

➤ **Learning Objective:**

- ❖ To learn how to blink LED using embedded C programming language using Keil uVision LDE and Proteus Simulation on 8051 microcontroller.

➤ **Input and Output:**

- **Input:**
 - ❖ Two switches (sw1 and sw2) are used as inputs to control the LEDs.
- **Output:**
 - ❖ The output is the state of the LEDs connected to port P1 of the microcontroller, which changes based on the switch inputs.

➤ **Logic:**

- ❖ The program initializes the microcontroller and enters an infinite loop (while(1)).
- ❖ The states of the switches (sw1 and sw2) are checked using conditional statements (if-else).
- ❖ Different patterns are output to port P1 depending on the combination of switch states:
 - ❖ If both switches are off (sw1==0 && sw2==0), port P1 outputs 0x00.
 - ❖ If sw1 is off and sw2 is on (sw1==0 && sw2==1), port P1 alternates between 0xA0 and 0x00 with a 50 ms delay.
 - ❖ If sw1 is on and sw2 is off (sw1==1 && sw2==0), port P1 alternates between 0x0B and 0x00 with a 50 ms delay.
 - ❖ If both switches are on (sw1==1 && sw2==1), port P1 alternates between 0xCD and 0x00 with a 50 ms delay.

➤ **Results:**

- ❖ The LEDs connected to port P1 will light up in different patterns based on the combination of switch states.
- ❖ The specific patterns for the given combinations are:
 - ❖ All LEDs off when both switches are off.
 - ❖ A specific pattern (0xA0 or 0x0B or 0xCD) displayed when one or both switches are on, alternating with all LEDs off with a 50 ms delay.

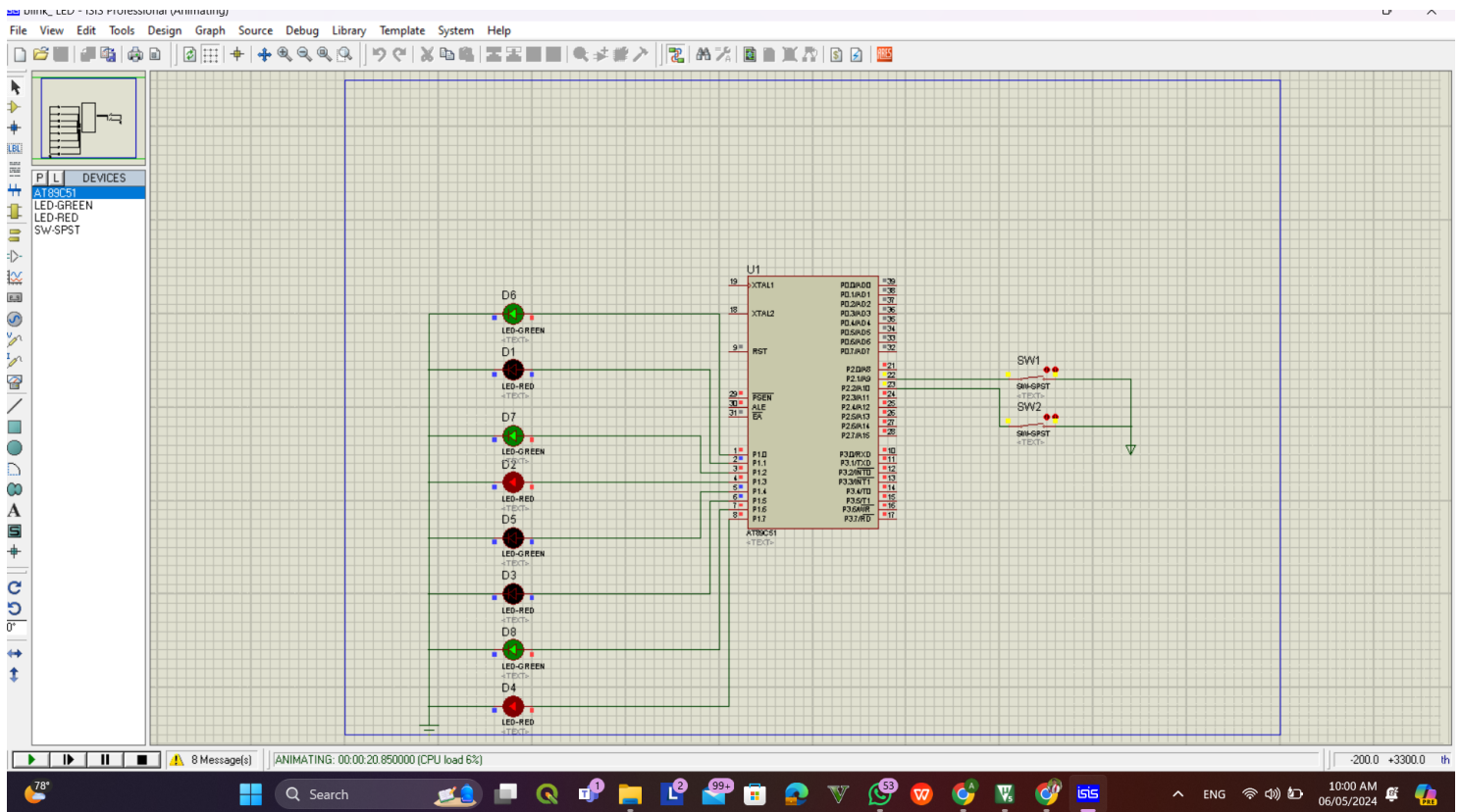
Embedded Systems Report

DAY 1 and 2

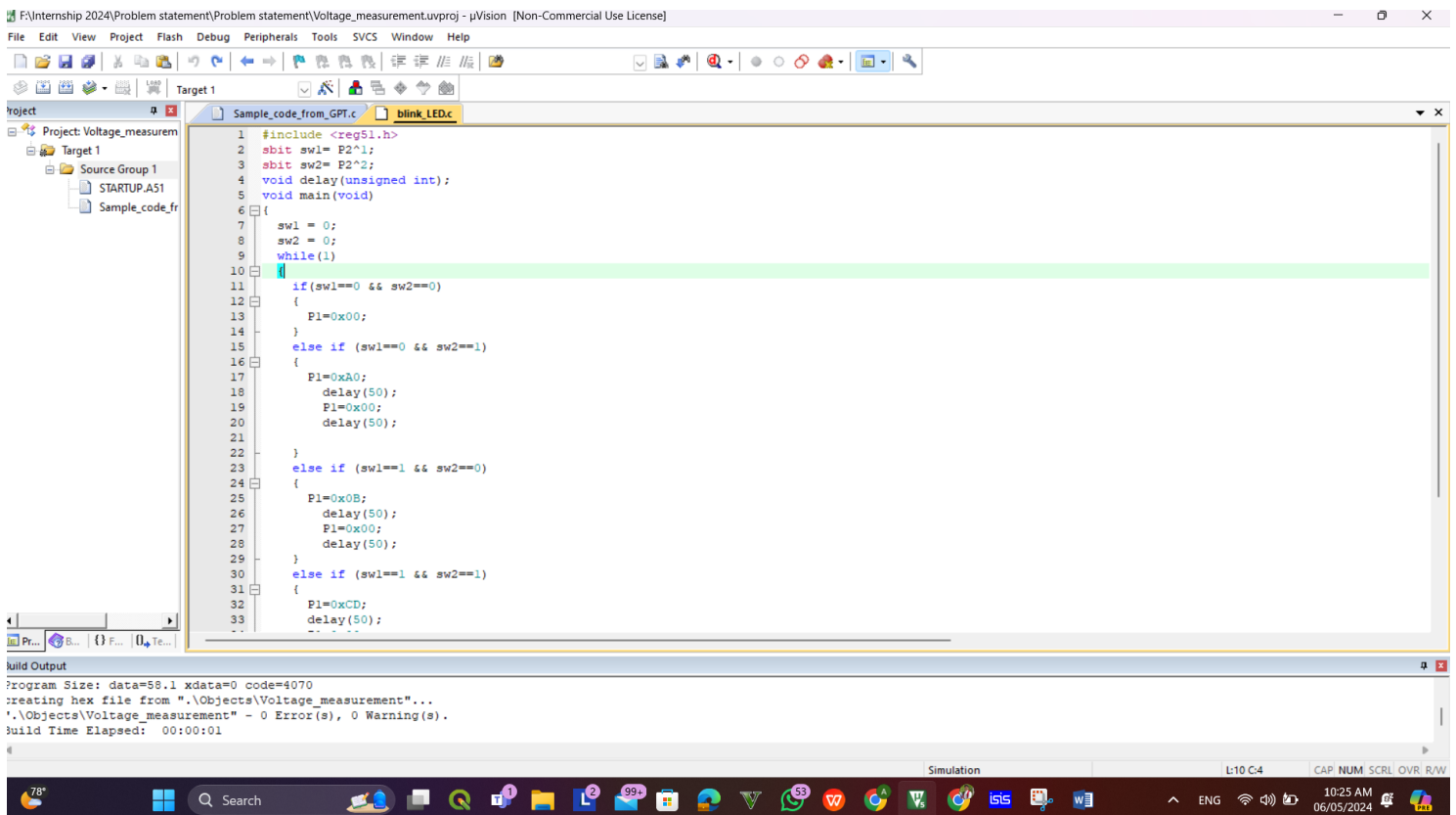
LED Blinking

Ahmed
BU21EECE0200026

➤ Screen Shots:



Proteus Simulation



A screenshot of the Keil uVision IDE showing the source code for the LED blinking project. The code is written in C and is located in the file 'blink_LED.c'. The code includes the 8051 header file, defines the switch pins, and implements a main function that toggles the LEDs based on the state of the switches. The code is as follows:

```
1 #include <reg51.h>
2 sbit sw1 = P2^1;
3 sbit sw2 = P2^2;
4 void delay(unsigned int);
5 void main(void)
6 {
7     sw1 = 0;
8     sw2 = 0;
9     while(1)
10     {
11         if (sw1==0 && sw2==0)
12         {
13             P1=0x00;
14         }
15         else if (sw1==0 && sw2==1)
16         {
17             P1=0xA0;
18             delay(50);
19             P1=0x00;
20             delay(50);
21         }
22         else if (sw1==1 && sw2==0)
23         {
24             P1=0x0B;
25             delay(50);
26             P1=0x00;
27             delay(50);
28         }
29         else if (sw1==1 && sw2==1)
30         {
31             P1=0xCD;
32             delay(50);
33         }
34     }
35 }
```

The bottom status bar shows 'Simulation' and 'L10: C4'.

Keil uVision