IOT PROJECT REPORT

ON

SMART ROBOTIC CAR

Submitted by:-

SIMRAN (12203073)
DILPREET KAUR (12201157)
MANJEET KUMAR(12200996)

in partial fulfillment for the award of the degree of
BCA

Guided By: -

DR. AJAY ROY

"SCHOOL OF COMPUTER APPLICATION"

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA , PUNJAB

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to all those who have provided invaluable assistance throughout the development of this project. We feel profound pleasure in bringing out this projects report for which we have to go from pillar to post to make it a reality. This project work reflects contributions of many people with whom we had long discussions and without which it would not have been possible. We must first of all, express our heartiest gratitude to respected Dr. Ajay Roy .(School of Electronics and Electrical Eng.)for providing us all guidance to complete project. Furthermore, we would like to extend our gratitude to our friends, who have been a constant source of inspiration and support throughout this journey. Their encouragement and invaluable advice have played a crucial role in the preparation and execution of this project. Last but not the least we express our sincere thanks to the institute LOVELY PROFESSIONAL UNIVERSITY, PHAGWARA, PUNJAB for providing such a platform for implementing the ideas in our mind.

CONTENTS

ABSTRACT

The project's goal is to design, develop, and install a robotic car equipped with ultrasonic sensors for obstacle detection and WiFi control for remote operation. The robotic automobile is a platform for testing autonomous navigation algorithms, sensor integration, and wireless communication protocols in a real-world setting.The project's main components are a modified chassis, motors, wheels, ultrasonic sensors, a WiFi module, and a micro controller for interpreting sensor data and executing control commands. The vehicle's ultrasonic sensors detect obstructions in its route, allowing it to operate automatically. Meanwhile, the WiFi module allows remote control and monitoring of the robotic automobile using a computer or mobile device.The project's objectives include designing, assembling, and integrating hardware components, as well as creating software algorithms for obstacle recognition, avoidance, and wireless communication. In addition, the performance of the robotic automobile is tested, optimized, and validated in a variety of environments and scenarios.The project helps enhance autonomous vehicle technology by providing hands-on experience in robotics, electronics, and programming. Furthermore, it investigates the possible applications of robotic automobiles in transportation, surveillance, exploration, and other areas, laying the groundwork for future study and innovation in the sector.

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

In today's fast-paced world, technology improvements have transformed many facets of our lives, and household cleaning is no exception. With rising demands on our time and energy, the quest for novel solutions to routine tasks has resulted in the development of automated cleaning machines. Among these, the dust cleaning robotic car stands out as a game-changing solution, providing ease, efficiency, and efficacy in preserving cleanliness in our living environments.The dust cleaning robotic car is a sophisticated piece of technology that can autonomously explore indoor settings, collecting dust, grime, and debris as it goes. Compared to typical manual cleaning methods that entail substantial time and effort, this robotic car offers a hands-free alternative, allowing users to focus on other tasks. dust cleaning robotic car is equipped with cutting-edge sensors, cameras, and mapping technology, giving it the intelligence to recognize obstructions, navigate around furniture, and optimize its cleaning route for maximum coverage. This level of autonomy offers complete cleaning outcomes without requiring continual supervision, making it a great choice for busy people looking to keep their homes clean and tidy.Furthermore, the dust cleaning robotic car is meant to be versatile, able to easily navigate a variety of floor surfaces such as hardwood, carpet, or tile. Its fast mobility and small design let it to reach hard-to-reach locations such as under furniture and along baseboards, where dust can gather unnoticed.

1.2 BACKGROUND

In addition to the increasing demands of work, family and personal commitments, modern lifestyle trends have further exacerbated the challenge of balancing household chores such as cleaning. With the rise of telecommuting and the gig economy, individuals are faced with ever-increasing workloads and less time for homework. Additionally, societal changes such as urbanization and single-parent households have intensified the need for innovative cleaning solutions that can accommodate different living arrangements and schedules.In addition, the advent of smart home and Internet of Things (IoT) technology has increased the expectation that home appliances will not only be functional, but also intuitive and efficient. As people look for ways to streamline their daily routines, autonomous cleaning devices have gained considerable interest for their potential to ease the burden of manual cleaning while increasing overall productivity and well-being. The autonomous vacuum cleaner presented in this project is in line with these evolving needs by offering a solution that prioritizes safety, efficiency .

1.3 PROJECT OBJECTIVES

Obstacle avoidance and navigation

The ability to identify obstructions and navigate around furniture and other items in the house is a feature of smart robotic car. The aim is to guarantee smooth and effective functioning while avoiding obstructions or harming furnishings.

Safety Enhancement:

The major goal of robotic automobiles is to improve road safety by drastically lowering the number of accidents caused by human error. Robotic automobiles use advanced sensors and artificial intelligence algorithms to decrease dangers associated with distracted driving, drunk driving, and dangerous behavior.

Autonomous Navigation:

Robotic cars strive to acquire fully autonomous navigation skills, allowing them to operate without human intervention in a variety of situations and conditions. The goal is to create powerful navigation algorithms that allow the vehicle to perceive its surroundings, determine optimal routes, and carry out maneuvers safely and effectively.

Efficient cleaning:

Robotic car Efficient Cleaning: The major goal is to offer homeowners an effective and efficient cleaning solution. The vacuum cleaner should be capable of completely cleaning a wide range of flooring surfaces, including carpets, hardwood floors, and tiles.

Accessibility:

Robotic cars are designed to provide accessible transportation options for people who are unable to drive due to age, disability, or other reasons. The goal is to improve mobility and independence for a diverse group of users, including the elderly, those with impairments, and others who do not have access to traditional transportation choices.

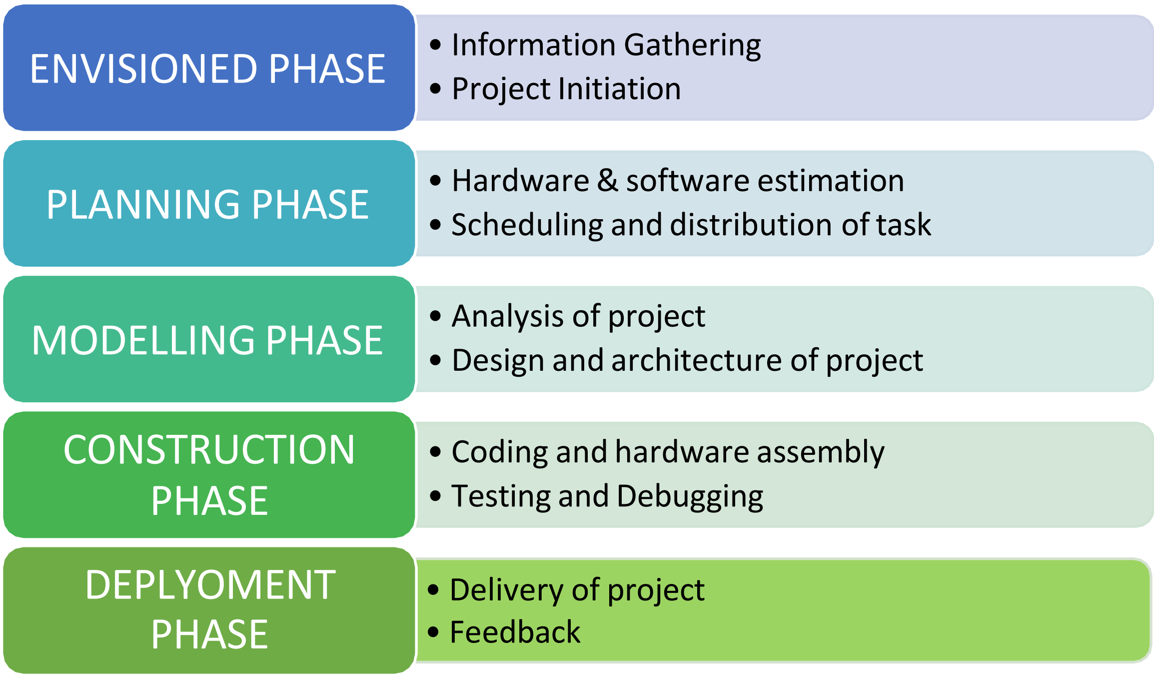
1.4 SCOPE

The goal of the smart robotic car cleaner project's scope is to create an autonomous cleaning system that can adapt to the changing needs of contemporary homes. In order to guide the design and implementation process, this entails conducting in-depth study on current technologies, such as robots, sensor technology, and artificial intelligence. In order to allow the vacuum cleaner to function independently and effectively in interior spaces, the project will integrate cutting-edge features including remote control, ultrasonic sensors, and sophisticated navigation algorithms. To ensure peak performance and user satisfaction, additional considerations will be given to the system's safety, dependability, and energy efficiency. To increase its impact and reach, the initiative will also look into possible uses for it outside of the home, like in commercial and industrial environments. As a whole, the goal is to deliver a superior cleaning solution that offers convenience, efficiency and increased cleanliness while reducing the burden of manual cleaning on users.

1.5 PROJECT MANAGEMENT

Management of any project can be briefly disintegrated into several phases. Our project has been decomposed into the following phases:

Figure 1. Model of phases in project management.



Experimentation

We put the autonomous smart robotic car through its paces in simulated homes as part of our project's experimentation phase. We evaluate how well it navigates crowded locations, stays clear of obstructions, and effectively cleans assigned areas. To guarantee optimal performance and customer pleasure, this phase entails fine-tuning algorithms, modifying sensor values, and improving cleaning processes.

Design

In the design phase, we will create an architectural concept of the autonomous car cleaner, taking into account elements such as integration, arrangement and selection of components. We produce detailed plans and schematics that describe the general design and operation of the system. This phase creates the framework for the development and implementation phases of the project.

Development and testing

We implement the design by connecting the individual parts together and creating the required code during the development and testing phase. We conduct extensive testing to verify system functionality, reliability and performance under many conditions. To ensure that the autonomous vacuum cleaner meets the necessary needs and specifications, this phase involves iterative refinement and optimization.

Real world testing

The design comes to life during the development and testing phase, when we put the pieces together and write the required code. In order for the system to function as it should and be reliable under various conditions, we subjected it to extensive testing. During this phase, the autonomous smart car cleaner is iteratively refined and optimized to ensure that it meets the necessary needs and specifications.

1.6 OVERVIEW AND BENEFITS

The benefits of an established robotic car include:

Improved Safety:

Robotic cars are outfitted with advanced sensors and artificial intelligence algorithms that allow them to detect and respond to possible threats faster and more precisely than human drivers. This technology has the potential to drastically reduce human-caused accidents, making roads safer for everyone.

Efficient cleaning:

Robotic car Efficient Cleaning: The major goal is to offer homeowners an effective and efficient cleaning solution. The vacuum cleaner should be capable of completely cleaning a wide range of flooring surfaces, including carpets, hardwood floors, and tiles.

Dust Collection and Filtration:

Dust Collection and Filtration: Another goal is to properly remove dust, filth, and debris from the floor while maintaining high indoor air quality. Smart vacuum cleaners may contain innovative filtration technologies that trap allergens and pollutants, thereby enhancing the overall health and comfort of the home.

Environmental Benefits:

By encouraging shared rides , optimizing driving patterns, and easing the adoption of electric and alternative fuel vehicles, robotic cars have the potential to cut greenhouse gas emissions and lessen the environmental impact of transportation. This helps to battle climate change and improve air quality.

Cost Savings:

Robotic automobiles have the potential to save individuals and organizations money by eliminating the need for car ownership, parking fees, and fuel use. Shared autonomous transportation services have the potential to provide economical alternatives to traditional means of transportation, particularly in cities.

CHAPTER 2

LIERATURE REVIEW

Khan et al. (2019) present an overview of the functions usually found in smart cleaning robots. These include autonomous navigation, obstacle avoidance, surface and dust detection, and intelligent path planning. Smart cleaning robots that incorporate these features can effectively navigate indoor spaces, detect and remove dust and debris, and optimize cleaning routes for maximum efficiency.

Wu et al. (2020) and Li et al. (2021) investigate the significance of improved sensors and artificial intelligence algorithms in improving the capabilities of smart cleaning robots. These technologies allow robots to adapt to changing settings, learn from interactions, and enhance their cleaning performance over time.

Wang et al. (2020) highlight the use of machine learning methods in dust detection and categorization. Smart cleaning robots can discriminate between different forms of debris and change their cleaning tactics accordingly, resulting in increased overall cleaning efficacy.

Liang et al. (2018) explore the use of laser-based sensors, ultrasonic sensors, and computer vision systems for mapping and navigation. These technologies enable robots to generate precise maps of their surroundings, identify impediments, and plan effective cleaning paths.

Kim et al. (2019) investigated the cleaning effectiveness of a robotic vacuum cleaner on various floor surfaces. The results showed that the robot cleaned as well as manual vacuuming, but with the added bonus of user convenience and time savings.

Furthermore, Chen et al. (2021) investigated the effects of smart cleaning robots on indoor air quality. The findings showed that the robots' filtering systems effectively reduced airborne particles, allergens, and pollutants, resulting in better air quality and respiratory health for inhabitants.

Zhang et al. (2022) show that swarm robotics can be used for commercial floor cleaning, highlighting the potential for increased scalability, coverage, and productivity above single-agent techniques.

CHAPTER 3

THEORY

3.1 NODE MCU

The low-cost, open-source IOT platform is called NodeMCU (Node Microcontroller Unit). It was initially shipped with hardware based on the ESP-12 module and firmware, which uses the Espressif Systems ESP8266 Wi-Fi SoC. Subsequently, ESP32 32-bit MCU support was included.

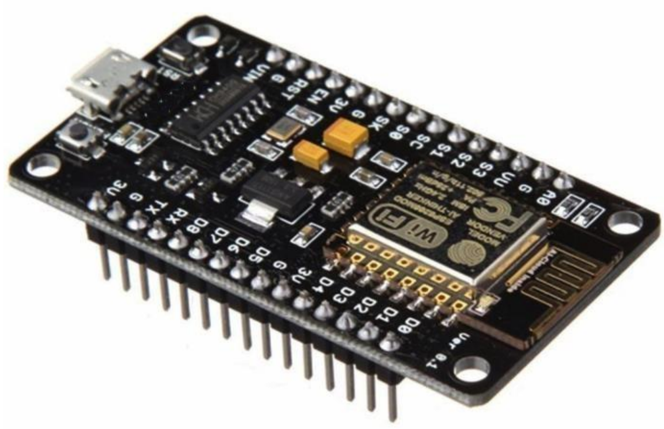


Figure 4. Node MCU Development Board.

Open source prototype board designs are available for NodeMCU firmware. "NodeMCU" is a portmanteau of "node" and "MCU" (microcontroller unit). To be precise, the firmware - rather than the associated development kits - is referred to as "NodeMCU".

Prototype board and firmware designs are also available for free.

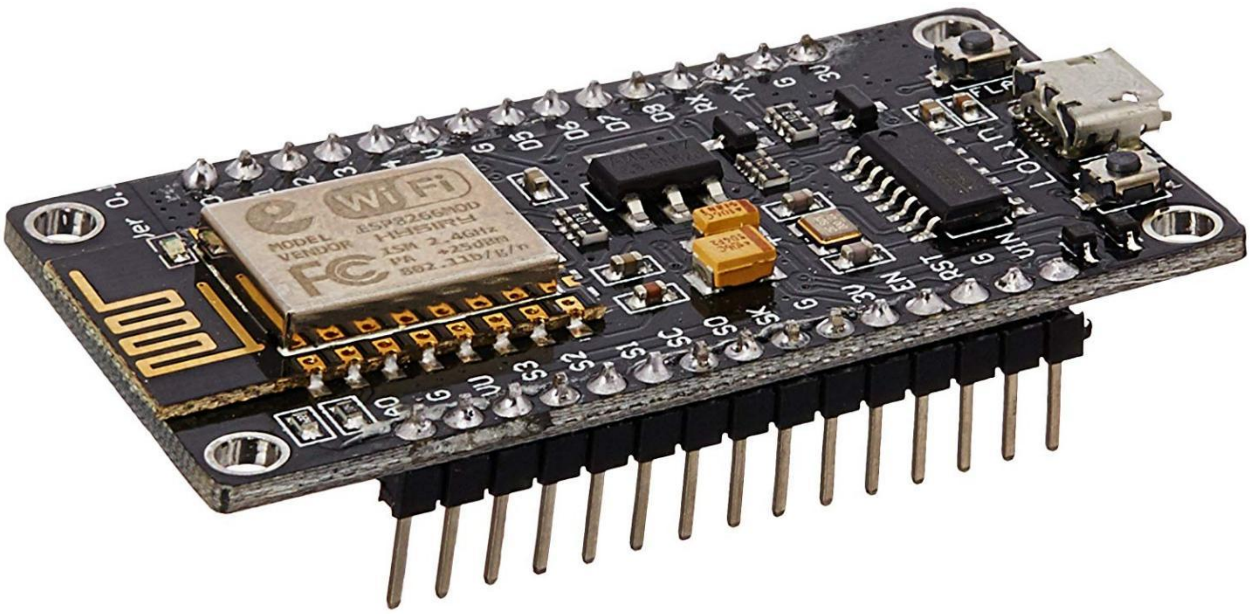
The firmware uses the Lua scripting language.

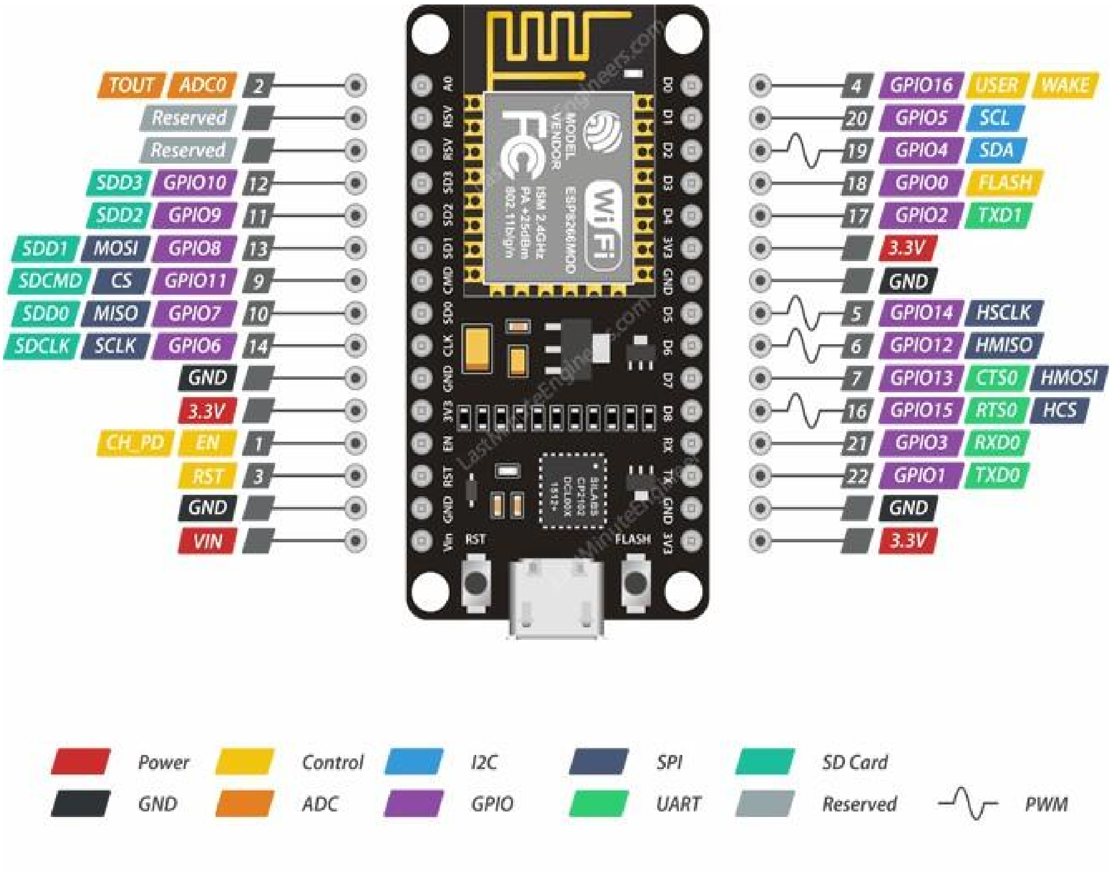
The firmware was developed using the Espressif Non-OS SDK for ESP8266 and is based on the eLua project. It uses numerous open-source initiatives, including SPIFFS and lua-cjson. Due to limited resources, customers must select modules that are relevant to their project and create firmware that meets their requirements. Additionally, support for 32-bit ESP32 has been included.

Commonly used as prototyping hardware is a circuit board that connects the USB controller to a smaller surface board holding the MCU and antenna. This circuit board works as a dual in-line package (DIP). Breadboard prototyping is easy with the choice of DIP format. The Tensilica Xtensa LX106 core, which is a popular IOT core in Wi-Fi SoCs, is integrated into the ESP-12 ESP8266 module on which the original design was based.

3.1.1 Pin Configuration of Node MCU Development Board

The GPIO subsystem is accessible through this module. Instead of using the native GPIO pins, all access is based on the index number of the MCU Node's I/O sets. For example, GPIO pin 16 is mapped to pin D0 on the development kit. The GPIO pins are accessible through the node MCU and the API documentation has a pin mapping table that follows.





PIN NAME ON NODE MCU DEVELOPMENT KIT	ESP8266 INTERNAL GPIO PIN NUMBER	PIN NAME ON NODE MCU DEVELOPMENT KIT	ESP8266 INTERNAL GPIO PIN NUMBER
0 [*]	GPIO16	**7**	GPIO13
1	GPIO5	**8**	GPIO15
2	GPIO4	**9**	GPIO3
3	GPIO0	**10**	GPIO1
4	GPIO2	**11**	GPIO9
5	GPIO14	**12**	GPIO10
6	GPIO12		

[*] D0 (GPIO16) can only be used for GPIO read/write. It does not support open-drain/interrupt/PWM/I²C or 1-Wire.

The ESP8266 Node MCU has total 30 pins that interface it to the outside world. The pins are grouped by their functionality as:

Power Pins: One VIN pin and three 3.3V pins make four power pins. If you have a regulated 5V power supply, you can use the VIN pin to directly power the ESP8266 and its peripherals. The output of the onboard voltage regulator is represented by the 3.3V pins. External components can be powered from these pins.

GND: The development board for the ESP8266 node MCU is grounded using GND.

12 IC Pins: Twelve IC pins are used in projects to connect various I2C sensors and accessories. Support is provided for both I2C Master and I2C Slave. The maximum clock frequency for the I2C interface is 100 kHz and its functionality can be implemented programmatically. It should be noted that the I2C clock frequency should be higher than the slowest clock frequency of the slave device.

GPIO Pins: The 17 GPIO pins of the ESP8266 node MCU can be dynamically assigned to various purposes, including I2C, I2S, UART, PWM, IR remote control, LED light and button. Each digitally enabled GPIo has the option of high impedance tuning or internal pull-up or pull-down. It can also be set to be edge-triggered or level-triggered when set as an input to generate a CPU interrupt.

ADC Channel: An integrated 10-bit precision SAR ADC is present in the MCU node. The ADC can be used to perform the following two tasks: evaluating the input voltage of the TOUT pin and measuring the supply voltage of the VDD3P3 pin.
However, they cannot be put into practice at the same time.

UART Pins: The two UART ports on the ESP8266 node MCU, UART0 and UART1, enable asynchronous communication (RS232 and RS485) with a maximum data rate of 4.5 Mbps.

Communication can be facilitated using UART0 (TXD0, RXD0, RST0 and CTS0 pins). Allows flexible control. But since UART1 (TXD1 pin) can only transfer data, printing the log is usually done with it.

SPI Pins: The ESP8266 has two SPIs in slave and master mode, SPI and HSPI. These SPIs also support the following general purpose SPI functions::

- 4 timing modes of the SPI format transfer
- Up to 80 MHz and the divided clocks of 80 MHz
- Up to 64-Byte FIFO

SDIO Pins: The Secure Digital Input/output (SDIO) contact on the ESP8266 allows direct contact with SD cards. Supported SDIO versions are 4-bit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0.

PWM Pins: There are four Pulse Width Modulation (PWM) channels on the board. PWM outputs can be used to control LEDs and digital motors through a programming implementation. The PWM frequency range can be set between 100 Hz and 1 kHz or 1000 µs to 10000 µs.

Control Pins: These are used to control ESP8266. These pins include Chip Enable pin (EN),Reset pin (RST) and WAKE pin.

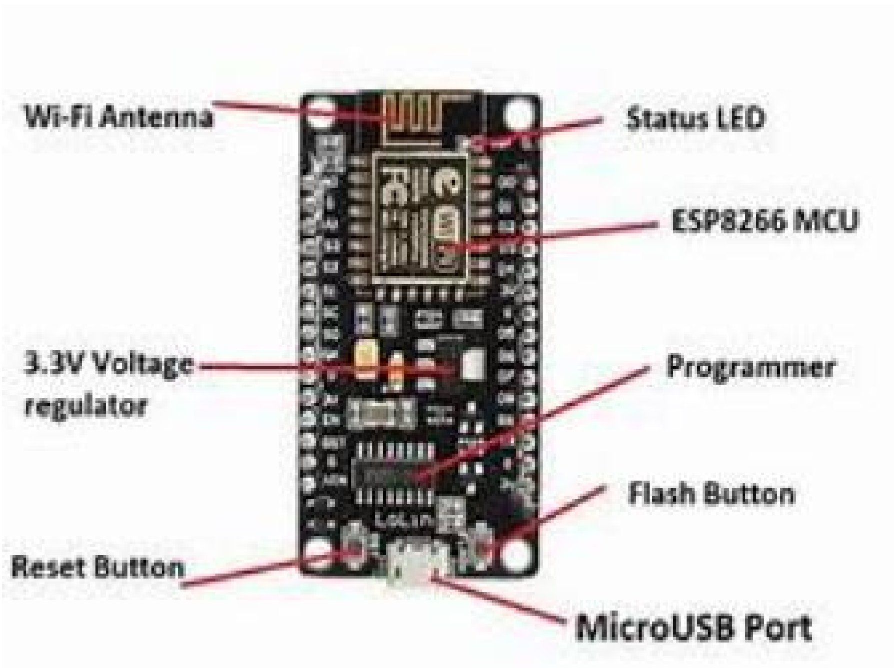
- EN pin – The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
- RST pin – RST pin is used to reset the ESP8266 chip.
- WAKE pin – Wake pin is used to wake the chip from deep-sleep.

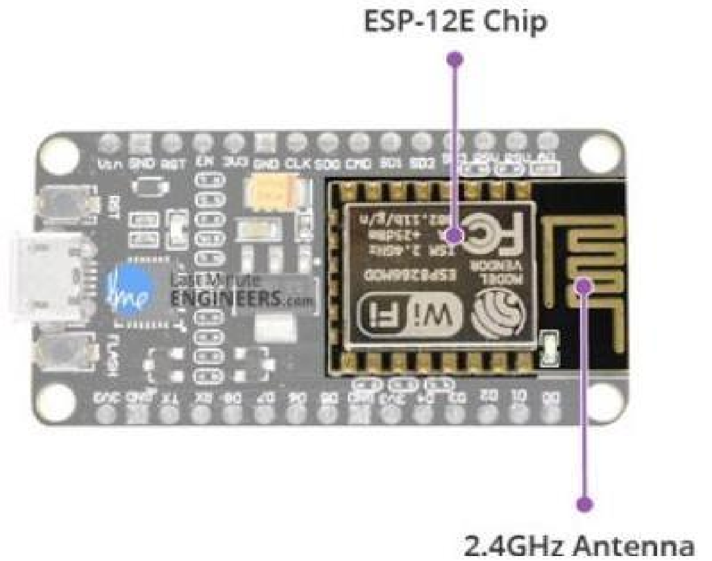
3.1.2 Parts of Node MCU Development Board

3.1.1.1 ESP 12-E Module

The development board equips the ESP-12E module with an ESP8266 chip, which contains a Tensilica Xtensa® LX106 32-bit RISC microprocessor. This microprocessor supports RTOS and runs at a configurable frequency of 80 to 160 MHz. In addition, 128 KB of RAM and 4 MB of Flash memory (for storing applications and data) is enough to handle the huge strings that make up web pages, JSON/XML data and other requirements for modern IOT devices.

In addition to connecting to a Wi-Fi network and interacting with the Internet, the ESP8266 integrates an 802.11b/g/n HT40 Wi-Fi transmitter. This allows it to create its own network that other devices can connect to directly. This further increases the versatility of the ESP8266 Node MCU.





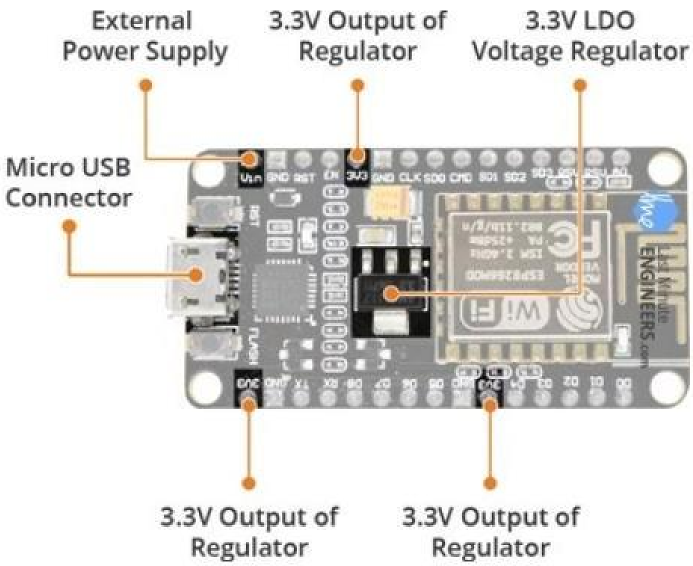
- Tensilica Xtensa® 32-bit LX106
- 80 to 160 MHz clock frequency
- 128 kb internal RAM
- 4 MB external flash
- 802.11b/g/n HT40 Wi-Fi transceiver

Figure 6. ESP 12E module in Node MCU Development board.

3.1.1.2 Power Requirements

The board has an LDO voltage regulator to maintain a constant voltage of 3.3V, as the ESP8266 operates in a voltage range of 3V to 3.6V. As the ESP8266 draws up to 80mA during RF communication, it should have more than enough power supply capacity to reliably handle up to 600mA. In addition, the output of the regulator is marked 3V3 and is broken out on one side of the board. You can use this pin to power external components.

The Micro B USB port on the ESP8266 node MCU provides power to the device. Alternatively, the VIN pin can be used to directly power the ESP8266 and its peripherals if you have a controlled 5V voltage source.



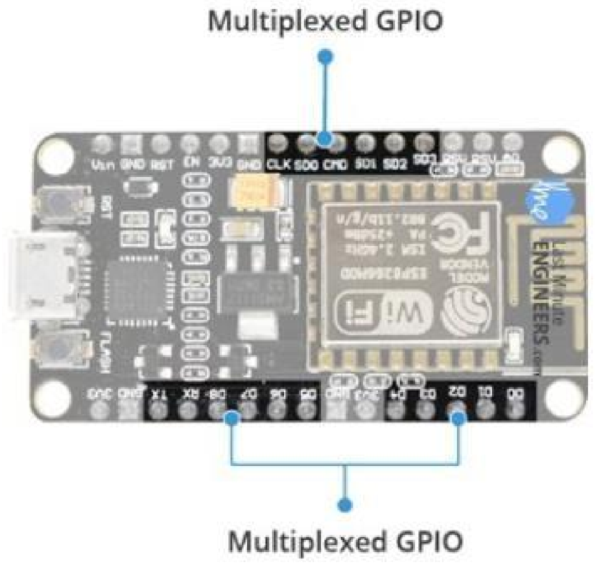
- Operating voltage 2.5V to 3.6V
- On-board 3.6V 600mA regulator
- 80 mA operating current
- 20 µA during sleep mode

Figure 7. Power module on a Node MCU development board.

3.1.1.3 Peripheral I/O

There are a total of 17 GPIO pins on the ESP8266 node MCU, which are separated into pin headers on both sides of the development board. These pins can be used for a wide variety of auxiliary tasks such as:

- ADC channel – A 10-bit ADC channel.



- UART interface – UART interface is used to load code serially.
- PWM outputs – PWM pins for dimming LEDs or controlling motors.
- SPI, I2C & I2S interface – SPI and I2C interface to hook up all sorts of sensors and peripherals.
- I2S interface – I2S interface if you want to add sound to your project.

As a result of the pin multiplexing feature (Multiple peripherals multiplexed on a single GPIO pin), a single GPIO pin can act as PWM/UART/SPI.

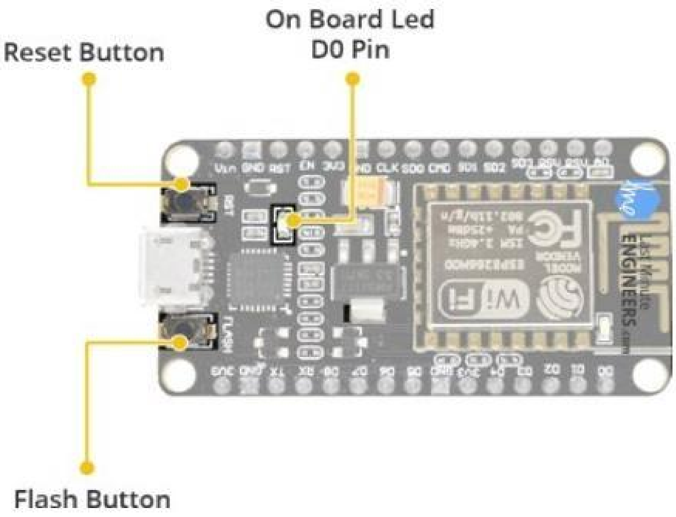
3.1.1.4 On Board Switches and LED Indicators

The Node MCU ESP8266 has two buttons. Naturally, the Reset button (designated as RST) in the upper left corner is used to reset the ESP8266 chip. The download button is located in the bottom left corner of the device and is utilized for firmware upgrades. Additionally, the board contains an LED indicator that can be programmed by the user and is attached to the D0 pin.

3.2.1 Installation of Node MCU

Nowadays, most devices download and install drivers automatically on their own. Windows is unable to recognize the board as a node MCU and function normally because it is unable to communicate with the USB driver on the node MCU. The development board based on the ESP8266 Wi-Fi module is called Node MCU Amica. It can be directly connected to a computer or other USB host device thanks to the Micro USB slot. These can be placed on a cutting board and have 15x2 header pins and a Micro USB slot. The Micro USB slot is used to connect to a USB host device.

The USB to serial port converter CP2120 is included. The user must obtain the driver for the CP2120 (USB to serial converter) before installing it.The system connects to the Node MCU when the user downloads the necessary drivers for the operating system. A newly connected USB device (Node MCU) requires the user to have a COM post node assigned by the system's device management. You will need this communication port number to use Amica's Node MCU. After installing the CP2120 driver, the MCU Node can be programmed in embedded C using the Arduino IDE software. This requires installing the ESP8266 board from the board manager in the Arduino IDE and assigning a communication port.



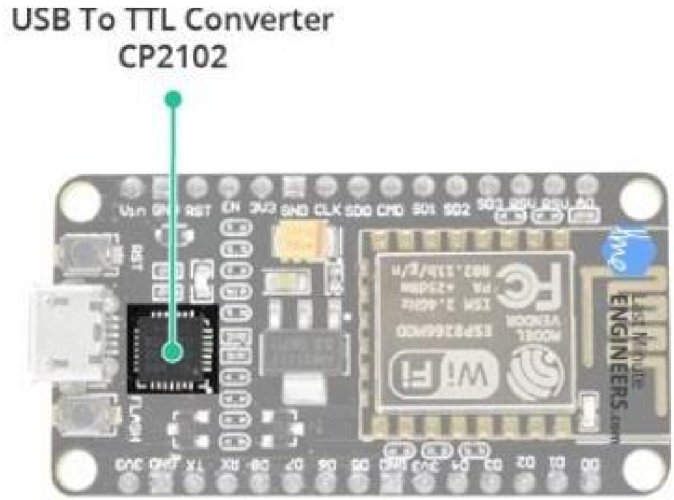
Switches and indicators

RST: Reset the ESP8266 chip
FLASH: Download new programs
Blue LED: User programmable

Figure 9. ON board switches and LED indicators on Node MCU development board.

Serial Communication

On board is Silicon Labs' CP2102 USB-to-UART Bridge Controller, which transforms USB signals to serial and enables computer programming and communication with the ESP8266 chip.



- CP2120 USB-to-UART converter
- 4.5 Mbps communication speed
- Flow control support

Figure 10. CP2120 on Node MCU development board.

3.2 OVERVIEW OF PROJECT

Below is an explanation of how to sign up for a Blynk account and create a special ID that is specific to your device. On the Blynk server, this ID is used to identify a specific device.

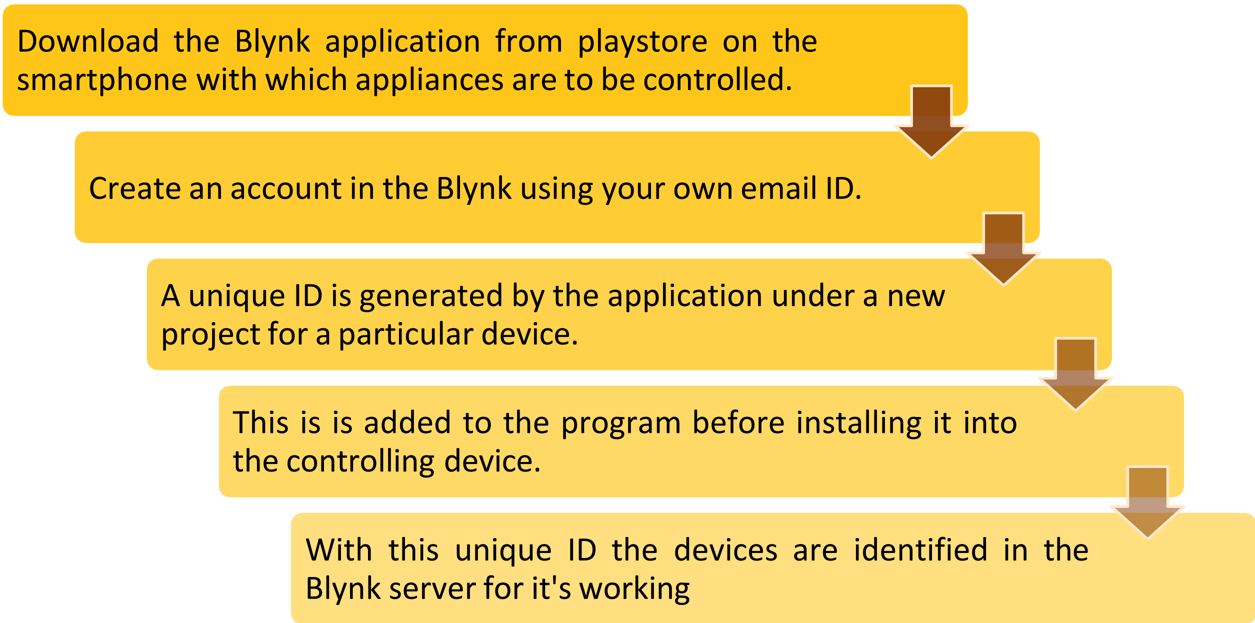


Figure 12. Creating an account and generating unique ID in Blynk Server.

After generating a unique ID, the next step would be to add this key to the embedded C code to enable communication between the MCU node and the Blynk server. Below is an explanation of this process. NodeMCU also need to connect with wifi. Making the connection with user need wifi SSID and WIFI Password. Connecting nodemcu with wifi is essential to connect it with blynk app.

Blynk app will be used to remote control of smart robotic car.

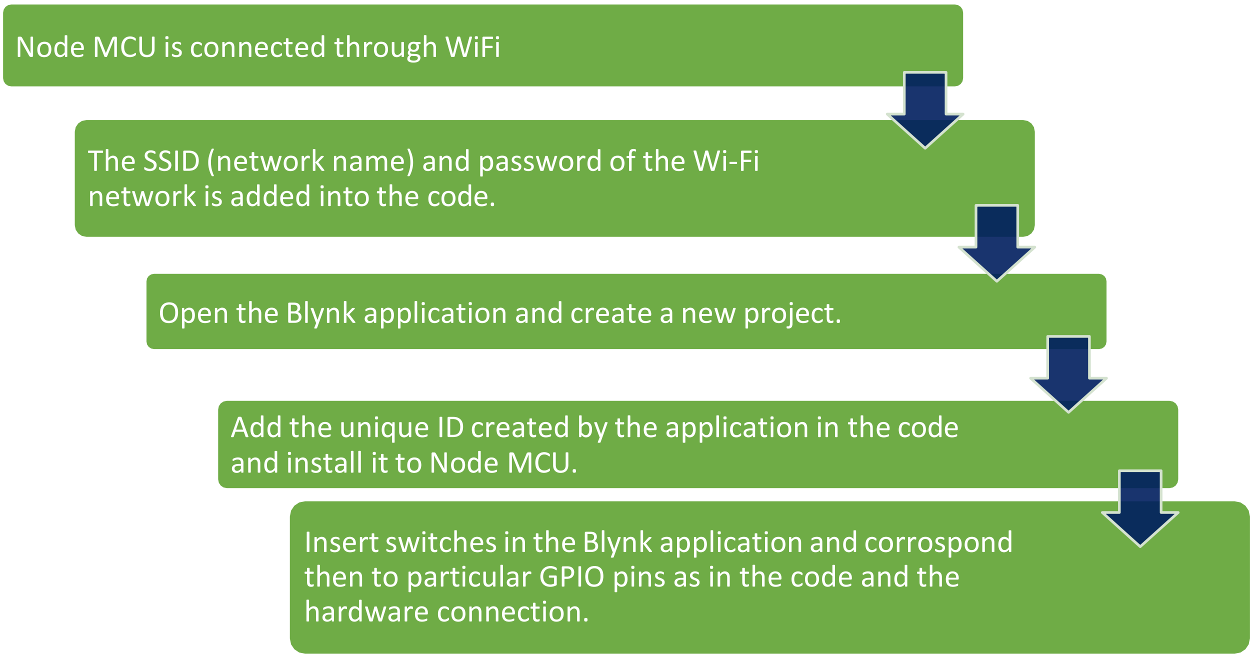


Figure 13. Setup to control Node MCU from Blynk application

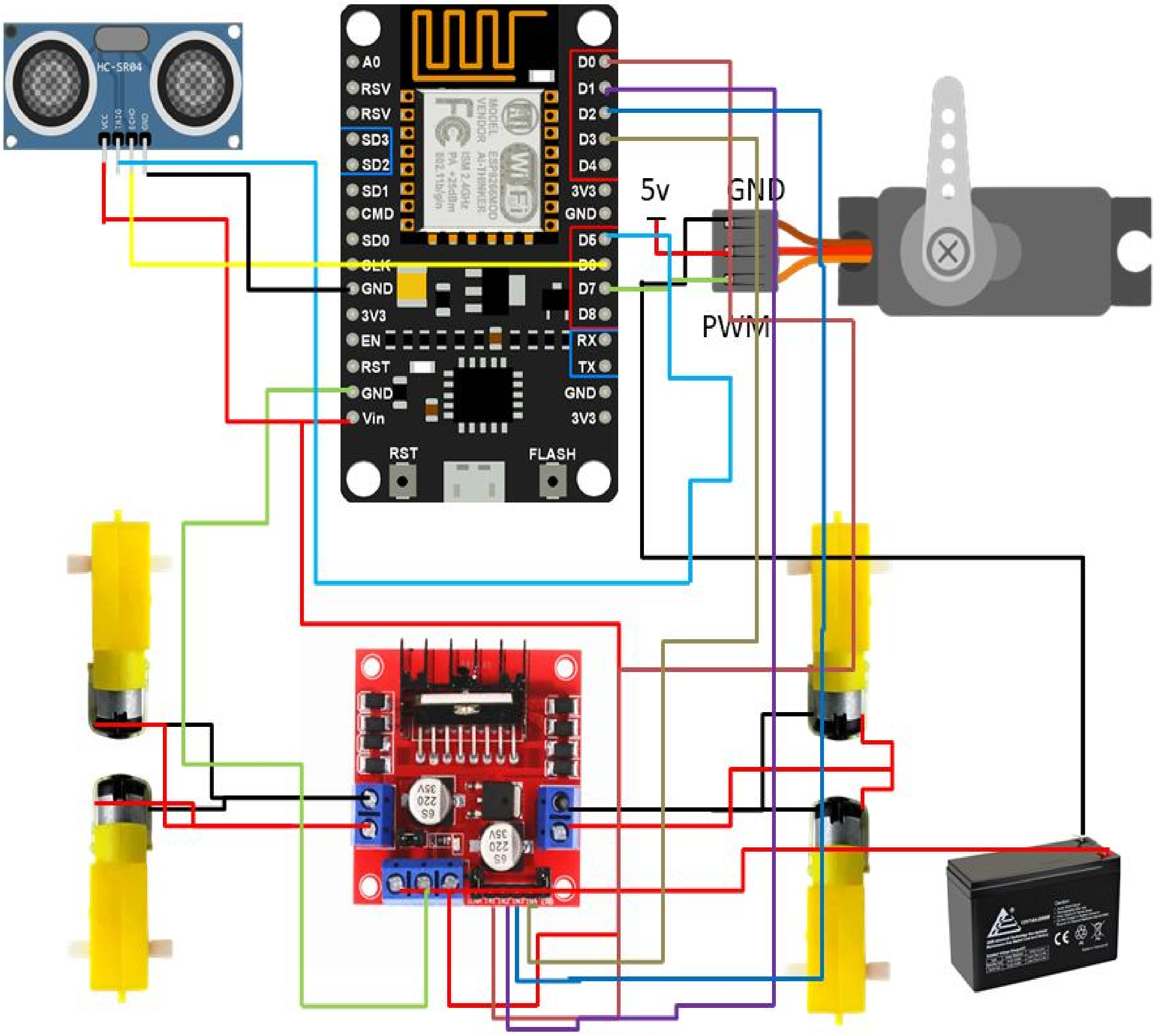
After generating a unique ID, the next step would be to add this key to the embedded C code to enables communication between the MCU node and the Blynk server. Below is an explanation of this process.

The smart cleaning and obstacle-avoiding robotic car is a unique solution for automating domestic cleaning activities while driving autonomously and avoiding obstacles. This robotic car combines powerful sensors, cognitive algorithms, and robotic technology to offer a versatile and effective cleaning partner for both residential and commercial settings.

The robotic automobile has a cleaning system that includes brushes, suction, or mopping pads for sweeping, vacuuming, and mopping. The cleaning method can be tailored to the kind of surface and cleaning requirements.

To travel securely in indoor areas, the robotic car is outfitted with obstacle detection sensors such as ultrasonic, infrared, and LiDAR. These sensors constantly monitor the environment to detect obstructions including furniture, walls, and objects.

3.3 CIRCUIT DIAGRAM



CHAPTER 4

HARWARE MODELLING AND SETUP

4.1 MAIN FEATURES OF THE PROTOTYPE

The features of the developed prototype are:

The characteristics of a robotic car can differ based on its design, intended application, and complexity. Here are some typical characteristics observed in robotic cars:

Obstacle Detection and Avoidance:

To detect obstructions in their path, robotic cars use sensors such as ultrasonic sensors, Li DAR, radar, or cameras. Advanced algorithms let the car to read sensor data, detect obstructions, and plan alternate routes to avoid collisions.

Autonomous Navigation:

Robotic cars are capable of operating without direct human intervention. They can observe their surroundings, plan best routes, and perform driving actions including turning, accelerating, and braking with the help of sensors and onboard computers.

Wireless communication:

The robotic car may contain wireless communication options such as WiFi or Bluetooth, allowing customers to remotely manage and monitor the cleaning operation using a smartphone or other smart devices. Users can start, stop, and schedule cleaning sessions, as well as receive progress updates.

Efficient Cleaning:

The smart cleaning and obstacle-avoiding robotic car automates the cleaning process by navigating indoor spaces and completing cleaning chores with precision and efficiency. This eliminates the need for manual intervention, which saves time and effort when cleaning.

Versatility:

Because of its modular architecture and adjustable cleaning processes, the robotic car can handle a variety of cleaning activities and surface kinds. The robotic car can easily adjust to diverse cleaning requirements, such as vacuuming carpets, brushing hardwood floors, and wiping tile surfaces.

Node MCU:

The micro controller of the prototype is called Node MCU. It is equipped with an ESP8266 WiFi module that enables wireless remote switching of household appliances.

DC motors

are electric motors that convert electrical energy into mechanical motion. They are often used to drive robotic arms, conveyor belts and wheels.

Servo motor:

Rotary drive with precise control of acceleration, speed and angular position. widely used in automation systems, robotics and remote controlled cars.

Ultrasonic sensors

are commonly used in robotics, security systems, and vehicle parking. They work by using sound waves that are louder than what humans can hear to identify things or measure distance.

Motor Shield:

An expansion board that makes it easy to use microcontrollers like Arduino to control motors. It has functions like current detection, speed control and motor direction control.

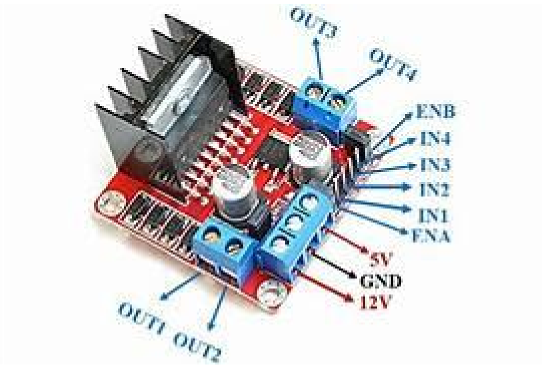
Blynk App:

The Internet of Things was the target market for the **Blynk app**. Among other things, it can store data, visualize it, display sensor data, and remotely control hardware. The Blynk program is the main tool that the prototype uses to sense user commands sent over the wireless network to the hardware.

4.2 COMPONENTS REQIRED

Sr. No	Component	Quantity
1.	Node MCU	1
2.	Ultrasonic Sensor	1
3.	12V battery	1
4.	DC motors	4

5.	Wheels	4
6.	Motor Shield	1
7.	Male pin header	1
8.	Female pin header	1
9.	Jumper wires	8
10.	Servo motor	1
11	L298 motor driver	1



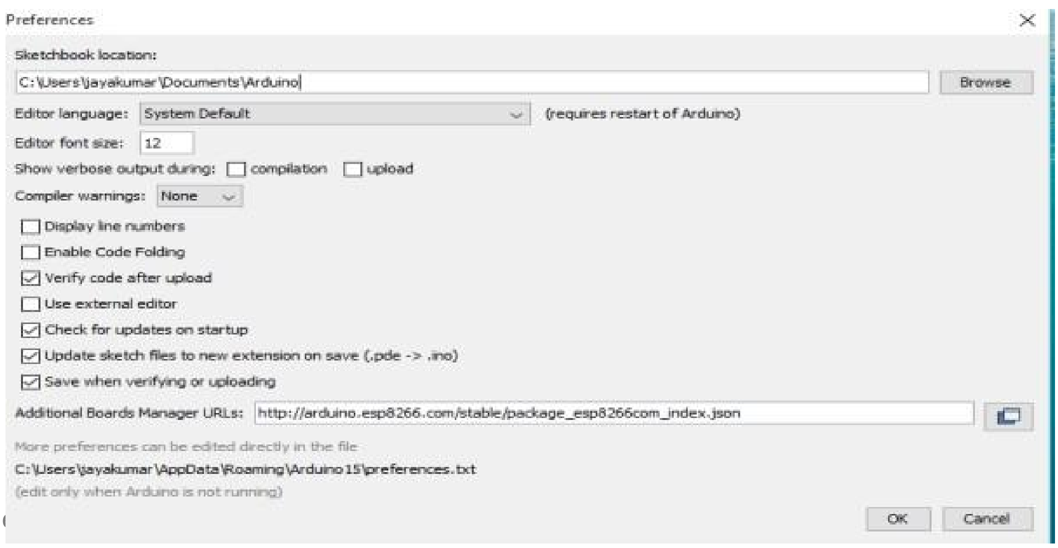
Driver installation for hardware interfacing

Nowadays, most gadgets automatically download and install drivers. Windows is unable to recognize the board as a node MCU and function normally because it is unable to communicate with the USB driver on the node MCU.

- MCU Node Amica is a development board based on the ESP8266 Wi-Fi module. It can be directly connected to a computer or other USB host device thanks to the Micro USB slot. In addition to the Micro USB slot and 15X2 header pins that can be placed on the breadboard, Ti also offers a Micro USB port for connecting to a USB host device. The USB to serial port converter CP2120 is included.
- The user must obtain the driver for the CP2120 (USB to serial converter) before installing it.
- The system connects to the Node MCU when the user downloads the necessary drivers for the operating system.
- A newly connected USB device (Node MCU) requires the user to have a COM post node assigned by the system device management. You will need this communication port number to use Amica's Node MCU.

4.1.2 Interfacing Node MCU with Arduino IDE

First, we will need to update the board manager with a custom URL using the latest version of the Arduino IDE. After launching the Arduino IDE, go to File > Preferences. Then enter the following URL in the text box labeled "Additional Board Manager URLs" at the bottom of the window:*http://arduino.esp8266.com/stable/package_esp8266com_index.json*



Good. The board manager can then be accessed by selecting Tools > Boards > Board Manager. In addition to the typical Arduino boards, there need to be a few new ones. Enter esp8266 to narrow your search. After selecting Install, click on this item.

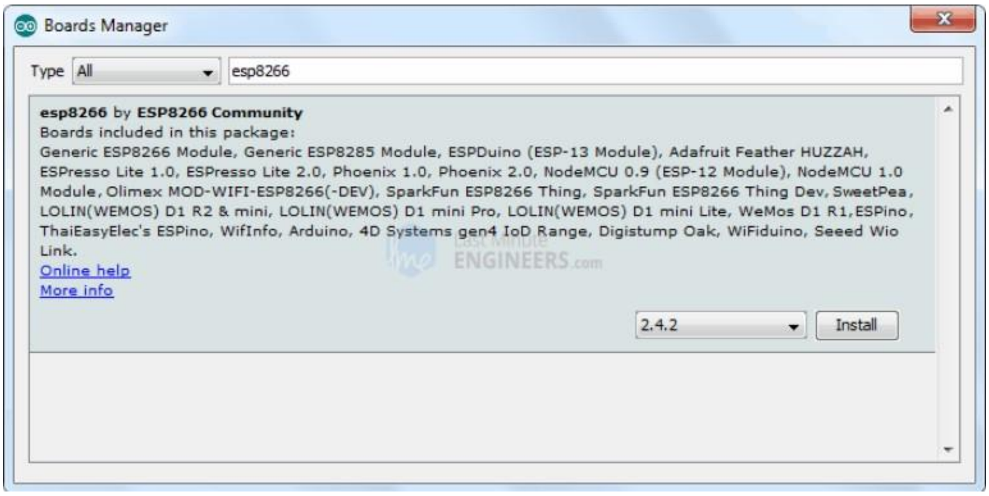


Figure 18. ESP8266 board installation in Arduino IDE.

Before uploading the sketch and experimenting with the LEDs, make sure the board is correctly selected in the Arduino IDE. When it opens, select Node MCU 0.9 (ESP-12 Module) from the Arduino IDE > Tools > Board menu.

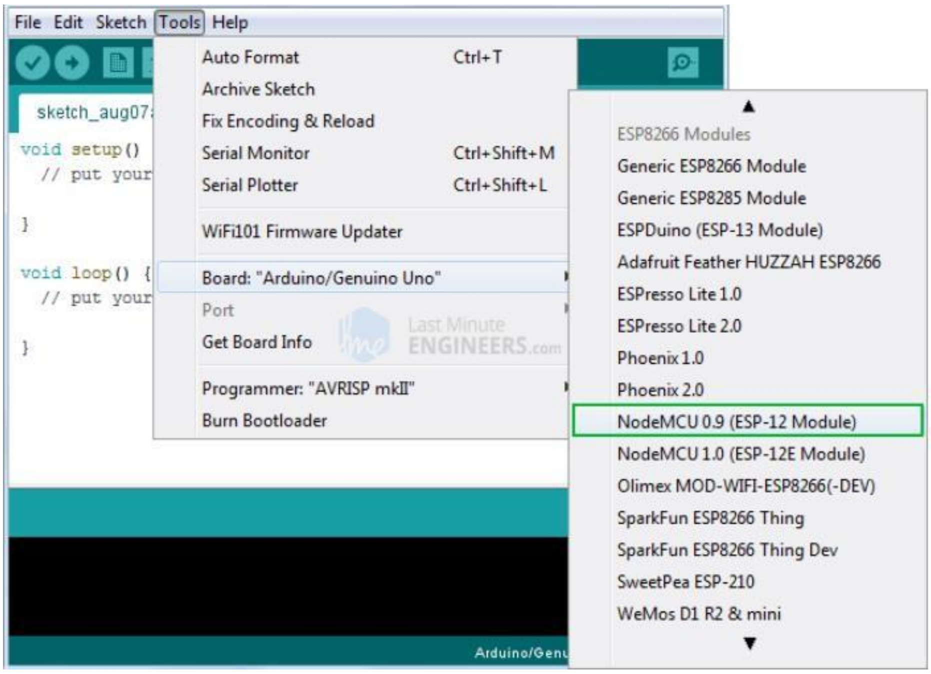


Figure 19. Arduino IDE board manager installation.

Now connect the micro-B USB cable to the computer and plug in the ESP8266 NodeMCU. The board must be assigned a different COM port once plugged in. This will appear as something like COM# on Windows machines and as /dev/tty.usbserial-XXXXXX on Mac and Linux machines. Select this serial port from Arduino IDE > Tools > Port. Next, select Upload speed: 115200.

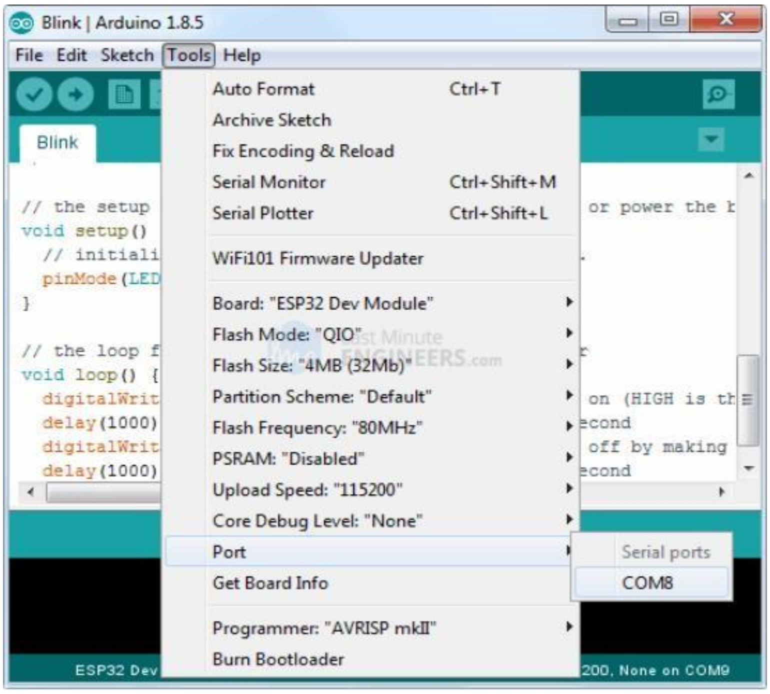


Figure 20. Assigning communication port on Arduino IDE.

4.1.3 Uploading code to Node MCU

- · A USB cable is used to connect the NodeMCU to the PC.
- We will now configure the Arduino IDE by making various changes to its settings. Now launch the Arduino IDE.
- Select "NodeMCU 1.0 (ESP-12E Module)" as the board by selecting Tools > Board. We only need to update the settings once. We'll start writing the code right away.
- Choose Documents > Examples > Blynk > Boards_WIFI > ESP8266_Standalone. A new file will open that has some pre-written code. The code has been modified in the following ways.

1. Replace "YourAuthToken" with the authentication token generated by your Blynk server on the line that reads "char auth[] = "YourAuthToken."
2. Change the part of YourNetworkName in the line that says "char ssid[] = "YourNetworkName" to the name of the WiFi network that the MCU of the node

needs to connect to.

Figure 21. Code in Arduino IDE to be installed to Node MCU

3. Note the line that reads "char pass[] = "Your password" and change it to your Wi-Fi password.

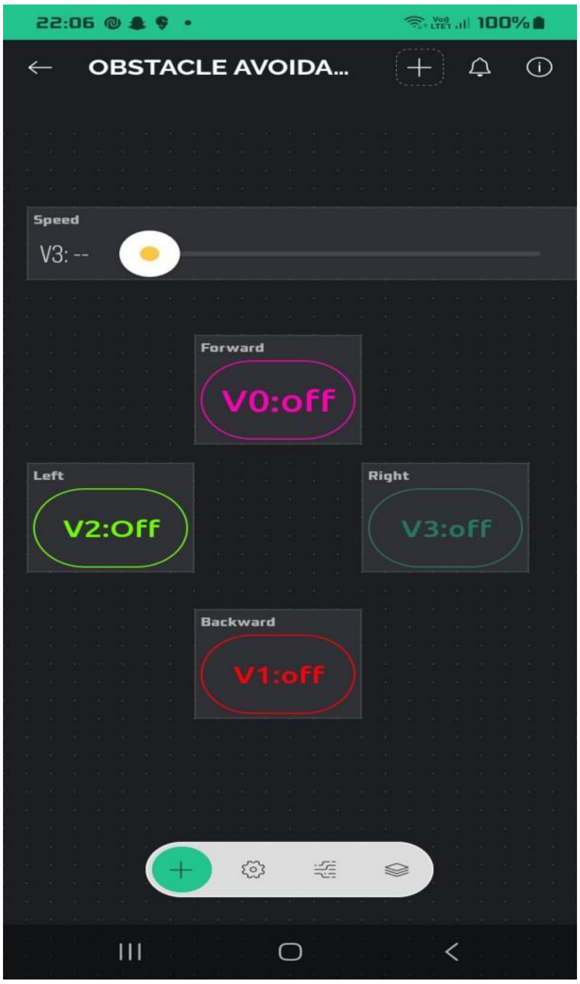
- Code is ready for hardware upload. The code uploads to the MCU node when the upload button is clicked, and the next time it's powered on, it immediately connects to the designated Wi-Fi network.

4.3 SETTING UP THE SYSTEM

Downloading and installing and Blynk application on smartphone

- Install Blynk by downloading it from the Play Store.
- Log in after creating a new account.
- Create a new project and name it. Choose the right hardware - for example Arduino Uno or ESP8266 - and the type of connection - for example Wi-Fi.

- The authentication token that will be used to identify the hardware will be sent to your email by Blynk when you create the project.
- Add relevant widgets, such as buttons to control motors or sliders to select cleaning modes, as a smart vacuum cleaner uses various components, including sensors and DC motors.
- Name each widget and select the hardware pin that corresponds to it to make it unique.
- Be sure to set up additional widgets for sensor feedback, such as dust sensors to monitor cleanliness or ultrasonic sensors to identify obstacles.
- After everything is configured, the Blynk app will act as an interface to manage and monitor the smart robotic car ..



4.4 HARDWARE ASSEBLY

Hardware assembly mainly includes connecting specific digital pins of NodeMCU to the various sensors and devices including the connection of supply and ground pins. The main functional assemble in this prototype is simple.

The vital part in hardware assembly is taking into account the digital pin that corresponds. This connection is done as per the setup of Blynk application. The radio buttons on Blynk application are set up to switch a particular digital pin in Node MCU.

ULTRASONIC SENSOR

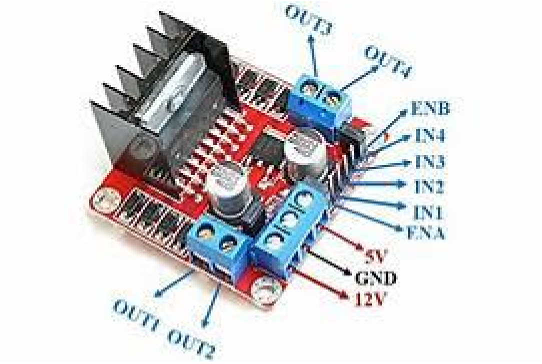
Ultrasonic sensors are electronic devices that emit ultrasonic sound waves and transform them into electrical signals to determine the distance to a target. Emitted ultrasonic waves move quicker than audible sound. There are two key components: the transmitter and the receiver. The transmitter generates sound using piezoelectric crystals, which then travels to the target before returning to the reception component. To determine the distance between the target and the sensor, the sensor measures the time it takes for sound to travel from transmitter to receiver. The calculation is performed as follows.

D = 1/2 T * C

Where 'T' represents time measured in seconds.

'C' represents sound speed = 343 measured in mts/sec.

L298 MOTOR MODULE



This L298N Motor Driver Module is a high-power module that can drive both DC and stepper motors. This module has an L298 motor driver IC and a 78M05 5V regulator. The L298N Module can control up to four DC motors, or two with directional and speed control.

Pin Name	Description
IN1 & IN2	Motor A input pins. Used to control the spinning direction of Motor A
IN3 & IN4	Motor B input pins. Used to control the spinning direction of Motor B
ENA	Enables PWM signal for Motor A
ENB	Enables PWM signal for Motor B
OUT1 & OUT2	Output pins of Motor A
OUT3 & OUT4	Output pins of Motor B
12V	12V input from DC power Source
5V	Supplies power for the switching logic circuitry inside L298N IC
GND	Ground pin

SERVO MOTOR

A servo motor is an electric motor that can precisely control its angular or linear location, speed, and torque. It consists of an appropriate motor connected to a sensor for position feedback and a controller that governs the motor's movement in accordance with a predetermined set point.

Servo motors are widely utilized in industrial applications such as robotics, CNC machines, and auto production, which demand high accuracy, quick reaction, and smooth motion.
A servo motor is an electric motor that may spin or move to a predetermined location, speed, or torque in response to an input signal from a controller.
The phrase servo derives from the Latin word servus, which means servant or slave. This reflects the historical use of servo motors as auxiliary drives to supplement the primary drive system.
However, modern servo motors can provide excellent performance and precision as principal drives in a variety of applications.
A servo motor has three essential components:
A motor can be either a DC or an AC motor, depending on the power supply and application needs. The motor provides mechanical power for rotating or moving the output shaft.
A sensor: This could be a potentiometer, an encoder, a resolver, or another device that detects the position, speed, or torque of the output shaft and delivers feedback signals to the controller.
A controller is an analog or digital circuit that compares the sensor's feedback signals to the desired setpoint signals from an external source (such as a computer or a joystick) and creates control signals to modify the motor's voltage or current as needed.

DC MOTOR

A motor is an electrical machine that converts direct current electrical energy into mechanical energy. It is based on electromagnetic induction, which occurs when a conductor carrying electricity (often a coil of wire) is put in a magnetic field and is forced to rotate. This rotation is utilized to accomplish mechanical tasks.

CHAPTER 5

LOGIC AND OPERATION

5.1 FLOW CHART

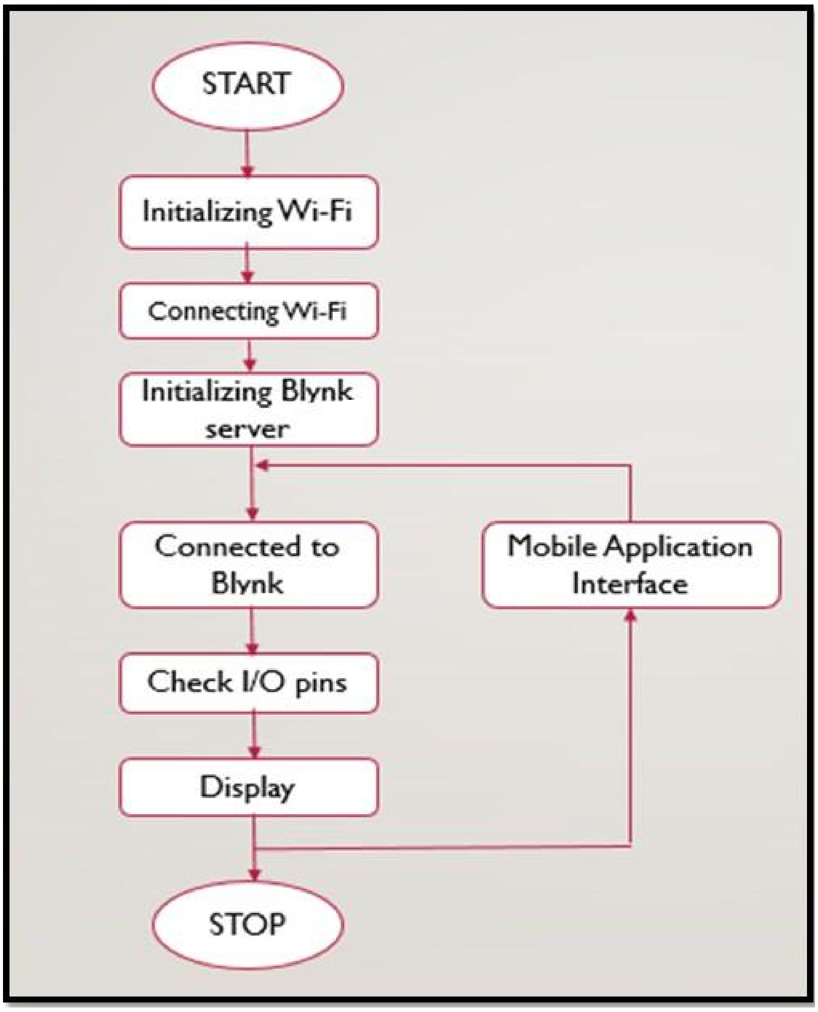


Figure 25. Flow chart of prototype function.

This flow chart shows the working of the project. The process starts be initializing the Wi-Fi, the network name and password are written in the code and uploaded to Node MCU. The android device is connected to Node MCU over Wi-Fi. The Blynk server is set up and connection is made, the devices is identified in the Blynk server using the generated authentication token. The command for controlling the load is given to the application, and this command, over Wi-Fi network is sent to the Node MCU.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 RESULT

Smart cleaning robotic cars have produced encouraging results, demonstrating their efficacy in a variety of household cleaning tasks. Below are some major conclusions from studies and user feedback:

Cleaning Efficiency: Smart cleaning robotic cars have shown excellent cleaning performance across a variety of floor kinds and surface areas. these robots effectively remove dust, grime, and debris, with cleaning outcomes comparable to or better than manual cleaning methods. Their capacity to maneuver independently and adapt to changing conditions enables them to cover a large area and reach difficult-to-access areas, resulting in thorough cleaning performance.

Time Savings: One of the most significant benefits of smart cleaning robotic cars is their capacity to save time for users. By performing cleaning tasks automatically, these robots minimize the need for manual intervention, allowing users to focus on other activities. people value the convenience and time savings provided by smart cleaning robots.

User Satisfaction: According to user feedback and satisfaction studies, smart cleaning robotic cars have received a positive response. users like the robots' convenience, effectiveness, and ease of usage. Reliability, cleaning performance, and user-friendly interfaces all help to increase user happiness and acceptance.

Energy Efficiency: Efforts to enhance the energy efficiency of self-cleaning robotic automobiles have yielded encouraging results. highlight the use of energy-efficient motors and optimization methods to reduce energy consumption while maintaining cleaning performance. This emphasis on energy conservation is consistent with sustainability goals and minimizes the environmental impact of robotic cleaning systems.

Overall, the findings of smart cleaning robotic cars show that they are effective, efficient, and user-friendly for residential cleaning duties. Continued research and development activities are projected to improve their capabilities, alleviate problems, and broaden their uses in a variety of indoor settings.

6.2 LIMITATIONS

Navigation Challenges: Smart cleaning robotic autos use sensors and algorithms to traverse indoor spaces. However, they may have difficulty effectively mapping complicated layouts, particularly in cluttered or poorly lit locations. Furthermore, obstructions like as small objects, loose wires, or elevation changes might impair navigation, resulting in inadequate cleaning or crashes.

Inability to Reach Tight places: Despite their small size, smart cleaning robotic cars may struggle to enter tight places and thin gaps between furniture or appliances. This constraint may result in sections remaining uncleaned, necessitating manual intervention to ensure comprehensive cleaning.

Limited Cleaning Power: While smart cleaning robotic cars are good at eliminating surface dust and debris, they may lack the power of regular vacuum cleaners or mops when it comes to thorough cleaning or removing persistent stains. Certain types of grime, such as pet hair or big debris, may also cause problems for robotic cleaning systems.

Maintenance Requirements: Smart cleaning robotic cars, like other mechanical devices, require regular maintenance to ensure peak performance. To avoid problems and preserve cleaning efficiency, components such as brushes, filters, and sensors must be cleaned or replaced on a regular basis. Failure to conduct maintenance procedures might reduce the performance and lifespan of the robotic automobile.

Limited Battery Life: Battery life is an important factor in determining the operational length of smart cleaning robotic automobiles. While battery technology has advanced, robotic cleaners may still have a limited runtime and must be recharged frequently during extended cleaning sessions. This limitation can cause cleaning cycles to be interrupted and the entire cleaning process to take longer.

Cost: Smart cleaning robotic cars can be more expensive than regular cleaning equipment. The initial investment expense may dissuade some consumers from embracing this technology, particularly if they see it as an unneeded luxury rather than a useful home appliance.

Lack of Versatility: While smart cleaning robotic cars excel at cleaning floors, they may not be capable of cleaning other surfaces such as windows, shelves, or upholstery. This lack of adaptability restricts their effectiveness in full household cleaning procedures, necessitating the use of other cleaning instruments or processes.

6.3 FUTURE SCOPE

Enhanced Navigation and Mapping: Future smart cleaning robotic automobiles are likely to have better navigation and mapping capabilities. Advanced sensors, artificial intelligence algorithms, and simultaneous localization and mapping (SLAM) techniques will allow robots to generate more precise maps of their surroundings, navigate complicated areas with higher precision, and adapt to dynamic changes in their surroundings.

Multi-Surface Cleaning: To meet a variety of cleaning needs, future robotic cleaners may include multi-surface cleaning capabilities. These robots will be developed to clean a variety of floor kinds, including hardwood, carpet, tile, and linoleum, while maintaining cleaning performance and efficiency. Adaptive cleaning mechanisms and replaceable cleaning attachments will allow robots to vary their cleaning procedures dependent on the surface type and condition.

Integration with Smart Home Systems: By integrating smart home systems and Internet of Things (IoT) platforms, smart cleaning robotic cars and other connected equipment in the home will be able to communicate and interoperate seamlessly. Users will be able to remotely control and monitor robotic cleaners via smartphone apps or voice commands, set cleaning schedules, and receive real-time notifications and status updates.

Autonomous charging : Future robotic cleaners may include enhanced charging and docking facilities that allow for independent recharge and maintenance. These stations will be outfitted with self-aligning connectors, wireless charging technology, and automatic cleaning and maintenance capabilities, allowing robots to recharge themselves when not in use and restart cleaning chores without human involvement.

Integration of smart cleaning technologies : To increase cleaning performance and efficiency, future smart cleaning robotic cars may incorporate advanced cleaning technologies such as ultraviolet (UV) sterilization, electrostatic cleaning, and chemical-free cleaning procedures. These technologies will allow robots to disinfect surfaces, remove persistent stains and allergies, and keep a clean and healthy living environment.

6.2 CONCLUSION

In conclusion, the creation of smart cleaning obstacle-avoiding cars represents a big step forward in the evolution of domestic cleaning technology. These revolutionary technologies provide numerous benefits, ranging from increased cleaning efficiency and time savings to improved user experience and convenience. Smart cleaning obstacle-avoiding cars can efficiently navigate indoor spaces by combining advanced sensors, and autonomous navigation systems, effectively avoiding obstacles and cleaning surfaces with precision. Their capacity to adapt to changing conditions and learn from previous interactions allows them to provide thorough cleaning outcomes across a wide range of floor types and surface areas. Looking ahead, the broad adoption of smart cleaning obstacle-avoiding cars has the potential to alter the way we approach cleaning jobs, providing unprecedented ease, efficiency, and effectiveness. By harnessing cutting-edge technology and embracing constant innovation, these devices are transforming the future of household cleaning, making daily chores easier, more manageable, and less time-consuming for consumers all over the world.

CHAPTER 7
REFFERENCES

"Smart Vacuum Robot"by M. Vijayalakshmi,G. Bhargavi Baljoshi.G. Lavanya G. Master Sushil, Department of Computer Science and Engineering, KLE Technological Univeristy, Vidyanagar, Hubballi.Int. Res. J. Eng. Technol. (IRJET, 7 (7) (July 2020)

"Low cost autonomous robot cleaner using mapping algorithm based on Internet of Things (IoT), IOP Conference Series: Materials Science and Engineering." by R.J. Ong and KNFKu Azir

"Designing and Optimization of An Autonomous Vacuum Floor Cleaning Robot",by Md.F. Islam, S. Islam, N. Hossain,S. Datta, School of Electrical and Electronic Engineering, Shahjalal University of Science & Technology, Sylhet, Bangladesh.

"A smart autonomous floor cleaner with an Android-based controller" by A.P. Murdan, P.K. Ramkissoon

"Smart Floor Cleaning Robot"Indronil Dey Niloy;Sayed Mohaiminul Hoque;Mashfiqul Hoque;Afif Bin Arfan;Israt Jahan;Islam Bin Mursalin;Mohammad Shidujaman;Mohammad Rejwan Uddin;Mahady Hasan 2023 IEEE Region 10 Symposium (TENSYMP)Year: 2023 | Conference Paper | Publisher: IEEE

"Design and development of a DTMF controlled room cleaner robot with two path-following method"Humayun Rashid;Akash Mahmood;Sarjahan Shekha;S.M. Taslim Reza;Md. Rasheduzzaman 2016 19th International Conference on Computer and Information Technology (ICCIT)Year: 2016 | Conference Paper | Publisher: IEEE

"Sweepy – The Smart Floor Cleaner"Varsha P.H.;Lavanya V.;Meghana K.;Rohan P.S.;Sneha R. 2018 International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C) Year: 2018 | Conference Paper | Publisher: IEEE

"Research on Intelligent System of Automatic Cleaning Robot Based on Data Analysis Algorithm"Liwen Sun;Yanhang Li2023 3rd International Symposium on Computer Technology and Information Science (ISCTIS)Year: 2023 | Conference Paper | Publisher: IEEE

"A Survey On Smart Vacuum Leaning Robot"Ramesh A;Rameshkumar V;Raghu Ram Raj N;Ragabharathi A;Sowndar K;Sivakumar P 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)Year: 2023 | Conference Paper | Publisher: IEEE

"Swachh Robo Ones Cleanliness Identity"J.J Jijesh;Shivashankar;H R Pooja;Shwetha Nagesh;T Gagandeep;Immanuel S Shimron2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)Year: 2019 | Conference Paper | Publisher: IEEE

"Automatic Floor Cleaning Robot"K Saravanan;E Siva Prasanna;R Sattish;R Udhaya Abinesh;P Anandakumar2022 IEEE International Conference on Data Science and Information System (ICDSIS) Year: 2022 | Conference Paper | Publisher: IEEE

"Path planning algorithm for sweeping robot full traversal cleaning area"Shijie Guo;Yonglin Huang;Dunjie You;Teng Wang;Qiang Long2022 Asia Conference on Advanced Robotics, Automation, and Control Engineering (ARACE)Year: 2022 | Conference Paper | Publisher: IEEE

Mohd. Shahbaz Khan et al “Bluetooth control cleaning robot using Arduino”. They have designed a robot and the robot is controlled using Bluetooth which is present at both transmitter and receiver end [1]. Vijayalakshmi M et al proposed “Smart Vacuum Robot” with progressive technology. S-curve planning is used for efficient working along with sensors to avoid obstacles

Niloy ID, Hoque SM, Hoque M, Arfan AB, Jahan I, Mursalin IB, Shidujaman M, Uddin MR, Hasan M(2023 Sep 6) Smart Floor Cleaning Robot. In2023 IEEE Region 10 Symposium (TENSYMP) (pp. 1-4). IEEE.

Rahayu ES, MARDIONO M, AZIS D(2020 Aug 6) Design of Vacuum Cleaner Dual Mode Robot Prototype Based on Arduino Uno with Bluetooth and Smartphone Communication. JournalNX.:26-35