

What is NODEMCU?



Node MCU is a firmware based on ESP8266 WiFi. It can be seen as a WiFi-enabled advanced microcontroller which can connect various devices and transfer data not on the internet. Being a firmware, Node MCU is a microcontroller unit with memory, GPIO pins, ADCe, etc. Having several advantages over the basic Arduino Uno board which is commonly used, the NodeMCU proves to be a good choice while dealing with online data transfer and storage.

Scientific Fact and Applications

Similar to Arduino, NodeMCU is a open source platform originally built in Lua script. Previously only Lua-based scripts used to run on it. However, NodeMCU can now be programmed in C after it was flashed by Arduino IDE.

Applications:

Serial Port Monitor

Since NodeMCU follows serial communication and is also Wi-Fi-enabled, data can be sent serially over the internet and can be monitored easily. It is usually followed where large amount of data are collected continuously and it is difficult to view the data by multiple user offline. So, it shares the media data using its ESP8266 capabilities.

Internet of Things (IOT)

IOT is a connection of various computing devices and the ability to transfer data over the shared network. The various devices can be connected together using the microcontroller unit and data can be transferred over the network using the Wi-Fi capabilities. NodeMCU proves to be a better option for several projects where data is to be gathered and shared simultaneously like security system, IoT home automation, etc.



IOT with NodeMCU

<http://www.skillthon.com/iot-using-node-mcu/>

Project

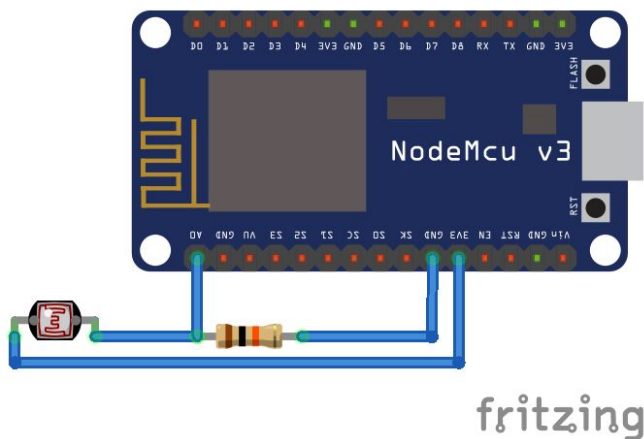
Components Required

Component	Part No.	Qty
Arduino UNO	EMX-00001-A	1
NODE MCU	-	1
LDR	EMS-00017-A	1
Resister - 220Ohm	EDR-00001-220Z	1

Procedure

1. Connect one leg of the LDR to the **3V3/ 3V** pin of Node MCU.
2. Connect the other leg to the **A0** pin of Node MCU and with the same leg, connect it to the **GND** of Node MCU with a 220ohm resistor in between.

Schematic



Challenge Yourself

Code

```
#include <ESP8266WiFi.h>;
/*Including ESP8266WiFi.h library*/
#include <WiFiClient.h>;
/*Including WiFiClient.h library*/
#include <ThingSpeak.h>;
/*Including ThingSpeak.h library*/

const char* ssid = "Your SSID Here";
/*Your Network SSID - ie Your WiFi
Name*/
const char* password = "Your Password
Here"; /*Your WiFi Network Password*/
int value;
/*Defining an integer named "val" to
store the read value from sensor*/
int LDRpin = A0; /*LDR Pin Connected at
A0 Pin*/

WiFiClient client;
unsigned long myChannelNumber = YYYYYY;
/*Your Channel Number (Without
Brackets)*/
const char * myWriteAPIKey =
"XXXXXXXXXXXXXXXXXX"; /*Your Write API
Key*/

void setup() {
  Serial.begin(9600); /*Setting the
baudrate to 9600*/
  delay(10);
  /* Connect to WiFi network*/
  WiFi.begin(ssid, password);
  ThingSpeak.begin(client);
}

void loop(){
  value = analogRead(LDRpin);
  /*Read Analog values and Store in val
variable*/
  Serial.print(value);
  /*Print on Serial Monitor*/
  delay(1000);
  ThingSpeak.writeField(myChannelNumber,
1,value, myWriteAPIKey);
  /*Update in ThingSpeak*/
  delay(100);
}
```