# Smart Air Conditioning Control for Thermal Comfort

First A. Akhilesh (20BCE2306), Second B. Venkata Laxman Kumar (20BCE0829), Third C. Kanala Harika (20BKT0101), and Fourth D. SujayKumar Reddy (20BDS0294).

*Abstract*—**Thermal comfort brings a great influence that may affect the satisfaction of an occupant about the surrounding indoor environment, as this could led to the level of productivity and social interactions. One of the most suitable tools that have been utilized in tropical regions to create a thermally comfortable indoor environment is the air conditioning (AC) system. However, the conventional method of controlling the AC may not guarantee thermally comfortable indoor ambience, furthermore excessive cooling chosen by users may also contribute to negative side effects toward the occupancy's health. To overcome this problem, an IoT-based smart system that can control the AC to provide a suitable thermally comfortable environment is proposed in this project.**

***Index Terms*—Thermal Comfort, Air Conditioning, IoT, Health.**

## I. Introduction

The widespread use of the internet has made it an integral part of modern human life, facilitating numerous aspects such as social media, banking transactions, news, shopping, and e-government. This has led to a notable increase in the standard of living. The rapid development of technology has also brought convenience to various aspects of daily life through the implementation of IoT and M2M technologies, allowing devices such as TVs, refrigerators, and washing machines to communicate with each other. The smart city concept has become increasingly prevalent in recent years, with emphasis on smart applications such as transportation, health, education, energy, and environment. Among these, smart air conditioning is a highly sought-after individual application. This study proposes an implementation of smart air conditioning using the NodeMCU embedded system microcontroller and Blynk iOS/Android interface developer. Smart air conditioners can be controlled via a smartphone application, as well as being integrated with smart home systems or voice assistants. Their internet connectivity allows them to interact with other devices, which is what distinguishes them from traditional air conditioning systems. One major advantage of smart air conditioners is their energy-saving potential, with the United States being a significant consumer of energy for air conditioning. The adoption of smart air conditioners can therefore contribute to a significant reduction in energy consumption.

## II. Literature survey

### A. IoT based Smart-Air Conditioning System using Machine Learning

The paper [1] proposes an IoT-based smart air conditioning system that utilizes machine learning algorithms to predict thermal comfort levels based on indoor and outdoor environmental data. The authors also propose a machine learning model using artificial neural networks (ANN) to predict the thermal comfort level based on indoor and outdoor environmental data. The proposed system was evaluated using data collected from a real-world testbed, and the results show that the system is capable of accurately predicting the thermal comfort level. The authors also compared their system with a conventional thermostat-based air conditioning system and found that their system was more energy-efficient while maintaining the same level of thermal comfort. The proposed system also provides a user-friendly mobile application for users to monitor and control the air conditioning system remotely.

### B. IoT based Intelligent Control for Thermal Comfort in Buildings.

*The paper [2] proposes an IoT-based intelligent control system for improving thermal comfort in buildings. The proposed system uses a combination of wireless sensor networks (WSNs) and machine learning algorithms to optimize heating, ventilation, and air conditioning (HVAC) systems in buildings. The system is designed to monitor and analyze environmental data such as temperature, humidity, CO2 levels, and occupancy patterns to determine the optimal settings for HVAC systems. They conducted a case study to evaluate the performance of the proposed system in a real-world building. The results show that the system is capable of improving thermal comfort while reducing energy consumption by up to 25%. The authors also compared their system with a traditional HVAC system and found that their system achieved better thermal comfort levels with lower energy consumption.*

### *C. IoT-based thermal comfort monitoring and control in smart homes*

The paper [3] proposes an IoT-based system for monitoring and controlling thermal comfort in smart homes. The system uses a wireless sensor network (WSN) to collect environmental data such as temperature and humidity, and then applies machine learning algorithms to optimize the indoor temperature and humidity settings for thermal comfort. They developed a prototype of the proposed system and conducted experiments in a smart home environment. The results show that the proposed system is capable of maintaining a comfortable indoor environment while reducing energy consumption by up to 20%. The system is also designed to provide real-time feedback to occupants regarding their thermal comfort levels, which may help to promote more sustainable behaviors.

### *D. IoT-based thermal comfort monitoring and control in smart homes*

The paper [4] proposes an IoT-based system that monitors and controls the indoor thermal comfort of office buildings. The system aims to provide a comfortable environment for occupants while reducing energy consumption. This system uses a wireless sensor network (WSN) to monitor various environmental parameters such as temperature, humidity, CO2 levels, and occupancy. The data collected from the WSN is then sent to a central server for analysis and decision-making. Based on the data analysis, the system can adjust the HVAC system's settings to maintain a comfortable indoor temperature and humidity level. They have conducted experiments to validate the system's performance, and the results showed that the system was effective in maintaining thermal comfort while reducing energy consumption. The proposed system has the potential to be implemented in various office buildings and can contribute to the sustainable development of smart cities.

### *E. An IoT-based thermal comfort control system for energy-efficient buildings*

The paper [5] presents a system that utilizes the Internet of Things (IoT) technology to provide thermal comfort control in buildings, with the aim of reducing energy consumption. The proposed system consists of sensors that monitor the indoor temperature and humidity, an actuator that controls the HVAC system, and a central server that processes the data and provides control signals to the actuator. Machine learning algorithms are also utilized to optimize the thermal comfort control system based on occupancy patterns and weather conditions. The proposed system was tested in a real building, and the results showed that it could effectively provide thermal comfort control while reducing energy consumption. The study concludes that IoT-based thermal comfort control systems have great potential for reducing energy consumption in buildings while maintaining occupant comfort.

## III. Findings/Inferences From Literature

Today, the internet has become an indispensable part of human life. With our social media, banking transactions, news, shopping, e-government and many more applications, we have reached a higher level in our standard of living. This rapid development of technology provides convenience in different areas of our lives every day. Many devices such as TVs, refrigerators, washing machines we use in daily life are able to communicate with each other by becoming smart thanks to the Internet of Things (IoT) and Machine to Machine (M2M) technologies. In recent days we have become more and more aware of the concepts of smart city such as smart home, transportation, health, education, energy and smart environment. One of these concepts, "smart air conditioning" applications undoubtedly attracts the most demanded applications individually. In this study, a smart conditioning application is implemented using NodeMCU embedded system microcontroller, Blynk iOS/Android interface developer. Smart ACs allow you to maintain your home temperature using a smartphone.

Their functionality can be controlled via an app that you can download on your phone or tablet. Moreover, they can also be connected to smart home systems or voice assistants. The ability of a smart air conditioning system to connect to the internet and consequently other devices is what makes them smart. While a smart AC offers numerous benefits thanks to its ever-growing feature set, one amazing advantage is that users are able to save energy. With the United States using more energy for air conditioning than the rest of the world, this is an extremely important consideration. If conventional air conditioner users convert to a smart air conditioner, this can significantly lower energy consumption.

## IV. Proposed System

The proposed system for a smart air conditioning system using NodeMCU typically involves the use of sensors to monitor the indoor environment, such as temperature, humidity, and air quality. The NodeMCU is then used as a microcontroller to collect the sensor data and communicate it to a central server using Blynk Dashboard. The system also includes an application or interface for users to interact with the air conditioning system, such as adjusting the temperature, turning the system on or off, or setting schedules for when the system should operate. Overall, the NodeMCU-based smart air conditioning system offers greater control, convenience, and energy efficiency compared to traditional air conditioning systems.

## V. Methodology/Procedure

We can control the system via internet using the NodeMCU v1.0 board which also acts as a microcontroller board which helps us in interacting with the DHT11 and PIR sensor which helps us in identifying if a person is available in the room or not and the temperature is also sensed which helps us in deciding whether AC is needed to be ON or the FAN should be on, which in turn helps in energy savings indirectly.
Basic idea is that when a human is detected, the Fan or AC should turn ON based on the temperature in the room and the AC will be ON if the temperature is higher than 22 degree celsius and FAN will turn ON when the temperature gets below 22 degree celsius. We are using two 5V relays to

control the FAN and AC with the help of controlling pins (IN1 and IN2 respectively).

## VI. Experimental setup

### A. PIR Motion Sensor

PIR sensors allow you to sense motion, almost always used to detect whether a
human has moved in or out of the sensor's range. They are small, inexpensive, low-power, easy to use and don't wear out. For that reason, they are commonly found in appliances and gadgets used in homes or businesses. They are often referred to as PIR, "Passive Infrared", "Pyroelectric", or "IR motion" sensors. PIRs are basically made of a pyroelectric sensor (which you can see below as the round metal can with a rectangular crystal in the center), which can detect levels of infrared radiation. Everything emits some low-level radiation, and the hotter something is, the more radiation is emitted. The sensor in a motion detector is actually split in two halves. The reason for that is that we are looking to detect motion (change) not average IR levels. The two halves are wired up so that they cancel each other out. If one half sees more or less IR radiation than the other, the output will swing high or low.

### B. PIR Motion Sensor

PIR sensors allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensor's range. They are small, inexpensive, low-power, easy to use and don't wear out. For that reason they are commonly found in appliances and gadgets used in homes or businesses. They are often referred to as PIR, "Passive Infrared", "Pyroelectric", or "IR motion" sensors. PIRs are basically made of a pyroelectric sensor (which you can see below as the round metal can with a rectangular crystal in the center), which can detect levels of infrared radiation. Everything emits some low level radiation, and the hotter something is, the more radiation is emitted. The sensor in a motion detector is actually split in two halves. The reason for that is that we are looking to detect motion (change) not average IR levels. The two halves are wired up so that they cancel each other out. If one half sees more or less IR radiation than the other, the output will swing high or low.

### C. DHT11 Sensor

DHT11 is a Humidity and Temperature Sensor, which generates calibrated digital output. DHT11 can be interfaced with any microcontroller like Arduino, Raspberry Pi, etc. and get instantaneous results. DHT11 is a low cost humidity and temperature sensor which provides high reliability and long term stability. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and outputs a digital signal on the data pin (no analog input pins needed). Its very simple to use, and libraries and sample codes are available for Arduino and Raspberry Pi. This module makes it easy to connect the DHT11 sensor to an Arduino or microcontroller as it includes the pull up resistor required to use the sensor. Only three connections are required to be made to use the sensor - Vcc, Gnd and Output. It has high reliability and excellent long-term stability, thanks to the exclusive digital signal acquisition technique and temperature & humidity sensing technology.

Specifications: Power Supply: 3.3~5.5V DC, Output: 4 pin single row, Measurement Range: Humidity 20-90%RH, Temperature 0~50°C, Accuracy: Humidity +-5%RH, Temperature +-2°C, Resolution: Humidity 1%RH, Temperature 1°C, Interchangeability: Fully Interchangeable, Long-Term Stability: < 1%RH/year

Pin Description: Power +Ve (3.3VDC to 5.5VDC Max wrt. GND), Serial Data Output and Power Ground or Power –Ve.

### D. Two Channel Relay

The 2 Channels Relay Module is a convenient board which can be used to control high voltage, high current load such as motor, solenoid valves, lamps and AC load. It is designed to interface with microcontrollers such as Arduino, PIC, etc. The relay terminal (COM, NO and NC) is being brought out with a screw terminal. It also comes with a LED to indicate the status of the relay.

Specification: Digital output controllable, Compatible with any 5V microcontroller such as Arduino, Rated through-current: 10A (NO) 5A (NC), Control signal: TTL level, Max. switching voltage 250VAC/30VDC, Max switching current 10A, Size: 50mm x 38mm x 17mm

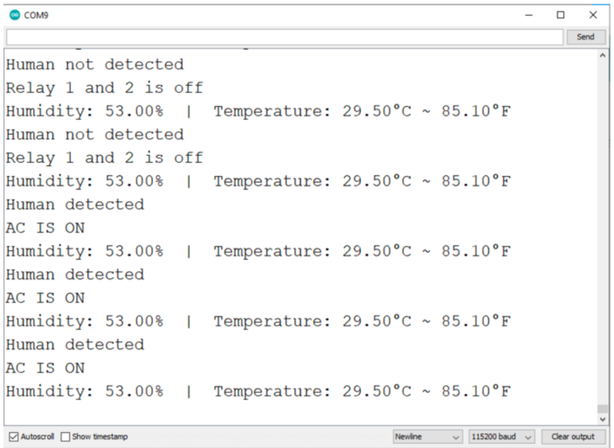
### E. LED

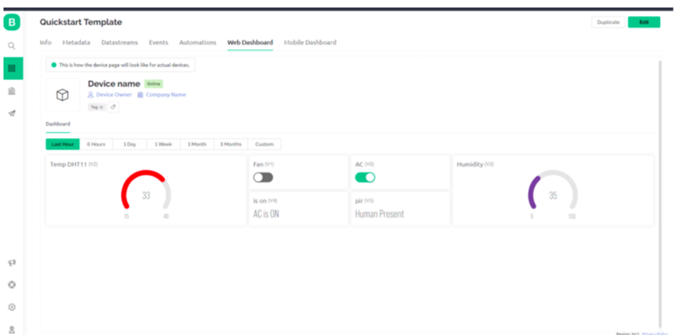
A light-emitting diode (LED) is a semiconductor device that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. The color of the light (corresponding to the energy of the photons) is determined by the energy required for electrons to cross the band gap of the semiconductor. White light is obtained by using multiple semiconductors or a layer of light-emitting phosphor on the semiconductor device.

### F. NodeMCU

Ai Thinker NodeMCU-ESP8266 is an open-source firmware and development kit that helps you to prototype or build IoT products. It includes firmware that 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. WIFI module: ESP-12E, Processor: ESP8266, CP2102 Chip, Built-in Flash: 32Mbit, Antenna: Onboard PCB antenna, Peripheral interface: UART/SPI/I2C/SDIO/GPIO/ADC/PWM, WiFi protocol: IEEE 802.11 b/g/n, Frequency range: 2.4G ~ 2.5G (2400M ~ 2483.5M), WIFI mode: Station / SoftAP / SoftAP+Station, Power supply: 5V, Logic level: 3.3V

## VII. Results and Discussion





## VIII. Conclusions and Futurework

We have designed and tested the prototype that would switch on the fan connected using a relay when the room has achieved the desired temperature so as to save energy efficiently while also being able to monitor the device status from anywhere using Blynk cloud platform and it also has Android and iOS applications which will enable us to access our device and control it remotely.

Future Works include the following

1. We can able to call to nearby fire ambulance immediately when fire is caught.
2. We can able to make the doors open automatically when fire accident is occurs
3. We can make sure that we even send a message to persons nearest neighbors such a way that they can come and stop the accident from occuring.
4. We can design the chip such a way that it is easy to handle and portable.
5. We can make possible, that whenever there is emergency occured, then automatically the information goes to police station.
6. We can implement Auto CO2 emission whenever there is emergency occurred, here, if the fire is caught to the house then the C02 gas present in the house starts on automatically.

## IX. References/Bibliography


[1] M. A. Bhuiyan, M. S. Islam, M. S. Islam, and S. M. S. Islam, "IoT-based smart air conditioning system using machine learning," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 10, pp. 11759–11769, 2021. DOI: 10.1007/s12652-021-03757-1.

[2] M. F. Fattah and M. S. Islam, "IoT-based intelligent control for thermal comfort in buildings," Sustainable Cities and Society, vol. 51, 2019, pp. 101736. DOI: 10.1016/j.scs.2019.101736.

[3] Wang, P., & Xu, Y. (2019). IoT-based thermal comfort monitoring and control in smart homes. Journal of Ambient Intelligence and Humanized Computing, 10(1), 67-76. https://doi.org/10.1007/s12652-017-0574-6

[4] Feng, L., Zhang, J., & Du, J. (2018). An IoT-based thermal comfort monitoring and control system for office buildings. Applied Sciences, 8(8), 1309. doi:10.3390/app8081309

[5] Zhang, W., Yang, L., Zhang, Y., Guo, M., & Hu, H. (2019). An IoT-based thermal comfort control system for energy-efficient buildings. Energy and Buildings, 194, 212-222. doi:10.1016/j.enbuild.2019.04.038


## X. Appendix (Code)

```
#define BLYNK_TEMPLATE_ID "TMPL_jBoi2KY"
#define BLYNK_TEMPLATE_NAME "IOT SMART AC"
#define BLYNK_AUTH_TOKEN
"xaXwfBbiiigTr3Ri_mpQiNjCGBdc8PX2" #define
BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include "DHT.h"
#define DHTTYPE DHT11
#define dht_dpin 0
DHT dht(dht_dpin, DHTTYPE);
int flamepin=D7;
int buzzer=D8;
int LED = D4;
int SENSOR_OUTPUT_PIN = D2;
int gasleak=D6;
int relay1=D0;
int relay2=D1;
```

```
int Gas_analog = A0;
BlynkTimer timer;
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "vivo1935";
char pass[] = "123498765";
void setup() {
dht.begin(); pinMode(SENSOR_OUTPUT_PIN, INPUT);
pinMode(LED, OUTPUT); pinMode(flamepin,INPUT);
pinMode(relay1,OUTPUT); pinMode(relay2,OUTPUT);
pinMode(buzzer,OUTPUT); pinMode(Gas_analog, INPUT);
pinMode(gasleak,OUTPUT); Serial.begin(115200);
Blynk.begin(auth, ssid, pass);
}
Serial.print(t);
Serial.println("°C");
Serial.println("°F");
Serial.println("Gas Sensor: ");
Serial.println(gassensorAnalog);
Blynk.virtualWrite(V0,gassensorAnalog);
int sensorvalue = digitalRead(SENSOR_OUTPUT_PIN); if
((sensorvalue== HIGH)&&(Flame==1)){
Serial.println("Human is present in the room");
digitalWrite(LED, HIGH);
Serial.println("Fire Not Detected"); digitalWrite(buzzer,
LOW); Blynk.virtualWrite(V1,1); digitalWrite(buzzer,LOW);
Blynk.virtualWrite(V7,0);
if(t>=22) {
Serial.println("Temperature is more than 22.So AC is turn
on"); digitalWrite(relay1,HIGH);
digitalWrite(relay2,LOW);
Blynk.virtualWrite(V2,1);
}
if(t>=16 & t<22){
Serial.println("Temperature is less than 20.So Fan is turn on");
digitalWrite(relay1,LOW); digitalWrite(relay2,HIGH);
Blynk.virtualWrite(V3,1);
}
if(t<16 && t>0){
Serial.println("Temperature is less than 16.So Fan and AC is
turned off");
digitalWrite(relay1,HIGH); digitalWrite(relay2,HIGH);
Blynk.virtualWrite(V3,1);
}
delay(500); }

if(gassensorAnalog>600) {
Blynk.logEvent("gas_leak"); Serial.println("Gas Leak
Detected"); digitalWrite(D6,HIGH);
Blynk.virtualWrite(V4,1);
} if(gassensorAnalog<600) {
Serial.println("AC Is Working Fine"); digitalWrite(D6,LOW);
Blynk.virtualWrite(V4,0);
}
if ((sensorvalue== LOW)&&(Flame==1)) {
Serial.println("Human is not present in the room");
Serial.println("Fire Not Detected"); digitalWrite(LED, LOW);
digitalWrite(buzzer,LOW);
Serial.println("Fan and AC is turned off");
digitalWrite(relay1,HIGH); digitalWrite(relay2,HIGH);
Blynk.virtualWrite(V2,0); Blynk.virtualWrite(V3,0);
Blynk.virtualWrite(V1,0); Blynk.virtualWrite(V7,0);
} if(Flame==0){
Blynk.logEvent("fire"); Serial.println("Fire Detected");
digitalWrite(LED, LOW); Serial.println("Fan and AC is
turned off"); digitalWrite(relay1,HIGH);
digitalWrite(relay2,HIGH); digitalWrite(buzzer,HIGH);
Blynk.virtualWrite(V2,0); Blynk.virtualWrite(V3,0);
Blynk.virtualWrite(V1,0); Blynk.virtualWrite(V7,1);
} Blynk.run(); timer.run();
}

digitalWrite(buzzer, LOW); Blynk.virtualWrite(V1,1);
digitalWrite(buzzer,LOW); Blynk.virtualWrite(V7,0);
if(t>=22)
{
Serial.println("Temperature is more than 22.So AC is turn
on");
digitalWrite(relay1,HIGH); digitalWrite(relay2,LOW);
Blynk.virtualWrite(V2,1);
}
if(t>=16 & t<22){
Serial.println("Temperature is less than 20.So Fan is turn on");
digitalWrite(relay1,LOW);
digitalWrite(relay2,HIGH);
Blynk.virtualWrite(V3,1);
}
if(t<16 && t>0){
Serial.println("Temperature is less than 16.So Fan and AC is
turned off"); digitalWrite(relay1,HIGH);
digitalWrite(relay2,HIGH);
Blynk.virtualWrite(V3,1);
}
delay(500); }
if(gassensorAnalog>600) {
Blynk.logEvent("gas_leak"); Serial.println("Gas Leak
Detected"); digitalWrite(D6,HIGH);
Blynk.virtualWrite(V4,1);
} if(gassensorAnalog<600) {
Serial.println("AC Is Working Fine"); digitalWrite(D6,LOW);
Blynk.virtualWrite(V4,0);
}
if ((sensorvalue== LOW)&&(Flame==1)) {
Serial.println("Human is not present in the room");
Serial.println("Fire Not Detected"); digitalWrite(LED, LOW);
digitalWrite(buzzer,LOW);
Serial.println("Fan and AC is turned off");
digitalWrite(relay1,HIGH); digitalWrite(relay2,HIGH);
Blynk.virtualWrite(V2,0); Blynk.virtualWrite(V3,0);
Blynk.virtualWrite(V1,0); Blynk.virtualWrite(V7,0);
} if(Flame==0){
Blynk.logEvent("fire"); Serial.println("Fire Detected");
digitalWrite(LED, LOW); Serial.println("Fan and AC is
turned off"); digitalWrite(relay1,HIGH);
digitalWrite(relay2,HIGH); digitalWrite(buzzer,HIGH);
Blynk.virtualWrite(V2,0); Blynk.virtualWrite(V3,0);
Blynk.virtualWrite(V1,0); Blynk.virtualWrite(V7,1);
} Blynk.run(); timer.run();
}
```

```
Serial.println("°C");
Serial.println("°F");
Serial.println("Gas Sensor: ");
Serial.println(gassensorAnalog);
Blynk.virtualWrite(V0,gassensorAnalog);
int sensorvalue = digitalRead(SENSOR_OUTPUT_PIN); if
((sensorvalue== HIGH)&&(Flame==1)){
Serial.println("Human is present in the room");
digitalWrite(LED, HIGH);
Serial.println("Fire Not Detected"); digitalWrite(buzzer,
LOW); Blynk.virtualWrite(V1,1); digitalWrite(buzzer,LOW);
```