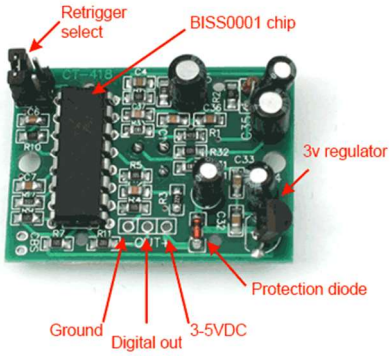
Experiment – 1

AIM: Introduction to various sensors and actuators

a) PIR Motion Sensor.



PIR Motion Sensor HC-SR501 Motion Detector Pyroelectric Infrared Module is a pyroelectric device that detects motion by measuring changes in the infrared levels emitted by surrounding objects. This motion can be detected by checking for a high signal on a single I/O pin. PIR sensor full form stands as Passive Infrared Sensor.

This Motion Detector Sensor Module provides an optimized circuit that will detect motion up to 6 meters away and can be used in burglar alarms and access control systems. The PIR sensor and Fresnel lens are fitted onto the PCB. This enables the board to be mounted inside a case with the detecting lens protruding outwards, whilst still allowing easy access to the controls inside the case.

It is used to sense movement of people, animals, or other objects. They are commonly used in burglar alarms and automatically-activated lighting systems and compatible with Arduino and other microcontroller boards.

b) Rain Drop Sensor.

Water Drop Rain sensor can be used to monitor rain or slushy snow/hail and send closure requests to electronic shutters, windows, awnings or skylights whenever the rain is detected. The working of the rain sensor is pretty straightforward. The sensing pad with a series of exposed copper traces together acts as a variable resistor (just like

a potentiometer) whose resistance varies according to the amount of water on its surface.

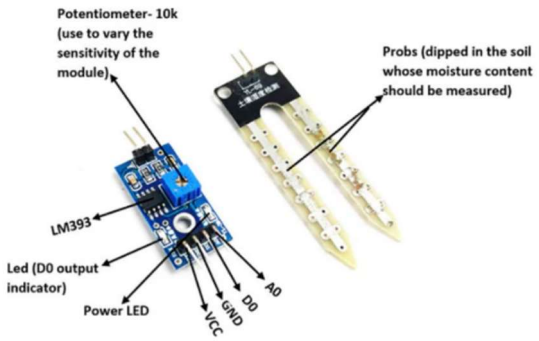
This resistance is inversely proportional to the amount of water:

More water on the surface means better conductivity and will result in a lower resistance. The less water on the surface means poor conductivity and will result in higher resistance.

The sensor produces an output voltage according to the resistance, which by measuring we can determine whether it's raining or not.

The rain sensor output can be digitized using our LM393 Comparator module and then fed to any microcontroller such as Arduino.

c) Soil moisture Sensor.



The soil moisture sensor is commonly used in smart agriculture or other garden automation projects to measure the moisture content present in the soil. It consists of 4 pins in which two pins, Vcc and Gnd are connected to supply voltage. The remaining two pins are digital (D0) and analog (A0) are the output pins. When the moisture content present in the soil goes beyond the threshold level, the output of the digital pin (D0) will go low (the output of the digital pin is either logic 0 or 1). The threshold value of the sensor module can be set by varying the onboard potentiometer. The analog output pin can be used to calculate the approximate level of moisture content present in the soil.

The sensor probe will be dipped in the soil and connected to the measurement module. The measurement module will compare the measured value with the set threshold value (can be set using 10k pot) using the LM393 OP-Amp comparator and provide output on the digital pin.

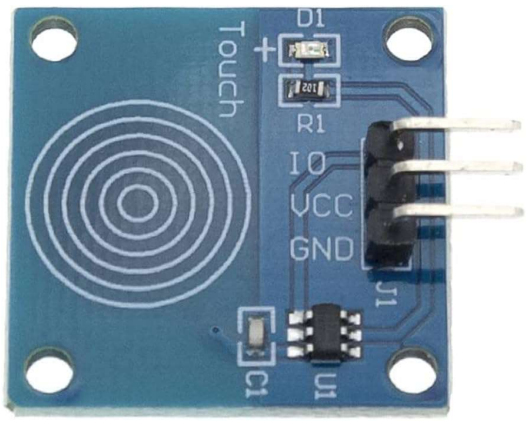
d) Temperature Sensor.

DHT11 is a popular temperature and humidity based digital sensor. There is an upgraded version of the DHT11 temperature and humidity sensor available, which is DHT22 Sensor with higher sensing ranges.

The sensor uses a capacitive humidity sensor and a thermistor based temperature sensor to measure the ambient humidity and temperature. The humidity sensing ranges from 20% to 90% with ±5% accuracy and the temperature sensing ranges from 0 degrees to the 50 degrees Celsius with ±2°C accuracy. The sampling time of this sensor is 2 seconds almost. This Temperature and Humidity Sensor uses digital pins to communicate with the microcontroller unit and does not have any kind of analog pins.

The module also has the inbuilt pull-up resistor and additional filter capacitor to support the DHT11 sensor. Thus the module is available in ready to go mode and can be directly connected with the microcontroller unit without using any kind of additional components.

e) Touch Sensor.



The sensor consists of the TTP223 IC which is based on the capacitive sensing principle. It consists of a sensing electrode and an oscillator circuit.

When a conductive material that is finger comes in contact with the sensing electrode, it changes the capacitance of the electrode.

This change is detected by the oscillator circuit, then the oscillator circuit generates a digital output signal.

This indicates the touch or proximity of the object. It is widely used in various electronic devices such as toys and touch switches.

f) Infrared Sensor.

The sensor module adaptable to ambient light, having a pair of infrared emitting and receiving LEDs, transmitting LED emit infrared certain frequency, when the direction of an obstacle is detected (reflection surface), the infrared reflected is received by the reception LED, After a comparator circuit processing, the obstacle light is on, but the signal output interface output digital signal (a low-level signal), you can adjust the detection distance trim potentiometer, the effective distance range of 2 ~ 20cm, the working voltage of 3.3V- 5V. Detection range of the sensor can be obtained by adjusting potentiometer, with little interference, easy to assemble, easy to use features, can be widely used in robot obstacle avoidance, avoidance car, line count, black and white line tracking and many other applications.

When the module detects an obstacle in front of the signal, the obstacle indicator lights on the board level, while the OUT port sustained low signal output, the module detects the distance 2 ~ 20cm, detection angle 35 °, the distance can detect potential is adjusted clockwise adjustment potentiometer, detects the distance increases; counterclockwise adjustment potentiometer, reducing detection distance.

The sensor active infrared reflection detection, target reflectivity and therefore the shape is critical detection distance. Where the minimum detection distance black, white, maximum; small objects away from a small area, a large area from the Grand, the sensor module output port OUT port can be directly connected to the microcontroller IO can also be directly drive a 5V relay; Connection: VCC-VCC; GND-GND; OUT-IO, comparators LM393, stable, the module can be 3-5V DC power supply. When the power is turned on, the red power indicator lights, with the screw holes 3mm, easy fixed installation, board size: 3.2CM * 1.4CM.

g) RFID Sensor.

Radio-Frequency Identification (RFID) Sensor technology utilizes electromagnetic fields to identify and track tags attached to objects. Unlike barcodes that require line-of-sight scanning, RFID operates wirelessly, allowing for quick and seamless data capture. Each RFID tag contains a unique identifier that can be read by a compatible reader within range.

It is a wireless identification technology that uses radio waves to transfer data from the card tag to an RFID reader and identify the object presence.

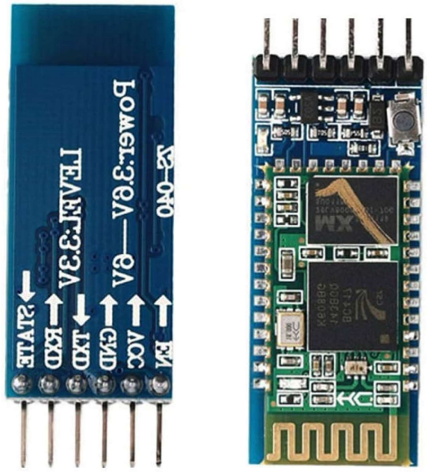
Just like the bar code technology, RFID is used to identify objects, persons, by reading the card tag. This is better than the bar code because the bar code can sometimes be damaged or unreadable.

h) Ultrasonic Sensor.

The Ultrasonic Sensor HCSR04 module is a transmitter, a receiver and a control circuit in one single module. It offers an excellent range of accuracy and stable readings in an easy-to-use package. Also, its operation is not affected by sunlight or black material like Sharp rangefinders are (although acoustically soft materials like cloth can be difficult to detect).

The Trigger and the Echo pins are the I/O pins that can be connected to the I/O pins of the microcontroller/Arduino. It needs 10 us high-level signal at least Module will send eight 40kHz square waves automatically and will test if there is any signal returned. If there is a signal returned, the output will be high-level signals via IO port ECHO. The duration of the high-level signal is the time from the transmitter to receiving the ultrasonic. Testing distance = duration of high-level sound velocity (340m/s) / 2 You can use the above calculation to find the distance between the obstacle and the ultrasonic module.

i) Bluetooth Module.



HC 05 6-Pin Bluetooth Module with TTL Output, To setup Wireless Serial Communication, HC-05 Bluetooth Module is most demanding and popular due to its low price and extremely high features. The module can be used in Master or Slave Mode and easy switchable between these two modes, By default Slave mode is configured.

Modes can be changed using AT Commands. The slave mode in HC-05 cannot initiate a connection to another Bluetooth device, but can accept connections. Master mode can initiate a connection to other devices.

j) Wi-Fi Module.

This is WiFi serial transceiver module, based on ESP8266 SoC., The SOC has Integrated TCP/IP protocol stackESP8266 is a highly integrated chip designed for the needs of a new connected world. It offers a complete and self-contained Wi-Fi networking solution, allowing it to either host the application or to offload all Wi-Fi networking functions from another application processor. ESP8266 has powerful on-board processing and storage capabilities that allow it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, and the entire solution, including front-end module, is designed to occupy minimal PCB area.

k) LED/OLED display

This i2c display module is 0.96 inch diagonal, Resolution: 128 x 64, View angle: > 160°, Support voltage: 3.3V-5V DC, Power consumption: 0.04W during normal operation, full screen lit 0.08W,Color:Yellow Blue

The IIC address can be modified, making it simple to use with various machines. Installing four square holes is simple.

Using a 0.96-inch OLED module, you may display text and graphics on microcontroller projects. It works with the Raspberry Pi, 51 MCU, and STIM 32.

The display unit doesn't need a backlight and can be self-luminous. Even tiny texts are readable because to the Super High Contrast, bright, and clear dots on the screen.The OLED controller doesn't have any inherent typefaces, but users can make their own using font generating software.

l) Servo Motor.

SG90 is a low cost and high output power servo motor. It can not move continuously but rotates up to 180 degrees and each step can be of maximum 90 degrees. Moreover, it is small enough that it can easily fit into your robotics ARM or obstacle avoidance robotics projects. On top of that, it requires only one output pulse signal to control its movement.

Experiment – 2

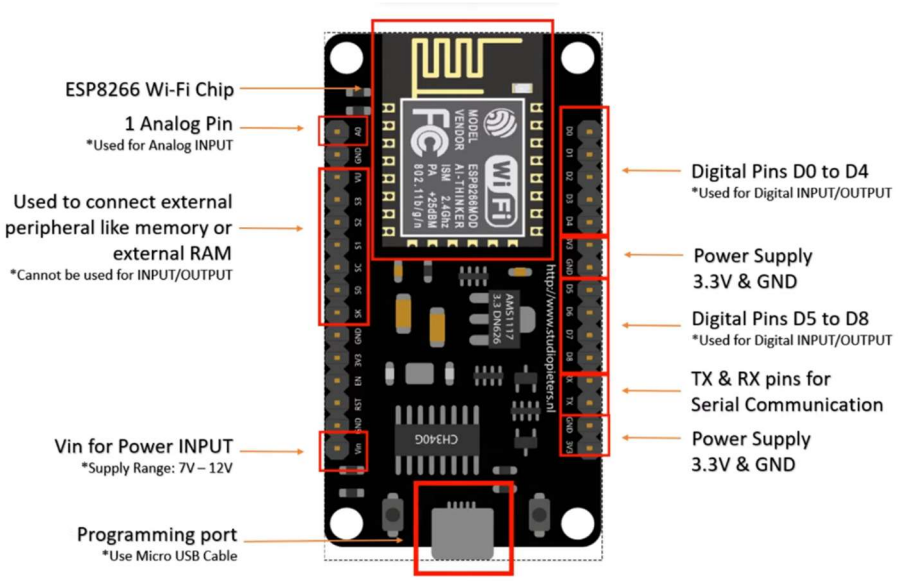
AIM: Acquaintance with NodeMCU and perform essential programming establishment.

NodeMCU is an open source platform based on ESP8266 which can connect objects and let data transfer using the Wi-Fi protocol. In addition, by providing some of the most important features of microcontrollers such as GPIO, PWM, ADC, and etc, it can solve many of the project's needs alone. The general features of this board are as follows:

- Easy to use
- Programmability with Arduino IDE or IUA languages
- Available as an access point or station
- practicable in Event-driven API applications
- Having an internal antenna
- Containing 13 GPIO pins, 10 PWM channels, I2C, SPI, ADC, UART,

NodeMCU is an open-source LUA based firmware developed for the ESP8266 wifi chip. By exploring functionality with the ESP8266 chip, NodeMCU firmware comes with the ESP8266 Development board/kit i.e. NodeMCU Development board. NodeMCU is an open-source platform, its hardware design is open for edit/modify/build. The **ESP8266** is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol.

NodeMCU Development Board/kit

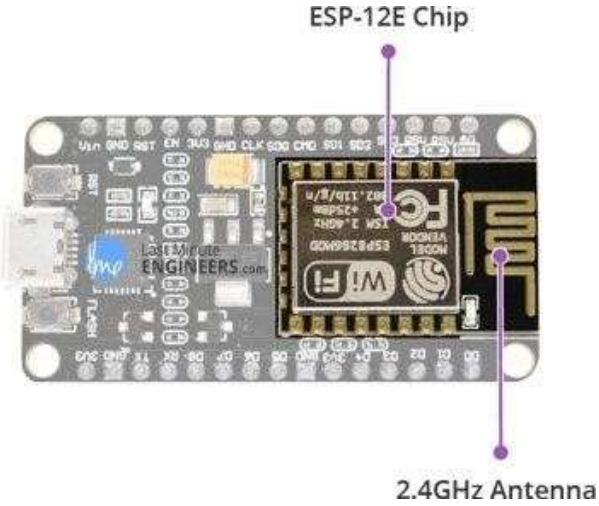


NodeMCU Development board is featured with wifi capability, analog pin, digital pins, and serial communication protocols. To get started with using NodeMCU for IoT applications first we need to know about how to write/download NodeMCU firmware in NodeMCU Development Boards. And before that where this NodeMCU firmware will get as per our requirement.

ESP8266 NodeMCU Features & Using It With Arduino IDE

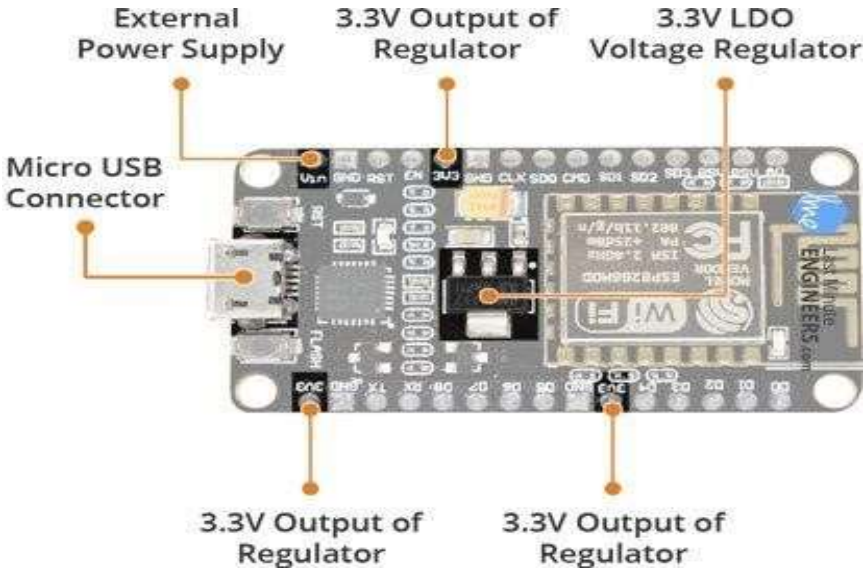
The Internet of Things (IoT) has been a trending field in the world of technology. It has changed the way we work. Physical objects and the digital world are connected now more than ever. Keeping this in mind, Espressif Systems (A Shanghai-based Semiconductor Company) has released an adorable, bite-sized WiFi enabled microcontroller – ESP8266, it can monitor and control things from anywhere in the world – perfect for just about any IoT project.

1. **ESP-12E Module :** The development board equips the ESP-12E module containing ESP8266 chip having Tensilica Xtensa® 32-bit LX106 RISC microprocessor which operates at 80 to 160 MHz adjustable clock frequency and supports RTOS. Tensilica Xtensa® 32-bit LX106 80 to 160 MHz Clock Freq.
    - 128kB internal RAM
    - 4MB external flash
    - 802.11b/g/n Wi-Fi transceiver



There's also 128 KB RAM and 4MB of Flash memory (for program and data storage) just enough to cope with the large strings that make up web pages, JSON/XML data, and everything we throw at IoT devices nowadays. The ESP8266 Integrates 802.11b/g/n HT40 Wi-Fi transceiver, so it can not only connect to a WiFi network and interact with the Internet, but it can also set up a network of its own, allowing other devices to connect directly to it. This makes the ESP8266 NodeMCU even more versatile.

2. **Power Requirement :** As the operating voltage range of ESP8266 is 3V to 3.6V, the board comes with a LDO voltage regulator to keep the voltage steady at 3.3V. It can reliably supply up to 600mA, which should be more than enough when ESP8266 pulls as much as 80mA during RF transmissions. The output of the regulator is also broken out to one of the sides of the board and labeled as 3V3. This pin can be used to supply power to external components.
    - Operating Voltage: 2.5V to 3.6V
    - On-board 3.3V 600mA regulator
    - 80mA Operating Current
    - 20 µA during Sleep Mode

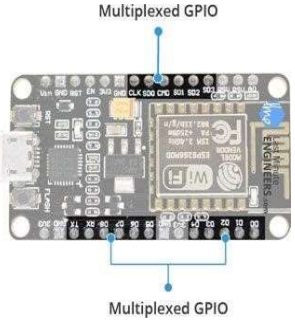


Power to the ESP8266 NodeMCU is supplied via the on-board MicroB USB connector. Alternatively, if you have a regulated 5V voltage source, the VIN pin can be used to directly supply the ESP8266 and its peripherals.

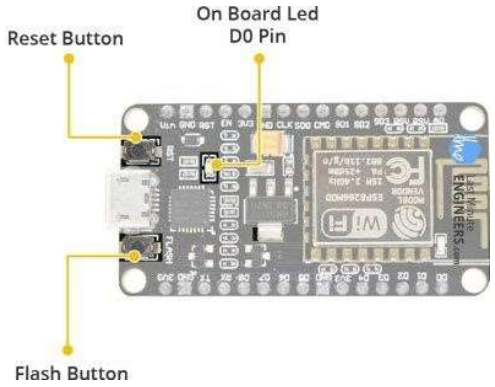
The ESP8266 requires a 3.3V power supply and 3.3V logic levels for communication. The GPIO pins are not 5V-tolerant! If you want to interface the board with 5V (or higher) components, you'll need to do some level shifting.

3. **Peripherals and I/O :** The ESP8266 NodeMCU has total 17 GPIO pins broken out to the pin headers on both sides of the development board. These pins can be assigned to all sorts of peripheral duties, including:
    - ADC channel – A 10-bit ADC channel.
    - UART interface – UART interface is used to load code serially.
    - PWM outputs – PWM pins for dimming LEDs or controlling motors.
    - SPI, I2C & I2S interface – SPI and I2C interface to hook up all sorts of sensors and peripherals.
    - I2S interface – I2S interface if you want to add sound to your project. Multiplexed I/Os
    - 1 ADC channels
    - 2 UART interfaces
    - 4 PWM outputs
    - SPI, I2C & I2S interface

Thanks to the ESP8266's pin multiplexing feature (Multiple peripherals multiplexed on a single GPIO pin). Meaning a single GPIO pin can act as PWM/UART/SPI.

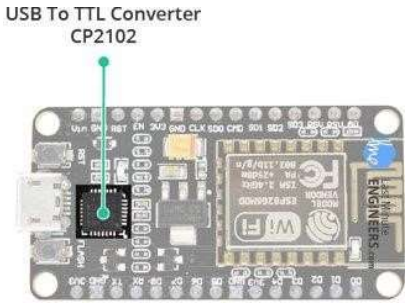


4. **On-board Switches & LED Indicator :** The ESP8266 NodeMCU features two buttons. One marked as RST located on the top left corner is the Reset button, used of course to reset the ESP8266 chip. The other FLASH button on the bottom left corner is the download button used while upgrading firmware.
    - RST – Reset the ESP8266 chip
    - FLASH – Download new programs
    - Blue LED – User Programmable



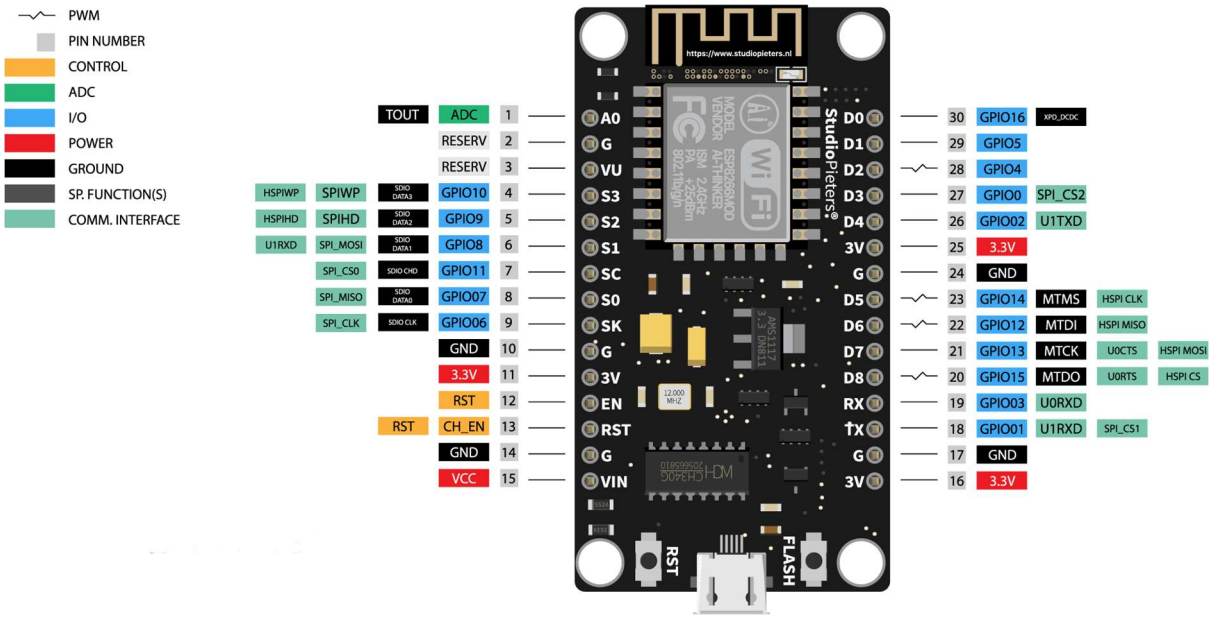
The board also has a LED indicator which is user programmable and is connected to the D0 pin of the board.

5. **Serial Communication :** The board includes CP2102 USB-to-UART Bridge Controller from Silicon Labs, which converts USB signal to serial and allows your computer to program and communicate with the ESP8266 chip.
    - CP2102 USB-to-UART converter
    - 4.5 Mbps communication speed
    - Flow Control support



ESP8266 NodeMCU Pinout

The ESP8266 NodeMCU has total 30 pins that interface it to the outside world. The connections are as follows:



1. Power Pins There are four power pins viz. one VIN pin & three 3.3V pins. The VIN pin can be used to directly supply the ESP8266 and its peripherals, if you have a regulated 5V voltage source. The 3.3V pins are the output of an on-board voltage regulator. These pins can be used to supply power to external components.
2. GND is a ground pin of ESP8266 NodeMCU development board.
3. I2C Pins are used to hook up all sorts of I2C sensors and peripherals in your project. Both I2C Master and I2C Slave are supported. I2C interface functionality can be realized programmatically, and the clock frequency is 100 kHz at a maximum. It should be noted that I2C clock frequency should be higher than the slowest clock frequency of the slave device.
4. GPIO Pins ESP8266 NodeMCU has 17 GPIO pins which can be assigned to various functions such as I2C, I2S, UART, PWM, IR Remote Control, LED Light and Button programmatically. Each digital enabled GPIO can be configured to internal pull-up or pull-down, or set to high impedance. When configured as an input, it can also be set to edge-trigger or level-trigger to generate CPU interrupts.
5. ADC Channel The NodeMCU is embedded with a 10-bit precision SAR ADC. The two functions can be implemented using ADC viz. Testing power supply voltage of VDD3P3 pin and testing input voltage of TOUT pin. However, they cannot be implemented at the same time.
6. UART Pins ESP8266 NodeMCU has 2 UART interfaces, i.e. UART0 and UART1, which provide asynchronous communication (RS232 and RS485), and can communicate at up to 4.5 Mbps. UART0 (TXD0, RXD0, RST0 & CTS0 pins) can be used for communication. It supports fluid control. However, UART1 (TXD1 pin) features only data transmit signal so, it is usually used for printing log.
7. SPI Pins ESP8266 features two SPIs (SPI and HSPI) in slave and master modes.

These SPIs also support the following general-purpose SPI features:

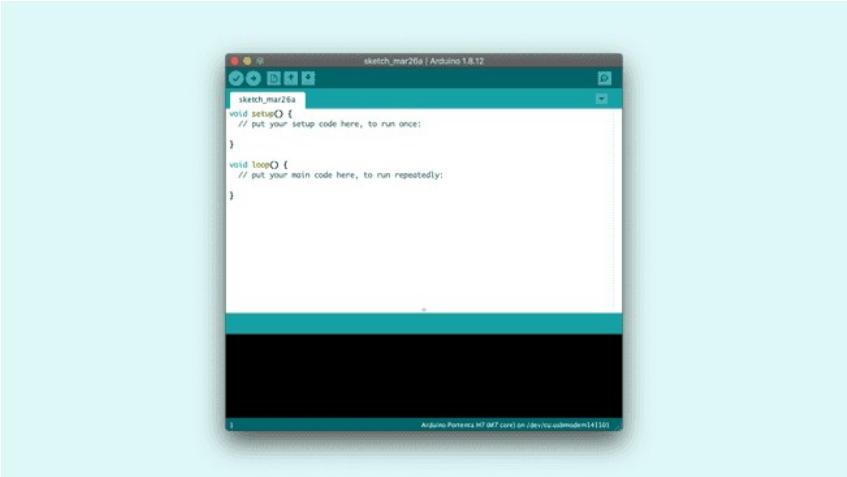
- 4 timing modes of the SPI format transfer
- Up to 80 MHz and the divided clocks of 80 MHz
- Up to 64-Byte FIFO

8. SDIO Pins ESP8266 features Secure Digital Input/Output Interface (SDIO) which is used to directly interface SD cards. 4-bit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0 are supported.
9. PWM Pins The board has 4 channels of Pulse Width Modulation (PWM). The PWM output can be implemented programmatically and used for driving digital motors and LEDs. PWM frequency range is adjustable from 1000 µs to 10000 µs, i.e., between 100 Hz and 1 kHz.
10. Control Pins are used to control ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pin.
    - EN pin – The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
    - RST pin – RST pin is used to reset the ESP8266 chip.

Experiment – 3

AIM: Connect LED/Buzzer with NodeMCU and compose a program to turn ON LED for 1 sec later at regular intervals

Setting up ESP32 in the Arduino IDE on Windows



1. Download Arduino ide from Arduino site and install
2. Install the drivers USB chip, download and Install nodemcu Drivers for window
3. restart system

How to setup Nodemcu(esp8266) board

1. Open Arduino IDE, go to file--> preference--> additional board manager URL:paste json link here
2. paste the link inside the URL
   “https://arduino.esp8266.com/stable/package_esp8266com_index.json”

Adding Nodemcu board to Arduino IDE

1. Open Arduino IDE, go to tools--> board--> board manger--> search:"esp8266" by esp8266 community OR "nodemcu" --> install
2. Selecting Nodemcu board
   Open Arduino IDE, go to tools--> boards--> NodeMCU 1.0 (ESP 12E Module)
3. Open and run hello world program
   Open Arduino IDE, go to file--> example--> blink

```
// The setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  // turn the LED on (HIGH is the voltage level)
  digitalWrite(LED_BUILTIN, HIGH);
  // wait for a second
  delay(1000);

  // turn the LED off by making the voltage LOW
  digitalWrite(LED_BUILTIN, LOW);
  // wait for a second
     delay(1000);
}
```

4. go to menu Tools--> Port--> COM(Arduino)
5. click on ✅ to compile, then ➡️ to upload sketch on nodemcu

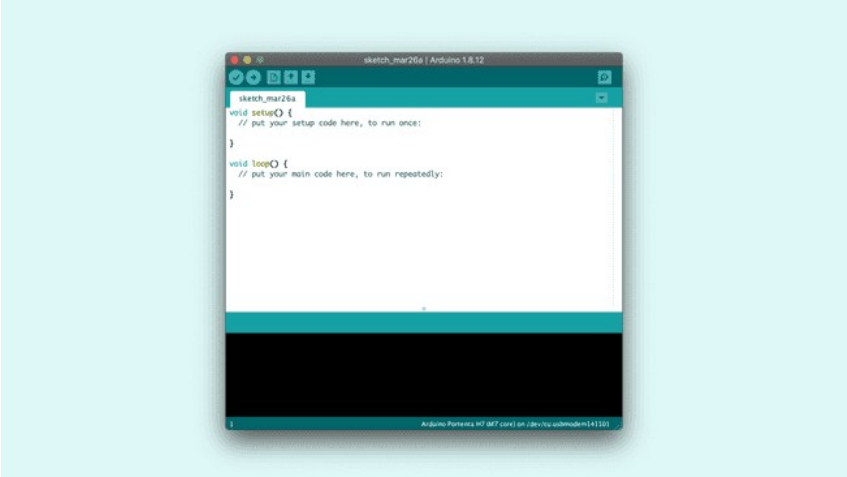
Experiment – 4

AIM: Perform Experiment to use NodeMCU ESP8266 as HTTP Server using WiFi Access Point (AP) mode.

The main difference between NodeMCU station mode and Wi-Fi access point mode is that in station mode, the NodeMCU joins an existing network, while in access point mode, it creates its own network

- Nodemcu is creating it's on Wi-Fi local area network
- Its working as a hotspot

Setting up ESP32 in the Arduino IDE on Windows



1.Download Arduino ide from Arduino site and install

2. Install the drivers USB chip, download and Install nodemcu Drivers for window

3. Restart system

How to setup Nodemcu(esp8266) board

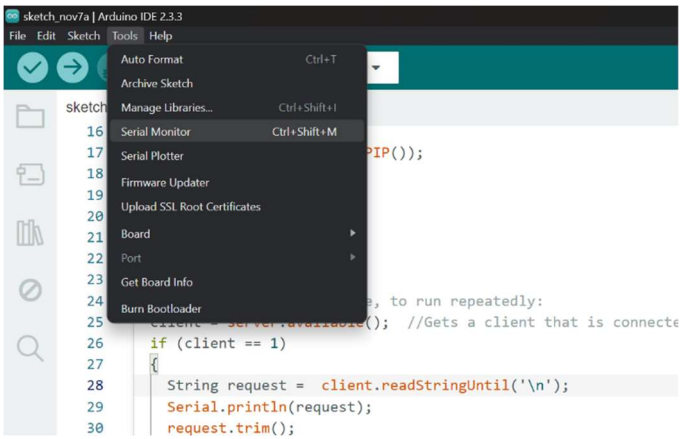
1. Open Arduino IDE, go to file--> preference--> additional board manager URL:paste json link here
2. paste the link inside the URL "https://arduino.esp8266.com/stable/package_esp8266com_index.json"

Adding Nodemcu board to Arduino IDE

1. Open Arduino IDE, go to tools--> board--> board manger--> search:"esp8266" by esp8266 community OR "nodemcu" --> install
2. Selecting Nodemcu board
   Open Arduino IDE, go to tools--> boards--> NodeMCU 1.0 (ESP 12E Module)
3. Write and run program

```
#include <ESP8266WiFi.h>

WiFiClient client;
WiFiServer server(80);

#define led D5

void setup()
{
  // put your setup code here, to run once:
  Serial.begin(9600);
  WiFi.softAP("WIFI_name_change", "password");
  Serial.println();
  Serial.println("NodeMCU Started!");

  // generate nodemcu ip
  Serial.println(WiFi.softAPIP());
  server.begin();
  pinMode(led, OUTPUT);
}

void loop()
{
  // put your main code here, to run repeatedly:
  client = server.available();  //Gets a client that is connected to the
server and has data available for reading.
  if (client == 1)
  {
    String request =  client.readStringUntil('\n');
    Serial.println(request);
    request.trim();
    if(request == "GET /ledon HTTP/1.1")
    {
      digitalWrite(led, HIGH);
    }
    if(request == "GET /ledoff HTTP/1.1")
    {
      digitalWrite(led, LOW);
    }
  }
}
```

4. go to menu Tools--> Port--> COM(Arduino)
5. click on ✅ to compile, then ➡️ to upload sketch on nodemcu
6. Get the ip address from serial monitor, paste it on browser where mobile/pc is connected to Nodemcu station



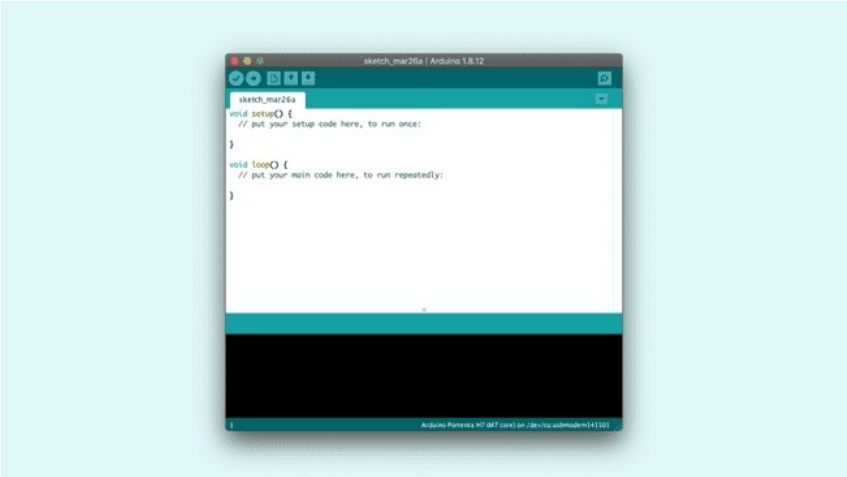
7. On phone/pc after Ip address type /on or /off to switch on/off LED
8. On phone/pc to make led turn on: “198.164.4.1/on” in URL
9. On phone/pc to make led turn off: “198.164.4.1/off” in URL

Experiment – 5

AIM: Perform Experiment for Controlling LED through an HTTP page Using NodeMCU Station Mode (STA)

The main difference between NodeMCU station mode and Wi-Fi access point mode is that in station mode, the NodeMCU joins an existing network, while in access point mode, it creates its own network

Setting up ESP32 in the Arduino IDE on Windows



1.Download Arduino ide from Arduino site and install

2. Install the drivers USB chip, download and Install nodemcu Drivers for window

3. Restart system

How to setup Nodemcu(esp8266) board

1. Open Arduino IDE, go to file--> preference--> additional board manager URL:paste json link here
2. paste the link inside the URL
   “https://arduino.esp8266.com/stable/package_esp8266com_index.json”

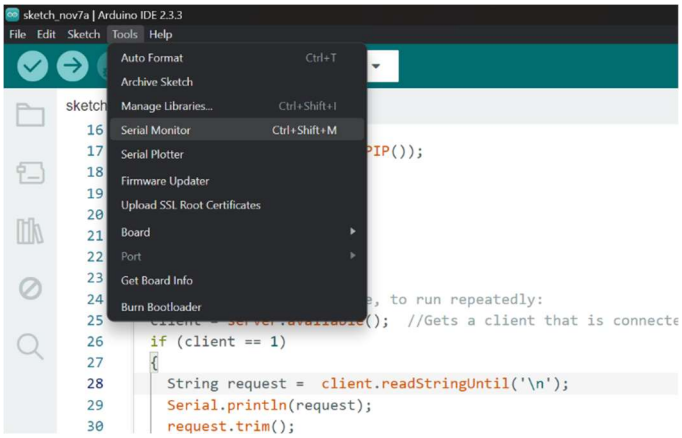
Adding Nodemcu board to Arduino IDE

1. Open Arduino IDE, go to tools--> board--> board manger--> search:"esp8266" by esp8266 community OR "nodemcu" --> install
2. Selecting Nodemcu board
   Open Arduino IDE, go to tools--> boards--> NodeMCU 1.0 (ESP 12E Module)
3. Write and run program

```
#include <ESP8266WiFi.h>

WiFiClient client;
WiFiServer server(80);

#define led D5

void setup()
{
  Serial.begin(9600);
  WiFi.softAP("AIML", "@123@123");
  Serial.println("NodeMCU Started!");
  Serial.println(WiFi.softAPIP());
  server.begin();
  pinMode(led, OUTPUT);
}

void loop()
{
  // put your main code here, to run repeatedly:
  client = server.available(); //Gets a client that is connected to the
server and has data for reading.
  if (client == 1)
  {
    String request = client.readStringUntil('\n');
    Serial.println(request);
    request.trim();

  if(request == "GET /on HTTP/1.1")
  {
    digitalWrite(led, HIGH);
  }

  if(request == "GET /off HTTP/1.1")
  {
    digitalWrite(led, LOW);
  }
}

client.println("<html>");
client.println("<body>");
client.println("<h1>Welcome to the Webpage!</h1>");
client.println("<h3>LED Controls<h3>");
client.println("<br>");
client.println("<a href=\"/on\"\"><button>LED 1 OFF</button></a>");
client.println("<a href=\"/off\"\"><button>LED 1 ON</button></a><br/>");
client.println("</body>");
client.println("</html>");
```

```
}
```

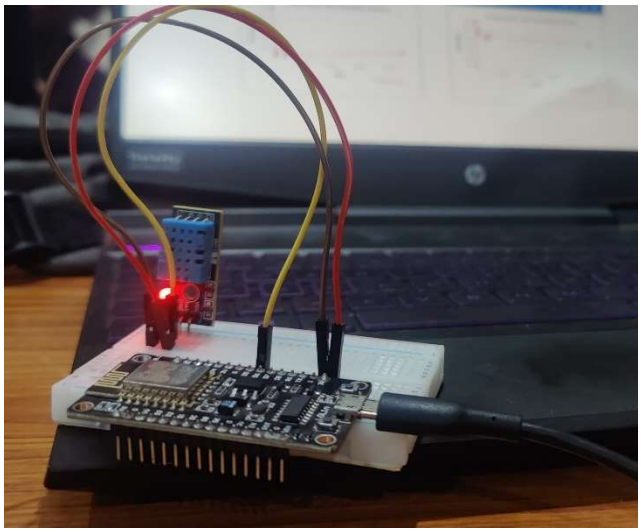
4. go to menu Tools--> Port--> COM(Arduino)
5. click on ✅ to compile, then ➡️ to upload sketch on nodemcu
6. Get the ip address from serial monitor, paste it on browser where mobile/pc is connected to Nodemcu station



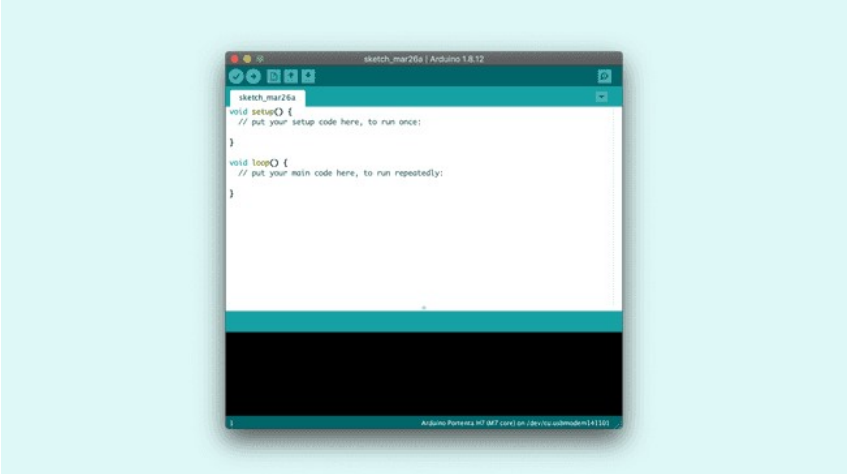
7. On phone/pc type Ip address to switch on/off LED

Experiment – 6

AIM: Interact with DHT11 sensor with NodeMCU and compose a program to print temperature and humidity readings on screen



Setting up ESP32 in the Arduino IDE on Windows



1. Download Arduino ide from Arduino site and install
2. Install the drivers USB chip, download and Install nodemcu Drivers for window
3. Restart system

How to setup Nodemcu(esp8266) board

1. Open Arduino IDE, go to file--> preference--> additional board manager URL:paste json link here
2. paste the link inside the URL "https://arduino.esp8266.com/stable/package_esp8266com_index.json"

How to install library

Open arduino IDE and follow steps

- Go to sketch--> include library--> manage library--> Search: "DHT11" by adafruit --> install all
-

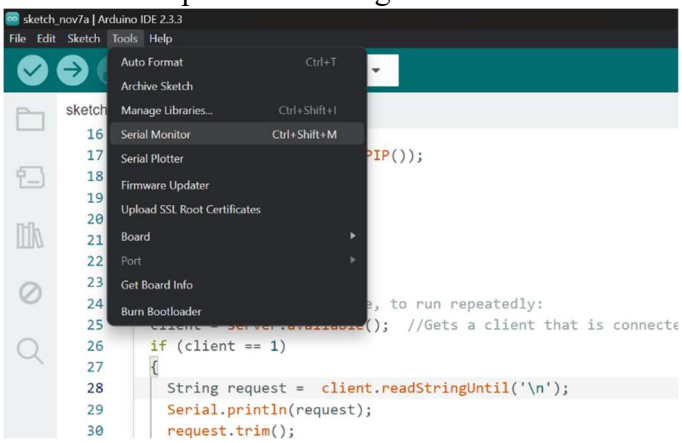
Adding Nodemcu board to Arduino IDE

1. Open Arduino IDE, go to tools--> board--> board manger--> search:"esp8266" by esp8266 community OR "nodemcu" --> install
2. Selecting Nodemcu board
   Open Arduino IDE, go to tools--> boards--> NodeMCU 1.0 (ESP 12E Module)
3. Write and run program

```
#include "DHT.h"
#define DHTPIN 2 // what digital pin we're connected to
#define DHTTYPE DHT11 // DHT 11

DHT dht(DHTPIN, DHTTYPE);

void setup()
{
  Serial.begin(9600);
  Serial.println("Room Temperature and Humidity");
  dht.begin();
}

void loop()
{
  delay(2000);
  float h = dht.readHumidity();
  float t = dht.readTemperature(); // Read temperature as Celsius (the
default)
  float f = dht.readTemperature(true); // Read temperature as Fahrenheit
(isFahrenheit = true)

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f))
  {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.println(" %");
  Serial.print("Temperature: ");
  Serial.print(t);
```

```
  Serial.print(" *C ");
  Serial.print(f);
  Serial.println(" *F");
}
```

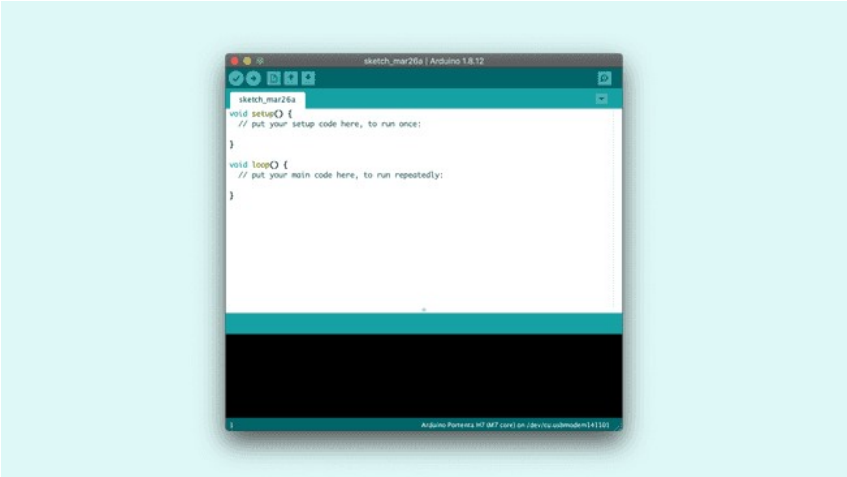
4. go to menu Tools--> Port--> COM(Arduino)
5. click on ✅ to compile, then ➡️ to upload sketch on nodemcu
6. Open serial monitor to view temperature reading send from nodemcu to pc



Experiment – 7

AIM: Communicate Bluetooth with Arduino/ NodeMCU and compose a program to send sensor information to cell phone utilizing Bluetooth.

Setting up ESP32 in the Arduino IDE on Windows

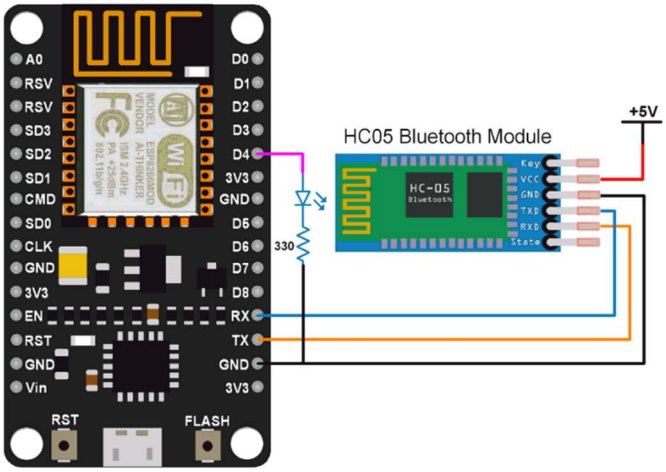


1. Download Arduino ide from Arduino site and install
2. Install the drivers USB chip, download and Install nodemcu Drivers for window
3. Restart system

How to setup Nodemcu(esp8266) board

1. Open Arduino IDE, go to file--> preference--> additional board manager URL:paste json link here
2. paste the link inside the URL "https://arduino.esp8266.com/stable/package_esp8266com_index.json"

Interface nodemcu with Bluetooth module



Adding Nodemcu board to Arduino IDE

1. Open Arduino IDE, go to tools--> board--> board manger--> search:"esp8266" by esp8266 community OR "nodemcu" --> install
2. Selecting Nodemcu board
   Open Arduino IDE, go to tools--> boards--> NodeMCU 1.0 (ESP 12E Module)
3. Write and run program

```
void setup() {
 pinMode(D0, OUTPUT);
 Serial.begin(9600);
}

void loop() {

if (Serial.available()){
  char data;
  data=Serial.read();
  Serial.println(data);

  if (data =='o'){
    digitalWrite(D0,HIGH);
    delay(1000);
  }

  else if(data =='c'){
    digitalWrite(D0,LOW);
    delay(1000);
  }
 }

}
```

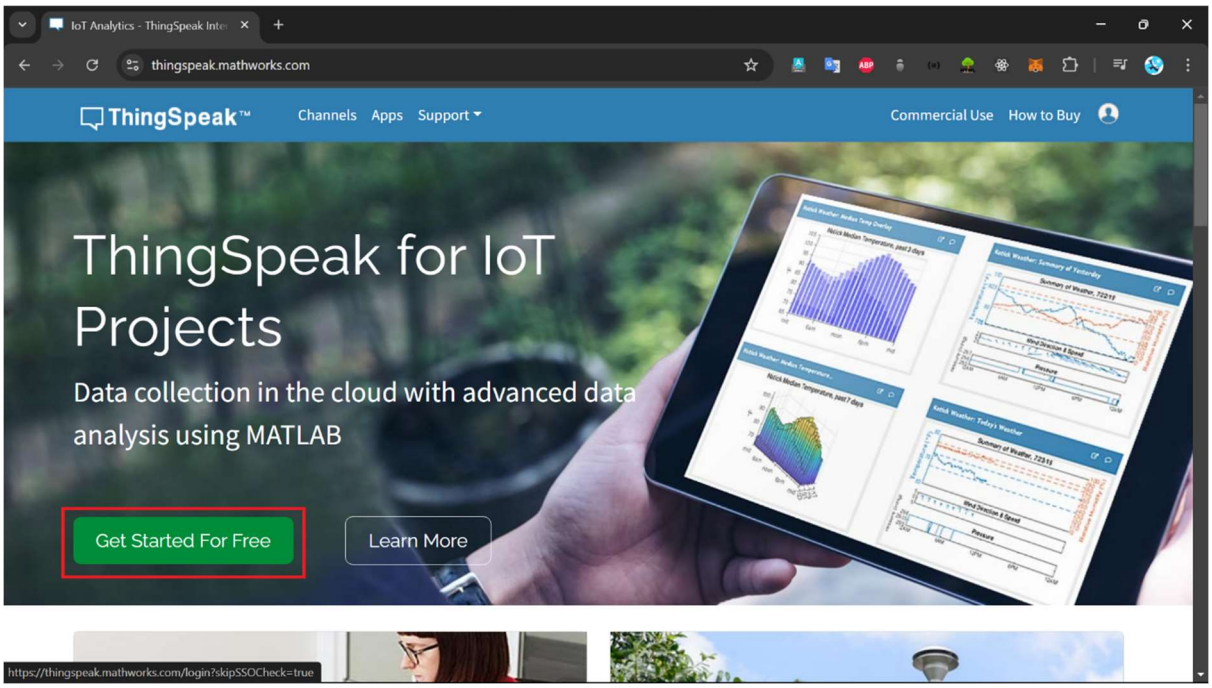
4. go to menu Tools--> Port--> COM(Arduino)
5. click on ✅ to compile, then ➡️ to upload sketch on nodemcu
6. Download Bluetooth HC terminal app from github or playstore
7. Open Bluetooth terminal app, make it pair with Bluetooth
8. In Bluetooth terminal app in send console type “1”, press send to turn on led
9. In Bluetooth terminal app in send console type “0”, press send to turn off led

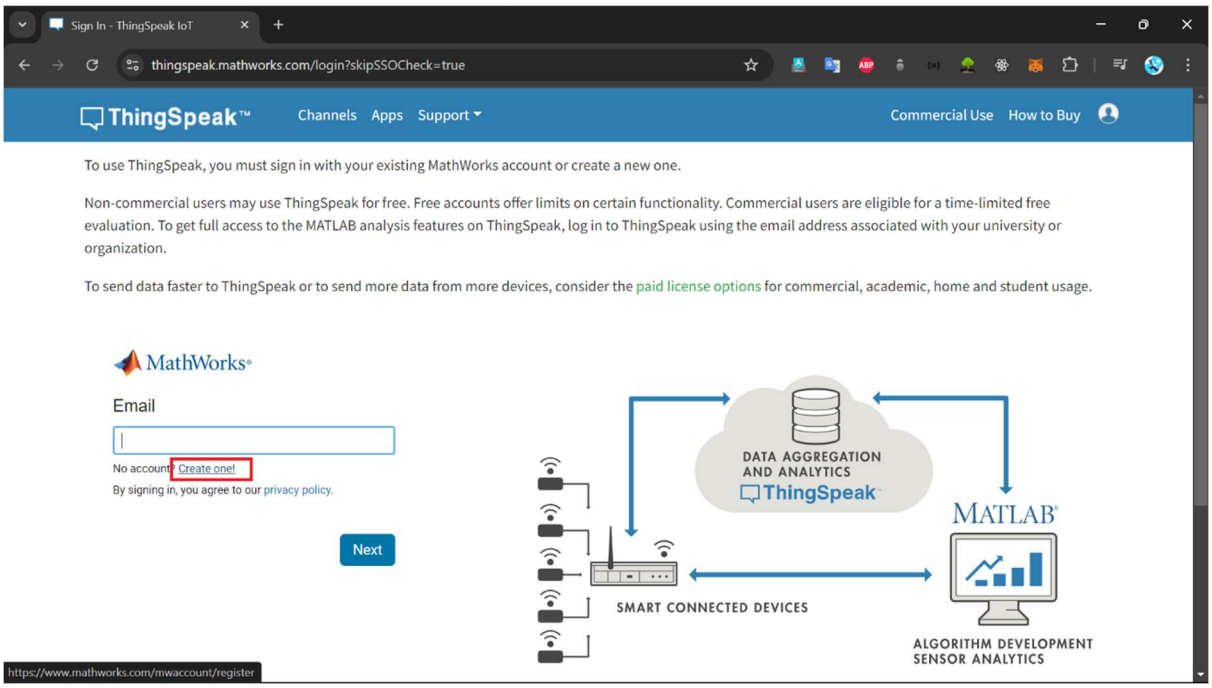
Experiment – 8

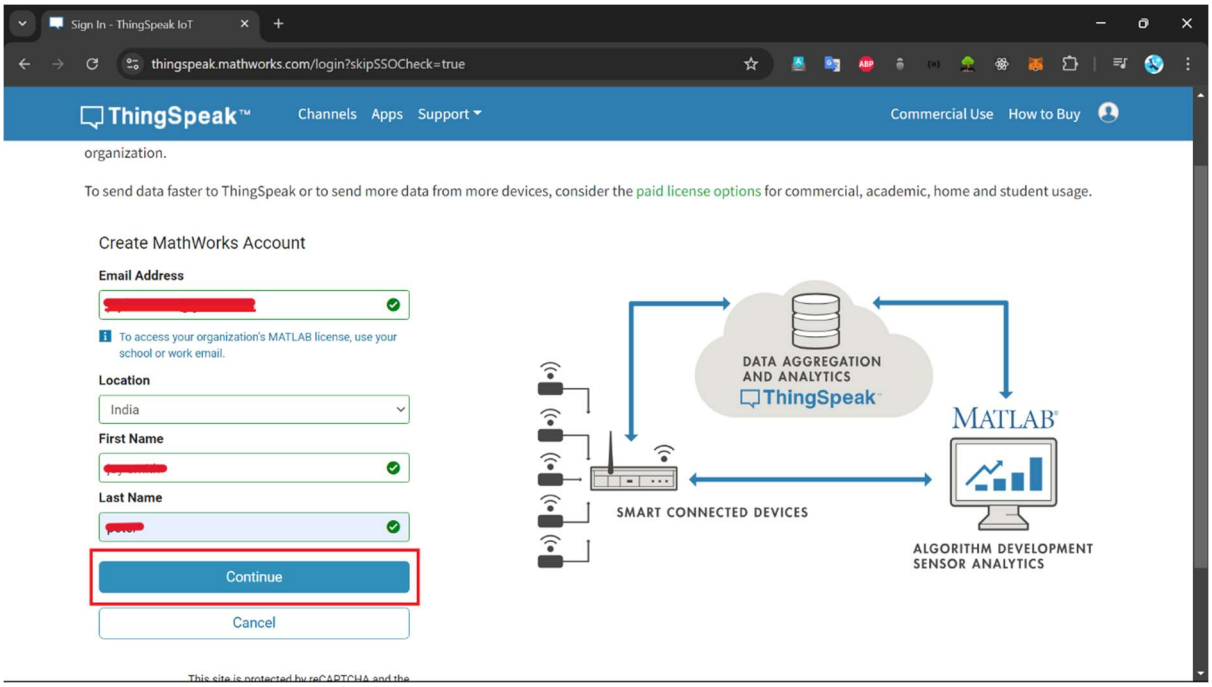
AIM: Compose a program on NodeMCU to transfer temperature and stickiness information to thingspeak, Blynk or any other free cloud

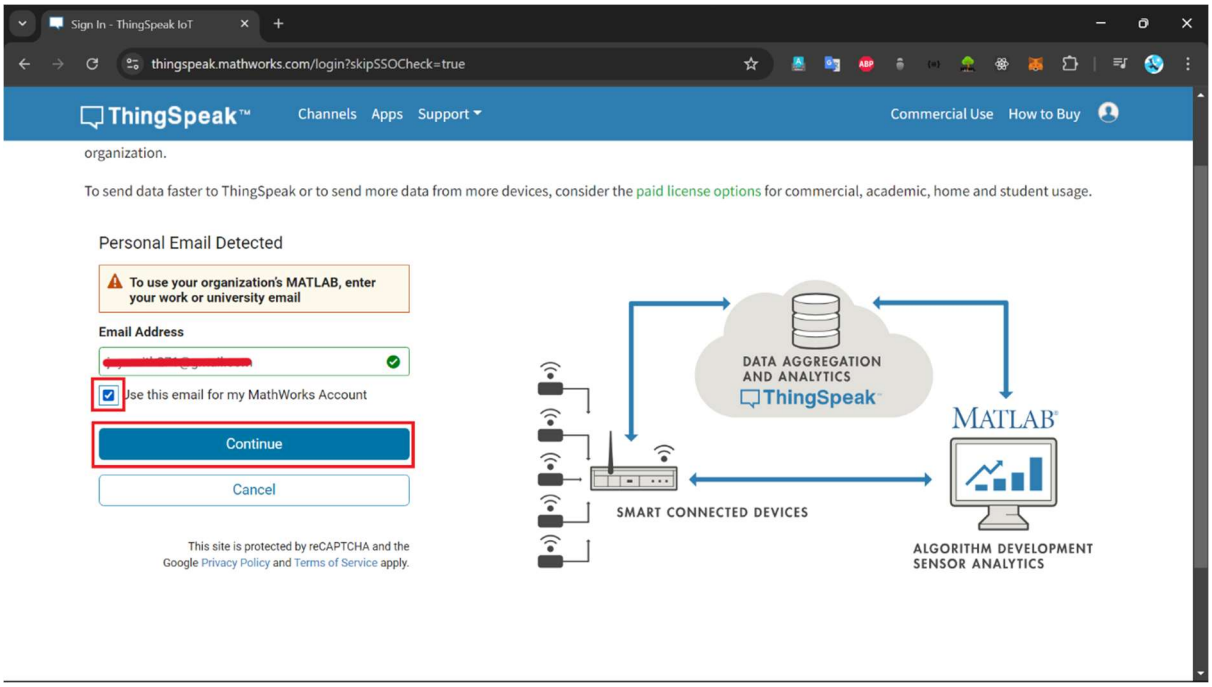
Thing Speak cloud account

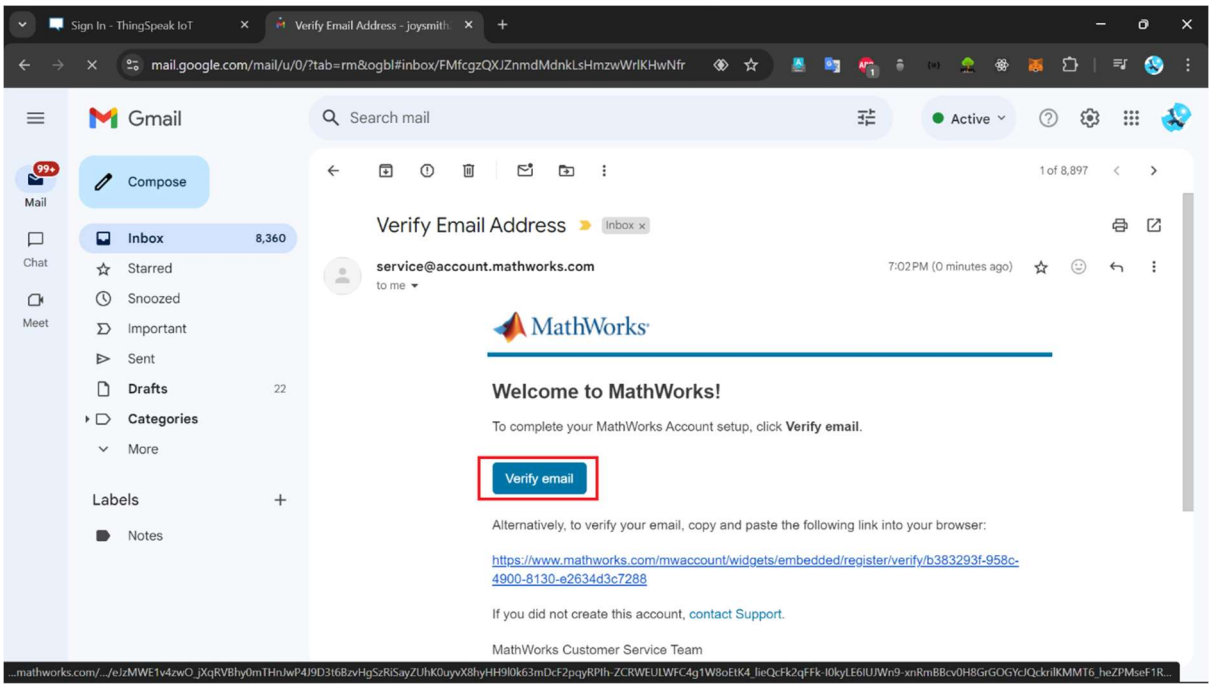
Create thing-speak account

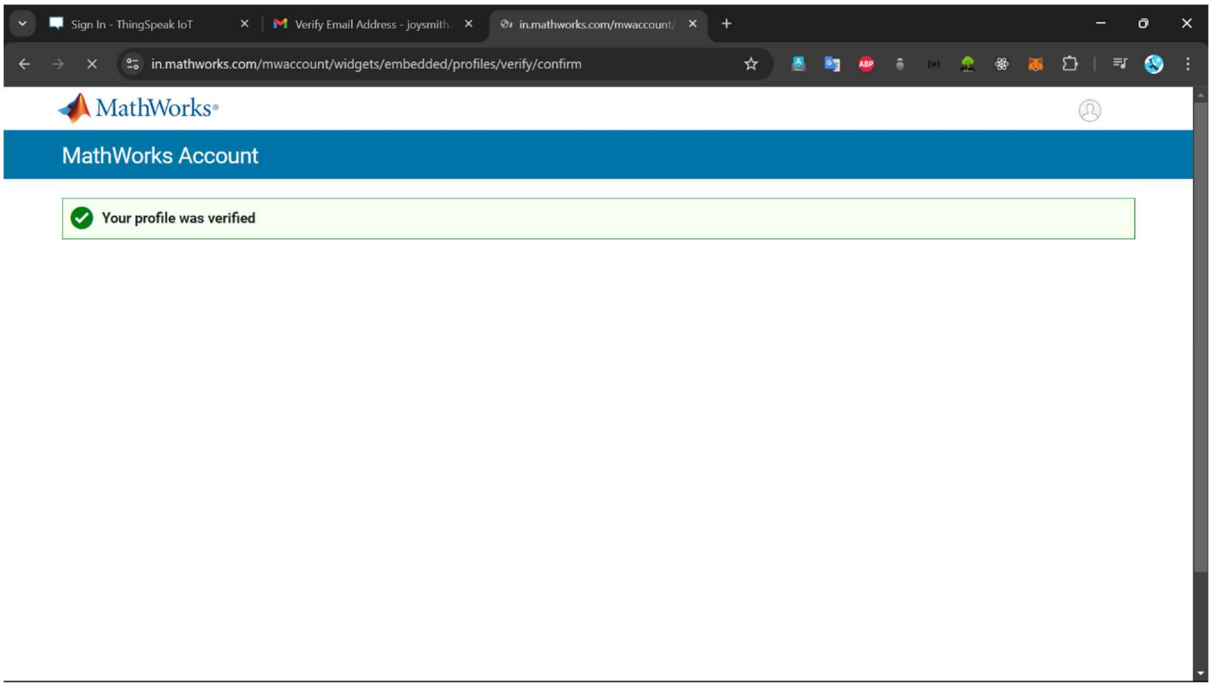


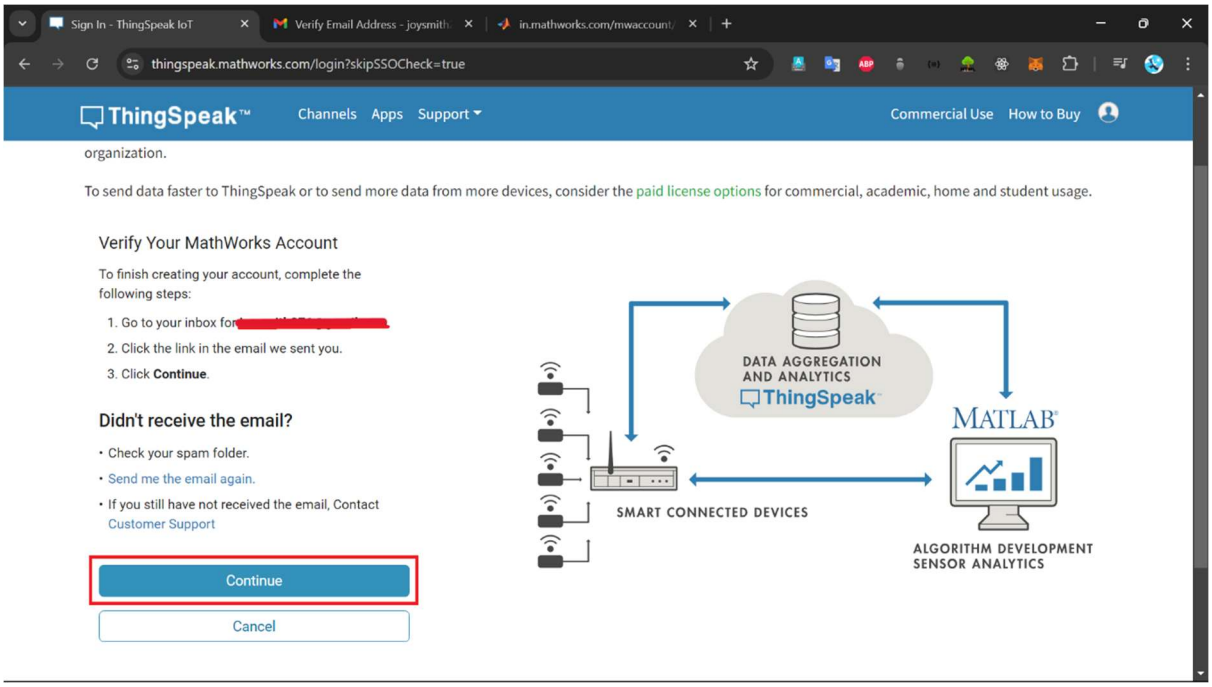


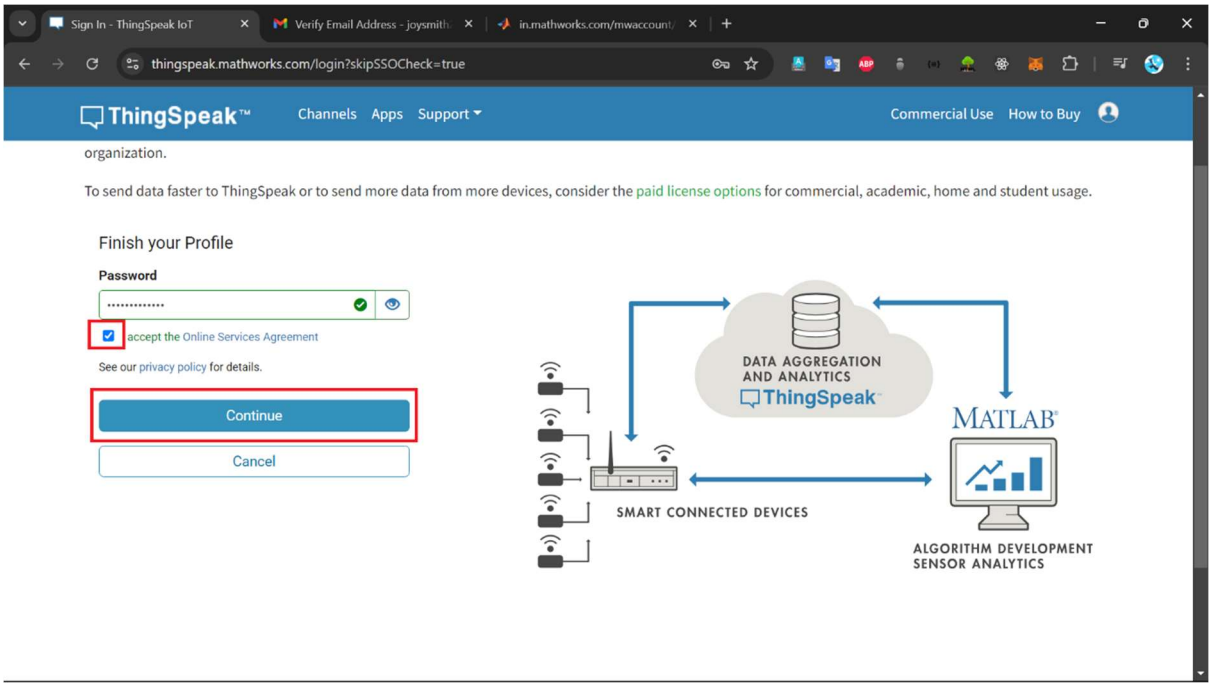

Sign In - ThingSpeak IoT
thingspeak.mathworks.com/login?skipSSOCheck=true
ThingSpeak™
Channels
Apps
Support
Commercial Use
How to Buy
organization.
To send data faster to ThingSpeak or to send more data from more devices, consider the paid license options for commercial, academic, home and student usage.
Create MathWorks Account
Email Address
To access your organization's MATLAB license, use your school or work email.
Location
India
First Name
Last Name
Continue
Cancel
DATA AGGREGATION AND ANALYTICS
ThingSpeak
MATLAB
SMART CONNECTED DEVICES
ALGORITHM DEVELOPMENT
SENSOR ANALYTICS

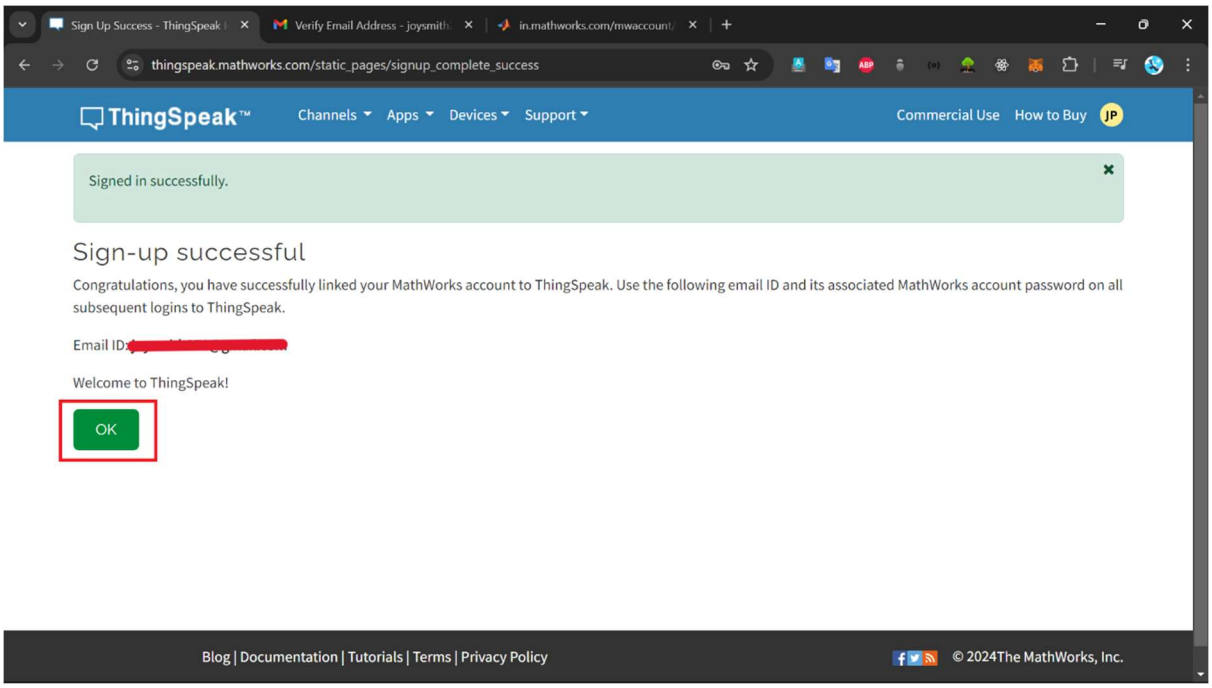

Sign In - ThingSpeak IoT
thingspeak.mathworks.com/login?skipSSOCheck=true
ThingSpeak™
Channels
Apps
Support
Commercial Use
How to Buy
organization.
To send data faster to ThingSpeak or to send more data from more devices, consider the paid license options for commercial, academic, home and student usage.
Personal Email Detected
To use your organization's MATLAB, enter your work or university email
Email Address
Use this email for my MathWorks Account
Continue
Cancel
This site is protected by reCAPTCHA and the Google Privacy Policy and Terms of Service apply.
DATA AGGREGATION AND ANALYTICS
ThingSpeak
MATLAB
SMART CONNECTED DEVICES
ALGORITHM DEVELOPMENT
SENSOR ANALYTICS

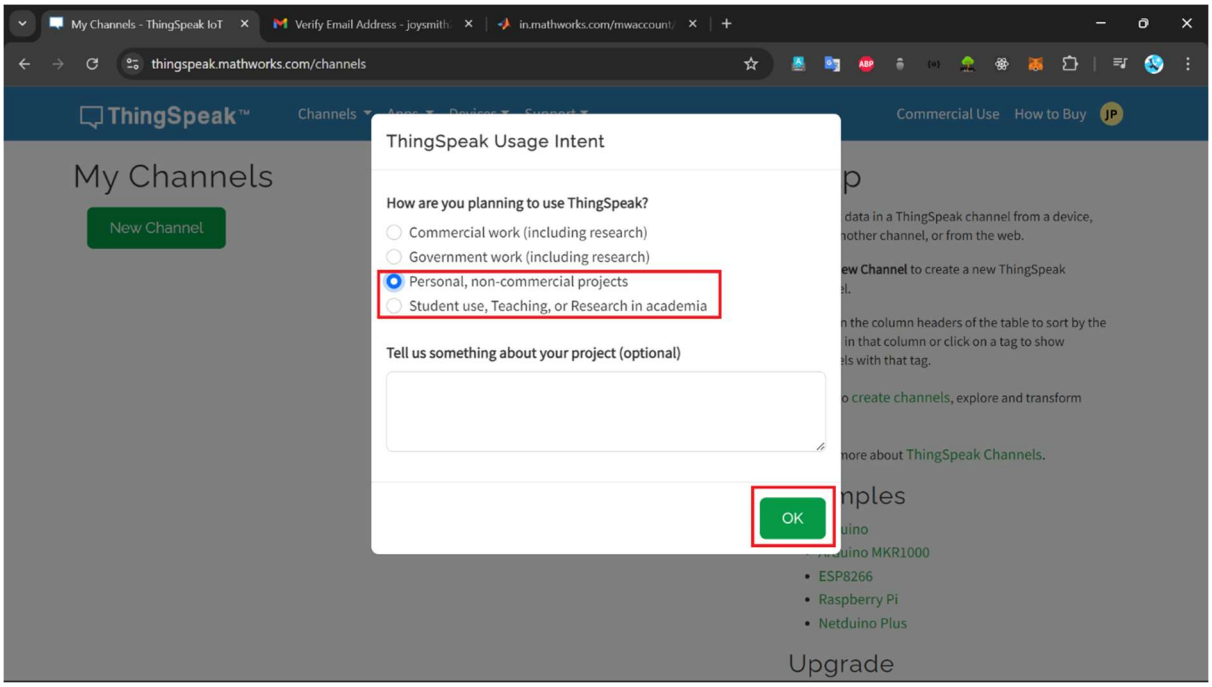

Sign In - ThingSpeak IoT
Verify Email Address - joysmith
mail.google.com/mail/u/0/?tab=rm&ogbl#inbox/FMfcgzQXJZnmdMdnkLsHmzwWrlKHwNfr
Gmail
Search mail
Active
Compose
Inbox 8,360
Starred
Snoozed
Important
Sent
Drafts 22
Categories
More
Labels
Notes
Mail
Chat
Meet
1 of 8,897
Verify Email Address
Inbox
service@account.mathworks.com
to me
7:02 PM (0 minutes ago)
MathWorks
Welcome to MathWorks!
To complete your MathWorks Account setup, click Verify email.
Verify email
Alternatively, to verify your email, copy and paste the following link into your browser:
https://www.mathworks.com/mwaccount/widgets/embedded/register/verify/b383293f-958c-4900-8130-e2634d3c7288
If you did not create this account, contact Support.
MathWorks Customer Service Team

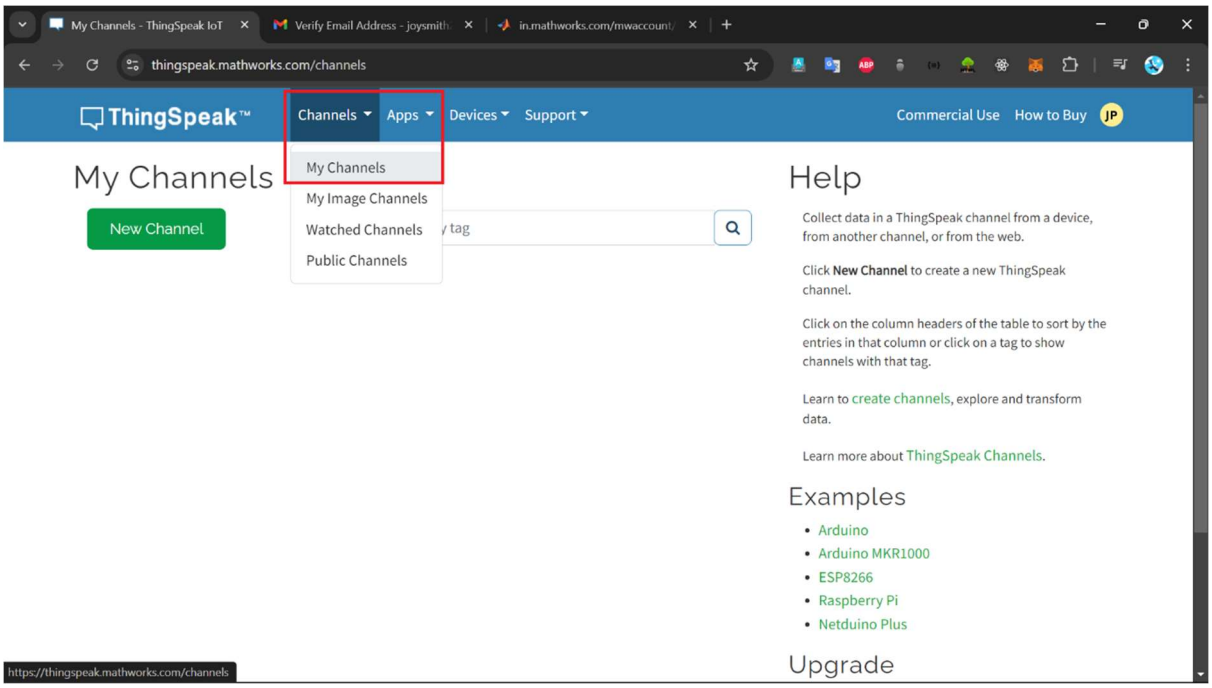

Sign In - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
in.mathworks.com/mwaccount/widgets/embedded/profiles/verify/confirm
MathWorks
MathWorks Account
Your profile was verified

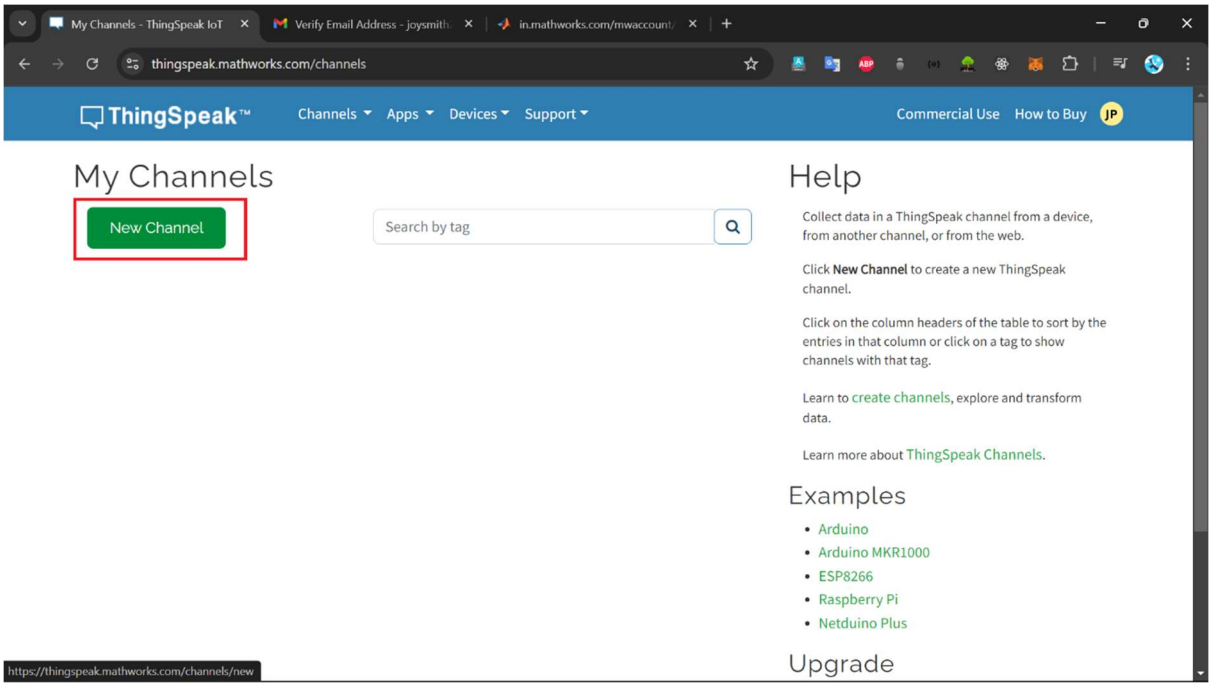

Sign In - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/login?skipSSOCheck=true
ThingSpeak™
Channels
Apps
Support
Commercial Use
How to Buy
organization.
To send data faster to ThingSpeak or to send more data from more devices, consider the paid license options for commercial, academic, home and student usage.
Verify Your MathWorks Account
To finish creating your account, complete the following steps:
1. Go to your inbox for [redacted]
2. Click the link in the email we sent you.
3. Click Continue.
Didn't receive the email?
• Check your spam folder.
• Send me the email again.
• If you still have not received the email, Contact Customer Support
Continue
Cancel
DATA AGGREGATION AND ANALYTICS
ThingSpeak
MATLAB
SMART CONNECTED DEVICES
ALGORITHM DEVELOPMENT
SENSOR ANALYTICS

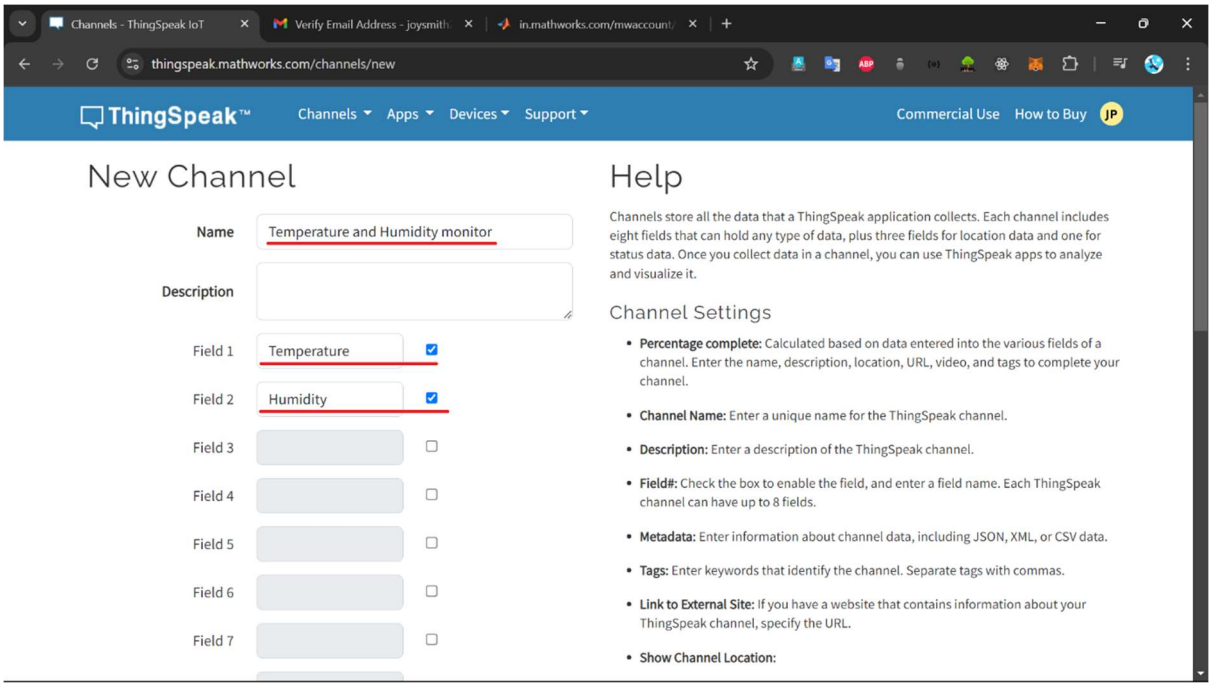

Sign In - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/login?skipSSOCheck=true
ThingSpeak™
Channels
Apps
Support
Commercial Use
How to Buy
organization.
To send data faster to ThingSpeak or to send more data from more devices, consider the paid license options for commercial, academic, home and student usage.
Finish your Profile
Password
accept the Online Services Agreement
See our privacy policy for details.
Continue
Cancel
DATA AGGREGATION AND ANALYTICS
ThingSpeak
MATLAB
SMART CONNECTED DEVICES
ALGORITHM DEVELOPMENT
SENSOR ANALYTICS

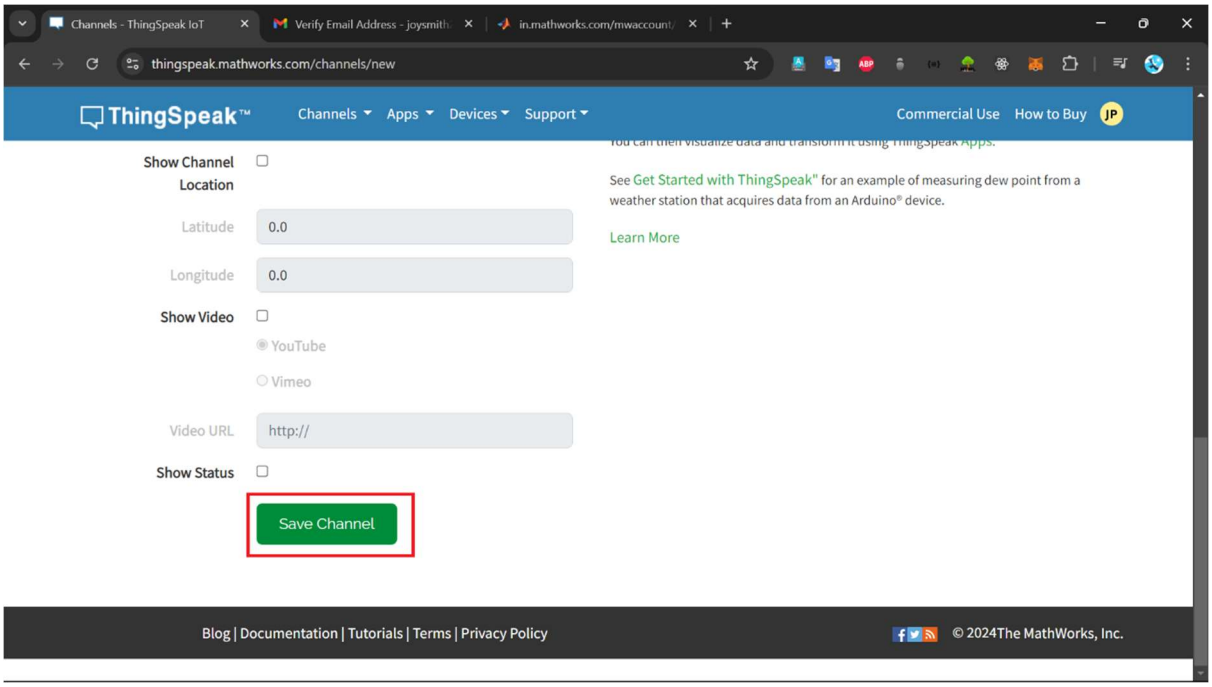

Sign Up Success - ThingSpeak
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/static_pages/signup_complete_success
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
JP
Signed in successfully.
Sign-up successful
Congratulations, you have successfully linked your MathWorks account to ThingSpeak. Use the following email ID and its associated MathWorks account password on all subsequent logins to ThingSpeak.
Email ID:
Welcome to ThingSpeak!
OK
Blog | Documentation | Tutorials | Terms | Privacy Policy
© 2024The MathWorks, Inc.

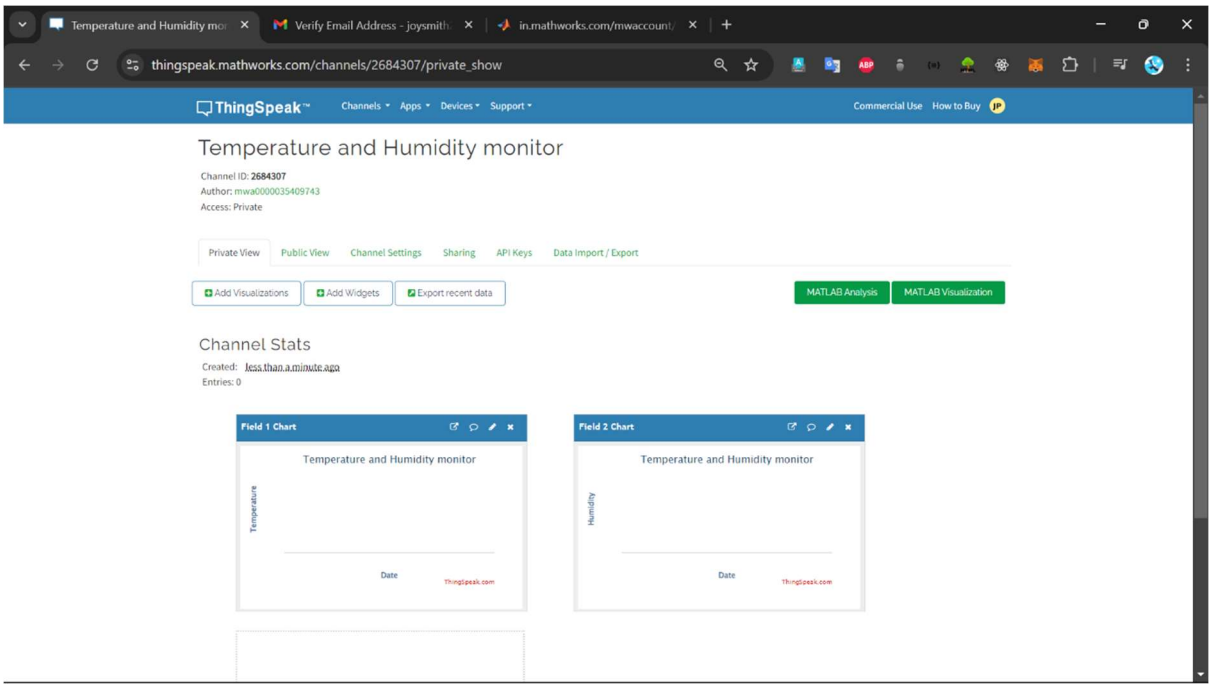

My Channels - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/channels
ThingSpeak™
Channels
Commercial Use
How to Buy
JP
My Channels
New Channel
ThingSpeak Usage Intent
How are you planning to use ThingSpeak?
Commercial work (including research)
Government work (including research)
Personal, non-commercial projects
Student use, Teaching, or Research in academia
Tell us something about your project (optional)
OK
ESP8266
Raspberry Pi
Netduino Plus
Upgrade


My Channels - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/channels
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
JP
My Channels
My Channels
My Image Channels
Watched Channels
Public Channels
New Channel
tag
Help
Collect data in a ThingSpeak channel from a device, from another channel, or from the web.
Click New Channel to create a new ThingSpeak channel.
Click on the column headers of the table to sort by the entries in that column or click on a tag to show channels with that tag.
Learn to create channels, explore and transform data.
Learn more about ThingSpeak Channels.
Examples
Arduino
Arduino MKR1000
ESP8266
Raspberry Pi
Netduino Plus
Upgrade
https://thingspeak.mathworks.com/channels


My Channels - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/channels
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
JP
My Channels
New Channel
Search by tag
Help
Collect data in a ThingSpeak channel from a device, from another channel, or from the web.
Click New Channel to create a new ThingSpeak channel.
Click on the column headers of the table to sort by the entries in that column or click on a tag to show channels with that tag.
Learn to create channels, explore and transform data.
Learn more about ThingSpeak Channels.
Examples
Arduino
Arduino MKR1000
ESP8266
Raspberry Pi
Netduino Plus
Upgrade
https://thingspeak.mathworks.com/channels/new


Channels - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/channels/new
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
JP
New Channel
Name
Temperature and Humidity monitor
Description
Field 1
Temperature
Field 2
Humidity
Field 3
Field 4
Field 5
Field 6
Field 7
Help
Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.
Channel Settings
Percentage complete: Calculated based on data entered into the various fields of a channel. Enter the name, description, location, URL, video, and tags to complete your channel.
Channel Name: Enter a unique name for the ThingSpeak channel.
Description: Enter a description of the ThingSpeak channel.
Field#: Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
Metadata: Enter information about channel data, including JSON, XML, or CSV data.
Tags: Enter keywords that identify the channel. Separate tags with commas.
Link to External Site: If you have a website that contains information about your ThingSpeak channel, specify the URL.
Show Channel Location:


Channels - ThingSpeak IoT
Verify Email Address - joysmith
in.mathworks.com/mwaccount/
thingspeak.mathworks.com/channels/new
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
JP
Show Channel Location
Latitude
0.0
Longitude
0.0
Show Video
YouTube
Vimeo
Video URL
http://
Show Status
Save Channel
See Get Started with ThingSpeak" for an example of measuring dew point from a weather station that acquires data from an Arduino® device.
Learn More
Blog | Documentation | Tutorials | Terms | Privacy Policy
© 2024The MathWorks, Inc.



How to install library

Open arduino IDE and follow steps

- Go to sketch--> include library--> manage library--> Search: "DHT11" by adafruit --> install all
- Go to sketch--> include library--> manage library--> Search: "Thingspeak" by mathwork --> install

Adding Nodemcu board to Arduino IDE

1 Open Arduino IDE, go to tools--> board--> board manger--> search:"esp8266" by esp8266 community OR "nodemcu" --> install

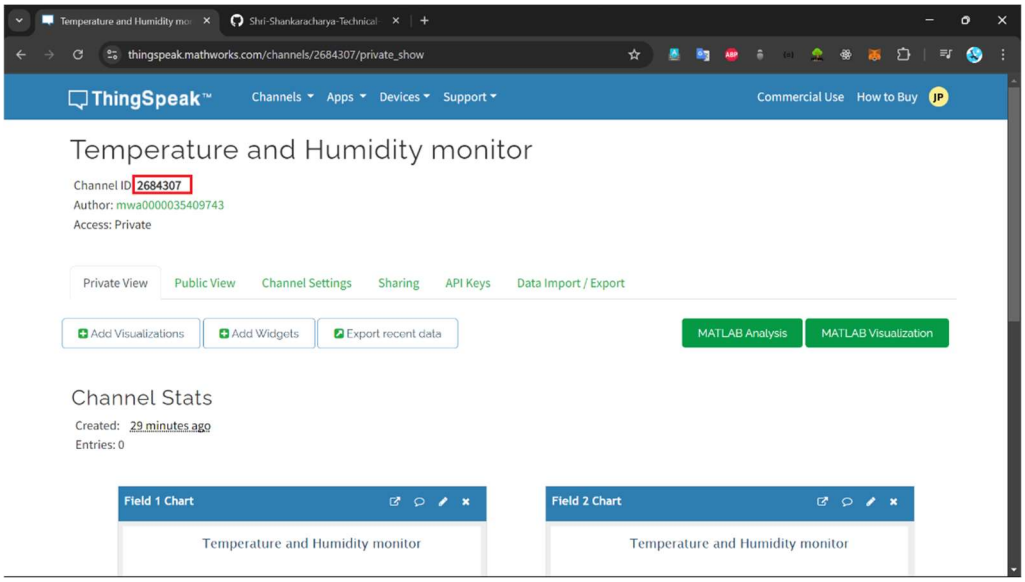
2 Selecting Nodemcu board
Open Arduino IDE, go to tools--> boards--> NodeMCU 1.0 (ESP 12E Module)

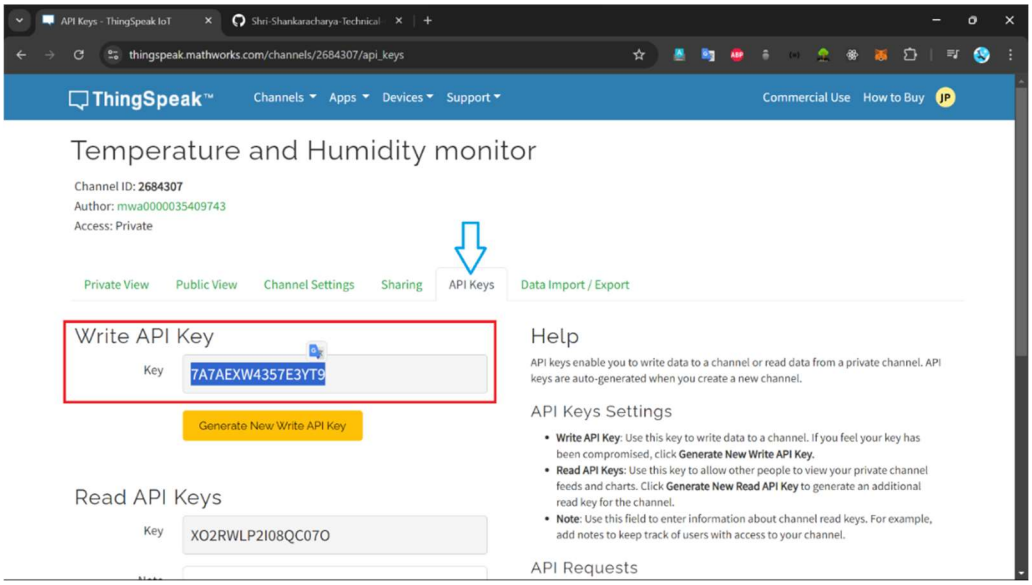
3 Write and run program

```
#include <ESP8266WiFi.h>
#include <DHT.h>
#include <ThingSpeak.h>

DHT dht(D5, DHT11);

WiFiClient client;

// 1 copy channel number from Thingspeak account
long myChannelNumber = 2684307;
```

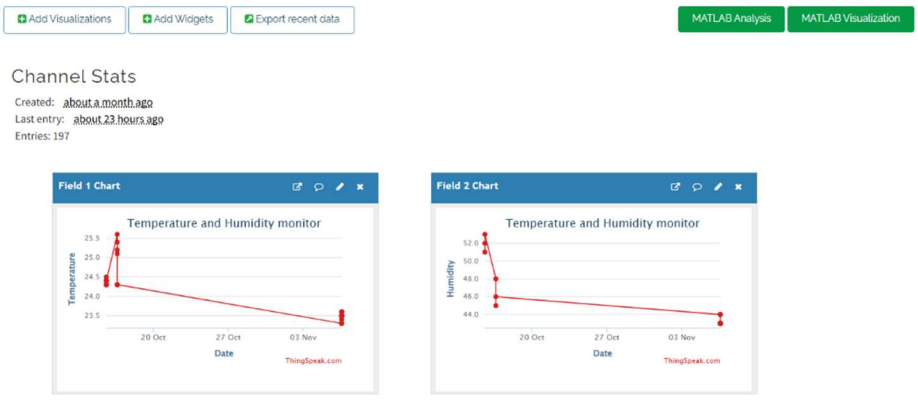
```
// 2. get the api Write key from thingspeak account
const char myWriteAPIKey[] = "7A7AEXW4357E3YT9";

void setup() {
// put your setup code here, to run once:
  Serial.begin(9600);

  // 1. write your wifi name and password inside double-quote
  WiFi.begin("WiFiname", "password");
  while(WiFi.status() != WL_CONNECTED)
  {
    delay(200);
    Serial.print("..", \n");
  }
  Serial.println();
  Serial.println("NodeMCU is connected!");
  Serial.println(WiFi.localIP());
  dht.begin();
  ThingSpeak.begin(client);
}

void loop() {
  // put your main code here, to run repeatedly:
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  Serial.println("Temperature: " + (String) t);
  Serial.println("Humidity: " + (String) h);

  // Real data uncomment below lines if sensor is connected
 // ThingSpeak.writeField(myChannelNumber, 1, t, myWriteAPIKey);
 // ThingSpeak.writeField(myChannelNumber, 2, h, myWriteAPIKey);

  // dummy data 5 , 9 change these value to see fluctuation on chart
   ThingSpeak.writeField(myChannelNumber, 1, 5, myWriteAPIKey);
   ThingSpeak.writeField(myChannelNumber, 2, 9, myWriteAPIKey);


  // delay(100);
}
```

Modify code: write apikey, channel id





Result: Go to thing-speak site and check the data



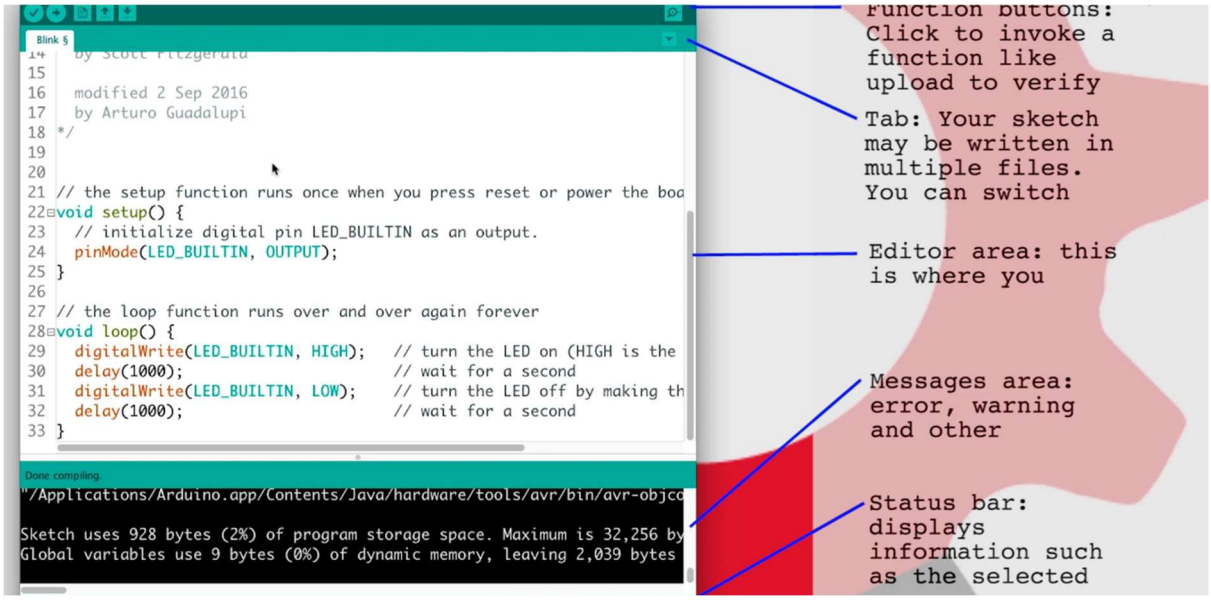
Experiment – 9

AIM: Study of other IoT Boards and components available (Arduino)

How to Select sketch

- From menu select file--> example--> digital--> BlinkwithoutDelay

The Arduino IDE - Understanding the Preferences pane



How to upload a sketch to your Arduino

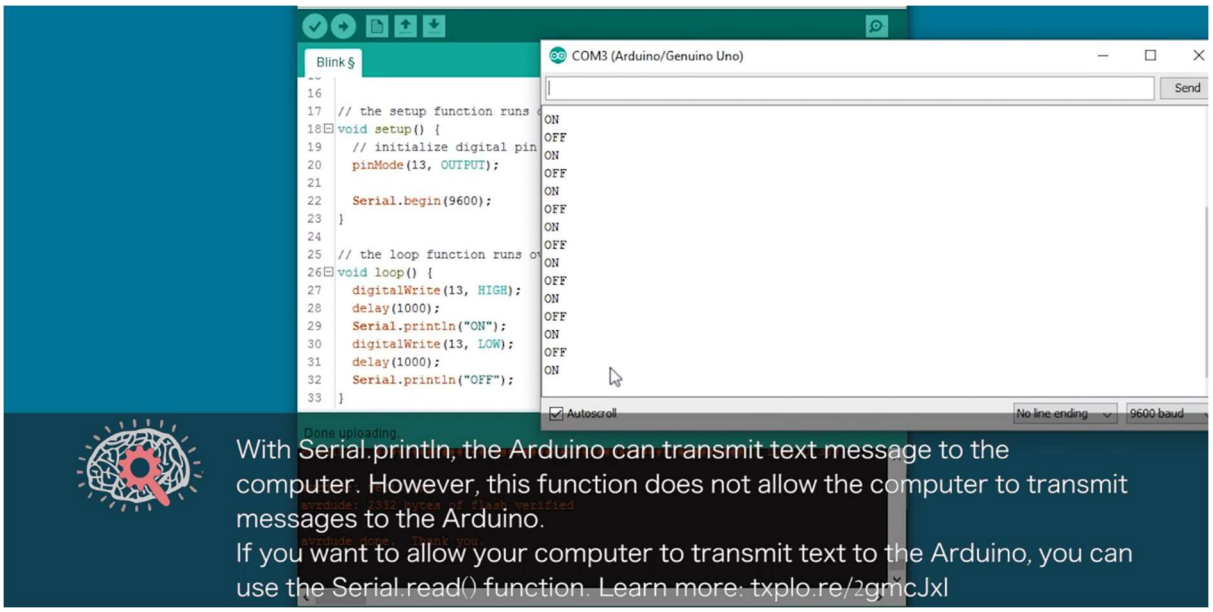
1. go to menu Files--> Example--> basic--> blink

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);  // turn the LED on (HIGH is the voltage
level)
  delay(1000);                      // wait for a second

  digitalWrite(LED_BUILTIN, LOW);   // turn the LED off by making the
voltage LOW
  delay(1000);                      // wait for a second
```

```
}
```

2. go to menu Tools--> Boards--> arduino avr board--> arduino uno
3. go to menu Tools--> Port--> COM(arduino)
4. click on ✅ to compile, then ➡️ to upload sketch on uno

How to use serial monitor

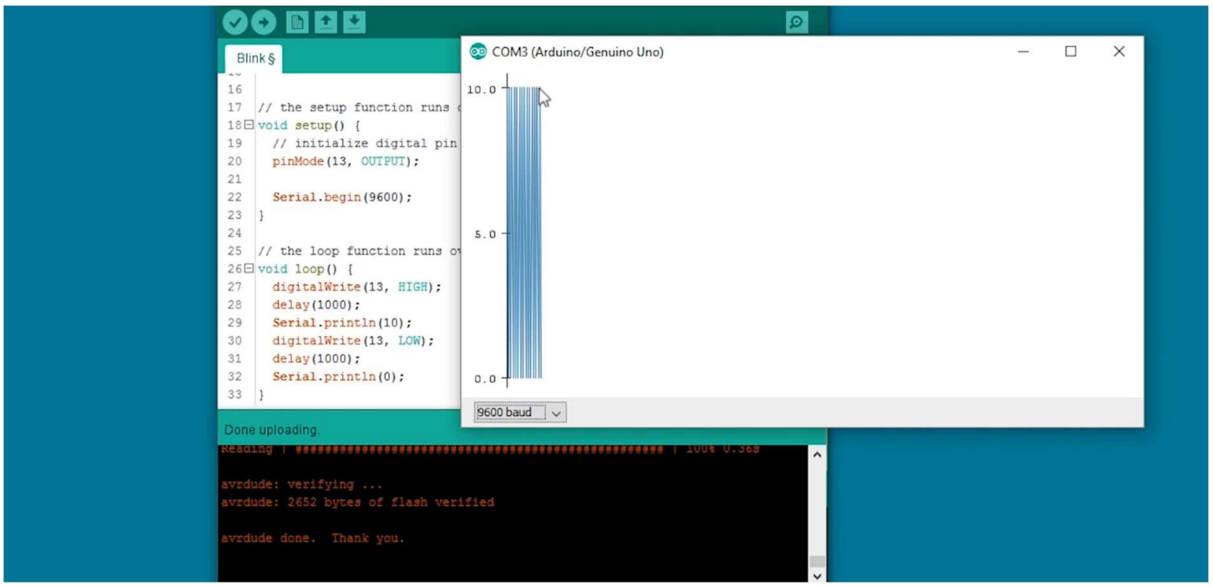


1. go to menu Files--> Example--> basic--> blink

```
// the setup function runs once when you press reset or power the board
   void setup() {
     // initialize digital pin LED_BUILTIN as an output.
     pinMode(LED_BUILTIN, OUTPUT);
     Serial.begin(9600);
   }

   // the loop function runs over and over again forever
   void loop() {
     digitalWrite(LED_BUILTIN, HIGH);  // turn the LED on (HIGH is the
   voltage level)
     delay(1000);                       // wait for a second
     Serial.println("ON");

     digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the
   voltage LOW
     delay(1000);                      // wait for a second
     Serial.println("OFF");
   }
```

2. go to menu Tools--> Boards--> arduino avr board--> arduino uno
3. go to menu Tools--> Port--> COM(arduino)
4. click on ✅ to compile, then ➡️ to upload sketch on uno
5. go to menu Tools--> Serial Monitor

How to use Serial Plotter



1. go to menu Files--> Example--> basic--> blink

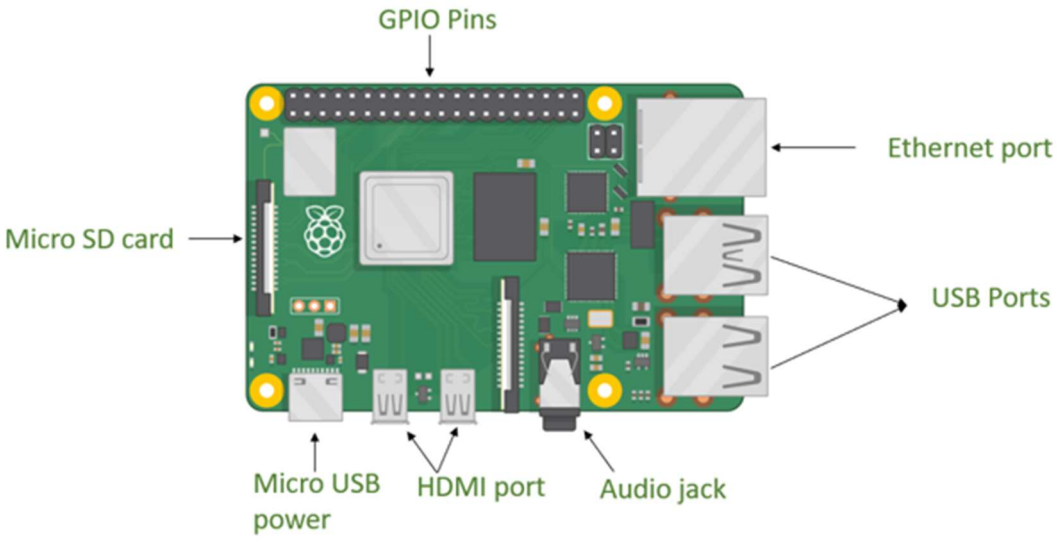
```
// the setup function runs once when you press reset or power the board
void setup() {
     // initialize digital pin LED_BUILTIN as an output.
     pinMode(LED_BUILTIN, OUTPUT);
     Serial.begin(9600);
   }

      // the loop function runs over and over again forever
   void loop() {
     digitalWrite(LED_BUILTIN, HIGH);  // turn the LED on (HIGH is the voltage
   level)
     delay(1000);                      // wait for a second
     Serial.println("ON");

     digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the
   voltage LOW
     delay(1000);                     // wait for a second
     Serial.println("OFF");
   }
```

2. go to menu Tools--> Boards--> arduino avr board--> arduino uno
3. go to menu Tools--> Port--> COM(arduino)
4. click on ✅ to compile, then ➡️ to upload sketch on uno
5. go to menu Tools--> Serial Plotter

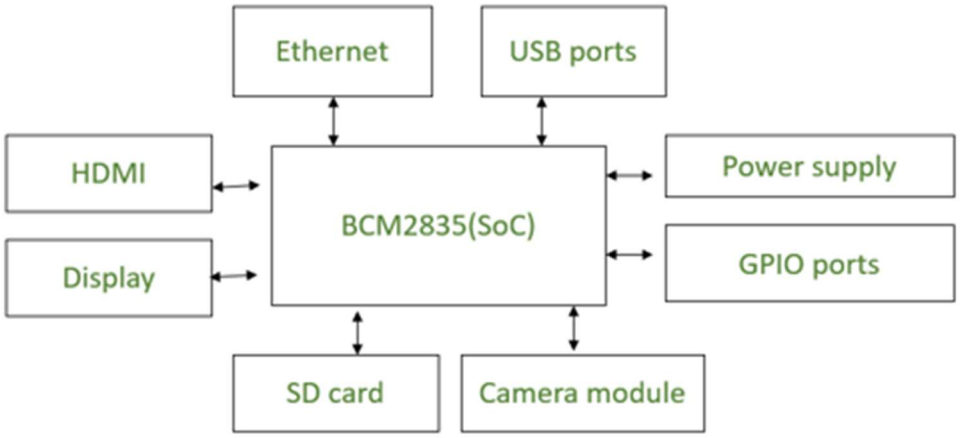
Experiment – 10

AIM: Study of other IoT Boards and components available (Raspberry pi)

Raspberry Pi Development Board



Raspberry Pi is a credit card sized single computer board. It can be used for many of the things that your desktop PC does, like spreadsheets, word-processing, games and it can also play high-definition video. It was developed by Raspberry Pi foundation from UK. The raspberry pi is ready for public consumption since 2012 with the idea of producing a cheep educational microcomputer for students and children. The raspberry pi is created to be something to encourage learning, innovation and experimentation. The raspberry pi computer is portable and less expensive. Most of the raspberry boards are used to built raspberry pi projects, mobile phones and also used in solar street lights. The start of 21st century saw a huge growth in mobile computing technologies, a large segment of this being driven by the mobile phone industry. The 98 percent mobile phones were using ARM technology. The ARM technology would later end up being featured on the raspberry pi with ARM processor core being used.



The system comes into two different models that is model A and model B. The main differences between these two are USB ports. The model A board which does not includes a Ethernet port and will consume less power. The model B that includes an Ethernet port and manufactured in China. The raspberry pi comes with set of open source technologies that is

communication and multimedia of web technologies and ability to interact with microcontroller. It is portability of a mobile device.

Raspberry Pi Specifications:

Memory:

The raspberry pi comes equipped with 256 Mb of SDRAM in older version model A and 512 Mb on the newer versions model B. It is a small size PC compare with other PCs. The normal PCs RAM memory is available in gigabytes but in this type of application the RAM memory is more than 256 Mb or 512 Mb.

CPU:

The CPU is the main component of the raspberry pi. It is responsible for carrying out the instructions of a computer via mathematical and logical operations. The raspberry pi is a good company using the ARM11 series processor. It has joined the ranks of phone, Samsung galaxy.

GPU:

The graphics processing unit (GPU) is a specialized chip in raspberry pi. It is designed to speed up the manipulation of image calculations. The raspberry pi comes equipped with a Broadcom video core IV and it is supported for OpenGL.

Ethernet Port:

The raspberry pi Ethernet port is main gateway for communicating with other devices. You will able to use raspberry pi Ethernet port to plug your home router to access the internet.

GPIO Pins:

The general purpose input and output pins on the raspberry pi are used to connect with the other electronic boards. The GPIO pins can accept input and output commands based on programmed raspberry pi.

The raspberry pi provides digital GPIO pins. These pins are used to connect other electronic components. For example you can connect it to the temperature sensor which transmitts digital data.

XBee Socket:

The raspberry pi proves two XBee socket for wireless communication purpose.

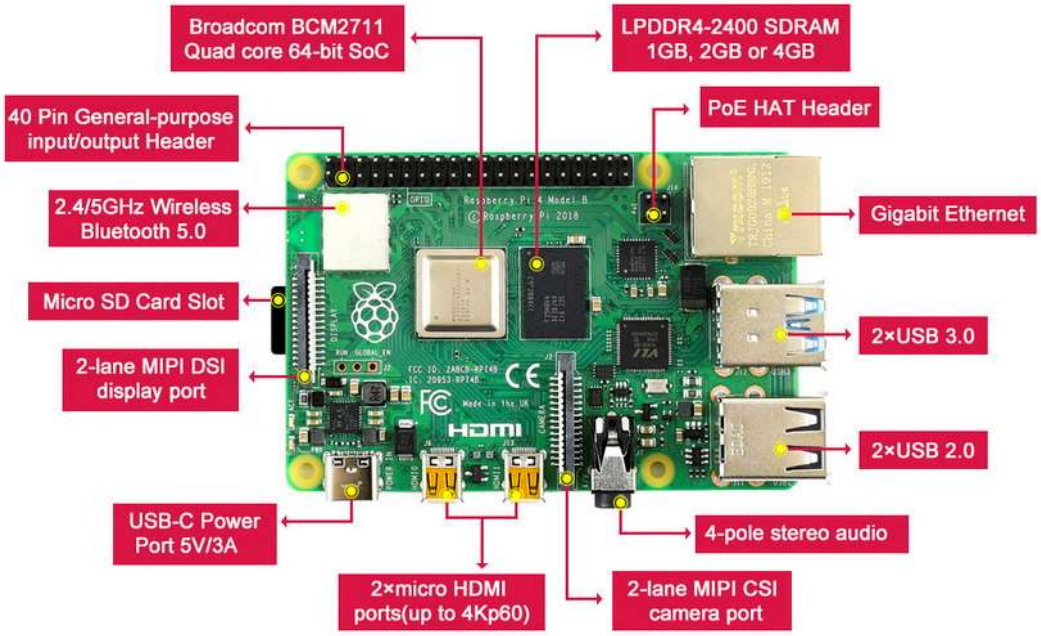
Power Source Connector:

The power source selector is a small switch it is located on side of the shield that can be used to enable an external power source.

UART:

The UART are a serial input and output port. This can be used to transfer the serial data such as text and it is useful for transforming the debugging code.

Raspberry pi 4B Development Board:



Raspberry Pi 4 B has a 64-bit quad-core processor running at 1.5Ghz, supports dual displays with up to 4K resolution refreshed at 60fps, up to 4GB RAM (1GB, 2GB, 4GB available according to model), 2.4/5.0 Ghz Dual-band wireless LAN, Bluetooth 5.0/BLE, Gigabit Ethernet, USB3.0, and PoE functionality.