

WIPRO NGA Program – Embedded Testing

Capstone Project Presentation – 05 August 2024

Project Title – Basic LED Controller Simulation

Presented by – A Akhil

Introduction

- **Background:**

- This is a Basic LED Controller Simulation project is designed to simulate the functionality of an LED controller within a software environment.
- This project allows for turning LEDs on and off and adjusting their brightness using pulse-width modulation (PWM).
- The simulation is implemented in C and can be run entirely in a software environment using Visual Studio Code without actual hardware.

- **Problem Statement:**

- Simulating the functionality of an LED Controller.

- **Objective:**

- **Simulate LED control:** Implement software to simulate LEDs turning on and off.
- **Adjust LED brightness:** Develop a method to simulate brightness adjustment of LEDs using PWM.
- **Software-only environment:** Ensure all simulations run entirely in software, without the need for physical hardware.

Methodology

- **Approach:**

- **SetUp:** We have installed the following things

- Visual code for running code.
 - GDB debugger for debugging the code

- **Project Structure**

- **The project is organized into three main components:**
 - **Header File (led_controller.h):** Defines the interface and data structures for the LED controller.
 - **Source File (led_controller.c):** Implements the functions declared in the header file.
 - **Main File (main.c):** Contains the main function to test and demonstrate the LED controller functionality.

Code Execution Process

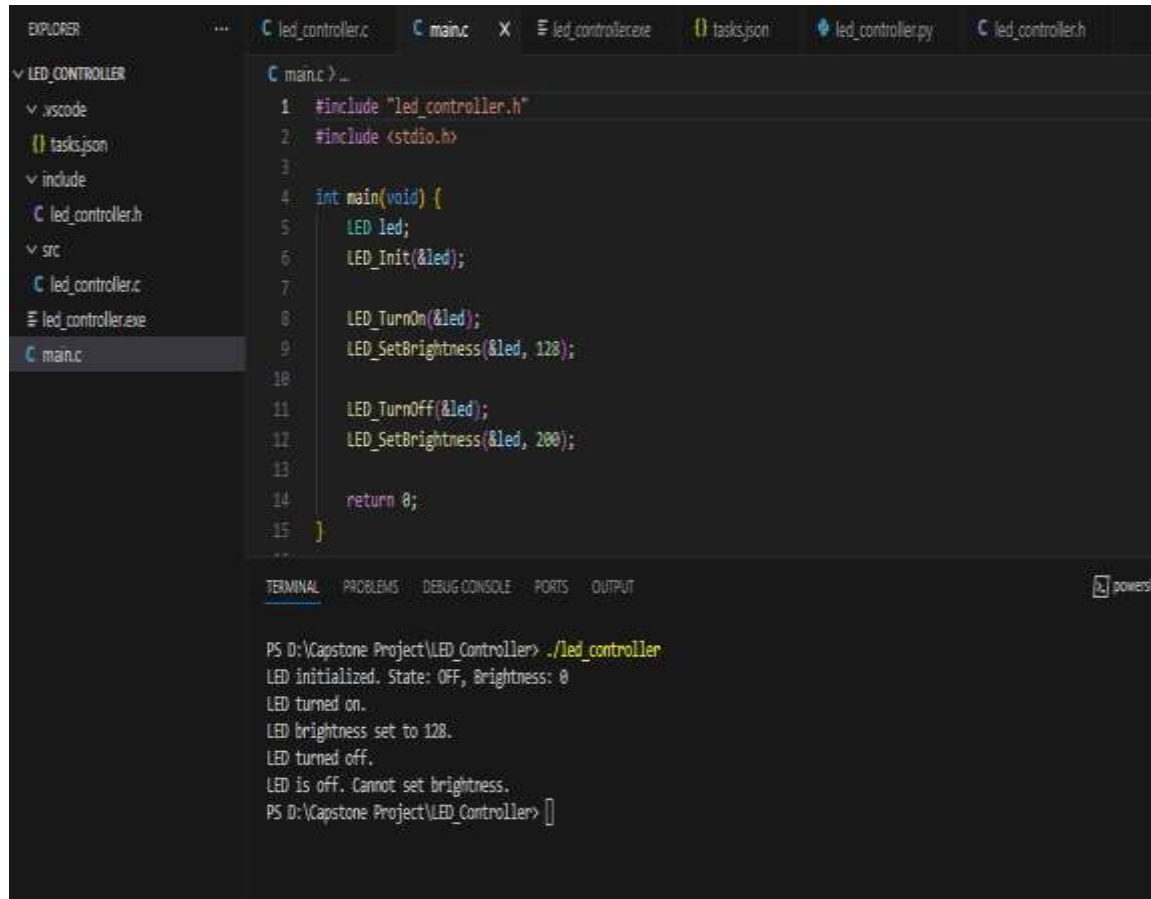
•Steps to Run the Program:

- **Set up the development environment:** Install Visual Studio Code and configure it for C programming.
- **Compile the code:** Use a C compiler to compile the project files.
- **Execute the main.c file:** Run the main function to test the LED controller.
- **Verify output:** Check the console output to verify the LED state and brightness levels.

•GDB Debugging Steps:

- **Set breakpoints:** Place breakpoints in key parts of the code (e.g., main.c, led_controller.c).
- **Start GDB:** Launch GDB and load the compiled executable.
- **Run the program:** Use the run command to start the program within GDB.
- **Step through the code:** Use step and next commands to move through the code line by line.
- **Inspect variables:** Use the print command to check the values of variables and program state.
- **Continue execution:** Use the continue command to resume program execution or terminate the debugging session as needed.

Outputs



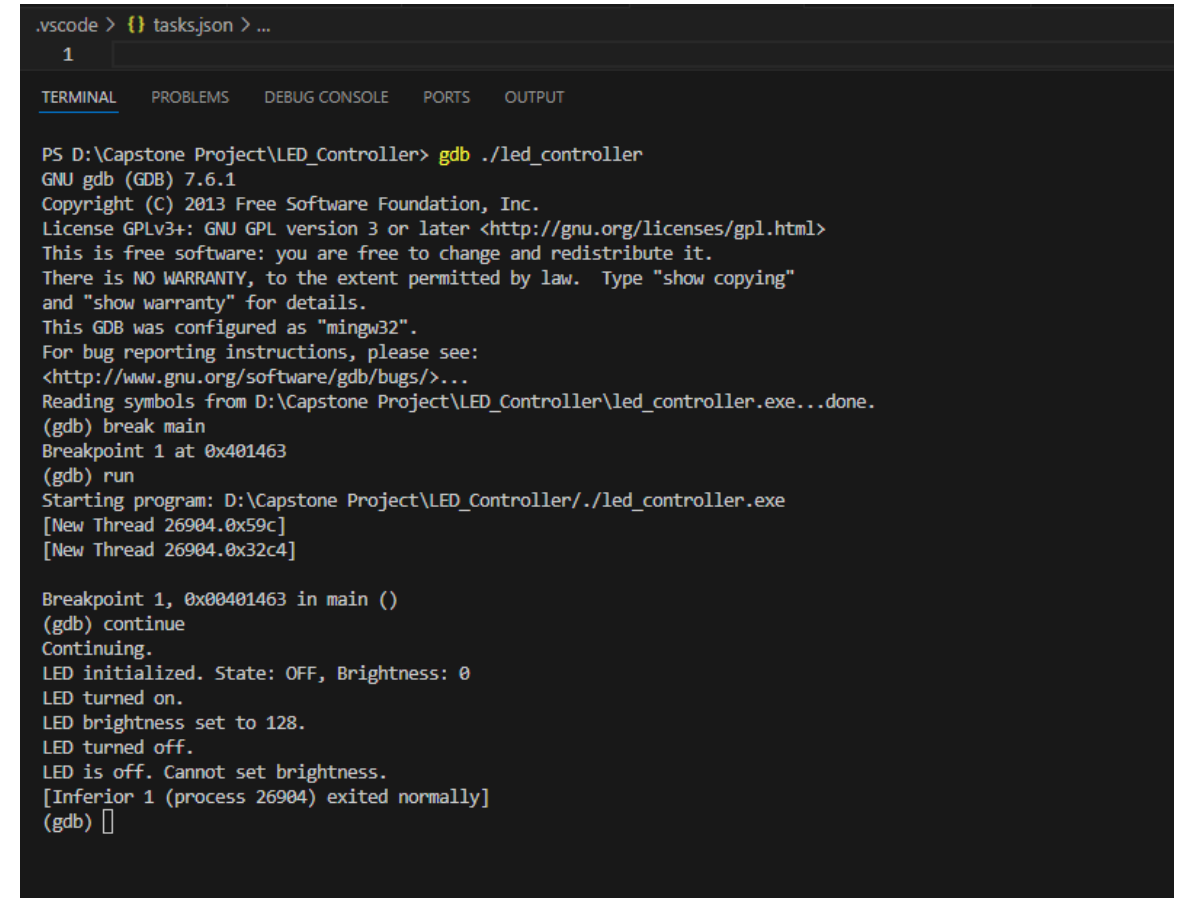
The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure: LED_CONTROLLER, .vscode, tasks.json, include, led_controller.h, src, led_controller.c, led_controller.exe, and main.c. The main.c file is open in the editor, showing the following code:

```
1 #include "led_controller.h"
2 #include <stdio.h>
3
4 int main(void) {
5     LED led;
6     LED_Init(&led);
7
8     LED_TurnOn(&led);
9     LED_SetBrightness(&led, 128);
10
11    LED_TurnOff(&led);
12    LED_SetBrightness(&led, 200);
13
14    return 0;
15 }
```

The terminal at the bottom shows the output of the program:

```
PS D:\Capstone Project\LED_Controller> ./led_controller
LED initialized. State: OFF, Brightness: 0
LED turned on.
LED brightness set to 128.
LED turned off.
LED is off. Cannot set brightness.
PS D:\Capstone Project\LED_Controller>
```

Fig.1: Building and running code



The screenshot shows the Visual Studio Code editor with the tasks.json file open in the editor. The terminal at the bottom shows the output of the GDB debugging session:

```
.vscode > {} tasks.json > ...
1

TERMINAL PROBLEMS DEBUG CONSOLE PORTS OUTPUT

PS D:\Capstone Project\LED_Controller> gdb ./led_controller
GNU gdb (GDB) 7.6.1
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "mingw32".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from D:\Capstone Project\LED_Controller\led_controller.exe...done.
(gdb) break main
Breakpoint 1 at 0x401463
(gdb) run
Starting program: D:\Capstone Project\LED_Controller\./led_controller.exe
[New Thread 26904.0x59c]
[New Thread 26904.0x32c4]

Breakpoint 1, 0x00401463 in main ()
(gdb) continue
Continuing.
LED initialized. State: OFF, Brightness: 0
LED turned on.
LED brightness set to 128.
LED turned off.
LED is off. Cannot set brightness.
[Inferior 1 (process 26904) exited normally]
(gdb)
```

Fig.2: Executing gdb debugging.

Conclusion

- Summary:

1. We have runned and debugged a simple led controller simulation program with any hardware implementation
2. By following the described steps, one can successfully simulate LED control operations without the need for physical hardware