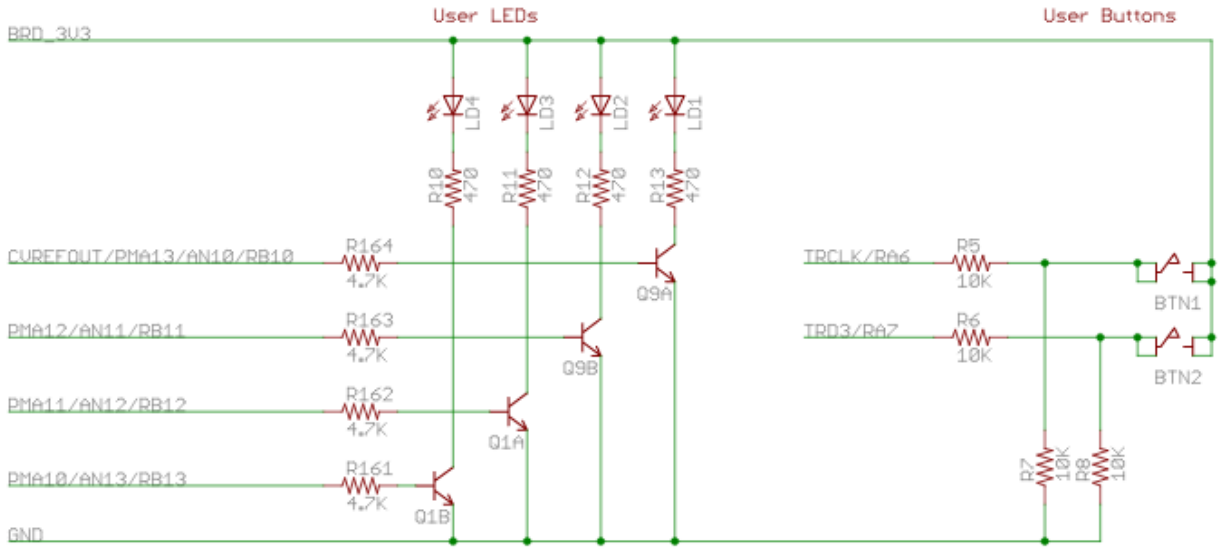

IOPORT_A (inputs – user buttons):

- BTN1: RA6 (BIT_6)
- BTN2: RA7 (BIT_7)

IOPORT_B (outputs – LEDs):

- LD1: RB10 (BIT_10)
- LD2: RB11 (BIT_11)
- LD3: RB12 (BIT_12)
- LD4: RB13 (BIT_13)

Code:

```
/*
 * File:   Blink.c
 * Author: petrusn & ahtouw
 *
 * Created on September 26, 2016, 8:56 AM
 */

#include <stdio.h>
#include <stdlib.h>
#include <plib.h>

/* all ports have names:                          */
/* e.g. IOPORT_A IOPORT_B, etc.                   */
/* all port bits have names:                      */
/* e.g. BIT_0, BIT_1,... BIT_14, BIT_15, etc.     */

/*
 *
 */
int main(int argc, char** argv) {
```

```
    INTConfigureSystem(INT_SYSTEM_CONFIG_MULT_VECTOR);
    INTEnableInterrupts();

    // Useful functions: (see PeripheralLibraries pdf file for more)
    //  PORTSetPinsDigitalOut(IOPORT_B, BIT_10 | BIT_11 | BIT_12 | BIT_13);
    //  PORTSetPinsDigitalIn(IOPORT_B, BIT_10 | BIT_11 | BIT_12 | BIT_13);
    //  PORTClearBits(IOPORT_B, BIT_10 | BIT_11 | BIT_12 | BIT_13);     //
clear bits
    //  PORTSetBits(IOPORT_B, BIT_10 | BIT_11 | BIT_12 | BIT_13);       //
set bits
    //  PORTToggleBits(IOPORT_B, BIT_10 | BIT_11 | BIT_12 | BIT_13);    //
toggle state of the bits
    //  PORTReadBits(IOPORT_A, BIT_6);   // read the state of button on RA6

   // Configure ports for onboard LEDs as outputs
    PORTSetPinsDigitalOut(IOPORT_B, BIT_10 | BIT_11 | BIT_12 | BIT_13);
   // Configure built in buttons as inputs
    PORTSetPinsDigitalIn(IOPORT_A, BIT_6 | BIT_7);

    while (1) // continuous loop
    {
        if (PORTReadBits(IOPORT_A, BIT_6))
        {
            PORTSetBits(IOPORT_B, BIT_10 | BIT_11);
        }
        else
        {
            PORTClearBits(IOPORT_B, BIT_10 | BIT_11);
        }
        if (PORTReadBits(IOPORT_A, BIT_7))
        {
            PORTSetBits(IOPORT_B, BIT_12 | BIT_13);
        }
        else
        {
            PORTClearBits(IOPORT_B, BIT_12 | BIT_13);
        }
    }

    return (EXIT_SUCCESS);
}
```