Project Proposal

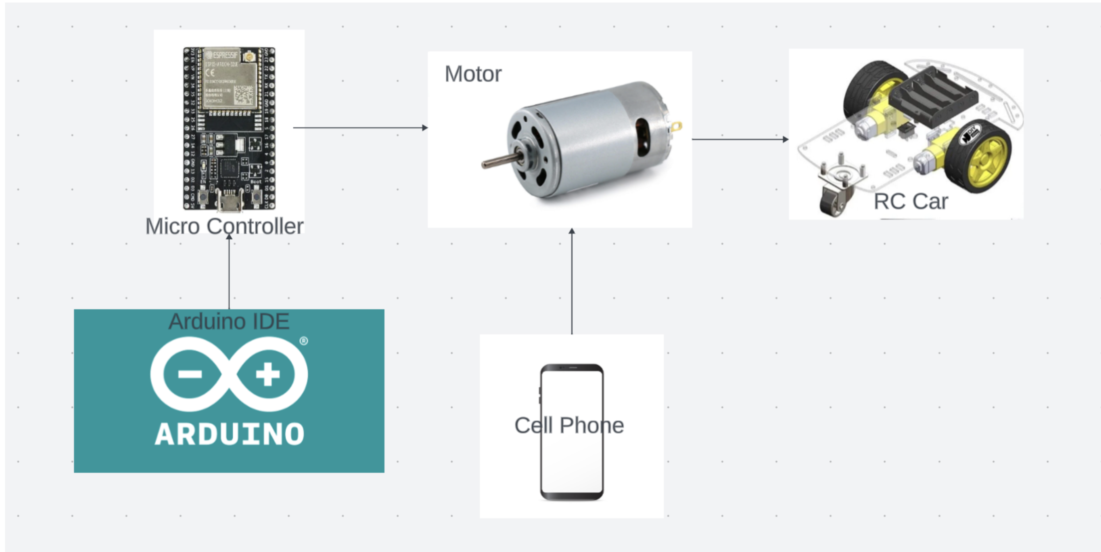
Aaron Chen, aaronkc2
Kyungha Kim, kyungha2
Lee Boon Sheng, bsl3

1. Introduction
    a. Problem
        i. The need for efficient and convenient motor control is prevalent in various applications, such as robotics, automation, and remote-controlled vehicles. Existing solutions often lack simplicity and ease of use, making them less accessible to a broader range of users. Therefore, there is a demand for a wireless remote motor controller that is simple, user-friendly, and suitable for a variety of applications, including robotics and small wireless carts.

    b. Solution
        i. Our project aims to develop a Wireless Remote Motor Controller that provides an adjustable speed range of 0 to 100%. This controller will be designed to work with a simple wireless remote control using either infrared (IR) or radio frequency (RF) technology. The key features of the controller will include functions like start, stop, accelerate, and decelerate, making it intuitive and easy to learn for users of all skill levels. Additionally, it will be designed to send a single signal that can be used in conjunction with the immediately preceding motor control project, facilitating compatibility with existing systems.

            Furthermore, as an alternative design, we will explore the possibility of controlling a pair of motors to support steering, opening up the potential for building highly efficient robotic platforms or small wireless carts. It should feature closed loop speed control, current limiting control and this machine will be operated under 24DC.

c. Visual Aid

i. 

d. High-level requirements list

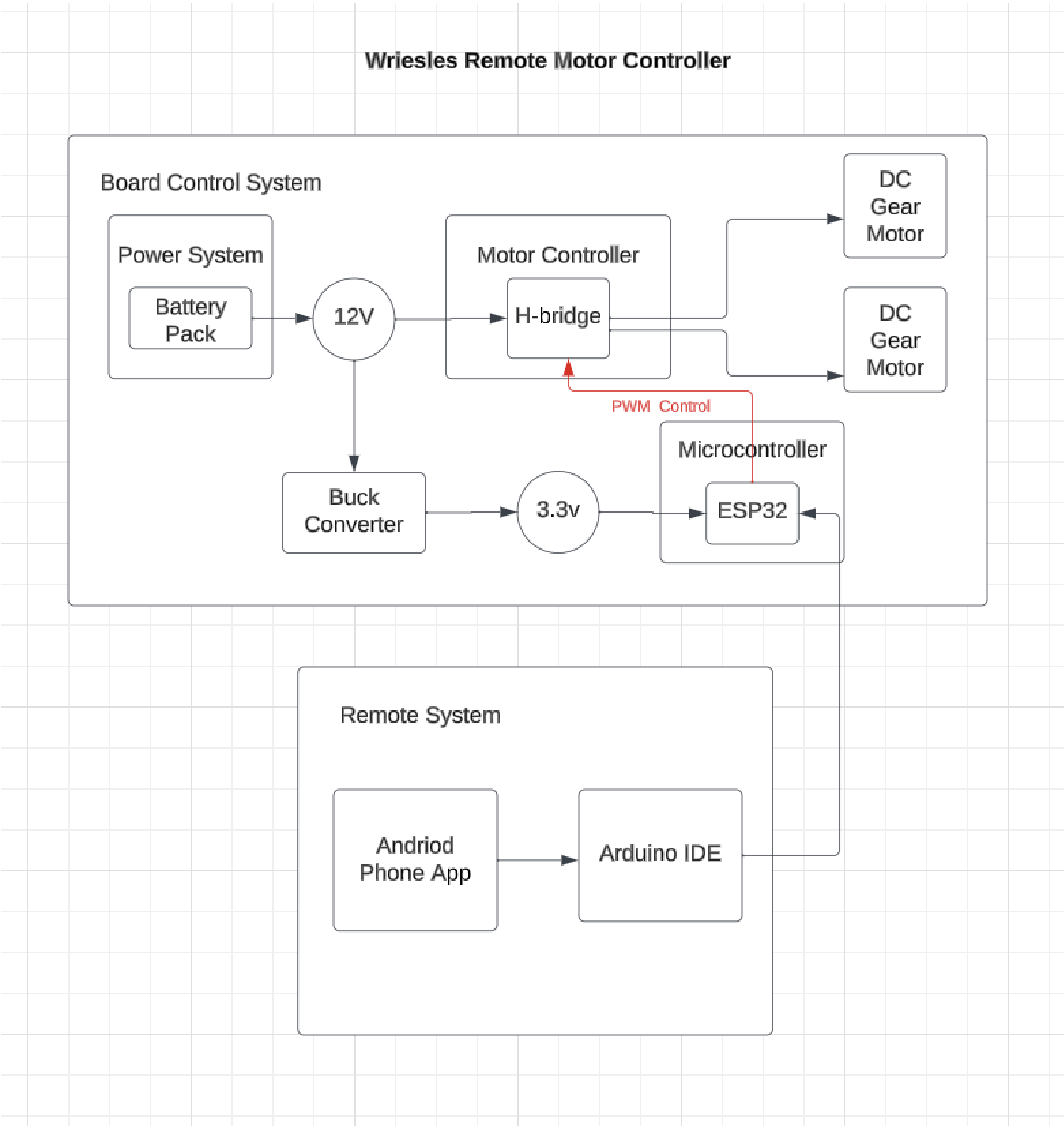
i. Wireless Control
   1. The system requires wireless control of a DC motor using Wi-Fi and allows remote control within a 5-10 meter range in open spaces.

ii. Voltage and Current Control:
   1. The motor controller needs to support a specified voltage range, preferably 12-24V DC and handle a maximum of 10A current.

iii. Speed Control
   1. Users should be able to control the motor's speed from 0 to 100%

2. Design

a. Block Diagram



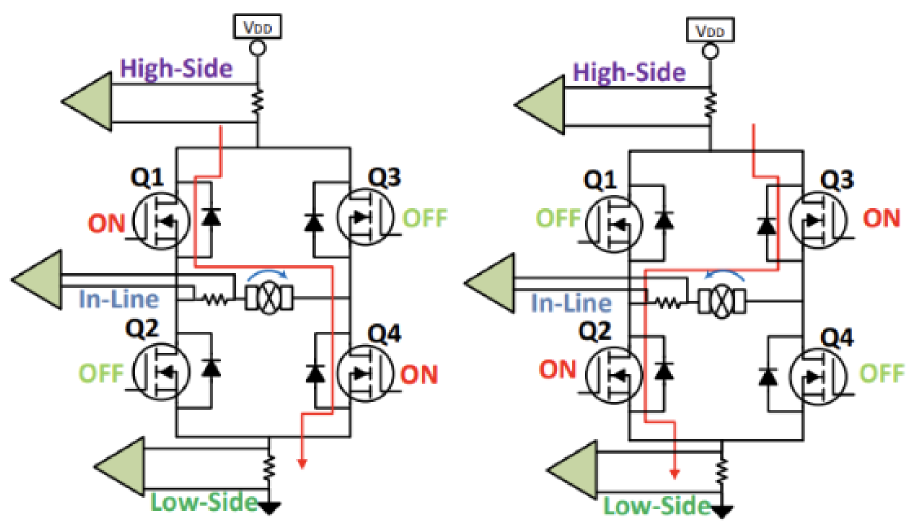
b. Subsystem Overview

i. **Board Subsystem:**

1. The **Power Subsystem** is responsible for managing the electrical power supply to the entire system. In our design it will comprise of a 12V DC battery input, a 3.3V solder jumper coupled with a buck converter for the ESP32-S3 module, and a battery voltage sensor

with Schottky diodes to safeguard the system's stability and safety. Without a stable power supply, none of the other subsystems can function effectively.

2. The **Motor Controller Subsystem** will include the hardware and software necessary to control the motor's speed, direction, and braking. The heart of this subsystem is the H Bridge circuit design includes four MOSFETS, two gate drivers for each side of the bridge with the associated bootstrap capacitors, and an Arduino Uno used to create the PWM signals that are fed into the gate drivers. The figure below shows the basic circuit of an H Bridge circuit. When Q1 and Q4 are on, the left lead of the motor will be connected to the power supply(battery pack for RC Car demo) and current will start flowing in the forward direction and the DC gear motor shaft will start spinning. Conversely, when Q2 and Q3 are turned on, the current will flow in the opposite direction and the dc geared motor shaft will start spinning backwards.



H-Bridge Circuit

3. For the **Current Sensing Subsystem**, as we are dealing with a maximum current rating of 10A, current sensing will be used to monitor, manage, and control the load currents leading to improvement in safety, and reliability of our motor controller circuit. From the H-bridge circuit shown below, if Q1, and Q2 are both on or Q3, and Q4 are both on, it will cause a short circuit from battery to ground. In our H-bridge circuit design, the current sensor will be placed at High-Side to prevent short circuit from

happening. The current sensor will also be placed at In-Line to direct motor current measurement and low-bandwidth amplifier.

4. For the **Microcontroller Subsystem**, the microcontroller employed for this project is the ESP32S that interfaces with the motor controller. 3.3V DC supply is designed for the ESP32S microcontroller and it is used to feed PWM signals to the motor driver. PWM is achieved by varying the pulses applied to the enable pin of the H-bridge to control the applied voltage of the motor.
5. The **Buck Converter**, an essential part of the power system, efficiently steps down the voltage from the 12V source to the required 3.3V level. This conversion process ensures that the ESP32-S3 module receives a stable and precisely regulated power supply. Buck converters are known for their efficiency, making them an excellent choice for conserving battery power while providing a clean and consistent voltage source.

ii. Remote Subsystem:

1. The **Android Phone App** provides wireless control for the DC motor, allowing users to send a signal that commands the behavior of the motor. These commands are delivered to the Motor Controller Subsystem and determines whether the DC geared motor to accelerate, decelerate, move forward, move backward, or stop. The H-bridge circuit on the board system drives the DC gear motor based on these commands.
2. **Arduino IDE** programs the ESP32 microcontroller. The code defines how the DC geared motors should respond to inputs received from the Android Phone App. For instance, if deceleration commands are sent via the app, the code lowers the duty cycle to reduce motor speed. In short, the Arduino IDE enables ESP32 to control and fine-tune the motor controller based on user inputs.

c. Subsystem Requirements

i. Board Subsystem:

1. **Power Subsystem**: This system must be able to supply at least 12V from the battery pack to the rest of the system.
2. **Motor Controller:** The motor controller subsystem must receive a 12 to 24V DC supply voltage to drive the DC geared motor. Besides, a gate driver is needed to boost the signals from the

ESP32S to send high or low signals to the H-bridge circuit. Subsequently, the H-Bridge circuit will determine if the motor will rotate in the forward or backward direction.

3. **Current Sensing Subsystem:** An INA240 current sense amplifier with enhanced PWM rejection, which ranges from -4V to 80V will be used in the current sensing subsystem. The current sensor must receive a stable 3.3 power supply to be switched on. A shunt resistor has to be placed in between the voltage source and the motor to measure the current flowing through the motor.
4. **Microcontroller Subsystem:** ESP32-S3 module must receive a stable 3.3 power supply to be switched on. This subsystem must be able to send PWM signals to the motor controller subsystem to control the DC geared motor's speed, direction, and braking.
5. **Buck Converter:** This system is needed to step down the voltage supply to the input voltage of the microcontroller. This is because the ESP32 microcontroller will be operated at 3.3V.

ii. Remote Subsystem:

1. The microcontroller employed for this project is the ESP32S that interfaces with the motor controller. 3.3V DC supply is designed for the ESP32S microcontroller and it is used to feed PWM signals to the motor driver. PWM is achieved by varying the pulses applied to the enable pin of the H-bridge to control the applied voltage of the motor. The board system will then wirelessly communicate with the remote system via a Phone App designed using Arduino IDE. The DC gear motor will be driven by the H-bridge circuit.
2. The Pulse width Modulation signals(PWM) to the H bridge will range from 0 to 100%, where 0 the car will be stopped and 100% will depend on the max output voltage from the source. The encoder of the DC motor will then output the RPM of the motor and display the real-time value on the app.
3. The remote app system will have less than a second response time ensuring its near instantaneous performance when pressing each button command..
4. The Remote subsystem should also be able to communicate with the microcontroller from 5-10 meters in open space.

Tolerance Analysis

- **Voltage Regulator**

  - Because we are using the ESP32-S3-WROOM, we do not require a linear voltage regulator. Our input voltage will be ranging from 12 to 24V, and the motor controller can accept input voltage of this range. ESP32 microcontrollers usually draw a maximum current of 240mA. The dropout voltage would be 24-3.3=19.7V and P=V*I=19.7*240m=4.728W dissipated in the regulator. Therefore, we would need to use a buck converter to step down the high voltage for the 3.3V for the ESP32-S3 module. This provides a much greater power efficiency than the linear regulators.
- **Choosing MOSFETs for H-Bridge**
  - The majority charge carriers of p-channel MOSFETs are holes, and it has a lower mobility compared to electrons. This has resulted in higher on-resistance of p-channel MOSFETS compared to n-channel MOSFETS. The formula for power is I^2R, to minimize the power dissipated in the H-bridge circuit, four N-MOSFETs are used at the low side and high side instead of two n-channel types at the low side and two p-channel types at the high side.
  - While choosing the N-MOSFETS for the H-bridge circuit, it is important to pay attention to the datasheets. According to the datasheet, IRF520N has a drain-to-source breakdown voltage of 100V. It is important to keep the voltage across drain-to-source below 100V, so the drain current will not exceed 250μA. To avoid the inductive spike during switching transients exceeding the breakdown voltage, we will derate the voltage range by 60-70%. Since our system uses 12-24 DC, we will use the MOFSETs with a breakdown voltage rating of at least 20.4-40.8V. besides, the Rds(on) is specified at 10V VGS and our input voltage is ranged from 12 to 24VDC and the mosfet will be fully switched on.
- **Capacity of Microprocessor**
  - The ESP32 microcontroller is a dual-core processor, Wi-Fi and Bluetooth connectivity, and a variety of I/O options. Since we will achieve speed control using the PWM waveform and duty cycle, it is important to discuss the PWM frequency of the microprocessor. The user is allowed to set the duty cycle (0-255 for 8-bit resolution). In the code, we measure the pulse_width and pulse_period for the duration of time during which the PWM signal remains in its "high" state, or the "on" state. Then, the duty cycle is calculated as a percentage using the formula: duty_cycle = (pulse_width / (float)pulse_period) * 100.0.

3. Ethics and Safety
   a. In the development of the Wireless Remote Motor Controller project, we are committed to upholding the highest ethical and safety standards as outlined in the IEEE and ACM Code of Ethics. Specifically, we will prioritize safety by

ensuring the device complies with ethical design and sustainable development practices. Safety is a paramount concern throughout the development and operation of the Wireless Remote Motor Controller project. It extends to various aspects, including power control, voltage regulation, soldering practices, and the proper use of equipment. Here, we emphasize the safety measures and considerations associated with these critical project components: To prevent overheating and protect the motor and other components, the power system must implement current limiting mechanisms. For this purpose, we have implemented a current sensor between the DC motor and the voltage source to prevent any short circuit when both MOFSETs are switched on on one side. This ensures that the motor operates within safe limits, reducing the risk of damage or accidents. Maintaining a stable voltage supply, as achieved through the buck converter, is essential for the safety of the entire system. Fluctuations in voltage can lead to erratic motor behavior and pose risks to users and equipment. We will continuously monitor the voltage levels as a safety measure. The battery voltage sensor helps in this regard, allowing the system to take corrective actions if voltage levels fall outside safe operating limits. When working with electrical equipment, including the buck converter and voltage sensor, it is crucial to follow electrical safety practices, such as isolating power sources when making connections. Incorporating these safety measures and considerations into the Wireless Remote Motor Controller project not only ensures the safety of the development process but also contributes to the overall safety of the end product. During soldering, we will make sure to use safety equipment like safety glasses and be extremely careful with the soldering iron. Prioritizing safety at each stage of the project's lifecycle, from design and assembly to testing and operation, demonstrates a commitment to delivering a reliable and secure product. For the H-Bridge circuit design, we will incorporate voltage input protection mechanisms to safeguard against voltage spikes, surges, or reverse polarity, reducing the risk of damage to the controller and connected motors. This requirement enhances the controller's durability and reliability. I, Aaron Chen, Kyungha Kim, and Lee Boon Sheng Adhere to this.