INTERFACING OF RGB, STEPPER MOTER, DC MOTOR

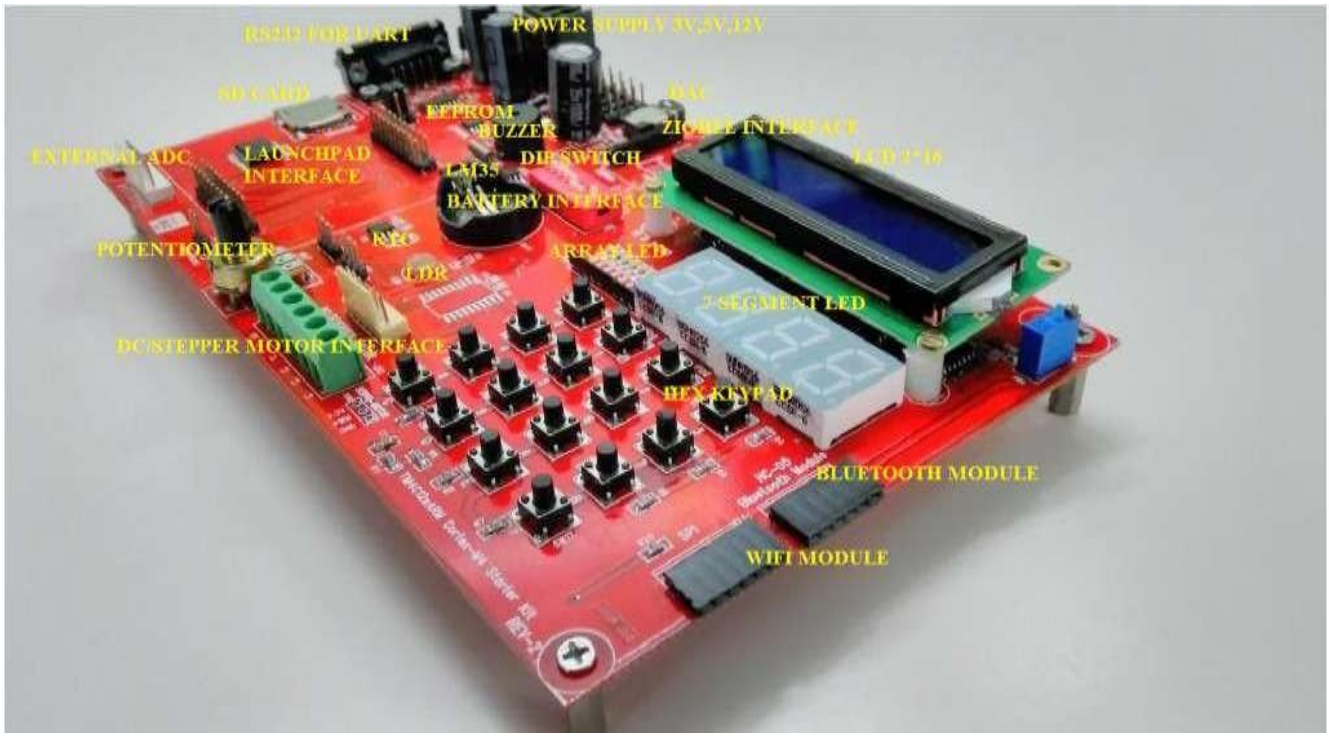
interfaceing the DC motor, RGB LED, and a stepper motor to the Tiva C microcontroller TM4C123GH6PM.

DC Motor: The motor is interfaced with PF3 to turn it on for 500 ms using an infinite loop.
RGB LED: The RGB LED changes color with red on PF1, blue on PF2, and green on PF3. The color remains on for 500 ms.
Stepper Motor: The motor steps through four phases, which energize PF2, PF3, PB3, and PC4 in sequence, separated by 500 ms.
Each peripheral is GPIO port-configured, clock gating enabled, and digital functions activated. The timings used by all three devices are controlled by a singular delayMs function.

Applications of each of the devices used are as follows:

DC Motor
Robotics: To provide actuation to wheels or mechanical parts in a robot
Conveyor Belts: To control motors with the view of enabling the movement of items within automated systems.
RGB LED
Status Indicators: Color-coded feedback in devices, such as charging, error, and active states.
Decorative Lighting: Ambient lighting systems or displays.
Stepper Motor
CNC Machines: It provides precise control for cutting, drilling, or engraving.
3D Printers: Stepping motors are applied to achieve precise positioning of the print head and bed.
These components can be put into automation, robotics, and/or embedded systems for precise control, feedback, or signaling.

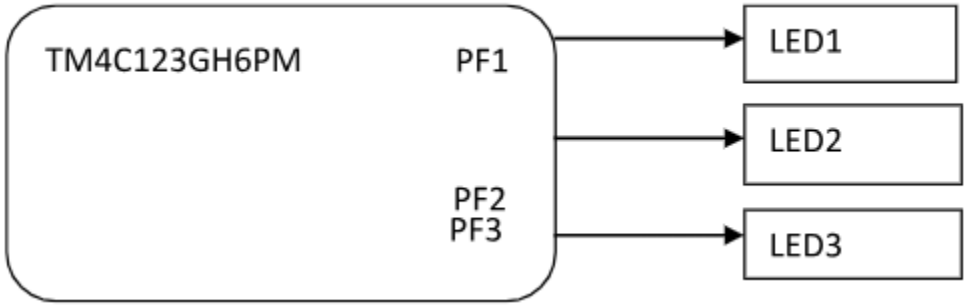
TIVA C Series Launch Pad



Features:

- 12v dc voltage is used to power up the board
- On board DAC with stereo output connector
- On 16*2 LCD Module
- On board module to connect wireless Devices such as ***Zigbee, Bluetooth, RF and Wi-Fi***
- Slot for micro SD card
- Analog Hex keypad(4*4)
- 7-Segment Display
- LED Array
- On board RTC with Battery module
- On board 4 channel ADC
- Port to connect External thermocouple
- LDR
- Motor Driver with connecting Pins
- Potentiometer
- Buzzer
- Temperature sensor (LM35)
- Port for UART Communication

Interfacing of RGB LED



OUTPUT PORTS:
PORT F Pin1 – LED1
PORT F Pin2 – LED2
PORT F Pin3 – LED3

Expected Output: RED, GREEN & BLUE Lights in Launch Pad blinks in accordance with delay

RGB:

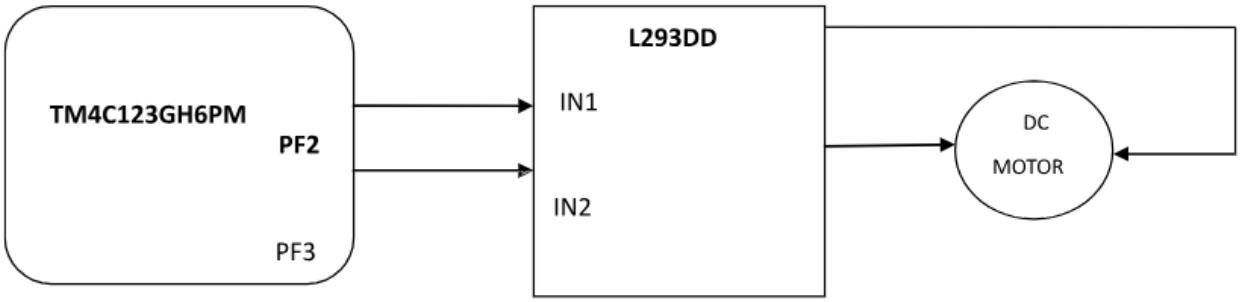
```
                                        /* RGB LED */
#include <stdint.h>
#include "inc\tm4c123gh6pm.h"
void delayMs(int n);
int main(void)
{
/* enable clock to GPIOF/B/C at clock gating control register */
SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R5;//PF
/* enable the GPIO pins for the PF2/PF3/PB3/PC4 as output */
GPIO_PORTF_DIR_R = 0x0E;//PF2/3
/* enable the GPIO pins for digital function */
GPIO_PORTF_DEN_R = 0x0E;
while(1)
{
GPIO_PORTF_DATA_R = 0x02;
delayMs(500);

GPIO_PORTF_DATA_R = 0x04;

delayMs(500);
GPIO_PORTF_DATA_R = 0x08;
```

```
delayMs(500);
}}
/* delay n milliseconds (16 MHz CPU clock) */
void delayMs(int n)
{
int i, j;
for(i = 0 ; i < n; i++)
for(j = 0; j < 4180; j++)
{} /* do nothing for 1 ms */
}
```

Interfacing of DC Motor Using L29DD



(**TM4C123GH6PM**)

- PORTF Pin2 Input 1
- PORTF Pin3 Input 2

INPUT PINS (L293DD)

- IN1 (Pin No.2)
- IN2 (Pin No.7)

OUTPUT PINS (L293DD)

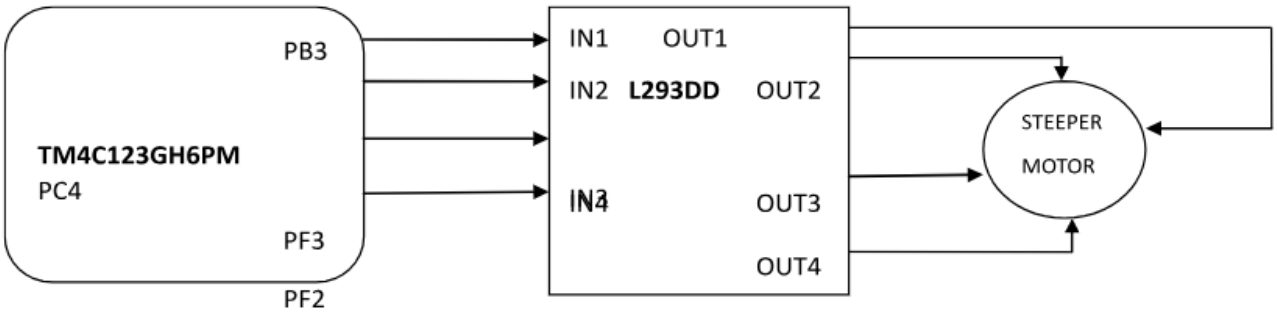
- L1
- L2

DC Motor:

```
/* DC_Motor _CLK*/
#include <stdint.h>
#include "inc\tm4c123gh6pm.h"
void delayMs(int n);
int main(void)
```

```
{
/* enable clock to GPIOF/B/C at clock gating control register */
SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R5;//PF
/* enable the GPIO pins for the PF2/PF3/PB3/PC4 as output */
GPIO_PORTF_DIR_R = 0x0E;//PF2/3
/* enable the GPIO pins for digital function */
GPIO_PORTF_DEN_R = 0x0E;
while(1)
{
GPIO_PORTF_DATA_R = 0x08;
delayMs(500);
} }
/* delay n milliseconds (16 MHz CPU clock) */
void delayMs(int n)
{
int i, j;
for(i = 0 ; i < n; i++)
for(j = 0; j < 4180; j++)
{} /* do nothing for 1 ms */
}
```

Interfacing of Stepper Motor Using L29DD



(**TM4C123GH6PM**)

- PORTF Pin2 Input 1
- PORTF Pin3 Input 2
- PORTB Pin3 Input 3
- PORTC Pin4 Input 4

INPUT PINS (L293DD)

- IN1 (Pin No.2)
- IN2 (Pin No.7)
- IN3 (Pin No.10)
- IN4 (Pin No.15)

OUTPUT PINS (L293DD)

- OUT1 (Pin No.3)
- OUT2 (Pin No.6)
- OUT3 (Pin No.11) and OUT4 (Pin No.14)

```
/* Stepper Motor Interface */
#include <stdint.h>
#include "inc\tm4c123gh6pm.h"
void delayMs(int n);
int main(void)
{
/* enable clock to GPIOF/B/C at clock gating control register */
SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R5;//PF
SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R1;//PB
SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R2;//PC
/* enable the GPIO pins for the PF2/PF3/PB3/PC4 as output */
GPIO_PORTF_DIR_R = 0x0E;//PF2/3
GPIO_PORTB_DIR_R = 0x08;//PB3
GPIO_PORTC_DIR_R = 0x10;//PC4
/* enable the GPIO pins for digital function */
GPIO_PORTF_DEN_R = 0x0E;
GPIO_PORTB_DEN_R = 0x08;
GPIO_PORTC_DEN_R = 0x10;
while(1)
{
GPIO_PORTF_DATA_R = 0x04; /* PF2 =1*/
```

```
delayMs(500);
GPIO_PORTF_DATA_R = 0;
delayMs(500);
GPIO_PORTF_DATA_R = 0x08; /* PF3 =1 */
delayMs(500);
GPIO_PORTF_DATA_R = 0;
delayMs(500);
GPIO_PORTB_DATA_R = 0x08; /* PB2 =3 */
delayMs(500);
GPIO_PORTB_DATA_R = 0;
delayMs(500);
GPIO_PORTC_DATA_R = 0x10; /* P2 =1*/
delayMs(500);
GPIO_PORTC_DATA_R = 0;
delayMs(500);
}
}
/* delay n milliseconds (16 MHz CPU clock) */
void delayMs(int n)
{
int i, j;
for(i = 0 ; i < n; i++)
for(j = 0; j < 4180; j++)
{} /* do nothing for 1 ms */
}
```