Lab 12

Interfacing ADC with msp430

Spring 2025

Submitted by: Mohsin Sajjad

Registration No: 22pwsce2149

Class Section: A

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: _____

Submitted to:

Engr. Faheem Jan

Month Day, Year (01 06, 2025)

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

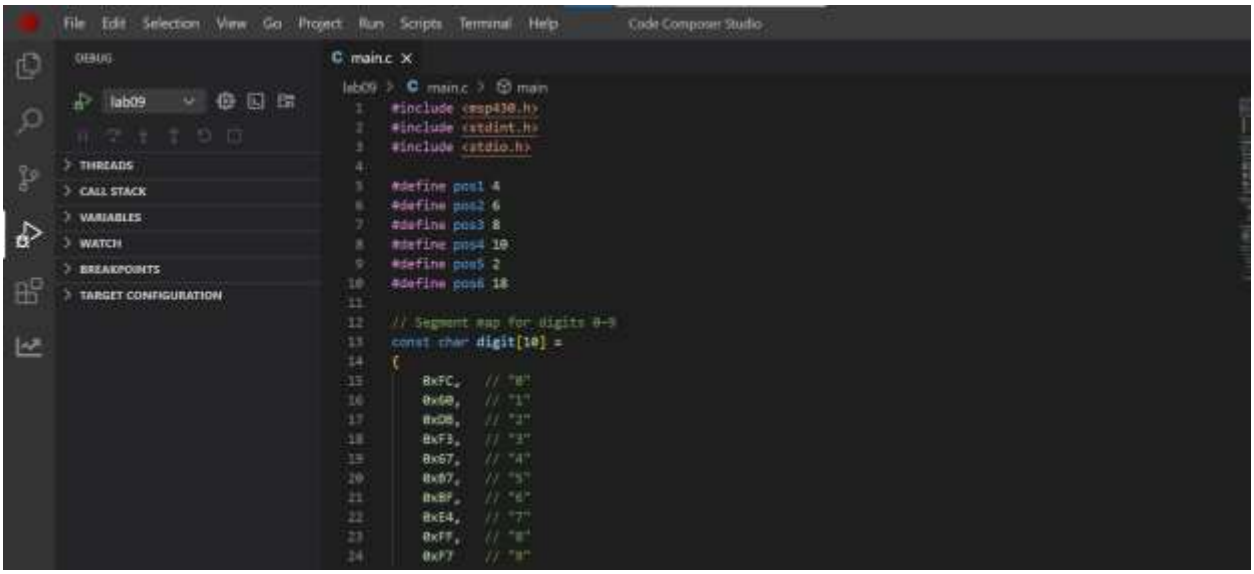
Interfacing ADC with msp430

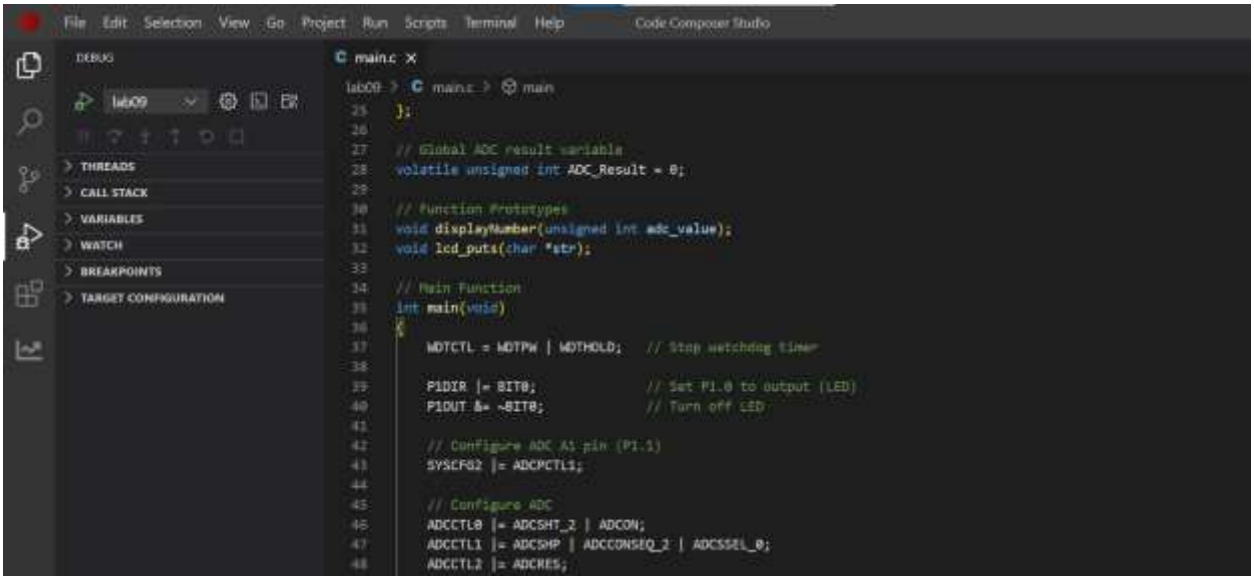
TASKS:

TASK 01:

Run the code given in the lecture

CODD:

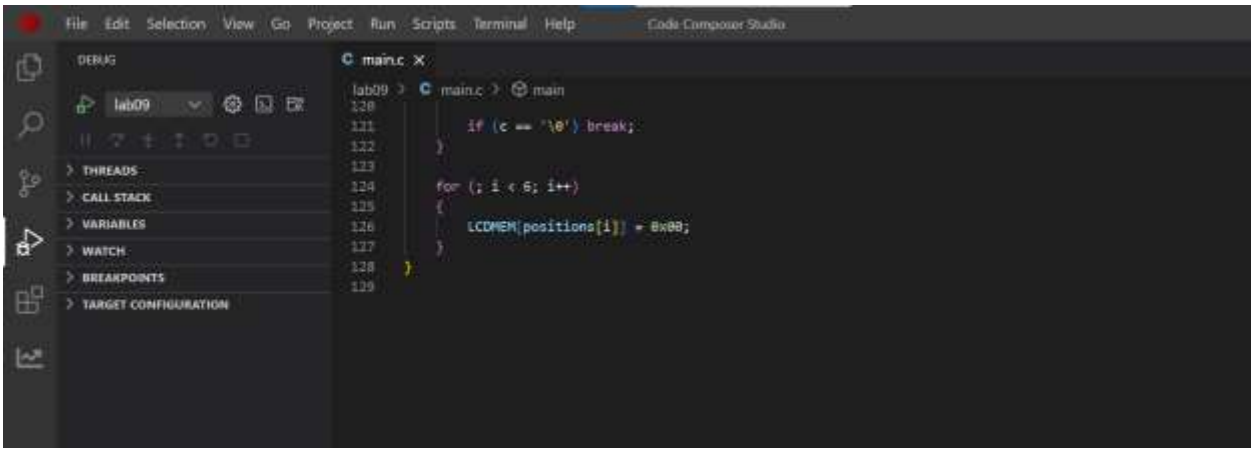




```c
    ADCMCTL0 |= ADCINCH_1 | ADCSREF_1;
    ADCIE |= ADCIE0;

    // Timer A1 for ADC trigger
    TA1CCR0 = 1024 - 1;
    TA1CCR1 = 512 - 1;
    TA1CCTL1 = OUTMOD_6;
    TA1CTL = TASSEL__ACLK | MC_1 | TACLR;

    // Internal reference voltage = 2.5V
    PMMCTL0_H = PMMPW_H;
    PMMCTL2 |= INTREFEN;
    __delay_cycles(400);

    ADC12CTL0 |= ADC12ENC;

    __bis_SR_register(LPM0_bits | GIE);
    __no_operation();
}

// ADC ISR
#pragma vector=ADC_VECTOR
__interrupt void ADC_ISR(void)
{
```

```c
    switch (__even_in_range(ADC12IV, ADC12IV_ADC12IFG0))
    {
        case ADC12IV_ADC12IFG0:
            ADC_Result = ADC12MEM0;
            displayNumber(ADC_Result);

            if (ADC12MEM0 < 0x155)
                P1OUT &= ~BIT0;
            else
                P1OUT |= BIT0;
            break;
        default:
            break;
    }
}

// Function to convert ADC result to string and show on LCD
void displayNumber(unsigned int value)
{
    char buffer[5];

    buffer[0] = (value / 1000) % 10 + '0';
    buffer[1] = (value / 100) % 10 + '0';
    buffer[2] = (value / 10) % 10 + '0';
```

```c
    buffer[3] = (value % 10) + '0';
    buffer[4] = '\0';

    lcd_puts(buffer);
}

// Show up to 6 characters on LCD
void lcd_puts(char *str)
{
    const uint8_t positions[6] = {pos1, pos2, pos3, pos4, pos5, pos6};
    int i;

    for (i = 0; i < 6; i++)
    {
        char c = str[i];
        if (c >= '0' && c <= '9')
        {
            LCDMEM[positions[i]] = digit[c - '0'];
        }
        else
        {
            LCDMEM[positions[i]] = 0x00; // Blank
        }
```



TASK02:

From temperature sensor (LM35) read temperature and convert it into Digital value by using ADC0804 and display the value on the LCD. In LCD at first line write your registration Number and on the second line display the value of the temperature sensor attached with ADC0804. (use proteus can also use MSP430FR4133)

CODE:

main.c

```c
#include <msp430.h>
#include <stdio.h>

#define ADC_DATA_IN   P1IN
#define ADC_WR_PIN    BIT5
#define ADC_RD_PIN    BIT4
#define ADC_INTR_PIN  BIT2

#define LCD_DATA_OUT  P2OUT
#define LCD_RS_PIN    BIT7
#define LCD_E_PIN     BIT6

void delay_us(unsigned int us) {
    while (us--) {
        __delay_cycles(3);
    }
}

void delay_ms(unsigned int ms) {
    while (ms--) {
        __delay_cycles(3000);
    }
}

void writecmd(unsigned char cmd) {
    P3OUT&= ~LCD_RS_PIN
    LCD_DATA_OUT= cmd;
    P3OUT|= LCD_E_PIN
    delay_us(50);
    P3OUT&= ~LCD_E_PIN
    delay_us(50);
}

void writedata(unsigned char data) {
    P3OUT|= LCD_RS_PIN
    LCD_DATA_OUT= data;
    P3OUT|= LCD_E_PIN
```

```
38      delay_us(50);
39      P3OUT&= ~LCD_E_PIN;
40      delay_us(50);
41  }
42
43 void lcdinit(void) {
44      delay_ms(20);
45      writecmd(0x38);
46      delay_ms(2);
47      writecmd(0x0C);
48      delay_ms(2);
49      writecmd(0x01);
50      delay_ms(2);
51      writecmd(0x06);
52      delay_ms(2);
53  }
54
55 unsigned int adc_read_raw(void) {
56      unsigned char raw;
57      P3OUT&= ~ADC_WR_PIN;
58      delay_us(2);
59      P3OUT|= ADC_WR_PIN;
60      while (P3IN & ADC_INTR_PIN);
61      }
62      P3OUT&= ~ADC_RD_PIN;
63      delay_us(2);
64      raw = ADC_DATA_IN;
65      P3OUT|= ADC_RD_PIN;
66      return (unsigned int)raw;
67  }
68
69 double adc_to_celsius(unsigned int raw) {
70      return ((double)raw * 1.953125);
71  }
```

```
74      int temp_times10 = (int)(temp * 10 + 0.5);
75      int integer_part = temp_times10 / 10;
76      int decimal_part = temp_times10 % 10;
77      char buf[4];
78      int idx = 0;
79      if (integer_part >= 100) {
80          buf[idx++] = '0' + (integer_part / 100);
81          buf[idx++] = '0' + ((integer_part / 10) % 10);
82          buf[idx++] = '0' + (integer_part % 10);
83      } else if (integer_part >= 10) {
84          buf[idx++] = '0' + (integer_part / 10);
85          buf[idx++] = '0' + (integer_part % 10);
86      } else {
87          buf[idx++] = '0' + integer_part;
88      }
89      buf[idx] = '\0';
90      {
91          int i;
92          for (i = 0; i < idx; i++) {
93              writedata(buf[i]);
94          }
95      }
96      writedata('.');
97      writedata('0' + decimal_part);
98      writedata(0xDF);
99      writedata('C');
100 }
101
102 int main(void) {
103     WDTCTL= WDTPW|WDTHOLD;
104     P1DIR= 0x00;
105     P1REN= 0x00;
106     P2DIR= 0xFF;
107     P2OUT= 0x00;
108     P3DIR|= (ADC_RD_PIN|ADC_WR_PIN|LCD_E_PIN|LCD_RS_PIN);
109     P3DIR&= ~ADC_INTR_PIN;
```

Output: