

Lab 10

Interfacing an external LCD to the MSP430 Microcontroller and Onboard LCD



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.”

A handwritten signature in black ink that reads "Mohsin Sajjad".

Student Signature: _____

Submitted to:

Engr. Faheem Jan

Month Day, Year (18 05, 2025)

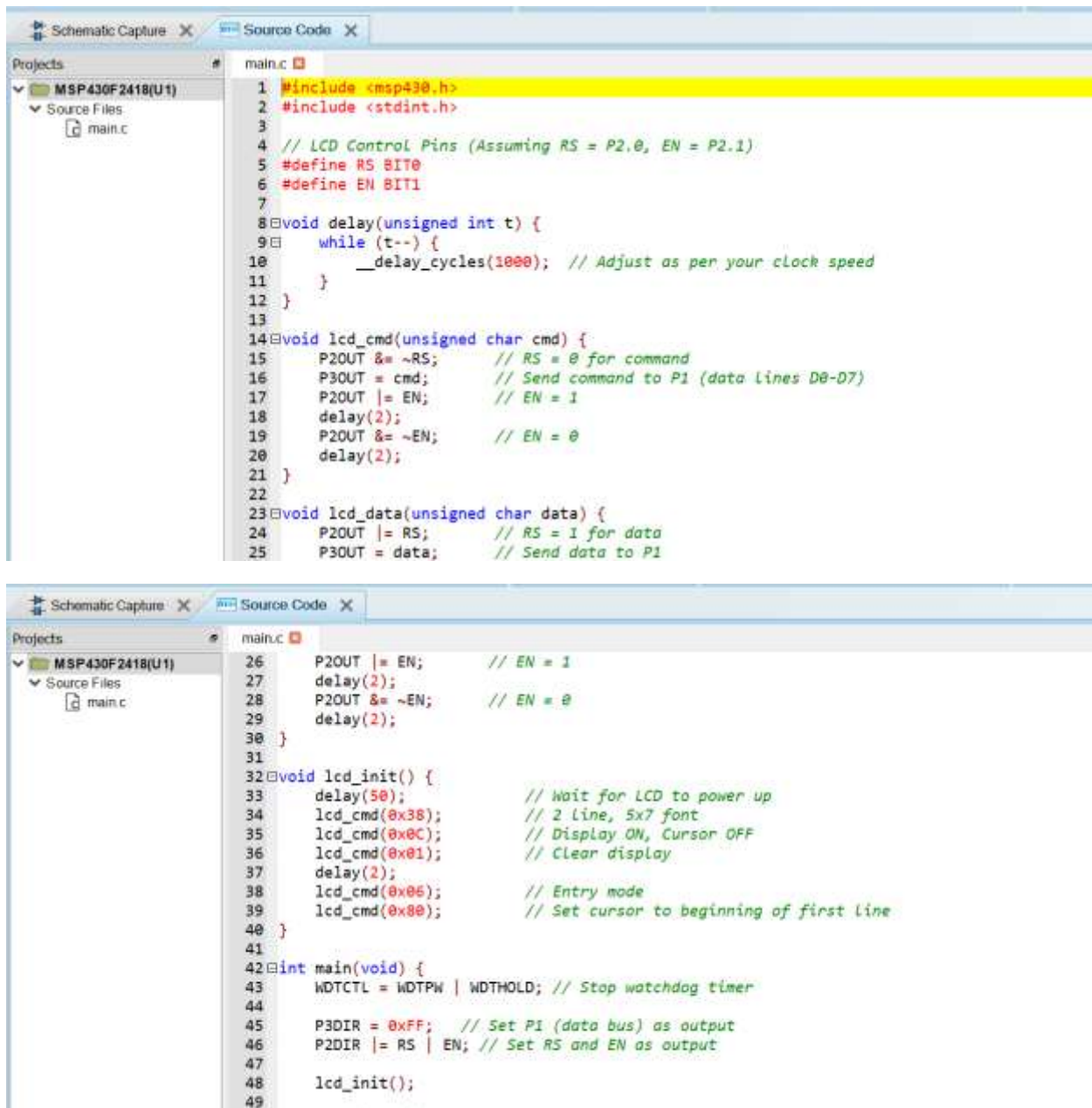
Department of Computer Systems Engineering
University of Engineering and Technology, Peshawar

Interfacing an external LCD to the MSP430 Microcontroller and Onboard LCD

TASKS:

1) Write Your Name on First Line and registration Number on second line of LCD (use proteus or attach external LCD with MSP).

CODE:



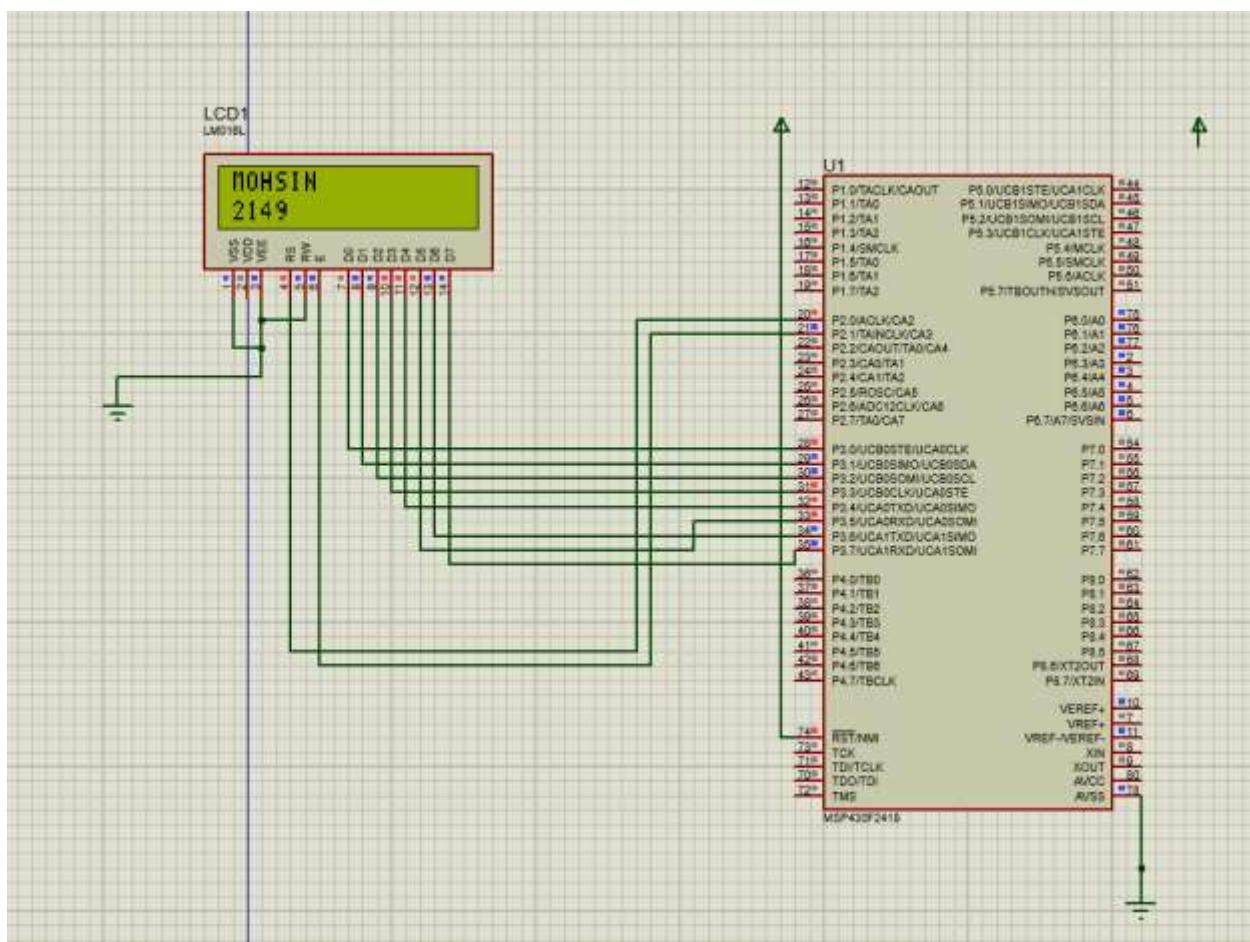
```
1 #include <msp430.h>
2 #include <stdint.h>
3
4 // LCD Control Pins (Assuming RS = P2.0, EN = P2.1)
5 #define RS BIT0
6 #define EN BIT1
7
8 void delay(unsigned int t) {
9     while (t--) {
10         __delay_cycles(1000); // Adjust as per your clock speed
11     }
12 }
13
14 void lcd_cmd(unsigned char cmd) {
15     P2OUT &= ~RS; // RS = 0 for command
16     P3OUT = cmd; // Send command to P1 (data lines D0-D7)
17     P2OUT |= EN; // EN = 1
18     delay(2);
19     P2OUT &= ~EN; // EN = 0
20     delay(2);
21 }
22
23 void lcd_data(unsigned char data) {
24     P2OUT |= RS; // RS = 1 for data
25     P3OUT = data; // Send data to P1
26
27     P2OUT |= EN; // EN = 1
28     delay(2);
29     P2OUT &= ~EN; // EN = 0
30     delay(2);
31 }
32 void lcd_init() {
33     delay(50); // Wait for LCD to power up
34     lcd_cmd(0x38); // 2 line, 5x7 font
35     lcd_cmd(0x0C); // Display ON, Cursor OFF
36     lcd_cmd(0x01); // Clear display
37     delay(2);
38     lcd_cmd(0x06); // Entry mode
39     lcd_cmd(0x80); // Set cursor to beginning of first line
40 }
41
42 int main(void) {
43     WDTCTL = WDTPW | WDTHOLD; // Stop watchdog timer
44
45     P3DIR = 0xFF; // Set P1 (data bus) as output
46     P2DIR |= RS | EN; // Set RS and EN as output
47
48     lcd_init();
49 }
```

```

Schematic Capture X Source Code X
Projects
  MSP430F2418(U1)
    Source Files
      main.c
45  P3DIR = 0xFF; // Set P1 (data bus) as output
46  P2DIR |= RS | EN; // Set RS and EN as output
47
48  lcd_init();
49
50  // First Line: "22PW"
51  lcd_data('M');
52  lcd_data('O');
53  lcd_data('H');
54  lcd_data('S');
55  lcd_data('I');
56  lcd_data('N');
57
58  // Move to second line
59  lcd_cmd(0xC0); // Address for Line 2
60
61  // Second Line: "2149"
62  lcd_data('2');
63  lcd_data('1');
64  lcd_data('4');
65  lcd_data('9');
66
67  while (1); // Infinite Loop
68 }

```

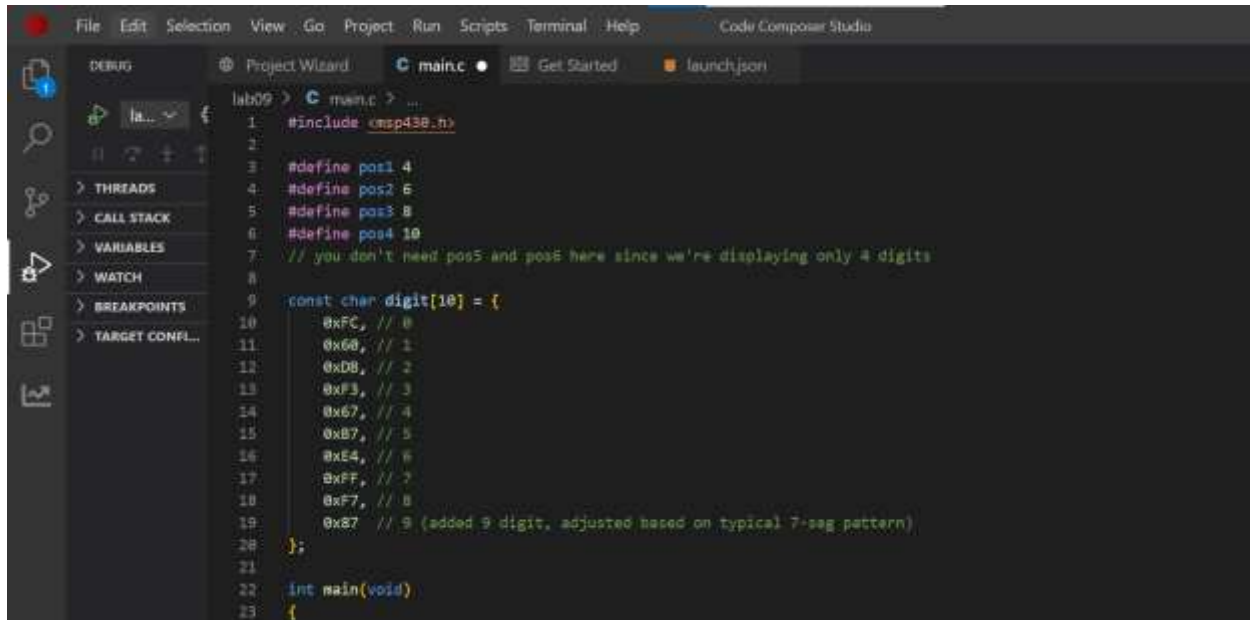
OUTPUT:



Task 2:

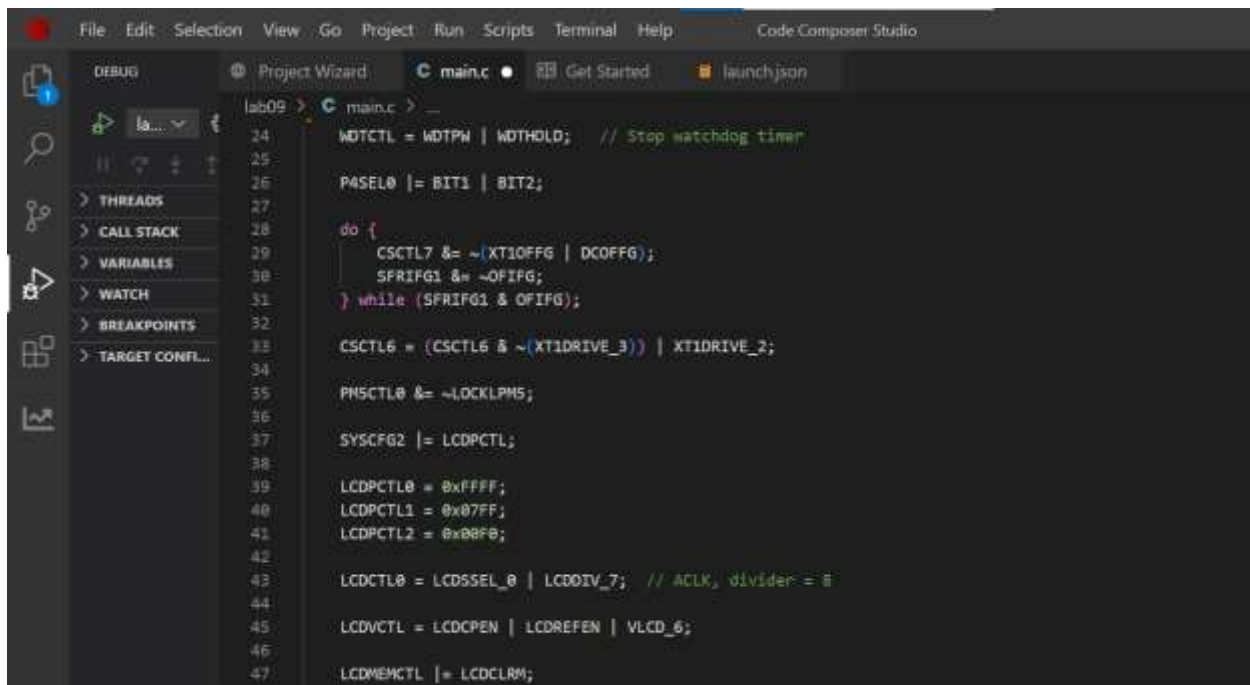
CODE:

Write last four digit of your registration Number On the Onboard LCD of msp430fr4133.



The screenshot shows the Code Composer Studio interface with the file `main.c` open. The code defines four positions for the LCD display and initializes a digit array.

```
1: #include <msp430.h>
2:
3: #define pos1 4
4: #define pos2 6
5: #define pos3 8
6: #define pos4 10
7: // you don't need pos5 and pos6 here since we're displaying only 4 digits
8:
9: const char digit[10] = {
10:     0xFC, // 0
11:     0x6B, // 1
12:     0xDB, // 2
13:     0xF3, // 3
14:     0x67, // 4
15:     0xB7, // 5
16:     0xE4, // 6
17:     0xFF, // 7
18:     0xF7, // 8
19:     0x87 // 9 (added 9 digit, adjusted based on typical 7-seg pattern)
20: };
21:
22: int main(void)
23: {
```



The screenshot shows the continuation of the C code in `main.c`, focusing on hardware configuration for the LCD and watchdog timer.

```
24: WDTCTL = WDTPW | WDTHOLD; // Stop watchdog timer
25:
26: P4SEL0 |= BIT1 | BIT2;
27:
28: do {
29:     CSCTL7 &= ~(XT1OFFG | DCOFFG);
30:     SFRIFG1 &= ~OFIFG;
31: } while (SFRIFG1 & OFIFG);
32:
33: CSCTL6 = (CSCTL6 & ~(XT1DRIVE_3)) | XT1DRIVE_2;
34:
35: PMSCTL0 &= ~LOCKLPM5;
36:
37: SYSCFG2 |= LCDPCTL;
38:
39: LCDPCTL0 = 0xFFFF;
40: LCDPCTL1 = 0x07FF;
41: LCDPCTL2 = 0x08F0;
42:
43: LCDCTL0 = LCDSSSEL_0 | LCDIV_7; // ACLK, divider = 8
44:
45: LCDVCTL = LCDCPEN | LCDREFEN | VLCD_6;
46:
47: LCDMEMCTL |= LCDCLRM;
```

