

**Embedded Project**

Mohamed Ghazouly 49-4660

Table of contents:

a) A cover page containing the team number, name, and members information.

b) Brief description .

c) The components used and their functionalities.

d) The project full circuit

e) The names of the libraries used and their functions.

f) Handling Inputs

g) Handling Outputs

h) how the features were prioritized and divided into tasks using freeRTOS.

i) The problems or limitations faced during the implementation.

**B.**

In this project we designed a car using Arduino Uno and Mega boards, giving each feature a suitable priority. We used RTOS to divide priority among all functioning components of the car and set priorities for processes with the higher need.

We integrated the following features:-

1. Current Gear.

• Display the current gear on a 7-Segment display.

2) Adaptive Headlights.

• Adapt the car headlights according to the surrounding light intensity.

3.Lane Assist.

• Autonomous lane departure system that detects if the vehicle is leaving its lane using sensors.

• It will gently steer the car back into the lane if it begins to drift out of it while alerting the driver.

Seeking internships in Multinational companies and joining student activities in university to enrich my skills and gain experience in different fields.

**C.**

We used the following components to design the car:

1. Arduino UNO Wi-Fi edition

* Used tom control the Bluetooth RC
* Motor driver
* Infrared sensors
* Adaptive headlights using photo resistor.

1. Arduino MEGA 2560 controls:

* Joystick module.
* 7 segment display.
* 2.8inches TFT touch screen.
* 16 Ohm Speakes.
* Sd card module.

1. L298N Motor Driver used to

* Controls speed of motors
* Control direction of motors

1. Infrared Sensor 3 Pins

* Detects if car departs the lane.
* Sends control to revert the car back.

1. Bluetooth module HC-06

* Receive command form Android Bluetooth mobile.
* Controls the car to move in four directions: forward, backward, left, right.

1. Photo Resistor.

* Detects current light level.
* Helps in controlling front facing Leds.

1. Front facing Leds

* The change intensity according to. Current light level.
* Helps in controlling front facing Leds.

1. 3 18650 Battery Holder.

* Power the motor driver that helps powers the whole circuit.

1. MAR2808 2.8 inchTFT Touchscreen

* Controls the sound system
* Play/Pause.
* Previous.
* Next.
* Display Song number.

1. SD card module HW-125

* Fetch mp3 songs to be processed by the Arduino MEGA.

1. Speaker 16 ohm

* Receives sampled .WAV music songs.
* Play the .WAV songs.

1. Potentiometer 1K ohm

* Controls the level of the sound.

1. Photo Resistor.

* Detects current light level.
* Helps in controlling front facing Leds.

1. Buzzer.

* Beeps when the car drifts out of lane to notify the driver.

1. Analog joystick module.

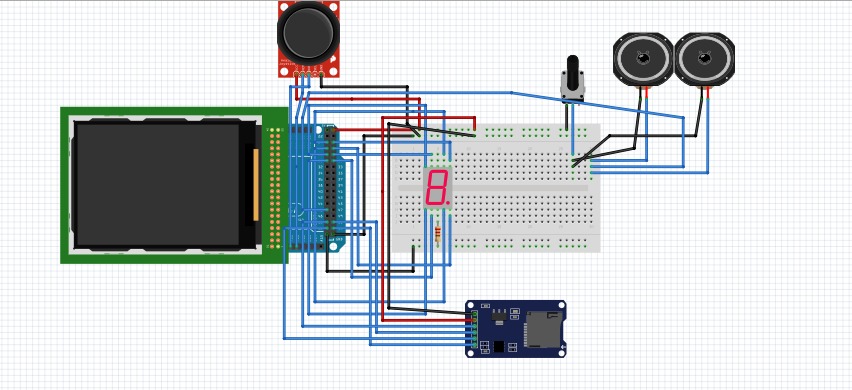
* Controls the current gear to be displayed on 7-Segment.

1. 7-Segment display

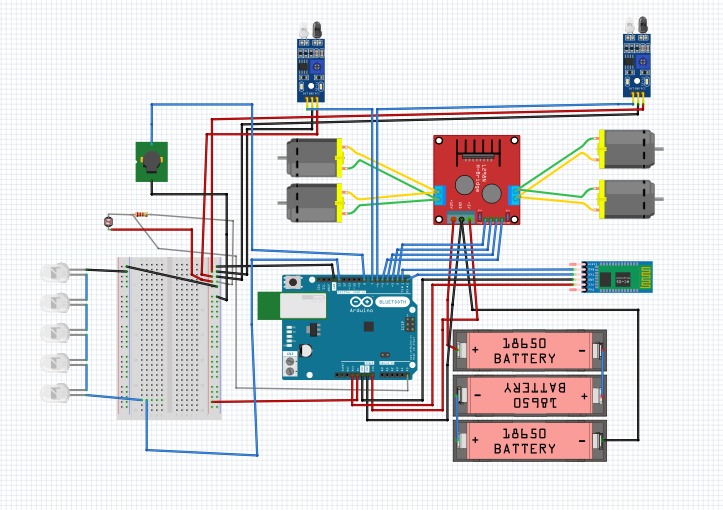
* Displays the current gear that is received from the joystick.

**D.**

**The Top floor**



**THE BOTTOM FLOOR**



**E.**

WE USED THE FOLLOWING LIBRARIES:

1. **< Arduino\_FreeRTOS>.**

* FreeRTOS is an open source, real time operating system for microcontrollers that makes small, low-power edge devices easy to program, deploy, secure, connect, and manage.

1. **< TouchScreen>**

* Used to control the touch functionality of the screen

1. **< LCDWIKI\_GUI >**

* Core Graphics library used to draw buttons and fill the screen

1. **< LCDWIKI\_KBV >**

* Hard ware specific library to interface with Mar2808

1. **< SD>**

* Used to interface with the SD card module

1. **< TMRpcm>**

* Used for asynchronous playback of PCM/WAV audio files fetched from SD Card.

1. **< SPI >**

* Used to interface with the SPI Bus.

**F. Handling Inputs :**

There are a lot of control inputs sent to each board, each of them was used handled by the same approach of always taking the input in a dedicated variable and pin whether it was a digital input in any of the digital pins i.e. IR sensor

or taking an analog input in any of the analog input pins i.e. Joystick module

and then the dedicated variable for that input is used to control the logic of the car.

**G. Handling Outputs:**

Initially each output pin is set as an output pin in the setup of each board Where each output has a dedicated pin and each small functionality was wrapped in a function and tested individually to ensure their reliability when RTOS was deployed.

**H. Handling RTOS Priorities:**

1. Arduino Uno

It’s responsibilities was to control the movement of the car using the remote control and ensure its safety by staying in its lane.

As well as controlling the ambient light.

Those two tasks where prioritized equally and executed in a round robin scheduler to ensure their continuous updates because the ambient light task had no delays and thus the lane assist task was executed efficiently nearly alone with no flickering in the ambient lights.

1. Arduino Mega

It had two main responsibilities

1. Control the Sound system along with the tft screen
2. Control the Gear selector and the 7 Segment display

It had lots of bugs initially and the 7-segment was flickering but after some code changes and restructuring it is executed efficiently using round robin algorithm with no flickering or distortion of the sound where each task had equal priority to the other task.

1. **Project Limitations.**

From the start of the project we had to think about the physical problems of the car due to its moving nature. Everything had to be fitted exactly where it was designed to be or else the car will break during operations. This forced us for example to have no bread board in the bottom level of the car which required lots of soldering to replace these connections. Also we had to comb the wires efficiently to ensure that the moving wheels don’t cut the wires during movement. Also everything had to be glued in place to ensure their stability. We also had to think a lot about the car’s center of mass or else it would have stability issues that would lead to its destruction. So the heavy battery holder was placed in the bottom back of the car. And all the boards and the front upper power bank was placed upward to ensure that the car’s center of mass remains approximately in the middle of the car which enhanced the stability of the car greatly.