

PHONE CONTROLLED CAR

DOBRETE Andrei-Robert, MIHALACHE Denisa Ana-Maria

Faculty: Industrial Engineering and Robotics, Specialization: Robotics, Year of study: 1, e-mail: andy3153@protonmail.com

Scientific leader: prof. Andrei Mario IVAN

SUMMARY: This project is a Bluetooth controlled car, controllable using a phone. This means it is an extensible base for many different applications.

KEY WORDS: car, robot, Bluetooth.

1. Introduction

Due to the extensibility of the platform, this car can be used for multiple tasks, like:

- RC car
- car that can go through labyrinths
- car that can avoid obstacles
- line follower
- robot with an arm

2. Current stage

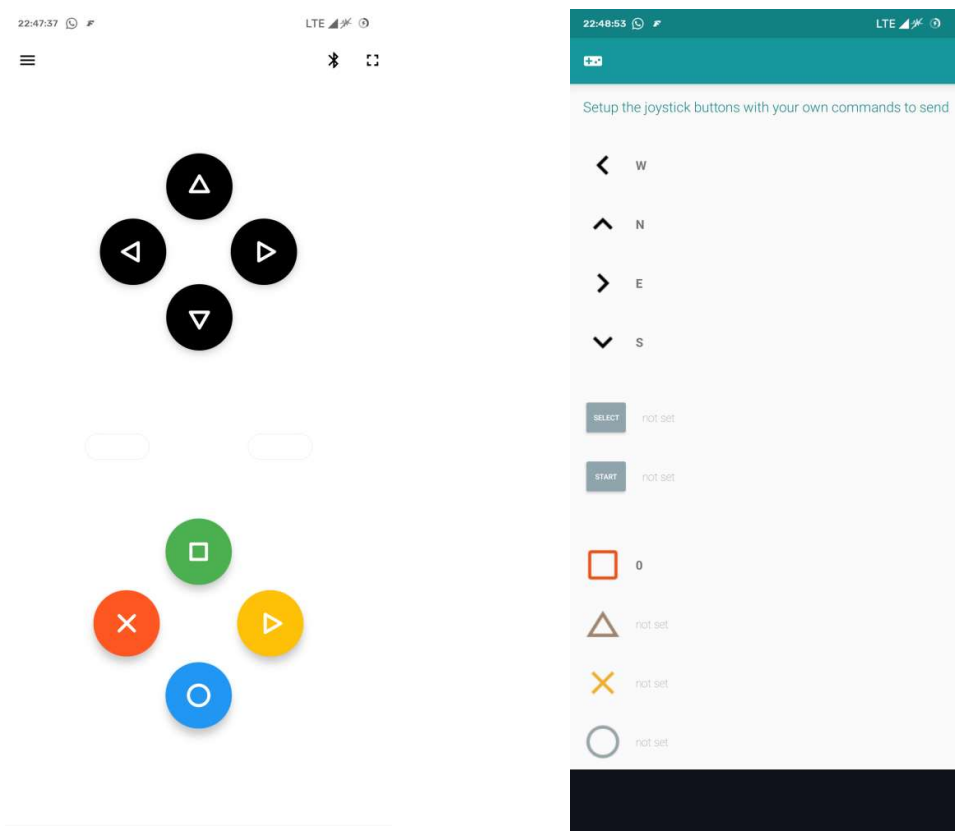
In order to use this project, you need a device capable of sending Bluetooth signals, using the RFCOMM protocol. Any computer with a Bluetooth adapter will suffice, but, for ease of use, it is recommended to use an Android phone, with one of the specialized apps for using with an Arduino. Here are some of the apps that can be used this way:

- <https://play.google.com/store/apps/details?id=uncia.robotics.joystick>
- <https://play.google.com/store/apps/details?id=com.giumig.apps.bluetoothserialmonitor>
- <https://play.google.com/store/apps/details?id=strikesoftware.bluetoothino>
- <https://play.google.com/store/apps/details?id=com.arbl.arduino bluetooth>

2.1. App configuration

All that needs to be done is to set four buttons in the program that send the **N**, **S**, **W** and **E** characters to the car for the directions front, rear, left and right respectively, and another button that sends the **0** char, which will stop the motors.

2.2. Configuration example



2.3. Hardware components

These are the main components on the robot:

- 4 motors with reducing gears and wheels: https://www.optimusdigital.ro/ro/motoare-altele/139-motor-cu-reductor-si-roata.html?search_query=motor&results=663
- Motor driver module L298N: https://www.optimusdigital.ro/ro/drivere-de-motoare-cu-perii/145-driver-de-motoare-dual-l298n.html?search_query=l298n&results=4
- Bluetooth module HC-05: https://www.optimusdigital.ro/ro/wireless-bluetooth/153-modul-bluetooth-master-slave-hc-05-cu-adaptor.html?search_query=hc05&results=2
- Batteries of type 18650: https://www.optimusdigital.ro/ro/acumulatori-li-ion/3152-acumulator-samsung-li-ion-3000-mah-18650-inr18650-30q.html?search_query=18650&results=59

- Development board Arduino Uno: https://www.optimusdigital.ro/ro/placi-avr/4561-placa-de-dezvoltare-compatibila-cu-arduino-uno-r3-atmega328p-atmega16u2-cablu-50-cm.html?search_query=uno&results=316

3. Source code

Since we used an Arduino-compatible development board, the chosen programming language is Arduino.

The source code uses simple concepts, like serial communication between the development board and the Bluetooth module, and the interface with the motor driver is made through four digital pins and is simplified through the usage of the **L298N** library: <https://github.com/AndreaLombardo/L298N>

```
// {{{ Libraries
#include <L298NX2.h> // library for the motor driver
// }}}

// {{{ Variables
// Declare pins
const int IN1 = 2, IN2 = 3, IN3 = 4, IN4 = 5;
L298NX2  motors(IN1, IN2, IN3, IN4); // Declare motors on the L298N
int  state; // Bluetooth state
// }}}

// {{{ Setup
void setup()
{
  // {{{ Set pin modes
  // For the L298N inputs
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  // }}}

  // {{{ Initialize serial communication
  Serial.begin(9600);
```

Phone controlled car

```
    //}}}
}
//}}}

// {{{ Loop
void loop()
{
    if(Serial.available() > 0) state = Serial.read(); // Save the state into a variable

    switch(state)
    {
        case 'N':
            motors.forward();          // move forward
            break;
        case 'S':
            motors.backward();         // move backward
            break;
        case 'E':
            motors.forwardA();         // rotate right
            motors.backwardB();
            break;
        case 'W':
            motors.backwardA();        // rotate left
            motors.forwardB();
            break;
        case '0':
            motors.stop();              // stop motors
            break;
    }
}
//}}}
```

6. Conclusion

In conclusion, this project is useful and extensible through its simplicity, from the simplicity of the source code to the simplicity of the physical model.