FABRICATION OF THE CUSTOMIZED ROBOTIC CLEANER

Mr. Rajamahanti Surya Kiran[1], Pentakota Aknadh[2], Indana Gnaneswara Sriram[3] , Dendukuri Srihari Raju[4] , Nakka Mohit Venkata Sai[5] , Vandana Ranjith Kumar[6] , Godi Sai Pavan[7] , Koruvada Yamini Prasad[8]

Assistant professor[1,2]Mechanical Engineering, Sanketika Vidya Parishad Engineering College.
[3,4,5,6,7,8] BTECH Student, Sanketika Vidya Parishad Engineering College.

Abstract:
This project presents the design and implementation of a novel robotic mob cleaner, addressing the need for efficient and autonomous floor cleaning in domestic and commercial settings. The robot features a modular architecture, allowing for interchangeable cleaning modules dedicated to specific tasks like mopping, and disinfecting. Precise navigation is achieved through a combination of ultra-sonic sensors for obstacle avoidance , enabling efficient cleaning paths and obstacle avoidance. User interaction is enhanced through a mobile app providing control, scheduling, and monitoring functionalities. In today's dynamic landscape, the demand for efficient and autonomous cleaning solutions is escalating, spanning diverse environments from commercial complexes to residential spaces. This project delves into the development and deployment of a smart floor cleaning robot poised to redefine traditional cleaning methodologies. Historically, manual labor reliant on brooms and mops has been the cornerstone of cleaning routines. However, the emergence of smart technology heralds a paradigm shift, with robots seamlessly supplanting conventional methods. Employing an Arduino controller, this endeavor introduces a sophisticated cleaning robot endowed with dust collection, mopping, and sanitizing capabilities. The robot's adaptability transcends myriad settings, encompassing bustling commercial hubs and intimate household domains. Featuring programmable functionalities, it navigates spaces autonomously, ensuring optimal cleanliness and hygiene. Upon activation, the robot operates autonomously, obviating the need for continuous human oversight. This initiative signifies a pivotal stride in the realm of cleaning automation. By amalgamating state-of-the-art technology into everyday cleaning tasks, it furnishes efficiency, dependability, and elevated hygiene standards. The advent of the smart floor cleaning robot heralds a promising solution for modernizing cleaning practices and addressing the evolving requisites of diverse environments.

Keywords: Customized Robotic Cleaners,IOT, Arudino

1.Introduction:

Effective cleaning and sanitizing helps and protect the health of the human beings directly and indirectly. Also, cleaning and sanitizing prevents the pest infestations by reducing residues that can attract and support bees, pests etc. It also improves the self life of the floor, walls etc due to regular cleaning and maintenance. In recent years, most of the people prefer to go by rail or bus, hence these locations are strewn with things like cold drink bottles, biscuit coverings, and other items. As a result, frequent cleaning of bus stops and train stations is required.

There is no one cleaning procedure that works for everything, and the equipment needs to be easy to use. Due to the physical demands of cleaning work and the necessity for procedures for systematic ergonomics evaluation, the use of robots is growing among older and busier populations that are experiencing a labor shortage. The goal of this project is to construct a robotic vehicle that can clean floors on its own. The designed machines have the ability to clean the entire floor on their own.

It is not an easy task to conduct research and development for an autonomous mobile robot and a manual phone application control prototype that can vacuum a room or even the full house.

As a result, some functional requirements that might enhance the robot's performance were overlooked because of their mechanical consequences or inherent complexity. These autonomous or semi-autonomous robots carry out tasks that benefit both people and machinery. The following requirements were discovered in order to keep our robot as basic as possible while still accomplishing the original objectives, which were to create an autonomous vacuum cleaner robot that could randomly move across a room or a house with the least amount of human assistance:

- Light sensing
    - Avoiding obstacles
- Floor detection
- Collision detection
- Real-time clock System turned on if required

Some of the anticipated behaviors that will be programmed into the robot are in line with these standards. Additional actions that will improve the robot's overall performance.

Materials flow and movement, as well as the movement of any other parts inside the device, are managed by mechanical control mechanisms. Actuators are employed to operate a mechanism that in turn manages a specific component of the apparatus.

Sensors are detecting devices that send and receive signals. Based on the signals they receive, they gather diverse environmental data, which is then sent to a microcontroller to determine how the machines should operate. The microcontroller, which has a programmed on it, is the brains of the robot. It is coupled to sensors for input and actuators for output. The controller removes the environment value based on what it receives.

Additionally, the device uses two kind of cleaning in which brushes and vacuum technology are used to clean the floor. The device features an internal power supply, enabling it to also clean the floor. The "Smart Floor Cleaning Robot" project's architecture is designed to make life easier for
those who cannot afford the current generation of autonomous floor cleaners on the market. In general, there are two kinds of controllers: one is a PID-based controller, while the other is a continuous controller, PID controllers are more sophisticated and adjust based on the current, whereas continuous controllers are more straightforward and less effective state and produces effective outcomes.

Controlling provides various advantages over human powered work. Following are some of them:-

- It gives accurate results and eliminates possibility of manual error.
- It is very first and efficient and the control system used in industries are 100 times efficient than human work.
- In some part of the work areas it lessens the human efforts. Washing machine comes under this category.
- It also plays the great role in bringing entertainment in human life in different work. Television is the live example of these type of robots
- There are two modes available here:

1. Switching; 2. RTC-using. The ESP8266 microcontroller, battery, RTC (Real Time Circuit), keypad, sonar sensor (Ultrasonic sensor), encoder sensor, motor driver, drive train, LCD display, buzzer, relay, and pump are all included in the basic block diagram.

Advantages of Customized Robotic Cleaners:

1. Tailored to Specific Needs: Customized robotic cleaners can be designed to meet the specific cleaning needs of different environments, such as homes, offices, hospitals, or warehouses.

2. Increased Efficiency: By optimizing the design for a particular task, customized robotic cleaners can often clean more efficiently and effectively than generic models.

Disadvantages of Customized Robotic Cleaners:

1. Higher Cost: Designing and manufacturing customized robotic cleaners can be more expensive than purchasing off-the-shelf models, especially for small-scale production runs.
2. Longer Development Time: Developing a customized robotic cleaner from scratch requires time and resources for design, prototyping, and testing, which can delay the product's availability compared to existing options.

Applications of Customized Robotic Cleaners:

1. Residential Cleaning: Customized robotic cleaners can be designed for use in homes to automate routine cleaning tasks such as vacuuming carpets, mopping floors, or cleaning windows.
2. Commercial Cleaning: In commercial settings such as offices, hotels, or retail spaces, customized robotic cleaners can help maintain cleanliness and hygiene efficiently and cost-effectively.

IOT

IoT stands for the Internet of Things. It refers to the network of interconnected devices that can communicate and exchange data with each other over the internet without human intervention. These devices can range from everyday objects like household appliances and wearable devices to industrial machines and sensors in smart cities.
Here's a simplified explanation of how IoT works:
Sensors and Devices: IoT begins with the deployment of sensors or smart devices. These devices are embedded with sensors, actuators, and other necessary components to collect data or perform specific actions.
Connectivity: The data collected by these devices is transmitted over the internet or other communication networks like Wi-Fi, Bluetooth, or cellular networks. This connectivity allows the devices to communicate with each other and with centralized systems.
Data Processing and Analysis: The data collected from IoT devices is processed and analyzed in real-time or near real-time. This can happen locally on the device itself, on a gateway device, or in the cloud. Advanced analytics and machine learning algorithms may be applied to derive insights from the data
Action and Automation: Based on the insights gained from data analysis, actions can be triggered automatically. For example, a smart thermostat can adjust the temperature based on occupancy patterns detected by motion sensors, or an industrial machine can adjust its operations to optimize efficiency based on real-time data from sensors.
User Interface and Control: Users can interact with IoT devices and systems through various interfaces such as mobile apps, web dashboards, or voice commands. This allows users to monitor and control connected devices remotely.
Security and Privacy: Since IoT involves the collection and transmission of sensitive data, ensuring security and privacy is crucial. Encryption, authentication, access control, and other security measures are implemented to protect data and prevent unauthorized access.

2.LITERATURE REVIEW

Vacuum, mopping, and hybrid robots are the three types of floor cleaning robots according to research by Smith et al. (2021). While mopping robots use water dispensing and mop designs, vacuum robots clean floors using suction mechanisms. Robots that are hybrids combine the capabilities of mopping and vacuuming to provide thorough cleaning.

In floor cleaning robots, a variety of sensors are essential components. According to Johnson et al.'s (2019) research, infrared sensors allow for obstacle identification and navigation. According to Chen et al. (2020), ultrasonic sensors help in object avoidance and distance measurement. Kim et al. (2022) studied camera-based sensors, which offer visual feedback for localization and mapping.

Various floor cleaning robot types use different cleaning mechanisms. Li et al. (2018) found that brush systems work well for gathering trash from vacuum robots. According to Wang et al. (2021), suction systems improve dust removal. Liu et al. (2023) examined mop designs and found that they maximize water dispersion and cleaning effectiveness.

Robots that clean floors efficiently are made possible by control systems. Zhang et al. (2017) provide details on motor control algorithms that control cleaning and propulsion. Wu et al. (2022) investigated feedback loops, which increase navigation accuracy. According to Tan et al.'s (2019) research, autonomous operation makes intelligent cleaning patterns possible.

User interfaces are essential for smooth communication between floor cleaning robots and humans. According to Brown et al. (2020), scheduling and remote control functions are available in smartphone apps. According to a study by Garcia et al. (2021), voice instructions improve user convenience. According to Lee et al. (2018), safety elements guarantee user protection when a robot is operating.

Eco-friendly floor cleaning robots must have energy-efficient designs. Zhao et al.'s (2019) study on battery management systems maximizes power consumption. Park et al. (2023) investigated eco-friendly materials that lessen their impact on the environment. Comparative studies assess energy usage between various robot models;

3.WORKING OF ROBOTIC MOP CLEANER

The motor shield is attached to a 9V battery, which is the only power supply. The battery's negative is connected to GND and its positive to +M. Two of the four BO motors that the robot uses, motors R and L, are linked to the M4 and M3 ports of the motor drive, respectively.

A breadboard at the robot's front is connected to the M1 port of the motor drive, which in turn is connected to the other two BO motors (motors BR and BL).

Additionally, the ultrasonic sensor is mounted on the bread board.

The motorBL and motorBR each have two circular brushes attached to them that rotate in opposing directions. The robot's tail is secured by two tiny, immobile As a last step, brushes with intake remove any last bits of dust in their path.

Every piece of hardware is fastened to the chassis's base structure, which is supported by two wheels and one supporting wheel.

The Trig pin of Ultrasonic is connected to the 11th pin on the Arduino, the Echo pin is connected to the 12th pin, the voltage pin to the 5V pin and the Ground pin to the ground pin. The Echo pin and the Trig pin allow the Arduino to communicate with the sensor. Power is delivered to the sensor through the voltage and Ground pins, and the Trig and Echo pins allow it to send and receive data with the Arduino.

The robot needs to be first connected to the smartphone via WIFI in order to begin operating. An Android app will be used to control the robot's functions. The following is the application's user interface, along with the labeling of the buttons that it will be used for:

3.1 The actual steps involved in operating the robot are as follows:

1. First, the electricity is turned on.
2. The smartphone running the Android app needs to establish a WIFI connection with the robotic system once power is restored.

3. After the connection is established, the red LED light on the ESP8266 will go out, allowing you to verify whether the connection was successful. By doing this, the devices' successful connection is guaranteed.

4. After the connection is established, the water level needs to be guaranteed. In the event that the water level is low, turn off the electricity, fill the water tank to the brim, and then repeat steps I through III.
5. Press the Mop On/Off button to turn on the mop after this is finished. Here, the robot is signaled by the app to begin the mop as soon as the button is pressed. After the signal is processed, a crucial task is to get the mop going. When the function is activated,
The required relay switch is signaled by the controller to turn on, and the mop begins to rotate.

6. Next, you need to press the Pump button. When this button is pressed, a signal is sent to the controller telling it to turn on the pump and fan. However, the controller activates the float sensor's check function to see if there is the right amount of water in the tank before turning on these two parts. The float sensor signal will be utilized to confirm that there is the appropriate amount of water in the tank. Following this confirmation, the start pump and The fan function will begin operating, sending a signal to the required relay to turn on these two parts.

7.After that, the user can move the robot in any direction using the direction keys. Whichever key the user presses, the signal travels instantly to the robot, and the controller processes the signal and transmits a required signal to the Motor driver module which will control the movement of the robot.

ARDUINO ATMEGA2560 :
The Arduino Mega2560 can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack.
Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts. The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.
The power pins are as follows:
• VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
• 5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
• 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
• GND. Ground pins.

Memory: The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the EEPROM library).
Each of the 54 digital pins on the Mega can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions.
They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms.

In addition, some pins have specialized functions:

• Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .

• External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.

• PWM: 0 to 13. Provide 8-bit PWM output with the analogWrite() function.

• SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Duemilanove and Die cimila.

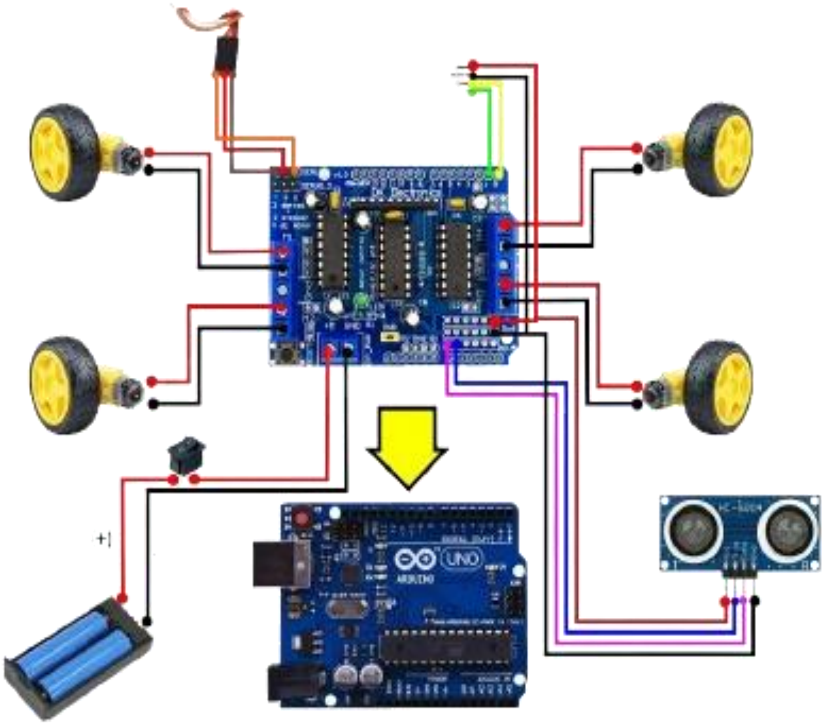
• LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

• I 2C: 20 (SDA) and 21 (SCL). Support I2C (TWI) communication using the Wire library (documentation on the Wiring website). Note that these pins are not in the same location as the I2C pins on the Due milanove. The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values).
By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and analog Reference() function.
There are a couple of other pins on the board:

• AREF. Reference voltage for the analog inputs. Used with analog Reference().

• Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

ARDUINO MEGA 2560



DIGITAL PINS:

Pin	Function	Type	Description
1	D21/SCL	Digital Input/I2C	Digital input 21/I2C Dataline
2	D20/SDA	Digital Input/I2C	Digital input 20/I2C Dataline
3	AREF	Digital	Analog Reference Voltage
4	GND	Power	Ground
5	D13	Digital/GPIO	Digital input 13/GPIO
6	D12	Digital/GPIO	Digital input 12/GPIO
7	D11	Digital/GPIO	Digital input 11/GPIO
8	D10	Digital/GPIO	Digital input 10/GPIO
9	D9	Digital/GPIO	Digital input 9/GPIO
10	D8	Digital/GPIO	Digital input 8/GPIO
11	D7	Digital/GPIO	Digital input 7/GPIO
12	D6	Digital/GPIO	Digital input 6/GPIO
13	D5	Digital/GPIO	Digital input 5/GPIO
14	D4	Digital/GPIO	Digital input 4/GPIO

Pin	Function	Type	Description
15	D3	Digital/GPIO	Digital input 3/GPIO
16	D2	Digital/GPIO	Digital input 2/GPIO
17	D1/TX0	Digital/GPIO	Digital input 1 /GPIO
18	D0/Tx1	Digital/GPIO	Digital input 0 /GPIO
19	D14	Digital/GPIO	Digital input 14 /GPIO
20	D15	Digital/GPIO	Digital input 15 /GPIO
21	D16	Digital/GPIO	Digital input 16 /GPIO
22	D17	Digital/GPIO	Digital input 17 /GPIO
23	D18	Digital/GPIO	Digital input 18 /GPIO
24	D19	Digital/GPIO	Digital input 19 /GPIO
25	D20	Digital/GPIO	Digital input 20 /GPIO
26	D21	Digital/GPIO	Digital input 21 /GPIO

ANALOG PINS:

Pin	Function	Type	Description
1	NC	NC	Not Connected
2	IOREF	IOREF	Reference for digital logic V - connected to 5V
3	Reset	Reset	Reset
4	+3V3	Power	+3V3 Power Rail
5	+5V	Power	+5V Power Rail
6	GND	Power	Ground
7	GND	Power	Ground
8	VIN	Power	Voltage Input
9	A0	Analog	Analog input 0 /GPIO
10	A1	Analog	Analog input 1 /GPIO
11	A2	Analog	Analog input 2 /GPIO
12	A3	Analog	Analog input 3 /GPIO
13	A4	Analog	Analog input 4 /GPIO
14	A5	Analog	Analog input 5 /GPIO
15	A6	Analog	Analog input 6 /GPIO
16	A7	Analog	Analog input 7 /GPIO
17	A8	Analog	Analog input 8 /GPIO
18	A9	Analog	Analog input 9 /GPIO
19	A10	Analog	Analog input 10 /GPIO
20	A11	Analog	Analog input 11 /GPIO
21	A12	Analog	Analog input 12 /GPIO
22	A13	Analog	Analog input 13 /GPIO
23	A14	Analog	Analog input 14 /GPIO
24	A15	Analog	Analog input 15 /GPIO

The overall summary of a Arduino ATMEGA2560 :

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Based on the ATmega2560 microcontroller, the Arduino Mega 2560 is a flexible platform appropriate for a broad spectrum of applications, from professional use to hobby projects. It encourages users to get creative and realize their ideas in the fields of electronics and embedded systems with its many I/O pins, communication ports, and expandability choices. Whether you're an expert developer working on intricate projects or a novice learning the fundamentals of electronics and programming, the Mega 2560 offers a versatile and potent answer.

APPLICATIONS:

Robotics & Automation Applications: Robotic system and home automation project control is made easy with the Mega 2560's comprehensive I/O capabilities and communication ports.

Data Acquisition: Due to its analog inputs, it can be used to acquire sensor data for a range of applications, such as scientific research, industrial automation, and environmental monitoring.

Embedded Systems: When it comes to embedded systems that need sophisticated processing power and networking possibilities, the board can act as their brain.

Prototyping: Because of its adaptability and user-friendliness, the Arduino Mega 2560 is frequently used for electronic project prototyping, enabling developers to test ideas and concepts fast.

For educational purposes: Because of its ease of use and wealth of online resources, it is frequently used in school settings to teach electronics, programming, and embedded systems design.

Art Projects: Using the Mega 2560, designers and artists can produce kinetic sculptures, interactive installations, and other electronic artworks.

NODE MCU:

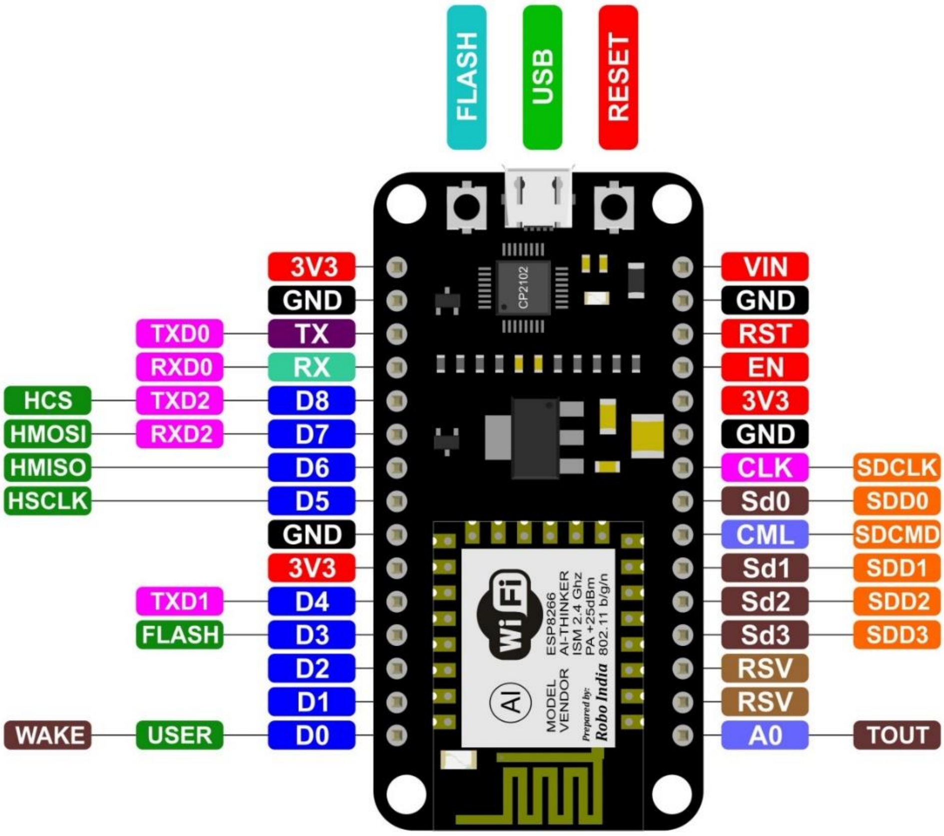
The Node MCU is an open source firmware and development kit that helps you to prototype your IoT product with ArduinoIDE or in few Lau script lines.

It includes firmware which runs on the ESP8266 Wi-Fi SoC. And hardware which is based on the ESP-12 module.

In this tutorial we explain how to use NodeMCU with Arduino IDE

- Open up the Arduino IDE.
- Go to File -> Preferences -> Additional Boards Manager
- URLs: *http://arduino.esp8266.com/stable/package_esp8266com_index.json* -> click OK
- Close the IDE and open it up again.
- Go to Tools -> Board (where you'd select your version of Arduino) -> Boards Manager, find the ESP8266 and click Install. You now should be able to use the ESP8266 as an Arduino. Simply select the NODEMCU 1.0 as your board with Port and you should be ready to code.

Now, with ESP8266 board installed to Arduino IDE, we can program NodeMCU using Arduino IDE directly.



NODEMCU ESP8266:

NODEMCU ESP8266:

- The ESP8266 series, or family, of Wi-Fi chips is produced by Espressif Systems, a fabless semiconductor company operating out of Shanghai, China. The ESP8266 series presently includes the ESP8266EX and ESP8285 chips.
- **ESP8266EX** (simply referred to as ESP8266) is a system-on-chip (SoC) which integrates a 32-bit Tensilica microcontroller, standard digital peripheral interfaces, antenna switches, RF balun, power amplifier, low noise receive amplifier, filters and power management modules into a small package.

- It provides capabilities for 2.4 GHz Wi-Fi (802.11 b/g/n, supporting WPA/WPA2), general-purpose input/output (16 GPIO), Inter-Integrated Circuit (I²C), analog-to-digital conversion (10-bit ADC), Serial Peripheral Interface (SPI), I²S interfaces with DMA (sharing pins with GPIO), UART (on dedicated pins, plus a transmit-only UART can be enabled on GPIO2), and pulse-width modulation (PWM).

- The processor core, called "L106" by Espressif, is based on Tensilica's Diamond Standard 106Micro 32-bit processor controller core and runs at 80 MHz (or overclocked to 160 MHz). It has a 64 KiB boot ROM, 32 KiB instruction RAM, and 80 KiB user data RAM. (Also, 32 KiB instruction cache RAM and 16 KiB ETS system data RAM.)

- External flash memory can be accessed through SPI. The silicon chip itself is housed within a 5 mm × 5 mm Quad Flat No-Leads package with 33 connection pads — 8 pads along each side and one large thermal/ground pad in the center.

- The ESP8266 is a System on a Chip (SoC), manufactured by the Chinese company Espressif. It consists of a Tensilica L106 32-bit **micro controller** unit (MCU) and a **Wi-Fi transceiver**. It has **11 GPIO pins*** (General Purpose Input/Output pins), and an **analog input** as well. This means that you can program it like any normal Arduino or other microcontroller. And on top of that, you get Wi-Fi communication, so you can use it to connect to your Wi-Fi network, connect to the Internet, host a web server with real web pages, let your smartphone connect to it, etc ...

- The possibilities are endless! It's no wonder that this chip has become the most popular IOT device available

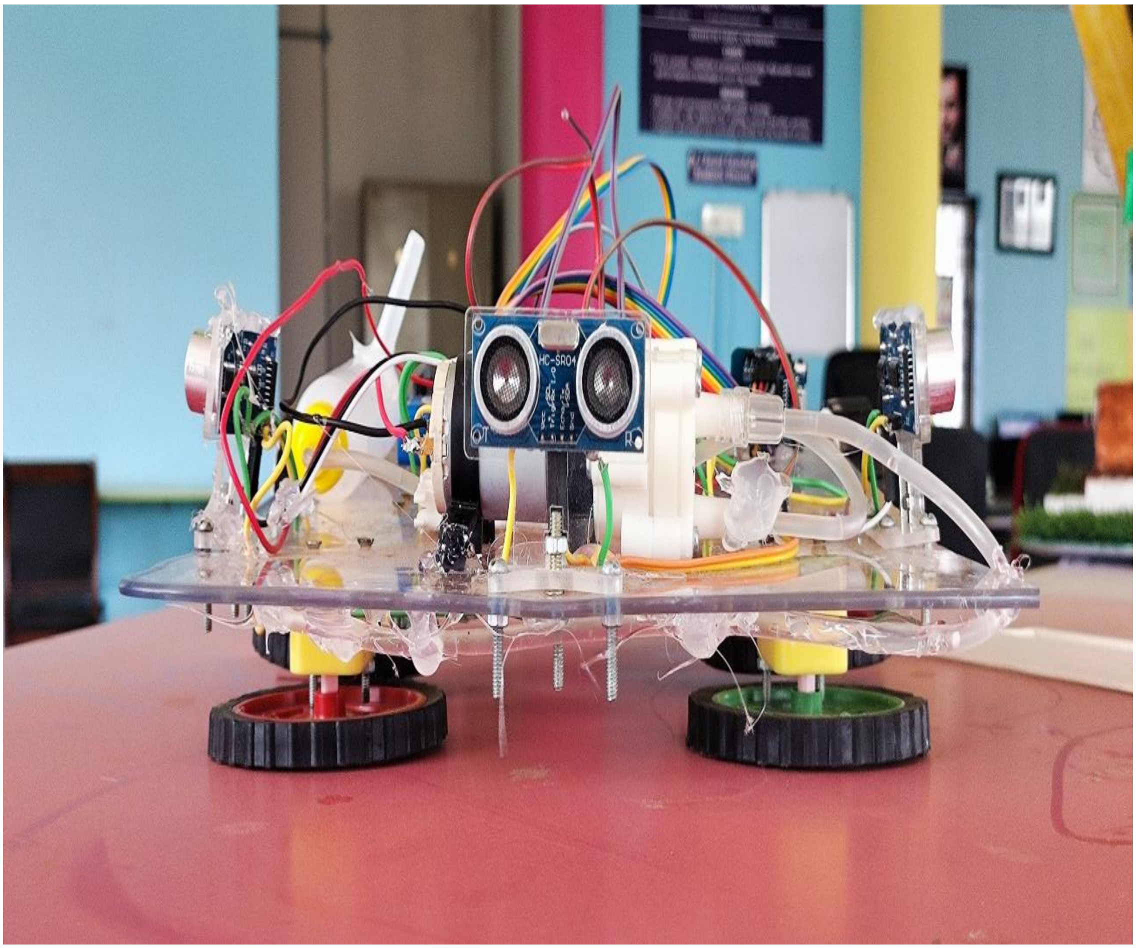
4.Data collection and interpretation

s.no	Image of the component	Component name	Component cost	No.of component s	Total cost
1		Generic: 12V Centre Shaft DC Geared Motor - Random RPM	360.46	1.00	360.46
2		Center Shaft Gear Motor L Clamp (Bracket)	20.43	2.00	40.86
3		L298N 2A Based Motor Driver Module	150.25	2.00	300.50
4		Mega 2560 ATmega2560-16AU Board without USB Cable for Arduino (Improved Version)	3170.03	1.00	3170.03
5		R385 6-12V DC Diaphragm Based Mini Aquarium Water Pump	199.33	1.00	199.33
6		Ultrasonic Sensor Range Finder Module HC-SR04	86.33	3.00	259.79
7		[Type 3] Acrylic Ultrasonic Sensor Mounting Bracket For HC-SR04	50.28	3.00	150.85

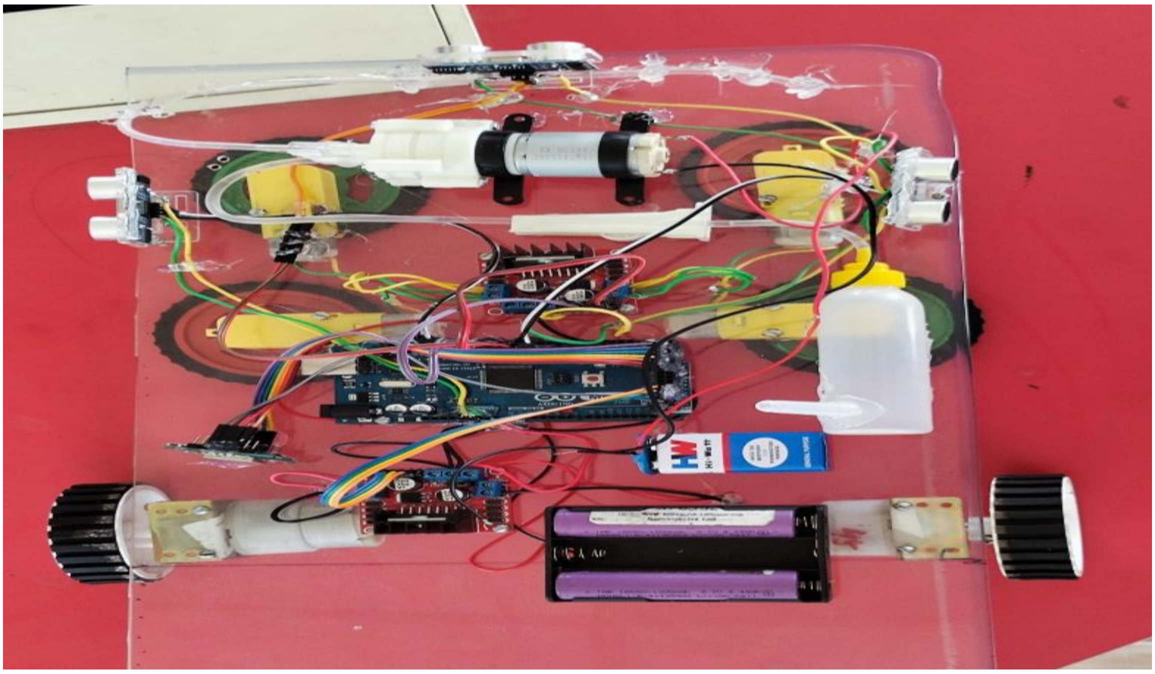
8		Robotic DC Wheels 7cm x 2cm - Pack of 1	32.69	2.00	65.38
9		BO Gear Motor Robot Wheels 7cm x 1cm - Pack of 1 / Green	6.06	4.00	24.25
10		Dupont Jumper Wire Ribbon Cable - 20cm/200mm / Male to Female (M-F) / Pack of 10	20.46	2.00	40.92
11		TP4056 (Micro USB with Current Protection) 1A Li-ion Battery Charging Module	28.43	1.00	28.43
12		Vigor: Premium BO Motor Straight Single Shaft up - 300rpm	65.00	4.00 pcs	260.00
13		Wemos D1 Mini - IOT ESP8266 Based Development Board	240.00	1.00 pcs	240.00
14		18650 3.7V 1200mAh Lithium-Ion Rechargeable Cell	52.50	4.00	210.24
15		18650 Battery In-Series Lithium Cell Holder Open Case with Wire - 18650 X 3 (Triple)	20.67	3.00	62.01
16		IV Set	200.00	1.00	200.00
17		$1^1/_2$ inch Nut and bolt	5.00	10.00	50.00
18		1 inch Nut and bolt	5.00	20.00	100.00

19		HW Battery	25.00	1.00	25.00
20		Gum bottle	10.00	1.00	10.00
21		Wire packet	60.00	1.00	60.00

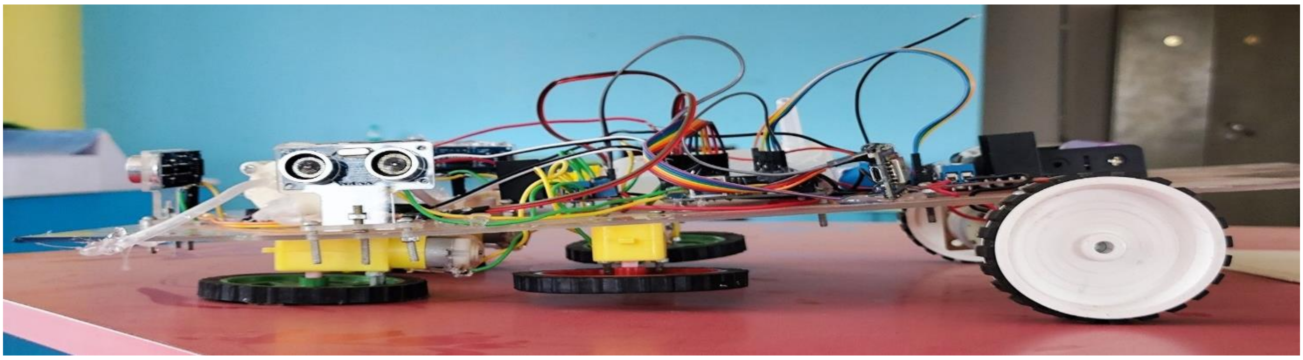
5.Data Analysis And Validation



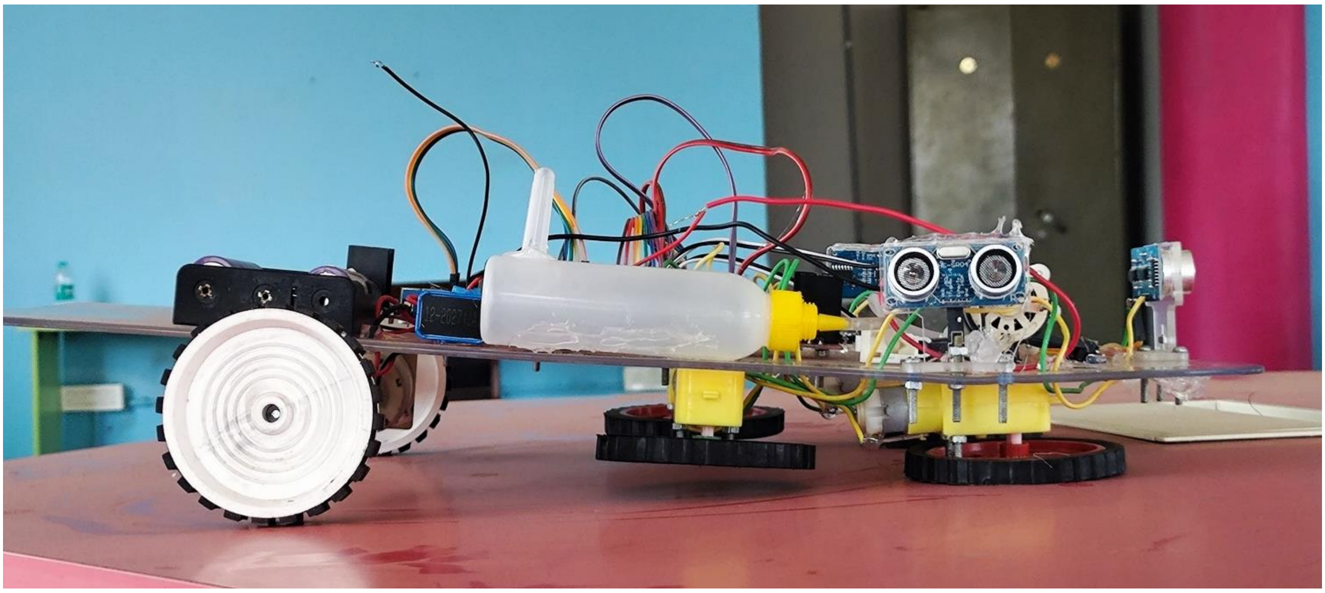
FRONT VIEW



TOP VIEW



LEFT SIDE VIEW



RIGHT SIDE VIEW

6.CONCLUSION

The created, developed, and manufactured Floor Cleaning Robot proves to be incredibly effective, economical, portable, and user-friendly. There is adequate water storage available.

Compared to other comparable robotic devices on the market, the cost of the robot is far lower. There is hardly any visible lag in communication between the mobile app and the robot thanks to their highly effective wireless connection. This robot is also a lot of fun to use because it functions similarly to a remote-control automobile. At the designated time, the machine can travel automatically across the floor and clean itself. The developed project includes a mechanism that stores a trigger time, which is then used to initiate the robotic vehicle's autonomous cleaning of the house.

This study makes floor cleaning more effective. The floor cleaner in the project will be simple to operate, save time, and function automatically to clean homes and offices while being economical because it is integrated with many devices including DC motor(s), ultrasonic sensors, etc. The cleaner will be able to navigate tight spaces and cover huge floor surfaces with ease thanks to a straightforward algorithm and program.

Consequently, the project is a complete success. However, if this idea is to be introduced to the market, the robot still needs a great deal of study and development because it is not yet flawless. It is outside the purview of this project to further develop this robot. However, this is a positive step for robotics in the future.

7.REFERENCES:

[1] Goon, L. H., Isa, A. N. I. M., Choong, C. H., and W. A. F. W. Othman. Using Arduino, create a simple automatic floor polisher robot(1), 17-23, International Journal of Engineering Creativity & Innovation, 2019.

[2] M. Kaur and P. Abrol. Design and development of a floor cleaner robot (automatic and manual). International Journal of Computer Applications, vol. 97, no. 1, 2014. (19).

[3] Sahu, N. K., Sharma, N. K., Khan, M. R., and Gautam, D. K. Floor Cleaner Comparative Study. Journal of Pure Applied and Industrial Physics, 8(12), pp. 233-236, 2018.

[4] Liu, C. C., Kang, Y. P., and Yu, S. N. Integration of hardware and software for a domestic stair cleaning robot. In the 2011 SICE Annual Conference (pp. 663-670). IEEE.

[5] M. Kukde, S. Nagpurkar, A. Dhakulkar, and A. Amdare. Automatic and manual vacuum cleaning robot. 5(02), 2196-2198, International Research Journal of Engineering and Technology (IRJET).

[6] Juang, H. S., & Lurrr, K. Y. Design and control of a two-wheel self-balancing robot using the arduino microcontroller board. In 2013 10th IEEE International Conference on Control and Automation (ICCA) (pp. 634-639). IEEE.

[7] Al-Busaidi, A. M. Development of an educational environment for online control of a biped robot using MATLAB and Arduino. In 2012 9th France-Japan & 7th Europe-Asia Congress on Mechatronics (MECATRONICS)/13th Int'l Workshop on Research and Education in Mechatronics (REM), 2012 (pp. 337-344). IEEE.

[8] Araújo, A., Portugal, D., Couceiro, M. S., & Rocha, R. P. Integrating Arduino-based educational mobile robots in ROS. Journal of Intelligent & Robotic Systems, 2015, 77(2), 281-298.

[9] Gargava, P., Sindwani, K., & Soman, S. Controlling an arduino robot using Brain Computer Interface. In Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization (2014, pp. 1-5). IEEE.

[10] Patil, D. A., Upadhye, M. Y., Kazi, F. S., & Singh, N. M. Multi-robot communication and target tracking system with controller design and implementation of swarm robot using Arduino. In 2015 International Conference on Industrial Instrumentation and Control (ICIC) (pp. 412-416). IEEE.