Major Task LV

1. Introduction:

The primary task can be described as a consolidation of the previous tasks, with a specific emphasis on the core content of the learning phase. It encompasses various elements, including communication protocols, power electronics, and dashboard interfacing, which play a significant role in the design phase.

2. Project objective:

2.1 Dashboard

The primary objective is to begin by creating a dashboard interface that presents four main pieces of information: the speed of the car, the voltage of the battery, the status of the parking sensor, and the state of the car door's lock. Additionally, the touchscreen will be used to send data relating to the activation of the front lights, the turning flashers, and the horn. The next step involves establishing a connection between the mentioned dashboard device (7" DWIN LCD model DMT80480C070_16WT) and the Main Arduino board via serial communication.

2.2 Main Arduino

To connect the main Arduino, which is already linked to the dashboard, you'll need to wire a limit switch in a pulldown configuration as a representation of the car lock. Additionally, attach a buzzer to serve as the horn. Power electronic circuits are necessary to act as light drivers for the front and flasher lights. Once these components are set up, establish a serial communication between the main Arduino and the secondary one to enable smooth data exchange and instructions between the two Arduinos.

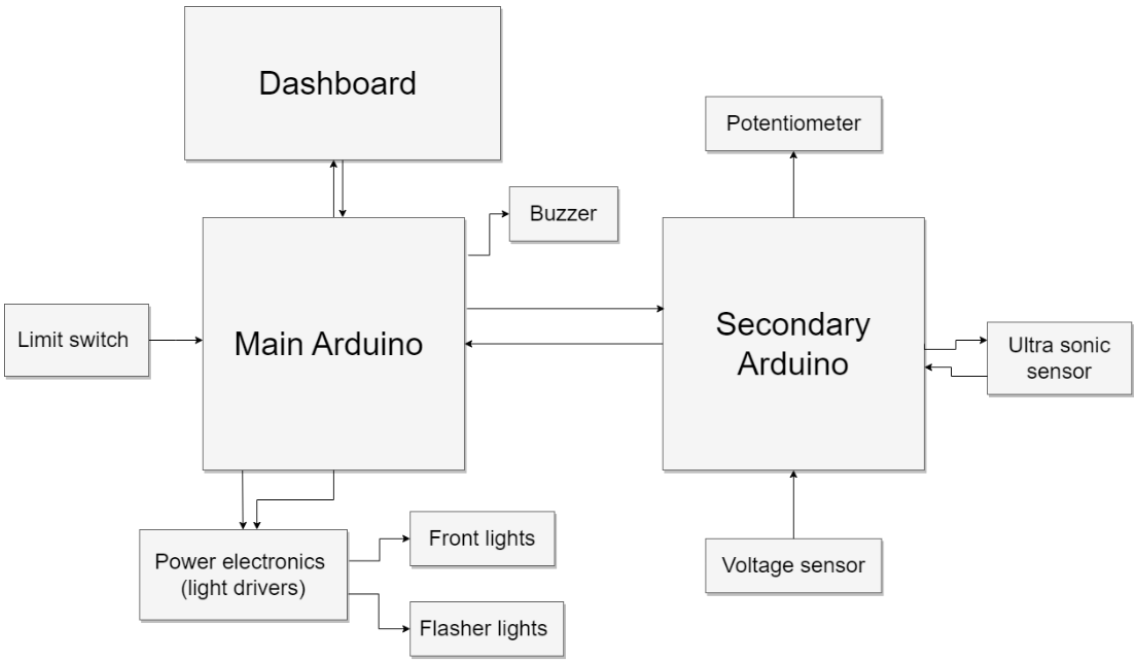
- For the front and flasher lights, you'll **require power electronic circuits of your design** to function as light drivers. These circuits will allow the Arduino to control the lights effectively.

2.3 Secondary Arduino

For the secondary Arduino, which is connected to the main Arduino, we will incorporate several components. Firstly, we will attach a potentiometer to serve as a speedometer. By utilizing the potentiometer's read value and use it as motor RPM (revolutions per minute) and calculate the speed in kilometers per hour based on the 17-inch wheel diameter. The speed will be represented as the nearest integer.

Next, an ultrasonic sensor will be integrated as a parking sensor. This sensor will provide feedback on the distance between the vehicle and obstacles in its vicinity, assisting with parking maneuvers.

Lastly, a voltage sensor will be employed to detect the battery's voltage. This will enable us to monitor the battery's energy level and ensure it remains within the desired range.



Circuit illustration block diagram

3. Project sequence:

3.1 First part

To create a graphical user interface (GUI) using the LCD and the provided application:

- Use the application to initialize the LCD and set up the necessary parameters.
- Create different GUI elements such as text boxes, buttons, and graphical representations using the LCD library functions.
- Define the data representation and visualization according to your requirements. For example, you can display the motor RPM, speed in kilometers per hour, distance measured by the ultrasonic sensor, and battery voltage in separate text boxes or graphical bars.
- Map the touch commands to corresponding actions on the GUI elements. For instance, if you want to change a value or modify a setting, you can associate touch events with specific functions or methods.
- Implement the necessary logic to update the displayed values as the input data changes. This can be achieved by continuously reading the potentiometer, ultrasonic sensor, and voltage sensor values within the Arduino code.
- Test the GUI by uploading the code to the Arduino and observing the displayed data and touch functionality on the LCD.

3.2 Second part

Build your own power electronics drivers for the front LEDs or the flashers, you are allowed to use any application to test it if you want then implement it in the final simulation on Proteus.

3.3 Third part

Simulate the entire circuit using Proteus, including the main Arduino, secondary Arduino, sensors, and drivers:

- Open Proteus and create a new project.
- Search for and add the Arduino component to the schematic diagram of your project. This will represent your main Arduino.
- Connect the necessary components to the main Arduino. This includes the, sensors, and drivers mentioned previously.
- Search for and add the Arduino component to the schematic diagram. This will represent your secondary Arduino.
- Connect the necessary components to the secondary Arduino, such as potentiometer or additional sensors.
- Configure the pins and communication protocols according to your circuit design in the software environment of Proteus.
- Set up the interaction between the main Arduino and the secondary Arduino, such as sending/receiving data or commands via serial communication.
- Run the simulation to observe the behavior of your circuit. Verify that the GUI on the LCD behaves as expected and that the interaction between the main and secondary Arduino functions correctly.

- Make any necessary adjustments or modifications to your circuit design or code based on your observations during the simulation.
- Remember to refer to the Proteus documentation or online tutorials for specific instructions on how to add components, configure pins, and set up simulations.

4. Project Submission:

This submission holds great importance as it carries significant weight in the filtration phase. It is crucial to adhere to the submission deadline, as late submissions are not accepted. Furthermore, the quality of the submission and its organization are key factors that are taken into consideration. It is essential to ensure that the submission is of high quality and well-structured.

Part of the project requirements, all submitted designs must now be uploaded to a GitHub repository. This will serve as a centralized platform for sharing and storing your innovative creations. When preparing your project report, please ensure that you include the link to your GitHub repository for easy access by others. This will allow fellow participants and evaluators to review and learn from your work.

https://forms.gle/LKVtz56pZcDhgFDB7

Deadline of the First part: Friday 3/11/2023 at 11:59pm

Deadline of the fully integrated project: Monday 20/11/2023 at 11:59pm

Good luck ^_^