

BLUETOOTH - CONTROLLED CAR USING 8051 MICROCONTROLLER

## ( PROJECT REPORT)

Submitted by

## BHUPATI RAM KUMAR K – 22BEC1452 ADITYA KUMAR JHA – 22BEC1074

MUKHIL LAKSHMANAN – 22BEC1005 ASHWIN KARTHIKEYAN – 22BEC1149 ALRIC SAM – 22BEC1254

ADITHYA HARI V – 22BEC1206 LEE THATHEYUS RALPH – 22BEC1040

1

# CERTIFICATE

This is to certify that the Project work titled “ **BLUETOOTH - CONTROLLED CAR USING 8051 MICROCONTROLLER**” is being submitted by BHUPATI RAM KUMAR K – 22BEC1452 , ADITYA KUMAR JHA – 22BEC1074 , MUKHIL LAKSHMANAN – 22BEC1005 , ASHWIN KARTHIKEYAN - 22BEC1149, ALRIC SAM – 22BEC1254, LEE THATHEYUS RALPH AR – 22BEC1040 , ADITHYA HARI V – 22BEC1206 for the

course **Microprocessors and Microcontrollers** is a record of bonafide work done under my guidance. The contents of this project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University.

# Dr.Sourabh Paul

Project Guide

2

# ABSTRACT

This project aims to develop a Bluetooth-controlled car utilizing the 8051

microcontroller, with a mobile phone serving as the remote control unit. With the widespread availability of smartphones and Bluetooth technology, this system offers a convenient and intuitive interface for remotely controlling the car's movements. Mobile phone serves as medium of transferring commands from user side to the proposed system.

The remote control unit is a smartphone equipped with a custom-designed mobile application. The application establishes a Bluetooth connection with

the car, enabling bidirectional communication between the phone and the car's onboard 8051 microcontroller. Users can control the car's direction by sending commands through the mobile application's user interface.

On the car side, another Bluetooth module is interfaced with the 8051

microcontroller, allowing it to receive control signals from the smartphone. The microcontroller processes these signals and activates the appropriate motors or actuators to execute the desired actions, such as moving forward, backward , turning left or turning right

Key features of the system include:

1. Smartphone-based remote control interface, providing users with a familiar and accessible means of controlling the car.
2. Bluetooth connectivity for wireless communication, enabling untethered operation and increased mobility.
3. Real-time responsiveness, ensuring immediate feedback and smooth execution of control commands.
4. Expandability and customization options, allowing for the integration of additional sensors or functionalities to enhance the car's capabilities.

The proposed Bluetooth-controlled car offers a versatile platform for exploring concepts in embedded systems, microcontroller programming, and mobile

application development. It not only serves as an engaging educational tool but also holds potential for practical applications in areas such as robotics, automation, and entertainment.

4

# ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide**,** for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to **Dr. Susan Elias,** Dean of School of Electronics Engineering, VIT , for extending the facilities of the School towards our project and for his unstinting support.

We express our thanks to our Head of the Department **Dr. K. Mohanaprasad** for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

5

|  |  |  |
| --- | --- | --- |
| **SI.NO** | **CONTENTS** | **PAGE.NO** |
| 1 | Certificate | 1 |
| 2 | Abstract | 2-3 |
| 3 | Acknowledgement | 4 |
| 4 | Project Description | 6-13 |
| 5 | Algorithm | 14 |
| 6 | Hardware Model | 18 |
| 7 | Biodata | 21 |

# TABLE OF CONTENTS

6

**PROJECT DESCRIPTION**

## INTRODUCTION

In the realm of embedded systems and robotics, the integration of

microcontrollers and wireless communication technologies has opened up a plethora of possibilities. One such exciting application is the

development of a Bluetooth-Controlled Car, where the mobility of a vehicle is governed wirelessly through Bluetooth connectivity. In this

project, we employ the venerable 8051 microcontroller, a stalwart in the field of embedded systems, to orchestrate the movements of our car.

The Bluetooth-Controlled Car is a miniature yet robust embodiment of the fusion between hardware and software engineering. At its core lies the 8051 microcontroller, renowned for its versatility and reliability,

serving as the brain of our vehicle. Paired with Bluetooth technology, our car transcends the limitations of traditional wired control mechanisms, offering a seamless and intuitive means of maneuvering.

7

The primary objective of this project is to design and implement a Bluetooth-Controlled Car prototype using the 8051 microcontroller.

Through this endeavour, we aim to showcase the integration of hardware components, software algorithms, and wireless communication protocols to realize a fully functional and remotely controllable vehicle.

By leveraging Bluetooth connectivity, we enhance the user experience by providing a convenient and wire-free interface for commanding the car's movements.

The project encompasses both hardware and software components intricately intertwined to achieve seamless functionality. Hardware

elements include the chassis of the car, motor drivers, wheels, Bluetooth module, and other auxiliary peripherals. On the software front, we delve into firmware development for the 8051 microcontroller, crafting

algorithms for interpreting Bluetooth commands, orchestrating motor movements, and managing peripheral operations.

## PURPOSE

The purpose of the Bluetooth-controlled car project using the 8051

microcontroller is to demonstrate a wireless remote control system for a car. Utilizing Bluetooth technology, the project aims to provide a

convenient and intuitive interface for users to control the car's

movements remotely using a smartphone or similar Bluetooth-enabled device. This project serves educational, practical, and innovative

purposes by showcasing embedded systems, microcontroller

8

programming, wireless communication, and remote control concepts. It also encourages exploration and experimentation in robotics, automation, and entertainment applications. Overall, the project aims to create an engaging and functional platform for learning and

experimentation.

## SCOPE

The Bluetooth-controlled car project with the 8051 microcontroller

encompasses hardware design, microcontroller firmware development for Bluetooth signal interpretation, establishing wireless communication, rigorous testing, and comprehensive documentation. It involves selecting and integrating car components, programming firmware to interpret

Bluetooth commands, configuring Bluetooth modules, and ensuring system reliability through thorough testing. Documentation includes

circuit diagrams, firmware code, testing procedures, and user manuals. Optional integration with existing mobile apps for control can be explored. Through these tasks, the project aims to deliver a fully functional, reliable, and user-friendly remote control system, facilitating learning in embedded systems, microcontroller programming, and

wireless communication within resource constraints.

9

* 1. PROCESS EXPLAINED

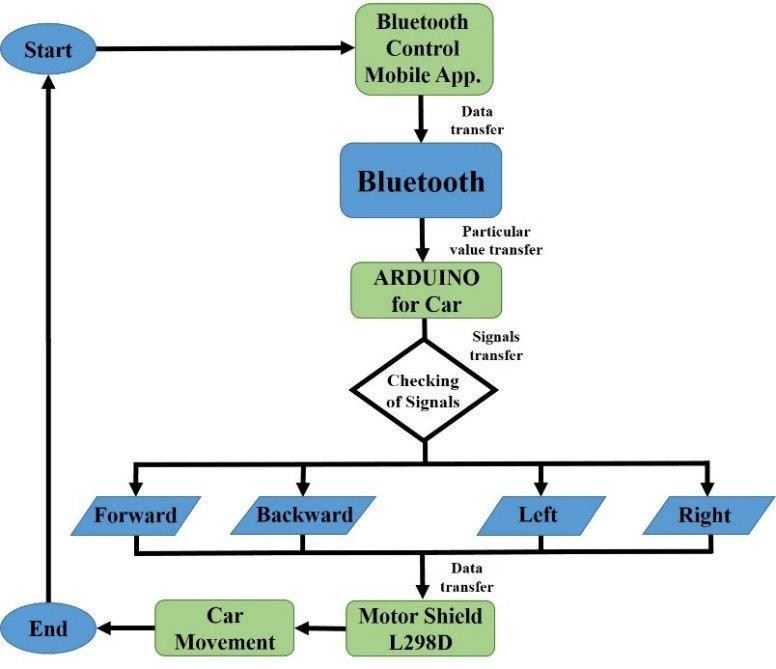
# Initial Configuration Setup

To begin the operation of our Bluetooth-Controlled Car, we first set up the initial configuration. This involves configuring Port 2 of the 8051 microcontroller as output to control the motors and initializing serial communication to establish a connection with a Bluetooth-enabled device.

Upon power-up, the microcontroller initializes the necessary pins for motor control and sets up the serial communication interface. This

ensures that the car is ready to receive commands wirelessly via Bluetooth.

# Main Loop Execution

Once the initial configuration is complete, the microcontroller enters the main loop, where it continuously monitors for incoming commands from the Bluetooth module. The main loop operates as follows:

10

# Reading Incoming Characters

The microcontroller reads incoming characters from the serial communication interface. These characters represent commands sent

from a paired Bluetooth device, instructing the car on various actions to perform.

# Command Interpretation

After receiving a character, the microcontroller checks the command to determine the corresponding action:

i. If 'F' is received, the car is instructed to move forward. ii. If 'B' is received, the car is instructed to move backward. iii. If 'L' is

received, the car is instructed to turn left. iv. If 'R' is received, the car is instructed to turn right. v. If 'S' is received, all movements are stopped.

# Implementing Corresponding Actions

Once the received command is identified, the microcontroller executes the corresponding action. For example, if the command 'F' is received, the microcontroller activates the motors to move the car forward.

# Serial Communication Initialization

The serial communication initialization procedure ensures that the microcontroller is configured to communicate with the Bluetooth module. This involves setting the baud rate and configuring the serial port parameters to match those of the Bluetooth module.

# Serial Write Procedure

The serial write procedure is responsible for sending data from the microcontroller to the Bluetooth module. This is used when the

microcontroller needs to transmit data or feedback to the paired

Bluetooth device, such as acknowledgment messages or sensor readings.

11

# Serial Send Procedure

The serial send procedure sends data byte by byte from the microcontroller to the Bluetooth module. This ensures reliable

transmission of data over the serial communication interface, allowing for seamless communication between the microcontroller and the

Bluetooth device.

# Serial Read Procedure

The serial read procedure is responsible for receiving data from the

Bluetooth module. It continuously monitors the serial communication

interface for incoming data and processes it accordingly. This allows the microcontroller to receive commands and instructions from the paired Bluetooth device, enabling remote control of the car.

In summary, the Bluetooth-Controlled Car operates by continuously monitoring for commands from a paired Bluetooth device, interpreting these commands, and executing corresponding actions to control the movement and lighting of the car. The integration of the 8051

microcontroller and Bluetooth technology enables wireless communication and remote control, making for an engaging and versatile robotics project.

## COMPONENTS USED

* Motor controller - L298N
* Micro controller 8051
* Bluetooth module - HC05
* Dc motor
* Wheels
* Battery

12

* 1. COMPONENTS ANALYSIS:

# MOTOR CONTROLLER - L298N

1. **Motor Control:** L298N controls DC motor direction for precise movement in the Bluetooth-controlled robot project.
2. **Microcontroller Interface:** 8051 microcontroller communicates with L298N, sending signals to control motor behavior effectively.
3. **H-Bridge Configuration:** L298N's H-bridge design facilitates bi-

directional motor control, enabling forward, backward, and turning motions.

1. **Current Protection:** L298N incorporates safeguards to prevent motor damage, enhancing the longevity and reliability of the robot system.

# 2. MICRO CONTROLLER 8051

1. **Bluetooth Communication:** Utilize 8051 for establishing communication with a smartphone to control the robot via Bluetooth module.
2. **Motor Control:** Interface 8051 with motor drivers to manage the movement of the robot wheels.
3. **Data Processing:** Employ 8051 to process incoming data from Bluetooth and sensors for robot operation.

# 3. BLUETOOTH MODULE - HC05

13

1. **Wireless Communication:** HC-05 enables Bluetooth communication between the robot and external devices like smartphones or tablets.
2. **Serial Interface:** It interfaces with the microcontroller (8051) via UART for transmitting and receiving data wirelessly.
3. **Pairing Capability:** HC-05 can pair with Bluetooth-enabled devices, facilitating remote control or data exchange.
4. **Configurable Modes:** It supports both master and slave modes, offering flexibility in establishing connections with other Bluetooth devices.
5. **Low Power Consumption:** HC-05 operates efficiently, conserving power in the robot system for extended operation periods.

# 3. DC MOTOR

1. Direction Control: DC motors can rotate clockwise or

counterclockwise by changing the polarity of the applied voltage.

1. Speed Regulation: The speed of DC motors is adjustable through voltage variation or PWM techniques, offering precise control.
2. Simple Construction: DC motors have a straightforward design with a stator, rotor, and brushes, making them cost-effective and versatile.

# WHEELS

Wheel movement enables a robot to traverse across surfaces, providing mobility and allowing it to navigate its environment effectively.

# BATTERY

14

Batteries provide the necessary electrical power to operate the robot, supplying energy for its components and enabling autonomous functionality.

## ALGORITHMN:

The operational logic of our Bluetooth-Controlled Car hinges on an algorithmic framework meticulously crafted to interpret incoming

commands from a Bluetooth-enabled device. Upon receiving instructions wirelessly, the 8051 microcontroller deciphers the transmitted characters and executes corresponding actions, such as moving forward, backward, left or right. This algorithm forms the cornerstone of our project,

ensuring precise and responsive control over the car's locomotion.

* + 1. Set up initial configuration: Set Port 2 as output, initialize serial communication.
    2. Enter the main loop:
       1. Read incoming character from serial communication.
       2. Check the received character:
          1. If 'F' received, move forward.
          2. If 'B' received, move backward.
          3. If 'L' received, turn left.
          4. If 'R' received, turn right.
          5. If 'S' received, stop all movements.
    3. Implement corresponding actions based on the received characters.
    4. Define serial communication initialization procedure.
    5. Define serial write procedure to send data.

15

* + 1. Define serial send procedure to send data byte by byte.
    2. Define serial read procedure to receive data.

## CONCLUSIONS:

The Bluetooth-Controlled Car project serves as a testament to the convergence of cutting-edge technologies and practical applications in

the realm of embedded systems and robotics. Through the integration of the venerable 8051 microcontroller and Bluetooth connectivity, we have created a versatile and engaging platform for remote control of a

miniature vehicle.

Throughout the course of this project, we have achieved several milestones. Firstly, we successfully designed and implemented the

hardware architecture of the car, including motor drivers, wheels, and the Bluetooth module. Additionally, our firmware development efforts resulted in the creation of robust algorithms for interpreting Bluetooth commands and orchestrating precise movements of the vehicle. By

leveraging the power of the 8051 microcontroller, we have

demonstrated the feasibility of wirelessly controlling a mobile robot with agility and precision.

The project has provided invaluable learning experiences for all involved. We gained hands-on exposure to microcontroller programming,

16

Bluetooth communication protocols, motor control mechanisms, and circuit design principles. Through troubleshooting challenges and

iteratively refining our solutions, we have honed our problem-solving skills and fostered a deeper understanding of embedded systems

development.

## FUTURE APPLICATIONS:

* + 1. Autonomous Navigation: Integrating sensors such as ultrasonic or

infrared sensors can enable the car to detect obstacles and navigate autonomously. Implementing algorithms for obstacle avoidance and path planning would enhance the car's capabilities, making it suitable for use in environments where manual control is impractical or unsafe.

* + 1. Multi-Agent Coordination: Building multiple Bluetooth-Controlled Cars and implementing coordination algorithms would enable collaborative tasks and swarm robotics applications. These cars could work together to accomplish complex tasks such as exploration, surveillance, or search and rescue operations.
    2. Machine Learning Integration: Leveraging machine learning algorithms for behaviour prediction and decision-making could enhance the car's autonomy and adaptability. By analysing sensor data and historical usage

17

patterns, the car could learn to anticipate user commands and optimize its behaviour accordingly.

* + 1. Environmental Sensing: Equipping the car with environmental sensors such as temperature, humidity, and air quality sensors could enable

environmental monitoring applications. The car could be deployed in

indoor or outdoor environments to collect data for research, monitoring, or environmental assessment purposes.

## REFERENCES:

1. Miller, Peter. "Bluetooth Wireless Technology: A Technical Overview." IEEE Communications Magazine, vol. 39, no. 12, 2001, pp. 86-94.
2. Ayala, Kenneth J. \*The 8051 Microcontroller\*. Cengage Learning, 1999.
3. Agarwal, Dhananjay V., and Neeraj K. Gupta. \*Bluetooth: Principles, Protocols, and Applications\*. Prentice Hall, 2003.
4. Tanenbaum, Andrew S., and David J. Wetherall. \*Computer Networks\*. Pearson, 2011.
5. "HC-05 Bluetooth Module Datasheet." Elecrow.

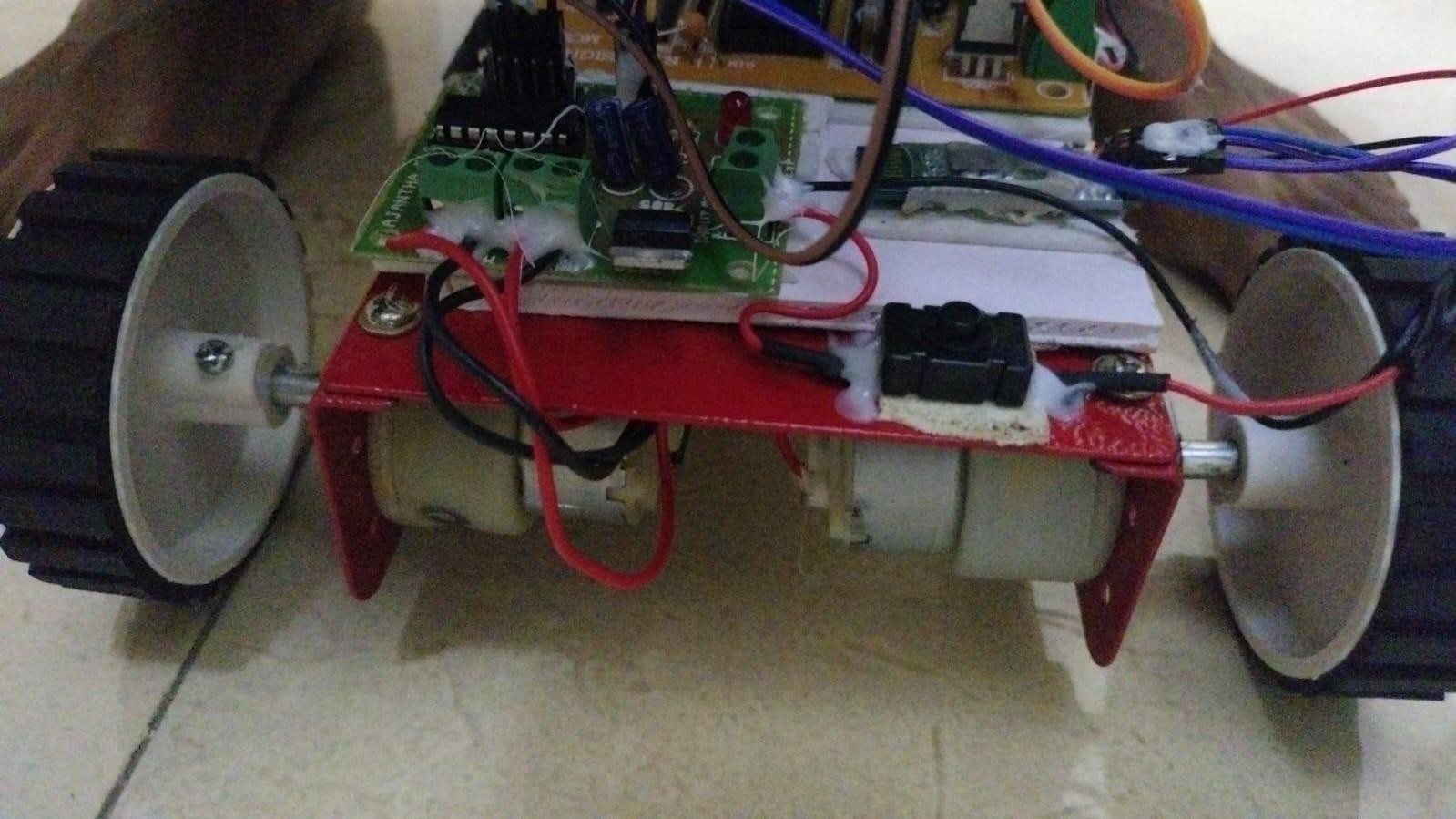
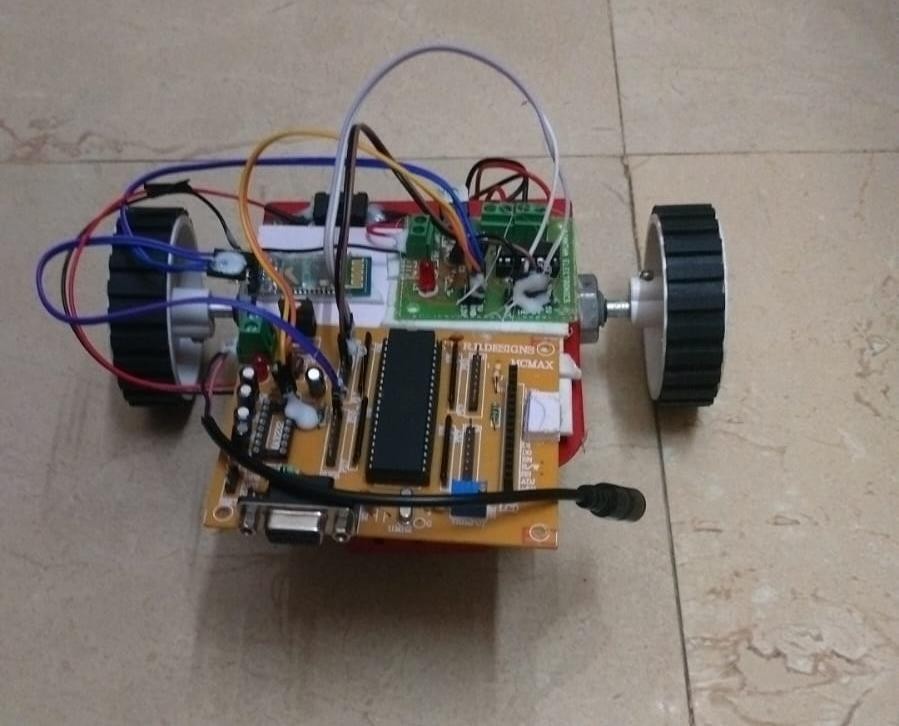
[https://[www.elecrow.com/wiki/index.php?title=HC-](http://www.elecrow.com/wiki/index.php?title=HC-)

18

05\_Bluetooth\_Module]([https://www.elecrow.com/wiki/index.php?title=HC-](https://www.elecrow.com/wiki/index.php?title=HC-05_Bluetooth_Module) [05\_Bluetooth\_Module](https://www.elecrow.com/wiki/index.php?title=HC-05_Bluetooth_Module))

19

# HARDWARE MODEL PROOF



21

