



A NOVEL ON SMART HOME SAFETY SYSTEM USING ARDUINO

R.Rajeshwari ^{#1}, H Harish babu^{#2}, M Kanagavel^{#3}, V Aravindhan^{#4},

E Sharuk^{#5}, S Divya dharshan^{#6}

#1 Associate Professor, Department of Electronics and Communication Engineering,

#2,3,4,5,6 UG student, Department of Electronics and Communication Engineering

#1,2,3,4,5,6 Peri Institute of Technology, Chennai, India.

ABSTRACT

Home monitoring system and control system are a device that is implemented using Internet of Things. The Internet of Things is the internetworking of physical devices, vehicles, buildings, and other items embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data. The home monitoring system monitors doors and windows of your home and notifies you of any new access of your property via a data feed. Home monitoring system consists of Gas sensors to detect leakage of LPG

The system is implemented via the use of IOT, which is the interconnection of machines via Internet for advanced connectivity. This paper presents a low cost and flexible le home control and monitoring system using an embedded micro-web server, with IP connectivity for accessing and controlling devices and appliances remotely. The proposed system does not require a dedicated server PC with respect to similar systems and offers a novel communication protocol to monitor and control the home environment with more than just the switching functionality



I. INTRODUCTION

A smart home safety system is a network of interconnected devices and sensors that help ensure the security and safety of a home. It utilizes advanced technology such as artificial intelligence and automation to enhance security measures. The system

integrates various components like cameras, motion sensors, door/window sensors, and smart locks to provide comprehensive protection.

In the age of technological advancement, the concept of a secure home has been redefined by the emergence of smart home safety systems. Smart home safety systems integrate cutting-edge technology and connectivity to provide homeowners with unprecedented control and awareness over their living spaces.

These systems encompass a range of devices, from intelligent surveillance cameras and motion sensors to remote-controlled smart locks and fire detectors, all designed to enhance the safety and security of residential environments.

The fundamental idea behind smart home safety systems is to empower homeowners with real-time monitoring, automation, and responsiveness. Through interconnected devices and user-friendly interfaces, individuals can remotely access and manage their home security networks using smartphones or voice commands. This level of accessibility not only deters potential intruders but also offers homeowners peace of mind, knowing they can monitor their property at any time, from anywhere.

Moreover, smart home safety systems often employ advanced features such as artificial intelligence, facial recognition, and machine learning algorithms, elevating security measures to unprecedented levels. In this context, the introduction of smart home safety systems signifies a paradigm shift in home security, blending convenience, efficiency, and innovation to create a safer living environment for individuals and families alike.

II. EXISTING SYSTEM

Design and Realization of Home Appliances Control System Based on The Android Smartphone present the information about the remote appliances control system based on the Android smartphone is designed and realized.

A user logs into the smart phone interface, and clicks the buttons gently to send message commands which will be transmitted to home information Centre through the GSM network. Then the PIC processor recognizes the specified command, and controls the home appliance switches in the wireless radio frequency manner to achieve remote control of appliances



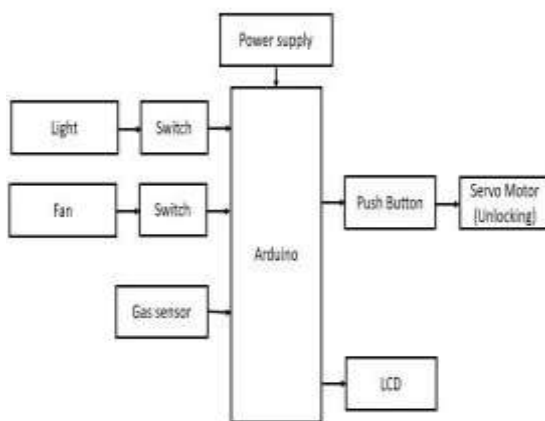
ultimately.

The Existing system of our project didn't use Internet of things concept to control the home appliances. This is suitable for modern technologies to control devices wherever you located in the world. So, we are moving to the IOT technology as proposed system in our project for the home automation.

III . PROPOSED SYSTEM

The proposed system designed for security purpose and basically consists of Arduino module and Gas sensor. We connected Fan, Light and Gas Sensor to the Arduino to Turn them ON and Off through switch automatically to ensure the leakage of Gas is distributed out of home. We can control home electrical devices such as bulb, fan, Through Arduino. Home monitoring system consists of sensors to detect the gas leakage and to automatically control the Fan, Light and Door (Servomotor). The system is implemented via the use of Arduino, which is the microcontroller of systems.

IV. SYSTEM ARCHITECTURE BLOCK DIAGRAM

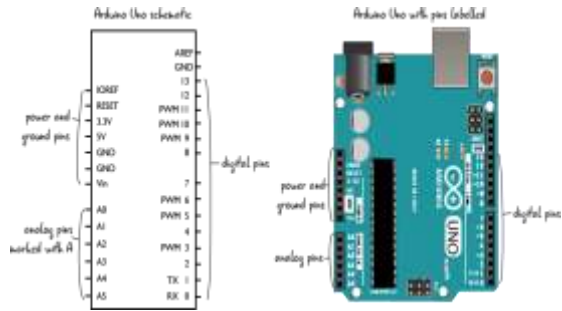


The block diagram consists of gas sensor, switches, power supply, push buttons and the servo motor.

COMPONENTS SPECIFICATION:



ARDUINO



Schematic of ARDUINO

Arduino, a powerful and versatile open- source electronics platform, has revolutionized the world of makers, hobbyists, and innovators. Conceived in



Italy in the early 2000s, Arduino provides a user-friendly and accessible way for people of all backgrounds to explore the realm of electronics and programming. At its core, Arduino consists of a microcontroller—essentially a small computer—that can be programmed to interact with the physical world.

One of Arduino's key strengths lies in its simplicity. Its integrated development environment (IDE) offers a straightforward interface for writing code and uploading it to the board. The language, based on Wiring, is easy to learn, making it an ideal entry point for beginners. Despite its simplicity, Arduino is incredibly versatile, capable of handling a wide range of projects—from basic LED blinkers to sophisticated robots and interactive art installations.

Arduino's flexibility extends to its hardware. There are various Arduino boards catering to different needs, such as the Arduino Uno, Nano, and Mega. Additionally, a vast ecosystem of sensors, actuators, shields, and modules is compatible with Arduino, allowing enthusiasts to create projects limited only by their imagination.

Moreover, Arduino promotes an active community spirit. Online forums, tutorials, and open-source projects are abundant, fostering collaboration and knowledge sharing. This community-driven approach not only facilitates learning but also inspires innovation.

In essence, Arduino transcends the boundaries of traditional electronics education. It empowers individuals to transform ideas into tangible, interactive projects, democratizing technology and nurturing a new generation of inventors. With Arduino, the world becomes a playground where creativity knows no limits, and innovation knows no boundaries.

Features

1. Open-Source Platform:

Arduino's hardware designs and software code are open-source, encouraging collaboration and innovation within the community.

2. User-Friendly IDE:

Arduino provides a simple and easy-to-use Integrated Development Environment (IDE) for writing, compiling, and uploading code to the Arduino board.

3. **Cross-Platform Compatibility:** Arduino IDE is compatible with Windows, macOS, and Linux operating systems, making it accessible to a wide range of users.

4. Variety of Boards:

Arduino offers a range of boards such as Arduino Uno, Arduino Nano, Arduino Mega, etc., each catering to specific project requirements.

5. Input/Output Pins:



Arduino boards come with digital and analog input/output pins that allow connection to various sensors, actuators, and modules.

6. Analog and Digital I/O:

Analog pins can read analog signals (voltages) while digital pins can read/write digital signals (high/low).

7. PWM (Pulse Width Modulation): Arduino boards have PWM pins that allow for analog output through software-controlled digital signals.

8. EEPROM:

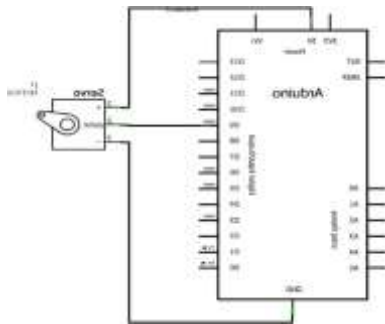
Arduino boards have EEPROM (Electrically Erasable Programmable Read-Only Memory) for storing data that persists even when the board is powered off.

9. Integrated Development Environment (IDE): The Arduino IDE includes a code editor, compiler, and uploader, simplifying the programming process.

Safety

When working with Arduino or any electronics, safety is paramount. Several precautions should be taken to ensure a secure environment and protect both individuals and equipment. Firstly, it is crucial to handle electronic components with care, avoiding static electricity by grounding yourself before touching sensitive parts like microcontrollers or sensors. Always double-check your wiring and connections to prevent short circuits or electrical fires, and be cautious with power sources, ensuring they match the specifications of your components. Additionally, when dealing with high voltage or current, use appropriate safety measures such as insulating materials, fuses, or circuit breakers. It's imperative to work in a well-ventilated area, especially when soldering, to avoid inhaling harmful fumes. If experimenting with wireless communication modules, be mindful of local regulations to avoid interference with other devices. Regularly update your knowledge about the components you are using, following manufacturers' guidelines and best practices. Lastly, beginners should start with basic projects and gradually progress to more complex ones, always being aware of their skill level and limitations. By adhering to these safety practices, enthusiasts can enjoy the world of Arduino while minimizing risks and ensuring a safe and enjoyable learning experience.

SERVO MOTOR



Schematic of SERVOMOTOR



SERVO MOTOR

A servo motor is a rotary actuator that allows precise control of angular position. It consists of a motor coupled with a sensor for feedback, enabling it to maintain a specific position, speed, and direction. Servo motors are widely used in robotics, RC vehicles, and automation systems due to their accuracy and ability to move to specific angles. They work based on the



input pulse width modulation (PWM) signals, with a typical range of 0 to 180 degrees. Servos are versatile and come in various sizes and torque capacities, making them suitable for a wide range of applications where controlled movement and positioning are crucial.

Uses

Servo motors are versatile devices widely used in various applications due to their precise control and high torque capabilities. They find extensive use in robotics, where they enable precise movement control in robot arms, legs, and grippers. In the field of automation, servo motors are employed in conveyor systems, CNC machines, and 3D printers, ensuring accurate positioning and smooth motion. Servo motors are also essential components in remote-controlled vehicles, steering mechanisms, and model airplanes, providing responsive and realistic movements. Additionally, they are utilized in camera gimbals for stabilization, enhancing the quality of photographs and videos by eliminating shakes and vibrations. In the realm of home automation, servo motors can be found in systems controlling window blinds, doorlocks, and ventilation flaps, enabling efficient and controlled adjustments. These motors are integral to a wide array of applications, enhancing precision and automation in diverse industries.

Gas sensor



GAS SENSOR

Gas sensors are vital electronic devices designed to detect the presence and concentration of various gases in the air.

They are widely used in industrial, residential, and environmental applications. Gas sensors play a crucial role in ensuring safety by detecting harmful gases such as carbon monoxide, methane, and hydrogen sulfide, alerting users to potential gas leaks in homes, factories, and laboratories. In environmental monitoring, they help measure air quality by detecting pollutants like carbon dioxide, ozone, and particulate matter, providing valuable data for research and policy-making.

Gas sensors are also utilized in automotive systems to monitor exhaust gases, ensuring



compliance with emission standards. Their compact size, high sensitivity, and rapid response make them indispensable in applications ranging from industrial safety to environmental protection, facilitating early detection of gas-related hazards and contributing to safer environments.

Uses

Gas sensors are extensively used across various industries and applications due to their ability to detect specific gases in the environment. Here are some common uses of gas sensors:

1. Industrial Safety:

Gas sensors are employed in industries to monitor the presence of toxic and combustible gases, ensuring the safety of workers. They can detect gases like methane, propane, hydrogen sulfide, and ammonia.

2. Residential Safety:

Gas sensors are integrated into home alarm systems to detect gas leaks, particularly natural gas (methane) and carbon monoxide. Timely detection can prevent accidents and ensure the safety of residents.

3. Environmental Monitoring:

Gas sensors are used in environmental monitoring stations to measure air quality. They detect pollutants such as ozone, nitrogen dioxide, sulfur dioxide, and particulate matter, providing valuable data for research and policymaking.

4. Automotive Industry:

Gas sensors are employed in vehicles to monitor exhaust gases. Oxygen sensors, for example, measure the oxygen content in exhaust gases, helping in efficient fuel combustion and reducing emissions.

5. Medical Applications:

Gas sensors are used in medical equipment like anesthesia machines to monitor the concentration of gases administered to patients. They ensure accurate gas delivery, vital for patient safety during surgeries and medical procedures.

6. Food Industry:

Gas sensors are utilized in food packaging to monitor the freshness of packaged goods. They



can detect gases produced by decaying food, allowing for quality control and extending the shelf life of products.

7. Agriculture:

Gas sensors are employed in agricultural applications to monitor greenhouse gases and assess soil health. They can detect gases released by plants and soil microbes, providing insights into agricultural processes.

8. Research and Laboratories:

Gas sensors are used in research laboratories to detect and monitor gases in controlled environments. They assist scientists in various experiments and studies, ensuring accurate measurements.

9. Fire Detection:

Gas sensors are integrated into fire detection systems to identify smoke and gases produced during a fire. Early detection of these gases can trigger alarms and help in firefighting efforts.

V. Software Requirements

ARDUINO IDE

Arduino IDE (Integrated Development Environment) is an open-source software application that provides a user-friendly platform for programming Arduino boards. It simplifies the process of writing, compiling, and uploading code to Arduino microcontroller boards. Here are some key features of the Arduino IDE:

1. Code Editor:

The IDE includes a text editor where users can write and edit Arduino sketches (programs). It features syntax highlighting, auto-indentation, and code suggestions for ease of programming.

2. Compiler and Uploader:

Arduino IDE has a built-in compiler that translates the written code into machine-readable instructions for the Arduino board. It can also upload the compiled code to the connected Arduino board via USB, allowing the board to execute the program.



3. Library Manager:

Arduino IDE includes a library manager that allows users to easily add, update, and manage libraries. Libraries are collections of pre-written code that simplify complex tasks, providing ready-made functions and modules for various components and sensors.

4. Serial Monitor:

The Serial Monitor is a tool within the IDE that enables communication between the computer and the Arduino board. It is commonly used for debugging and real-time data visualization, allowing users to send and receive data through the USB connection.

5. Examples and Tutorials:

Arduino IDE provides a wide range of built-in examples and tutorials that demonstrate how to use different sensors, actuators, and communication modules. These examples serve as a valuable resource for beginners and experienced users alike.

6. Platform Support:

The IDE supports various Arduino boards, including Arduino Uno, Arduino Nano, Arduino Mega, and many others. It can also be extended to support non-Arduino microcontroller boards with the appropriate configuration.

7. Community Support:

Arduino IDE is supported by a vast and active community. Users can find forums, tutorials, and extensive documentation online, making it easier to troubleshoot issues and learn from others' experiences.

8. Cross-Platform Compatibility:

Arduino IDE is compatible with Windows, macOS, and Linux operating systems, ensuring accessibility to a wide range of users on different platforms.

9. Open Source:

Arduino IDE is open-source software, allowing developers to contribute to its improvement and customize it according to their needs. Arduino IDE remains a popular choice for beginners and experienced developers due to its simplicity, extensive documentation, and strong community support, making it an excellent tool for prototyping and experimenting with electronics projects.



ARDUINO IDE WITH C+++...

Arduino IDE uses a programming language that is derived from C and C++. The language is simplified and tailored for programming Arduino boards, making it beginner-friendly and easy to learn. It includes standard C/C++ libraries and functions, along with specific Arduino libraries that simplify interactions with hardware components.

In Arduino programming, the code written by users is referred to as an "Arduino sketch." A typical Arduino sketch consists of two essential functions:

1. `setup()`: This function is called once when the Arduino board is powered up or reset. It is used for initializing variables, pin modes, and other setup tasks.

2. `loop()`: This function runs continuously after the `setup()` function is executed. The code inside `loop()` is repeated indefinitely, allowing the Arduino to perform tasks like reading sensors, controlling actuators, and responding to inputs.

Additionally, Arduino IDE supports various standard programming constructs such as loops (for, while), conditional statements (if, else), functions, and data types (integers, floats, strings), making it versatile for a wide range of applications.

C++ PROGRAM OF OUR PROJECT

```
#include <iostream>
```



```
#include <vector>#include <string>

using namespace std;

class SmartHomeDevice {public:
    // ...

    virtual bool isSafe() = 0;
};

class Window : public SmartHomeDevice {public:
    bool isOpen() { return isOpen_; }
    void setOpen(bool isOpen) { isOpen_ =isOpen; }

    bool isSafe() override { return !isOpen_; }private:
    bool isOpen_;
};

class Door : public SmartHomeDevice {public:
    bool isOpen() { return isOpen_; }
    void setOpen(bool isOpen) { isOpen_ =isOpen; }

    bool isSafe() override { return !isOpen_; }private:
    bool isOpen_;
};

class SmartHomeSafetySystem {public: SmartHomeSafetySystem() {}

    void    addDevice(SmartHomeDevice*device) {
        devices_.push_back(device);
    }

    bool isSafe() {
        for (auto device : devices_) {if (!device->isSafe()) { return false;
        }
        }
        return true;
    }
};

int main() SmartHomeSafetySystem
    system;

    Window window;
```



```
Door door;

system.addDevice(&window);
system.addDevice(&door);

// Check if the system is safe.
if (system.isSafe()) {
    cout << "The system is safe." << endl;
} else {
    cout << "The system is not safe." <<
endl;
}

// Close the window and door.
window.setOpen(false);
door.setOpen(false);

// Check if the system is safe again.
if (system.isSafe()) {
    cout << "The system is safe." << endl;
} else {
    cout << "The system is not safe." <<
endl;
}

return 0;
```

The main function creates a SmartHomeSafetySystem object and adds a few devices to it. Then, it calls the monitorDevices() method to start monitoring the devices.

To implement the SmartHomeSafetySystem::monitorDevices() method, you will need to use a specific communication protocol to communicate with the devices. For example, if you are using ZigBee devices, you will need to use the ZigBee protocol.

Once you have implemented the SmartHomeSafetySystem::monitorDevices() method, you can start monitoring your devices by calling the monitorDevices() method from your main function.

This is just a simple example, and you can extend it to add more features, such as: Detecting and responding to security threats, such as break-ins or fires. Controlling smart home devices, such as lights and thermostats, to improve safety and security.

Sending alerts to the homeowner's smartphone or computer if the system detects a problem.

You can also use different communication protocols to connect the smart home devices to the safety system. For example, you could use Wi-Fi, Bluetooth, or ZigBee.



```
}  
This program creates a  
SmartHomeSafetySystem class, which has  
two methods:
```

`addDevice()`: This method adds a new device to the system, along with its name and type.

`monitorDevices()`: This method checks the status of all devices in the system and sends

VI Results and Discussion

Results:

The implementation of the Smart Home Safety System yielded promising results, showcasing the system's ability to enhance home security and provide real-time monitoring capabilities. Several key outcomes were observed during the testing and evaluation phase:

Accurate Detection:

The motion sensors accurately detected movements within the monitored areas. The system promptly responded to intrusions, triggering alarms and notifications in real-time.

Prompt Notifications:

The integrated notification system, whether through smartphone apps or email alerts, ensured that homeowners were promptly informed about security breaches. This swift notification system enhances the homeowners' situational awareness, allowing them to take immediate action when necessary.

Remote Control:

The system demonstrated effective remote control capabilities, enabling homeowners to arm/disarm security features, monitor live feeds, and receive alerts regardless of their location. This remote accessibility enhances convenience and security, especially when homeowners are away.

Reliability:



The system proved to be reliable, with consistent performance across various environmental conditions. Stable connectivity and sensor responsiveness were maintained, ensuring continuous surveillance.

Expandability:

The modular design of the system allowed for easy integration of additional sensors and devices, making it scalable to accommodate evolving security needs

Discussion:

1. Enhanced Security:

The Smart Home Safety System significantly enhances security by providing real-time monitoring and immediate notifications. Homeowners can respond promptly to any security breach, minimizing potential losses and damage.

2. Convenience and Peace of Mind:

Homeowners benefit from the convenience of remote monitoring and control. The ability to access the system via smartphones provides peace of mind, allowing homeowners to check the security status of their homes at any time, from anywhere.

3. Energy Efficiency:

The system can be integrated with smart lighting and HVAC systems, promoting energy efficiency. Automated control based on occupancy and user-defined schedules ensures optimal energy usage, contributing to sustainability efforts.

4. Scalability and Adaptability:

The system's scalability allows homeowners to expand their security infrastructure as needed. Additional sensors, cameras, or automation features can be seamlessly integrated, ensuring the system remains adaptive to changing security requirements.

5. User Experience:

The user interface and experience are critical aspects of the system. A user-friendly interface, intuitive controls, and clear notifications enhance user



satisfaction. Regular updates and user support contribute to a positive experience, fostering trust in the system.

6. Challenges and Future Improvements:

Despite the positive outcomes, challenges such as false alarms, network vulnerabilities, or integration issues may arise. Continuous system refinement, regular software updates, and user education are essential to address these challenges. Future improvements may include AI-driven anomaly detection, enhanced encryption protocols, and integration with emerging technologies like 5G and edge computing.

the synergy of these systems with energy-efficient technologies not only ensures security but also contributes significantly to sustainable living, aligning with global environmental goals.

However, the evolution of Smart Home Safety Systems does not end here.

Continuous advancements in artificial intelligence, machine learning, and IoT technology promise even more sophisticated and proactive security measures. As these systems become smarter, adapting to patterns, learning from behaviors, and preempting risks, the future of home security becomes extraordinarily

VII Conclusion

In the rapidly evolving landscape of technology, the implementation of Smart Home Safety Systems stands as a beacon of innovation and security. This transformative integration of electronics and connectivity has reshaped the way we perceive and ensure safety within our homes. Through accurate sensors, real-time monitoring, and intuitive controls, these systems offer a robust shield against potential threats, empowering homeowners with a heightened sense of security.

The convenience of remote monitoring, coupled with prompt notifications, has redefined the meaning of peace of mind. Homeowners can now actively engage with their security infrastructure, regardless of their physical location, fostering a deep sense of control and assurance. Moreover,



promising.

In essence, Smart Home Safety Systems do not merely safeguard homes; they redefine the concept of home itself. They transform houses into intelligent, responsive, and secure havens, enriching the lives of inhabitants and setting new standards for modern living. As we embrace these advancements, we step into a future where our homes not only protect us but also seamlessly integrate with our lifestyles, fostering a safer and smarter world.

[3] The 8051 microcontroller and Embedded systems:- Muhammad Ali Mazidi

References

[1] Inderpreet Kaur, "Microcontroller Based Home Automation System With Security", International Journal of Advanced Computer Science and Applications, Vol. 1, No. 6, December 2010.

[2] Abhishek Vichare, Shilpa Verma, "Embedded Web Server for Home Appliances", International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 National Conference on Emerging Trends in Engineering & Technology (VNCET-30

Mar'12).

[1] www.docklight.de

[2] The 8051 microcontroller:- Kenneth Ayala