BECS 31421

EXPERIMENT 02

**Introduction to Input and Output in PIC Programming**

STUDENT NAME: G.P.D. THAMARA

STUDENT NUMBER: EC/2021/005

**Source Code**

sbit sw at RA2\_bit;

void main() {

CMCON = 0x07 ; // Disable Comparator

TRISA = 0x04 ; //configure TRISA register

TRISB = 0x00 ; // Configure TRSB register

PORTB = 0xFF; //Intialize PORTB register

RA2\_bit = 0x00; // Set RA\_2bit to low state

//Enter the first loop

do{

//Check the state of the switch

if(sw==1){

PORTB = 0x00;

//If the switch is pressed ,set PORTB to be low

}

else{

//If the switch is not pressed,set PORTB to be high

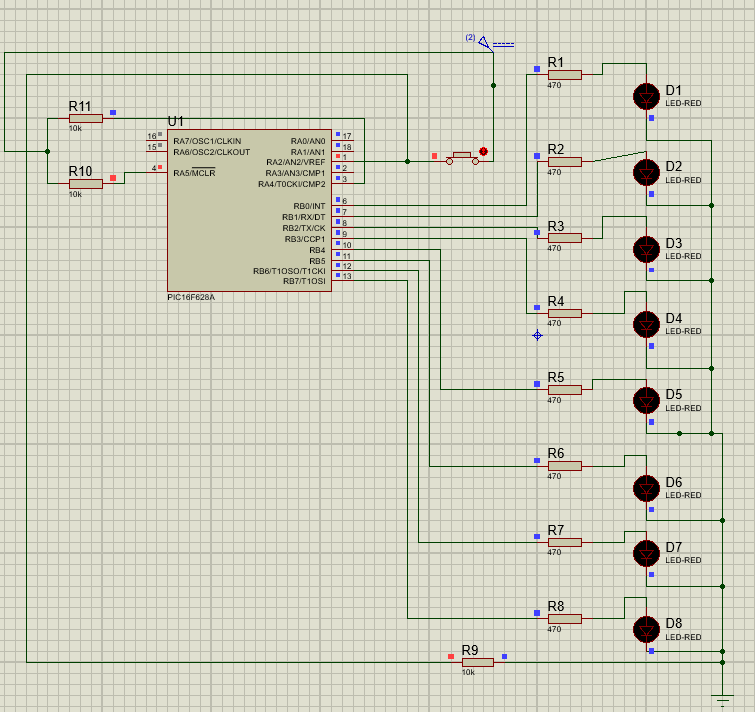
PORTB = 0xFF;

}

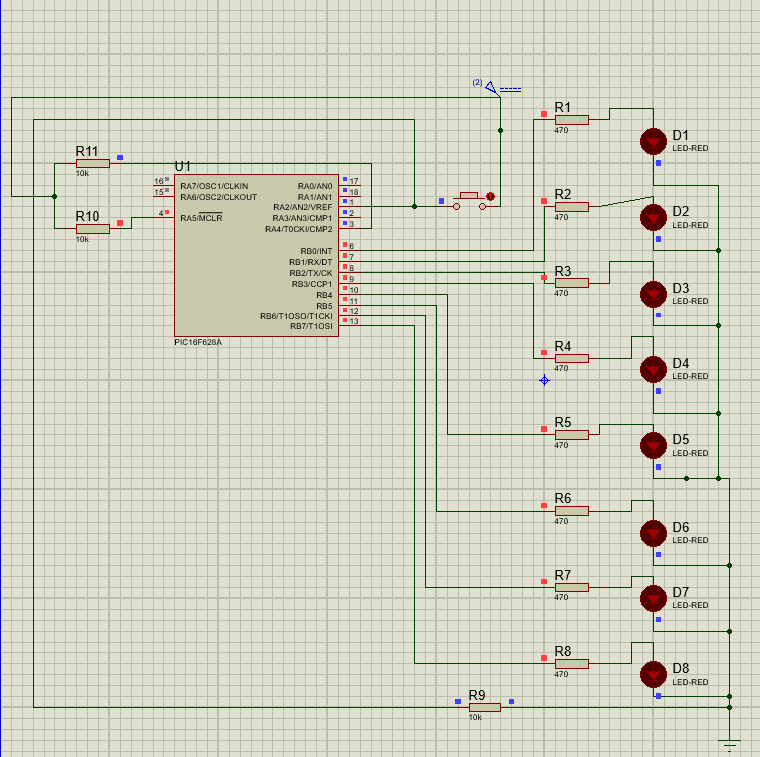
}while(1); // Enter a condition for the infinite loop

}

**When the switch is pressed.**

****

**When the switch is released.**

****

**Discussion**

The experiment, implemented through the provided code, effectively illustrated the critical role of the TRIS register in PIC microcontroller programming. Configuring TRISA = 0x04 and TRISB = 0x00 successfully designated RA2 as an input and PORTB as an output, enabling precise interaction with a switch and LEDs. The **sbit sw at RA2\_bit** declaration facilitated real-time switch state detection, driving the conditional logic within an infinite **do-while(1)** loop. This resulted in PORTB transitioning to 0x00 when the switch was pressed **(sw == 1)** and reverting to **0xFF** when released **(sw==0)**, demonstrating dynamic input-output synchronization. Disabling the comparator **(CMCON = 0x07)** optimized performance by eliminating potential disruptions. Through this, I learned the practical significance of TRIS register manipulation, binary notation for pin control, and conditional programming, deepening my understanding of embedded systems design and microcontroller applications.