Design and Implementation of a Remote-Controlled Racing Car using Audio Wireless Link

Student Number: 2118091

School of Electrical and Information Engineering, University of the Witwatersrand, Johannesburg 2050, South Africa

***Abstract*—The Electrical Engineering Design project is to create a remote-controlled racing car that is controlled using an audio wireless link. The system is equipped with an audio transmitter (remote) and sound receiver module that will enable the user to control the car. The project involves the design and building of the control system of the car, including the programming of the microcontroller and the electronics that will control the car's movement. The car is designed to reach high speeds while maintaining stability and control. The remote is used to transmit dual-tone multi-frequency signals, which are received by a microphone mounted on the car. The commands are processed by the microcontroller (Arduino Uno) using Goertzel's Algorithm, which then controls the motors using Pulse Width Modulation to execute the desired movements. Signalling techniques used to control the car are discussed. This project allowed students to consolidate concepts from different modules throughout the degree and develop communication skills.**

Keywords—audio-control; Arduino; car; Goertzel; modulation; motors; wireless

I. Introduction

In this project, a remote-controlled racing car is developed that is controlled with audible signals. The car is driven by two independent DC motors. The project includes the design and development of a system that controls the car's movement.

Dual Tone Multi-frequency (DTMF) signalling is used to control the car. Each DTMF signal corresponds to a certain directional command on the car. The signal is received by a sound module on the car and interpreted by an Arduino Uno. The motors are then controlled by setting a Pulse Width Modulation (PWM) from the Arduino and then sent to a motor driver integrated circuit (IC) controlling the motors. Two methods are implemented for the remote that would produce the DTMF signals. An Arduino-controlled remote control and a mobile application are discussed as ways to control the remote-controlled car.

Group dynamics and communications are also discussed in the report, along with improvements that could be made to the present design.

II. Background

The project brief for the design task specifies that the car should be controlled by audio signals in the range of 20 Hz to 20 kHz. The car should be able to receive commands wirelessly from five to ten meters away. The car should be able to navigate a specified racetrack.

A. History of DMTF

Dual Tone Multi Frequency signaling has been used since 1963 and was originally used to transfer information across phone lines without any interference from voice [1].

DTMF works by defining a certain frequency for columns (high frequencies) and rows (low frequencies) on a keypad. When a button is pressed a superposition of the column and row frequencies are played. Playing two frequencies in different bands makes it resilient against voice interference as voice cannot produce a high and low frequency at the same time [2].

B. Goertzel's Algorithm

The Goertzel Algorithm is an algorithm used to decode DTMF signals. This algorithm evaluates a small portion of the discrete Fourier transform (DFT) to determine the real and imaginary components of the frequency of the signal identified. This method of evaluation is very fast and resource-efficient [3]. A library is available for the Arduino integrated development environment that allows the use of the algorithm in Arduino code.

III. Design Specifications

The implemented design uses DTMF signalling to control the remote-controlled car. The DTMF tones are produced by either a remote that is controlled by an Arduino and hardware for testing [4], or a mobile app for demonstration. The audio signal is then received by a KY-038 sound sensor [5]. The audio signal received is then processed by an Arduino using the Goertzel Algorithm [6]. The processed signal then gives a certain frequency combination which will be coded to a directional command for the car. The Arduino sets a PWM for each motor to control direction. The motors are driven by a TB6612FNG motor driver. A circuit diagram of the implementation along with a block diagram is found in Appendix A and C respectively.

A. Remote Design

Two remote options are considered in the design, namely a remote built by the group and a mobile phone app. The remote design is an Arduino and hardware that produced DTMF tones [4], without any integrated circuitry. This option produces a specific DTMF tone when a button is pressed. Buttons are structured in a "plus sign" formation with forward, stop and reverse buttons in a line and then left and right on either side.

This option outputs the DTMF tone to a speaker. This option produces accurate DTMF tones as detected by the Goertzel Algorithm but is not loud enough to be used for demonstration purposes. A schematic circuit diagram for this option is found in Appendix B.

The second option is a remote based on a phone keypad. An app is used that implements the DTMF tones of a dialling pad. This application is used for demonstration purposes as it increases the range the car can be used, by connecting the phone to a Bluetooth® speaker (JBL Charge 3). The remote is used similarly to the built remote, with button ‘2’ being used for forward, buttons ‘4’ and ‘6’ being used for left and right respectively, ‘8’ being used for stopping, and ‘0’ being used for reverse.

B. *Audio receiver and processing*

Audio signals are received by a KY-038 sound sensor that is made up of a capacitance-sensitive electret microphone and an amplification circuit [5]. The audio signal is read from the analogue pin of the sound sensor to the Arduino.

The audio signal is received by the A0 analogue pin on the Arduino. The signal is then processed by the Goertzel algorithm on the Arduino which identifies which frequencies have been received. The frequencies are then mapped on a matrix and the column and row they belong to are found to find which number is being pressed. This defines the command for the motors.

An error-correction ‘algorithm’ is implemented when the command to the car is given. Signals need to be processed correctly twice in a row for the command to be sent to the motors.

C. *Motor Drivers*

A high-efficiency motor driver (TB6612FNG) with dedicated PWM pins is used in the design. The motors are driven with the Arduino’s 5 V output. The directions of the car are manipulated by changing the PWM duty cycle to each motor. Forward is specified as an 85% duty cycle to each motor (discussed in *Motor Driver Evaluation*), left is specified as a 65% duty cycle to the left wheel and 85% duty cycle to the right wheel, right is specified as a 65% duty cycle to the right wheel and 85% duty cycle to the left wheel. PWM is controlled through the Arduino and is output from a specific pin to the motor driver.

IV. DESIGN EVALUTAION

At the beginning of the project, the main design proposed was a design that implemented Audio Frequency Shift Keying (AFSK) as a modulation technique. This design proved too technical to implement in the time frame provided, additionally, the design also encountered problems with memory limits, processing speeds and sampling speeds on the Arduino. A suggestion was made to implement the design on a Raspberry Pi but after a cost-benefit analysis, and time evaluation this was decided against. The final solution implemented was chosen as it was simpler.

A. *SWAT Analysis of the current design*

The current design is a simpler solution but is easier to implement and still efficient as the car completed the track in a competitive time. The design provides some immunity against other groups and environmental noise, although errors in the commands received are mitigated yet still evident. The advantages of this design are the abundance of resources and the responsiveness of the car to given commands. The trade-offs for this design were the decreased immunity to noise and signals from other groups.

B. *Remote Evaluation*

The remote used for the demonstration works efficiently. The car can be controlled from five meters away when the speaker is at full volume. This design could be improved by developing a unique application for the group with a better user interface, instead of the application used which used a dial pad. A developed application would have specified buttons for directional controls.

The remote designed and used for testing produces accurate DTMF tones but the speaker used in the design is not able to produce the tones loud enough for the commands to be received over a distance. The group attempted to add an auxiliary cable to the output of the remote circuitry, but this was unsuccessful due to incompatible connections with the speaker.

C. *Audio Processing Evaluation*

The sound module found has a built-in amplifier which assists with receiving the audio signals. This alleviates the need to build an amplification circuit.

Audio processing through the Goertzel algorithm is beneficial over the Fast Fourier Transform algorithm as it is faster at determining the frequencies played. Error is still identified using the Goertzel algorithm and is corrected using the error-correction ‘algorithm’ as discussed in Section III, C.

Error is also encountered across columns of the dial pad. The high (column) frequency is identified, and the incorrect (low) row frequency is identified. This often causes the number five to be interpreted as a two or eight. This is corrected by carefully choosing which buttons lead to the least error.

The audio processing has a fast response time. During testing, it is seen that the car responds in under 0,4s from 1,5m for each command given at full volume on the Bluetooth speaker.

D. *Motor Driver Evaluation*

The motor driver used is chosen as it provides multiple benefits over the L298N h-bridge that is available through the University. The TB6612FNG is smaller and lighter than the L298N. It is much more efficient at 90% - 95% efficiency compared to 40-70% efficiency [7]. The TB6612FNG also has a smaller voltage drop compared to the L298N. The high efficiency of the motor driver helps the car achieve a fast time of 1:07 min around the testing track. The car travels 3 m in 7,76 s giving it a straight-line speed of 0,387 m/s.

To increase the accuracy of the steering and to make the car easier to use, the duty cycle of the forward direction is decreased from 100% to 85%. For a left and right turn, the duty cycle is set to 65% as a 50% duty cycle has a turning circle that is too small for the track requirements.

V. Project Management

A. Group Work

The group split tasks evenly throughout the duration of the project. The three main sections that the work was split into are remote development, audio processing and motor drivers. Each group member took one of these sections and then assisted in other sections when needed or when their section was completed. This method of work was efficient and allowed each member to research certain parts of their subsystem effectively and then share the knowledge attained.

B. Time Management

Throughout the project, a lot of time was spent trying to implement the AFSK modulation solution. Approximately a week and a half were spent implementing the AFSK on a remote and three weeks were spent attempting demodulation for the solution on various methods such as MATLAB, Simulink, and C code. Roughly a week was spent on the DTMF solution. This solution was simpler and took less time to implement.

If time was not a constraint the AFSK solution could have been implemented fully. Due to the demodulation process of the AFSK solution taking longer than expected, the group was forced into implementing a solution that would be achievable before the deadline.

This could be avoided by investigating the solutions more thoroughly before implementing them and getting an understanding of how long they would take to implement.

VI. Discussion

Throughout the evaluation of the remote-controlled car above it is evident that the car performs as required. The car does veer to one side when going in a straight line. This was corrected with fast command response and a turn in the opposite direction.

Improvements to this design could be made to improve the modulation technique. This would increase the noise immunity of the solution and allow it to be usable in noisy environments where other frequencies are being played. Another improvement could be to use a microphone with a higher sensitivity. This would allow the reduction of the volume of the Bluetooth speaker while still maintaining the response time of the car.

VII. Conclusion

This project shows that an audio remote-controlled car can be implemented using dual-tone multi-frequency signalling. The KY-038 sound module receives audio signals efficiently which are processed accurately enough through an Arduino Uno for useful commands to be interpreted. The TB6612FNG motor driver proves to be an efficient motor driver to control a motor control car with high efficiency and minimal voltage drop. Overall, the implemented design was effective and was able to complete a loop around the given racetrack timeously. The car can reach a straight-line speed of 0,387 m/s and completes the given track in 1:07 min. An evaluation of the strengths and weaknesses of the design and the group's project management strategies was performed.

References

[1] Call Centre Helper, "The History of DTMF in Contact Centres," Call Centre Helper.com, 1 June 2018. [Online]. Available: https://www.callcentrehelper.com/history-dtmf-contact-centres-128079.htm. [Accessed 18 May 2023].

[2] M. D'Sa-Wilson, "What is DTMF? How Does Dual-Tone Multi-Frequency Work?," United World Telecom, 18 November 2022. [Online]. Available: https://www.unitedworldtelecom.com/blog/what-is-dtmf-dual-tone-multi-frequency/ . [Accessed 18 May 2023].

[3] S. Westerhold, "Simple Goertzel algorithm for (multi-)tone decoding," Github, 11 November 2022. [Online]. Available: https://github.com/AI5GW/Goertzel. [Accessed 2023 May 22].

[4] D. Jayakody, "Generate DTMF Tones Only Using Arduino," hackster.io, 9 May 2019. [Online]. Available: https://www.hackster.io/dilshan-jayakody/generate-dtmf-tones-only-using-arduino-7d3713. [Accessed 2023 May 8].

[5] Microcontrollerslab.com, "KY-038 Microphone Sound Sensor Module with Arduino," Microcontrollerslab, 2023. [Online]. Available: https://microcontrollerslab.com/ky-038-microphone-sound-sensor-module-arduino-tutorial/. [Accessed 16 April 2023].

[6] M. Shoaib, "DTMF Decoder Using Only Arduino," Arduino Project Hub, 13 December 2018. [Online]. Available: https://projecthub.arduino.cc/MM_Shoaib/3280abb6-ff7e-41d0-b254-d440bdc41b45. [Accessed 2023 May 8].

[7] J. Cook, "TB6612FNG Motor Driver: Better Than the L298N?," hackster.io, 2020. [Online]. Available: https://www.hackster.io/news/tb6612fng-motor-driver-better-than-the-l298n-7499a7574e63#:~:text=The%20TB6612FNG%20is%20smaller%20than,mode%20that%20the%20L'%20lacks.. [Accessed 28 April 2023].

Appendix A: Circuit Diagram showing the workings of the circuitry of the car.

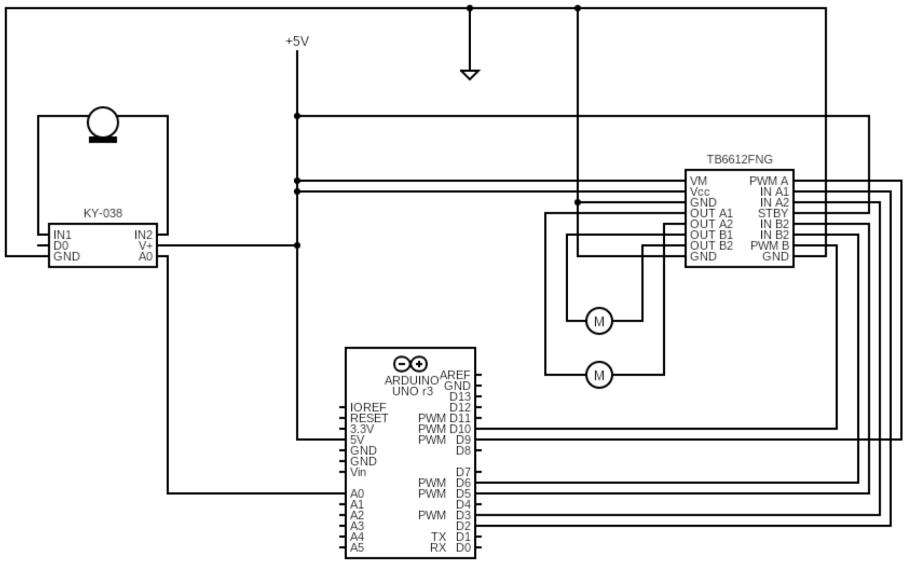

Fig. 1. Figure showing the circuitry of the workings of the car.

Appendix B: Circuit diagram of the remote used for testing.

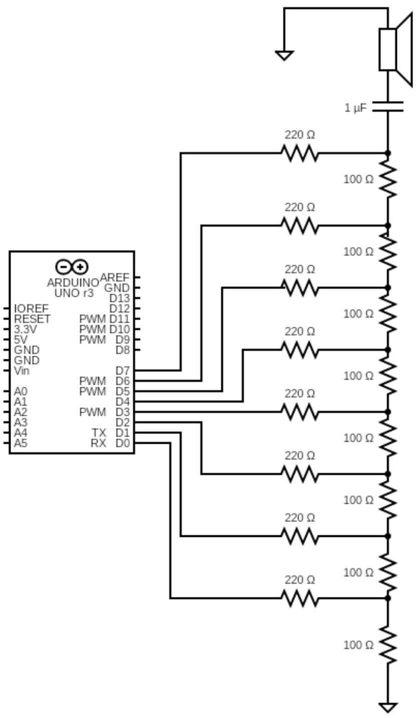

Fig. 2. Figure showing the circuitry of remote design.

Appendix C: Block diagram of the implemented solution.

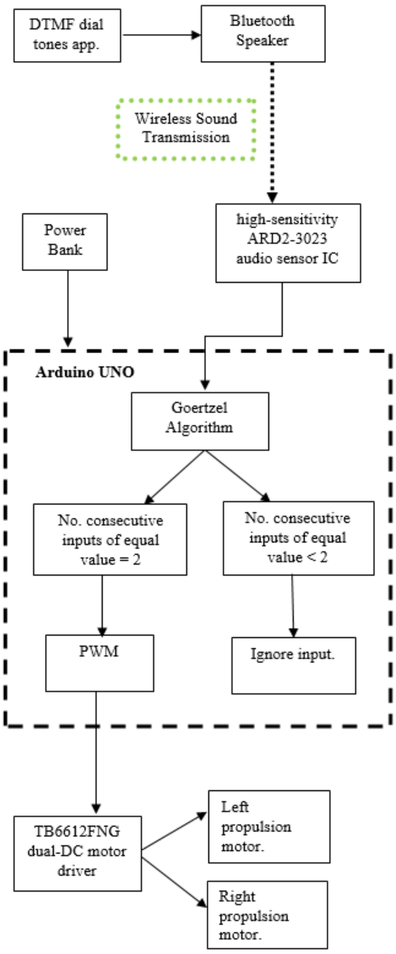

Fig. 3. A block diagram of the overview of the implemented solution.