University of Barishal

Lab Report

Course Name: Robotics and Automation Lab

Course Code: CSE-4102

Submitted to:

Dr. Rahat Hossain Faisal
Associate Professor & Chairman
Department of Computer Science and Engineering
University of Barishal

Submitted by:

Name: Shawmitra Das Dwip
Roll: 20CSE033
Semester: 7th Semester
Session: 2019-20
Department of Computer Science and Engineering
University of Barishal

Date: 20 July 2025

Department of Computer Science and Engineering

Faculty of Science and Engineering

University of Barishal

01. Experiment Name: Obstacle Avoiding Robot Using Arduino and Ultrasonic Sensor

1. Objectives

- To design and implement an autonomous robot that can detect and avoid obstacles.
- To use an ultrasonic sensor (HC-SR04) for real-time distance measurement.
- To control motor direction and movement using Arduino and a motor driver.
- To demonstrate basic obstacle avoidance behavior in robotics.

2. Components Used

S.No	Component	Quantity	Description
1	Arduino Uno	1	Microcontroller board to control the project.
2	Ultrasonic Sensor (HC-SR04)	1	Measures distance to detect obstacles.
3	Motor Driver Shield (YFROBOT or L298)	1	Controls motor speed and direction.
4	Geared DC Motors	2	Provides movement with torque.
5	Wheels	2	Attached to motors for movement.
6	Chassis	1	Frame to hold all parts.
7	Battery Pack (6V–9V)	1	Powers the system.
8	Jumper Wires	As needed	Connects components electrically.
9	Breadboard (optional)	1	For circuit prototyping.

3. Block Diagram / Circuit Diagram

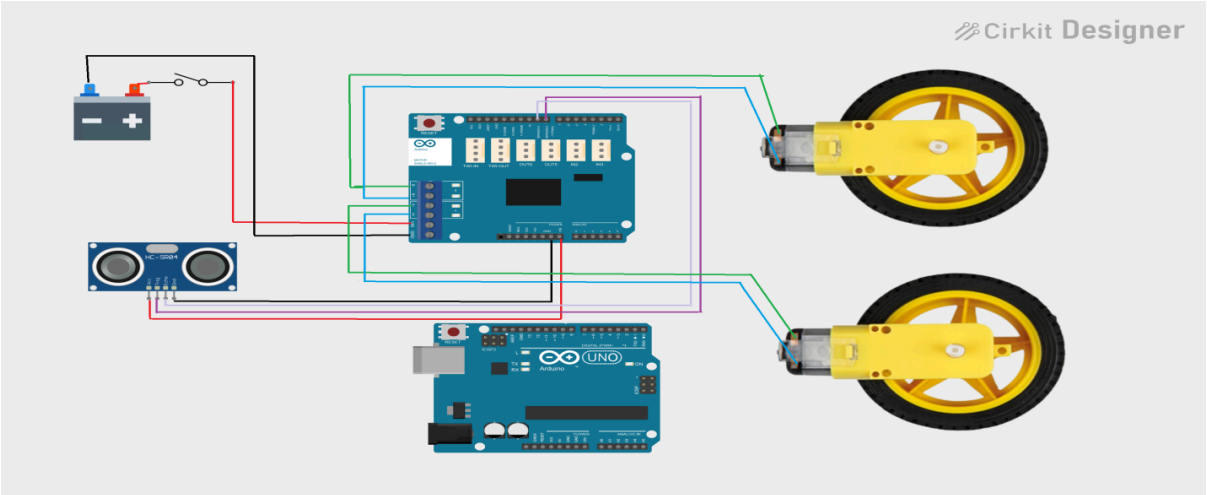

Figure: Circuit diagram of Obstacle Avoiding Robot

The circuit connections are as follows:

- Ultrasonic Sensor (HC-SR04):
  - Trig pin connected to Arduino pin A8.
  - Echo pin connected to Arduino pin A9.
  - VCC and GND connected to 5V and GND, respectively.
- YFRobot Motor Shield:
  - Stacked on top of the Arduino UNO.
  - Motors connected to ports: M1 (Front Left), M2 (Rear Left), M3 (Rear Right), M4 (Front Right).
- Power Supply:
  - A 7.4V battery pack powers the motors through the motor shield.
  - Arduino powered via USB or Vin pin.
  - All component grounds are connected together.

4. Flowchart / Algorithm

4.1 Flowchart:

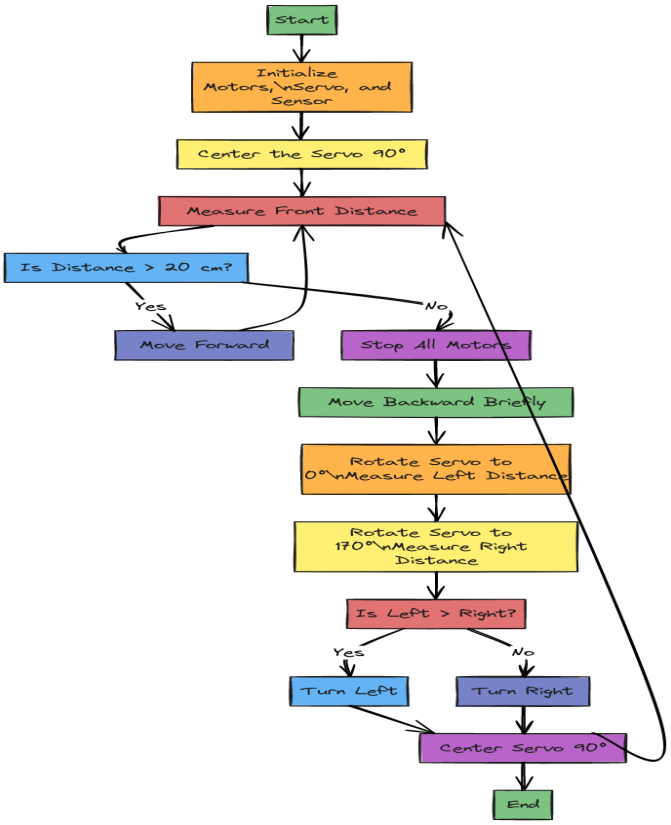

Figure: flow chart of algorithms

4.2 Algorithm:

1. Start the Arduino(upload code) and initialize all pins.
2. Move the robot forward.
3. Continuously read distance from the ultrasonic sensor.
4. If distance > 20 cm, continue moving forward.
5. If obstacle detected:

    i. Stop motors
    ii. Check distance on the left
    iii. Check distance on the right
    iv. Compare both sides
    v. Turn in the direction with more distance

6. Resume forward motion.
7. Repeat.

5. Methodology

- The robot uses the HC-SR04 ultrasonic sensor to measure distance from obstacles in front.
- The Arduino Uno controls the motors using a motor driver shield based on distance readings.
- If no obstacle is detected within 20 cm, the robot moves forward.
- If an obstacle is detected:
  The robot stops, turns left, and measures, then turns right and measures.
  It compares both distances and chooses the safest (longest) path to avoid collision.
- This method ensures basic autonomous navigation.

6. Output

The robot moves independently and avoids obstacles in its path. Demonstrates turning behavior to change direction automatically.

7. Result

- The obstacle avoiding robot successfully detects objects in its path.
- It stops and makes decisions based on distance readings.
- The car selects a safer path (left or right) and continues motion without manual input.
- The robot met the expected behavior of basic autonomous obstacle avoidance.

8. Conclusion

The obstacle avoiding robot project demonstrates how sensors, logic, and actuators can be integrated to create autonomous machines. By using an ultrasonic sensor and a simple decision-making algorithm, the robot was able to navigate safely without human control. This experiment lays the foundation for more advanced autonomous systems used in real-world robotics applications.

2. Experiment Name: Voice Controlled Car with Obstacle Detection Using Arduino

1. Objectives

- To control a robot car using voice commands via Bluetooth.
- To detect obstacles using an ultrasonic sensor and avoid collisions.
- To integrate manual (voice) control and automatic safety (obstacle detection).
- To demonstrate communication between smartphone and microcontroller.

2. Components Used

S.No	Component	Quantity
1	Arduino Uno	1
2	HW-03 Bluetooth Module	1
3	HC-SR04 Ultrasonic Sensor	1
4	YFROBOT Motor Shield	1
5	DC Geared Motors (2x)	2
6	Smartphone (with voice command app)	1
7	Battery Pack (6–9V)	1
8	Jumper Wires	As needed
9	Chassis + Wheels	1 set

4. Block Diagram / Circuit Diagram

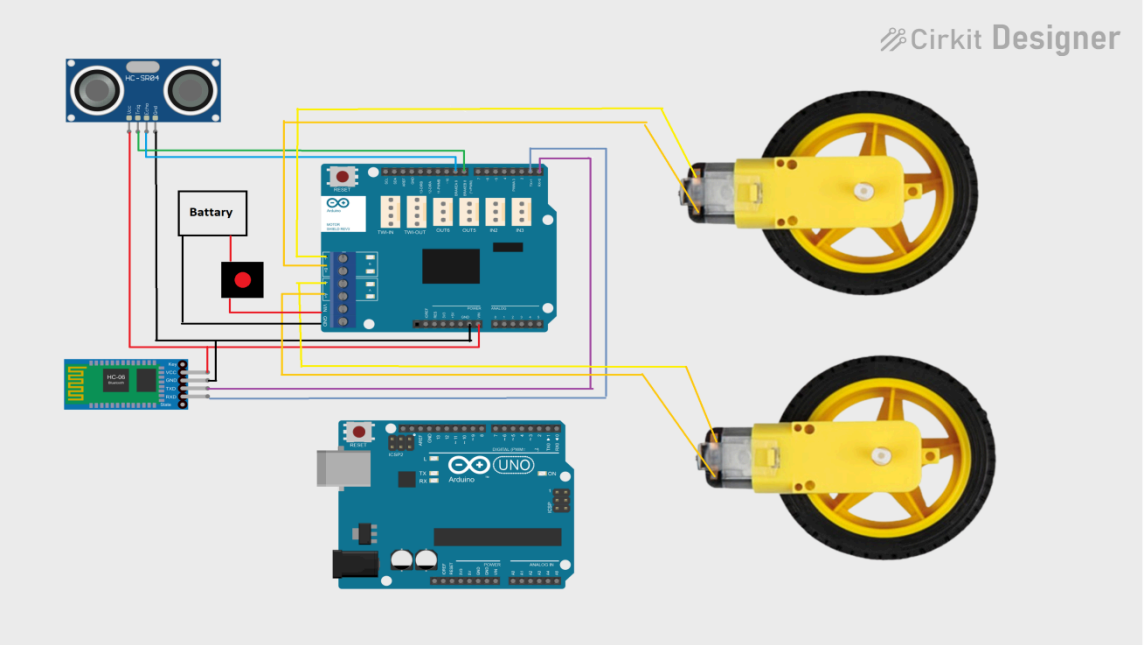

Figure: Circuit diagram of Voice Controlled Car

DC Motors to Motor Driver Shield

Motor A (Right wheel) connected to:
OUT1 and OUT2 (yellow wires)
Motor B (Left wheel) connected to:
OUT3 and OUT4 (yellow wires)

Motor Driver Shield to Arduino UNO

The shield is mounted directly onto the Arduino UNO.

Pin on HC-SR04	**Wire Color**	**Connects To sheild**
VCC	Red	5V
GND	Black	GND
TRIG	Green	D8
ECHO	Blue	D9
Pin on HC-06	**Wire Color**	**Connects To sheild**
VCC	Red	5V
GND	Black	GND
TXD	Purple	RX (D0)
RXD	Grey	TX (D1)

5. Flowchart / Algorithm

Flowchart:

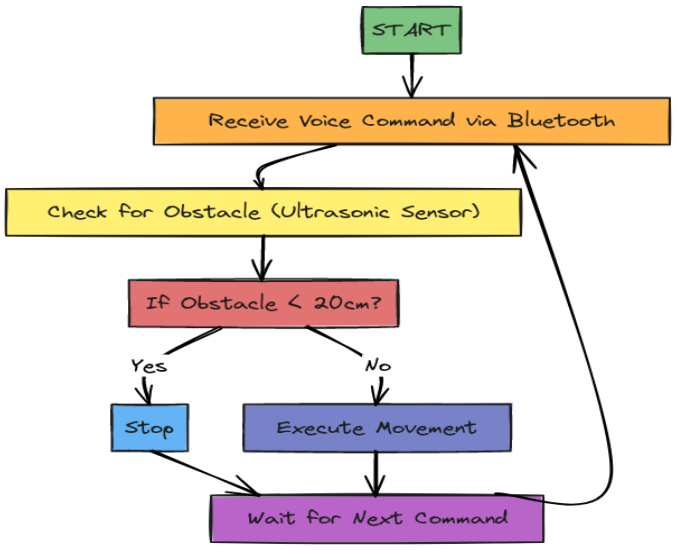

Figure: flowchart diagram of Voice Controlled Car algorithm

Algorithm:

1. Initialize all pins and modules (Bluetooth, Motor Shield, Ultrasonic).
2. Wait for a voice command from the Bluetooth module.

3. When a command is received (forward, backward, left, right, stop):
    i. Check the distance using the ultrasonic sensor.
    ii. If distance < 20 cm and command is "forward", ignore and stop.
    iii. Otherwise, execute the command using the motor shield.
4. Repeat the loop.

6. Methodology

- The user speaks a command like “forward” or “stop” using a Bluetooth voice control app.
- The HW-03 module receives the text-based command and sends it to Arduino.
- Arduino checks for any nearby obstacle using the ultrasonic sensor.
- If the path is clear, Arduino uses the YFROBOT motor shield to drive the motors.
- If an obstacle is detected too close, Arduino ignores movement and waits for the next command.
- The car responds to forward, backward, left, right, and stop.

7. Output

- The car moves in the direction commanded via voice.
- It automatically stops if an obstacle is detected in front.

8. Result

- The voice-controlled robot successfully receives commands via HW-03 Bluetooth.
- It responds correctly to “forward”, “stop”, “backward”, “left”, “right”.
- When an obstacle is detected < 20 cm in front, the robot prevents forward motion, ensuring safety.
- The integration of manual voice control + automatic obstacle handling works effectively.

9. Conclusion

This project demonstrates a hybrid control system where voice input via Bluetooth and autonomous obstacle detection are combined. The robot can be manually directed while still being aware of its surroundings using an ultrasonic sensor. This model is foundational for developing smart vehicles, wheelchair automation, or assistive robotics

3. Experiment Name : Line Following Robot Using Arduino UNO and IR Sensors

1. Objectives

- To design and build a robot that follows a black line autonomously using IR sensors.
- To understand the use of sensors, microcontrollers, and motor drivers in robotics.
- To implement real-time decision-making logic using Arduino based on sensor input.

2. Components Used

S.No	Component	Quantity	Description
1	Arduino UNO	1	Microcontroller board that acts as the brain of the robot.
2	IR Sensor Modules	2	Used to detect black line on white surface. Placed on the underside front.
3	L293D Motor Driver Module	1	Dual H-Bridge motor driver to control direction and speed of DC motors.
4	DC Motors (with wheels)	2	TT gear motors with wheels for movement of the robot.
5	7–12V DC Power Supply	1	Powers the motors and Arduino (usually via battery pack).
6	Switch	1	provides on/off control for power.
7	Connecting Wires	As needed	Jumper wires or male-female wires for all electrical connections.
8	Robot Chassis	1	Base structure that holds all components

3.Block Diagram / Circuit Diagram

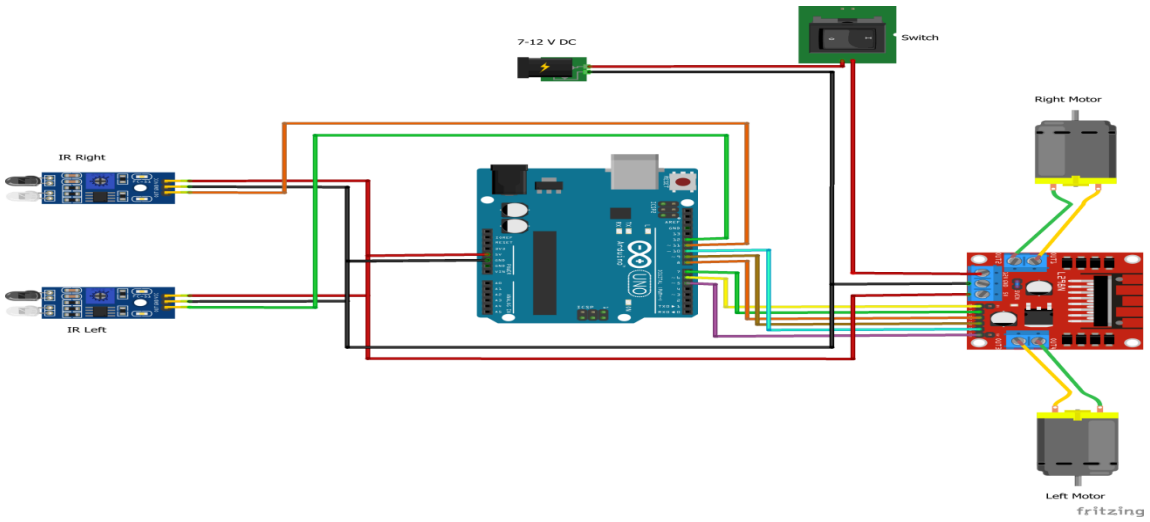

Figure: Circuit diagram of line following robot

Below is the circuit connection summary (see attached image for full diagram):

- IR Left Sensor OUT → D11
- IR Right Sensor OUT → D12
- L293D IN1, IN2 → D7, D8 Arduino
- L293D IN3, IN4 → D9, D10
- L293D ENA, ENB → D5, D6
- Shared GND and 5V connections
- Motors connected to L293D outputs (Left: OUT1/OUT2, Right: OUT3/OUT4)

Note: The motor driver is powered by the battery, while Arduino handles logic control.

4/Flowchart / Algorithm
Flow chart is in the following:

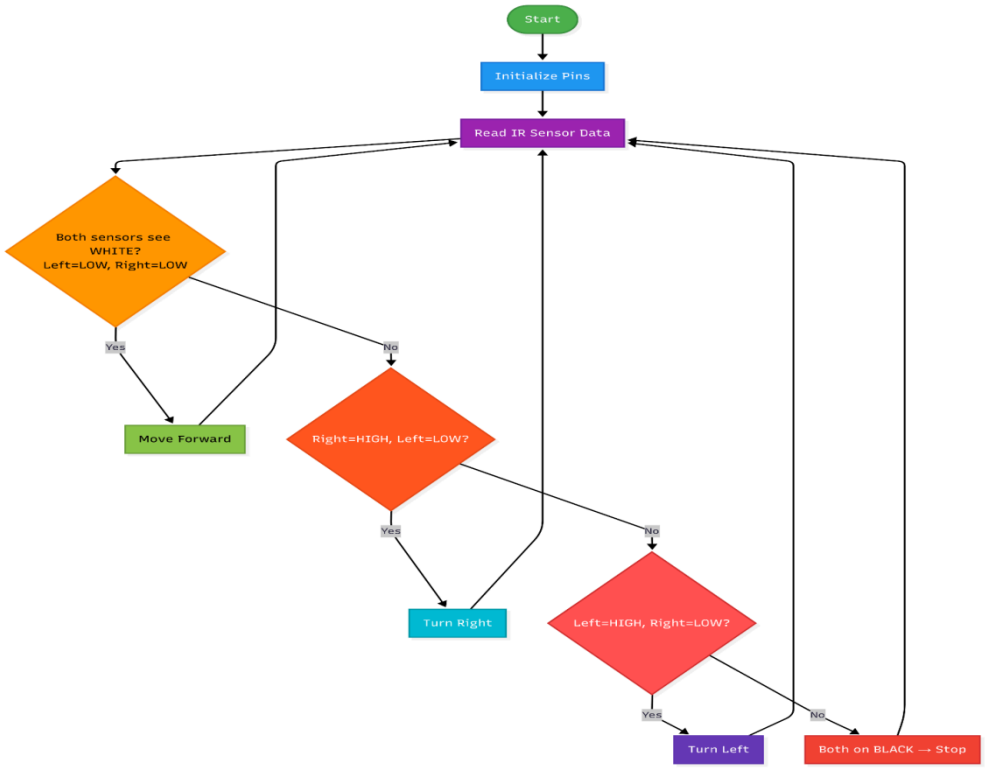

Figure: Flow chart of Line following Robot

Algorithms:

IR Sensors detect black/white surface:

- Black → HIGH signal (1)
- White → LOW signal (0)

Arduino reads the sensor values:

- If both sensors detect white, move forward
- If left detects black, turn right
- If right detects black, turn left
- If both detect black, stop

Based on this logic, Arduino sends HIGH/LOW signals to IN1–IN4 and controls motor movement using L293D.

5. Methodology

1. Assemble the chassis and fix the motors, IR sensors, and Arduino.
2. Wire the circuit as per the diagram (Arduino, sensors, motors, driver, power).
3. Upload Arduino code that reads sensor input and drives motors accordingly.
4. Place the robot on a white surface with a black line.
5. Power ON the system using the switch and test line-following behavior.
6. Adjust sensor alignment or code if tracking fails.

6. Output

When powered on:

- The robot starts moving and follows a black line on a white background.
- It can turn left or right based on sensor feedback.
- Stops when both sensors are over the black line (end of path or obstacle).

7. Result

- The robot successfully detects and follows a line using IR sensors.
- It responds to curves and turns in the line path with smooth motion.
- No major hardware failures were observed; behavior was consistent.

8. Conclusion

This project demonstrates the basics of autonomous robotics using simple electronic components and Arduino programming. The robot efficiently follows a black line using two IR sensors and adjusts its path using motor control logic. This principle can be expanded into more complex applications such as maze solvers, delivery bots, or path-based industrial automation.

4. Experiment Name: bluetooth Controlled Car Using Arduino

1. Objectives

- To control a robot car using voice commands via Bluetooth.
- To detect obstacles using an ultrasonic sensor and avoid collisions.
- To integrate manual (voice) control and automatic safety (obstacle detection).
- To demonstrate communication between smartphone and microcontroller.

2. Components Used

S.No	Component	Quantity
1	Arduino Uno	1
2	HW-03 Bluetooth Module	1
3	HC-SR04 Ultrasonic Sensor	1
4	YFROBOT Motor Shield	1
5	DC Geared Motors (2x)	2
6	Smartphone (command app)	1
7	Battery Pack (6–9V)	1
8	Jumper Wires	As needed
9	Chassis + Wheels	1 set

3. Block Diagram / Circuit Diagram

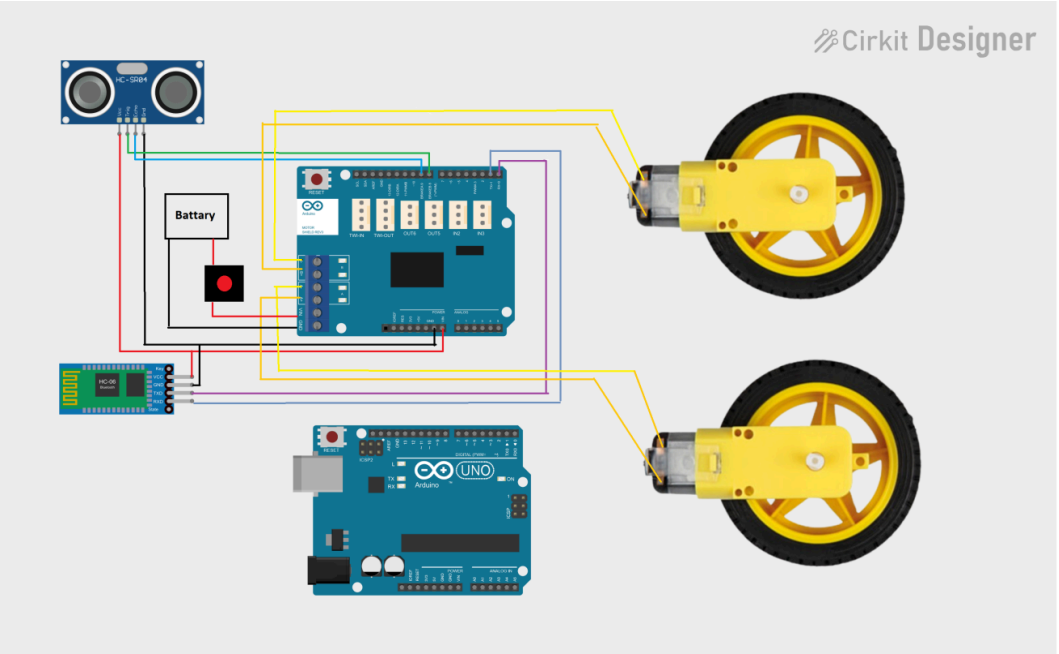

Figure: Circuit diagram of Bluetooth control car

4. Flowchart / Algorithm

4.1 Flowchart:

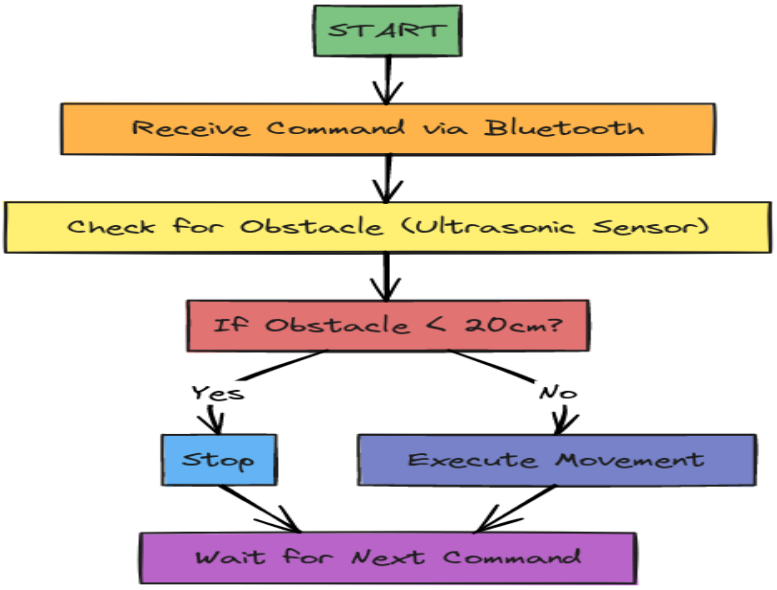

Figure: Flowchart diagram of Bluetooth control car algorithm

4.2 Algorithm:

5. Initialize all pins and modules (Bluetooth, Motor Shield, Ultrasonic).
6. Wait for a voice command from the Bluetooth module.
7. When a command is received (forward, backward, left, right, stop):
    iv. Check the distance using the ultrasonic sensor.
    v. If distance < 20 cm and command is "forward", ignore and stop.
    vi. Otherwise, execute the command using the motor shield.
8. Repeat the loop.

5. Methodology

- The user speaks a command like “forward” or “stop” using a Bluetooth voice control app.
- The HW-03 module receives the text-based command and sends it to Arduino.
- Arduino checks for any nearby obstacle using the ultrasonic sensor.
- If the path is clear, Arduino uses the YFROBOT motor shield to drive the motors.
- If an obstacle is detected too close, Arduino ignores movement and waits for the next command.

- The car responds to forward, backward, left, right, and stop.

6. Output

- The car moves in the direction commanded via voice.
- It automatically stops if an obstacle is detected in front.

7. Result

- The voice-controlled robot successfully receives commands via HW-03 Bluetooth.
- It responds correctly to “forward”, “stop”, “backward”, “left”, “right”.
- When an obstacle is detected < 20 cm in front, the robot prevents forward motion, ensuring safety.
- The integration of manual voice control + automatic obstacle handling works effectively.

8. Conclusion

This project demonstrates a hybrid control system where voice input via Bluetooth and autonomous obstacle detection are combined. The robot can be manually directed while still being aware of its surroundings using an ultrasonic sensor. This model is foundational for developing smart vehicles, wheelchair automation, or assistive robotics.