1. **INTRODUCTION**

Most of the human life is spent indoors, either at home, jobs or at schools. The air at these places must be properly monitored, as it is inhaled by human beings in large amounts during respiration. Indoor air pollution refers to the degradation of air at indoors. The main sources of indoor air pollution are biomass smoke, fireplaces, stoves, biological contaminants like pollen, household products, moulds, chimneys which contain pollutants such as particulate matter, nitrogen dioxide(NO2), sulphur dioxide(SO2) and carbon monoxide(CO), pesticides, materials used in the buildings such as asbestos, formaldehyde and lead . Most of the indoor air pollution is of human origin and measured in OLF units (emission rate of air pollution from a standard person). In reality, 15% of the indoor air pollution is caused by humans and 85% from non-human sources . Some other pollutants are radon, environmental tobacco smoke, Ammonia, carbon dioxide, and benzene.

It has adverse effects on health, i.e., exposure to high concentrations of certain pollutants may cause immediate death. According to WHO, nearly 2 million people die every year, prematurely, due to diseases caused because of usage of solid fuel. Almost 50% of pneumonia deaths in children are due to particulate matter inhaled from the indoor air. A lot of people die from chronic obstructive respiratory disease (COPD) that develop due to exposure to indoor smoke. Indoor air pollution may cause several diseases like lung cancer, flu, respiratory irritations, loss of coordination, asthma, and allergic reactions. Due to the serious effects caused by Indoor Air Pollution, there is a need for it to be monitored and controlled.

The CO2 level in air exhaled by humans is about 38,000 to 48,000 parts per million(ppm). Since predictable levels of carbon dioxide is exhaled by people in the indoors, its amount in the indoor air can be used as a significant indication of the quality of air. The adequacy of the supply of fresh air is often assessed by the level of CO2 in a room. An indoor CO2 concentration of less than 1000 ppm indicates the fresh air supply. The main sources of CO2 are human metabolic activities and combustion sources. Some other sources are kerosene, gas space heaters, and tobacco smoke. This IoT project uses Air quality sensor MQ135 to monitor the air pollution in its surroundings. According to the datasheet of MQ135, the sensor is sensitive for CO2, Alcohol, Benzene, NOx, and NH3. But CO2 is the fourth most abundant gas in the earth’s atmosphere with a concentration of about 409.8 ppm. This concentration is much higher than all the other gases the sensor detects. So, with the proper calibration of the sensor, the sensor can detect CO2 in the normal atmosphere.

|  |  |
| --- | --- |
| Concentration | Effect |
| 250-410ppm | Normal concentration in outdoor ambient air |
| 410-1000 ppm | Concentrations of occupied indoor spaces with good air exchange |
| 1000-2000 ppm | Poor air and causes drowsiness |
| 2000-5000 ppm | Stagnant and stuffy air. Headaches, sleepiness, increased heart rate and slight nausea may also be present |
| 5000 ppm | Workplace exposure limit |
| >40,000 ppm | Exposure may lead to serious oxygen deprivation resulting in permanent brain damage, coma, even death |

**Table 1:** Carbon dioxide levels and health problems

According to the National Institute for Occupational Safety and Health (NIOSH), the indoor air concentrations of carbon dioxide that exceed 1,000 ppm indicate inadequate ventilation.

In this system, the quantity of carbon dioxide in the indoor air is observed regularly and when the concentration goes beyond the safer level, i.e., 1000 ppm, the user will be notified through SMS and email alerts, and a fan is switched on/off automatically to increase ventilation. This fan can be an exhaust fan or a fan placed as close as possible to a window to push outdoor air into indoors or indoor air out.

* 1. **Scope**

This system can be used as a device in schools, homes, shopping malls or any other indoor places to sense quantity of CO2 in the air. In this IoT system, the quantity of carbon dioxide in the indoor air is observed regularly and when the concentration goes beyond the safer level, i.e., 1000 ppm, the user will be notified through SMS and email alerts, and a fan is switched on/off automatically to increase ventilation. This fan can be an exhaust fan or a fan placed as close as possible to a window to push outdoor air into indoors or indoor air out. Users are given instructions in alerts which help in increasing ventilation.

* + 1. **Helps in preventing the spread of air borne virus outbreak indoors**

The vast majority of SARS-CoV-2 transmission occurs [indoors](https://doi.org/10.1111/ina.12697), most of it from the [inhalation of airborne particles](https://doi.org/10.1016/j.envint.2020.105730) that contain the coronavirus. The best way to prevent the virus from spreading in a home or business would be to simply keep infected people away. But this is hard to do when an estimated [40% of cases are asymptomatic](https://theconversation.com/can-people-spread-the-coronavirus-if-they-dont-have-symptoms-5-questions-answered-about-asymptomatic-covid-19-140531) and asymptomatic people can [still spread the coronavirus to others](https://dx.doi.org/10.3201%2Feid2607.201595).

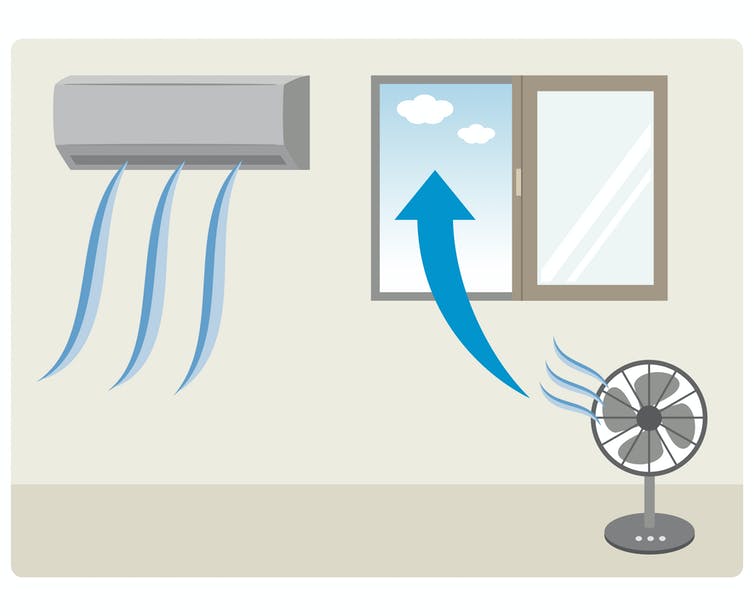
[Masks do a decent job](https://dx.doi.org/10.1126/science.abc6197) at keeping the virus from spreading into the environment, but if an infected person is inside a building, inevitably some virus will escape into the air.

Once the virus escapes into the air inside a building, we have two options: bring in fresh air from outside or remove the virus from the air inside the building.

**Using CO2 to measure air circulation**

Every time we exhale, we [release CO2](https://doi.org/10.1111/ina.12383) into the air. Since the coronavirus is most often spread by breathing, coughing or talking, we can use [CO2 levels](https://pdfs.semanticscholar.org/dd7e/b2870c38f70e5285e5118ed6f158c091f7cf.pdf) to see if the room is filling up with potentially infectious exhalations. The CO2 level lets you estimate if enough fresh outside air is getting in.

Outdoors, CO2 levels are just above 400 parts per million (ppm). A well-ventilated room will have around [800 ppm of CO2](https://doi.org/10.1111/j.1600-0668.1999.00003.x). Any higher than that and it is a sign the room might need more ventilation.



**Fig 1**: Fan near window to improve ventilation

Last year, researchers in Taiwan reported on the [effect of ventilation on a tuberculosis outbreak](https://doi.org/10.1111/ina.12639) at Taipei University. Many of the rooms in the school were under ventilated and had CO2 levels above 3,000 ppm. When engineers improved air circulation and got CO2 levels under 600 ppm, [the outbreak completely stopped](https://doi.org/10.1111/ina.12639). According to the research, the increase in ventilation was responsible for 97% of the decrease in transmission.

Since the coronavirus is spread through the air, higher CO2 levels in a room likely mean there is a [higher chance of transmission](https://doi.org/10.1111/ina.12639) if an infected person is inside.

This is where our system plays a crucial role, whenever CO2 is above 1000 ppm it alerts user and switches fan ON which pumps fresh air indoors and thereby helps in preventing virus outbreak.

* + 1. **Decreases indoor pollution**

By pumping fresh outdoor air into indoors, polluted indoor air gets diluted thereby decreasing indoor pollution. This decreases all the negative impacts of high CO2 environment or high air pollution environment and improves better sleep, cognitive and decision making and performance of an individual.

**1.2 Existing System**

Existing system does not come with a fan which improves rate of ventilation. Existing system has NodeMCU (Micro-Controller), MQ-135 air quality sensor and ThingSpeak IoT platform. It can only sense and send alert to user but doesn’t play any role in improving ventilation.

**The following are the drawbacks of the existing system:**

* Less active: Only alerts user
* Doesn’t have a ventilation fan, doesn’t decrease indoor pollution
  1. **Proposed System**

Proposed system has fan controlled by micro-controller. Fan is powered ON when CO2 level is above 1000 ppm and powered OFF when CO2 level goes below 800 ppm.

**Merits:**

* More active: Sends alerts and powers fan ON/OFF based on CO2 levels indoors
* Helps in decreasing indoor pollution

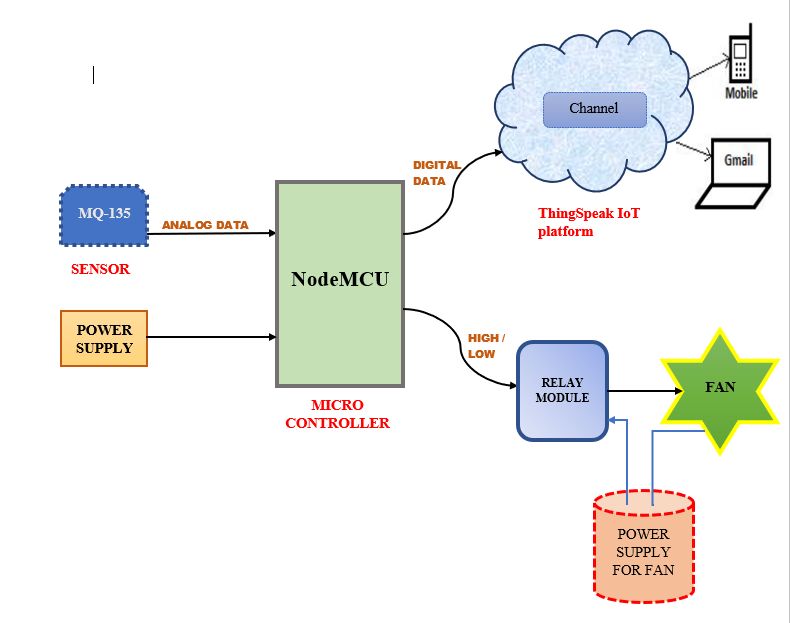
1. **SYSTEM ANALYSIS**

This System Analysis is closely related to [requirements analysis](http://en.wikipedia.org/wiki/Requirement_analysis). It is also "an explicit formal inquiry carried out to help someone (referred to as the decision maker) identify a better course of action and make a better decision than he might otherwise have made."This step involves [breaking down](http://en.wikipedia.org/wiki/Work_breakdown_structure) the system in different pieces to analyse the situation, analysing project goals, breaking down what needs to be created and attempting to engage users so that definite requirements can be defined.

* 1. **Functional Requirement Specification**
* System should be able to sense and measure air quality i.e.CO2
* System should be able to connect to a Wi-Fi network
* System should be able to send measured values to ThingSpeak IoT platform when connected to a Wi-Fi network
* System should be able to switch fan ON/OFF based on conditions automatically, even when not connected to a Wi-Fi network
* System should be able to display on a screen
* System should be able to send alerts to user through SMS and email
  1. **Performance Requirements:**
* Sensor should have fast response and high sensitivity
* Components should be durable
* Reading should be proportional to air quality
  1. **Software Requirements:**
* Arduino IDE (software)
* ThingSpeak IoT Platform
  1. **Hardware Requirements:**
* ESP8266 based NodeMCU-12E
* MQ135 Air Quality Sensor
* Wi-Fi Access
* Mobile phone to receive messages
* Breadboard and Wires
* Relay Module
* Battery
* Ventilation Fan
* Computer

**3.SYSTEM DESIGN**

**3.1 Architectural Design**

****

**Fig 3.1**: Architecture Diagram

NodeMCU is connected to a computer through USB. Computer acts as both power supply and display screen.

MQ135 is connected to micro-controller in following way:

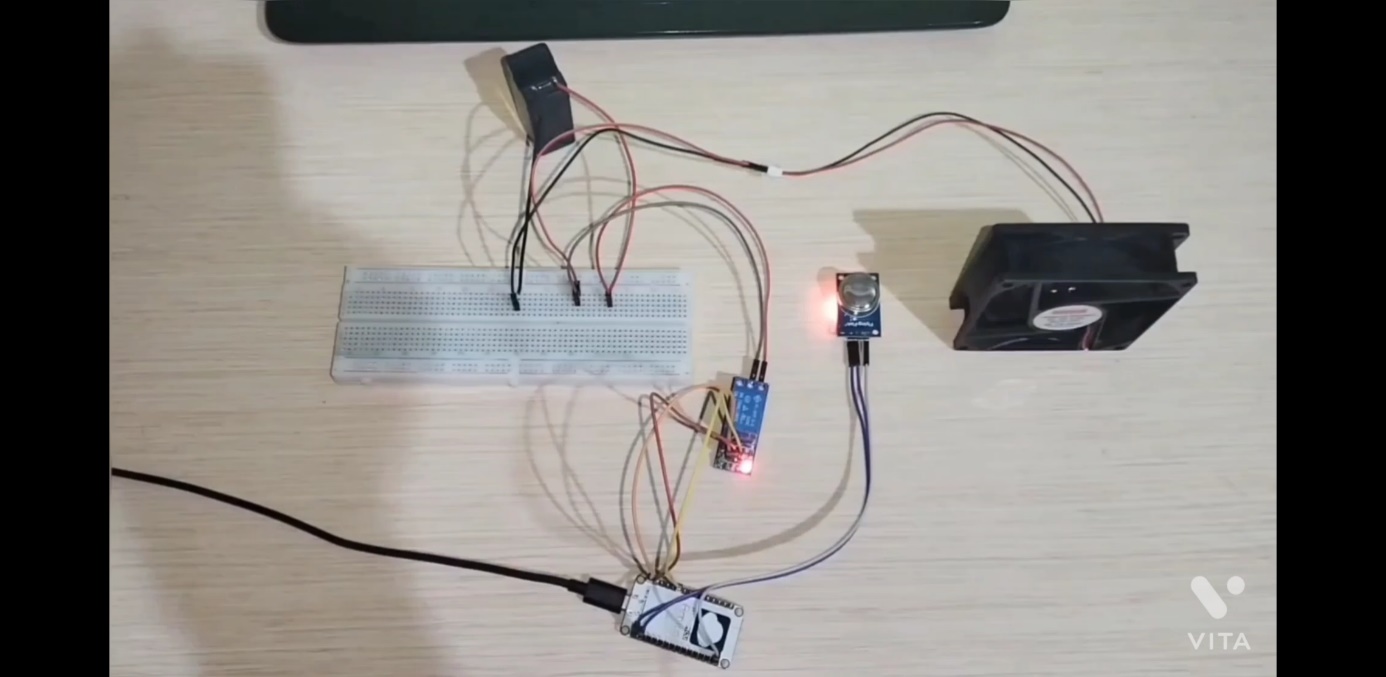
* A0 of sensor to A0 of micro-controller
* Vcc of sensor to Vin of micro-controller
* Gnd of sensor to Gnd of micro-controller

Relay module is connected to micro-controller in following way:

* IN of relay module to D7 of micro-controller
* Vcc of sensor to 3V3 of micro-controller
* Gnd of sensor to Gnd of micro-controller

Other connections are done in following way:

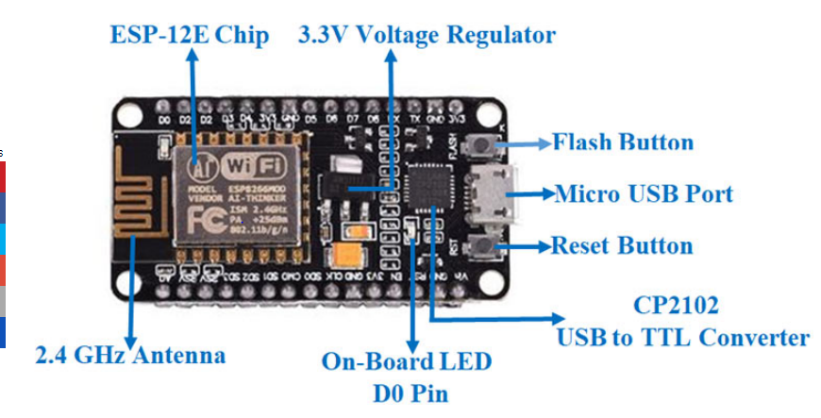
* Positive terminal of battery to the Common (C) of Relay
* Negative terminal of battery to the fan's black wire and Normally Open (NO) of the relay connect to fan's red wire.



**Fig 3.2**: Components and connections

**3.1.1 NodeMCU**

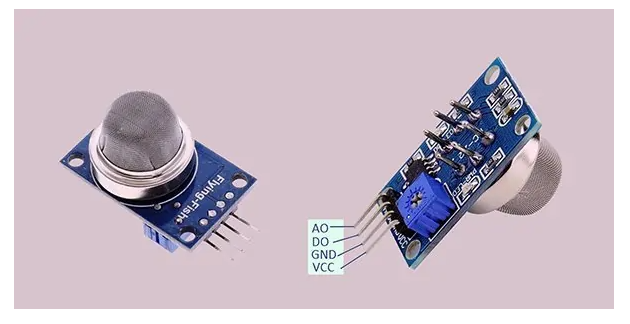
NodeMCU (Node Microcontroller Unit) is a microcontroller with built-in support for Wi-Fi connectivity, which makes it more useful for IoT applications. It is built around a very inexpensive System-On-Chip, i.e., ESP8266, which contains all the crucial elements of a modern computer, like CPU, RAM, networking(Wi-Fi), a modern OS and SDK (Software Development Kit).

****

**Fig 3.3:** NodeMCU

**3.1.2 MQ135 Air Quality Sensor**

The sensor used is Air Quality sensor, MQ135. It senses the presence of different kinds of toxic gases such as Ammonia, Smoke, CO2 Benzene, alcohol, and NOx. However, due to the abundance of CO2 compared to others, this sensor is used for measuring the concentration of CO2. Before measuring, the sensor is calibrated by running the calibration code in Arduino IDE, which uses getRZero function to find calibration resistance. It is run for 23-24 hours and the value of calibration resistance thus obtained is updated in the ‘MQ135.h’ code in the MQ135 library. One of the important feature of air quality sensor is its fast response and high sensitivity.

****

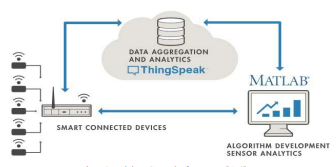
**Fig 3.4:** MQ135 Air Quality Sensor

**3.1.3 ThingSpeak IoT Platform**

The IoT platform used in this project is ‘ThingSpeak’, an open-source IoT platform to store and retrieve data using the HTTP protocol over the Internet. The data measured by the circuit is sent to this platform by NodeMCU, using channel ID, Write API Key and the field (in which it has to load the data) provided in the code. This open source platform allows the data to be analysed and visualised.

Whenever a new channel is created, a unique ID, called Channel ID is created. The API keys are generated for writing the data on the channel and for reading data from the channel. The entries from the microcontroller are plotted as a graph. When the measured concentration increases beyond the safer level (saved in React app of ThingSpeak), then alert is sent to the user’s mobile using IF This Then That (IFTTT) application.

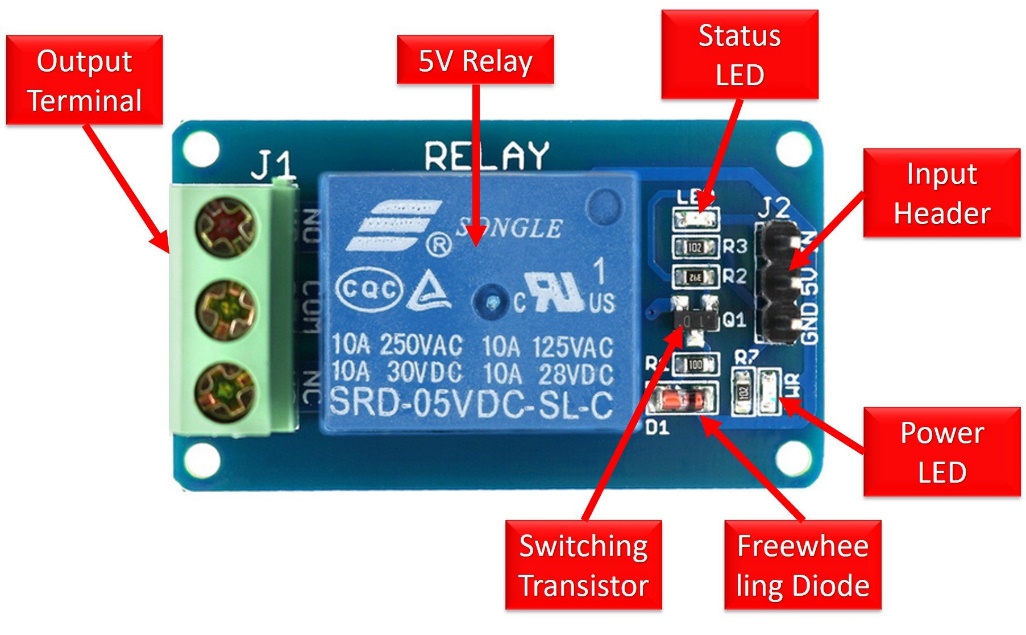
In ThingSpeak, the data is sent and received through “Hypertext Transfer Protocol” (HTTP) protocol . When a device sends the data via a HTTP request, it is processed by the ThingSpeak platform, which communicates with a virtual server. Both the server and the IoT platform communicate directly with the application. This data transfer can occur through plaintext, XML or JSON . Then the data is uploaded onto the cloud. Based on the conditions set on the data, the messages will be send to the user. Messages from ThingSpeak are communicated to user via a HTTP GET request method, which retrieves information from the server.



**Fig 3.5**: ThingSpeak for IoT

**3.1.4 Relay module**

A power relay module is an electrical switch that is operated by an electromagnet. The electromagnet is activated by a separate low-power signal from a micro controller. When activated, the electromagnet pulls to either open or close an electrical circuit.

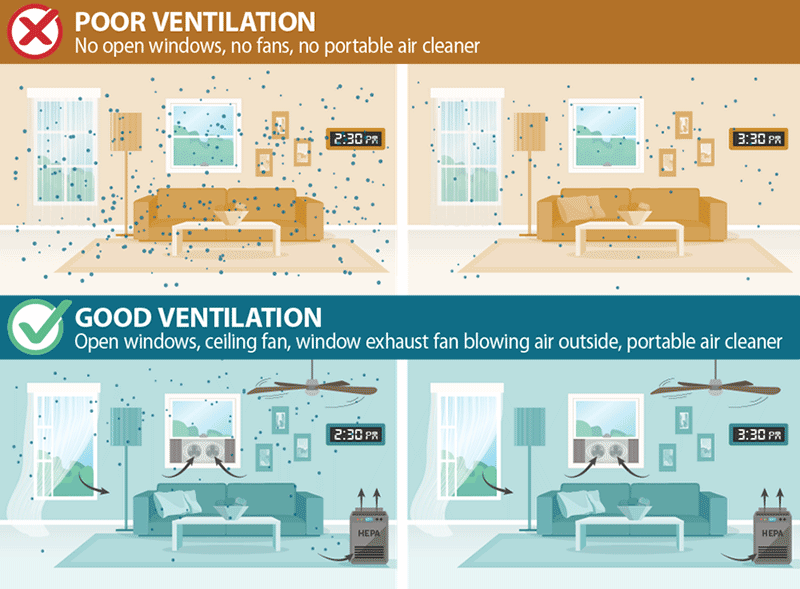
****

**Fig 3.6:** Relay module

* 1. **Modules**

1. **Physical Module:**

. Physical module consists of all the hardware components which sense air quality, control fan and display values. It functions even when Wi-Fi connection is not available. This is responsible for increasing rate of ventilation in room.



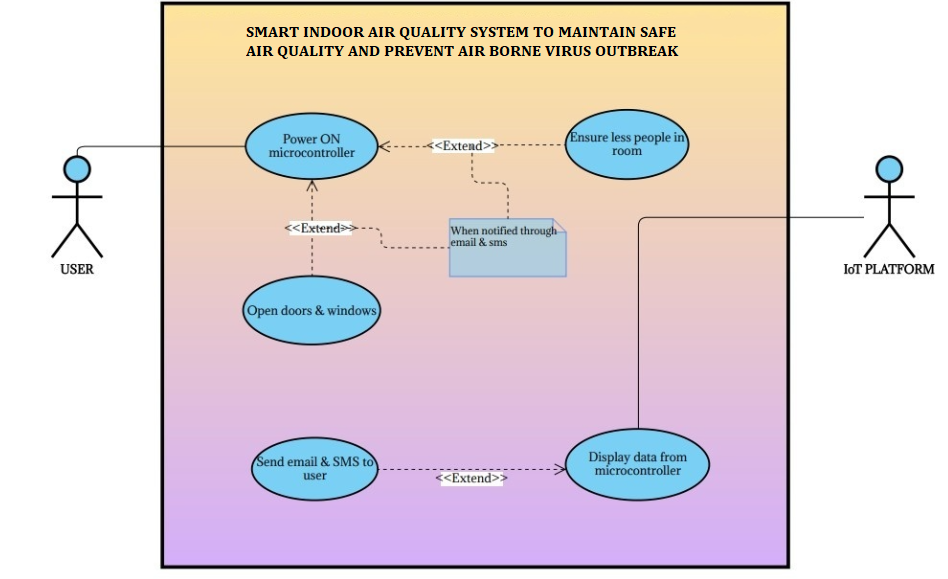
**Fig 3.7:** Ventilation comparision

1. **ThingSpeak Module:**

ThingSpeak module is connected to physical module through a Wi-Fi connection. It is responsible for receiving data from physical module, displaying them in ThingSpeak channel and sending SMS and email alerts to user. Channel’s API key and ID are provided in the program fed into micro-controller. This is responsible for alerting user through SMS and email.

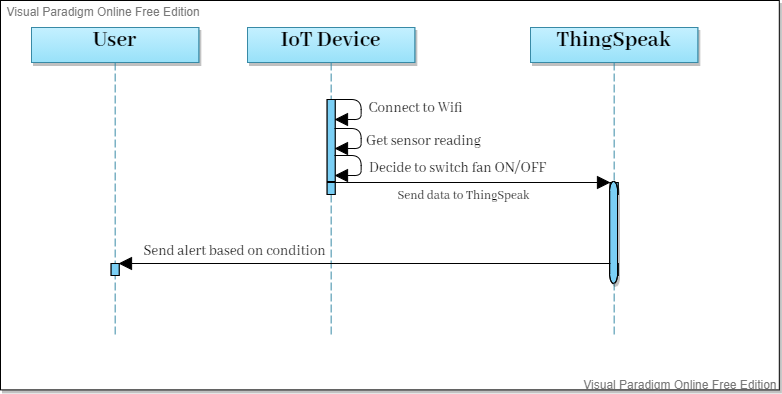
* 1. **UML Diagrams**

**3.3.1 Use Case Diagrams**

****

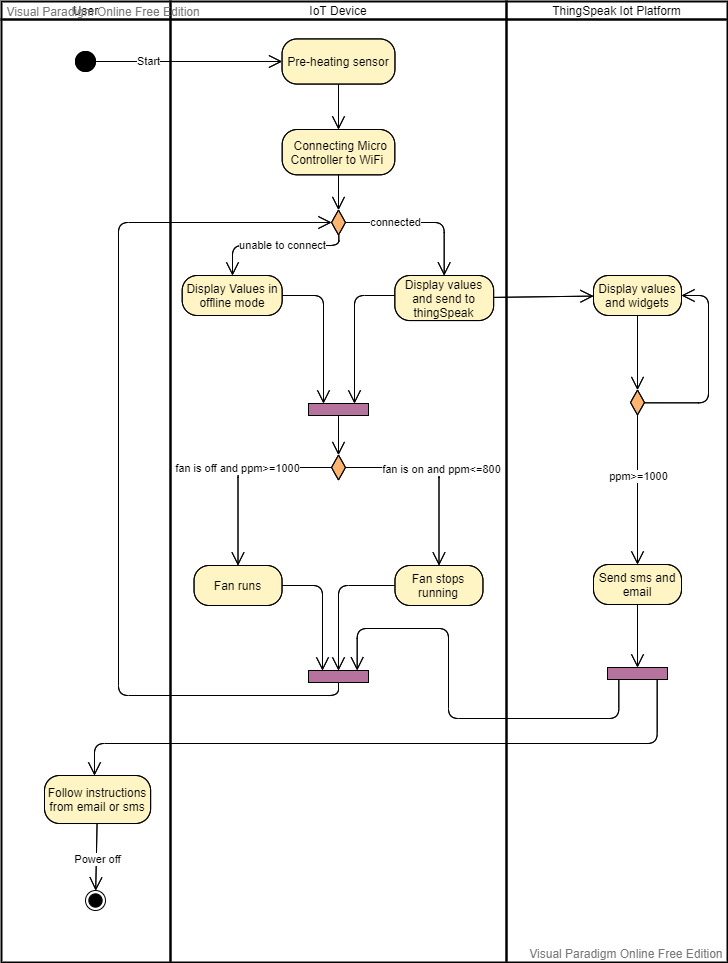
**Fig 3.8**: Use Case Diagram

**3.3.2 Sequence Diagram:**

****

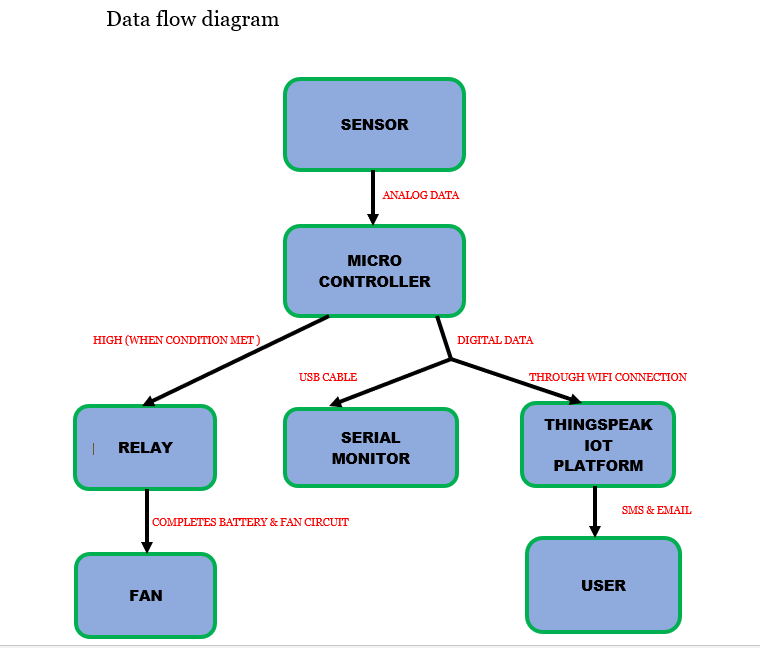
**Fig 3.9:** Sequence Diagram

**3.3.3 Activity Diagram:**

****

**Fig 3.10:** Activity Diagram

**3.3.4 Data Flow Diagram:**

****

**Fig 3.11:** Data Flow Diagram

**4.SYSTEM IMPLEMENTATION**

The NodeMCU microcontroller can be powered by a 9V battery (since a 3.3V voltage regulator is present) or by computer through a regular USB cable. One of the three 3.3V pin on the microcontroller is used to power the sensor and the analog output pin of the sensor is connected to the analog pin. And other connections are made as given architecture diagram. The Arduino IDE supports NodeMCU board. The program with corresponding libraries is to be downloaded and some settings like the board used, flash size and upload speed are to be changed accordingly.

The code is written and uploaded onto the microcontroller Wi-Fi access is provided. Now, the sensor starts measuring the quality of air and gives it to the microcontroller in voltage form, which is transformed into ppm by NodeMCU using inbuilt (library) code. Then the ESP8266 module on the board connects to Wi-Fi with the details provided in the code, using which the microcontroller sends the data to the IoT platform.

When a Wi-Fi network is not available, system is unable to send alerts but it displays execution and ppm values on the computer screen and it controls fan to improve ventilation in room automatically. Fan is powered on when ppm > 1000 and powered off when ppm < 800.

When a Wi-Fi network is available,the data measured by the sensor and status of fan is fed to the field in a channel of the ThingSpeak platform. Thus, the data measured at a place by a sensor can be accessed by the user from anywhere. The maximum allowable safer level of concentration of CO2 in the atmosphere that does not much affect human health is 1000 ppm.

A react (an app which works with ThingHTTP, ThingTweet, and Matlab Analysis apps to perform actions when channel data meets a certain condition) is created which triggers another app, ThingHTTP if concentration crosses 1000 ppm. The ThingHTTP app enables communication among devices, websites, and web services without implementing the protocol on the device level. A new ThingHTTP request is created and the actions are specified. It is triggered using React app when the concentration of carbon dioxide increases beyond 1000 ppm. Then an application IFTTT is used to send SMS to the user’s mobile. Based on the alert, the user can do actions such as opening the windows and doors, ensure less people in room (or) to leave the place for some time. The system is thus built using NodeMCU and the quality of air in the surrounding is monitored and controlled using a fan which increases rate of ventilation in room.

**4.1 Program:**

Program for system is written in Arduino IDE and then uploaded into micro-controller. General Arduino IDE code consists of two basic functions setup() and loop(), setup() is executed initially when system is started where as loop() is executed repeatedly after completion of setup().

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Program \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

#include "MQ135.h"

#include <ESP8266WiFi.h>

#include <ThingSpeak.h>

const char \*apiKey = "channelAPI";

const char \*ssid = "WiFi Username" ;

const char \*pass = "Wifi Password";

unsigned long myChannel = Channel\_ID;

const int relayInput = 13;

WiFiClient client;

unsigned int output\_pin = A0;

MQ135 gasSensor = MQ135(output\_pin);

float ppm;

int offline = 0;

int alert = 0;

int fan = 0;

int first = 0;

void setup(){

Serial.begin(115200);

delay(10);

preHeat();

WiFi.mode(WIFI\_STA);

internet();

ThingSpeak.begin(client); // Initialize ThingSpeak

pinMode(relayInput,OUTPUT);

digitalWrite(relayInput,HIGH);

}

void loop(){

get\_value();

if(!offline) upload();

else delay(2000);

Serial.println("-------------------------------------------------");

}

void internet(){

Serial.print("Trying to connect with -> ");

Serial.println(ssid);

WiFi.begin(ssid,pass);

unsigned long ElapsedTime;

unsigned long StartTime = millis();

while(WiFi.status() != WL\_CONNECTED)

{

delay(1000);

Serial.print(". ");

ElapsedTime = millis() - StartTime;

if(ElapsedTime >= 60000){

break;

}

}

Serial.println("");

if(WiFi.status() != WL\_CONNECTED){

offline = 1;

Serial.println("\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*");

Serial.println("\* Unable to connect to WiFi... \*");

Serial.println("\* CO2 levels in PPM (offline mode)... \*");

Serial.println("\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*");

}

else{

Serial.println("WiFi connected");

Serial.println("\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*");

}

}

void preHeat(){

Serial.println("Pre-Heating sensor");

delay(30000);

Serial.println("Pre-Heating done");

}

void get\_value(){

float rzero = gasSensor.getRZero();

//Serial.print(rzero);

//Serial.print(" ");

Serial.print("PPM : ");

ppm = gasSensor.getPPM();

Serial.println(ppm);

if(alert and ppm<=800){

alert = 0;

//code here for making fan pin low

digitalWrite(relayInput,HIGH);

Serial.println("-------------------------------------------------");

Serial.println("\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*");

Serial.println("\* Fan is switched off... \*");

fan = 0;

Serial.println("\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*");

}

if(!alert and ppm>=1000){

alert = 1;

Serial.println("-------------------------------------------------");

Serial.println("\*-\*-\*-\*-\*-\*-\*::: Alert :::-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*\*");

Serial.println("\* PPM greater than 1000 \*");

// code here for making fan pin high

digitalWrite(relayInput,LOW);

Serial.println("\* Fan is switched on... \*");

fan = 1;

first = 1;

Serial.println("\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*");

}

}

void upload(){

ThingSpeak.setField(1, ppm);

ThingSpeak.setField(2, fan);

ThingSpeak.setField(3, first);

int x = ThingSpeak.writeFields(myChannel, apiKey);

if(first)first = 0;

if(x == 200){

Serial.println("Channel update successful.");

Serial.println("-------------------------------------------------");

for(int i=0;i<10;i++){

get\_value();

Serial.println("-------------------------------------------------");

delay(2000);

}

}

else{

Serial.println("Problem updating channel. HTTP error code " + String(x));

internet();

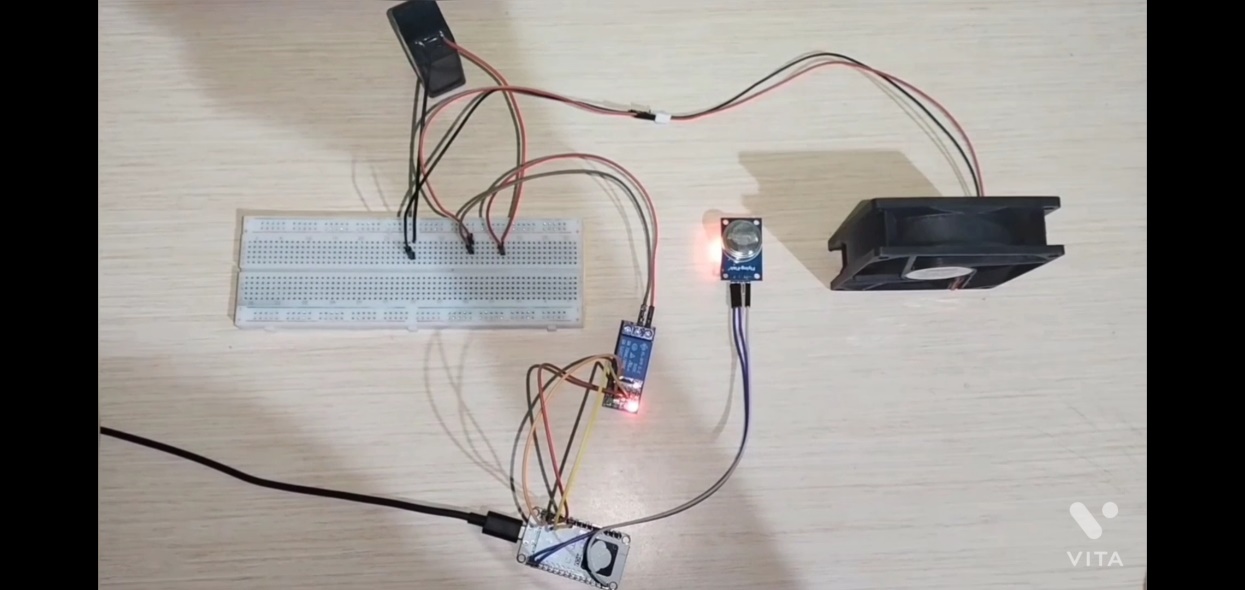
}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**4.2 Connections**

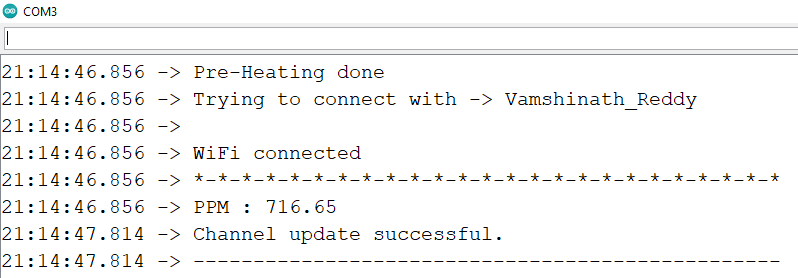
Connections are done as given in architectural diagram (refer section 3.1).



**Fig 4.1:** Connections

**5. OUTPUT**

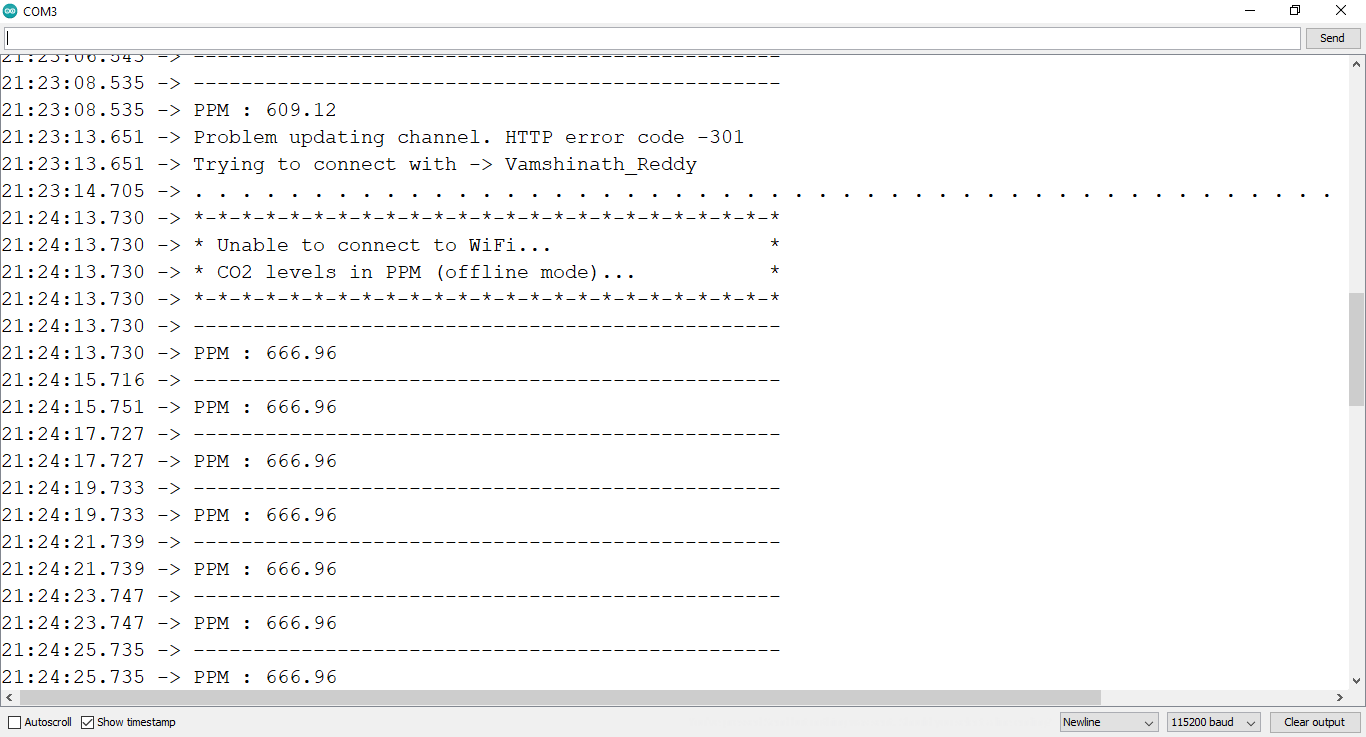
Output of various functionalities in our system are shown over here along with the description.



**Fig 5.1**: Starting phase of system

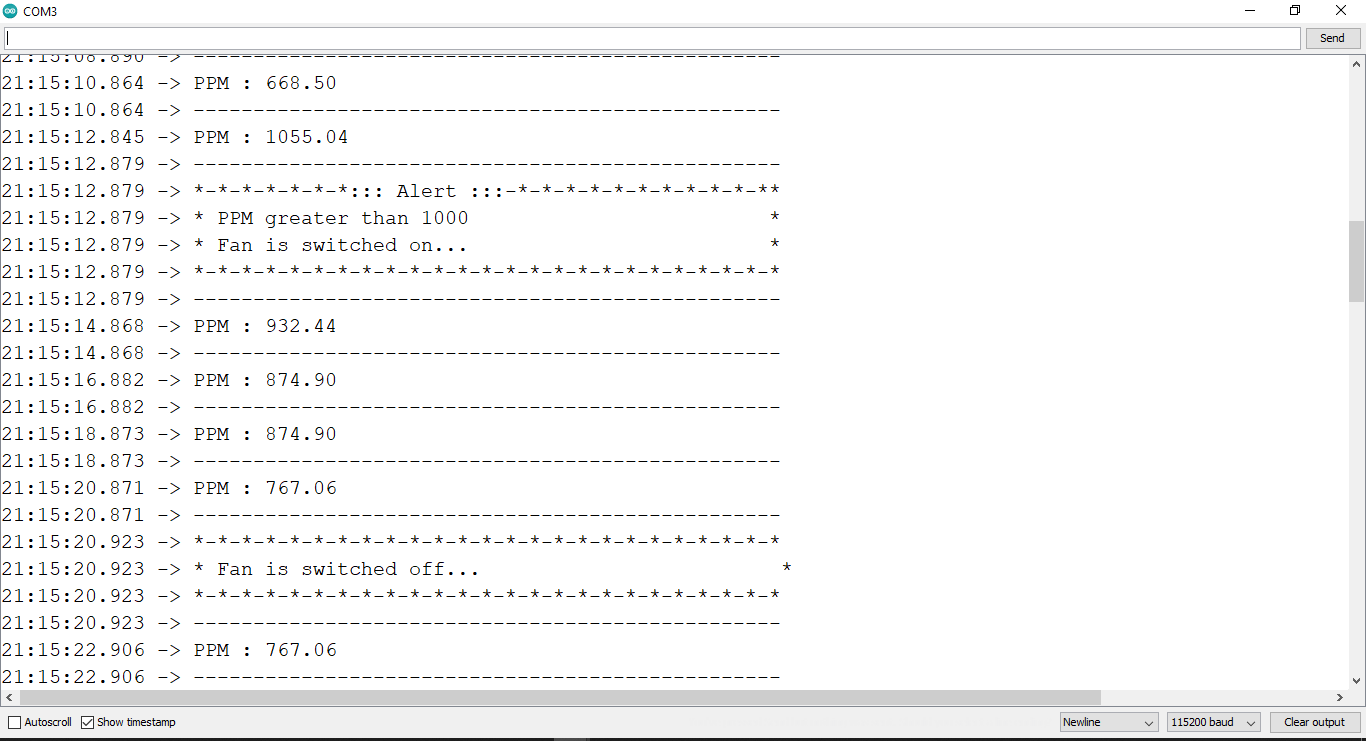
As soon as the USB is connected to the micro-controller, system starts displaying on serial monitor. Initially sensor requires some pre heating to work accurately and efficiently, the pre heat function here as a delay of 5 minutes during which sensor gets heated.

After the pre-heating of sensor, the NodeMCU tries to connect to Wi-Fi network provided in code. After successful connection values are received from sensor and are fed into ThingSpeak channel.



**Fig 5.2**: Serial monitor display screen when Wi-Fi connection is lost

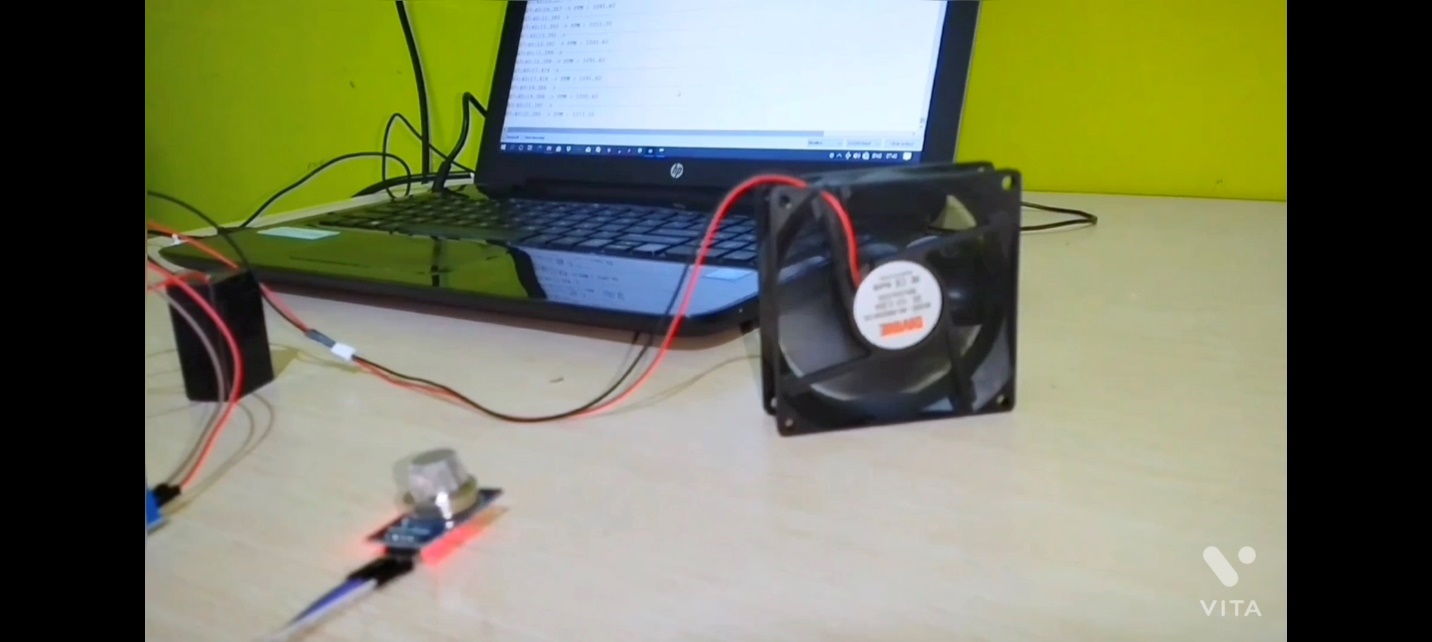
If Wi-Fi connection is not available initially or if Wi-Fi connection is lost during system operation, system turns itself into offline mode and starts displaying ppm values with 2 seconds delay. In this mode, system is unable to send alerts to the user but it can still control fan which helps in improving rate of ventilation indoors.



**Fig 5.3**: Serial monitor display screen when CO2 ppm is above 1000 ppm

When the value of ppm measured goes beyond 1000, system displays an alert message as shown in the above figure. It also powers fan ON and displays the message on serial monitor of Arduino IDE.

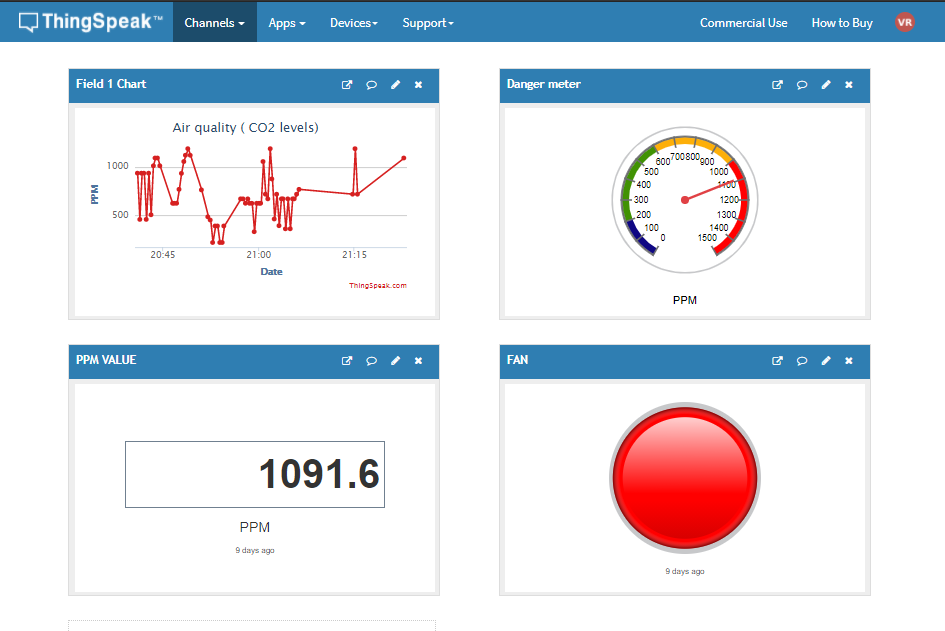
When the ppm is above 1000, a high signal is sent from NodeMCU to relay module. On receiving signal, relay module makes the fan and battery circuit complete, which turns fan ON. A LED on relay indicates the status of fan and battery circuit.



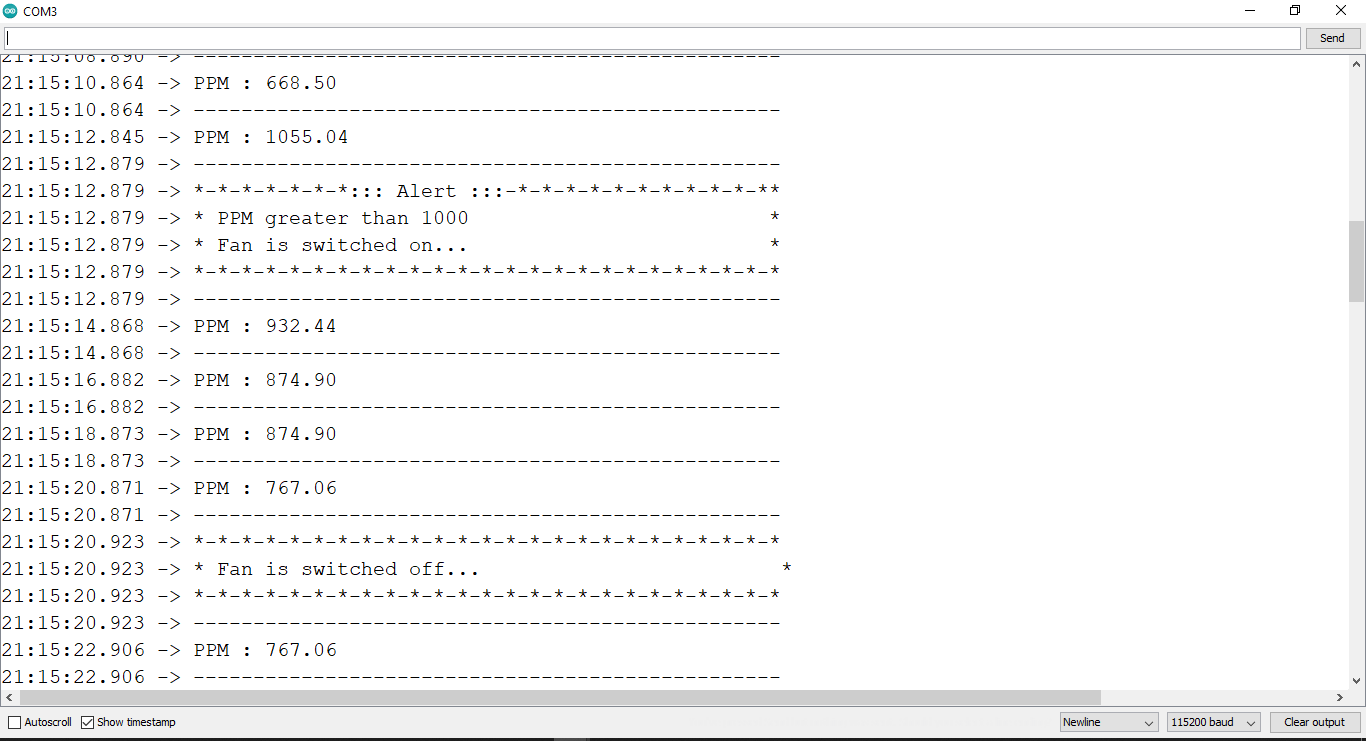
**Fig 5.4**: Fan running when CO2 is above 1000 ppm



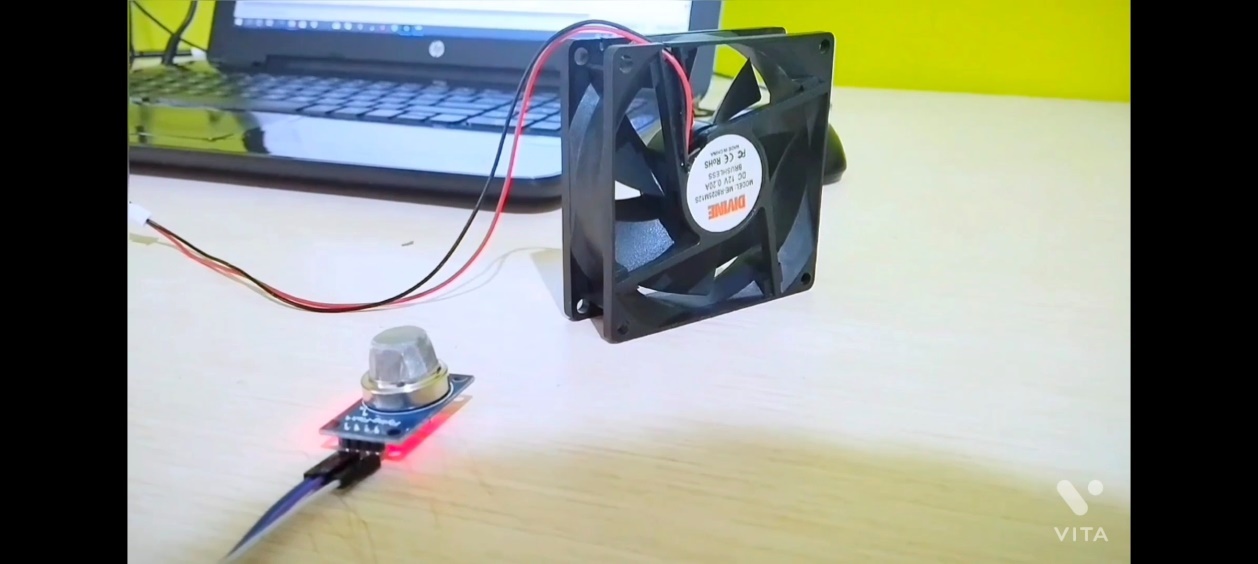
**Fig 5.5:** Demonstration of working system by generating CO2 and placing sensor in it



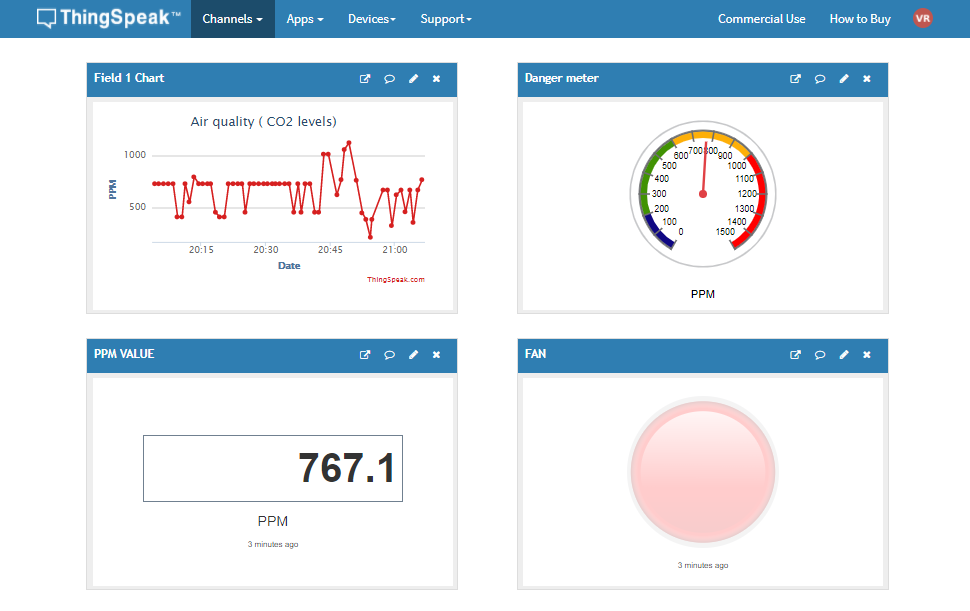
**Fig 5.6:** ThingSpeak channel visualization and widgets when received ppm is above 1000



**Fig 5.7:** Serial monitor display screen when ppm goes from above 1000 to below 800

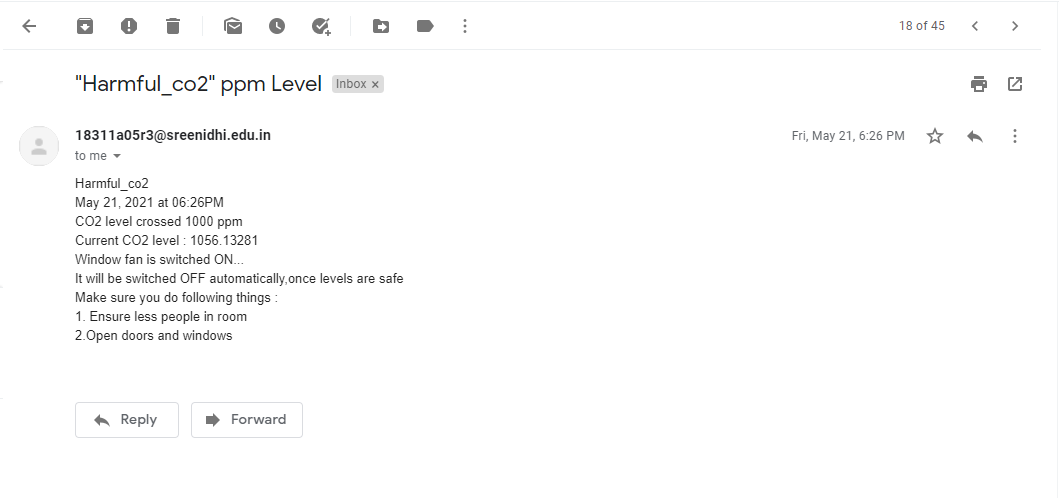


**Fig 5.8**: Fan in OFF state during safe indoor air quality environment

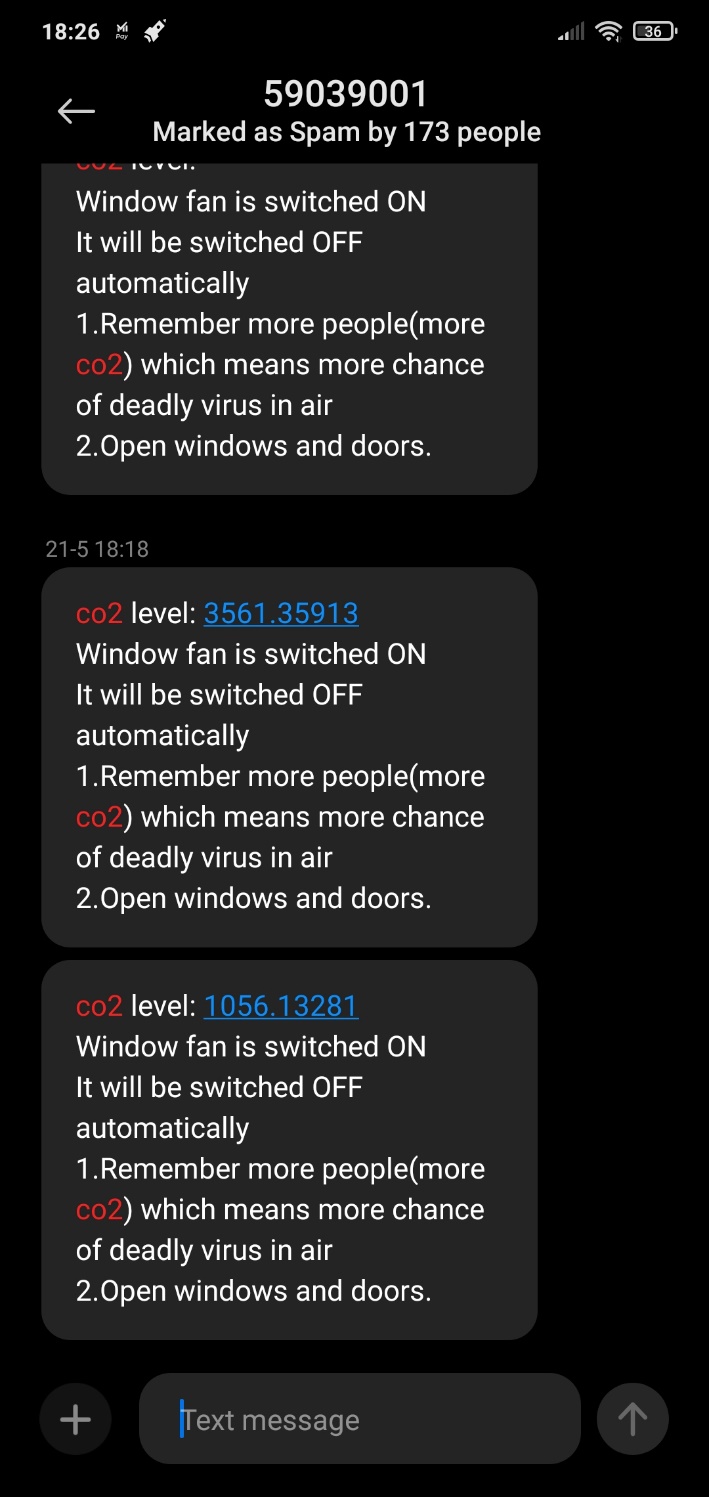


**Fig 5.9:** ThingSpeak channel when indoor air quality is safe and fan is in OFF mode

When CO2 level goes from above 1000 ppm to below 800 ppm, fan is turned OFF and this information is fed into ThingSpeak channel.



**Fig 5.10:** E-mail alert



**Fig 5.11:** SMS alert

**7. CONCLUSION AND FUTURE SCOPE**

**Conclusion**

The system to monitor the indoor air pollution is designed and the following advantages are observed in comparison to the existing models.

* The designed system not only monitors but also helps in improving ventilation in indoors.
* The designed system is cost effective compared to the conventional air pollution monitoring systems available in the market.
* There is no requirement of any smartphone application installed in users mobile to receive the alerts from the system. Only the system is to be connected to the Internet using NodeMCU microcontroller so that data is sent to cloud.
* The channel in the IoT platform can be set to private so that the data in the cloud is secured.
* Many deadly effects of high indoor air pollution, for ex : virus outbreak, insomnia etc, can be prevented using this system since it reduces indoor air pollution.

**Future Scope**

The proposed system has a fan with wired connection, which can be developed into wireless connection. The proposed system monitors the indoor air pollution and uses ThingSpeak as the IoT platform. Amazon Web Services IoT core is the similar type of platform, but with much more security and reliability. The AWS IoT device SDK allows the devices to connect, authenticate and exchange messages with AWS IoT Core using the MQTT, HTTP, or WebSockets protocols. The circuit can further be developed by using MQTT and AWS platform. Sensors measuring other dangerous pollutants can be included in the system along with temperature, humidity sensors to make the system monitor and control the ambience of a place.

**BIBLIOGRAPHY**

[1][*https://ieeexplore.ieee.org/document/9104080*](https://ieeexplore.ieee.org/document/9104080)

[2][*https://www.codrey.com/electronic-circuits/how-to-use-mq-135-gas-sensor/*](https://www.codrey.com/electronic-circuits/how-to-use-mq-135-gas-sensor/)

[3] [*https://components101.com/sensors/mq135-gas-sensor-for-air-quality*](https://components101.com/sensors/mq135-gas-sensor-for-air-quality)

[4] [*https://medium.com/@joeljean/im-living-in-a-carbon-bubble-literally- b7c391e8ab6#:~:text=Remember*](https://medium.com/@joeljean/im-living-in-a-carbon-bubble-literally-%20%20%20%20%20b7c391e8ab6#:~:text=Remember)

[5][*https://www.ashrae.org/news/ashraejournal/using-indoor-air-quality-tactics-to-sleep-better-at-night-perform-well-the-next-day*](https://www.ashrae.org/news/ashraejournal/using-indoor-air-quality-tactics-to-sleep-better-at-night-perform-well-the-next-day)

[6] [*https://theconversation.com/how-to-use-ventilation-and-air-filtration-to-prevent-the-spread-of-coronavirus-indoors-143732*](https://theconversation.com/how-to-use-ventilation-and-air-filtration-to-prevent-the-spread-of-coronavirus-indoors-143732)

[7][*https://electronics-project-hub.com/how-to-send-data-to-thingspeak-without-sensors/*](https://electronics-project-hub.com/how-to-send-data-to-thingspeak-without-sensors/)

[8][*https://thingspeak.com/pages/commercial\_learn\_more*](https://thingspeak.com/pages/commercial_learn_more)

[9][*https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/Improving-Ventilation-Home.html*](https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/Improving-Ventilation-Home.html)

[10] *https://www.instructables.com/How-to-Control-Fan-Using-ESP8266-From-WebServer/*

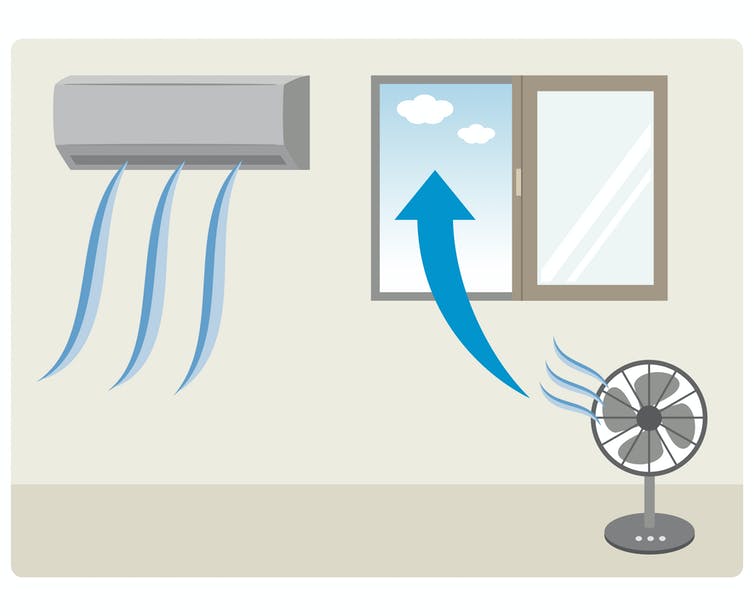
**APPENDIX-A: WHY DO WE NEED TO MONITOR AND CONTROL CO2**

**CASE -1: TO PREVENT THE SPREAD OF CORONVIRUS (OR SIMILAR AIR BORNE VIRUS) INDOORS**

The vast majority of SARS-CoV-2 transmission occurs [indoors](https://doi.org/10.1111/ina.12697), most of it from the [inhalation of airborne particles](https://doi.org/10.1016/j.envint.2020.105730) that contain the coronavirus. The best way to prevent the virus from spreading in a home or business would be to simply keep infected people away. But this is hard to do when an estimated [40% of cases are asymptomatic](https://theconversation.com/can-people-spread-the-coronavirus-if-they-dont-have-symptoms-5-questions-answered-about-asymptomatic-covid-19-140531) and asymptomatic people can [still spread the coronavirus to others](https://dx.doi.org/10.3201%2Feid2607.201595).

[Masks do a decent job](https://dx.doi.org/10.1126/science.abc6197) at keeping the virus from spreading into the environment, but if an infected person is inside a building, inevitably some virus will escape into the air.

Once the virus escapes into the air inside a building, you have two options: bring in fresh air from outside or remove the virus from the air inside the building.



Thankfully, it can be pretty easy to get more outside air into a building. Keeping [windows and doors open](https://doi.org/10.1016/S0960-1481(99)00012-9) is a good start. Putting a box fan in a window blowing out can greatly increase air exchange too. In buildings that don’t have operable windows, you can change the mechanical ventilation system to increase how much air it is pumping. But in any room, the more people inside, the faster the air should be replaced.

**1.1 Using CO2 to measure air circulation**

So how do you know if the room you’re in has enough air exchange? It’s actually a pretty hard number to calculate. But there’s an easy-to-measure proxy that can help. Every time you exhale, you [release CO2](https://doi.org/10.1111/ina.12383) into the air. Since the coronavirus is most often spread by breathing, coughing or talking, you can use [CO2 levels](https://pdfs.semanticscholar.org/dd7e/b2870c38f70e5285e5118ed6f158c091f7cf.pdf) to see if the room is filling up with potentially infectious exhalations. The CO2 level lets you estimate if enough fresh outside air is getting in.

Outdoors, CO2 levels are just above 400 parts per million (ppm). A well-ventilated room will have around [800 ppm of CO2](https://doi.org/10.1111/j.1600-0668.1999.00003.x). Any higher than that and it is a sign the room might need more ventilation.

Last year, researchers in Taiwan reported on the [effect of ventilation on a tuberculosis outbreak](https://doi.org/10.1111/ina.12639) at Taipei University. Many of the rooms in the school were under ventilated and had CO2 levels above 3,000 ppm. When engineers improved air circulation and got CO2 levels under 600 ppm, [the outbreak completely stopped](https://doi.org/10.1111/ina.12639). According to the research, the increase in ventilation was responsible for 97% of the decrease in transmission.

Since the coronavirus is spread through the air, higher CO2 levels in a room likely mean there is a [higher chance of transmission](https://doi.org/10.1111/ina.12639) if an infected person is inside. Based on the study above, I recommend trying to keep the CO2 levels below 600 ppm.

**CASE-2: BETTER SLEEP**

People tend to fall asleep more quickly in bedrooms with comfortable temperatures and good air quality. These conditions also improve sleep quality, according to Pawel Wargocki, Ph.D., Associate Member ASHRAE.

If a bedroom’s window and door are closed for privacy and energy conservation purposes, the room’s ventilation rate is “so low that carbon dioxide (CO2), an indicator of poor indoor air quality (IAQ), routinely exceeds 2,500 to 3,000 ppm, which is three times the recommended levels,” according to published measurements.

**2.1 INDOOR ENVIRONMENTAL CONDITIONS’ EFFECT ON SLEEP**

Wargocki and his colleagues at the Technical University of Denmark [examined the ventilation rate in a bedroom](http://orbit.dtu.dk/en/publications/the-effects-of-bedroom-air-quality-on-sleep-and-nextday-performance(7188d397-1f96-4423-b241-bff6254078b1).html) to see how it affected the sleep and next-day performance of a group of college students. The research showed that both sleep quality and next-day performance could be improved by the increasing the clean outdoor air supply rate in bedrooms.

In the main experiment, the rooms had two conditions, increased ventilation and low ventilation (as if the windows stayed closed all night), each condition lasting for one week. Each subject experienced both conditions, in balanced order (i.e., 50% meeting poor air quality first).

During the ventilated condition, an inaudible outdoor air supply fan was turned on automatically whenever the CO2 concentration increased to above 900 ppm. The fan was off all night during the “low ventilation” condition.

The students could open the windows during the day, but they had to remain closed during the night.

The CO2 concentration in each room ranged from 1,620 ppm to 3,300 ppm with fan off and from 795 ppm to 935 ppm with fan on, according to the research. The average CO2 concentration in the low ventilation condition was 2,395 ppm, with an average of 835 ppm, in the increased ventilation condition.

**2.2 THE RESULTS**

The students wore wrist-watch-type actigraphs that recorded arm movement to measure how long it took them to fall asleep, how often they woke up during the night, and how long they spent asleep; this information can be used to assess how well they slept.

The following morning, within 10 minutes of waking up, the students completed a self-assessment, online questionnaire asking how well they slept, how long they slept and other relevant questions. The students also performed two online tests that measured their concentration and attention, Wargocki said.

“What we observed was that when the fan was on and air quality improved, as indicated by lower CO2 concentration, the students spent a greater percentage of their time in bed asleep when the fan was on,” he said, adding that those students also tended to perform better on the cognitive tests.

The students reported the air was fresher when they got up in the condition with fan on and that they felt better and more rested. However, they felt that their mouth and skin were drier, according to Wargocki.

Although the results were obtained with students, it is likely that the results can be generalized to the general population as well as other climatic regions, their search noted.

“Given these findings, it is reasonable to hypothesize that next-day performance would be better after sleeping in the conditions that provide better bedroom air quality. It was possible to show that this was the case, and to my knowledge for the first time ever,” Wargocki noted.

Because people spend almost one-third of their lifetime in a bedroom, Wargocki considers that there should be more research on how the bedroom thermal environment and IAQ affect sleep quality. He also said ASHRAE should play an active role in this endeavor.

**CASE-3: EFFECTS OF CO2 ON COGNITIVE AND DECISION MAKING, PERFORMANCE**

Most of us live with high carbon dioxide (CO2) levels in our offices, bedrooms, classrooms, and cars without ever thinking about it. But recent double-blind studies suggest that CO2 exposure can reduce cognitive and decision-making performance dramatically — by 50% or more at common indoor levels.

The Earth’s atmosphere is about 0.04% carbon dioxide (CO2)—**400 parts per million**or**ppm**. But the air we breathe out is 100 times more concentrated in CO2 — around 4% or **40,000 ppm**. Every time we exhale, we make the air around us a little less hospitable. Every place we go, we raise CO2 levels.

And where there are lots of people — in our cities, our classrooms, our offices, our homes— CO2 levels rise. A lot. Several studies have measured CO2 concentrations in the human-dense and enclosed places where we spend most of our lives, and the numbers are striking:

* **City centers** can have outdoor CO2 levels above **500 ppm** due to the “urban CO2 dome” effect [Idso 2001, Jacobson 2010].
* **Offices** often have CO2 levels of **600 ppm** or higher. Only 5% of US offices have average CO2 concentrations above **1000 ppm** [Persily 2008], although one study suggests that a typical meeting room can reach up to **1900 ppm** CO2 during 30- to 90-minute meetings [Fisk 2010].
* **Classrooms** often reach average CO2 levels above **1000 ppm**, as observed across elementary school classrooms in Texas, Michigan, Washington, Idaho, Texas, South Carolina, Sweden, and England [Stafford 2015]. In a 2002 study, 21% of classrooms in Texas had CO2 concentrations over **3000 ppm** [Corsi 2002].
* **Public housing units** in Boston have average CO2 concentrations of **810** **ppm** in conventional apartments and **1200 ppm** in new LEED Platinum apartments (good insulation reduces air flow) [Colton 2014].
* **Passenger aircraft**have an average CO2 concentration of around **1400 ppm**during flight, with peak concentrations up to **4200 ppm** [NRC 2002].
* **Bedrooms** in dorms at the Technical University of Denmark reach CO2 levels of **2400 ppm** without ventilation and **840 ppm** with ventilation [Strøm-Tejsen 2016].
* **Cars** with one occupant reach an estimated steady-state CO2 concentration of **4100 ppm**, with the windows closed and the air recirculating [Satish 2012].

**3.1 CO2 on the Brain**

**High CO2 concentrations reduce our cognitive performance, our health, and our comfort**\*. High CO2 levels during sleep are associated with low comfort based on subjective measures of air freshness, mental state, restedness, and mouth/skin dryness [Strøm-Tejsen 2016]. In schools in Washington and Idaho, elevated CO2 levels are associated with increases in student absences: A **1000 ppm** increase in CO2 concentration leads to a 0.5–0.9% reduction in annual average daily attendance [Shendell 2004].

Even astronauts suffer: On the International Space Station, the odds of a crew member reporting a headache double for every **1300 ppm** increase in CO2 concentration [Law 2014]. And it’s hard to keep CO2 levels low in an airtight box filled with people.

\*Conventionally, indoor CO2 levels below **1000 ppm** are considered to be acceptable. Much higher CO2 levels can have direct health effects: **10,000 ppm** (1% CO2) causes increases in respiratory rate, **50,000 ppm** (5%) causes dizziness and confusion, **100,000 ppm** (10%) causes visual disturbances, tremors, vomiting, disorientation, hypertension, and loss of consciousness, and **250,000 ppm** (25%) can cause death [Lipsett 1994][Rice 2003].

The most compelling studies — are a pair of recent double-blind experiments looking at how CO2 levels affect human cognitive performance.

In both cases, the researchers put — college students in one case, working professionals in the other — in simulated office environments with controlled CO2 levels for several hours at a time. They measured the participants’ cognitive function using a standard, well-validated test of decision-making performance with 9 metrics, including *basic activity level* (number of actions taken), *initiative* (development of new or creative activities), *information usage* (ability to use information effectively), *breadth of approach*(flexibility in approach), and *basic strategy* (number of strategic actions taken). **All of these metrics are probably important for performing well at work — and in life.**

Both studies reached similar conclusions. The first study (Satish *et al.* 2012) found that decision-making performance of college students was massively impaired at high CO2 levels. Averaged across all metrics, **performance was reduced by** **12% at 1000 ppm and by 51% at 2500 ppm**compared to the **600 ppm** control scenario. At **2500 ppm**, the participants’ cognitive function — initially above-average compared to a reference population of 20,000 U.S. adults — dropped to **marginal or dysfunctional levels** on 5 of the 9 metrics\*. Keep in mind that these metrics cover things like the ability to use information, to be creative, to take initiative, and to make strategic decisions.

The second study (Allen *et al.* 2015) found that the decision-making performance of working professionals—managers, engineers, programmers, and designers—was also impaired at high CO2 levels. Averaged across all metrics, **performance was reduced by 15% at 945 ppm and 50% at 1400 ppm** compared to the **550 ppm** control\*. The study reports that a **400 ppm** increase in CO2 concentration is associated with a 21% decrease in cognitive function (averaged across all domains)\*\*, after controlling for volatile organic compound (VOC) levels and differences between participants.

*\*****945 ppm****is the CO2 level expected at the ASHRAE-recommended minimum ventilation rate of 20 cubic feet of outdoor air per minute per person.****1400 ppm****is a higher, but not uncommon indoor CO2 concentration, corresponding to the maximum observed 8-hour time-weighted-average CO2 concentration across many U.S. buildings.*

*\*\*This result is consistent with the findings of the first study, which examined CO2 levels up to****2500 ppm****. It’s not clear why—physiologically—such small increases in ambient CO2 levels impair cognitive performance so dramatically.*

**Increase CO2 by 400 ppm, and you decrease cognitive function by over 20%.**

Remember, CO2 levels in our classrooms, bedrooms, and cars are often above **1000 ppm**, **2000 ppm**, and **4000 ppm**, respectively—**600–3600 ppm**above outdoor levels. That means we could be handicapping our brains—as we learn, sleep, and drive—by 50% or more!

**APPENDIX B: HOW TO DECREASE CO2 LEVELS INDOORS?**

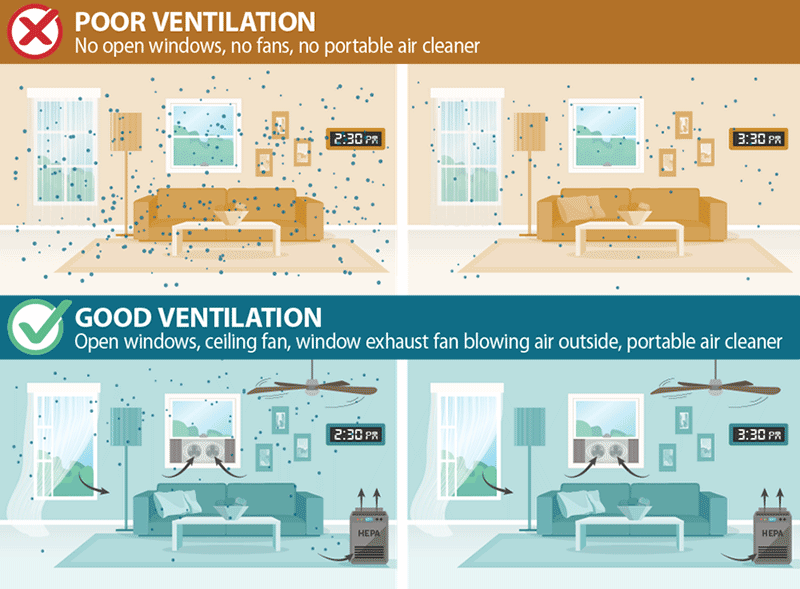
**IMPROVING VENTILATION**

Bring as much fresh air into your home as possible.

Bringing fresh, outdoor air into your home helps keep virus particles from accumulating inside.

* If it’s safe to do so, **open doors and windows** as much as you can to bring in fresh, outdoor air. While it’s better to open them wide, even having a window cracked open slightly can help.
* If you can, open**multiple**doors and windows to allow more fresh air to move inside.
* Do not open windows and doors if doing so is unsafe for you or others (for example, presence of young children and pets, risk of falling, triggering asthma symptoms, high levels of outdoor pollution).

**Turn on the exhaust fan in your bathroom and kitchen.**



With good ventilation, the concentration of virus particles in the air will be lower and they will leave your home faster than with poor ventilation.

Exhaust fans above your stovetop and in your bathroom that vent outdoors **can help move air outside**.  Although some stove exhaust fans don’t send the air to the outside, they can still improve air flow and keep virus particles from being concentrated in one place.

* Keep the exhaust fan turned on over your stovetop and in your bathroom if you have visitors in your home.
* Keep the exhaust fans turned on for an hour after your visitors leave to help remove virus particles that might be in the air.

**Use fans to improve air flow**.

* Place a **fan as close as possible to** an open window blowing outside. This helps get rid of virus particles in your home by blowing air outside. Even without an open window, fans can improve air flow.
* **Point fans away from people**. Pointing fans toward people can possibly cause contaminated air to flow directly at them.
* Use **ceiling fans**to help improve air flow in the home whether or not windows are open.

**APPENDIX C: ARDUINO PROGRAMMING FOR BEGINNERS**

Arduino programs are written in the Arduino Integrated Development Environment (IDE). Arduino IDE is a special software running on your system that allows you to write sketches (synonym for program in Arduino language) for different Arduino boards. The Arduino programming language is based on a very simple hardware programming language called processing, which is similar to the C language. After the sketch is written in the Arduino IDE, it should be uploaded on the Arduino board for execution.

The first step in programming the Arduino board is downloading and installing the Arduino IDE. The open source Arduino IDE runs on Windows, Mac OS X, and Linux. Download the Arduino software (depending on your OS) from the official website and follow the instructions to install.

Now let’s discuss the basics of Arduino programming.

The structure of Arduino program is pretty simple. Arduino programs have a minimum of 2 blocks,

Preparation & Execution

Each block has a set of statements enclosed in curly braces:

void setup( )

{

statements-1;

.

.

.

statement-n;

}

void loop ( )

{

statement-1;

.

.

.

statement-n;

}

Here, setup ( ) is the preparation block and loop ( ) is an execution block.

The setup function is the first to execute when the program is executed, and this function is called only once. The setup function is used to initialize the pin modes and start serial communication. This function has to be included even if there are no statements to execute.

void setup ( )

{

pinMode (pin-number, OUTPUT); // set the ‘pin-number’ as output

pinMode (pin-number, INPUT); // set the ‘pin-number’ as output

}

After the setup ( ) function is executed, the execution block runs next. The execution block hosts statements like reading inputs, triggering outputs, checking conditions etc..

In the above example loop ( ) function is a part of execution block. As the name suggests, the loop( ) function executes the set of statements (enclosed in curly braces) repeatedly.

Void loop ( )

{

digitalWrite (pin-number,HIGH); // turns ON the component connected to ‘pin-number’

delay (1000); // wait for 1 sec

digitalWrite (pin-number, LOW); // turns OFF the component connected to ‘pin-number’

delay (1000); //wait for 1sec

}

Note: Arduino always measures the time duration in millisecond. Therefore, whenever you mention the delay, keep it in milli seconds.

**APPENDIX D: INTERNET OF THINGS**

The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers ([UIDs](https://internetofthingsagenda.techtarget.com/definition/unique-identifier-UID)) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.

A [*thing*](https://internetofthingsagenda.techtarget.com/definition/thing-in-the-Internet-of-Things) in the internet of things can be a person with a heart monitor implant, a farm animal with a [biochip transponder](https://internetofthingsagenda.techtarget.com/definition/injectable-ID-chip-biochip-transponder), an automobile that has built-in [sensors](https://whatis.techtarget.com/definition/sensor) to alert the driver when tire pressure is low or any other natural or man-made object that can be assigned an Internet Protocol (IP) address and is able to transfer data over a network.

Increasingly, organizations in [a variety of industries are using IoT to operate more efficiently](https://searchcustomerexperience.techtarget.com/news/450402550/IoT-technologies-bring-efficiency-and-customization-to-manufacturing), better understand customers to deliver enhanced customer service, improve decision-making and increase the value of the business.

**How IoT works**

An IoT ecosystem consists of web-enabled smart devices that use embedded systems, such as processors, sensors and communication hardware, to collect, send and act on data they acquire from their environments. [IoT devices](https://internetofthingsagenda.techtarget.com/definition/IoT-device%20) share the sensor data they collect by connecting to an [IoT gateway](https://whatis.techtarget.com/definition/IoT-gateway) or other edge device where data is either sent to the cloud to be analyzed or analyzed locally. Sometimes, these devices communicate with other related devices and act on the information they get from one another. The devices do most of the work without human intervention, although people can interact with the devices -- for instance, to set them up, give them instructions or access the data.

The connectivity, networking and communication protocols used with these web-enabled devices largely depend on the specific IoT applications deployed.

IoT can also make use of artificial intelligence (AI) and machine learning to aid in making data collecting processes easier and more dynamic.

**Why IoT is important**

The internet of things helps people live and work smarter, as well as gain complete control over their lives. In addition to offering smart devices to automate homes, IoT is essential to business. IoT provides businesses with a real-time look into how their systems really work, delivering insights into everything from the performance of machines to supply chain and logistics operations.

IoT enables companies to automate processes and reduce labor costs. It also cuts down on waste and improves service delivery, making it less expensive to manufacture and deliver goods, as well as offering transparency into customer transactions.

As such, IoT is one of the most important technologies of everyday life, and it will continue to pick up steam as more businesses realize the potential of connected devices to keep them competitive.

**IoT benefits to organizations**

The internet of things offers several benefits to organizations. Some benefits are industry-specific, and some are applicable across multiple industries. Some of the common benefits of IoT enable businesses to:

* monitor their overall business processes;
* improve the customer experience (CX);
* save time and money;
* enhance employee productivity;
* integrate and adapt business models;
* make better business decisions; and
* generate more revenue.

IoT encourages companies to rethink the ways they approach their businesses and gives them the tools to improve their business strategies.

Generally, IoT is most abundant in manufacturing, transportation and utility organizations, making use of sensors and other IoT devices; however, it has also found use cases for organizations within the agriculture, infrastructure and home automation industries, leading some organizations toward [digital transformation](https://searchcio.techtarget.com/definition/digital-transformation).

IoT can benefit farmers in agriculture by making their job easier. Sensors can collect data on rainfall, humidity, temperature and soil content, as well as other factors, that would help automate farming techniques.

The ability to monitor operations surrounding infrastructure is also a factor that IoT can help with. Sensors, for example, could be used to monitor events or changes within structural buildings, bridges and other infrastructure. This brings benefits with it, such as cost saving, saved time, quality-of-life workflow changes and paperless workflow.

A home automation business can utilize IoT to monitor and manipulate mechanical and electrical systems in a building. On a broader scale, [smart cities](https://internetofthingsagenda.techtarget.com/definition/smart-city) can help citizens reduce waste and energy consumption.

IoT touches every industry, [including businesses within healthcare](https://internetofthingsagenda.techtarget.com/feature/Healthcare-IoT-security-issues-Risks-and-what-to-do-about-them), finance, retail and manufacturing.

**Pros and cons of IoT**

Some of the advantages of IoT include the following:

* ability to access information from anywhere at any time on any device;
* improved communication between connected electronic devices;
* transferring data packets over a connected network saving time and money; and
* automating tasks helping to improve the quality of a business's services and reducing the need for human intervention.

**Some disadvantages of IoT include the following:**

* As the number of connected devices increases and more information is shared between devices, the potential that a hacker could steal confidential information also increases.
* Enterprises may eventually have to deal with massive numbers -- maybe even millions -- of IoT devices, and collecting and managing the data from all those devices will be challenging.
* If there's a bug in the system, it's likely that every connected device will become corrupted.
* Since there's no international standard of compatibility for IoT, it's difficult for devices from different manufacturers to communicate with each other.

**IoT standards and frameworks**

There are several emerging IoT standards, including the following:

* [IPv6](https://searchnetworking.techtarget.com/definition/IPv6-Internet-Protocol-Version-6) over Low-Power Wireless Personal Area Networks (6LoWPAN) is an open standard defined by the Internet Engineering Task Force ([IETF](https://whatis.techtarget.com/definition/IETF-Internet-Engineering-Task-Force)). The 6LoWPAN standard enables any low-power radio to communicate to the internet, including 804.15.4, Bluetooth Low Energy ([BLE](https://internetofthingsagenda.techtarget.com/definition/Bluetooth-Low-Energy-Bluetooth-LE)) and [Z-Wave](https://internetofthingsagenda.techtarget.com/definition/Z-Wave) (for home automation).
* [ZigBee](https://internetofthingsagenda.techtarget.com/definition/ZigBee) is a low-power, low-data rate wireless network used mainly in industrial settings. ZigBee is based on the Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 standard. The ZigBee Alliance created Dotdot, the universal language for IoT that enables smart objects to work securely on any network and understand each other.
* [LiteOS](https://whatis.techtarget.com/definition/LiteOS) is a Unix-like operating system (OS) for wireless sensor networks. LiteOS supports smartphones, [wearables](https://searchmobilecomputing.techtarget.com/definition/wearable-technology), intelligent manufacturing applications, [smart homes](https://internetofthingsagenda.techtarget.com/definition/smart-home-or-building) and the internet of vehicles (IoV). The OS also serves as a smart device development platform.
* OneM2M is a machine-to-machine service layer that can be embedded in software and hardware to connect devices. The global standardization body, OneM2M, was created to develop reusable standards to enable IoT applications across different verticals to communicate.
* Data Distribution Service (DDS) was developed by the Object Management Group (OMG) and is an IoT standard for real-time, scalable and high-performance [M2M](https://internetofthingsagenda.techtarget.com/definition/machine-to-machine-M2M) communication.
* Advanced Message Queuing Protocol ([AMQP](https://whatis.techtarget.com/definition/Advanced-Message-Queuing-Protocol-AMQP)) is an open source published standard for asynchronous messaging by wire. AMQP enables encrypted and interoperable messaging between organizations and applications. The protocol is used in [client-server](https://searchnetworking.techtarget.com/definition/client-server) messaging and in IoT device management.
* Constrained Application Protocol ([CoAP](https://whatis.techtarget.com/definition/Constrained-Application-Protocol)) is a protocol designed by the IETF that specifies how low-power, compute-constrained devices can operate in the internet of things.
* Long Range Wide Area Network (LoRaWAN) is a protocol for WANs designed to support huge networks, such as smart cities, with millions of low-power devices.

**IoT frameworks include the following:**

* [Amazon Web Services (AWS) IoT](https://searchaws.techtarget.com/definition/AWS-IoT-Amazon-Web-Services-internet-of-things) is a cloud computing platform for IoT released by Amazon. This framework is designed to enable smart devices to easily connect and securely interact with the AWS cloud and other connected devices.
* Arm Mbed IoT is a platform to develop apps for IoT based on [Arm microcontrollers](https://whatis.techtarget.com/definition/ARM-processor). The goal of the Arm Mbed IoT platform is to provide a scalable, connected and secure environment for IoT devices by integrating Mbed tools and services.
* Microsoft's [Azure](https://searchcloudcomputing.techtarget.com/definition/Windows-Azure) IoT Suite is a platform that consists of a set of services that enables users to interact with and receive data from their IoT devices, as well as perform various operations over data, such as multidimensional analysis, transformation and aggregation, and visualize those operations in a way that's suitable for business.
* Google's Brillo/Weave is a platform for the rapid implementation of IoT applications. The platform consists of two main backbones: Brillo, an Android-based OS for the development of embedded low-power devices, and Weave, an IoT-oriented communication protocol that serves as the communication language between the device and the cloud.
* Calvin is an open source IoT platform released by Ericsson designed for building and managing distributed applications that enable devices to talk to each other. Calvin includes a development framework for application developers, as well as a runtime environment for handling the running application.

**Consumer and enterprise IoT applications**

There are numerous real-world applications of the internet of things, ranging from consumer IoT and enterprise IoT to manufacturing and industrial IoT ([IIoT](https://internetofthingsagenda.techtarget.com/definition/Industrial-Internet-of-Things-IIoT)). IoT applications span numerous verticals, including automotive, telecom and energy.

In the consumer segment, for example, smart homes that are equipped with smart thermostats, smart appliances and connected heating, lighting and electronic devices can be controlled remotely via computers and smartphones.

Wearable devices with sensors and software can collect and analyze user data, sending messages to other technologies about the users with the aim of making users' lives easier and more comfortable. Wearable devices are also used for public safety -- for example, improving first responders' response times during emergencies by providing optimized routes to a location or by tracking construction workers' or firefighters' vital signs at life-threatening sites.

In healthcare, IoT offers many benefits, including the ability to monitor patients more closely using an analysis of the data that's generated. Hospitals often use IoT systems to complete tasks such as inventory management for both pharmaceuticals and medical instruments.

Smart buildings can, for instance, reduce energy costs using sensors that detect how many occupants are in a room. The temperature can adjust automatically -- for example, turning the air conditioner on if sensors detect a conference room is full or turning the heat down if everyone in the office has gone home.

In agriculture, IoT-based [smart farming](https://internetofthingsagenda.techtarget.com/definition/smart-farming) systems can help monitor, for instance, light, temperature, humidity and soil moisture of crop fields using connected sensors. IoT is also instrumental in automating irrigation systems.

In a smart city, IoT sensors and deployments, such as smart streetlights and smart meters, can help alleviate traffic, conserve energy, monitor and address environmental concerns, and improve sanitation.

**IoT security and privacy issues**

The internet of things connects billions of devices to the internet and involves the use of billions of data points, all of which need to be secured. Due to its expanded attack surface, [IoT security](https://internetofthingsagenda.techtarget.com/definition/IoT-security-Internet-of-Things-security) and [IoT privacy](https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-privacy-IoT-privacy) are cited as major concerns.

In 2016, one of the most notorious recent IoT attacks was Mirai, a [botnet](https://searchsecurity.techtarget.com/definition/botnet) that infiltrated domain name server provider Dyn and took down many websites for an extended period of time in one of the biggest distributed denial-of-service ([DDoS](https://searchsecurity.techtarget.com/definition/distributed-denial-of-service-attack)) attacks ever seen. Attackers gained access to the network by exploiting poorly secured IoT devices.

Because IoT devices are closely connected, all a hacker has to do is exploit one vulnerability to manipulate all the data, rendering it unusable. Manufacturers that don't [update their devices regularly](https://internetofthingsagenda.techtarget.com/feature/How-to-create-a-workflow-for-IoT-software-updates) -- or at all -- leave them vulnerable to cybercriminals.

Additionally, connected devices often ask users to input their personal information, including names, ages, addresses, phone numbers and even social media accounts -- information that's invaluable to hackers.

Hackers aren't the only [threat to the internet of things](https://internetofthingsagenda.techtarget.com/tip/5-IoT-security-threats-to-prioritize); privacy is another major concern for IoT users. For instance, companies that make and distribute consumer IoT devices could use those devices to obtain and sell users' personal data.

Beyond leaking personal data, [IoT poses a risk](https://internetofthingsagenda.techtarget.com/tip/Internet-of-Things-IOT-Seven-enterprise-risks-to-consider) to critical infrastructure, including electricity, transportation and financial services.

**History of IoT**

Kevin Ashton, co-founder of the Auto-ID Center at the Massachusetts Institute of Technology (MIT), first mentioned the internet of things in a presentation he made to Procter & Gamble (P&G) in 1999. Wanting to bring radio frequency ID (RFID) to the attention of P&G's senior management, Ashton called his presentation "Internet of Things" to incorporate the cool new trend of 1999: the internet. MIT professor Neil Gershenfeld's book, When Things Start to Think, also appeared in 1999. It didn't use the exact term but provided a clear vision of where IoT was headed.

IoT has evolved from the convergence of wireless technologies, microelectromechanical systems ([MEMSes](https://internetofthingsagenda.techtarget.com/definition/micro-electromechanical-systems-MEMS)), [microservices](https://searchapparchitecture.techtarget.com/definition/microservices) and the internet. The convergence has helped tear down the silos between operational technology ([OT](https://whatis.techtarget.com/definition/operational-technology)) and information technology (IT), enabling unstructured machine-generated data to be analyzed for insights to drive improvements.

Although Ashton's was the first mention of the internet of things, the idea of connected devices has been around since the 1970s, under the monikers embedded internet and [pervasive computing](https://internetofthingsagenda.techtarget.com/definition/pervasive-computing-ubiquitous-computing).

The first internet appliance, for example, was a Coke machine at Carnegie Mellon University in the early 1980s. Using the web, programmers could check the status of the machine and determine whether there would be a cold drink awaiting them, should they decide to make the trip to the machine.

IoT evolved from M2M communication, i.e., machines connecting to each other via a network without human interaction. M2M refers to connecting a device to the cloud, managing it and collecting data.

Taking M2M to the next level, IoT is a sensor network of billions of smart devices that connect people, systems and other applications to collect and share data. As its foundation, M2M offers the connectivity that enables IoT.

The internet of things is also a natural extension of supervisory control and data acquisition ([SCADA](https://whatis.techtarget.com/definition/SCADA-supervisory-control-and-data-acquisition)), a category of software application programs for process control, the gathering of data in real time from remote locations to control equipment and conditions. SCADA systems include hardware and software components. The hardware gathers and feeds data into a computer that has SCADA software installed, where it is then processed and presented in a timely manner. The evolution of SCADA is such that late-generation SCADA systems developed into first-generation IoT systems.

The concept of the IoT ecosystem, however, didn't really come into its own until the middle of 2010 when, in part, the government of China said it would make IoT a strategic priority in its five-year plan.