International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Smart Dust Cleaning Robot Using Arduino

A. Narayana Murthy[1], P.S.Kavya[2], N.P.N.Bindu[3], S.Naga Jyothi[4], K.Sarath Kumar[5], D.Veni[6]

[1]AssistantProfessor, Department of Electronics and Communication Engineering, VSM college of Engineering, Ramachandrapuram, Andhra Pradesh, India
[2,3,4,5,6]UG Students, Department of Electronics and Communication Engineering, VSM college of Engineering, Ramachandrapuram, Andhra Pradesh, India

ABSTRACT

The Smart Vacuum Cleaner project marks a breakthrough in automating domestic cleaning chores with its uniquely designed vacuum system. Featuring a custom vacuum tube optimized for airflow and suction, the device is powered by a robust motor driver ensuring consistent performance. Advanced sensors and control algorithms enable precise, autonomous navigation in complex home environments. The integration of a relay module meticulously governs the operational state, conserving power and extending lifespan by preventing unnecessary wear. Committed to energy efficiency and environmental responsibility, this project exemplifies modern convenience and technological advancement. It redefines home maintenance expectations, merging integrated electronics and smart design to create a practical, sustainable cleaning solution. The Smart Vacuum Cleaner stands as a vital asset in achieving a cleaner, smarter, and more efficient household.

KEYWORDS: Arduino board, DC motor, ESP32 Camera, vacuum pump, L298N Motor Driver, Ultrasonic Sensor, Battery, BLUETOOTH Module.

INTRODUCTION

The advent of smart dust cleaning robots represents a significant advancement in the field of robotics and automation. These robots are designed to autonomously navigate indoor environments, detecting and removing dust particles efficiently. Leveraging advancements in sensor technology, AI algorithms, and robotics, these devices promise to revolutionize cleaning processes in various settings, from homes to co mercurial spaces. With the integration of live monitoring capabilities, users can remotely supervise the cleaning progress in q real-time, then enhancing convenience and effectiveness. Smart dust cleaning robots incorporate a diverse range of technological components to achieve their objectives. This includes high resolution cameras for navigation and obstacle detection, sensor arrays for dust detection, and sophisticated algorithms for path planning and dust removal. However, several challenges persist, such as optimizing the energy efficiency of these Leveraging advancements in sensor technology, AI algorithms, and robotics, these devices promise to revolutionize cleaning processes in various settings, from homes to co mercurial spaces. With the integration of live monitoring capabilities, users can remotely supervise the cleaning progress in a real-time, then enhancing convenience and effectiveness. Smart dust cleaning robots incorporate a diverse range of technological components to achieve their objectives. This includes high resolution cameras for navigation and obstacle detection, sensor arrays for

dust detection, and sophisticated algorithms for path planning and dust removal. However, several challenges persist, such aoptimizing the energy efficiency of these robots, ensuring robustness in various environmental conditions, and integrating seamless communication for live monitoring. Addressing these challenges is crucial to realizing the full potential of smart dust cleaning robots. The field of smart dust cleaning robots has witnessed significant research and development efforts in recent years. Various studies have explored different aspects of robot design, including mobility, sensing capabilities, and cleaning mechanisms. Additionally, researchers have investigated novel approaches to live monitoring, such as cloud-based platforms and mobile applications, enabling users to interact with and control the robots remotely. These advancements underscore the growing interest and potential applications of smart dust cleaning robots in both residential and commercial settings. Looking ahead, the development of smart dust cleaning robots with live monitoring capabilities holds immense promise for revolutionizing the cleaning industry. Beyond enhancing convenience and efficiency for users, these robots have the potential to contribute to improved indoor air quality and health outcomes by effectively removing dust particles. Moreover, as technology continues to evolve, we can expect further innovations in robot design, sensor technology, and AI algorithms, driving the proliferation and adoption of smart cleaning solutions in diverse contexts.
However, it will be essential to address ethical considerations, such as privacy concerns related to live monitoring, to ensure the responsible deployment and use of these robots.

LITERATURE SURVEY

The landscape of domestic cleaning has long been dominated by traditional vacuum cleaners requiring manual operation. Users must physically maneuver the unit around their living spaces, a task that can be both time-consuming the and physically demanding. Despite the advent of automated robotic cleaners, these devices often exhibit limitations that stem from their rudimentary navigation systems. The majority of existing robotic vacuum cleaners operate on basic algorithms and set patterns, lacking the sophistication to adapt to new obstacles or changes in the environment. These automated systems tend to be reactive rather than proactive, with obstacle detection often leading to simple collision-based avoidance strategies. This can result in inefficient cleaning paths and missed areas, particularly around complex configurations such as furniture-rich rooms or dynamic environments that include pets and moving objects.

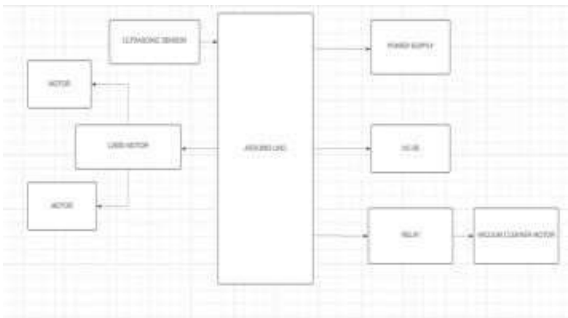
EXISTED METHOD

The landscape of domestic cleaning has long been dominated by traditional vacuum cleaners requiring manual operation. Users must physically maneuver the unit around their living spaces, a task that can be both
time-consuming and physically demanding. Despite the advent of automated robotic cleaners, these devices often exhibit limitations that stem from their rudimentary navigation systems. The majority of existing robotic vacuum cleaners operate on basic algorithms and set patterns, lacking the sophistication to adapt to new obstacles or changes in the environment. These automated systems tend to be reactive rather than proactive, with obstacle detection often leading to simple collision-based avoidance strategies. This can result in inefficient cleaning paths and missed areas, particularly around complex configurations such as furniture-rich rooms or dynamic environments that include pets and moving

objects. to unnecessary energy consumption.

PROPOSED METHOD

The proposed Smart Vacuum Cleaner system is designed to be a leap forward in home cleaning technology, addressing the deficiencies observed in the existing systems. It marries the convenience of automation with the intelligence of smart technology, resulting in a sophisticated,
efficient, and user-friendly cleaning companion. At the heart of the proposed system is a custom-designed vacuum tube, developed to optimize suction and dirt collection across various types of surfaces. The agility of this system is powered by a high- torque motor driver, which propels the unit with the required precision and consistency. To enhance user experience, a Bluetooth module has been enhance user experience, a Bluetooth module has been integrated to offer not only manual control options but also the potential for smart home connectivity. This allows users to seamlessly integrate the vacuum cleaner into their existing smart home ecosystems and control it through their smartphones or voice-assistant devices. One of the cornerstone innovations in the proposed vacuum cleaner is its advanced obstacle detection and navigation system. Unlike traditional robotic cleaners that often bump into objects before changing direction, our system uses a trio of ultrasonic sensors to detect obstacles from a distance. This enables proactive avoidance maneuvers, which not only protect furniture and other items from accidental bumps but also result in a more efficient cleaning route. The sensors map the room in real-time, allowing the vacuum cleaner to adapt its path instantly to changes in the environment. Furthermore, the proposed system incorporates a relay to intelligently manage the vacuum cleaner's power state. This feature enables the unit to conserve energy by operating only when cleaning is required and shutting down during idle periods. The relay also facilitates automatic recharging, where the vacuum cleaner can navigate back to its docking station when the battery is low or the 1cl2eaning cycle is complete. The Smart Vacuum Cleaner is a solution thoughtfully crafted to elevate the standards of automated cleaning. Its design takes into account not just the need for cleanliness but also the user's desire for a hassle-free and environmentally friendly product. With the ability to navigate and clean efficiently, respond to user commands, and integrate within a smart home environment, the proposed system represents a significant upgrade from the existing systems. It aims to redefine the cleaning experience, making it smarter, more effective, and more attuned style.



Software

The Arduino IDE, or Integrate development Environment, is a software tool used for programming Arduino microcontrollers. It provides a simple interface for writing, compiling, and uploading code to Arduino boards. The IDE includes a text editor, compiler, and uploader. It supports a simplified version of C++ programming language, making it accessible for beginners. Users can easily install libraries to extend the functionality of their projects. The IDE also features a serial monitor for debugging and testing. It's open-source software, fostering a vibrant community of developers and resources. Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The tool barbuttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor. NB: Versions of the Arduino Software (IDE) prior to 1.0 saved sketches with the extension.It is possible to open these files with version 1.0, you will be prompted to save the sketch with the .into extension on save.

Verify: Checks your code for errors compiling it.

Upload: Compiles your code and uploads it to the configured board. See uploading below for details.

Note: If you are using an external programmer with your board, you can hold down the "shift" key on your computer when using this icon. The text will change to "Upload using Programmer"

New: Creates a new sketch.

Open

Allows to load a sketch file browsing through the computer drives and folders.

Open Recent

Provides a short list of the most recent sketches, ready to be opened.

Sketchbook

Shows the current sketches within the sketchbook folder structure; clicking on any name opens the corresponding sketch in a new editor instance.

Examples

Any example provided by the Arduino Software (IDE) or library shows up in this menu item. All the examples are structured in a tree that allows easy access by topic or library.

Close

Closes the instance ofthe Arduino Software from which it is clicked.

Save

Saves the sketch with the current name. Ifthe file hasn't been named before, a name will be provided in a "Save as." window.

Save as...

Allows to save the current sketch with a different name.

Page Setup

It shows the Page Setup window for printing.

Print

Sends the current sketch to the printer according to the settings defined in Page Setup.

Preferences

Opens the Preferences window where some settings of the IDE may be customized, as the language of the IDE interface.

Quit

Closes all IDE windows. The same sketches open when Quit was chosen will be automatically reopened the next time you start the IDE. Since version 1.0.1, the Arduino Software (IDE) has been translated into

30+ different languages. Bydefault, the IDE loads in the language selected by your operating system. (Note: on Windows and possibly Linux, this is determined by the locale setting which controls currency and date formats, not by the language the operating system is displayed in.) If you would like to change the language manually, start the Arduino Software (IDE) and open the Preferences window. Next to the Editor Language there is a dropdown menu of currently supported languages. Select your preferred language from the menu, and restart the software to use the selected language. If your operating system language is not supported, the Arduino Software (IDE) will default to English. You can return the software to its default setting of selecting its language based on your operating system by selecting System Default from the Editor Language dropdown. This setting will take effect when you restart the Arduino Software (IDE). Similarly, after changing your operating system's settings, you must restart the Arduino Software (IDE) to update it to the new default language.

HARDWARE COMPONENT ARDUINO UNO

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and reset button. It contains everything needed to support the microcontroller; it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDIUSB- to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to- serial converter. "Uno" means one in Italian and is named to mark the upcoming release of Arduino.

Fig. Arduino uno

The Arduino Uno is a popular microcontroller board based on the ATmega328P microcontroller. Here are its key specifications:

Microcontroller

- Microcontroller: ATmega328P
- Operating Voltage: 5V
- Input Voltage (recommended): 7- 12V
- Input Voltage (limit): 6-20V

Digital I/O Pins

- Number of Digital I/O Pins: 14 (of which 6 can provide PWM output)
- PWM Digital I/O Pins: 6 (pins 3, 5, 6, 9, 10, 11)

Analog Input Pins

Number of Analog Input Pins: 6 (A0 to A5)

Current

DC Current per I/O Pin: 20 mA

DC Current for 3.3V Pin: 50 mA

Memory

Flash Memory: 32 KB (ATmega328P) of which 0.5 KB used by bootloader

SRAM: 2 KB (ATmega328P)

The Arduino Uno is a popular microcontroller board based on the ATmega328P microcontroller. Here are its key specifications:

Microcontroller

Microcontroller: ATmega328P

Operating Voltage: 5V

Input Voltage (recommended): 7- 12V

Input Voltage (limit): 6-20V

Digital I/O Pins

Number of Digital I/O Pins: 14 (of which 6 can provide PWM output)

PWM Digital I/O Pins: 6 (pins 3, 5,
6, 9, 10, 11)

Analog Input Pins

Number of Analog Input Pins: 6 (A0 to A5)

Current

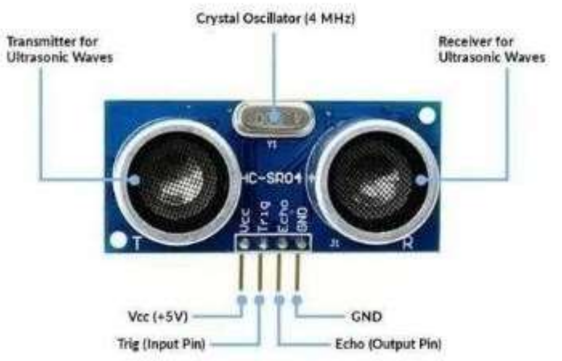
DC Current per I/O Pin: 20 mA

DC Current for 3.3V Pin: 50 mA

Memory

Flash Memory: 32 KB (ATmega328P) of which 0.5 KB used by bootloader

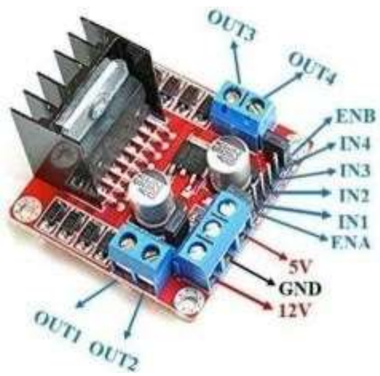
SRAM: 2 KB (ATmega328P)



Ultra Sonic Sensor

Working Principle: Ultrasonic sensors operate by emitting ultrasonic sound waves (typically at a frequency higher than the human ear can detect, usually above 20 kHz). These sound waves travel through the air and reflect back to the sensor when they encounter an object. The sensor then measures the time it takes for the sound waves to return and calculates the distance to the object based on the speed of sound.

Applications: Ultrasonic sensors are widely used in various applications due to their ability to measure distances accurately without physical contact. Common applications include: Obstacle detection in autonomous robots and vehicles. Level measurement in tanks and silos for liquids and solids. Proximity sensing in automated systems for object detection and avoidance. Advantages and Limitations: Advantages: Non-contact measurement: They can measure distances without physically touching the object, which is useful for delicate or hazardous materials .Versatility: Can work in various environments, including those with dust, light, or smoke, where optical sensors might fail. Wide range of detection: Effective over a range of distances from a few centimeters to several meters.Limitations:Surface dependency: The accuracy can be affected by the texture and angle of the object's surface.
Environmental factors:Temperature and humidity can influence the speed of sound, thus affecting accuracy.
Obstruction: Objects that absorb sound waves (e.g., soft fabrics) or have complex shapesmay not reflect sound waves effectively, leading to inaccurate readings.



L298N Motor Driver

Dual H-Bridge Configuration: The L298N motor driver is designed with a dual H-Bridge configuration, allowing it to control two DC motors independently or one stepper motor. Each H-Bridge consists of four switches (transistors or MOSFETs) that can change the direction of the current through the motor, enabling bidirectional control.

Current and Voltage Specifications: The L298N can handle motor supply voltages from 5V to 46V and can deliver a continuous current of up to 2A per channel (with peak currents up to 3A). This makes it suitable for a wide range of small to medium-sized motors commonly used in robotics and hobby projects.

Integrated Features: The L298N includes several integrated features that simplify motor control. It has built- in diodes for back EMF protection, which safeguard the driver and the rest of the circuit from voltage spikes generated when the motor changes speed or direction. Additionally, the module often comes with onboard 5V regulators, which can power the logic circuitry, and it has enable pins for each channel to control motor activation via PWM signals, allowing for precise speed and torque control.

DC Motor

Principle of Operation: DC motors operate based on the principle of converting direct current electrical energy into mechanical energy. This is achieved through the interaction of magnetic fields

generated by the stator (stationary part) and the armature (rotating part). When current flows through the armature windings, it creates a magnetic field that interacts with the field of the stator, producing a torque that causes the armature to rotate.
Types of DC Motors: There are several types of DC motors, each with unique characteristics and applications:
Brushed DC Motors: These are the traditional type of DC motors that use brushes and a commutator to deliver current to the motor windings. They are simple, cost-effective, and suitable for many general-purpose applications.

ESP 32 Camera

The ESP32-CAM is a compact, low-cost module that combines an ESP32-S microcontroller with a camera. Here are three key points about the ESP32-CAM:

- Integrated then for Camera and Microcontroller: The ESP32-CAM features an OV2640 camera module capable of capturing images and video at various resolutions. It is built on the ESP32-S microcontroller, which provides dual-core processing, Wi-Fi, and Bluetooth connectivity, making it suitable for a wide range of IoT applications involving image and video capture, such as security cameras, home automation, and surveillance systems.
- lExpandable Storage and Peripheral

Support: The module includes a microSD card slot, allowing for expandable storage to save captured images and videos. It also supports UART, SPI, and I2C interfaces, making it versatile for integrating with other sensors and peripherals. This flexibility enables the development of complex projects, such as time-lapse photography, facial recognition, and automated monitoring systems.

- lOpen-Source and Community Support: The ESP32-CAM is supported by a large community and extensive open-source libraries, particularly through the Arduino IDE and ESP-IDF (Espressif IoT Development Framework). This support makes it easier for developers to find resources, tutorials, and code examples, facilitating rapid development and prototyping of projects. The vibrant community also means frequent updates and shared projects, enhancing the module's functionality and applications.

Battery

Automotive Applications:

Primary Use: The 12-volt (12V) system is the standard for most automotive electrical systems, including cars, motorcycles, and some boats.

Components Powered: It powers various components such as the starter motor, headlights, dashboard electronics, and entertainment systems.

Battery: Vehicles typically use a 12V lead- acid battery for starting the engine and powering electrical accessories when the engine is off.

Renewable Energy Systems:

Solar Power: 12V batteries are commonly used in small-scale solar power setups, such as solar panels for RVs, boats, and off-grid cabins.

Energy Storage: These systems often include a 12V deep-cycle battery to store energy collected from solar panels for use during periods without sunlight.

Inverters: 12V systems frequently use inverters to convert the DC power from batteries to AC power for household appliances.

Consumer Electronics and DIY Projects:

Versatility: 12V is a common voltage for a wide range of consumer electronics and DIY electronics projects.

Compatibility: Many gadgets, from LED strip lights and security cameras to portable refrigerators and small motors, are designed to operate on 12V.

Accessibility: The widespread availability of 12V power supplies and batteries makes it easy for hobbyists to find components and build projects without complex power requirements.

These points illustrate the versatility and importance of 12V systems across different fields, highlighting their crucial role in automotive, renewable energy, and consumer electronics applications.

Advantages

1. The robot can efficiently clean dust and debris from various surfaces, ensuring a cleaner environment.
2. By using Arduino, the robot can be programmed to perform cleaning tasks autonomously, reducing the need for human intervention.
3. The robot can navigate through the cleaning area intelligently, avoiding obstacles and covering the entire space effectively.
4. Live monitoring allows users to observe the robot's performance in real-time, ensuring that cleaning tasks are being carried out effectively and efficiently.
5. With live monitoring capabilities, users can remotely control the robot's movements and cleaning functions,providing flexibility and convenience.

Applications

1. The robot can be used in homes.
2. Offices, restaurants, hotels, and other commercial establishments.
3. In hospitals, clinics, and other healthcare settings, the robot can help.
4. Schools, colleges, and universities can use these robots to keep classrooms, corridors, and other areas clean

EXPERIMENTAL RESULT

In summary, the project's goals were met by developing a smart vacuum cleaner that is both efficient and user-friendly, addressing the cleaning needs of modern households. The outcomes confirm that the components were successfully integrated and demonstrate how the system enhances the cleaning process through automation, improved user interaction, and energy efficiency. The implementation of advanced navigation algorithms and a robust sensor suite has allowed the vacuum cleaner to operate autonomously with minimal human intervention, navigating around obstacles with precision and adapting to different floor types seamlessly.

The integration of a user-friendly interface, accessible via both the device itself and a mobile application, has significantly improved the user interaction aspect of the vacuum cleaner. Users can now customize cleaning schedules, select specific cleaning modes, and even track the vacuum's progress in real time, all from the convenience of their smartphones. This level of control and transparency has not only increased user satisfaction but also ensured that the vacuum cleaner can be easily integrated into the daily lives of users.

Conclusion

In conclusion, the smart vacuum cleaner system represents a significant advancement in domestic technology, featuring effective integration of components like ultrasonic sensors, a high-efficiency motor, and IoT connectivity. By utilizing these technologies, the project has demonstrated that it is both feasible and effective to enhance traditional cleaning routines with automation, improving user control, mobility, and communication within a household. The project has confirmed the functionality and performance of the smart vacuum cleaner system through extensive testing and evaluation, demonstrating its capability to significantly enhance user convenience and efficiency while offering superior cleaning performance. The system is a versatile tool that can improve the quality of life for its users by offering freedom from manual cleaning tasks and incorporating seamlessly into the smart home ecosystem. Its user-friendly interface, enhanced connectivity options, and customization options further contribute to its appeal.

Future Scope

1. Further development of advanced AI algorithms to improve the robots' decision-making capabilities, enabling them to adapt to changing environments and optimize cleaning strategies.
2. Integration with IoT platforms to enable seamless the communication and coordination with other smart devices, facilitating synchronized cleaning operations and enhanced efficiency.
3. Research into more sensitive and precise dust detection sensors to improve the robots' ability to identify and target areas with high dust concentration accurately.

4. Exploration of self-recharging or wireless charging solutions to extend the operational time of the robots, allowing them to cover larger areas without interruption.
5. Development of eco-friendly materials and energy-efficient designs to minimize environmental impact and align with sustainability goals.
6. Exploration of applications beyond traditional indoor cleaning, such as outdoor environments or specialized settings like hospitals or industrial facilities.
7. Development of customization options to address specific cleaning requirements, such as handling different types of surfaces or targeting specific allergens or pollutants.
8. Integration of advanced analytic and predictive maintenance features into live monitoring capabilities, enabling proactive identification of cleaning needs and optimization of resource allocation.
9. Further enhancements to remote control features, allowing users to interact with and control the robots more intuitively and efficiently.
10. Continued exploration of ethical considerations, such as privacy concey

REFERENCE(S)

1. S. Pasika, S.T. Gandla, Smart water quality monitoring system with cost-effective using IoT, Heliyon 6 (7) (2020), doi:10.1016/j.heliyon.2020.e04096.
2. M. Mukta, S. Islam, S.D. Barman, A.W. Reza, M.S. Hossain Khan, "Iot based smart water quality monitoring system, in: Proceedings of the IEEE 4th International Conference on Computer and Communication Systems(ICCCS), 2019,pp.669–673, doi:10.1109/CCOMS.2019.8821742.
3. T.Sugapriyaa, S. Rakshaya, K. Ramyadevi, M. Ramya, P.G. Rashmi, Smart water quality monitoring system for real time applications, Int. J. Pure Appl. Math. 118 (2018) 1363–1369.
4. A. Jerom B., R. Manimegalai, R. Manimegalai, An IoT based smart water quality monitoring system using cloud, in: Proceedings of the International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), 2020,pp.1–7, oi:10.1109/ic- ETITE47903.2020.450. 2020doi.
5. S.Shridharan,"Water Quality Monitoring System Using Wireless Sensor Network," International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), pp. 399-402, 2014.