Autonomous Car Mega Project

Class 2

Under the supervision of Eng. Hesham Salah & Eng. Mohamed Tarek

Name	Code
Kareem Hany Mohamed Saad	890
Mohamed Ahmed Abdelaziz	806
Hussam Mustafa Mohamed	528
Marco Ashraf Alfy	1265

Contents

Introduction

Our Project Represents Autonomous Car that is implemented with Real Time Operating System using non-preemptive schedular that plays pivotal role in managing the tasks in our project.

This Report outlines the design and implementation of the schedular, delineates the tasks involved and presents a layered project Structure.

Additionally, this project strictly adheres to register-level configuration of peripherals, without recourse to TivaWare libraries, ensuring a comprehensive understanding of hardware interfacing.

Project Levels

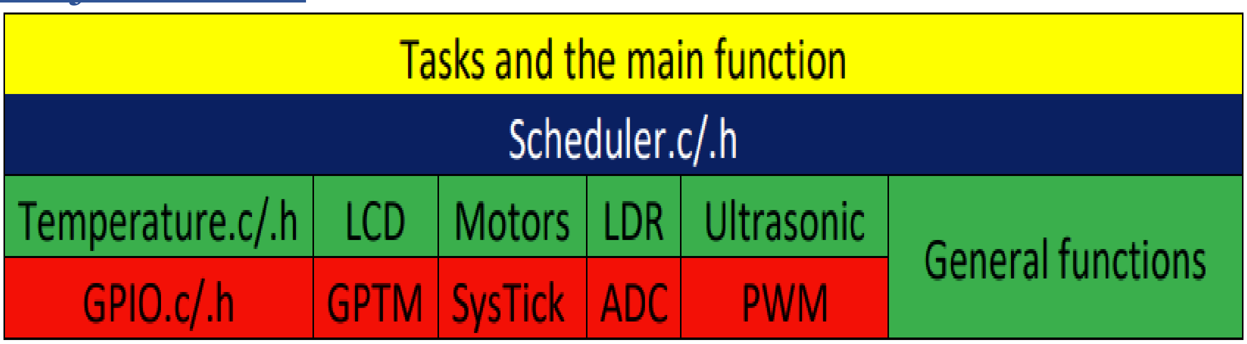

Figure 1 Project Levels

Sensors Quick Overview

1) Ultrasonic Sensor

- **Principle of Operation: Ultrasonic Sensors operate on the principle of sending out high-frequency sound waves and measuring the time it takes for the sound waves to bounce back after hitting an object. This time measurement is used to calculate the distance between the sensor and the object.**
- **Components:**
  - **Transmitter: Emits ultrasonic waves.**
  - **Receiver: Detects the reflected waves**
  - **Controller circuitry: Processes the received signal and calculates distance.**
- **Key Features:**
  Non-contact Sensing: Ultrasonic sensors do not require physical contact with the object being measured, making them suitable for various applications.

  Accuracy: They provide reasonably accurate distance measurements, especially at shorter ranges.

  Range: The range of an ultrasonic sensor can vary from a few centimeters to several meters, depending on the specific model.

  Environmentally Robust: They work well in different environments, including air, water, and other gases.

- **Usage in the Project: The Ultrasonic sensor is employed to detect obstacles within a specific range. When an obstacle is detected within 10 centimeters of the vehicle, it triggers a reversal of direction of 90-degrees rotation**

2) LDR Sensor

- **Principle of Operation: LDRs work on the principle of photoconductivity. They are made of semi-conductor materials whose electrical conductivity changes when it is exposed to light. In the presence of light, the resistance of an LDR decreases and in darkness, the resistance of LDR increases.**
- **Components:**
  - **Sensing element: The heart of LDR is a semi-conductor material sensitive to light.**
  - **Enclosure: protects the sensing element and may have a transparent cover to allow light in.**
  - **Electrodes: Connect the LDR to an electrical circuit.**
- **Key Features:**
  - **Light Sensitivity: LDRs are highly sensitive to changes in light levels, making them suitable for applications involving light detection.**
  - **Passive Component: They do not require a power source or external voltage to operate.**
  - **Response Time: LDRs respond to changes in light almost instantly. Wide Range of Resistance Values: Different LDRs have different resistance ranges, allowing for flexibility in applications.**
- **Usage in the project: The LDR Sensor is employed to determine the level of illumination in the environment as it will be used to make decisions regarding the direction and the movement of the vehicle.**

3) Temperature Sensor (Tiva C)

- **Principle of Operation: The temperature sensor works based on the principle that a silicon diode's forward voltage drop varies with temperature. By measuring this voltage drop, the microcontroller can estimate the temperature of the Tiva-C.**

- **Usage in the project: It monitors the temperature of the environment.**

Circuit Topology

- **F1 & F2 (Pins from Tiva C) → PWM**
- **E3 (Pin from Tiva C) → Temperature Sensor**
- **E1 & E2 (Pins from Tiva C) → LDR Sensor**
- **C4 (Pin from Tiva C) → Echo Pin in Ultrasonic sensor**
- **C5 (Pin from Tiva C) → Trigger Pin in Ultrasonic sensor**
- **F0 (Pins from Tiva C) → Start Button**
- **F1 (Pins from Tiva C) → Stop Button**
- **A3 (Pin from Tiva C) → RS pin in LCD**
- **A2 (Pin from Tiva C) → Enable pin in LCD**
- **A4-A7 (Pins from Tiva C) → Data pins in LCD**

Project Components Layout

- The Whole Module Layout

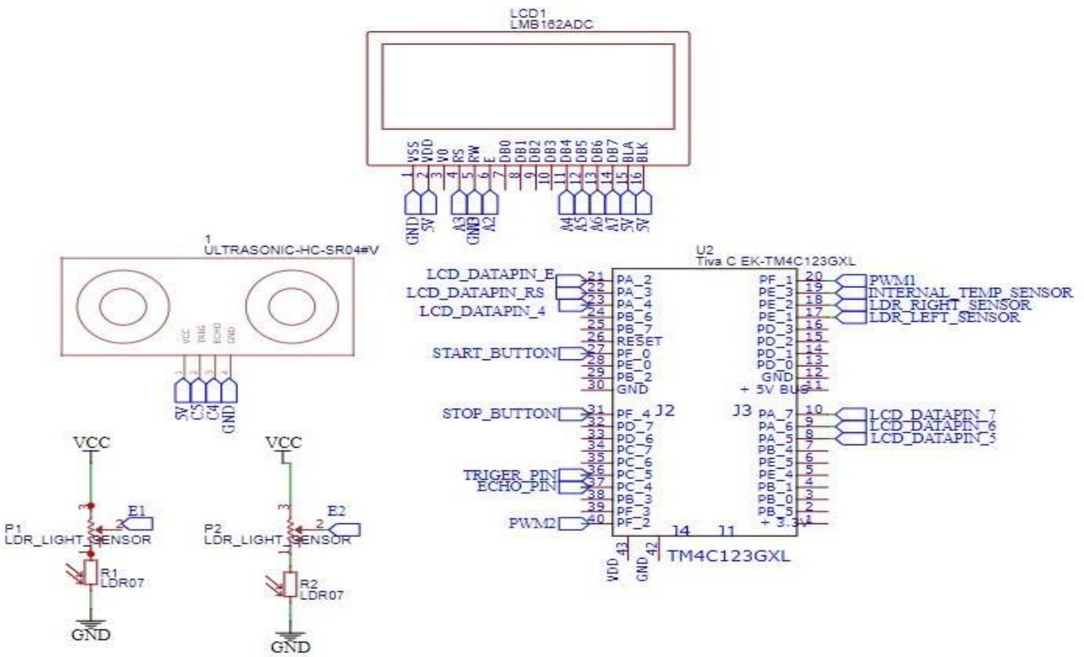

Figure 2 The whole Module Layout

- The Ultrasonic Layout

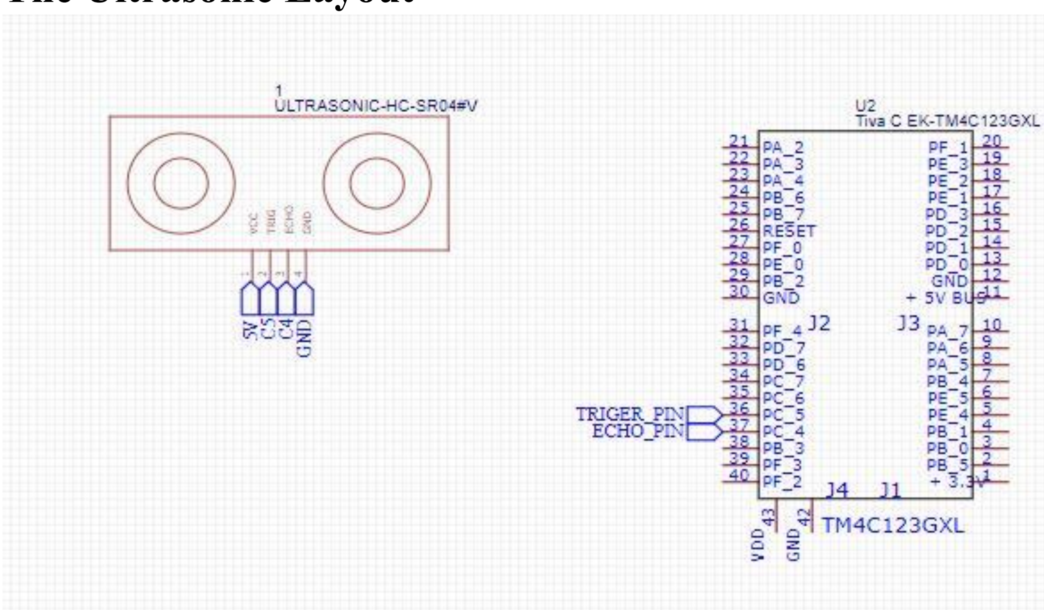

Figure 3 The ultrasonic Layout

- The LCD Layout

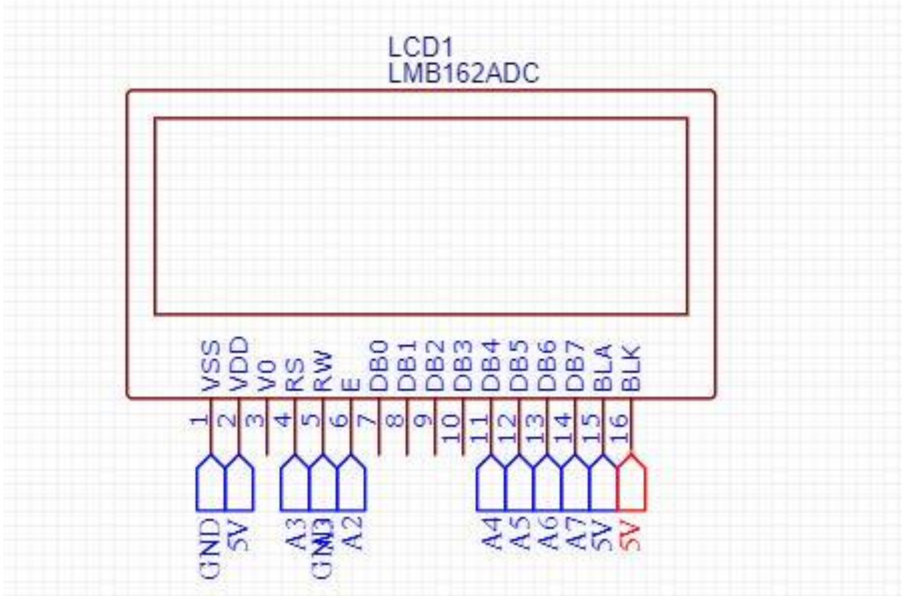

Figure 4 The LCD Layout

- The LDR Layout

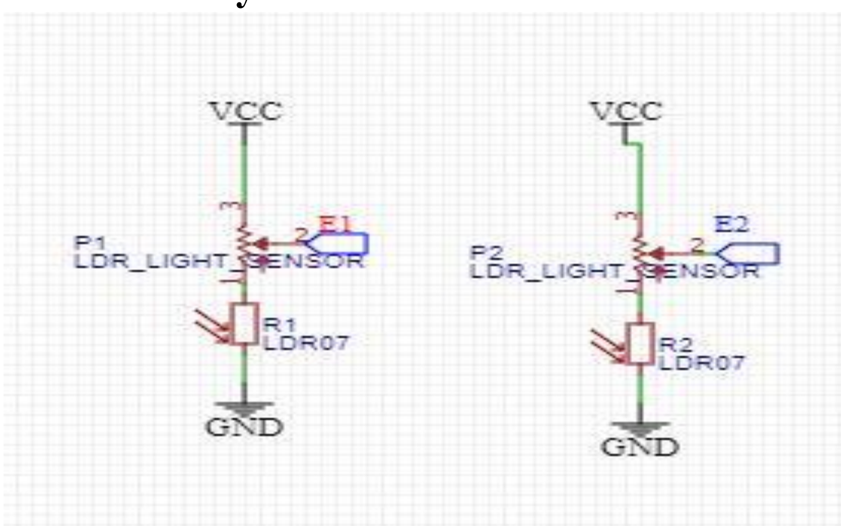

Figure 5 The LDR Layout

Validation & Verification

This Project has undergone rigorous validation and verification processes to ensure its functionality, reliability and adherence to design specification. Extension Testing experiments have been conducted to confirm the correct operation of all components and functionalities.

The implemented schedular, tasks and hardware interfaces have been thoroughly assessed. Additionally, the project has been evaluated for robustness under various operating conditions.

Overall, the project has successfully passed validation and verification stages, affirming its readiness for deployment in practical applications.

Pseudo- Code

```
/* Main Function */
int main(void)
{
    /* Initialization */
    Initialize push button, DC motor, LCD, Ultrasonic, LDR, internal temperature sensor, and scheduler.

    Create a task (Task_LDR) with a periodicity of 100ms.
    Create a task (Task_ultrasonic) with a periodicity of 1000ms.
    Create a task (Task_LCD) with a periodicity of 350ms.

    Start the scheduler.

    /* Infinite Loop */
    while (1)
    {
        // The program will not reach here as tasks are managed by the scheduler.
    }
}

/* Ultrasonic Task */
void Task_ultrasonic(void)
{
    // Read distance from the ultrasonic sensor and store it in 'obstacleDistance'.

    // Check if 'obstacleDistance' is greater than 99, limit it to 99.
    // Check if 'obstacleDistance' is less than 1, limit it to 1.

    // If 'obstacleDistance' is less than 10:
    //   - Stop both motors
    //   - Wait for 20ms
```

```
    //   - Move backward for 100ms
    //   - Stop both motors
    //   - Wait for 10ms
    //   - Rotate 90 degrees
    //   - Stop both motors
    //   - Wait for 10ms
    //   - Continue moving forward

    // Otherwise, do nothing.
}

/* LCD Task */
void Task_LCD(void)
{
    // Read the temperature from the internal temperature sensor and store it in 'tempInternalSensor'.

    // Display the obstacle distance on the LCD.
    // Display the temperature on the LCD.
    // Display the reading difference of the two LDRs on the LCD.
}

/* LDR Task */
void Task_LDR(void)
{
    // Read the right LDR value and store it in 'READ_R'.
    // Read the left LDR value and store it in 'READ_L'.

    // Compare the LDR values and determine the car direction (RIGHT, LEFT, or NONE).

    // Depending on the car direction:
    //   - Adjust the DC motor rotations.
}
```