



Faculty of Engineering
Ain Shams University
CSE 211s: Introduction to Embedded Systems
REPORT (3)

Name: Habiba El-Sayed Mowafy

Sec.: 1

Program: CSE

ID: 2100792

Question 2:

Q2. TIVAC LaunchPad has two build-in switches SW 1 (PF4) and SW 2(PF0), Three LEDs (Red (PF1), Blue (PF2), Green (PF3)).

- If both switches are pressed, → turn on the Red LED.
- If SW1 is pressed, SW2 is not pressed → turn on the Blue LED.
- SW2 is pressed, SW1 is not pressed → turn on the Green LED.

Check through the simulated Kit that the behavior of your code is correct.

Header file (IO.h):

```
#define GPIO_PORTF_DATA_R      (*(volatile unsigned long *)0x400253FC))
#define GPIO_PORTF_DIR_R      (*(volatile unsigned long *)0x40025400))
#define GPIO_PORTF_AFSEL_R    (*(volatile unsigned long *)0x40025420))
#define GPIO_PORTF_PUR_R      (*(volatile unsigned long *)0x40025510))
#define GPIO_PORTF_DEN_R      (*(volatile unsigned long *)0x4002551C))
#define GPIO_PORTF_LOCK_R     (*(volatile unsigned long *)0x40025520))
#define GPIO_PORTF_CR_R       (*(volatile unsigned long *)0x40025524))
#define GPIO_PORTF_AMSEL_R    (*(volatile unsigned long *)0x40025528))
#define GPIO_PORTF_PCTL_R     (*(volatile unsigned long *)0x4002552C))
#define PF4                    (*(volatile unsigned long *)0x40025040))
#define PF3                    (*(volatile unsigned long *)0x40025020))
#define PF2                    (*(volatile unsigned long *)0x40025010))
#define PF1                    (*(volatile unsigned long *)0x40025008))
#define PF0                    (*(volatile unsigned long *)0x40025004))
#define GPIO_LOCK_KEY         0x4C4F434B // Unlocks the GPIO_CR register
#define SYSCTL_RCGCGPIO_R     (*(volatile unsigned long *)0x400FE608))
#define SYSCTL_PRGPIO_R       (*(volatile unsigned long *)0x400FEA08))
#define RED                    0x02
#define BLUE                   0x04
#define GREEN                  0x08
#define PF123_mask            0x0E
#define PF04_mask             0x11
#define PF_mask               0x20
#define PF_SW1_mask           0x10
#define PF_SW2_mask           0x01
```

C code:

```
#include "IO.h"
```

```
void RGB_Init(void) {
    SYSTCL_RCGCGPIO_R |= PF_mask; //enable clk for port F
    while ((SYSTCL_PRGPIO_R & PF_mask) == 0); //check for the clk delay
    GPIO_PORTF_LOCK_R = GPIO_LOCK_KEY; //break the lock of port F
    GPIO_PORTF_CR_R |= PF123_mask; //enable making changes in leds pins (PF1,PF2&PF3)
    GPIO_PORTF_AMSEL_R &= ~PF123_mask; //disable analog mode
    GPIO_PORTF_AFSEL_R &= ~PF123_mask; //disable using alternate functions (use the
pins as GPIO)
    GPIO_PORTF_PCTL_R &= ~0x0000FFF0; //clear the bits corresponfing to the 3 pins
    GPIO_PORTF_DEN_R |= PF123_mask; //enable using the pins in digital mode
    GPIO_PORTF_DIR_R |= PF123_mask; //set the bits corresponding to the pins to 1 to
use them as output
    GPIO_PORTF_DATA_R &= ~PF123_mask;
}

void Switches_Init(void) {
    GPIO_PORTF_LOCK_R = GPIO_LOCK_KEY;
    GPIO_PORTF_CR_R |= PF04_mask; //enable making changes in switches pins (PF0&PF4)
    GPIO_PORTF_AMSEL_R &= ~PF04_mask;
    GPIO_PORTF_AFSEL_R &= ~PF04_mask;
    GPIO_PORTF_PCTL_R &= ~0x000F000F; //clear the bits corresponfing to the 2
switches
    GPIO_PORTF_DEN_R |= PF04_mask;
    GPIO_PORTF_DIR_R &= ~PF04_mask; //set the bits corresponding to the switches to 0
to use them as input
    GPIO_PORTF_PUR_R |= PF04_mask;
}

unsigned char SW1_Input(void) {
    char pressed1 = GPIO_PORTF_DATA_R & PF_SW1_mask;
    return pressed1;
}

unsigned char SW2_Input(void) {
    char pressed2 = GPIO_PORTF_DATA_R & PF_SW2_mask;
    return pressed2;
}

void RED_Output(unsigned char data) {
    GPIO_PORTF_DATA_R &= ~PF123_mask;
    GPIO_PORTF_DATA_R |= data;
}

void BLUE_Output(unsigned char data) {
    GPIO_PORTF_DATA_R &= ~PF123_mask;
    GPIO_PORTF_DATA_R |= data;
}
```

```

void GREEN_Output(unsigned char data){
    GPIO_PORTF_DATA_R &= ~PF123_mask;
    GPIO_PORTF_DATA_R |= data;}

void No_Led(){
    GPIO_PORTF_DATA_R &= 0x00;
    //GPIO_PORTF_DATA_R |= data;
}

unsigned char led;
signed char button1_in;
signed char button2_in;
int main(){
    RGB_Init();
    Switches_Init();
    while(1){
        button1_in = SW1_Input(); //store the return value of SW1_Input
        button2_in = SW2_Input(); //store the return value of SW2_Input
        if(button1_in && button2_in){
            No_Led();
        }
        if(!button1_in && !button2_in){
            led = RED;
            RED_Output(led);
        }
        else if(!button1_in && button2_in){
            led = BLUE;
            BLUE_Output(led);
        }
        else if(button1_in && !button2_in){
            led = GREEN;
            GREEN_Output(led);
        }
    }
}

```

SnapShots:

Blue Led>>

The left screenshot shows the 'Port F Hardware' configuration for a TM4C123. It displays a circuit diagram with two switches (SW1, SW2) connected to pins PF4, PF0, PF3, PF2, and PF1. A 16 MHz clock is connected to the LED pins. The 'Port F Registers' section shows the following values:

DATA: 0x05	PUR: 0x11	LOCK: 0x00
DIR: 0x0E	PDR: 0x00	CR: 0x1F
DEN: 0x1F	RCGCGPIO: 0x00000020	Clock enabled

The right screenshot shows the 'GPIOF' register window. The 'DATA' register is highlighted with a value of 0x00000005. The 'LOCK' register is also highlighted with a value of 0x00000000.

Green Led>>

The screenshot shows the TExaS LaunchPadDLL window with the Port F Hardware diagram and the Port F Registers section. The hardware diagram shows a TM4C123 microcontroller with pins PF3, PF2, PF1, PF4, and PF0. A 16 MHz clock is connected to PF3. A switch SW1 is connected to PF4, and a switch SW2 is connected to PF0. Two LEDs are connected to PF1 and PF2, labeled 'LED' and 'Green'. The Port F Registers section shows the following values: DATA: 0x18, PUR: 0x11, LOCK: 0x00, DIR: 0x0E, PDR: 0x00, CR: 0x1F, DEN: 0x1F, RCGCGPIO: 0x00000020, and Clock enabled.

The GPIOF window shows the following values:

Property	Value
DATA	0x00000018
DIR	0x0000000E
IS	0x00000000
IBE	0x00000000
IEV	0x00000000
IM	0
RIS	0x00000011
MIS	0
ICR	0
AFSEL	0x00000000
DR2R	0x000000FF
DR4R	0x00000000
DR8R	0x00000000
ODR	0x00000000
PUR	0x00000011
PDR	0x00000000
SLR	0x00000000
DEN	0x0000001F
LOCK	0x00000000
CR	0x0000001F
AMSEL	0x00000000

Code snippets in the editor:

```
>return the return value of SW1_Input  
>return the return value of SW2_Input
```

Red Led>>

The screenshot shows the TExaS LaunchPadDLL window with the Port F Hardware diagram and the Port F Registers section. The hardware diagram shows a TM4C123 microcontroller with pins PF3, PF2, PF1, PF4, and PF0. A 16 MHz clock is connected to PF3. A switch SW1 is connected to PF4, and a switch SW2 is connected to PF0. Two LEDs are connected to PF1 and PF2, labeled 'LED' and 'Red'. The Port F Registers section shows the following values: DATA: 0x02, PUR: 0x11, LOCK: 0x00, DIR: 0x0E, PDR: 0x00, CR: 0x1F, DEN: 0x1F, RCGCGPIO: 0x00000020, and Clock enabled.

The GPIOF window shows the following values:

Property	Value
DATA	0x00000002
DIR	0x0000000E
IS	0x00000000
IBE	0x00000000
IEV	0x00000000
IM	0
RIS	0x00000011
MIS	0
ICR	0
AFSEL	0x00000000
DR2R	0x000000FF
DR4R	0x00000000
DR8R	0x00000000
ODR	0x00000000
PUR	0x00000011
PDR	0x00000000
SLR	0x00000000
DEN	0x0000001F
LOCK	0x00000000
CR	0x0000001F
AMSEL	0x00000000

Code snippets in the editor:

```
.) {  
  
.a) {
```