*ECE 1000 Final Report: RC Car*

Tennessee Technological University

Department of Electrical and computer engineering

Cookeville, TN, USA

[cwarner42@tntech.edu](mailto:cwarner42@tntech.edu), [dpkuhlman42@tntech.edu](mailto:dpkuhlman42@tntech.edu), [armarcelli42@tntech.edu](mailto:armarcelli42@tntech.edu)

***Abstract***

***The RC Car outlined in this report showcases the design and development of a custom-built RC car utilizing physical electronic components, 3D printed pieces, Bluetooth technology and microcontroller programming. The chassis and other components were designed using 3D printing, allowing us to get very precise and customizable pieces. The RC car is powered by a Raspberry Pi Pico that is paired with Bluetooth. Allowing us to use wireless control via a smartphone. The RC car integrates electronics such as Servo Motors, Motor Drivers and a L298N which allows us to control the speed and direction of the motor. This project demonstrates the integration of mechanical, electrical and software engineering to create a functional RC Car***

1. **Introduction**

RC cars for a long time have been a staple in the field of engineering and hobbyists. They offer opportunities to explore Bluetooth technology, mechanical engineering, and Electronics. This project aimed to design and manufacture a fully custom RC car that uses Bluetooth and other technologies such as 3D printing and Micro Python programming to create a functioning RC car.

1. ***Background***

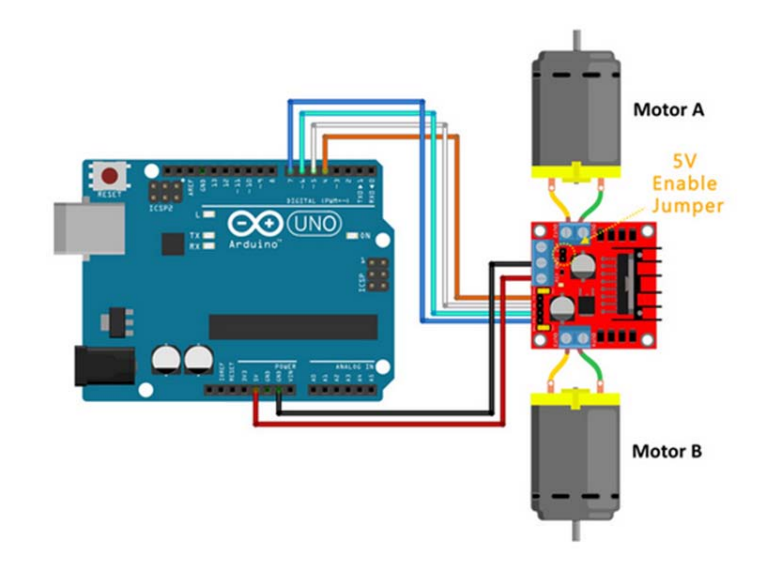
While developing our RC car, we utilized a variety of resources to ensure the success of the project. Datasheets for the electronic components provided critical technical specifications, while GitHub repositories offered example code that helped us understand and implement the functionality of these components. Additionally, YouTube tutorials served as valuable references for assembling and programming specific parts of the project.

1. ***Materials:***
2. Raspberry Pi Pico: serves as the central processing unit in the RC car. Its responsible for data processing, decision making, and controlling other components
3. DC Motor: This Motor is what is going to allow the RC car to accelerate and is going to provide power to the wheels.
4. Servo Motor: The servo motor is what is going to allow the RC car to turn it will rotate left or right to turn the wheels thus turning the RC car.
5. 9V battery: This battery is used to power all the electronics on the car.
6. L298N: Is our motor driver it will allow us to control the direction and speed of the DC motor it will also allow us to step down the voltage of the 9V battery
7. 5mm x 300mm Hex Rod: A hex rod cut to size is attached to the 3D printed wheels allowing the motor to drive them.
8. Ball Bearings: Ball bearings allow for the wheels to rotate smoothly and are more durable than being held in place just with 3D printed parts.

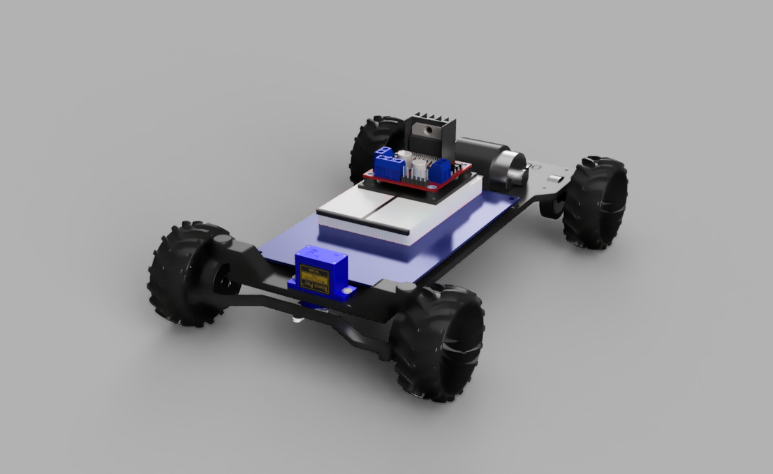
***Diagram:***

The diagram below shows an Arduino instead of a Raspberry Pi because we could not find a Raspberry pi in the software we used. The circuit still acts the exact same as our Raspberry Pi. In the image you can see that we are using a L298N Motor Driver which is what the Motors are connected to. The L298N allows us to Run the motors in different directions.

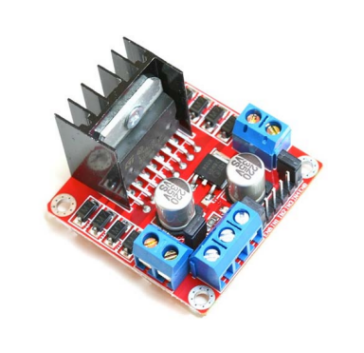
**Figure 1: Tinker cad circuit**



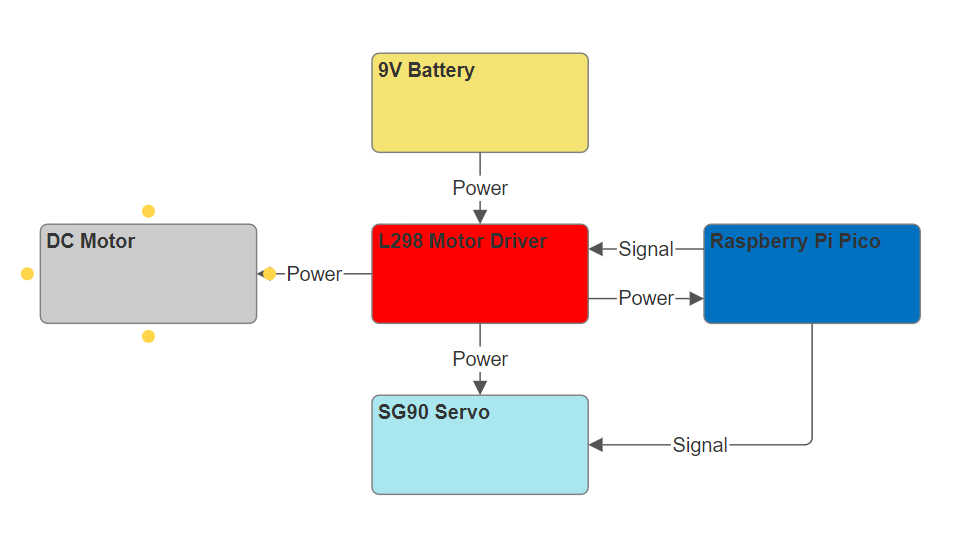
**Figure 2: 3d design of our RC car. We designed the car in fusion 360**



**Figure 3: Picture of the L298N Motor Driver**



**Figure 4: Here is a state diagram of our RC Car. This diagram shows where power is being sent and where signals are being sent in the RC Car for it to run.**



***Functionality:***

All the hardware components are connected by GPIO pins (General Purpose Input / Output) on the Raspberry Pi Pico. The Raspberry Pi Pico allows for seamless interactions between the electronic components and the Raspberry Pi Pico.

The L298N Motor Driver is the component that controls the motor in the RC Car. The L289N will either tell the motor to go forward or backwards depending on the command sent by the Pi Pico. The L298N Motor Driver is also what we use to step down the voltage. Sense we are using a 9V battery and we only need 5V for the Raspberry Pi Pico so this component will give us 5V exactly without sending too much voltage to the Raspberry Pie Pico.

The Servo Motor at the front of the car is what we use to steer the car. There are 3 possible commands that can be sent to the servo. Left which will turn the front wheels to the left. Right, which will turn the front wheels to the right and straight which will face the wheels straight forward.

The entire car is controlled through a smartphone console, which communicates with the RC car via Bluetooth. After connecting to the console, users can send specific commands to control the car's movements. For instance, sending the left command will turn the car to the left. The car responds to every command sent, enabling precise control over its actions.

1. ***Discussion And Results***

The RC car project yielded successful results, as the RC car was able to effectively drive with the commands it received through Bluetooth. The RC car demonstrated reliable performance while driving forward, reversing, and turning based on the user’s input.

The total cost for the project was $28 which while economical for an RC car project could have been brought down further by supplementing some of the metal parts for 3D printed ones.

However, with additional time, there are several areas in the RC car for improvement. One thing we could work on would be the addition of a braking system because the current model does not support breaking while it's running.

Overall, the project achieved its goal and the results we have showed highlight the effectiveness of using 3D printing, electronics, and Bluetooth technology to create a operational RC car.

**Conclusion**

**In conclusion our RC car successfully integrated 3D printing, Bluetooth, and electronics to create a working RC car. While the RC car did meet our goal of being able to drive additional features such as breaking and other system could have made the RC car better. Overall, the project helped us gain insight into mechanical, electrical and computer engineering.**

**REFERENCES**

**[1] HandsOn Tech – open-source electronics platform. (n.d.). https://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf**

**[2] Ramjipatel. (n.d.). GitHub - ramjipatel041/Micropython-Driver-for-L298N-Motor-Driver: Control Motors Using L298N Motor shield and MicroPython. GitHub. https://github.com/ramjipatel041/Micropython-Driver-for-L298N-Motor-Driver**

**[3] Peppe8o. (2024, April 13). SG90 Servo Motor with Raspberry PI Pico and MicroPython. Peppe8o. https://peppe8o.com/sg90-servo-motor-with-raspberry-pi-pico-and-micropython/**

**[4] Thingiverse.com. (n.d.). Robomow Wheel left - SPP7021A by Mr\_Clean. Thingiverse. https://www.thingiverse.com/thing:3786498**

**[5] Free CAD designs, files & 3D models | The GrabCAD Community Library. (n.d.). https://grabcad.com/library/sg90-micro-servo-9g-tower-pro-1**

**[6] Free CAD designs, files & 3D models | The GrabCAD Community Library. (n.d.-b). https://grabcad.com/library/l298n-h-bridge-motor-driver-1**

**[7] FriendlyWire - Discover the joy of understanding electronics. (n.d.). https://www.friendlywire.com/projects/ne555-servo-safe/SG90-datasheet.pdf**

**[8] Marcellino, A. (2024). Fusion. (n.d.). https://a360.co/4gidKkq**