
R6	0xFFFFFFFC
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x20000400
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x000002A4
xPSR	0x01000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	801
Sec	0.00005006
FPU	

```
    49:         LDR R1, =GPIO_PORTE_DATA_R        ;// R1 is pointer for Port E
0x000002A4 490D      LDR         r1,[pc,#52]  ; @0x000002DC
    50:         LDR R0, [R1]                  ;// R0 contains original input values
    51:
    52:
```

```
        AREA    |.text|, CODE, READONLY, ALIGN=2
        THUMB
        EXPORT  Start
Start
PortE_Init                               ;// Steps for initiating Port E
     LDR R1, =SYSCTL_RCGCGPIO_R     ;// Step 1) following lines activate clock for Port E (this line gets the address of SYSCTL_RCGCGPIO_R and puts it into R1)
     LDR R0, [R1]                   ;// gets the data stored at the address of SYSCTL_RCGCGPIO_R and puts it into R0
     ORR R0, R0, #0x10              ;// sets bit 4 of SYSCTL_RCGCGPIO_R's data to
     STR R0, [R1]                   ;// puts modified data back into R1, indicatin
     NOP
     NOP                            ;// allows time for clock to finish
     LDR R1, =GPIO_PORTE_DIR_R      ;// makes PE2 (LED) high
     MOV R0, #0x04
     STR R0, [R1]
     LDR R1, =GPIO_PORTE_DEN_R      ;// enables Port E digital port (we're only wo                    ical "analog" one)
     LDR R0, [R1]
     MOV R0, #0x0FF
     STR R0, [R1]



loop
                                    ;// Here is where the PE2 = (not(PE3)) and (no
     LDR R1, =GPIO_PORTE_DATA_R     ;// R1 is pointer for Port E
     LDR R0, [R1]                   ;// R0 contains original input values


     LSR R0, R0, #1                 ;// shift bits over to the right, this allows
     MVN R3, R0                     ;// flips bits R3 = NOT(PE3) in LED location

```

TExaS Lab 1

Port E Hardware

Port E Registers

DATA: 0x04   PUR: 0x00   LOCK: 0x01
DIR: 0x04   PDR: 0x00   CR: 0xFF
DEN: 0x3F   RCGCGPIO: 0x00000010   Clock enabled

Grading Controls

Number from EdX:

Score: 100

Copy this to EdX: ApBieGdm

```
2) SW1 pressed, SW2 pressed, and SW3 pressed
    Pass: All switches pressed has LED on
3) SW1/SW3 pressed and SW2 not pressed
    Pass: Switch2 not pressed has LED off
4) SW1 not pressed and SW2/SW3 pressed
    Pass: Switch1 not pressed has LED off
5) SW1/SW2 pressed and SW3 not pressed
    Pass: Switch3 not pressed has LED off
6) All three switches pressed
    Pass: All switches pressed has LED on
Done grading. Score is 100

>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR Display Enter EVALuate
```

Address: 0x400253FC

```
0x400253FC: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x4002541E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x40025440: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x40025462: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x40025484: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x400254A6: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x400254C8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x400254EA: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF 00 00 00 00 00 00 00 00
0x4002550C: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 01 01 1E 1E 1E 1E 00 00 00 00 00 00
0x4002552E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x40025550: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Simulation   t1: 0.00005006 sec   L:49 C:1   CAP NUM SCRL OVR R/W

```
;****************** main.s ***************
; Program written by: ***Elvin J. Galarza***
; Date Created: 1/22/2016
; Last Modified: 9/6/2016
; Section ***Tuesday 3-4***
; Instructor: ***Vijay Janapa Redi***
; Lab number: 1
; Brief description of the program
; The overall objective of this system is a digital lock
; Hardware connections
;  PE3 is switch input  (1 means switch is not pressed, 0 means switch is pressed)
;  PE4 is switch input  (1 means switch is not pressed, 0 means switch is pressed)
;  PE5 is switch input  (1 means switch is not pressed, 0 means switch is pressed)
;  PE2 is LED output (0 means door is locked, 1 means door is unlocked)
; The specific operation of this system is to
;   unlock if all three switches are pressed


GPIO_PORTE_DATA_R       EQU   0x400243FC
GPIO_PORTE_DIR_R        EQU   0x40024400
GPIO_PORTE_AFSEL_R      EQU   0x40024420
GPIO_PORTE_DEN_R        EQU   0x4002451C
GPIO_PORTE_AMSEL_R      EQU   0x40024528
GPIO_PORTE_PCTL_R       EQU   0x4002452C
SYSCTL_RCGCGPIO_R       EQU   0x400FE608
```

```
    AREA    |.text|, CODE, READONLY, ALIGN=2

    THUMB

    EXPORT  Start

Start

PortE_Init                                                              ;// Steps for initiating Port E

        LDR R1, =SYSCTL_RCGCGPIO_R    ;// Step 1) following lines activate clock for Port E (this line gets the address of SYSCTL_RCGCGPIO_R and puts it into R1)

        LDR R0, [R1]                                    ;// gets the data stored at the address of SYSCTL_RCGCGPIO_R and puts it into R0

        ORR R0, R0, #0x10                               ;// sets bit 4 of SYSCTL_RCGCGPIO_R's data to turn on Port E's clock

        STR R0, [R1]                                    ;// puts modified data back into R1, indicating we only want to deal with Port E

        NOP

        NOP                                                             ;// allows time for clock to finish

        LDR R1, =GPIO_PORTE_DIR_R       ;// makes PE2 (LED) high

        MOV R0, #0x04

        STR R0, [R1]

        LDR R1, =GPIO_PORTE_DEN_R       ;// enables Port E digital port (we're only working with the virtual "digital" port, not the physical "analog" one)

        LDR R0, [R1]

        MOV R0, #0x0FF

        STR R0, [R1]




loop
```

```
                                                          ;// Here is where the PE2 = (not(PE3))
and (not(PE4)  and (not(PE5)) function is implemented

     LDR R1, =GPIO_PORTE_DATA_R   ;// R1 is pointer for Port E

     LDR R0, [R1]                             ;// R0 contains original input values



     LSR R0, R0, #1                           ;// shift bits over to the right, this allows PE3 to be at
LED pin

     MVN R3, R0                                    ;// flips bits R3 = NOT(PE3) in LED location


     LDR R0, [R1]

     LSR R0, R0, #2

     MVN R4, R0                                    ;// R4 = NOT(PE4) in LED location


     LDR R0, [R1]

     LSR R0, R0, #3

     MVN R5, R0                                    ;// R5 = NOT(PE5) in LED location


     AND R6, R3, R4                           ;// AND after NOT(PE3-5), where R6 =(NOT(PE3)) AND
(NOT(PE4)) for line 64

     AND R6, R5, R6


     STR R6, [R1]                             ;// at this point, R6 will be holding the correct value for
the LED pin and all this does is store that value into the DATA for PortE


      B   loop
```

```
ALIGN       ; make sure the end of this section is aligned

END         ; end of file
```