FPT UNIVERSITY IOT102T

BLUETOOTH CONTROLLER CAR PROJECT

FPT UNIVERSITY

The Project is Implemented by Team 4

FPT University, Ho Chi Minh Campus, Vietnam
Email: minhdlhse180733, quanghdse180734, bacnvse180739, lampqse182754, quyettvse181574@fpt.edu.vn

FPT UNIVERSITY IOT102T
BLUETOOTH CONTROLLER CAR PROJECT

FPT UNIVERSITY

FPT UNIVERSITY

THE PROJECT IS IMPLEMENTED BY TEAM 4

Doan Ly Hong Minh – SE180733
Ho Dang Quang – SE180734
Nguyen Van Bac – SE180739
Phan Quoc Lam – SE182754
Tran Van Quyet – SE181574
FPT University, Ho Chi Minh Campus, VietNam
minhdlhse180733, quanghdse180734, bacnvse180739,
lampqse182754, quyettvse181574}@fpt.edu.vn.

Abstract

The Arduino Car project uses Bluetooth remote control technology [1], along with IR and ultrasonic sensors, to provide safe and flexible movement. Equipped with a motor control system, the car can move forward, backward, turn left, and turn right. The project also integrates a turn signal system using WS2812B LEDs for left and right turn signals and reverse lights. The IR sensors help detect obstacles, activating the car to reverse and stop when necessary. The ultrasonic sensors measure the distance to obstacles ahead and emit sound and light warnings when approaching obstacles. The Arduino car's control circuit, DC motors, L298N motor driver, combined with the FastLED library for LED control, enables this intelligent movement. The goal of the project is not just to create a car that can move and respond to control commands, but also to produce a unique, intelligent product that can provide economic benefits.

I. Introduction

Research in both domestic and international contexts focuses on challenges such as navigation, energy efficiency, communication, and cost-effectiveness. In Vietnam, common challenges include limited sensor range, short battery life, and unstable signals. For instance, many RC car products on the market today primarily focus on basic remote control capabilities without automation features, leaving users vulnerable to risks such as collisions with obstacles due to delayed recognition. This is largely because sensors, such as infrared (IR), can only detect obstacles when the car is already too close.

In contrast, our car is equipped with advanced sensors, including IR and ultrasonic sensors, to help users avoid unnecessary collisions and damage during remote control operation. These sensors can identify obstacles at a safe distance, provide warnings, and activate avoidance mechanisms if necessary. This ensures greater safety for users, particularly in complex environments such as crowded areas or uneven terrain.

These improvements not only enhance the car's performance but also significantly reduce the risk of accidents, especially in challenging remote control scenarios. Furthermore, these solutions lay the groundwork for the development of intelligent transportation vehicles in the future—vehicles capable of monitoring their surroundings, adapting to changing conditions, and optimizing energy efficiency. This represents not only a leap forward in remote control car technology but also potential applications in real-world transportation systems, supporting automation, enhancing safety, and reducing energy consumption on a larger scale.

II. Main Proposal

Necessity and topicality, the scientific and practical significance of the research topic.

A. System models and block diagram

The system is designed with the following hardware components:
left=0.5cm

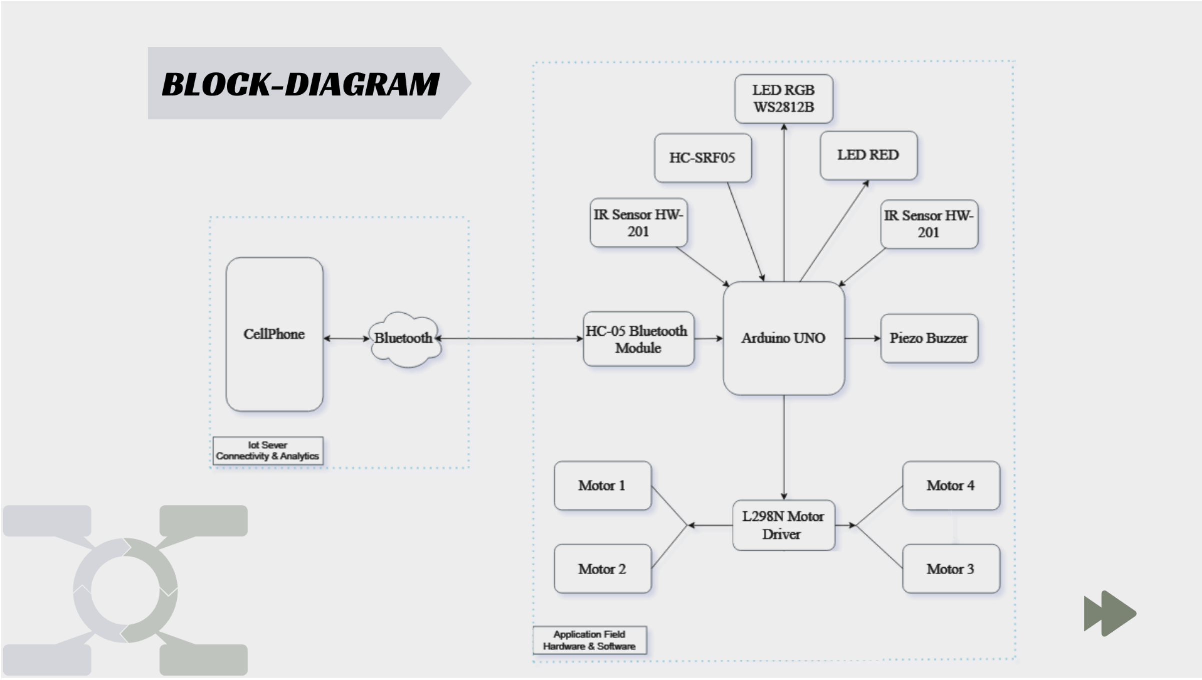


Fig. 1. Block diagram of the developed system

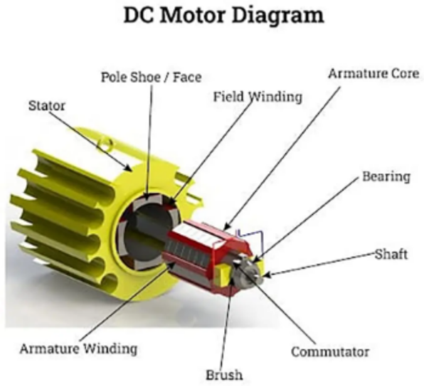
- **Arduino Uno:** Serves as the central processing unit.
- **Infrared Sensor (IR):**(2 devices) Detects nearby obstacles and prevents the vehicle from crossing defined boundaries.
- **Ultrasonic Sensor:** Measures the distance to obstacles, detects objects in the vehicle's path, and activates a buzzer when the distance is less than 20cm.
- **HC-05 Bluetooth Module:** Enables wireless communication to receive commands from a remote control device, such as a smartphone.
- **DC Motors and L298N Driver:** [2] Provide precise control over the vehicle's movement in different directions.
- **WS2812B LEDs:**(3 devices) Visual indicators of the vehicle's status:
  - Turn signals: Indicate left and right turns.
  - Reverse lights: Illuminate when the vehicle is moving backward.
  - Forward or normal state: Lights up in green.
- **Power Supply:** A portable battery pack powers the entire system.

The block diagram below provides a clear overview of the system's structure and interactions between components:

B. Components and peripheral devices

The development of the RC Car involves the integration of multiple components and peripheral devices, each serving a specific function to enhance the vehicle's capabilities. The key components include:

DC Motor



Structure

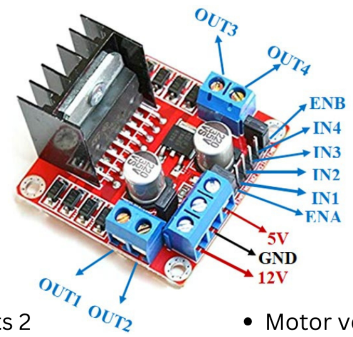
- Stator: Permanent magnets or field windings.
- Rotor (Armature): Wound coils and commutator.
- Brushes: Deliver current to the rotor.
- Shaft: Transfers rotational motion.
- Bearings: Reduce friction.

Features

- Voltage: Commonly 5V, 12V, or 24V.
- Speed: Proportional to the supply voltage.
- Torque: Inversely proportional to speed.
- Starting Current: Higher than the nominal current.
- Control: Via voltage adjustment or PWM.

Fig. 2. DC Motor

L298N Motor Driver



Structure:

- L298N IC: Dual H-Bridge, supports 2 channels, max current 2A/channel, max motor voltage 46V.
- Connectors:
- VCC, GND, 5V: Power inputs.
- IN1, IN2, IN3, IN4: Control motor direction.
- ENA, ENB: Enable speed control via PWM.
- OUT1, OUT2, OUT3, OUT4: Motor outputs.
- Heatsink: Metal heatsink for high current operations.
- Flyback Diodes: Protect against back EMF.

Features:

- Motor voltage: 5V - 35V, logic voltage: 5V.
- Max current: 2A/channel.
- Speed control: PWM signal on ENA/ENB.
- Direction control: Logic on IN1-IN4.
- Integrated jumper for logic power.

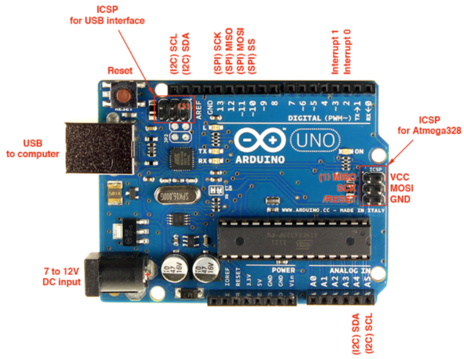
:

Fig. 3. L298N

Arduino UNO

Structure

- Microcontroller: ATmega328P.
- Pins:
- 14 Digital I/O (6 PWM).
- 6 Analog Inputs.
- Power:
- Vin: 7-12V.
- 5V, 3.3V: Regulated outputs.
- GND: Ground.
- USB Port: Power and programming.
- ICSP Header: For programming.
- Reset Button: Microcontroller reset.



Features

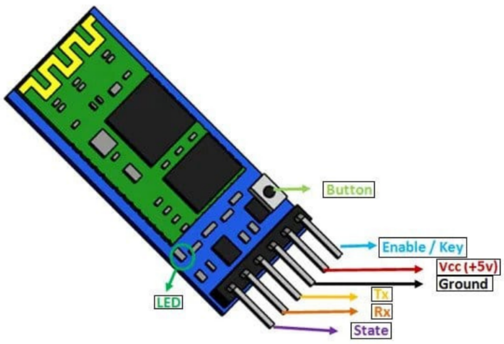
- Voltage: 5V logic, 7-12V input.
- Memory: 32 KB Flash, 2 KB SRAM, 1 KB EEPROM.
- Communication: Serial (UART), SPI, I2C.
- Programming: Arduino IDE.

Fig. 4. Arduino UNO

HC-05 Bluetooth Module

Structure

- Microcontroller: Handles Bluetooth communication.
- Antenna: Built-in for wireless signal transmission.
- Pins:
  - GND: Ground pin.
  - KEY/Enable: Puts the module into command mode for configuration.
  - TX: Transmits data to other devices.
  - RX: Receives data from other devices.
- Status LED: Indicates the module's state.

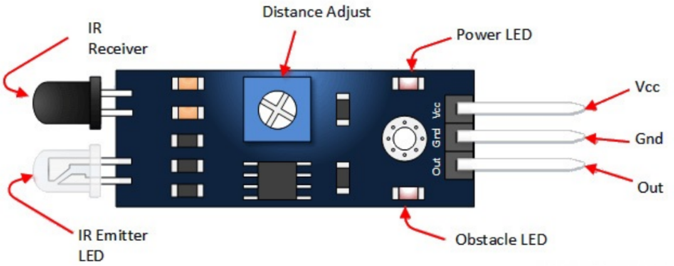


Features

- Bluetooth Version: 2.0+EDR.
- Range: Up to 10 meters.
- Baud Rate: Default 9600 bps, configurable.
- Modes: Command mode (configuration), Data mode (communication).
- Power Supply: 3.3V (supports 5V input).
- Interface: UART

Fig. 5. HC-05

IR Sensor HW-201



Structure

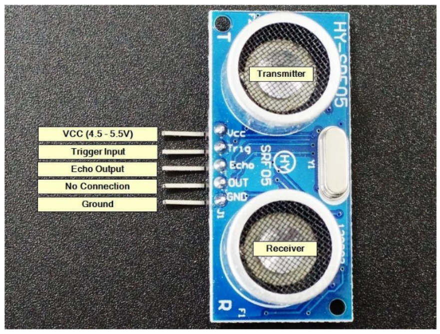
- IR Emitter LED: Emits infrared light.
- IR Receiver (Photodiode): Detects reflected IR signals.
- Distance Adjust (Potentiometer): Sets detection range.
- Power LED: Indicates the module is powered.
- Obstacle LED: Lights up when an obstacle is detected.
- Pins:
  - VCC: Power supply (3.3V - 5V).
  - GND: Ground pin.
  - OUT: Digital output signal.

Characteristics:

- Voltage: 3.3V - 5V.
- Range: Adjustable (2-30 cm).
- Output: Digital (HIGH/LOW based on obstacle presence).

Fig. 6. IR Sensor HW-201

HC-SRF05



Structure

- Transmitter: Emits 40 kHz ultrasonic waves.
- Receiver: Receives the reflected signal (echo).
- Pins:
  - VCC: Provides 5V power supply.
  - GND: Ground pin.
  - Trig: Trigger pin to emit ultrasonic waves.
  - Echo: Receives the reflected signal, with pulse width proportional to distance.
  - No Connection (OUT): Not used (no connection).

Features

- Operating Voltage: 5V.
- Measuring Range: 2 cm - 400 cm.
- Accuracy: ±3 mm.
- Operating Frequency: 40 kHz.
- Distance Calculation: Distance = (Pulse time × 0.034) / 2.

Fig. 7. HC-SRF05

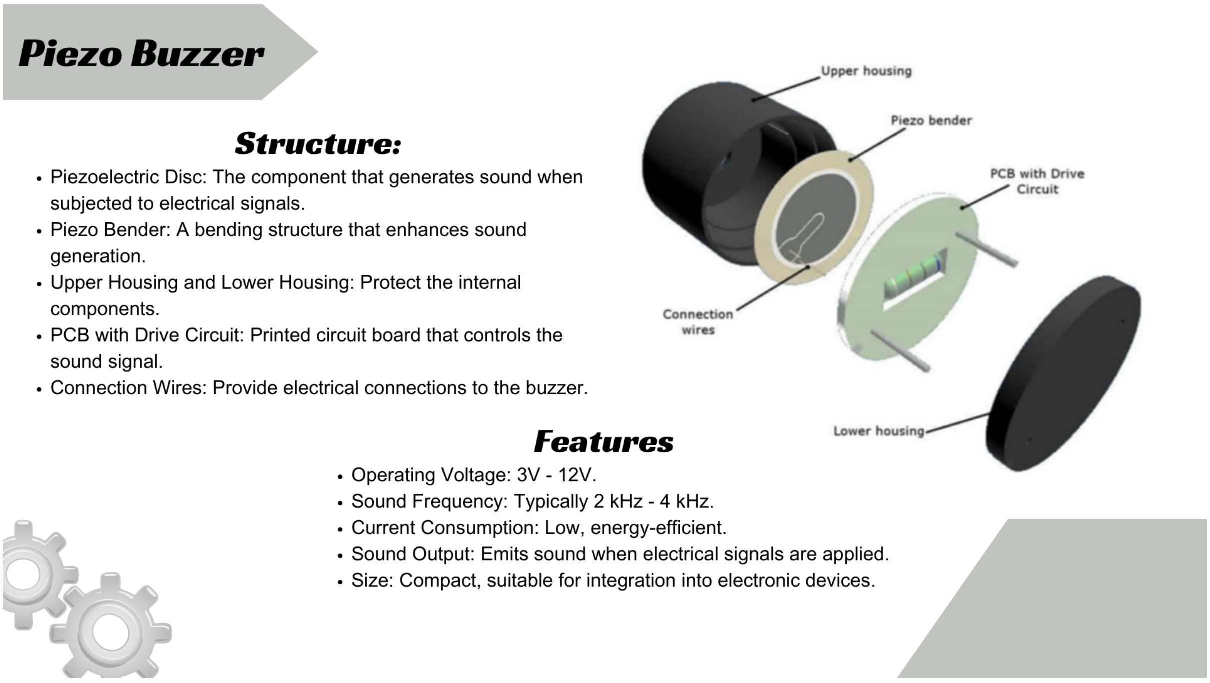


Fig. 8. Buzzer

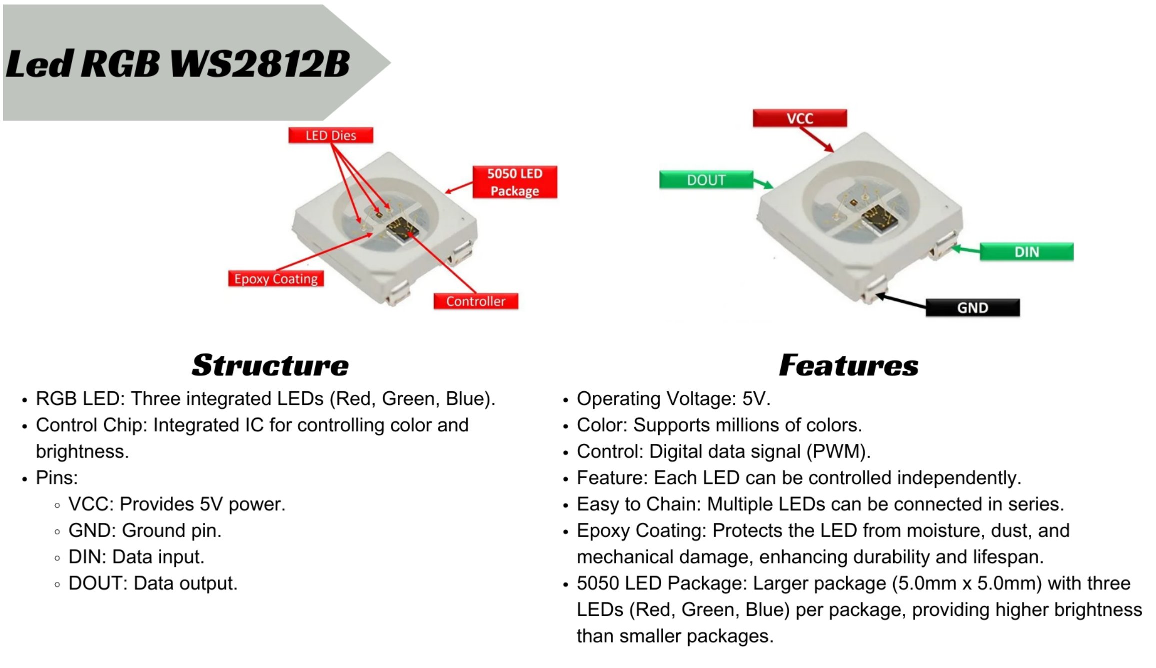


Fig. 9. Led RGB WS28128

C. Circuit Diagram

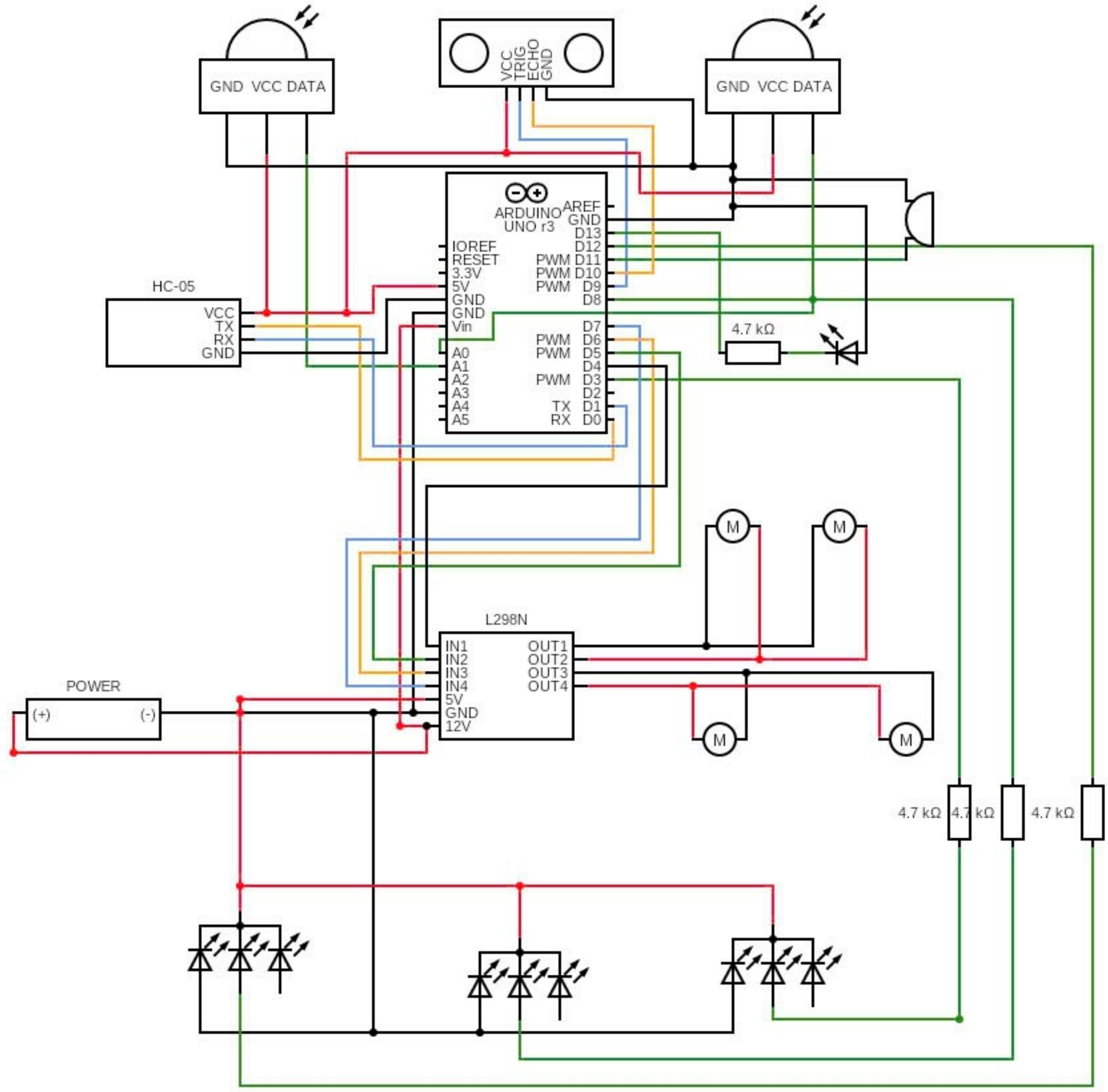


Fig. 10. Circuit Diagram

Arduino Uno	IR SENSOR-1	IR SENSOR-2	MOTOR DRIVER L298N	HC-05	HC-SRF05	WS2812B-LEFT	WS2812B-RIGHT	WS2812B-MID	BUZZER	RED-LED
GND	GND	GND	GND	GND	GND	GND	GND	GND	CATHODE	CATHODE
VCC 5V	VCC	VCC		VCC	VCC	4-7 VDC	4-7 VDC	4-7 VDC		
VCC 3.3V										
VIN			(+)12V							
A0	OUT									
A1		OUT								
D0				TX						
D1				RX						
D2										
D3						DIN				
D4			IN1							
D5			IN2							
D6			IN3							
D7			IN4							
D8								DIN		
D9					TRIG					
D10					ECHO					
D11									ANODE	
D12										ANODE
D13							DIN			

MOTOR DRIVER L298N	ARDUINO UNO	DC-MOTOR LEFT	DC-MOTOR RIGHT	POWER SUPPLY
(+)12V	VIN			(+)
(+)5V				
GND				(-)
OUT1			(+)	
OUT2			(-)	
OUT3		(+)		
OUT4		(-)		
IN1	D4			
IN2	D5			
IN3	D6			
IN4	D7			

D. Programming Flowchart

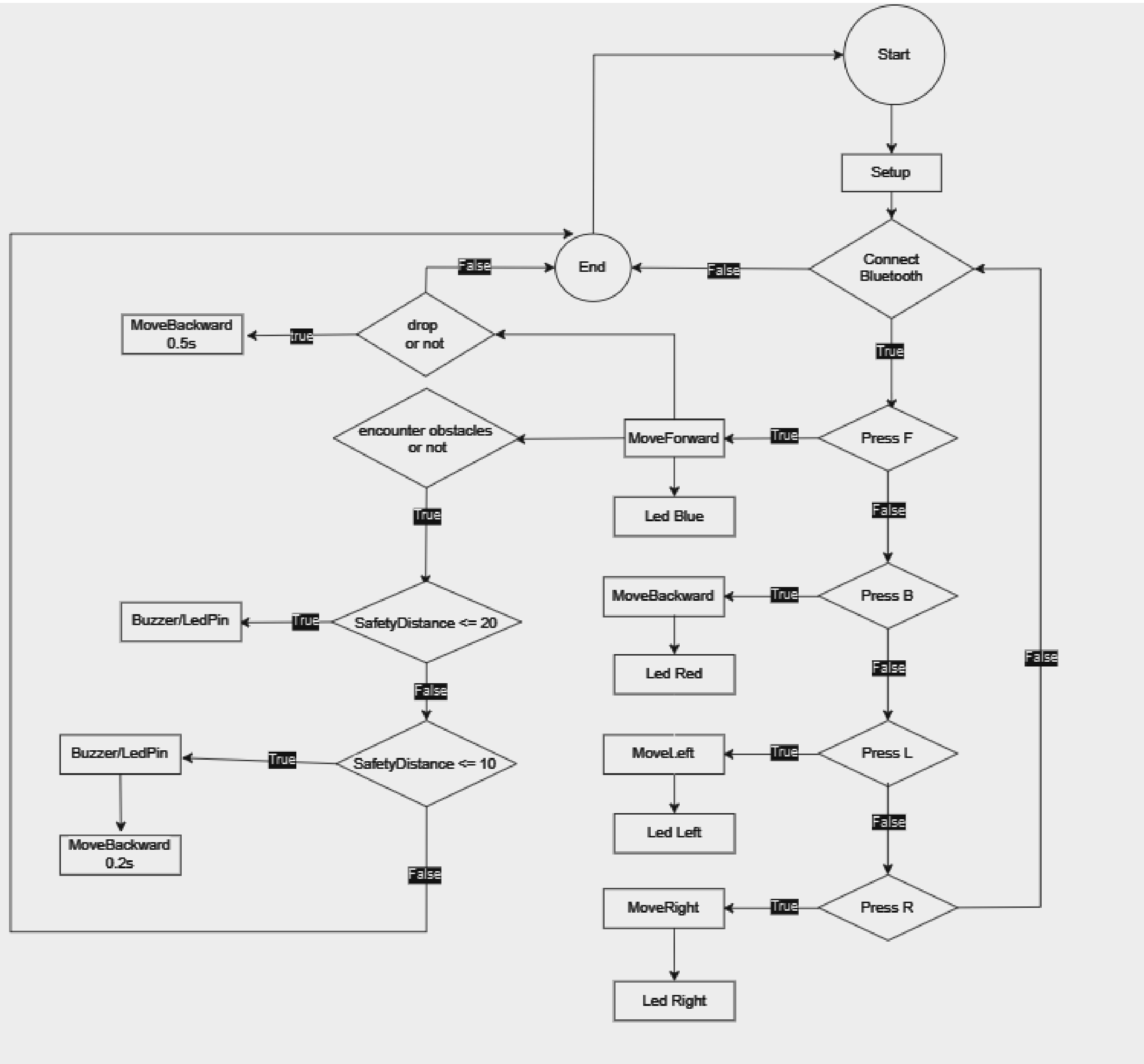


Fig. 11. Flowchart

E. System operation process

The operation process is divided into the following steps:
Receive control commands from the user The user sends commands via the phone via the Bluetooth application. Arduino receives commands from HC-05 and analyzes the content (forward, backward, turn left, turn right or stop). Sensor data processing
Infrared sensor: [3] Monitors obstacles below and automatically reverses when detecting sudden obstacles.
Ultrasonic sensor: [4] Emitting signal: When measuring distance, the ultrasonic sensor will emit high-frequency sound wave pulses in a very short time (microseconds).
Receiving feedback signal: After being emitted, the sound wave will reflect from the obstacle surface (such as walls, hard objects) back to the sensor. The time it takes for the sound wave to travel and return is called the Transmission and Return Time.
Calculating distance: Arduino or other controller will calculate the transmission and return time of the sound wave. The distance to the obstacle can be calculated by the formula:

$$\text{Distance (cm)} = \frac{\text{Time (µs)}}{2} \times \text{Speed of sound (cm/µs)}$$

With the speed of sound in the air being about 340 m/s (34,000 cm/s).
Measure the distance from the vehicle to the obstacle in front. When the distance is less than 20cm, the warning buzzer will be activated. If the obstacle is too close (10cm), the vehicle will stop or activate the reverse command to avoid collision.
Controlling the movement of the vehicle: Arduino sends a signal to the L298N controller to control the motor. The vehicle performs the following movement commands: Forward: The vehicle moves straight forward. Backward: The vehicle moves backward when encountering an obstacle or according to the user's command. Turn left/right: The vehicle changes direction according to the command. **Stop:** The motor stops immediately.
H-Bridge Operation (Using the L298N driver): The H-Bridge uses the L298N to control the vehicle's motor. The Arduino sends a signal to the L298N, which then controls the transistors (or MOSFETs) inside. To drive the motor forward, the Arduino will close transistors A and D, opening transistors B and C, allowing current to flow from A to D. Conversely, to reverse, the Arduino will close transistors B and C, opening transistors A and D, allowing current to flow in the opposite direction through the motor from B to C. Controlling a left or right turn is similar, by changing the state of the transistors on the H-Bridge.
The Arduino's digital pins D4, D5, D6, and D7 correspond to the L298N driver's IN1, IN2, IN3, and IN4 inputs. The direction of rotation of the motor is controlled by setting the appropriate logic HIGH or LOW at these input pins.

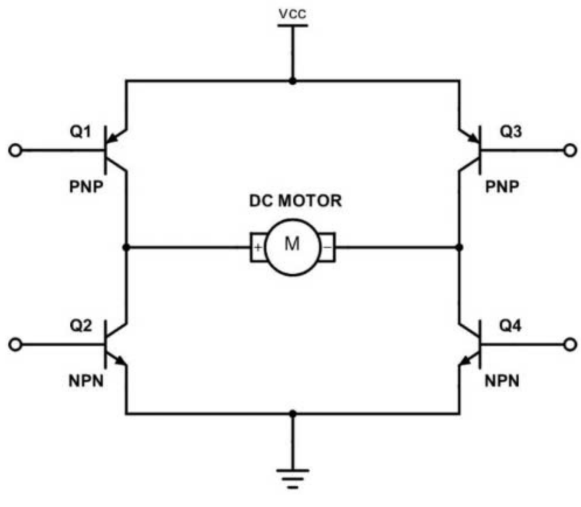


Fig. 12. H-Bridege

MOTOR CONTROL AND LED SIGNAL DISPLAY

Motor Control

- **Motor 1 and 2:**
  - Logic HIGH at `IN1` and LOW at `IN2` will make the motor rotate in one direction.
  - Logic LOW at `IN1` and HIGH at `IN2` will make the motor rotate in the opposite direction.

- **Motor 3 and 4:**
  - Logic HIGH at `IN3` and LOW at `IN4` will make the motor rotate in one direction.
  - Logic LOW at `IN3` and HIGH at `IN4` will make the motor rotate in the opposite direction.

Example: Forward Control

- **Motor A:**
  - `IN1`: Logic HIGH (set Arduino `D4` as OUTPUT)
  - `IN2`: Logic LOW (set Arduino `D5` as OUTPUT)
- **Motor B:**
  - `IN3`: Logic HIGH (set Arduino `D6` as OUTPUT)
  - `IN4`: Logic LOW (set Arduino `D7` as OUTPUT)

LED Signal Display

- **WS2812B LED displays vehicle status:**
  - Turn left/right: The corresponding turn signal LED lights up.
  - Backward: The rear LED lights up to warn.
  - Forward or normal: The LED turns blue.

Continuous Feedback and Updates

The system continuously reads values from sensors and Bluetooth commands to ensure instantaneous response to changes in the environment.

III. Results and Discussion

A. Prototype Implementation

The RC car prototype integrates an Arduino Uno microcontroller, DC motors, HC-05 Bluetooth module, Infrared sensor HW-201 ,LED WS29812B and HC-SRF05 ultrasonic sensors for obstacle detection and wireless communication. The system supports both autonomous and remote control functionalities, using infrared sensors for proximity detection and anti-fall mechanisms to ensure safe operation. The car successfully avoided collisions and prevented falls using sensor feedback, while receiving user commands via Bluetooth for movement control

B. Experimental Results

The RC car prototype was tested in several environments to assess its performance:

- **Collision Avoidance:** The ultrasonic sensors detected obstacles with a response time of approximately 2 seconds.
- **Edge Detection:** The anti-fall mechanism successfully activated when the car approached an edge.
- **Command Response:** Bluetooth communication was stable, with no significant delay in command execution.
- **Battery Life:** The battery lasted approximately 2 hours of continuous use.

C. Discussion

The experimental results validated the RC car's ability to perform basic autonomous functions, including collision avoidance, edge detection, and command response. However, improvements are needed in the following areas:

- **Power Efficiency:** Optimizing energy usage could extend the battery life.
- **Sensor Range:** Ultrasonic and infrared sensors are only available at the front of the vehicle, so obstacles on the sides and behind can still cause danger. Increasing the ultrasonic sensor range would enhance obstacle detection.
- **Durability:**Unstable chassis, loose wiring connections, wheels that do not perform well on smooth surfaces, which may result in failure to brake in time, all of which can result in damage if a severe collision occurs.

Despite these challenges, the prototype demonstrates significant potential for applications in remote surveillance, education,economy, traffic safety and research

IV. Conclusion

This project successfully developed a prototype of an autonomous RC car with obstacle detection, anti-fall mechanisms, and Bluetooth control, demonstrating the feasibility of integrating IoT technologies into practical applications. The research contributes to the development of low-cost, versatile solutions for robotics and IoT research and education

This is an example of a citation. For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation.

V. AUTHOR'S CONTRIBUTION

TABLE I
AUTHOR'S CONTRIBUTION

#	Student ID	Student Name	Tasks	Contribution
1	SE180733	Doan Ly Hong Minh	Leader,code,assembling components	20%
2	SE180734	Ho Dang Quang	Design block diagram,flowchart	20%
3	SE180739	Nguyen Van Bac	Write report, Circuit Diagram	20%
4	SE182754	Phan Quoc Lam	video demo,prepare hardware	20%
5	SE181574	Tran Van Quyet	powerpoint,prepare presentation	20%
Total				100%

REFERENCES

[1] "Bluetooth-controlled car with hc-05 module — arduino project hub," [Online; accessed 2024-12-12]. [Online]. Available: https://projecthub.arduino.cc/angadiameya007/bluetooth-controlled-car-with-hc-05-module-e90493

[2] "Cach dung module dieu kien dong co l298n - cau h de dieu khien dong co dc — cong dong arduino viet nam," 6 2016, [Online; accessed 2024-12-12]. [Online]. Available: http://arduino.vn/bai-viet/893-cach-dung-module-dieu-khien-dong-co-l298n-cau-h-de-dieu-khien-dong-co-dc

[3] "(88) how to make diy arduino table edge avoidance robot - youtube," [Online; accessed 2024-12-12]. [Online]. Available: https://www.youtube.com/watch?v=Tw9qCqcAsP4

[4] "Su dung cam bien khoang cach hc-sr04 — cong dong arduino viet nam," 7 2014, [Online; accessed 2024-12-12]. [Online]. Available: http://arduino.vn/bai-viet/233-su-dung-cam-bien-khoang-cach-hc-sr04

FPT UNIVERSITY IOT102T

BLUETOOTH CONTROLLER CAR PROJECT

Team 4

Members:
Doan Ly Hong Minh – SE180733
Ho Dang Quang – SE180734
Nguyen Van Bac – SE180739
Phan Quoc Lam – SE182754
Tran Van Quyet – SE181574

FPT Education

FPT UNIVERSITY

FPT University, Ho Chi Minh Campus, Vietnam
Email: minhdlhse180733, quanghdse180734, bacnvse180739, lampqse182754, quyettvse181574@fpt.edu.vn