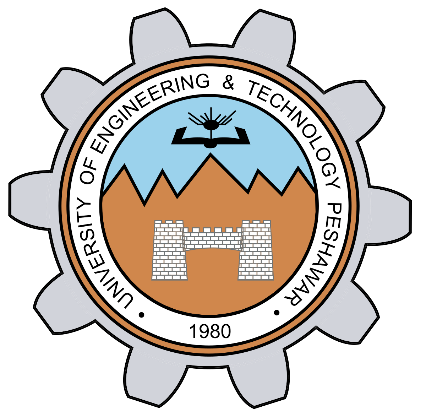
**Lab:3**

**Interfacing Push buttons to 8051 Development kit** **using Polling**



**MBSD Lab**

**Spring 2023**

**Submitted by:**

**Maaz Habib**

**Reg no:**

**20Pwcse1952**

“On my honour, as a student of University of Engineering and Technology Peshawar, I have neither nor received unauthorized assistance on this academic work”

**Submitted to:**

**Dr: Amaad Khalil**

**Task1: Interface Push buttons to 8051 microcontroller and blink even and odd LEDs using input buttons.**

**Source Code:**

ORG 0x0000 ; Start of program

MOV P1, #0x00 ; Initialize Port 1 as output

MOV P2, #0x00 ; Initialize Port 2 as output

MOV P3, #0xFF ; Initialize Port 3 as input with pull-up resistors

MAIN:

MOV A, P3 ; Read the state of Port 3

; Check if button 1 is pressed

ANL A, #0x01 ; Mask all bits except the first bit

CJNE A, #0x00, BLINK\_EVEN\_ODD ; Jump to blinking code if button 1 is pressed

; Check if button 2 is pressed

ANL A, #0x02 ; Mask all bits except the second bit

CJNE A, #0x00, ALL\_ON\_OFF ; Jump to all LEDs on/off code if button 2 is pressed

; Check if button 3 is pressed

ANL A, #0x04 ; Mask all bits except the third bit

CJNE A, #0x00, FIRST\_FOUR\_LAST\_FOUR ; Jump to first four on/last four off code if button 3 is pressed

JMP MAIN ; If no button is pressed, loop back to MAIN

BLINK\_EVEN\_ODD:

MOV P1, #0x55 ; Even LEDs (1, 3, 5, 7) are ON

MOV P2, #0xAA ; Odd LEDs (2, 4, 6, 8) are ON

ACALL DELAY ; Delay for some time

MOV P1, #0x00 ; Turn off all LEDs

MOV P2, #0x00

ACALL DELAY

JMP MAIN ; Loop back to MAIN

ALL\_ON\_OFF:

MOV P1, #0xFF ; All LEDs are ON

MOV P2, #0xFF

ACALL DELAY

MOV P1, #0x00 ; All LEDs are OFF

MOV P2, #0x00

ACALL DELAY

JMP MAIN ; Loop back to MAIN

FIRST\_FOUR\_LAST\_FOUR:

MOV P1, #0x0F ; First four LEDs (1, 2, 3, 4) are ON

MOV P2, #0xF0 ; Last four LEDs (5, 6, 7, 8) are OFF

ACALL DELAY

MOV P1, #0x00 ; Turn off all LEDs

MOV P2, #0x00

ACALL DELAY

JMP MAIN ; Loop back to MAIN

DELAY:

MOV R2, #0xFF ; Load initial value for delay counter

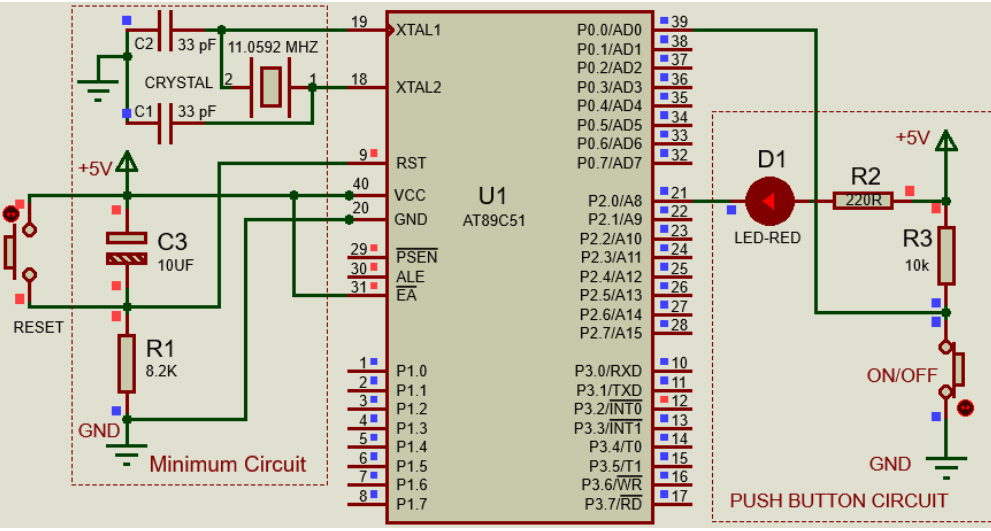
LOOP:

DJNZ R2, LOOP ; Decrement R2 and jump to LOOP if not zero

RET ; Return from subroutine

END ; End of program

**Schematic Diagram:**

****

**Task2: Interface Push buttons to 8051 microcontroller and display up counting and down counting using seven segment display using input buttons.**

#include <reg51.h>

// Define GPIO pins for push buttons

sbit buttonUp = P1^0; // Up counting button connected to P1.0

sbit buttonDown = P1^1; // Down counting button connected to P1.1

// Define GPIO pins for seven-segment display

sbit segmentA = P2^0; // Segment A connected to P2.0

sbit segmentB = P2^1; // Segment B connected to P2.1

sbit segmentC = P2^2; // Segment C connected to P2.2

sbit segmentD = P2^3; // Segment D connected to P2.3

sbit segmentE = P2^4; // Segment E connected to P2.4

sbit segmentF = P2^5; // Segment F connected to P2.5

sbit segmentG = P2^6; // Segment G connected to P2.6

sbit segmentCommon = P2^7; // Common pin connected to P2.7

// Delay function

void delay(unsigned int time) {

unsigned int i, j;

for (i = 0; i < time; i++)

for (j = 0; j < 1275; j++);

}

// Function to display a digit on the seven-segment display

void displayDigit(unsigned char digit) {

switch (digit) {

case 0:

segmentA = segmentB = segmentC = segmentD = segmentE = segmentF = 1;

segmentG = 0;

break;

case 1:

segmentB = segmentC = 1;

segmentA = segmentD = segmentE = segmentF = segmentG = 0;

break;

case 2:

segmentA = segmentB = segmentG = segmentE = segmentD = 1;

segmentC = segmentF = 0;

break;

case 3:

segmentA = segmentB = segmentC = segmentD = segmentG = 1;

segmentE = segmentF = 0;

break;

case 4:

segmentB = segmentC = segmentF = segmentG = 1;

segmentA = segmentD = segmentE = 0;

break;

case 5:

segmentA = segmentC = segmentD = segmentF = segmentG = 1;

segmentB = segmentE = 0;

break;

case 6:

segmentA = segmentC = segmentD = segmentE = segmentF = segmentG = 1;

segmentB = 0;

break;

case 7:

segmentA = segmentB = segmentC = 1;

segmentD = segmentE = segmentF = segmentG = 0;

break;

case 8:

segmentA = segmentB = segmentC = segmentD = segmentE = segmentF = segmentG = 1;

break;

case 9:

segmentA = segmentB = segmentC = segmentD = segmentF = segmentG = 1;

segmentE = segmentG = 0;

break;

}

}

void main() {

unsigned int counter = 0; // Counter variable for counting

// Set initial display to 0

displayDigit(counter % 10);

segmentCommon = 0; // Turn on

**Schematic Diagram:**

