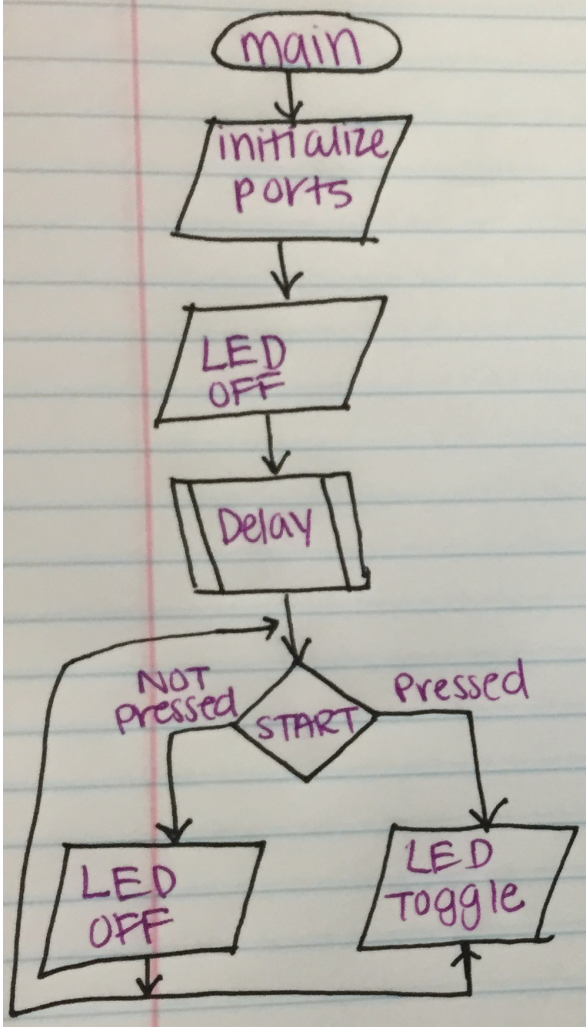
Lab #2

Mira sehgal
& Hyder shad

Flowchart



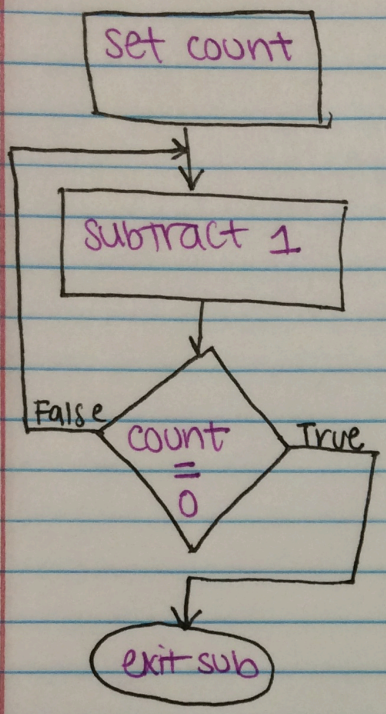
Pseudocode

main
Turn on Port F clock
set direction register
(PF 4 = input)
(PF 3 = output)
Digital enable PF 4/3
Enable PUR on pin 4
set PF3 = 0 → LED OFF

loop
Delay 100 ms
Read switch to see if pressed
IF PF4 = 0, toggle LED
IF PF4 = 1, loop

Mira sehgal
Hyder shad

Flowchart Delay



Pseudocode Delay

1. set a value to count down from
2. subtract one
3. compare count with 0
   True = exit subroutine
   False = loop to subtract 1 again
4. exit subroutine

Delay Subroutine 100ms delay Calculation:

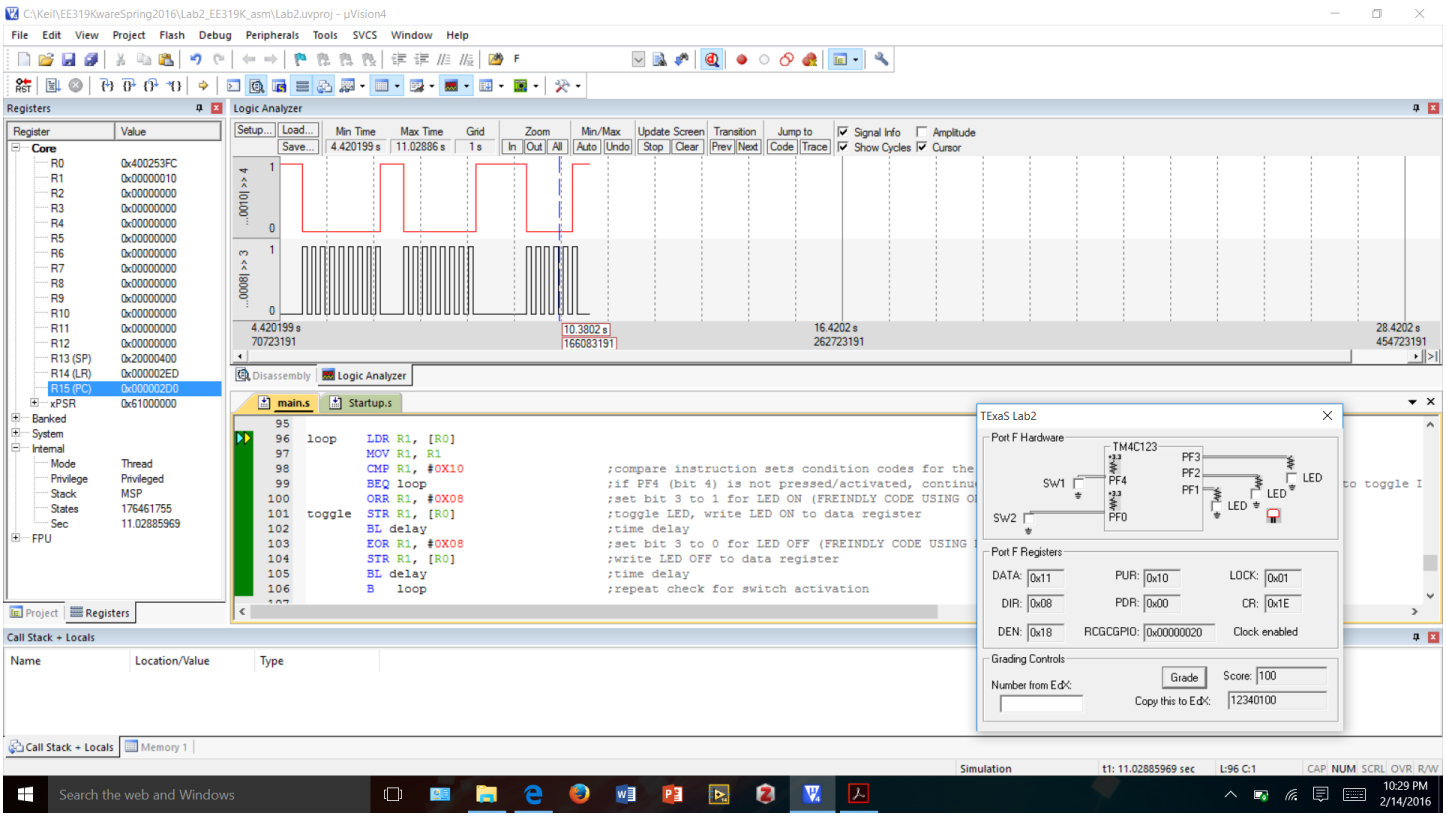
```
delay   LDR R3, = countdown
count   ADD R3, #-1
        CMP R3, #0
        BNE count
        BX  LR
```

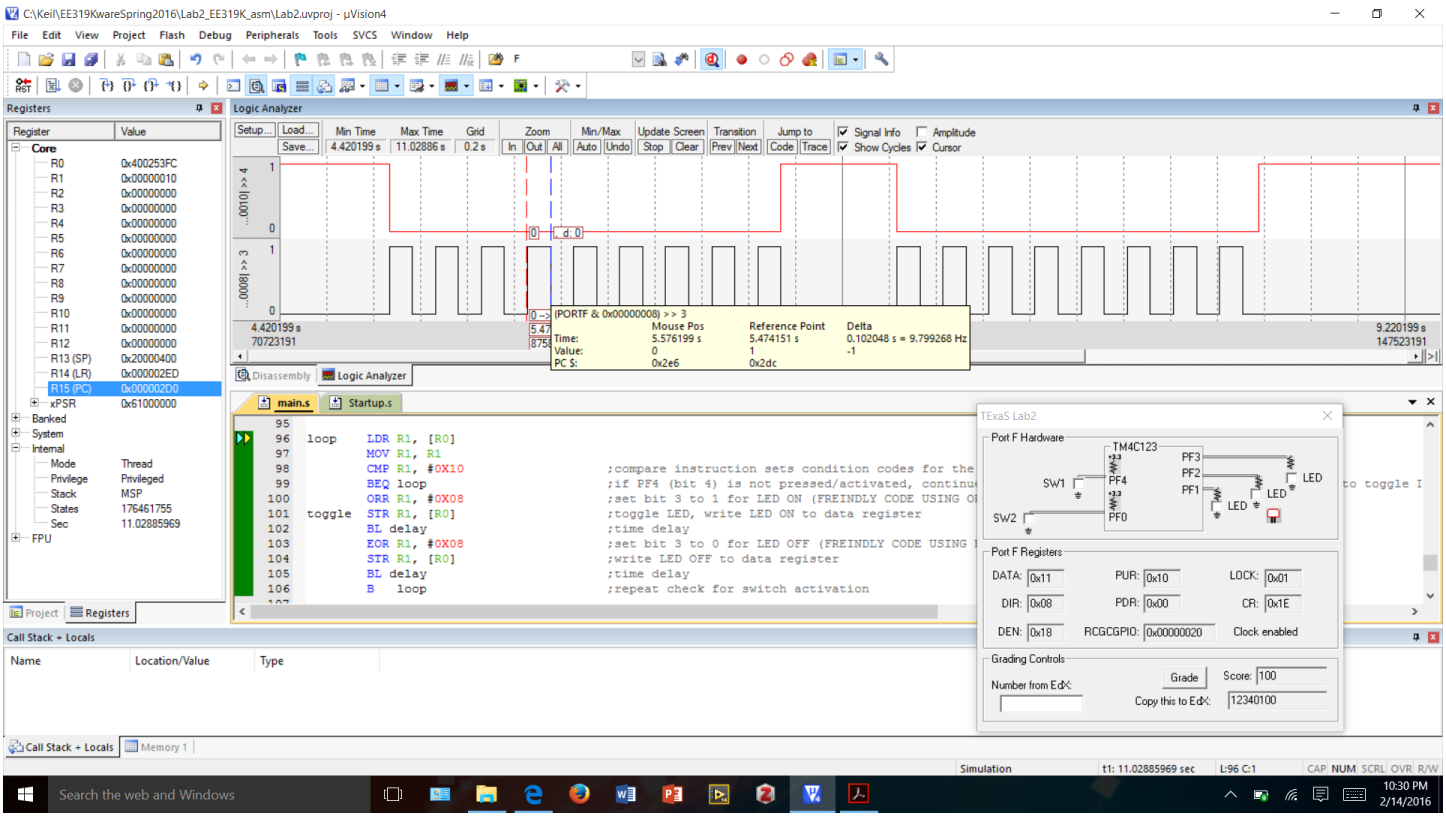
delay takes 5 cycles to complete

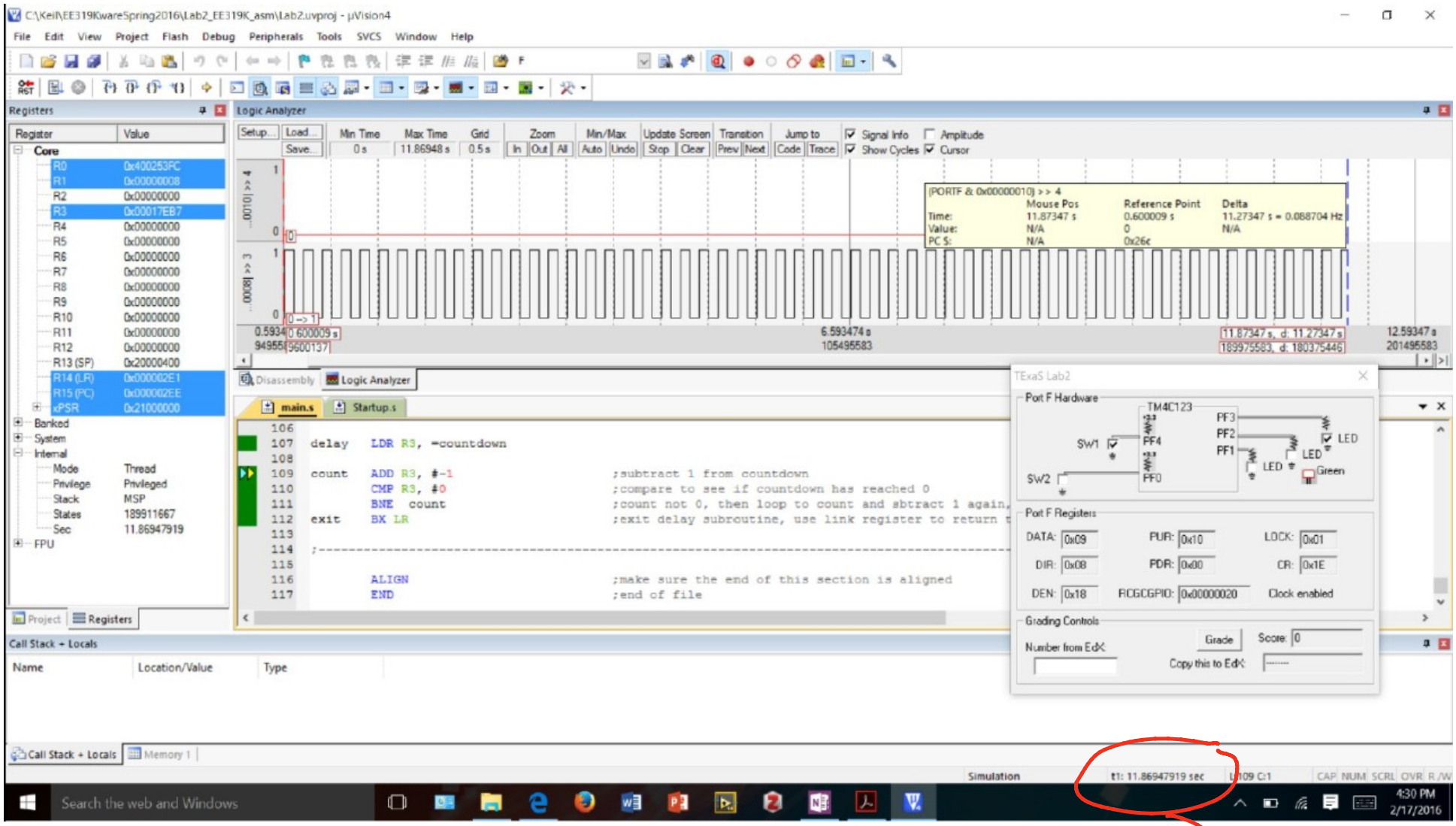
1 cycle = 0.0625 μs
5 cycles = 0.3125 μs

$\frac{100\ ms}{0.3125\ \mu s} = 320,000$ repetitions of the delay required to achieve 100ms toggle delay

countdown = 320,000 = 0x4E200


C:\Keil\EE319KwareSpring2016\Lab2_EE319K_asm\Lab2.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Register Value
Core
R0 0x400253FC
R1 0x00000010
R2 0x00000000
R3 0x00000000
R4 0x00000000
R5 0x00000000
R6 0x00000000
R7 0x00000000
R8 0x00000000
R9 0x00000000
R10 0x00000000
R11 0x00000000
R12 0x00000000
R13 (SP) 0x20000400
R14 (LR) 0x000002ED
R15 (PC) 0x000002D0
xPSR 0x61000000
Banked
System
Internal
Mode Thread
Privilege Privileged
Stack MSP
States 176461755
Sec 11.02885969
FPU
Logic Analyzer
Setup... Load... Save...
Min Time 4.420199 s
Max Time 11.02886 s
Grid 1 s
Zoom In Out All
Min/Max Auto Undo
Update Screen Stop Clear
Transition Prev Next
Jump to Code Trace
Signal Info
Show Cycles
Amplitude
Cursor
4.420199 s
70723191
10.3802 s
166083191
16.4202 s
262723191
28.4202 s
454723191
Disassembly
Logic Analyzer
main.s
Startup.s
95
96 loop LDR R1, [R0]
97 MOV R1, R1
98 CMP R1, #0X10 ;compare instruction sets condition codes for the
99 BEQ loop ;if PF4 (bit 4) is not pressed/activated, continu
100 ORR R1, #0X08 ;set bit 3 to 1 for LED ON (FREINDLY CODE USING O
101 toggle STR R1, [R0] ;toggle LED, write LED ON to data register
102 BL delay ;time delay
103 EOR R1, #0X08 ;set bit 3 to 0 for LED OFF (FREINDLY CODE USING
104 STR R1, [R0] ;write LED OFF to data register
105 BL delay ;time delay
106 B loop ;repeat check for switch activation
to toggle I
TExaS Lab2
Port F Hardware
TM4C123
SW1
SW2
PF4
PF0
PF3
PF2
PF1
LED
LED
LED
Port F Registers
DATA: 0x11
PUR: 0x10
LOCK: 0x01
DIR: 0x08
PDR: 0x00
CR: 0x1E
DEN: 0x18
RCGCGPIO: 0x00000020
Clock enabled
Grading Controls
Number from EdX:
Grade
Score: 100
Copy this to EdX:
12340100
Call Stack + Locals
Name Location/Value Type
Call Stack + Locals
Memory 1
Project
Registers
Simulation
t1: 11.02885969 sec
L:96 C:1
CAP NUM SCRL OVR R/W
Search the web and Windows
10:29 PM
2/14/2016


C:\Keil\EE319KwareSpring2016\Lab2_EE319K_asm\Lab2.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Register Value
Core
R0 0x400253FC
R1 0x00000010
R2 0x00000000
R3 0x00000000
R4 0x00000000
R5 0x00000000
R6 0x00000000
R7 0x00000000
R8 0x00000000
R9 0x00000000
R10 0x00000000
R11 0x00000000
R12 0x00000000
R13 (SP) 0x20000400
R14 (LR) 0x000002ED
R15 (PC) 0x000002D0
xPSR 0x61000000
Banked
System
Internal
Mode Thread
Privilege Privileged
Stack MSP
States 176461755
Sec 11.02885969
FPU
Logic Analyzer
Setup... Load... Save...
Min Time 4.420199 s
Max Time 11.02886 s
Grid 0.2 s
Zoom In Out All
Min/Max Auto Undo
Update Screen Stop Clear
Transition Prev Next
Jump to Code Trace
Signal Info
Show Cycles
Amplitude
Cursor
(PORTF & 0x00000008) >> 3
Mouse Pos
Reference Point
Delta
Time: 5.576199 s 5.474151 s 0.102048 s = 9.799268 Hz
Value: 0 1 -1
PC $: 0x2e6 0x2dc
4.420199 s
70723191
9.220199 s
147523191
Disassembly
Logic Analyzer
main.s
Startup.s
95
96 loop LDR R1, [R0]
97 MOV R1, R1
98 CMP R1, #0X10 ;compare instruction sets condition codes for the
99 BEQ loop ;if PF4 (bit 4) is not pressed/activated, continu
100 ORR R1, #0X08 ;set bit 3 to 1 for LED ON (FREINDLY CODE USING O
101 toggle STR R1, [R0] ;toggle LED, write LED ON to data register
102 BL delay ;time delay
103 EOR R1, #0X08 ;set bit 3 to 0 for LED OFF (FREINDLY CODE USING
104 STR R1, [R0] ;write LED OFF to data register
105 BL delay ;time delay
106 B loop ;repeat check for switch activation
to toggle I
TExaS Lab2
Port F Hardware
TM4C123
SW1
SW2
PF4
PF0
PF3
PF2
PF1
LED
LED
LED
Port F Registers
DATA: 0x11 PUR: 0x10 LOCK: 0x01
DIR: 0x08 PDR: 0x00 CR: 0x1E
DEN: 0x18 RCGCGPIO: 0x00000020 Clock enabled
Grading Controls
Number from EdX:
Grade
Score: 100
Copy this to EdX: 12340100
Call Stack + Locals
Name Location/Value Type
Call Stack + Locals Memory 1
Simulation t1: 11.02885969 sec L:96 C:1 CAP NUM SCRL OVR R/W
Search the web and Windows
10:30 PM
2/14/2016



Simulator run time is 11.86947919 seconds for 10.06 seconds of real time.

11.86947919s/10.06s = 1.179868707x faster than real time.

```
;****************** main.s ***************
; Program written by: HYDER SHAD, MIRA SEHGAL
; Date Created: 1/22/2016
; Last Modified: 2/7/2016 12:58 AM
; Section: THURSDAY 2-3 PM
; Instructor: V. JANAPA
; Lab number: 2
; Brief description of the program
; The overall objective of this system an interactive alarm
; Hardware connections
;  PF4 is switch input  (1 means SW1 is not pressed, 0 means SW1 is
pressed)
;  PF3 is LED output (1 activates green LED)
; The specific operation of this system
;    1) Make PF3 an output and make PF4 an input (enable PUR for PF4).
;    2) The system starts with the LED OFF (make PF3 =0).
;    3) Delay for about 100 ms
;    4) If the switch is pressed (PF4 is 0), then toggle the LED once,
else turn the LED OFF.
;    5) Repeat steps 3 and 4 over and over

SYSCTL_RCGCGPIO_R  EQU 0x400FE608
GPIO_PORTF_DATA_R  EQU 0x400253FC
GPIO_PORTF_DIR_R   EQU 0x40025400
GPIO_PORTF_AFSEL_R EQU 0x40025420
GPIO_PORTF_PUR_R   EQU 0x40025510
GPIO_PORTF_DEN_R   EQU 0x4002551C
GPIO_PORTF_LOCK_R  EQU 0x40025520
GPIO_PORTF_CR_R    EQU 0x40025524
GPIO_PORTF_AMSEL_R EQU 0x40025528
GPIO_PORTF_PCTL_R  EQU 0x4002552C
GPIO_LOCK_KEY      EQU 0x4C4F434B
countdown EQU 0x4E200 ; delay subroutine countdown set to 320,000 to
achieve 100ms toggle delay
       AREA    |.text|, CODE, READONLY, ALIGN=2
       THUMB
       EXPORT  Start

;-------------------------------------------------------------------
-------------------------------------------------------------


INIT

; 1) activate clock for Port F
    LDR R0, =SYSCTL_RCGCGPIO_R
    LDR R1, [R0]
    ORR R1, #0X20       ;SET BIT 5 HIGH TO ENABLE PORT F CLOCK
    STR R1, [R0]
    NOP
```

```
    NOP                 ;ALLOW TIME TO FINISH ACTIVATING (2+ CYCLES)
    NOP

; 2) enable pull-up resistors for negative logic

        LDR R0, =GPIO_PORTF_PUR_R  ;PUR LIMITS CURRENT APPLIED TO PIN
ON THE MICROCONTROLLER
        LDR R1, [R0]
        ORR R1, #0X10           ;PULL UP RESISTOR ON BIT 4 BECAUSE PF4
IS INPUT USING NEGATIVE LOGIC
        STR R1, [R0]

; 3) set direction register

        LDR R0, =GPIO_PORTF_DIR_R     ;SET BIT 3 HIGH FOR IT TO BE
OUTPUT
        LDR R1, [R0]
        ORR R1, #0X08
        STR R1, [R0]

; 4) disable analog functionality

        LDR R0, =GPIO_PORTF_AMSEL_R  ;DISABLE ANALOG CAPABILITIES,
SOLEY DIGITAL I/O
        MOV R1, #0X0
        STR R1, [R0]

; 5) configure as GPIO

        LDR R0, =GPIO_PORTF_PCTL_R
        MOV R1, #0X0            ;CLEAR PORT CONTROL FIELD TO SET UP
PINS FOR GPIO
        STR R1, [R0]

; 6) regular port function

        LDR R0, =GPIO_PORTF_AFSEL_R  ;DISABLE ALT FUNCTIONS FOR PIN BY
SETTING BITS TO ZERO
        MOV R1, #0
        STR R1, [R0]

; 7) enable digital port

        LDR R0, =GPIO_PORTF_DEN_R              ;R1 =
&GPIO_PORTD_DEN_R
        ORR R1, #0X18                          ;ENABLE DIGITAL I/O ON
PINS 5-2
        STR R1, [R0]

; 8) exit initialization process
```

```
        BX LR

;-------------------------------------------------------------------
--------------

Start   BL INIT                         ;initialize port f for I/O
        LDR R0, =GPIO_PORTF_DATA_R      ;load R0 with address for port
f data register


loop    LDR R1, [R0]
        CMP R1, #0X10   ;compare instruction sets condition codes for
the following conditional branch instruction
        BEQ loop        ;if PF4 (bit 4) is not pressed/activated,
continue checking bit, else if PF4 is pressed, continue to toggle LED
        ORR R1, #0X08           ;set bit 3 to 1 for LED ON (FREINDLY
CODE USING ORR)
toggle  STR R1, [R0]    ;toggle LED, write LED ON to data register

        BL delay                                                ;time
delay
        EOR R1, #0X08   ;set bit 3 to 0 for LED OFF (FREINDLY CODE
USING EOR)
        STR R1, [R0]    ;write LED OFF to data register
        BL delay        ;time delay
        B   loop        ;repeat check for switch activation


delay   LDR R3, =countdown

count   ADD R3, #-1     ;subtract 1 from countdown

        CMP R3, #0      ;compare to see if countdown has reached 0
        BNE  count      ;count not 0, then loop to count and sbtract 1
again, count = 0, then exit subroutine
exit    BX LR           ;exit delay subroutine, use link register to
return to main program code

;-------------------------------------------------------------------
-----------------------------------------------------------

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