**Praktik Real Hardware ESP32**

*Pelangi Anggel*

Fakultas Vokasi , Universitas Brawijaya

Email: pelangianggel283@gmail.com

**Abstract**

SP32 is a powerful microcontroller widely used in embedded systems due to its integrated Wi-Fi and Bluetooth capabilities. In this practice, an ultrasonic sensor is connected to a real ESP32 board to measure object distance in front of the sensor. Two LEDs—green and red—are also used to indicate distance status: green for safe distance and red for warning when the object is too close. This real hardware implementation enables participants to understand how sensors and indicators work together, providing essential skills in microcontroller programming, circuit design, and real-time response systems.

*Keywords—ESP32, Ultrasonic Sensor, LED Indicator, Real Hardware, Distance Measurement*

**1. Introduction**

**1.1 Background**

ESP32 is a versatile microcontroller that is frequently used in automation, robotics, and IoT applications. When paired with an ultrasonic sensor, it can measure distances without physical contact—making it suitable for obstacle detection and monitoring systems. To enhance this project, two LEDs (green and red) are added as visual indicators. The green LED indicates a safe distance, while the red LED signals that an object is too close. This practice provides a hands-on experience with real hardware and helps build foundational skills in electronics and embedded systems development.

**1.2 Objectives**

The purpose of this project is to learn how to interface an ultrasonic sensor with an ESP32 microcontroller in a real hardware environment, and to integrate visual indicators using LEDs. This includes sensor data processing, conditional logic for controlling LEDs, and real-time feedback using the serial monitor. By the end of this practice, participants will understand how to apply distance measurement and indicator logic in practical scenarios such as obstacle warning systems..

**2. Methodology**

**2.1 Tools & Materials**

 **Microcontroller:** ESP32 development board

 **Sensor:** Ultrasonic sensor (HC-SR04)

 **Actuators:** 1 Green LED, 1 Red LED

 **Resistors:** 2 × 220 ohm (for the LEDs)

 **Wires and Breadboard**

 **Power Supply or USB Cable**

 **Software:** Arduino IDE

 **Computer with USB port and Internet access**

* 1. **Implementation Steps**

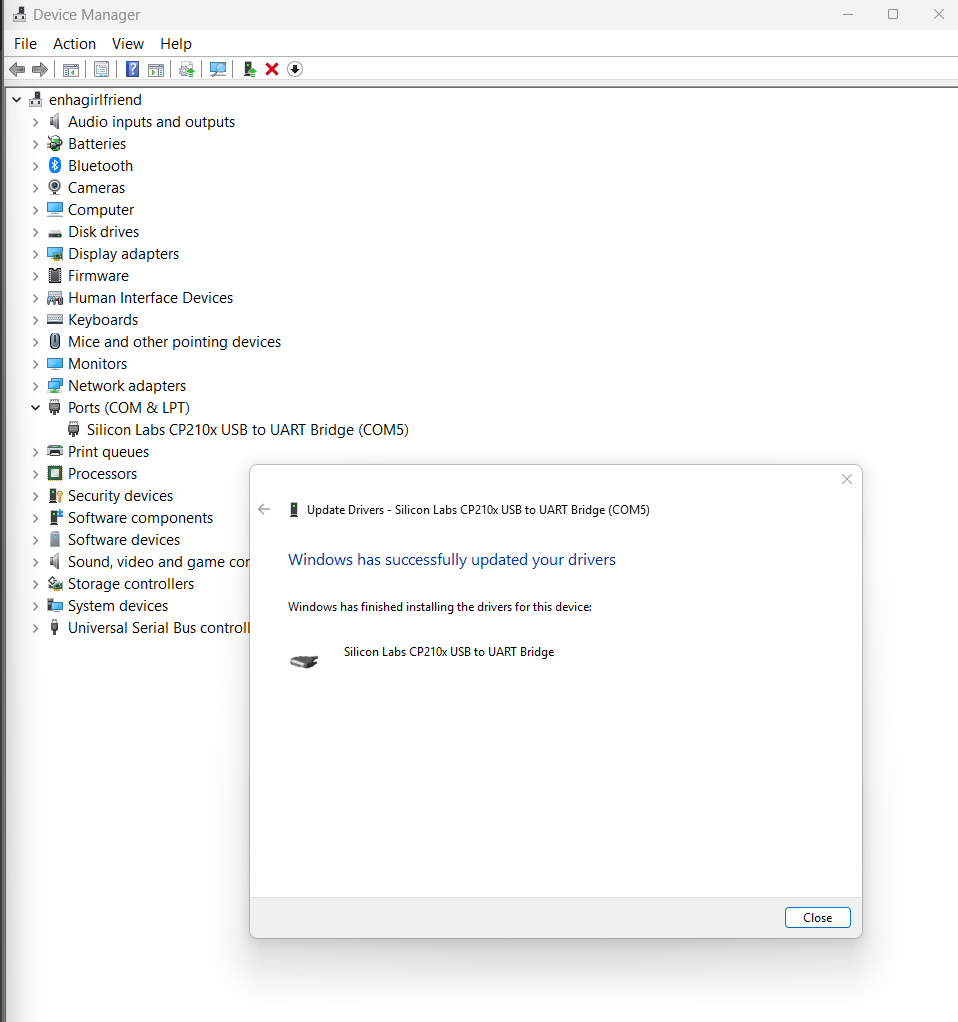
1. Ensure the ESP32 is recognized by the computer:
2. Connect the ESP32 board to your computer using a USB cable.
3. Open Device Manager in Windows and check under Ports (COM & LPT).
4. If you see "Silicon Labs CP210x USB to UART Bridge", your device is correctly recognized.
5. If not, install the CP210x driver manually:
   1. Download the driver from:  
      https://www.silabs.com/developer-tools/usb-to-uart-bridge-vcp-drivers
   2. Extract the downloaded ZIP file.
   3. In Device Manager, right-click the unknown device → Update Driver.
   4. Choose Browse my computer for drivers, then point to the extracted folder.
   5. Follow the prompts until the installation is complete and the device is recognized.
6. Connect the ultrasonic sensor and LEDs to the ESP32 as follows:
7. **Open Visual Studio Code and run PlatformIO:**
   1. Create a new folder and open it in VSCode.
   2. In PlatformIO, create a new project using the board: esp32doit-devkit-v1.
   3. Modify the platformio.ini
8. Open Visual Studio Code and run PlatformIO:
   1. Create a new folder and open it in VSCode.
   2. In PlatformIO, create a new project using the board: esp32doit-devkit-v1.
   3. Modify the platformio.ini

**3. Results and Discussion**

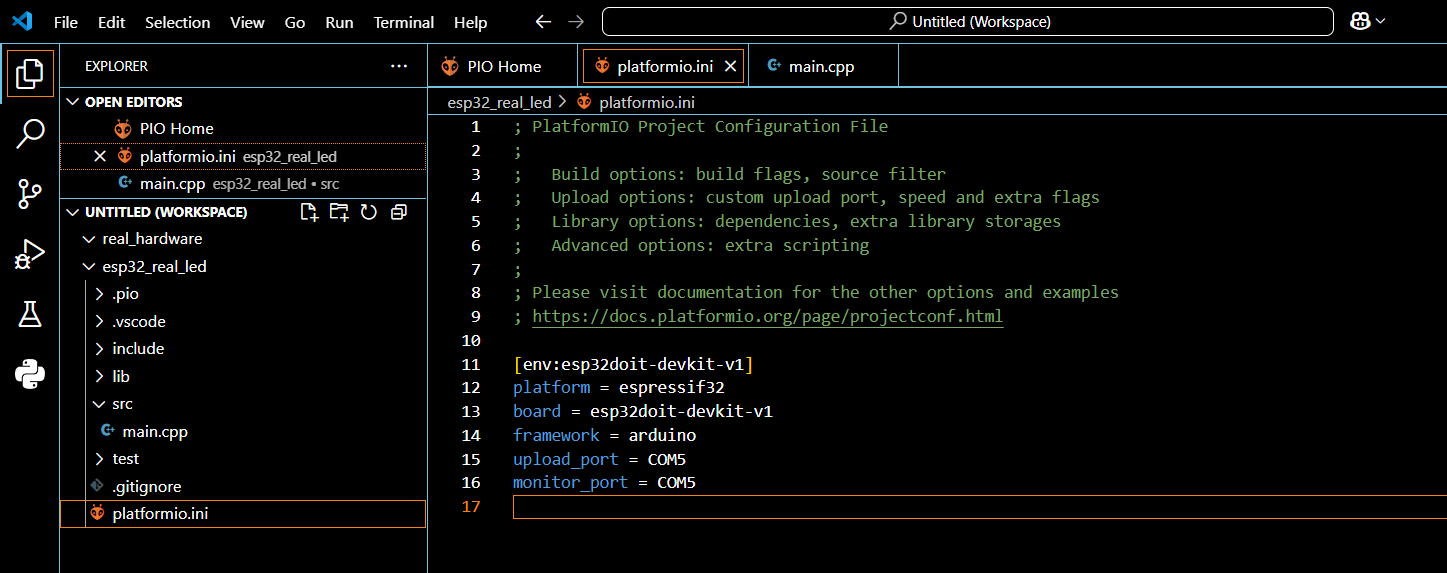
**3.1 Experimental Results**

|  |  |  |
| --- | --- | --- |
| **Platform** | **Completed Task** | **Key Outcome** |
| Wokwi | Ultrasonic sensor simulation | The ultrasonic sensor successfully measured distance and displayed real-time data in the simulation. |
| ESP32 (Real Hardware) | Sensor and LED control | The ESP32 successfully read real-time sensor data, activated the red LED when objects were close (≤ 20 cm), and turned on the green LED when the distance was safe (> 20 cm). |

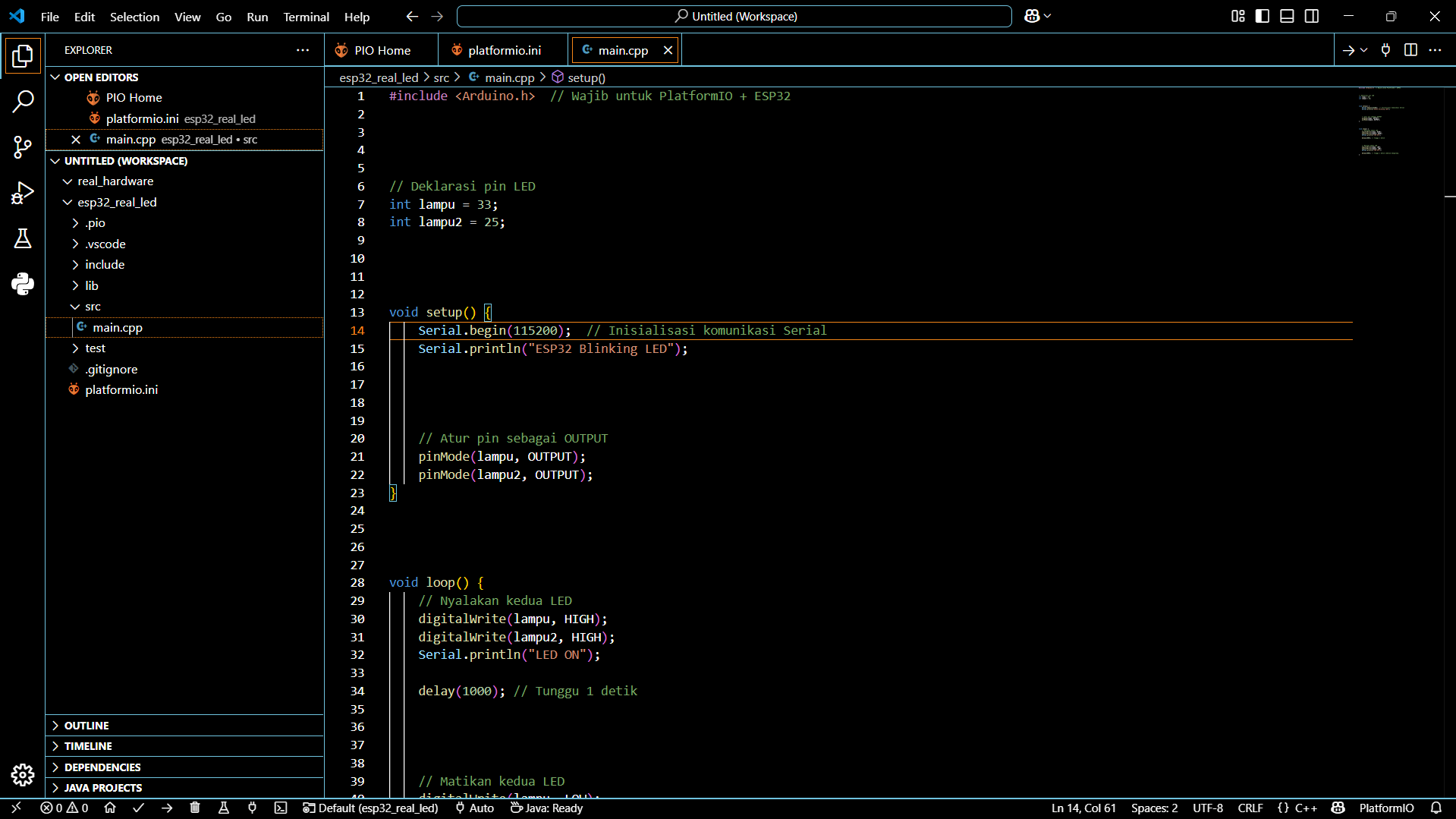
Driver Update result after downloaded and extract the driver file:



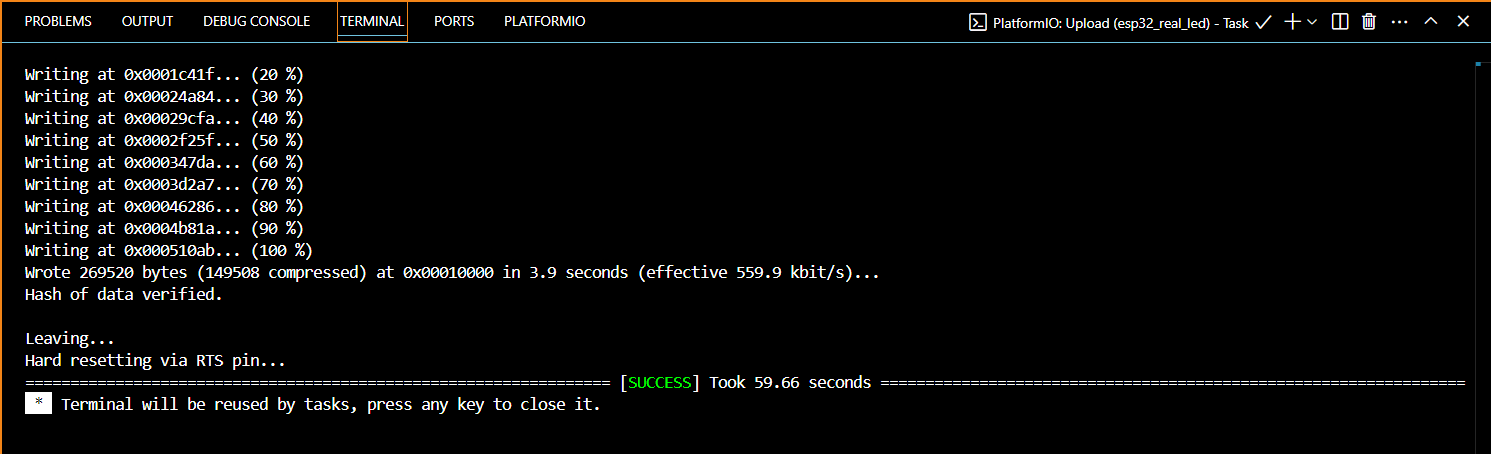
Platformio.ini file result:



Main.cpp file result:



Upload terminal result:



The ESP32 and lamp result:

