RC Rover

Submitted by:

Submitted to: OACETT
10 Four Seasons Place, Suite 404
Toronto, ON M9B 6H7

Discipline: Computer Engineering Technology

Date Submitted: TBD

Declaration of Joint Authorship

I, Mykal Bailey, and my partner David Uche, confirm that this work submitted for assessment is of our own and is expressed in our own words. Any uses made within it of the works of any other author, in any form (ideas, equations, figures, texts, tables, programs), are properly acknowledged at the point of use. A list of the references used is included. For the members work breakdown refer to 2.1 Requirements.

Signed: Mykal Bailey; David Uche;

Date: TBD

Abstract

The world is moving into the direction of robotic automation. There is still much to discover and learn. Companies like Tesla and Google have already started prototyping and producing self driving vehicles. The same can be said for the exploration and gathering of data on this planet earth. A manned venture into an unknown world can be dangerous and costly. Using robots to do the exploring while being able to observe and reporting is not only more cost efficient, but lowers the risks of loss for companies. Designing and implementing a robot to handle unpredictable conditions has challenges from beginning to end. NASA alone has spent billions of dollars building rovers. New designs are necessary to bring costs down whilst keeping or bettering the level of performance. For automation, or unmanned vehicles, to operate they must be able to learn, and make decisions that'll be beneficial in the long run.

Table of Contents

RC Rover

Declaration of Authorship **2**

Abstract **3**

Table of Contents **4**

List of Illustrations **5**

1.0 Introduction **6**

2.0 Planning (Method) **7**

2.1 Requirements **7**

2.2 Design Plan **7**

2.3 Implementation **8**

3.0 Design **10**

Figure 3.1 – Wiring Diagram - Two Wheel Drive Robot **10**

Figure 3.2 – Wiring Diagram – Swivel Camera **11**

4.0 Process **12**

5.0 Conclusions **13**

6.0 Recommendations **14**

7.0 Bibliography **15**

8.0 Appendices **16**

List of Illustrations

1.0 Introduction

This report entails the planning, design, process, and conclusion of combining two hardware devices to work interdependently with an android software application. This project will be conducted by two group members merging individual hardware projects to create a super bot. A swivel enabled camera which is to be mounted on top of a two-wheel drive robot allowing for mobility, surveillance, and remote control when paired with a Raspberry Pi. The design and creation of the individual hardware will also be covered. The two devices coupled will create what we will call an RC Rover. The robot will be mobile and controlled by an android software application either from a smartphone or a tablet. The RC Rover will eliminate the need to be physically present to maneuver and observe environments.

The ultimate goal of this project is to have the Rover be automated as well. To achieve this, we will be implementing open source software that will allow the Rover to learn. A trainable rover will allow it to learn environments, recognize objects, and be able to repeat or follow specific paths. DonkeyCar is the software that will be implemented with the project, which will provide simulated environments to train the rover. With the attained knowledge, the rover will be able to move autonomously and make quick decisions that will be beneficial in the long run.

2.0 Planning (Method)

2.1 Requirements

As a group, we have been given 14 weeks to plan, design, implement and create a report for our RC Rover project. We have identified tasks necessary for completion of the venture. These tasks include combining the two-wheel drive robot with a swivel camera into one RC Rover. The final outcome of the hardware device is meant to be remotely operated. To accomplish this, an android software application will need to be developed to control both hardware devices simultaneously. It also essential to create and implement a database to store files. Finally, there needs to be an observer and reporter.

For efficiency, the tasks required for successful completion of the project are split between two group members. Both members will be involved in the planning and design of the software application and final hardware device. Both members will also be responsible for supplying individual hardware components. Member A will be accountable for the Swivel Camera component, software design, database design, and report writing. Member B will be responsible for the Two-Wheel robot component, hardware design, android software development, and report editing. As a group, ideas and suggestions will be shared regularly for potential improvements of the final design.

2.2 Design Plan

The RC Rover is to have a Raspberry Pi 3 (RPi) used as a central control system for the hardware components. The RPi will be programmed to control the Two-Wheel bot and Swivel

Camera. The Rover is intended to be completely remote. The Wi-Fi-enabled RPi will need to be connected to a network in order to receive commands and provide live video feed. The RC rover will also need a portable power supply for the RPi and other hardware components. The base of the RC Rover is provided by the Two-Wheel bot. It has two wheels and motors bolted to a chassis to provide mobility. The swivel camera is basically a Pi Camera attached to swivel mount. The swivel movement is controlled by two servos which allows for two-directional movement. This design lets the camera rotate up to one-hundred-eighty degrees vertically and horizontally.

The RC rover is to be controlled by an android software application from a smartphone that is connected to the same network. The application will have an easy to use user interface. It will provide a live video feed of the RC rovers view directly from the software application. It will also include a controls to individually control camera view and movement of the RC rover. It will also include functionality to record and take snapshots from the RC rover which will be sent and stored to the database. Data files saved to the cloud will be available for viewing and editing directly from the application.

2.3 Implementation

The purpose of the RC Rover project is to have the final hardware portion work alongside the software application and the cloud database. Firstly, there needs to be hardware to work with. The RC Rover consists of a Raspberry Pi, the Two-Wheel Bot, Swivel Camera, and a portable power supply. The Two-Wheel Bot provide chassis, and wheels with motors to allow mobility. The two motors together require 12V of power for

operation. This power is provided by four triple-A batteries mounted to the chassis. A L298N H-Bridge Dual-Stepper Motor driver is connected to a portable power supply for power, the motors and the Raspberry Pi GPIO pins. The Swivel Camera is used for the purpose observing and recording. It has two important mechanisms, the swivel mount and it's two servos. The Swivel Camera is to be mounted on the Two-Wheel bot's chassis and will be connected directly to the Raspberry Pi's GPIO pins. Both hardware components will be connected to and controlled by a single Raspberry Pi. The software application will need to identify and connect to The RC Rover's Raspberry Pi through a network connection. Finally, to put it all together, both the application and Raspberry Pi have to be able to connect to the Database, which in this case will be Firebase Database by Google.

3.0 Design

Wiring Diagram - Two Wheel Drive Robot

Figure 3.1:

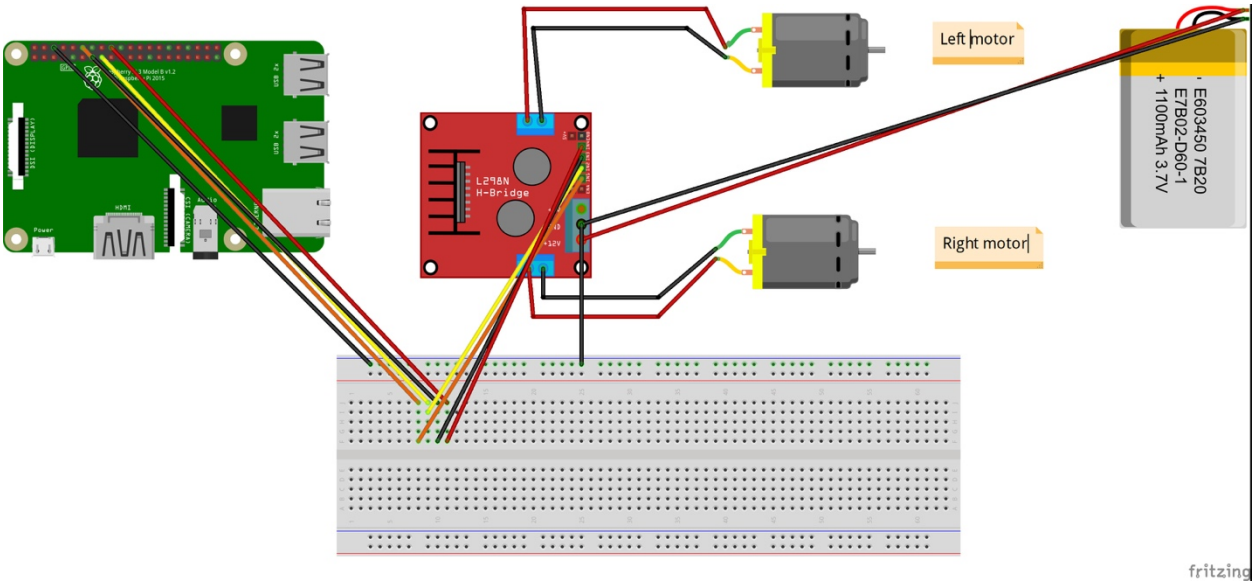
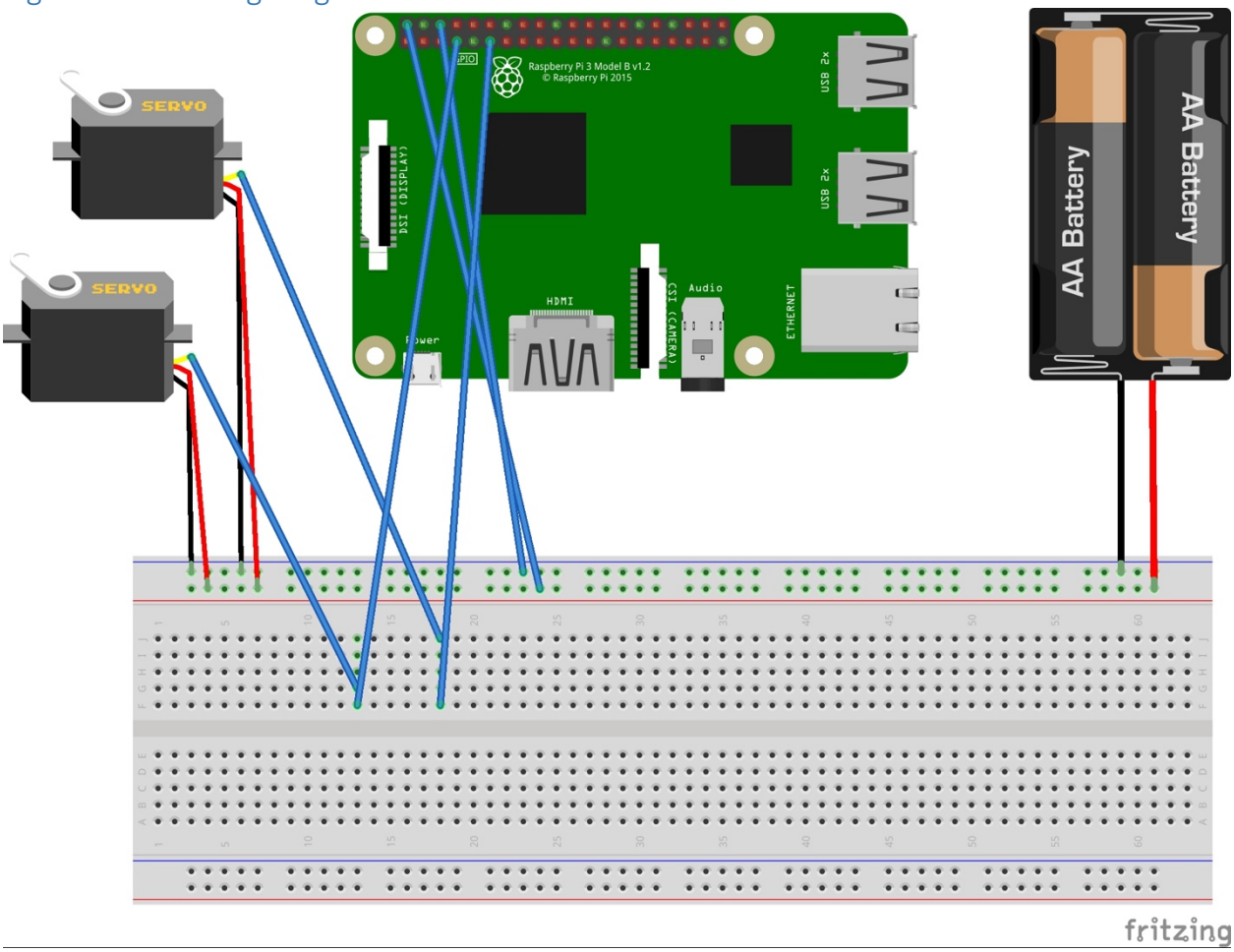


Figure 3.2 – Wiring Diagram – Swivel Camera



4.0 Process

5.0 Conclusions

6.0 Recommendations

7.0 Bibliography

"Remote Control Robots." *Robot College*, www.robots-and-androids.com/remote-control-robots.html.

Svitak, A. (2019). *Cost of NASA's Next Mars Rover Hits Nearly $2.5 Billion*. [online] Space.com. Available at: https://www.space.com/10762-nasa-mars-rover-overbudget.html [Accessed 6 Feb. 2019].

AWS DeepRacer - the fastest way to get rolling with machine learning. (n.d.). Retrieved February 7, 219, from https://aws.amazon.com/deepracer/

8.0 Appendices