ABSTRACT

Smart home automation system is a web based application that allows user to monitor home appliances using mobile devices. This system established for the entire home user after gaining access from administrator. This system includes remote control and monitoring domestic appliances, security and energy management. Once all the appliances in home are automated and connected it important to consider issue of security authentication and access control. The goal of this project is to create an IoT Smart House that can monitor specific criteria, as well as control specific devices. Data to be monitored are: temperature, humidity, movement, water, and power.

MID TERM REPORT ON Smart Home

Submitted by:

Aman Singh(171500036)

Akash Chaudhary(171500026)

Course: B.tech

Branch: CSE

Section : C

Submitted To:

Mr. Amir Khan

Technical Trainer

INTRODUCTION

While the cost of living is going up, there is a growing focus to involve technology to lower those prices. With this in mind the Smart Home project allows the user to build and maintain a house that is smart enough to keep energy levels down while providing more automated applications. A smart home will take advantage of its environment and allow seamless control whether the user is present or away. With a home that has this advantage, you can know that your home is performing at its best in energy performance. A low cost and efficient smart home system is presented here. This system has two main modules: the hardware interface module and the software communication module. At the heart of this system is the Arduino UNO microcontroller which is also capable of functioning as a micro web server and the interface for all the hardware modules. All communication and controls in this system pass through the microcontroller. The smart home system offers feature such as environmental monitoring using the temperature, humidity, gas and smoke sensors. It also offers switching functionalities to control lighting ,fans/air conditioners, and other home appliances connected to the relay system. Another feature of this system is the intrusion detection which it offers using the motion sensor and all these can be controlled from the Android Smartphone app.

EXISTING SYSTEM

The Existing system based on with the GSM Module & Bluetooth Module only. The recent developments in technology which permit the Use of Bluetooth and Wi-Fi have enabled different devices to have capabilities of connecting with each other. In this system we can control our all electronic equipments through our cell phone but this system does not has the smart sence. We can produce the smart sence by using various sensors.

PROPOSED SYSTEM

Our proposed system is a NODEMCU based home automation done with NODEMCU controlled via android app. This system deals with the safety in home and smart home technologies which will a be cost efficient. Arduino can sense the surroundings by receiving input signal from a variety of sensors and can affect its environment via actuators. The Passive Infra-Red (PIR) sensors allow one to sense motion, almost always and are used to detect whether a human has moved in or out of the sensors range. The PIR sensor is a pyroelectric device that detects motion by measuring changes in the infrared level emitted by surrounding objects.

USES OF THE PROJECT

1. Heating, ventilation and air conditioning: it is possible to have remote control of all home energy monitors over the internet incorporating a simple and friendly user interface.
2. Lighting control system: we can control our lights by using our cell phone.
3. Automatic warning system when anyone tries to enter in the home without admin's permission.
4. Counts no. of persons entered in the house.

FUNCTIONAL SPECIFICATION

1. We can control our all lights through using our android cell phone.
2. It will give you the count of persons entered in the room
3. Rain detection system
4. Smart security system

SOFTWARE SPECIFICATION

1. Technology Implemented: IoT
2. Language Used: Embedded C
3. User Interface: Android Application

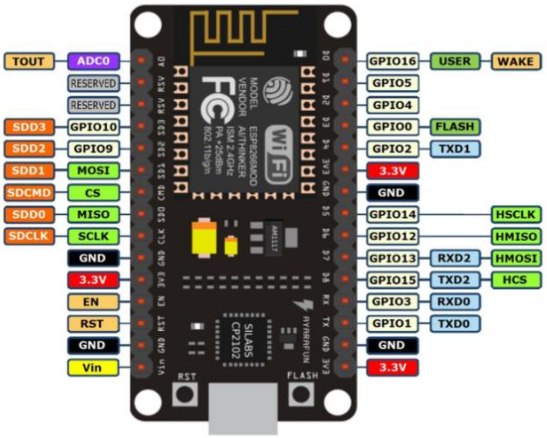
HARDWARE REQUIREMENTS

1. NODEMCU
2. Connecting wires
3. Power source
4. Resisters
5. Bread Board
6. PIR sensor
7. Rain sensor
8. RGB lights
9. Servo Moters
10. Temperature Sensor

HARDWARE DESCRIPTION

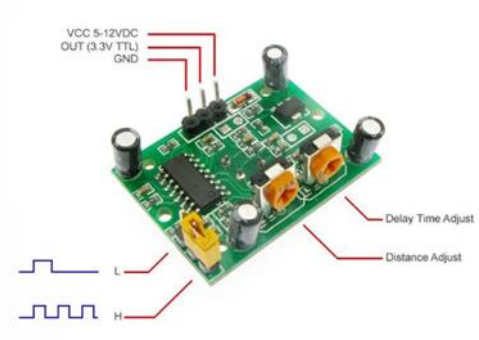
1. NODEMCU:

NodeMCU is an open-source firmware and development kit that helps you to prototype or build IoT product. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. The firmware uses the Lua scripting language. It is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. MCU stands for MicroController Unit - which really means it is a computer on a single chip. A microcontroller contains one or more CPUs (processor cores) along with memory and programmable input/output peripherals. They are used to automate automobile engine control, implantable medical devices, remote controls, office machines, appliances, power tools, toys etc



2. PIR SENSOR:

A passive infrared sensor (PIR sensor) is an electronic sensor that measures infrared (IR) light radiating from objects in its field of view. They are most often used in PIR-based motion detectors. PIR sensors are commonly used in security alarms and automatic lighting applications. PIR sensors detect general movement, but do not give information on who or what moved.

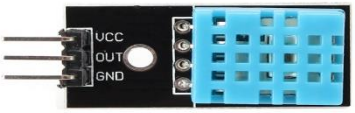


3. RAIN SENSOR:

A rain sensor or rain switch is a switching device activated by rainfall. There are two main applications for rain sensors. The first is a water conservation device connected to an automatic irrigation system that causes the system to shut down in the event of rainfall. The second is a device used to protect the interior of an automobile from rain and to support the automatic mode of windscreen wipers.

4. TEMPERATURE SENSOR:

The DHT11 is a basic, low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). It’s fairly simple to use, but requires careful timing to grab data. The only real downside of this sensor is you can only get new data from it once every 2 seconds, so when using our library, sensor readings can be up to 2 seconds old.

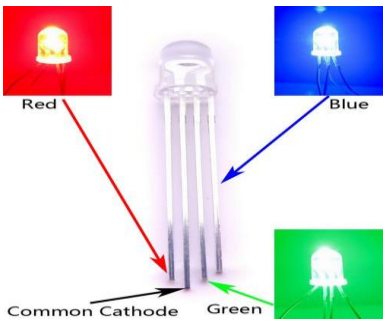


5. SERVO MOTER:

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback

6. RGB LIGHTS:

In this red, green and blue light are added together in various ways to reproduce a broad array of colors. This name comes from the initials of the three additive primary colors, red, green, and blue.



7. BREAD BOARD:

A thin plastic board used to hold electronic components (transistors, resistors, chips, etc.) that are wired together. Used to develop prototypes of electronic circuits.

PROJECT DESCRIPTION

PROBLEM DEFINITION:

The project consists of developing and implementing a smart system that has the smart sense. The developed system will reduce the human work as well as save human time. In present time security is main concern to all. The developed system will have an alert system to prevent theft. Developed system can work continuously without any fail.

OVERVIEW OF THE PROJECT:

The project is divided in several modules. All modules have the different functionality. All the modules are interconnected to each other and the whole system will be controlled via a smart phone.

MODULES DESCRIPTION:

● SECURITY MODULE:

In this module we have used passive infrared sensor for motion detection. We have implanted one PIR sensor on every wall just beside the windows. When any person tries to enter in the home through the windows, PIR sensor will send an alert notification to your phone.

● RAIN DETECTION MODULE:

In this module we have used a rain detection sensor to detect the rainfall. It is place on the rooftop. When it detects the rain, it automatically closes all the doors and windows and also send a notification to your smart phone.

● LIGHTNING CONTROL MODULE:

In this module RGB lights are used I order to provide lightning in the home. RGB lights have the several colors and through our phone we can change color of lights.

● MANUALLY DRIVEN MODULE:

To this module all the modules are connected. So that we can also control all our devices manually too. Functionality of this module is that we can manually open and the close the windows and doors.

TEMPERATURE MONITORING MODULE:

In this module we have used a sensor named DHT-11. Through which we monitor the temperature of the home. It continuously send the data to our smart phone by using blynk cloud.

IMPLEMENTATION DETAILS

● Till present date half of the system implementation has been done. Integration of sensors with controller has been done. Functioning of all devices has been checked with sample code.

● Code is not completely written because blynk functions are not implemented in the code. Only in security module blynk has been implemented.

● PROGRAM LISTING

```
#define BLYNK_PRINT Serial


#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "me-s-auQ9rjPi-KsSlaQmpWZVih6FJ-J";
int sensor=D6;
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "adult_dude";
char pass[] = "ldk123456";

BlynkTimer timer;

// This function sends Arduino's up time every second to Virtual Pin (5).
// In the app, Widget's reading frequency should be set to PUSH. This means
// that you define how often to send data to Blynk App.
void myTimerEvent()
{
  bool state=digitalRead(sensor);
  if(state==HIGH)
  {
    Serial.println("MESSAGE");
    Blynk.notify("ALERT MESSAGE !!!!!!!");
    }
    else
    {
      Serial.println("MESSAGE");
      }
  // You can send any value at any time.
  // Please don't send more that 10 values per second.
  Blynk.virtualWrite(V5, millis() / 1000);
}
```

```
void setup()
{
  // Debug console
  Serial.begin(9600);

  Blynk.begin(auth, ssid, pass);
  // You can also specify server:
  //Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
  //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);

  // Setup a function to be called every second
  timer.setInterval(1000L, myTimerEvent);
}

void loop()
{
  Blynk.run();
  timer.run(); // Initiates BlynkTimer
}
```

BIBLIOGRAPHY

● BOOK REFERENCES:

1. Programming Arduino Getting Started with Sketches **SIMON MONK**

● WEBSITES:

1. https://blynk.cc
2. https://www.nodemcu.com/index_en.html
3. https://en.wikipedia.org/wiki/Passive_infrared_sensor
4. https://en.wikipedia.org/wiki/Rain_sensor
5. https://circuitdigest.com/article/servo-motor-basics
6. https://www.waveshare.com/wiki/DHT11_Temperature-Humidity_Sensor