EDUCATIONAL STICK SHIFT ASSISTANT

By

Aadhar Patel (alpatel2)

Ian Kidder (ikidder2)

Maulin Patel (mpate222)

Design Document Check for ECE 445, Senior Design, Spring 2021

TA: Alex Sirakides

2 March 2021

Project No. 4

Contents

1. Introduction

1.1 Objective

Knowing how to drive a manual transmission vehicle is a useful skill for anyone traveling abroad, buying a new car, or looking to save some money. However, it can be hard to learn how to drive a manual without in-person instruction.

The Educational Stick Shift Assistant will provide instructions to anyone who wants to learn to drive a manual car independently. Our solution is a system that reads the car's speed and engine RPMs to output audio instructions for the driver to follow. Our goal is to have the driver become familiar with shifting gears, giving them the knowledge and confidence to drive a manual car.

1.2 Background

COVID-19 has caused mass layoffs, closed down businesses, and halted the economy. As a result, global production of cars has decreased [1]. In addition, due to social distancing and safety guidelines, many people want to avoid using public transportation. One safe and economical alternative is to buy used cars. However, the price of used cars has tremendously increased [2] as a result of this pandemic. An affordable option that remains are cars with manual transmissions which are on average $1000 cheaper [3] when compared to their automatic counterparts. However, manual transmission vehicles are inaccessible to people who do not know how to drive them. Knowing how to drive a manual car also equips the person with a vital travel skill as nearly 80% of cars on the road in Europe have manual transmissions as of 2020 [4]. In conclusion, knowing how to drive a manual car is a beneficial skill for almost anyone.

1.3 Visual Aid

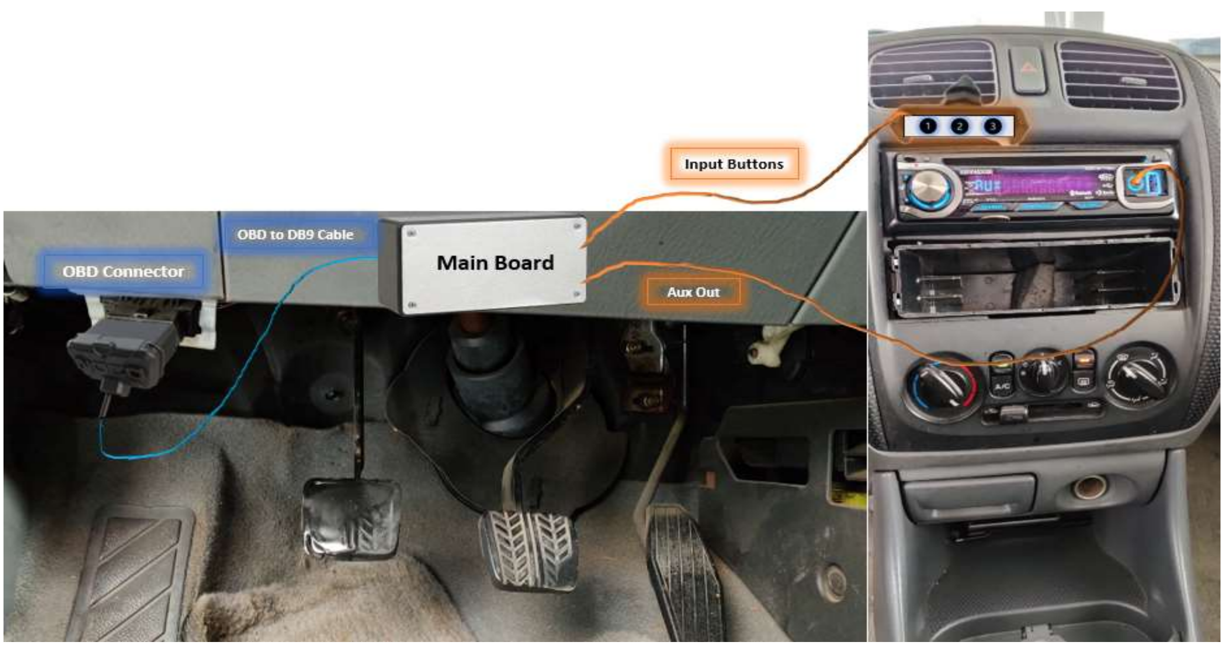

Figure 1: Visual Aid

In figure 1, a pictorial representation of our project is shown. We have one main board that will sit near the OBD Port where it will be held in place by Velcro strips. Another small enclosure will house our input buttons which will be mounted using an air vent clip. Our auxiliary cable will feed directly into the vehicle's auxiliary port.

1.4 High-Level Requirements

1. Instructs the driver to shift gears at 3000 RPMs with an audio cue.
2. Instruct to be in 1st gear between speeds of 0 mph - 15 mph and 2nd gear for speeds of 15 mph - 30 mph and gears above 2nd gear will not be used.
3. Must have audio cues for stall (less than 700 RPMs) and redline warnings (greater than 4500 RPMs).

2 Design

2.1 Block Diagram

Figure 2 represents the block diagram of the Educational Stick Shift Assistant and provides a visual representation of what components are used to achieve the high-level requirements set forth in chapter 1.4. The ECU Interface Module is responsible for retrieving vehicle data such as RPMs and vehicle speed when requested by the Control Unit. The Control Unit is then responsible for analyzing the data received from the ECU Interface Module as well as input from the user through the Button Breakout Board and sending correct control commands to the Audio Module. The Audio Module is responsible for outputting the desired audio cue stored on the SD Card through the AUX Connector for playback by Vehicle Speakers. All while the Power Module is responsible for regulating and providing power to all the sub-modules/components in the design. The Program & Debug Module is not user accessible and enables easy programming and monitoring of the microcontroller during prototyping and final build stages.

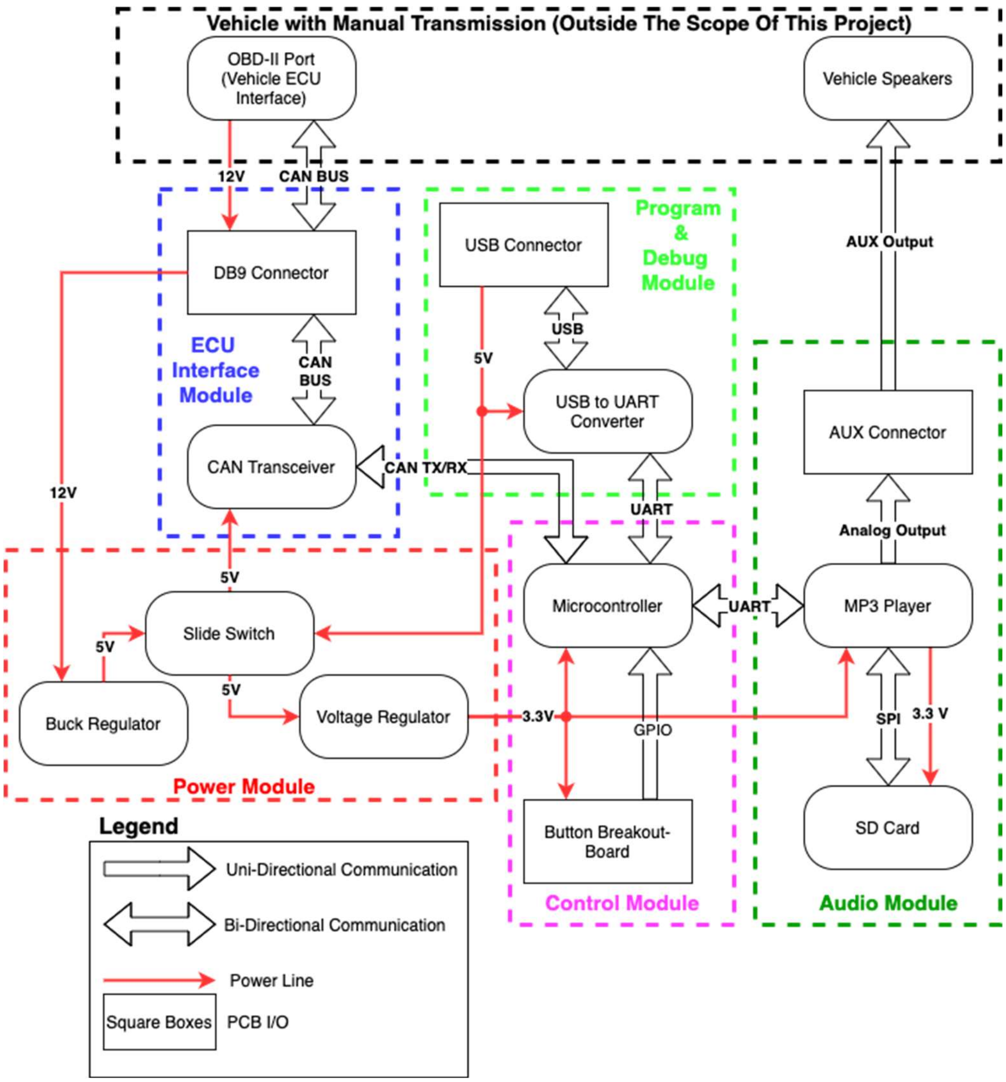

Figure 2: Block Diagram of Educational Stick Shift Assistant

2.2 Power Module

The Power Module is responsible for regulating and providing power to the rest of the components in the design via 5 V and 3.3 V rails. The power module is designed such that it can source power from two sources, The USB connector in the Program & Debug Module as well as the DB9 Connector. The 12 V supply from the DB9 connect is stepped down to 5 V by a Buck Regulator, which is then fed to a physical Slide Switch. The 5 V supply from the USB is also fed into the Slide Switch as the second input. Figure 3 shows a Buck Regulator connected to the Slide Switch with two 5 V inputs and one 5 V output. A 3.3 V power rail is then acquired by stepping down the 5 V output from the Slide Switch using a Voltage Regulator.

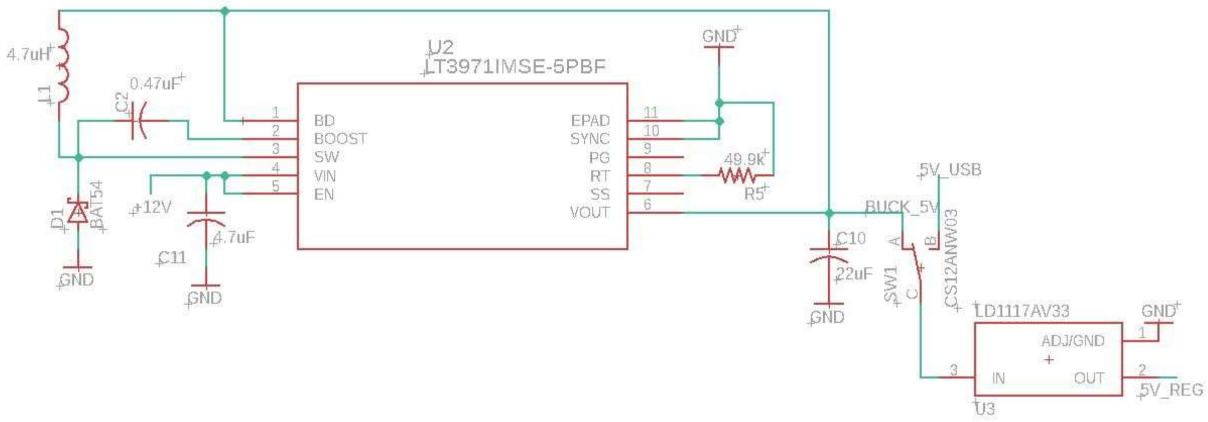

Figure 3: Schematic of Buck Regulator and Slide Switch

2.2.1 Voltage Regulator

The Voltage Regulator is a Low-Dropout (LDO) Regulator which is expected to convert a 5 V input to a 3.3 V output. The 3.3 V output power rail is expected to supply power to the Microcontroller in the Control Module.

2.2.2 Buck Regulator

The Buck Regulator is responsible for converting the vehicle battery voltage accessible through the OBD-2 Port via the DB9 Connector to a 5 V power rail. The input voltage to the Buck Regulator is expected to vary between 11 V and 14 V. The output voltage of 5 V is expected to be constant and will feed into one of the two inputs of the physical Slider Switch.

Table 1: Requirements and Verification for Buck Regulator

Requirement	Verification
1. Accept an input voltage (VIN) in the range: 11 V < VIN ≤ 14 V. 2. Must operate in stable conditions in temperatures between -15 °F and 100 °F.	1. Using a Digital Multimeter, we will probe the VIN pin of the buck converter to ensure that the voltage is in the range of 11 to 14 V when the car is on. 2. Use an IR thermometer to ensure the buck converter is operational and stays below 125°F.

3. Must output a maximum of 1.2 A to avoid burning out the Voltage Regulator.	3. Using a Digital Multimeter, we will probe the VOUT pin to ensure a maximum of 1.2 A.

2.2.3 Slider Switch **

The slider switch takes in the 5V outputs from both the USB cable and the Buck Regulator and allows us to choose which source we want to power the board with. This ensures that we can power the board while programming the ESP32 while retaining the ability to draw power from the 12V car battery once finished. This switch will be enclosed with the rest of the control unit, making it inaccessible to the end user. Figure 3 shows how the Slider Switch is connected to the Buck Regulator and the USB Connector.

** For debugging purposes only.

2.3 ECU (Engine Control Unit) Interface Module

The ECU Interface Module begins with an OBD-II to DB9 cable which provides access to the vehicle's ECU and 12V battery for us to use as a power source. The DB9 cable's data pins connect to the CAN Transceiver, allowing us to request and interpret ECU signals from the vehicle.

2.3.1 CAN Transceiver

The CAN Transceiver is the interface between the CAN physical bus and the CAN Protocol Controller (microcontroller) [5]. This sub-module is used to help convert the digital signals from the CAN Protocol Controller to Analog signals on the CAN physical bus when transmitting commands and vice-versa when receiving data. The CAN transceiver will enable communication between the CAN Protocol Controller and the ECU to request vehicle data such as vehicle RPMs and vehicle speed for further analysis by the microcontroller.

In figure 4, an example circuit diagram is provided for the MCP2551-I/P CAN Transceiver. In this figure, CANH and CANL are the high and low CAN Physical Bus pins respectively and connected to the vehicle through the DB9 connector. The TX and RX pins are transmitting and receiving pins respectively connected to the microcontroller via UART protocol. The IC is then powered by the 5 V power line and grounded to the signal ground provided by the DB9 connector.

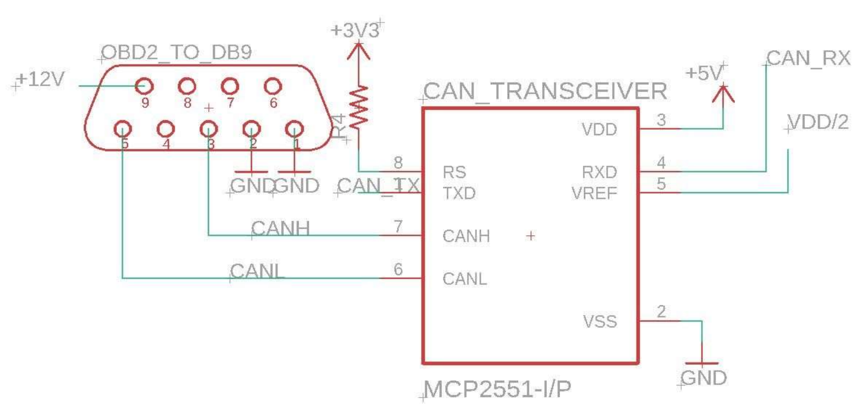

Figure 4: Schematic of MCP2551-I/P CAN Transceiver connected to the DB9 Pin

2.4 Control Module

The control module will consist of the microcontroller and input buttons. It will be in charge of handling, processing, analyzing, and transmitting data. The microcontroller will process data received from the ECU interface module and give commands to the audio module. The buttons allow the user to choose what kind of feedback they want from the assistant.

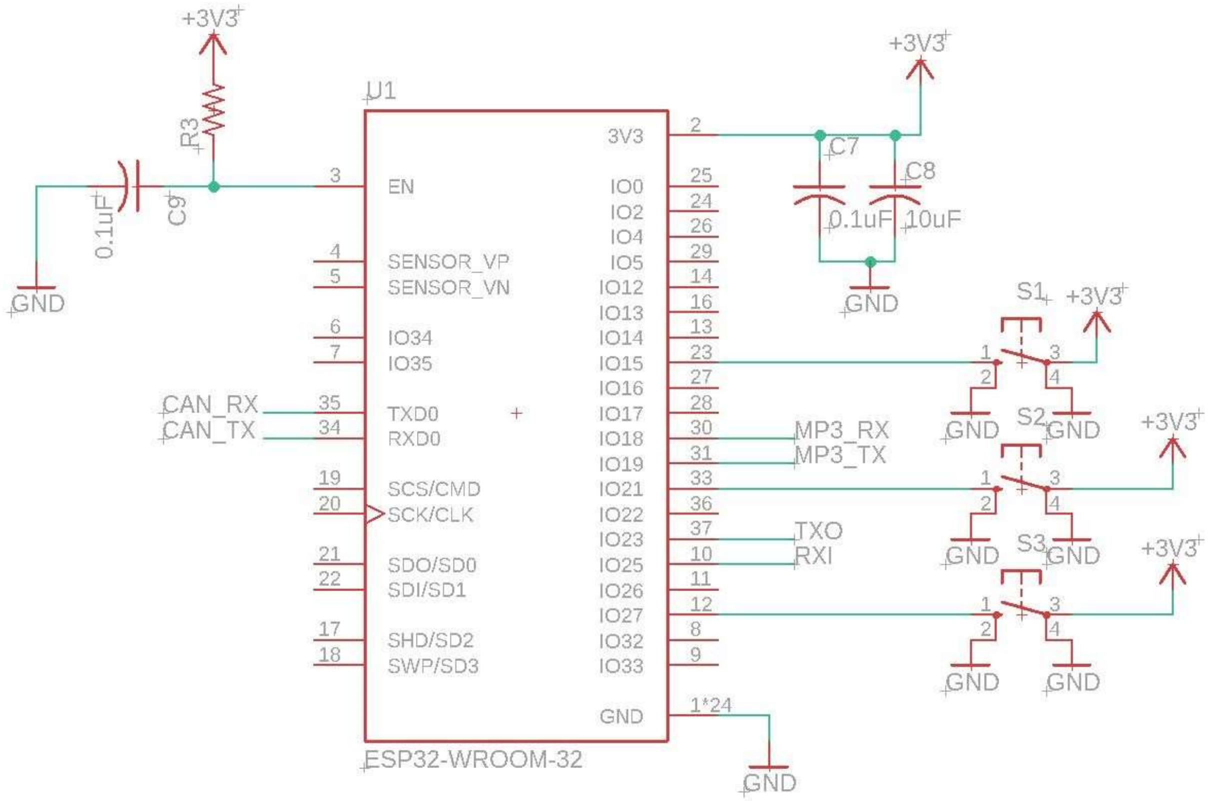

Figure 5: Schematic of the ESP32-WROOM-32 Microcontroller connected with Input Buttons

2.4.1 Microcontroller

The microcontroller is the central processing unit and acts as the glue circuit for the rest of the components and has three main functions. The three functions are retrieving live vehicle data such as RPMs and vehicle speed from the vehicle ECU, enabling the user to input which lesson to learn via the Button Breakout Board, and controlling the MP3 player to playback the correct audio cues through the Audio Module.

To achieve these functions, we chose the ESP-32 microcontroller from ESPRESSIF Systems. This microcontroller was chosen for its extensive user-friendly and open-source C programming libraries. Some of the libraries that will be used are the CAN protocol driver's library, the UART serial communication library, and the GPIO library. The CAN protocol driver's library gives the microcontroller the functionality of a CAN Protocol Controller and enables the use of CAN Protocol Commands. The UART communication library will be used to communicate with the CAN Transceiver, the MP3 Player, and the USB to UART converter. The GPIO library will be used to connect the buttons, which are intended to interface with the user.

Figure 5 shows how the microcontroller is connected to the rest of the circuits. Pin 2 in figure 5 is used to power the microcontroller with a 3.3 V input and two capacitors in parallel to stabilize the input power. Pin 3 is the microcontroller enable pin and will be connected to the 3.3 V power line with a resistor in series. The value of the resistor is to be decided and the capacitor will be used to stabilize the input power. Pins 23, 33, and 12 will be connected to active high buttons intended for user input and will be powered by the 3.3 V power line. Pins 35 and 34 are CAN_RX and CAN_TX UART pins respectively and connect to the CAN Transceiver. Similarly, pins 30 and 31 are UART pins connected to the MP3 Player and pins 37 and 10 are UART pins connected to the USB to UART converter. There are multiple GND (signal ground) pins on the ESP32-WROOM-32 and in figure 5 it is represented by pin 1*24.

Table 2: Requirements and Verification for Microcontroller

Requirement	Verification
1. Must have an input voltage (VDD) of 3.3 V +/- 5%. 2. Must be operational in temperatures < 85°C. 3. Must be able to communicate with the CAN Transceiver via TX and RX pins.	4. Connect terminals to an oscilloscope and verify the transceiver is receiving within 5% of 3.3 V. 5. Use an IR thermometer to ensure the microcontroller is operational and stays below 85°C. 6. Probe the TX and RX pins on the microcontroller with an oscilloscope to ensure signals are coming through.

2.4.2 Button Breakout Board

The button breakout board will be used for drivers to skip or go back to a particular driving lesson. If the driver chooses to relearn a lesson the option is available to do so. These input buttons communicate with the microcontroller via GPIO and are monitored to make various decisions in software. The buttons will be placed on a small breakout board and connected to the microcontroller via a semi-long wire external to the PCB. This allows for the buttons to be placed near/adjacent to the steering wheel where the user can have easy access without causing distractions. As shown in figure 6, the buttons will be designed to be active high with the 3.3 V power rail as their source. The buttons will also be debounced via a software library.

Table 3: Requirements and Verification for Input Buttons

Requirement	Verification
1. Input Buttons must be debounced to prevent processing multiple presses.	1. Using software, we will verify that a button press will only be counted once.

2.5 Program and Debug Module

The program and debug module involves the USB port and the USB to UART Converter. This module is strictly for programming and debugging the microcontroller as well as monitoring the data that is received from the ECU interface module during the prototyping and build phases. This module will not be accessible to the user nor will it be active in the final product.

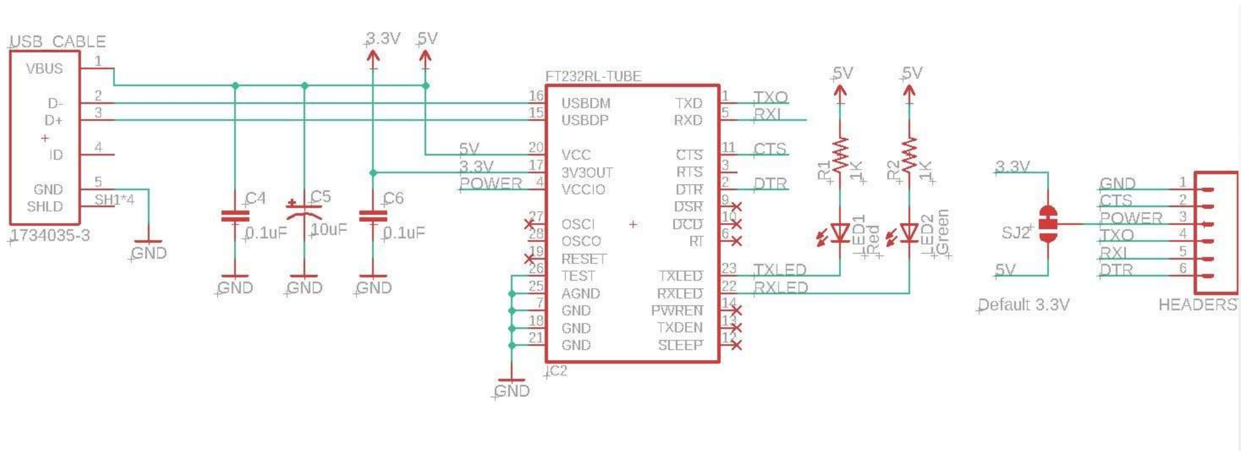

Figure 6: Schematic of the USB Port connected with USB-to-UART Converter

2.5.1 USB Port

The USB port will allow communication through a USB cable which will be connected to a laptop. This port connects the USB to UART Converter which allows communication to the microcontroller as the ESP32 microcontroller does not have USB D+/D- pinouts by default.

2.5.2 USB to UART Converter

The USB to UART converter will receive data from a laptop over USB and will convert the data to serial UART which can interface directly with the microcontroller. This device will serve as a bridge between the USB port and the microcontroller.

Table 4: Requirements and Verification for USB to UART Converter

Requirement	Verification
1. LEDs must light up when data is being transmitted / received. 2. Data from our laptops must be transmitted through a USB cable to the MCU.	1. We see LEDs light up on the board. 2. We will program the ESP32 to turn on and off an LED connected through GPIO pins to let us know that our instructions are transmitting successfully.

2.6 Audio Module

The audio module involves the AUX port and a breakout MP3 Player board with a built in SD Card slot. This module will provide audio instructions to the driver which can be heard by the vehicle's speakers. The design is to prerecord audio cues and store them on the SD Card.

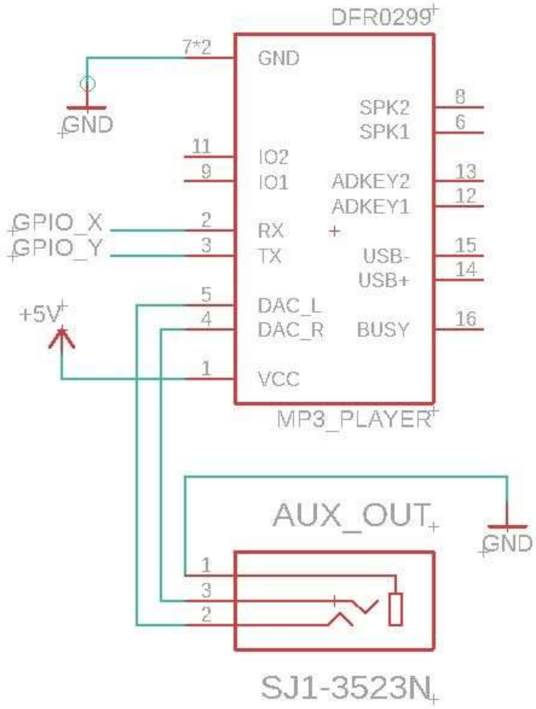

Figure 7: Schematic of MP3 Player Board connected with AUX Port

2.6.1 Mini MP3 Player

The MP3 player will receive playback commands from the microcontroller and output the appropriate audio file from the attached SD Card reader. The command set is specified in a provided library. The MP3 player will send its left and right channel audio output via an auxiliary cable. This auxiliary cable will connect to the vehicle's AUX Port.

Table 5: Requirements and Verification for Mini MP3 Player

Requirement	Verification
1. Must have an input voltage (VDD) of 3.3 V +/- 5%. 2. Must be operational in temperatures < 75°C. 3. Communicate with the AUX Port via auxiliary cable.	1. Connect terminals to an oscilloscope and verify the transceiver is receiving within 5% of 3.3 V. 2. Use an IR thermometer to ensure the MP3 player board is operational and stays below 75°C. 3. Use the vehicle speakers to verify audio is transmitted properly.

2.6.2 SD Card

The SD Card will be needed to store the prerecorded audio instructions which will ultimately be used by the MP3 player to playback the commands via the vehicle's speakers. An SD Card with a minimum of 2 GB will be used for our project.

2.6.3 AUX Port

An auxiliary port will be wired to connect with the MP3 player. An auxiliary cable will then connect this port to the vehicle's AUX Port which will then playback audio to the left and right speakers of the car.

2.7 Circuit Schematic

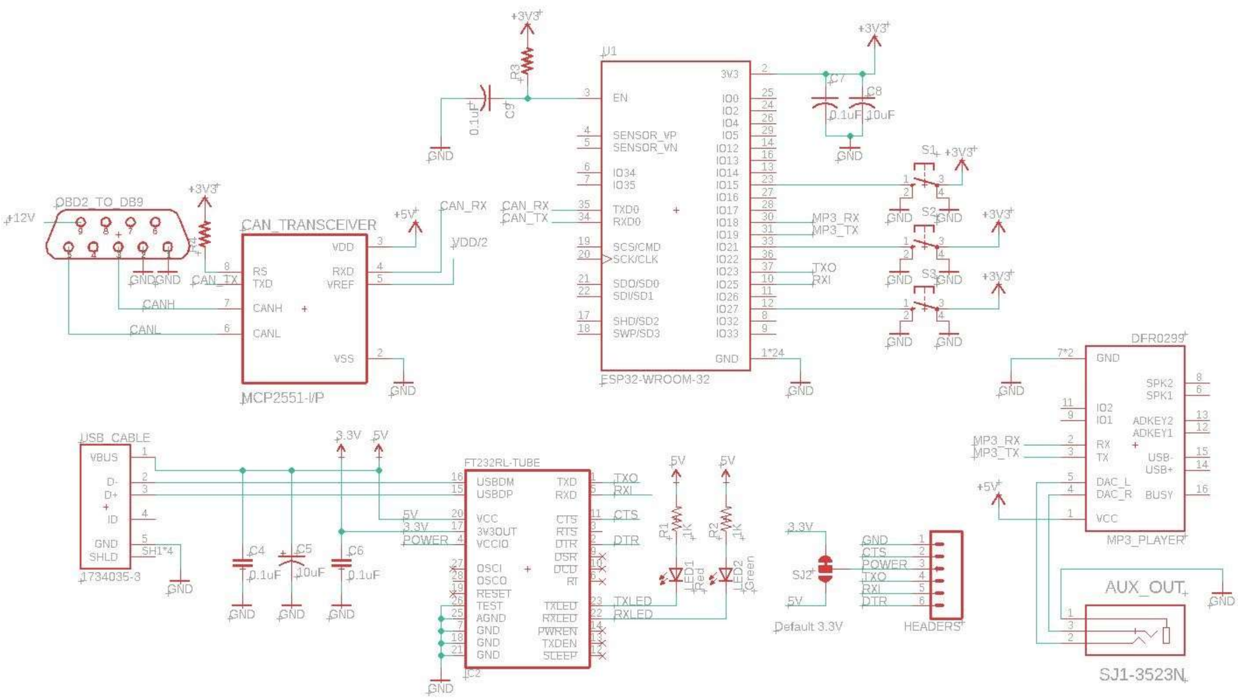

Figure 8: Overall Circuit Schematic

2.8 Software Flowchart

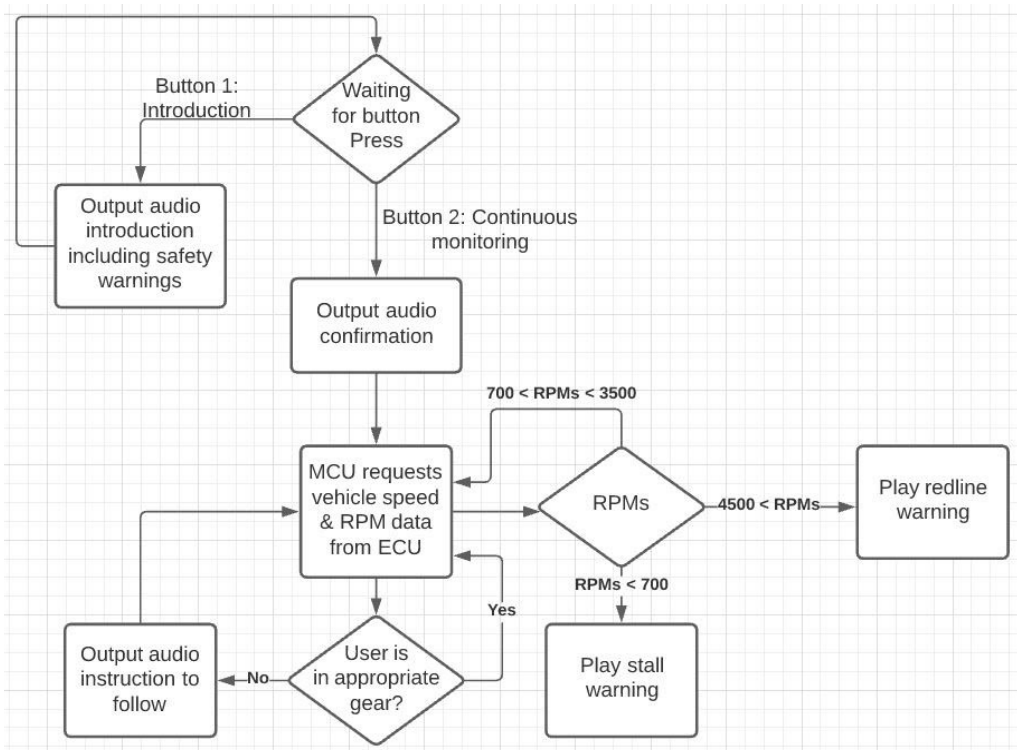

Figure 9: Software Flowchart

2.9 Tolerance Analysis

One crucial element for the success of this project is our ability to power everything on our board. Our tolerance analysis will therefore focus on the voltage tolerances and current draws of our components to ensure no module gets burned out nor is left unpowered.

Healthy 12V car batteries tend to output voltages between 11.5 V and 14.4 V. The Buck Regulator must be capable of handling the entire range of voltages between 11 V and 15 V to be safe. For this reason, we chose the LT3971-5 Buck Regulator, which can handle an input voltage range of 4.5 V to 38 V, giving us plenty of overhead.

The Buck Regulator outputs 5 V at a maximum of 1.2 A to the slide switch (CS12ANW03) which is rated for 3 A and up to 125 VAC or 176.77 VDC. The other output to the slide switch is the USB 2.0 connector which has a typical output of 5 V at 0.5 A, which fits within the switch's specifications as well.

The 5 V output of the slide switch feeds into the CAN Transceiver, the Voltage Regulator which converts 5 V to 3.3 V, and the MP3 Player. The CAN Transceiver has a maximum current draw of 365 µA and an input voltage tolerance of 4.5 V to 5.5 V. The LP3875-3.3 Voltage Regulator has a maximum output current of 1.5 A at 3.3 V ± 1.5% (3.2505 V to 3.3495 V) and can tolerate input voltages between 3.8 V and 7 V. This gives us a wide margin of error for the regulator input.

The ESP32-WROOM2 Microcontroller has an average operating current draw of 80 mA, but requires a minimum supply of 0.5 A. It accepts between 3 V and 3.6 V input, giving us a 10% margin of error for the voltage regulator's output, well within the specified 3.3 V ± 1.5%.

The MP3 player (DFR0299) is powered from the 3.3 V voltage regulator output and has a 20-mA standby current draw, though we are unable to find regular operating power needs. It can accept an input voltage between 3.2 V and 5 V, so we are confident that the voltage regulator's output will suffice.

3. Cost and Schedule

3.1 Cost of Parts

Table X: Part Cost Breakdown

Quantity	Part Name	Part Number	Cost
	Audio Jack	SJ1-3525N	$0.76
	CAN Transceiver	MCP2551	$1.20
	MP3 Mini Player Board	DFR0299	$8.90
	Voltage Regulator	LD117AV33	$1.95
	Microcontroller	ESP32-WROOM32	$8.95
	DB9 Connector Pin	PRT-00429	$1.75
	Mini USB Socket	1734035-3	$1.24
	USB to UART Bridge	FT232R	$4.50
	Input Buttons		
	Buck Converter	LT3982EDD-5	$3.34
	Slide Switch	CS12ANW03	$2.25
		Total Cost	$34.84

3.2 Labor Cost Breakdown

Table X: Labor Cost Breakdown

Team Member	Hourly Wage	Weekly Hours	Number of Weeks	Multiplier	Cost Per Member
Aadhar	$38.00	20	12	2.5	$22,800
Ian	$38.00	20	12	2.5	$22,800
Maulin	$38.00	20	12	2.5	$22,800
				Total Labor Cost	$68,400

3.3 Schedule

Table X: Schedule

	Aadhar	Ian	Maulin
2/22	Block Diagram, Design Document	Schematics, Design Document	R&V Tables, Design Document
3/1			
3/8			
3/22			
3/15			
3/29			
4/5			
4/12			
4/19			

4. Ethics and Safety

4.1 Ethics

When creating an educational tool, it is imperative to instruct the learner in a safe manner. Teaching someone how to drive prompts many factors that can harm the public. The IEEE Code of Ethics states "to disclose promptly factors that might endanger the public or the environment" [6]. In response to this, we will involve a safety warning system that discloses all the pertinent information before even instructing the driver. We also understand that misuse of this assistant may occur and that is largely up to the user's decisions. Our educational assistant can malfunction and this is something that we will mitigate through thorough testing, ensuring that the driver and anyone in the car is out of harm's way.

4.2 Safety

Our Educational Stick Shift Assistant will produce safety concerns. The Illinois 2020 Rules of the Road "prohibits the use of handheld cell phones, texting or using other electronic communications while operating a motor vehicle" [7]. This rule is something we expect anyone who uses this assistant to be aware of. However, there are concerns that distracted driving is something that can endanger others. Another concern is that some may drive in traffic intensive areas which is not the purpose of this project. It is an educational tool that is intended to be used in a safe area, preferably an empty parking lot. We are anticipating that anyone who uses this tool will have a license and has some information about the rules and regulations of driving on the road.

5. References

[1] T. Hofstätter, M. Krawina, B. Mühlreiter, S. Pöhler, and A. Tschiesner, "Reimagining the auto industry's future: It's now or never," *McKinsey & Company*, 06-Nov-2020. [Online]. Available: https://www.mckinsey.com/industries/automotive-and- assembly/our-insights/reimagining-the-auto-industry's-future-it's-now-or-never. [Accessed: 20-Feb-2021].

[2] J. M. Vincent, "What You Need to Know About Coronavirus and Cars," *U.S. News & World Report*, 14-Dec-2020. [Online]. Available: https://cars.usnews.com/cars-trucks/coronavirus-and-cars. [Accessed: 10-Feb-2021].

[3] Autolist Editorial, "Manual vs Automatic - Pros and Cons," *Autolist*, 29-Jul-2019. [Online]. Available: https://www.autolist.com/guides/manual-vs-automatic. [Accessed: 10-Feb-2021].

[4] S. Gautam, R. Pansare, A. Chaudhary, and K. Gupta, "Why Does Europe Prefer Manual Cars Over Automatic Ones?" *Get My Parking Blog*, 17-Feb-2020. [Online]. Available: https://blog.getmy parking.com/2020/01/20/why-does-europe-prefer-manual-cars-over-automatic-ones/. [Accessed: 10-Feb-2021].

[5] "CAN Transceivers," *Maxim Integrated*. [Online]. Available: https://www.maximintegrated.com/en /products/interface/transceivers/controller-area-network-transceivers.html#: ~:text=A controller area network (CAN, loading down the system microcontroller. [Accessed: 25-Feb-2021].

[6] "IEEE Code of Ethics," *IEEE*. [Online]. Available: https://www.ieee.org/about/corporate/governance/ p7-8.html. [Accessed: 12-Feb-2021].

[7] Publications/Forms. [Online]. Available: \https://www.cyberdriveillinois.com/publications/. [Accessed: 20-Feb-2021].