JITHENDHRA BU21EECE0100487

8051(MICRO-CONTROLLER) (Trainer: Dr. Jeevan K M)

Learning Objective:

- Understanding how to create patterns of LED blinking using the 8051 microcontroller.

Inputs and Outputs:

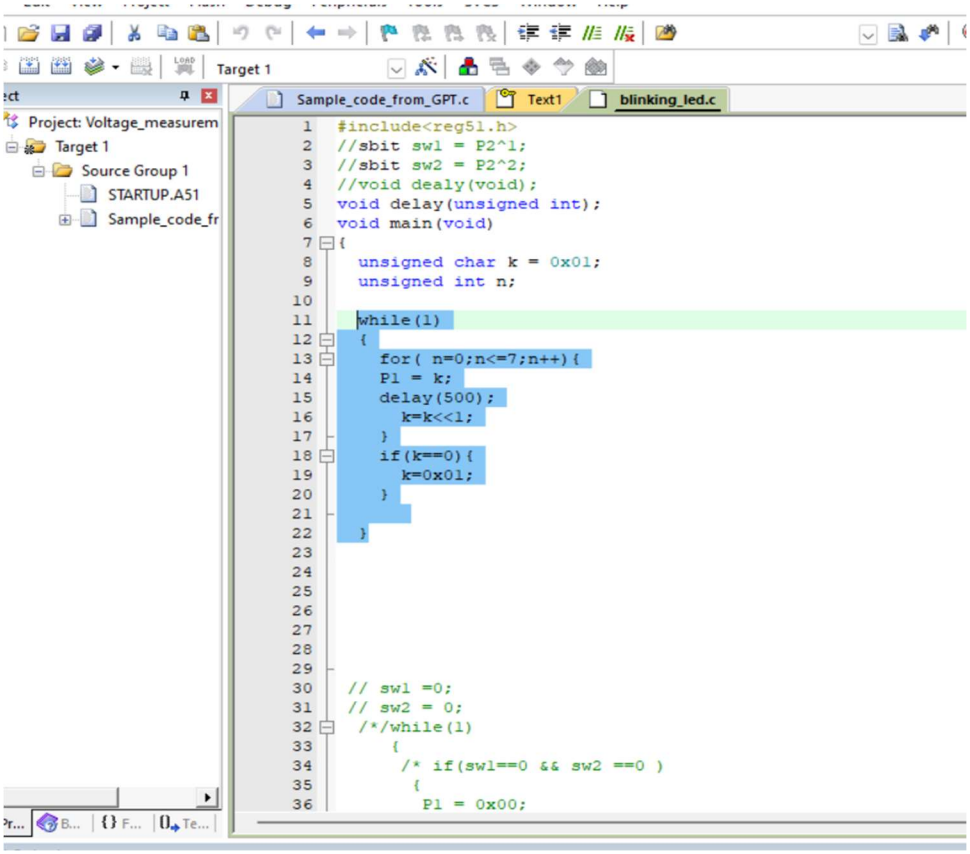
- **Inputs:** None
- **Outputs:** LED states (ON/OFF)

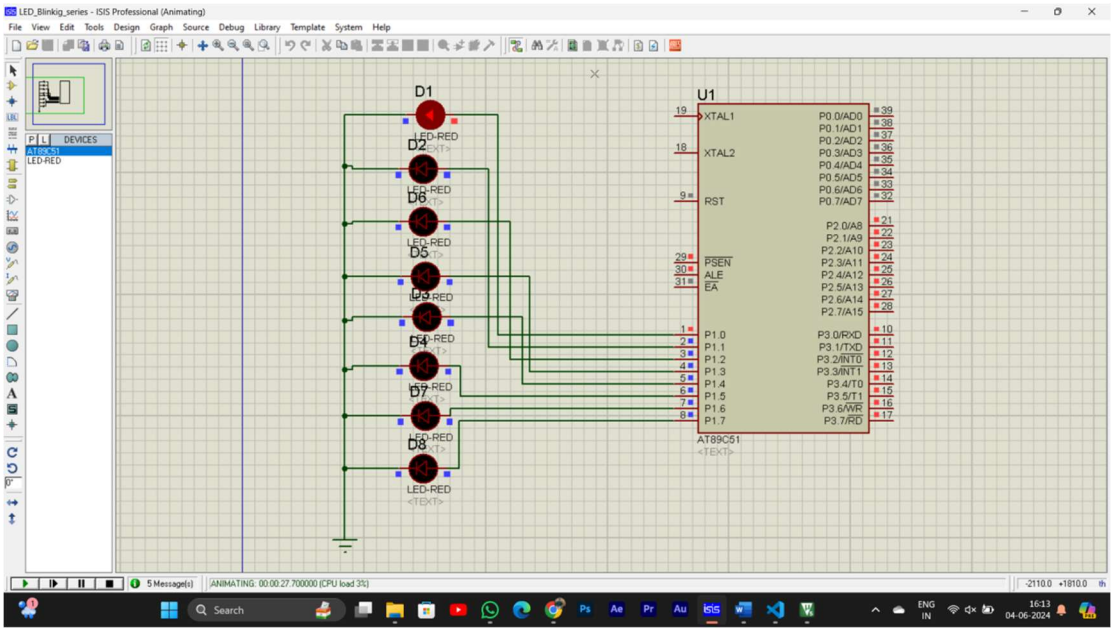
Logic:

- LEDs are connected to the microcontroller. By assigning a hexadecimal value to the port, the microcontroller converts this value to binary. Each bit in the binary representation controls an individual LED: a bit value of '1' turns the LED on, and '0' turns it off. For example, if `P1 = 0x01`, only the first LED (connected to the least significant bit) will be on, and the rest will be off.
- The other terminals of the LEDs are connected to the ground, creating a simple circuit for controlling the LEDs based on the port value.
- Various LED blinking patterns can be implemented using this basic principle. Two main approaches can be utilized:
    1. Direct assignment of hexadecimal values to the port.
    2. Using bitwise left shift operations within a loop to create sequential blinking patterns.

Common Mistakes:

- **Syntax Errors:** Common when writing embedded C code.
- **Indentation Errors:** Proper code formatting is crucial to avoid logical errors.
- **Port Mismatch:** Ensuring the correct port is being used for LED control is essential to achieve the desired output.


Target 1
Project: Voltage_measurem
Target 1
Source Group 1
STARTUP.A51
Sample_code_fr
Sample_code_from_GPT.c
Text1
blinking_led.c
#include<reg51.h>
//sbit sw1 = P2^1;
//sbit sw2 = P2^2;
//void dealy(void);
void delay(unsigned int);
void main(void)
{
unsigned char k = 0x01;
unsigned int n;
while(1)
{
for(n=0;n<=7;n++){
P1 = k;
delay(500);
k=k<<1;
}
if(k==0){
k=0x01;
}
}
// sw1 =0;
// sw2 = 0;
/*/while(1)
{
/* if(sw1==0 && sw2 ==0)
{
P1 = 0x00;


LED_Blinkig_series - ISIS Professional (Animating)
File View Edit Tools Design Graph Source Debug Library Template System Help
DEVICES
AT89C51
LED-RED
D1
D2
D6
D5
D3
D4
D7
D8
U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
P0.0/AD0
P0.1/AD1
P0.2/AD2
P0.3/AD3
P0.4/AD4
P0.5/AD5
P0.6/AD6
P0.7/AD7
P2.0/A8
P2.1/A9
P2.2/A10
P2.3/A11
P2.4/A12
P2.5/A13
P2.6/A14
P2.7/A15
P3.0/RXD
P3.1/TXD
P3.2/INT0
P3.3/INT1
P3.4/T0
P3.5/T1
P3.6/WR
P3.7/RD
AT89C51
5 Message(s)
ANIMATING: 00:00:27.700000 (CPU load 3%)
Search
ENG
IN
16:13
04-06-2024

ADC Using 8051

Learning Objective:

- Understanding the process of converting an analog signal to a digital signal using the ADC 0808.
- Learning how to interface the ADC 0808 with the 8051 microcontroller.

Inputs and Outputs:

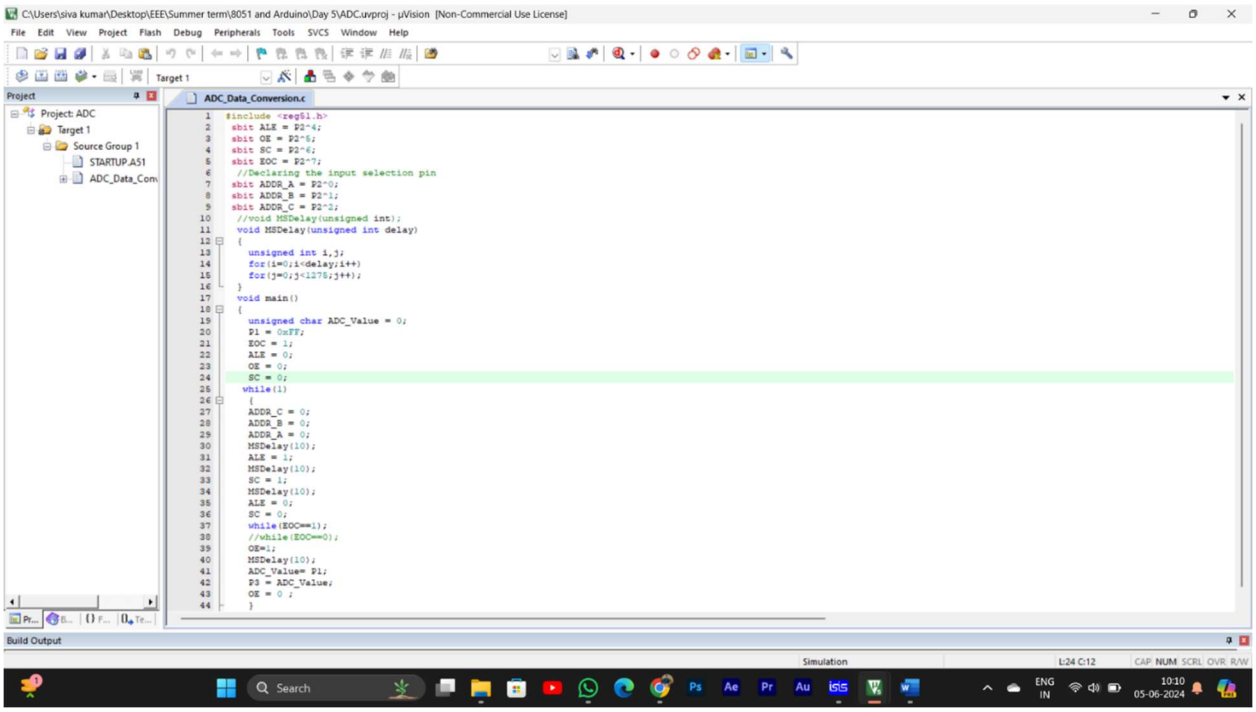
- **Input:** A potentiometer (1k or 10k ohms).
- **Outputs:** LEDs.

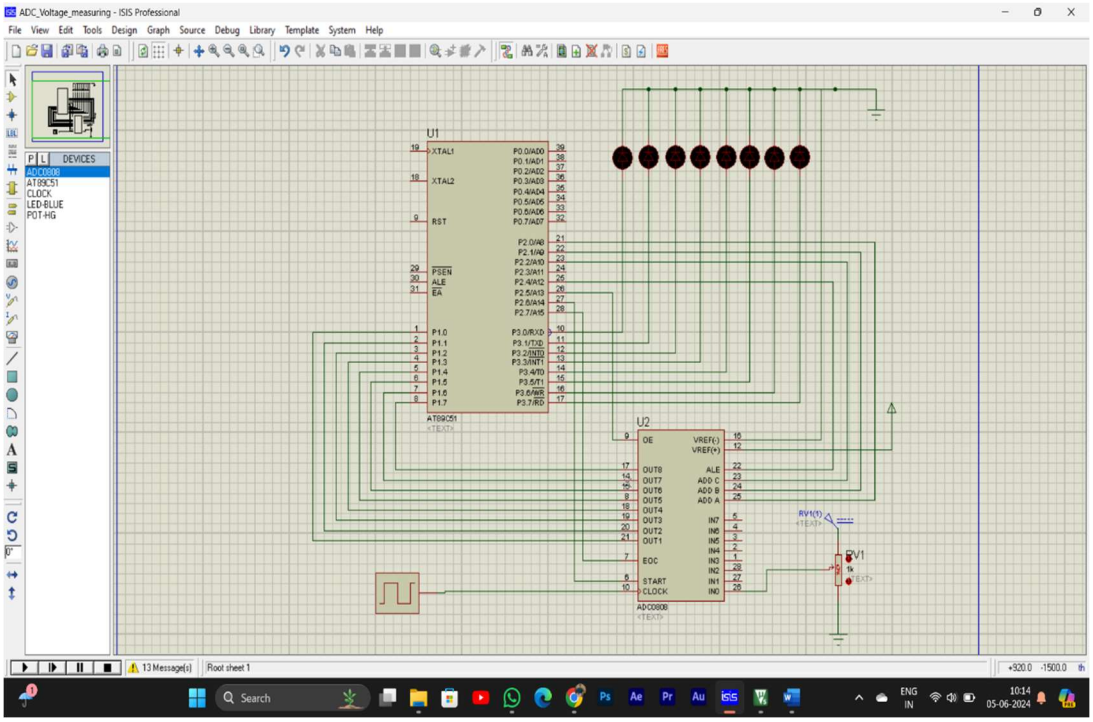
Logic:

- The analog output from the potentiometer is fed into one of the input channels of the ADC 0808.
- The digital output pins (D0-D7) of the ADC 0808 are connected to Port 1 of the 8051 microcontroller.
- The selection lines (A, B, C) of the ADC are connected to Port 2 of the 8051 to select the input channel.
- Additional control lines such as End of Conversion (EOC), Start, Address Latch Enable (ALE), and Clock Pulse are also connected to Port 2.
- The LEDs, which display the digital value of the potentiometer reading, are connected to Port 3 of the 8051 microcontroller.
- Based on the variation in resistance of the potentiometer, the digital value output from the ADC will change, which in turn will change the state of the LEDs connected to Port 3.

Common Mistakes:

- **Selection Line Configuration:** Double-check the configuration and connections of the selection lines (A, B, C) to ensure the correct input channel is selected on the ADC 0808.
- **EOC Pin Connection:** Ensure the End of Conversion (EOC) pin of the ADC 0808 is properly connected to the 8051 to correctly detect when a conversion is complete.


C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\Day 5\ADC.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: ADC
Target 1
Source Group 1
STARTUP.A51
ADC_Data_Com
ADC_Data_Conversion.c
#include <reg51.h>
sbit ALE = P2^4;
sbit OE = P2^5;
sbit SC = P2^6;
sbit EOC = P2^7;
//Declaring the input selection pin
sbit ADDR_A = P2^0;
sbit ADDR_B = P2^1;
sbit ADDR_C = P2^2;
//void MSDelay(unsigned int);
void MSDelay(unsigned int delay)
{
unsigned int i,j;
for(i=0;i<delay;i++)
for(j=0;j<1275;j++);
}
void main()
{
unsigned char ADC_Value = 0;
P1 = 0xFF;
EOC = 1;
ALE = 0;
OE = 0;
SC = 0;
while(1)
{
ADDR_C = 0;
ADDR_B = 0;
ADDR_A = 0;
MSDelay(10);
ALE = 1;
MSDelay(10);
SC = 1;
MSDelay(10);
ALE = 0;
SC = 0;
while(EOC==1);
//while(EOC==0);
OE=1;
MSDelay(10);
ADC_Value= P1;
P3 = ADC_Value;
OE = 0 ;
}
Build Output
Simulation
L:24 C:12
CAP NUM SCRL OVR R/W
Search
ENG
IN
10:10
05-06-2024


ADC_Voltage_measuring - ISIS Professional
File View Edit Tools Design Graph Source Debug Library Template System Help
DEVICES
ADC0808
AT89C51
CLOCK
LED-BLUE
POT-HG
U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
AT89C51
U2
OE
VREF(-)
VREF(+)
OUT0
OUT7
OUT6
OUT5
OUT4
OUT3
OUT2
OUT1
EOC
START
CLOCK
ALE
ADD C
ADD B
ADD A
IN7
IN6
IN5
IN4
IN3
IN2
IN1
IN0
ADC0808
RV1(1)
RV1
1k
13 Message(s)
Root sheet 1
Search
ENG
IN
10:14
05-06-2024

8051

Learning Objective:

- Learning how to control LED blinking using the 8051 microcontroller.

Inputs and Outputs:

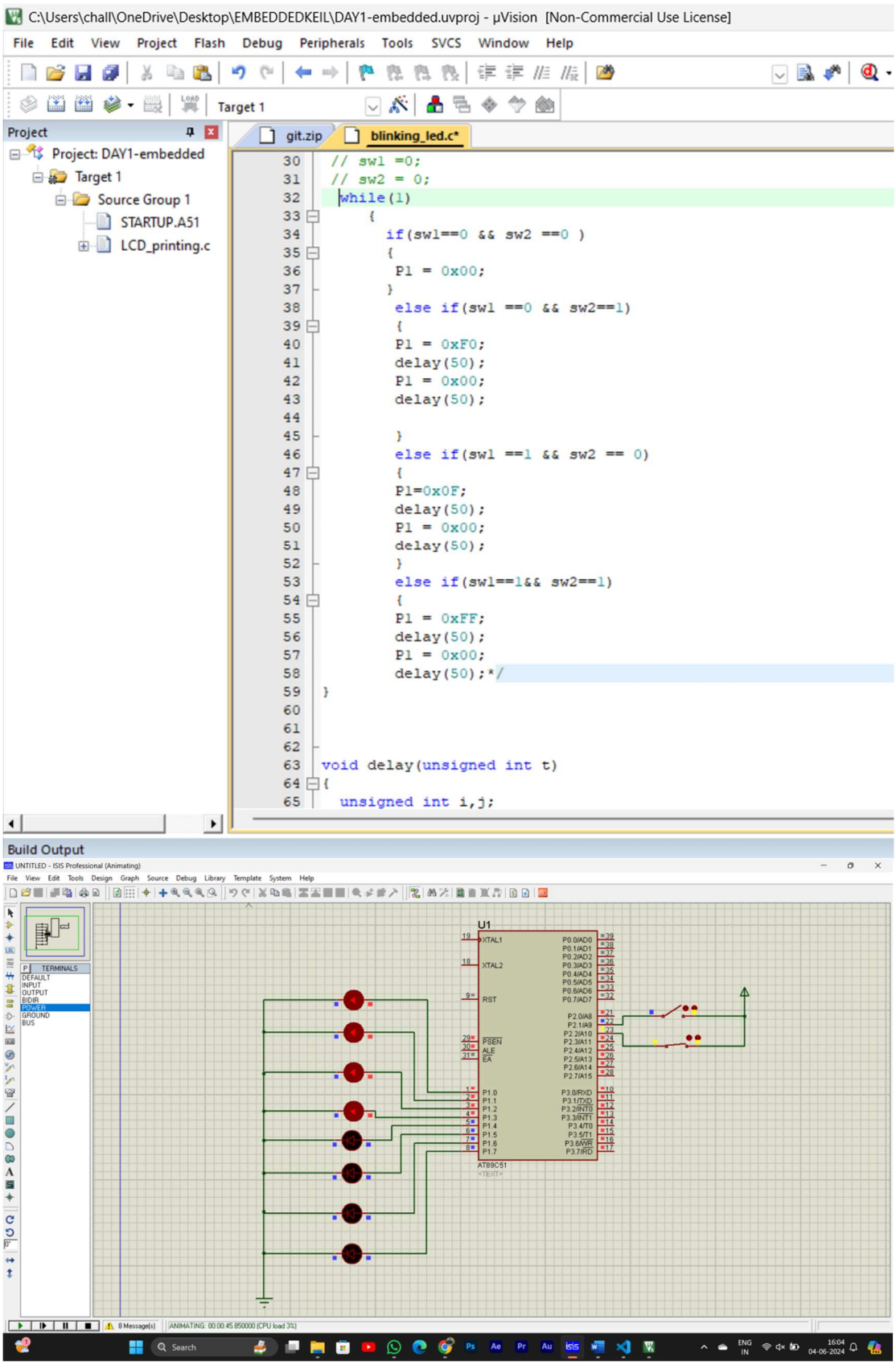
- **Inputs:** Switches
- **Outputs:** LEDs

Logic:

- LEDs are connected to Port 1 (P1) of the 8051 microcontroller.
- The code involves passing hexadecimal values to Port 1, which determines the state of each LED.
- A delay function is used to create the blinking effect by toggling the LEDs on and off.

Common Mistakes:

- **Project Creation in Keil Software:** Errors in creating a project or missing the step of creating a target before running the code.
- **Syntax and Indentation Errors:** Mistakes in code syntax and improper indentation can lead to compilation errors.
- **Port Confusion:** Errors in connecting the LEDs to the correct ports on the microcontroller.


C:\Users\chall\OneDrive\Desktop\EMBEDDEDKEIL\DAY1-embedded.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: DAY1-embedded
Target 1
Source Group 1
STARTUP.A51
LCD_printing.c
git.zip
blinking_led.c*
30 // sw1 =0;
31 // sw2 = 0;
32 while(1)
33 {
34 if(sw1==0 && sw2 ==0)
35 {
36 P1 = 0x00;
37 }
38 else if(sw1 ==0 && sw2==1)
39 {
40 P1 = 0xF0;
41 delay(50);
42 P1 = 0x00;
43 delay(50);
44
45 }
46 else if(sw1 ==1 && sw2 == 0)
47 {
48 P1=0x0F;
49 delay(50);
50 P1 = 0x00;
51 delay(50);
52 }
53 else if(sw1==1&& sw2==1)
54 {
55 P1 = 0xFF;
56 delay(50);
57 P1 = 0x00;
58 delay(50);*/
59 }
60
61
62
63 void delay(unsigned int t)
64 {
65 unsigned int i,j;
Build Output
UNTITLED - ISIS Professional (Animating)
File View Edit Tools Design Graph Source Debug Library Template System Help
TERMINALS
DEFAULT
INPUT
OUTPUT
BIDIR
POWER
GROUND
BUS
U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
AT89C51
8 Message(s)
ANIMATING: 00:00:45.850000 (CPU load 3%)
Search
ENG
IN
16:04
04-06-2024

Voltage Measurement and LCD displaying(8051)

Learning Objective:

- Understanding how to measure voltage variations using an ADC.
- Displaying the voltage variations on LEDs.
- Displaying the voltage measurement status on an LCD display.

Inputs and Outputs:

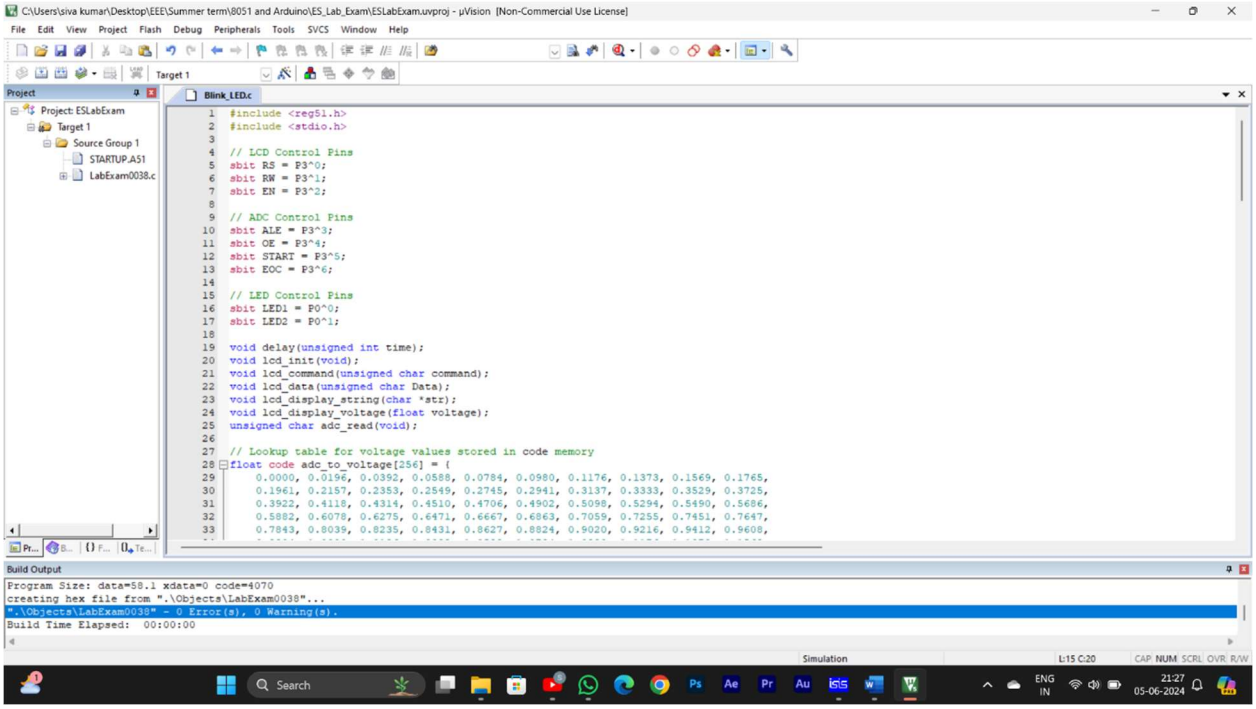
- **Input:** Potentiometer.
- **Outputs:** LEDs and LCD display.

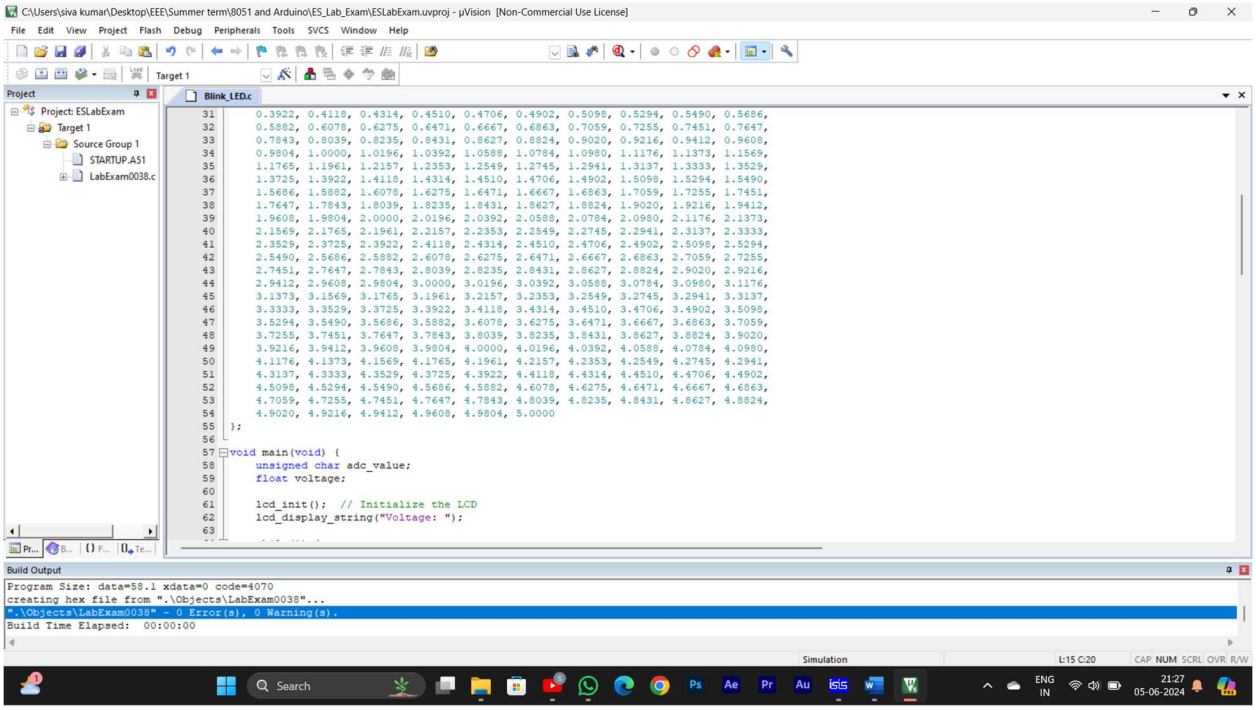
Logic:

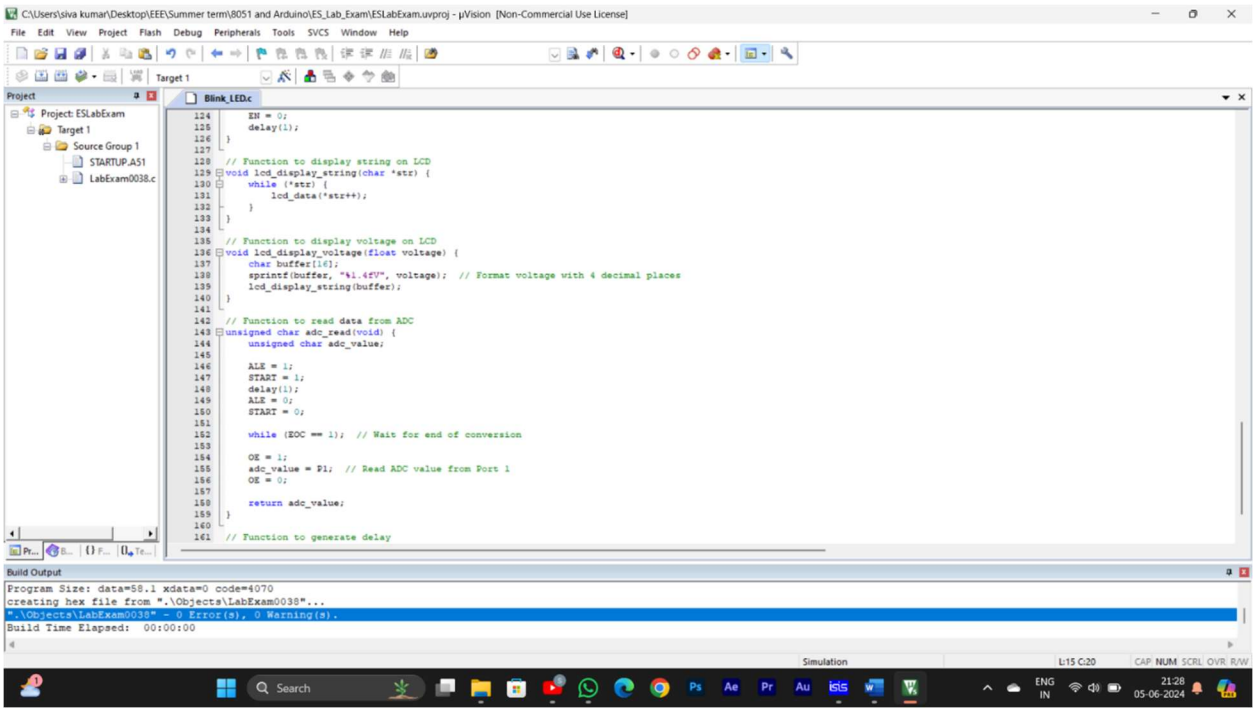
1. **Connections:**
    - Connect the potentiometer output to an input channel of the ADC 0808.
    - Provide a 5V power supply to the ADC 0808.
    - Connect the digital output pins (D0-D7) of the ADC 0808 to a port on the 8051 microcontroller.
    - Connect the LCD display and LEDs to the 8051 microcontroller.
2. **Operation:**
    - The ADC converts the analog signal from the potentiometer to a digital signal.
    - The 8051 microcontroller reads the digital signal from the ADC.
    - Based on the digital value, control the LEDs to indicate voltage levels.
    - Convert the digital value to a corresponding voltage value and display it on the LCD.

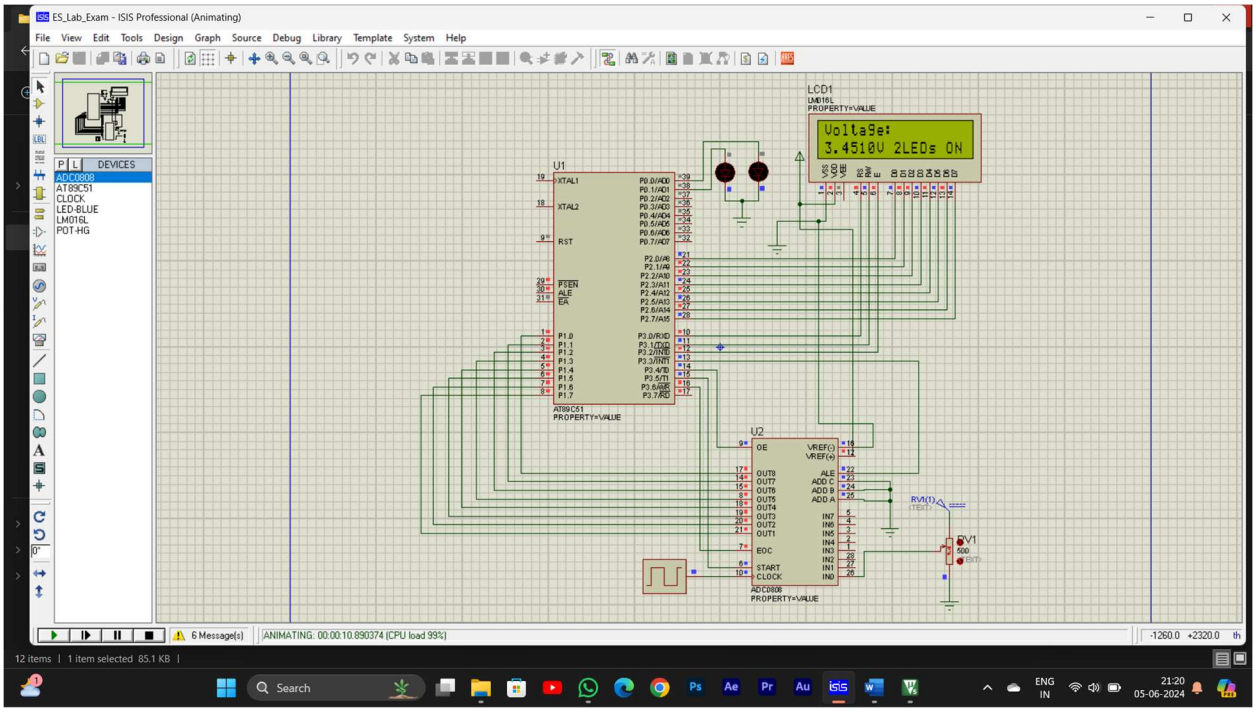
Common Mistakes:

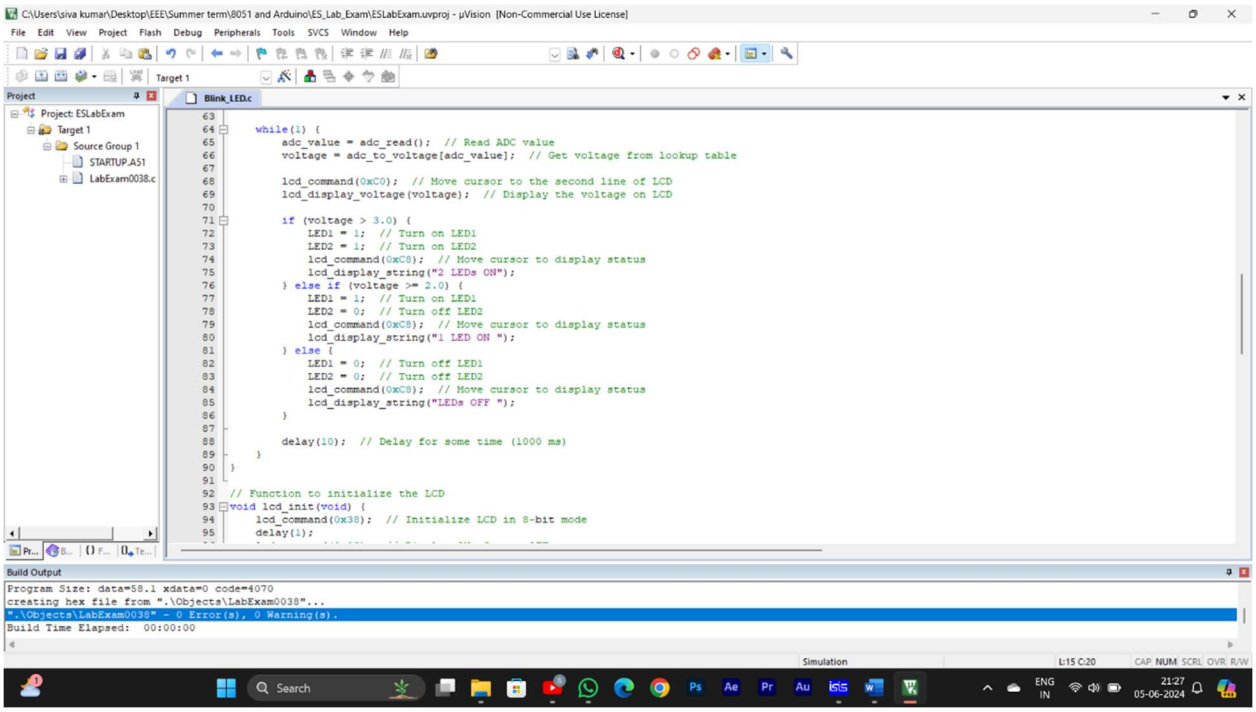
- **Port Mismatch:** Ensure that the port names used in the code match the actual circuit connections.
- **Proper Connections:** Verify that all connections are secure. A single loose connection can lead to data loss and no output.
- **Input Levels:** Ensure that input levels to the microcontroller and ADC are correct (either high or low as required).

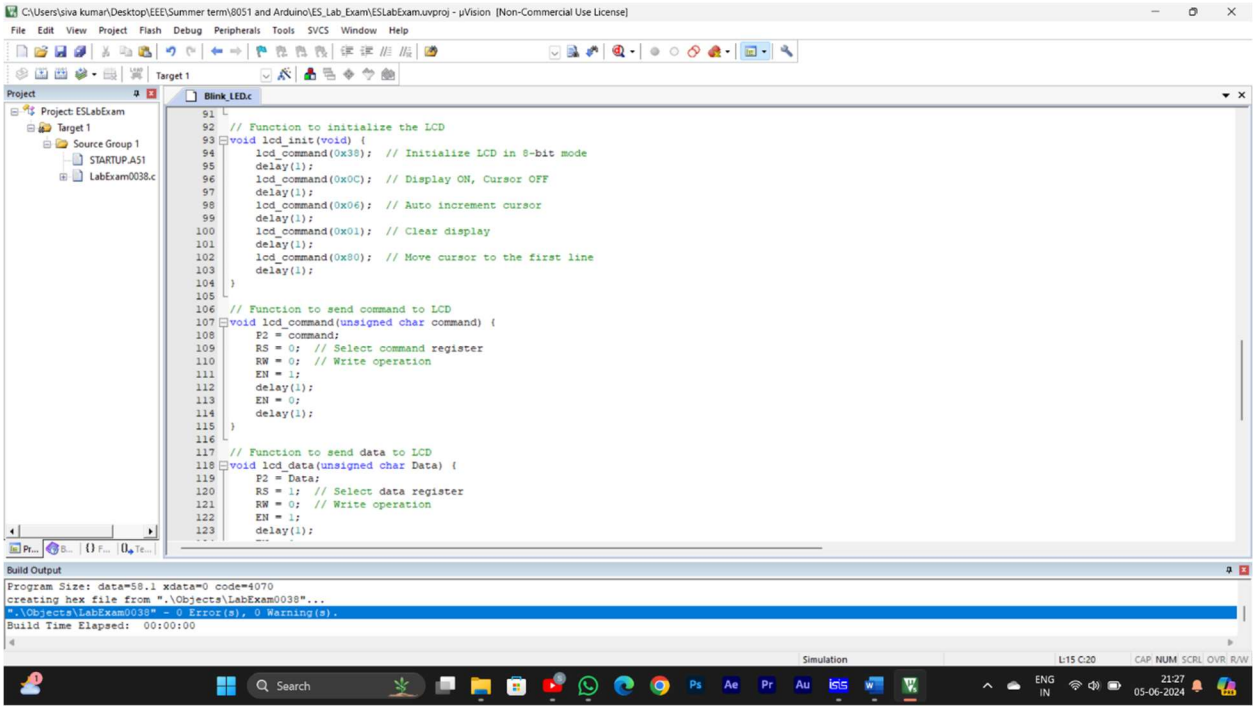

C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\ES_Lab_Exam\ESLabExam.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: ESLabExam
Target 1
Source Group 1
STARTUP.A51
LabExam0038.c
Blink_LED.c
#include <reg51.h>
#include <stdio.h>
// LCD Control Pins
sbit RS = P3^0;
sbit RW = P3^1;
sbit EN = P3^2;
// ADC Control Pins
sbit ALE = P3^3;
sbit OE = P3^4;
sbit START = P3^5;
sbit EOC = P3^6;
// LED Control Pins
sbit LED1 = P0^0;
sbit LED2 = P0^1;
void delay(unsigned int time);
void lcd_init(void);
void lcd_command(unsigned char command);
void lcd_data(unsigned char Data);
void lcd_display_string(char *str);
void lcd_display_voltage(float voltage);
unsigned char adc_read(void);
// Lookup table for voltage values stored in code memory
float code adc_to_voltage[256] = {
0.0000, 0.0196, 0.0392, 0.0588, 0.0784, 0.0980, 0.1176, 0.1373, 0.1569, 0.1765,
0.1961, 0.2157, 0.2353, 0.2549, 0.2745, 0.2941, 0.3137, 0.3333, 0.3529, 0.3725,
0.3922, 0.4118, 0.4314, 0.4510, 0.4706, 0.4902, 0.5098, 0.5294, 0.5490, 0.5686,
0.5882, 0.6078, 0.6275, 0.6471, 0.6667, 0.6863, 0.7059, 0.7255, 0.7451, 0.7647,
0.7843, 0.8039, 0.8235, 0.8431, 0.8627, 0.8824, 0.9020, 0.9216, 0.9412, 0.9608,
Build Output
Program Size: data=58.1 xdata=0 code=4070
creating hex file from ".\Objects\LabExam0038"...
".\Objects\LabExam0038" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:00
Simulation
L:15 C:20
CAP NUM SCRL OVR R/W
Search
ENG
IN
21:27
05-06-2024


C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\ES_Lab_Exam\ESLabExam.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: ESLabExam
Target 1
Source Group 1
STARTUP.A51
LabExam0038.c
Blink_LED.c
0.3922, 0.4118, 0.4314, 0.4510, 0.4706, 0.4902, 0.5098, 0.5294, 0.5490, 0.5686,
0.5882, 0.6078, 0.6275, 0.6471, 0.6667, 0.6863, 0.7059, 0.7255, 0.7451, 0.7647,
0.7843, 0.8039, 0.8235, 0.8431, 0.8627, 0.8824, 0.9020, 0.9216, 0.9412, 0.9608,
0.9804, 1.0000, 1.0196, 1.0392, 1.0588, 1.0784, 1.0980, 1.1176, 1.1373, 1.1569,
1.1765, 1.1961, 1.2157, 1.2353, 1.2549, 1.2745, 1.2941, 1.3137, 1.3333, 1.3529,
1.3725, 1.3922, 1.4118, 1.4314, 1.4510, 1.4706, 1.4902, 1.5098, 1.5294, 1.5490,
1.5686, 1.5882, 1.6078, 1.6275, 1.6471, 1.6667, 1.6863, 1.7059, 1.7255, 1.7451,
1.7647, 1.7843, 1.8039, 1.8235, 1.8431, 1.8627, 1.8824, 1.9020, 1.9216, 1.9412,
1.9608, 1.9804, 2.0000, 2.0196, 2.0392, 2.0588, 2.0784, 2.0980, 2.1176, 2.1373,
2.1569, 2.1765, 2.1961, 2.2157, 2.2353, 2.2549, 2.2745, 2.2941, 2.3137, 2.3333,
2.3529, 2.3725, 2.3922, 2.4118, 2.4314, 2.4510, 2.4706, 2.4902, 2.5098, 2.5294,
2.5490, 2.5686, 2.5882, 2.6078, 2.6275, 2.6471, 2.6667, 2.6863, 2.7059, 2.7255,
2.7451, 2.7647, 2.7843, 2.8039, 2.8235, 2.8431, 2.8627, 2.8824, 2.9020, 2.9216,
2.9412, 2.9608, 2.9804, 3.0000, 3.0196, 3.0392, 3.0588, 3.0784, 3.0980, 3.1176,
3.1373, 3.1569, 3.1765, 3.1961, 3.2157, 3.2353, 3.2549, 3.2745, 3.2941, 3.3137,
3.3333, 3.3529, 3.3725, 3.3922, 3.4118, 3.4314, 3.4510, 3.4706, 3.4902, 3.5098,
3.5294, 3.5490, 3.5686, 3.5882, 3.6078, 3.6275, 3.6471, 3.6667, 3.6863, 3.7059,
3.7255, 3.7451, 3.7647, 3.7843, 3.8039, 3.8235, 3.8431, 3.8627, 3.8824, 3.9020,
3.9216, 3.9412, 3.9608, 3.9804, 4.0000, 4.0196, 4.0392, 4.0588, 4.0784, 4.0980,
4.1176, 4.1373, 4.1569, 4.1765, 4.1961, 4.2157, 4.2353, 4.2549, 4.2745, 4.2941,
4.3137, 4.3333, 4.3529, 4.3725, 4.3922, 4.4118, 4.4314, 4.4510, 4.4706, 4.4902,
4.5098, 4.5294, 4.5490, 4.5686, 4.5882, 4.6078, 4.6275, 4.6471, 4.6667, 4.6863,
4.7059, 4.7255, 4.7451, 4.7647, 4.7843, 4.8039, 4.8235, 4.8431, 4.8627, 4.8824,
4.9020, 4.9216, 4.9412, 4.9608, 4.9804, 5.0000
};
void main(void) {
unsigned char adc_value;
float voltage;
lcd_init(); // Initialize the LCD
lcd_display_string("Voltage: ");
Build Output
Program Size: data=58.1 xdata=0 code=4070
creating hex file from ".\Objects\LabExam0038"...
".\Objects\LabExam0038" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:00
Simulation
L:15 C:20
CAP NUM SCRL OVR R/W
Search
ENG
IN
21:27
05-06-2024


C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\ES_Lab_Exam\ESLabExam.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: ESLabExam
Target 1
Source Group 1
STARTUP.A51
LabExam0038.c
Blink_LED.c
124 EN = 0;
125 delay(1);
126 }
127
128 // Function to display string on LCD
129 void lcd_display_string(char *str) {
130 while (*str) {
131 lcd_data(*str++);
132 }
133 }
134
135 // Function to display voltage on LCD
136 void lcd_display_voltage(float voltage) {
137 char buffer[16];
138 sprintf(buffer, "%1.4fV", voltage); // Format voltage with 4 decimal places
139 lcd_display_string(buffer);
140 }
141
142 // Function to read data from ADC
143 unsigned char adc_read(void) {
144 unsigned char adc_value;
145
146 ALE = 1;
147 START = 1;
148 delay(1);
149 ALE = 0;
150 START = 0;
151
152 while (EOC == 1); // Wait for end of conversion
153
154 OE = 1;
155 adc_value = P1; // Read ADC value from Port 1
156 OE = 0;
157
158 return adc_value;
159 }
160
161 // Function to generate delay
Build Output
Program Size: data=58.1 xdata=0 code=4070
creating hex file from ".\Objects\LabExam0038"...
".\Objects\LabExam0038" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:00
Simulation
L:15 C:20
CAP NUM SCRL OVR R/W
Search
ENG IN
21:28
05-06-2024


ES_Lab_Exam - ISIS Professional (Animating)
File View Edit Tools Design Graph Source Debug Library Template System Help
DEVICES
ADC0808
AT89C51
CLOCK
LED-BLUE
LM016L
POT-HG
LCD1
LM016L
PROPERTY=VALUE
Voltage:
3.4510V 2LEDs ON
U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
AT89C51
PROPERTY=VALUE
U2
OE
VREF(-)
VREF(+)
ALE
ADD C
ADD B
ADD A
OUT8
OUT7
OUT6
OUT5
OUT4
OUT3
OUT2
OUT1
EOC
START
CLOCK
IN7
IN6
IN5
IN4
IN3
IN2
IN1
IN0
ADC0808
PROPERTY=VALUE
RV1
500
6 Message(s)
ANIMATING: 00:00:10.890374 (CPU load 99%)
-1260.0 +2320.0
12 items | 1 item selected 85.1 KB
Search
ENG IN
21:20
05-06-2024


C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\ES_Lab_Exam\ESLabExam.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: ESLabExam
Target 1
Source Group 1
STARTUP.A51
LabExam0038.c
Blink_LED.c
while(1) {
adc_value = adc_read(); // Read ADC value
voltage = adc_to_voltage[adc_value]; // Get voltage from lookup table
lcd_command(0xC0); // Move cursor to the second line of LCD
lcd_display_voltage(voltage); // Display the voltage on LCD
if (voltage > 3.0) {
LED1 = 1; // Turn on LED1
LED2 = 1; // Turn on LED2
lcd_command(0xC8); // Move cursor to display status
lcd_display_string("2 LEDs ON");
} else if (voltage >= 2.0) {
LED1 = 1; // Turn on LED1
LED2 = 0; // Turn off LED2
lcd_command(0xC8); // Move cursor to display status
lcd_display_string("1 LED ON ");
} else {
LED1 = 0; // Turn off LED1
LED2 = 0; // Turn off LED2
lcd_command(0xC8); // Move cursor to display status
lcd_display_string("LEDs OFF ");
}
delay(10); // Delay for some time (1000 ms)
}
// Function to initialize the LCD
void lcd_init(void) {
lcd_command(0x38); // Initialize LCD in 8-bit mode
delay(1);
Build Output
Program Size: data=58.1 xdata=0 code=4070
creating hex file from ".\Objects\LabExam0038"...
".\Objects\LabExam0038" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:00
Simulation
L:15 C:20
CAP NUM SCRL OVR R/W
Search
ENG
IN
21:27
05-06-2024


C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\ES_Lab_Exam\ESLabExam.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: ESLabExam
Target 1
Source Group 1
STARTUP.A51
LabExam0038.c
Blink_LED.c
// Function to initialize the LCD
void lcd_init(void) {
lcd_command(0x38); // Initialize LCD in 8-bit mode
delay(1);
lcd_command(0x0C); // Display ON, Cursor OFF
delay(1);
lcd_command(0x06); // Auto increment cursor
delay(1);
lcd_command(0x01); // Clear display
delay(1);
lcd_command(0x80); // Move cursor to the first line
delay(1);
}
// Function to send command to LCD
void lcd_command(unsigned char command) {
P2 = command;
RS = 0; // Select command register
RW = 0; // Write operation
EN = 1;
delay(1);
EN = 0;
delay(1);
}
// Function to send data to LCD
void lcd_data(unsigned char Data) {
P2 = Data;
RS = 1; // Select data register
RW = 0; // Write operation
EN = 1;
delay(1);
Build Output
Program Size: data=58.1 xdata=0 code=4070
creating hex file from ".\Objects\LabExam0038"...
".\Objects\LabExam0038" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:00
Simulation
L:15 C:20
CAP NUM SCRL OVR R/W
Search
ENG
IN
21:27
05-06-2024

8051(micro-controller) using LCD Display

Learning Objective:

- Learning how to use the 8051 microcontroller to display text on an LCD.

Inputs and Outputs:

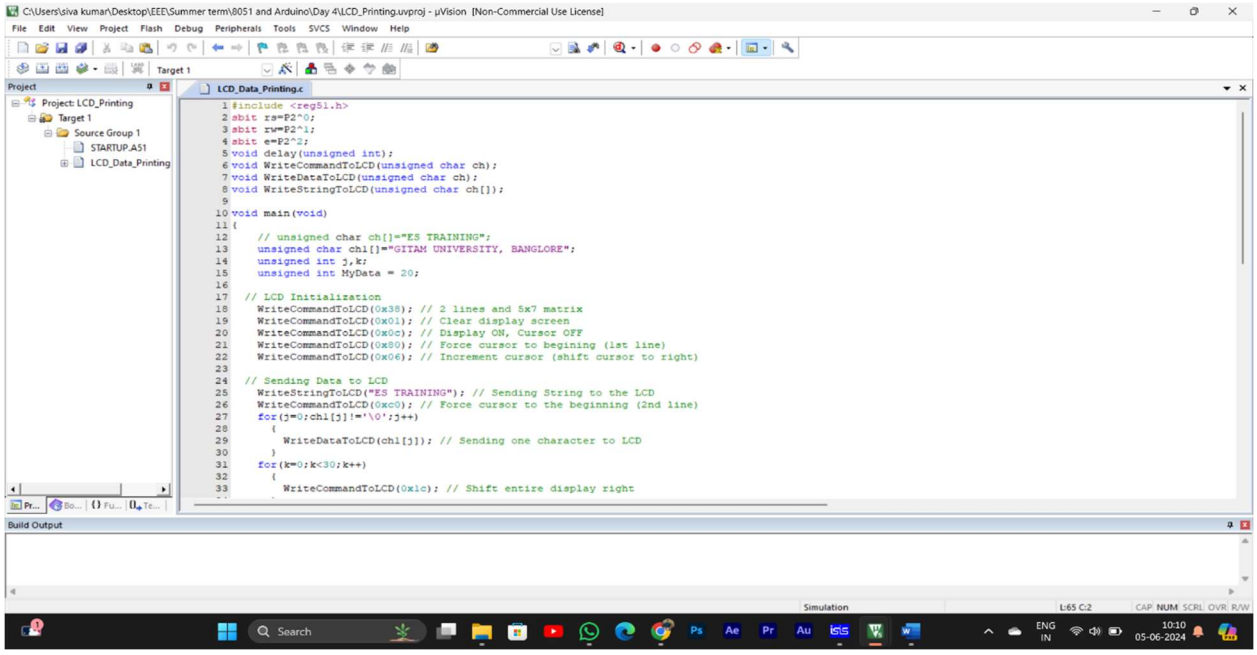
- **Inputs:** Commands for the LCD display.
- **Output:** Displayed text on the LCD.

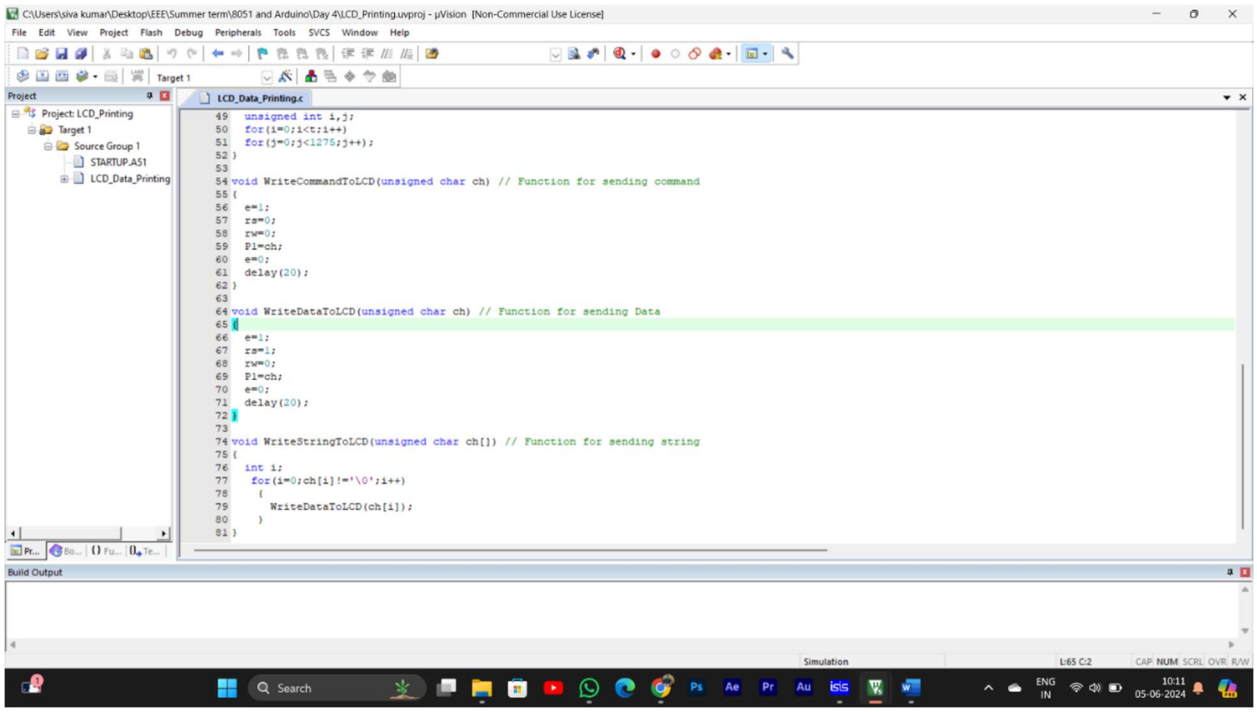
Logic:

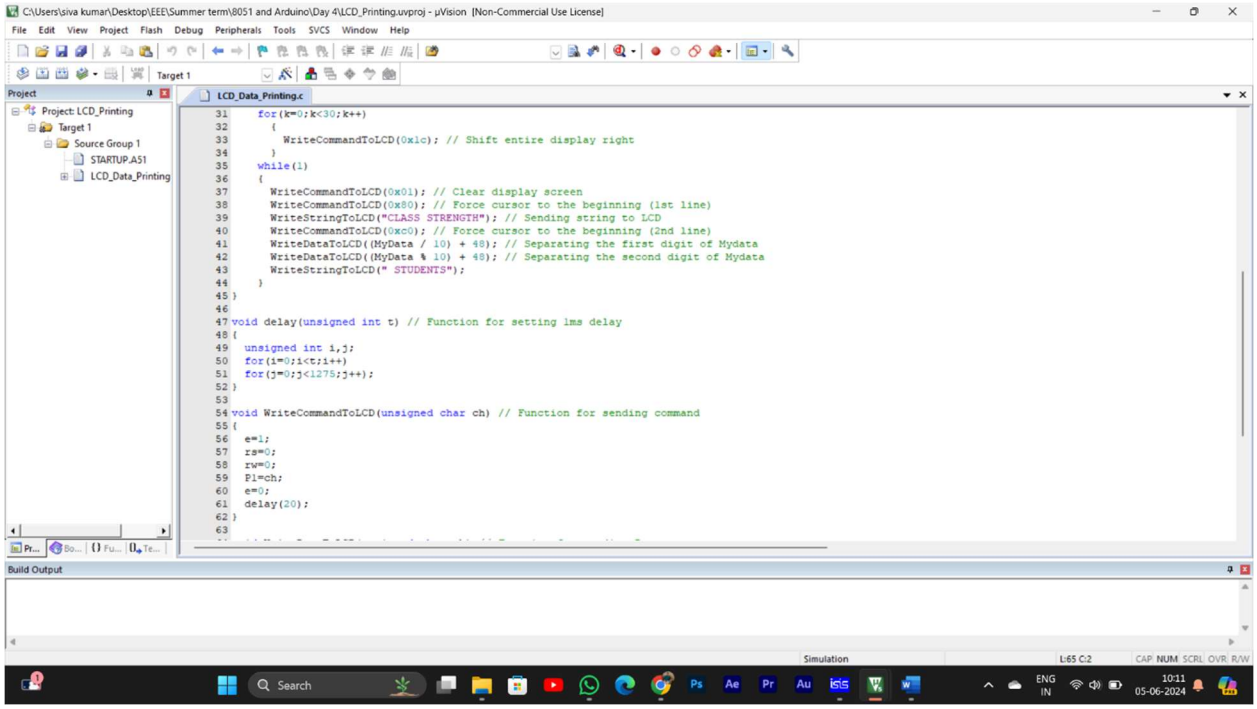
- Implement functions to send commands and data to the LCD.
- Use various LCD commands such as clear display, cursor positioning, and line shifting.
- Convert numerical values to ASCII before displaying them on the LCD.
- Make basic connections to power supply, ground, and enable pins.

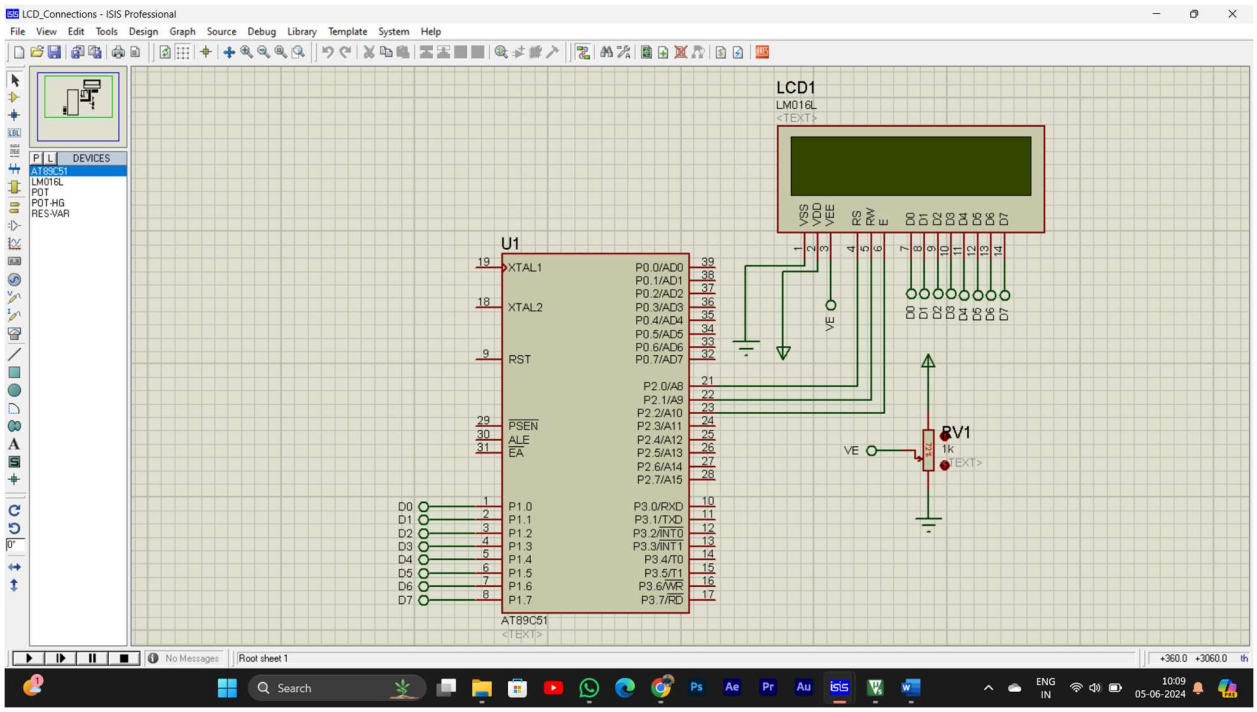
Common Mistakes:

- **Command Mismatch:** Ensure the correct commands are sent to the LCD.
- **Command Positioning:** Correct positioning of commands is crucial for proper display.
- **Clear Display and New Line Commands:** Use appropriate commands to clear the display or move to a new line when needed.




C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\Day 4\LCD_Printing.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: LCD_Printing
Target 1
Source Group 1
STARTUP.A51
LCD_Data_Printing
LCD_Data_Printing.c
49 unsigned int i,j;
50 for(i=0;i<t;i++)
51 for(j=0;j<1275;j++);
52 }
53
54 void WriteCommandToLCD(unsigned char ch) // Function for sending command
55 {
56 e=1;
57 rs=0;
58 rw=0;
59 P1=ch;
60 e=0;
61 delay(20);
62 }
63
64 void WriteDataToLCD(unsigned char ch) // Function for sending Data
65 {
66 e=1;
67 rs=1;
68 rw=0;
69 P1=ch;
70 e=0;
71 delay(20);
72 }
73
74 void WriteStringToLCD(unsigned char ch[]) // Function for sending string
75 {
76 int i;
77 for(i=0;ch[i]!='\0';i++)
78 {
79 WriteDataToLCD(ch[i]);
80 }
81 }
Build Output
Simulation
L:65 C:2
CAP NUM SCRL OVR R/W
Search
ENG
IN
10:11
05-06-2024


C:\Users\siva kumar\Desktop\EEE\Summer term\8051 and Arduino\Day 4\LCD_Printing.uvproj - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Target 1
Project
Project: LCD_Printing
Target 1
Source Group 1
STARTUP.A51
LCD_Data_Printing
LCD_Data_Printing.c
31 for(k=0;k<30;k++)
32 {
33 WriteCommandToLCD(0x1c); // Shift entire display right
34 }
35 while(1)
36 {
37 WriteCommandToLCD(0x01); // Clear display screen
38 WriteCommandToLCD(0x80); // Force cursor to the beginning (1st line)
39 WriteStringToLCD("CLASS STRENGTH"); // Sending string to LCD
40 WriteCommandToLCD(0xc0); // Force cursor to the beginning (2nd line)
41 WriteDataToLCD((MyData / 10) + 48); // Separating the first digit of Mydata
42 WriteDataToLCD((MyData % 10) + 48); // Separating the second digit of Mydata
43 WriteStringToLCD(" STUDENTS");
44 }
45 }
46
47 void delay(unsigned int t) // Function for setting 1ms delay
48 {
49 unsigned int i,j;
50 for(i=0;i<t;i++)
51 for(j=0;j<1275;j++);
52 }
53
54 void WriteCommandToLCD(unsigned char ch) // Function for sending command
55 {
56 e=1;
57 rs=0;
58 rw=0;
59 P1=ch;
60 e=0;
61 delay(20);
62 }
63
Build Output
Simulation
L:65 C:2
CAP NUM SCRL OVR R/W
Search
ENG
IN
10:11
05-06-2024


LCD_Connections - ISIS Professional
File View Edit Tools Design Graph Source Debug Library Template System Help
DEVICES
AT89C51
LM016L
POT
POT-HG
RES-VAR
LCD1
LM016L
<TEXT>
VSS VDD VEE RS RW E D0 D1 D2 D3 D4 D5 D6 D7
U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0 P1.1 P1.2 P1.3 P1.4 P1.5 P1.6 P1.7
P0.0/AD0 P0.1/AD1 P0.2/AD2 P0.3/AD3 P0.4/AD4 P0.5/AD5 P0.6/AD6 P0.7/AD7
P2.0/A8 P2.1/A9 P2.2/A10 P2.3/A11 P2.4/A12 P2.5/A13 P2.6/A14 P2.7/A15
P3.0/RXD P3.1/TXD P3.2/INT0 P3.3/INT1 P3.4/T0 P3.5/T1 P3.6/WR P3.7/RD
AT89C51
<TEXT>
VE
RV1
1k
<TEXT>
No Messages
Root sheet 1
Search
ENG IN
10:09
05-06-2024