*ECE 1000 Final Report: Wirelessly Controlled RC Car*

Brendan Thomas, Tuan Kiet Le

Tennessee Technological University

Department of Electrical and Computer Engineering

Cookeville, TN, USA

[bkthomas43@tntech.edu](mailto:bkthomas43@tntech.edu) , [tle43@tntech.edu](mailto:tle43@tntech.edu)

|  |  |
| --- | --- |
| **The Wirelessly Controlled RC Car is a miniature car, powered by a nine-volt battery, that is controlled either manually by a remote, or automatically by sonar sensors. This feat was accomplished through the usage of two ESP-32 microcontrollers programmed with Arduino C++. This report explains the design and functionality of the Wirelessly Controlled RC Car.**   1. INTRODUCTION   As my (Brendan) first time using anything related to transforming code to in-person functionalities, a Wirelessly Controlled RC Car proved to be a wonderful starting point for learning how to program with Arduino. This project showed how to transfer information between two microcontrollers along with all of the steps needed in order to do so.   1. BACKGROUND   Unfortunately, I was not involved in the discussions about the materials used for the project as we had formed our group after Kiet had formed his list of materials and ordered the parts, but after the parts had been delivered, we began developing the functionalities of the car. We had come to the conclusion that the car would have two functions; manual controls so that the user could have complete control of the car, and automatic controls that would cause the car to drive forward and avoid any obstacles, preventing the potential for crashes.   1. PROJECT DESCRIPTION AND FORMULATION   **Materials:**   1. Two ESP-32 microcontrollers: Allows Arduino code information to be wirelessly transferred between each other through a WiFi-like system involving both microcontrollers’ MAC addresses.   When we first had brainstormed how the functionality of the automatic mode would be, we did not consider the potential power constraints having two DC motors, an ESP-32 microcontroller, and three sensors would bring, so after developing the code, we were stuck with a car that would only partially function. This, however, was solved by adding small power supplies commonly used in commercial RC cars.   1. DISCUSSION AND RESULTS   The Wirelessly Controlled RC Car proved to be mostly successful. The car’s manual function would work in almost all functions. A small concern that we have involves the amount of turning the car does when told to do so. When only given a small amount of input stating to either turn right or left, the car would turn almost a full 90 degrees. We believe that this is due to the innate latency between the ESP-32s. It would seem that the information from the transmitter does take time, although very miniscule, to be received by the receiver. This was later enforced by the results of the automatic function, as the automatic function will stop and turn the RC car immediately when met with any obstruction.  **Responsibilities:**  Brendan Thomas was responsible for developing the Arduino code to control the ESP-32s. This includes receiving MAC addresses of both microcontrollers, the code needed to transmit information from the controller’s ESP-32, and the code needed to not only receive information onto the RC Car’s ESP-32, but also turn the information into actions made by the DC motors.  Tuan Kiet Le was responsible for the design and wiring of both the controller and RC Car. This includes developing the design of the breadboard wiring of the controller (developing a design where we would have one joystick to control speed, one joystick to control direction, and a switch that controls whether the RC car is in automatic or manual mode) and the RC car (developing a design that allowed the three sensors to be read into the ESP-32, allowed the ESP-32 to send information to the L298N motor driver, and offered enough power for the whole RC car to be able to function wirelessly).   1. CONCLUSION   The ESP-32 controller proved to be an excellent tool for transferring information from one microcontroller to another without the use of a wire. This piece of technology could be used for a wide variety of projects due to how versatile it is. | 1. L298N Motor Driver: Supplies power to the DC motors if told to do so through the connection with the car’s ESP-32 2. Two DC motors with miniature wheels 3. Three Sensors: Uses sonar-like reading in order to sense the distance between the sensor and the object in front of the sensor. 4. User Interface: Allows the user to control the car manually with two joysticks. Also allows the user to swap between two control functions; manual and automatic.   **Functionality:**  The Wirelessly Controlled RC Car first had to be able to receive information from the controller/transmitter. This was possible due to the ESP-32's innate ability to transmit and receive information after receiving both microcontroller’s MAC addresses. We were not able to develop the code to receive the MAC addresses of each microcontroller as this was our first project using the microcontrollers, so we referred to YouTube to find the MAC addresses.  After we were able to transmit information between the ESP-32s, we began developing the Arduino code for the manual control function of the car. This proved to be relatively simple, as the code we had developed allowed us to selectively tell each DC motor to either move forward or backward, depending on the direction the user inputs.  We then moved toward working on the automatic function of the car. The code would do two things primarily. First, the sensors would read in the distance information and send the information to the ESP-32. Second, the ESP-32 would receive the information, run the information through a branching statement that would check whether the car would need to either keep moving forward, turn left, or turn right, and send the result of the branching statement through the L298N motor driver to power the DC motors to do their needed task.  Overall, I had learned an extraordinary amount from both the opportunity of this project, and the abundance of knowledge from my project partner, Kiet. This is only the starting point of the potential for greatness that is an Electrical Engineering student at Tennessee Tech. |