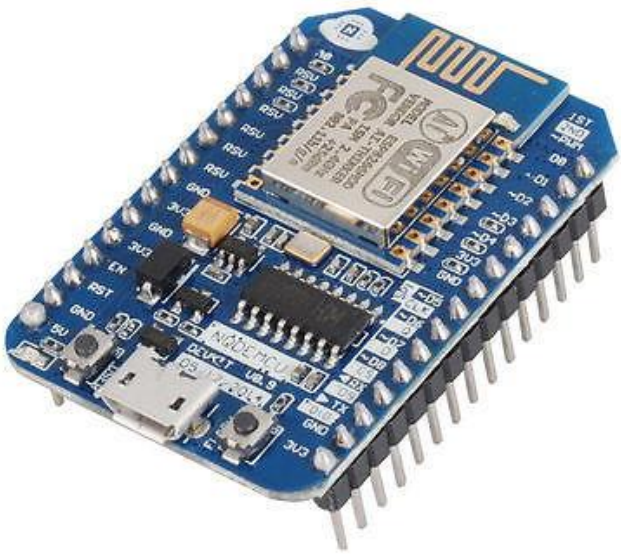
INTRODUCTION TO NODEMCU

INTRODUCTION

NodeMCU is an open source LUA based firmware developed for ESP8266 wifi chip. By exploring functionality with ESP8266 chip, NodeMCU firmware comes with ESP8266 Development board/kit i.e. NodeMCU Development board.



NodeMCU Development Board/kit v0.9 (Version1)

Since NodeMCU is open source platform, their hardware design is open for edit/modify/build. NodeMCU Dev Kit/board consist of ESP8266 wifi enabled chip. The **ESP8266** is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. For more information about ESP8266, you can refer ESP8266 WiFi Module.There is Version2 (V2) available for NodeMCU Dev Kit i.e. **NodeMCU Development Board v1.0 (Version2)**, which usually comes in black colored PCB.

NodeMCU Development Board/kit v1.0 (Version2)

NodeMCU Dev Kit has **Arduino like** Analog (i.e. A0) and Digital (D0-D8) pins on its board.
It supports serial communication protocols i.e. UART, SPI, I2C etc.
Using such serial protocols we can connect it with serial devices like I2C enabled LCD display, Magnetometer HMC5883, MPU-6050 Gyro meter + Accelerometer, RTC chips, GPS modules, touch screen displays, SD cards etc.

How to start with NodeMCU?

NodeMCU Development board is featured with wifi capability, analog pin, digital pins and serial communication protocols. To get start 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. There is online NodeMCU custom builds available using which we can easily get our custom NodeMCU firmware as per our requirement.

How to write codes for NodeMCU?

After setting up ESP8266 with Node-MCU firmware, let's see the IDE (Integrated Development Environment) required for development of NodeMCU.

NodeMCU with ESPlorer IDE

Lua scripts are generally used to code the NodeMCU. Lua is an open source, lightweight, embeddable scripting language built on top of C programming language.

NodeMCU with Arduino IDE

Here is another way of developing NodeMCU with a well-known IDE i.e. Arduino IDE. We can also develop applications on NodeMCU using Arduino development environment. This makes easy for Arduino developers than learning new language and IDE for NodeMCU.

Difference in using ESPlorer and Arduino IDE

Well, there is a programming language difference we can say while developing application for NodeMCU using ESPlorer IDE and Arduino IDE.We need to code in C\C++ programming language if we are using Arduino IDE for developing NodeMCU applications and Lua language if we are using ESPlorer IDE.
Basically, NodeMCU is Lua Interpreter, so it can understand Lua script easily. When we write Lua scripts for NodeMCU and send/upload it to NodeMCU, then they will get executes sequentially. It will not build binary firmware file of code for NodeMCU to write. It will send Lua script as it is to NodeMCU to get execute.
In Arduino IDE when we write and compile code, ESP8266 toolchain in background creates binary firmware file of code we wrote. And when we upload it to NodeMCU then it will flash all

NodeMCU firmware with newly generated binary firmware code. In fact, it writes the complete firmware.

That's the reason why NodeMCU not accept further Lua scripts/code after it is getting flashed by Arduino IDE. After getting flashed by Arduino sketch/code it will be no more Lua interpreter and we got error if we try to upload Lua scripts. To again start with Lua script, we need to flash it with NodeMCU firmware.

Since Arduino IDE compile and upload/writes complete firmware, it takes more time than ESPlorer IDE.

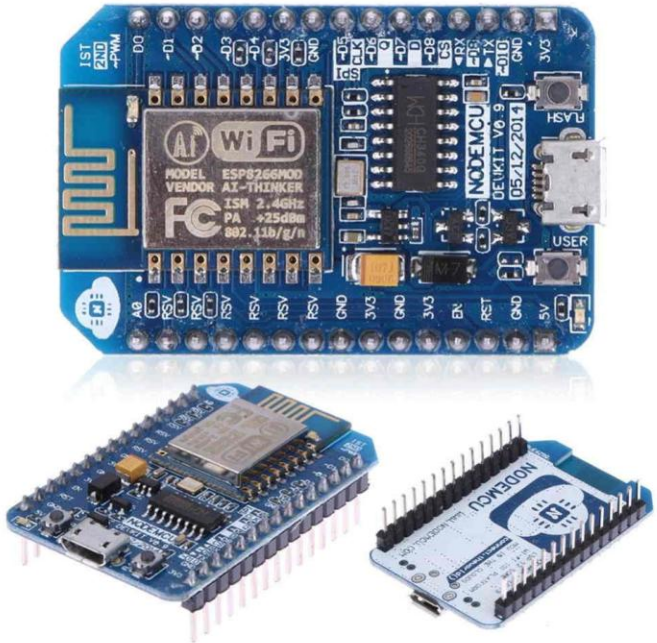
NODEMCU DEVELOPMENT KIT/BOARD

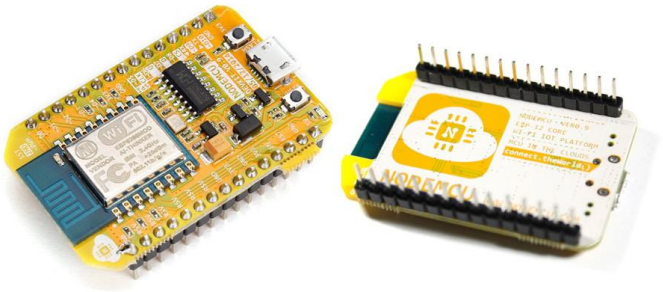
NodeMCU Development Kit/Board consist of ESP8266 wifi chip. ESP8266 chip has GPIO pins, serial communication protocol, etc. features on it.

ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. For more information about ESP8266, you can refer ESP8266 WiFi Module.

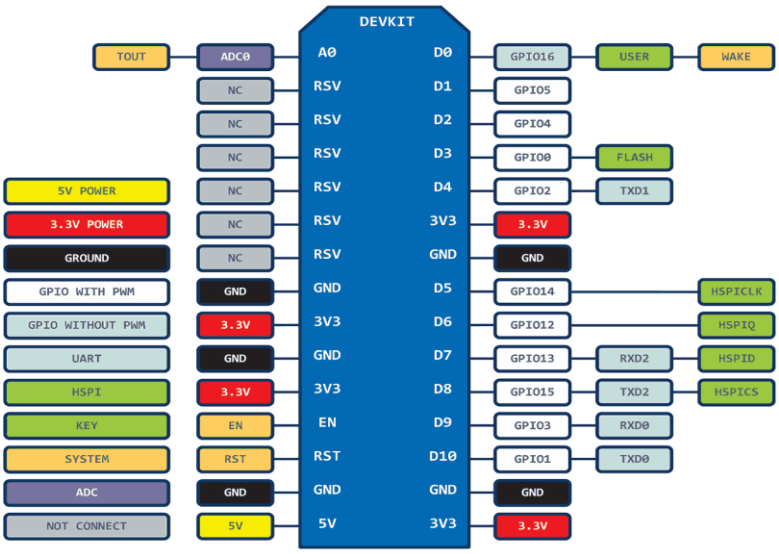
The features of ESP8266 are extracted on NodeMCU Development board. NodeMCU (LUA based firmware) with Development board/kit that consist of ESP8266 (wifi enabled chip) chip combines NodeMCU Development board which make it stand-alone device in IoT applications.

Let's see 1st version of NodeMCU Dev Kit and its pinout as shown in below images.





NodeMCU Development Board v0.9 (Version1)

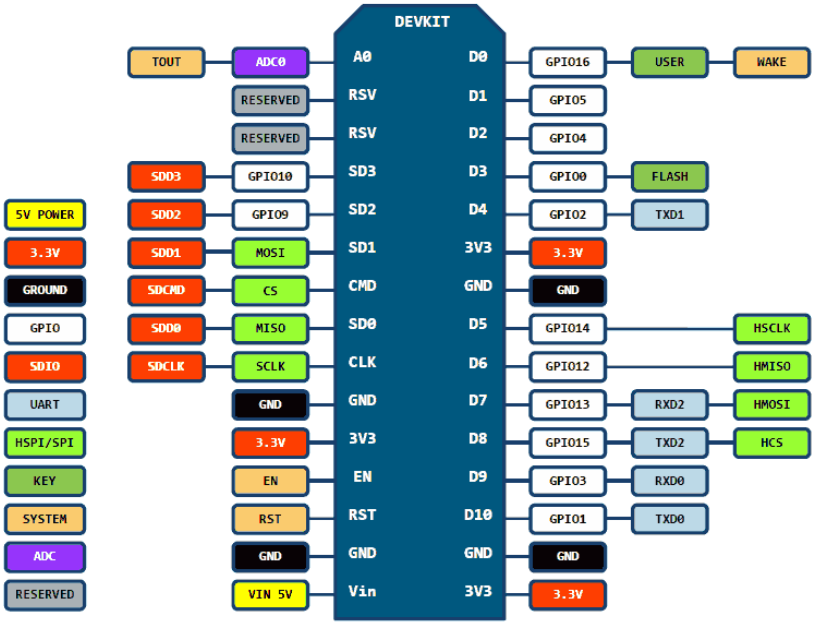


NodeMCU Dev Kit v0.9(V1) Pinouts

2nd version of NodeMCU Dev Kit and its Pinout as shown in below images.

Amica NodeMCU Dev Kit v1.0(Version2)

PIN DEFINITION



NodeMCU Dev Kit v1.0(V2) Pinouts

NodeMCU Dev Kit v1.0 pin descriptions

GPIO (General Purpose Input Output) Pins:

NodeMCU has general purpose input output pins on its board as shown in above pinout diagram. We can make it digital high/low and control things like LED or switch on it. Also, we can generate PWM signal on these GPIO pins.

ADC (Analog to Digital Converter) channel (A0):

NodeMCU has one ADC channel/pin on its board.

SPI (Serial Peripheral Interface) Pins:

NodeMCU based ESP8266 has Hardware SPI (HSPI) with four pins available for SPI communication. It also has SPI pins for Quad-SPI communication. With this SPI interface, we can connect any SPI enabled device with NodeMCU and make communication possible with it.

I2C (Inter-Integrated Circuit) Pins:

NodeMCU has I2C functionality support on ESP8266 GPIO pins. Due to internal functionality on ESP-12E we cannot use all its GPIOs for I2C functionality. So, do tests before using any GPIO for I2C applications.

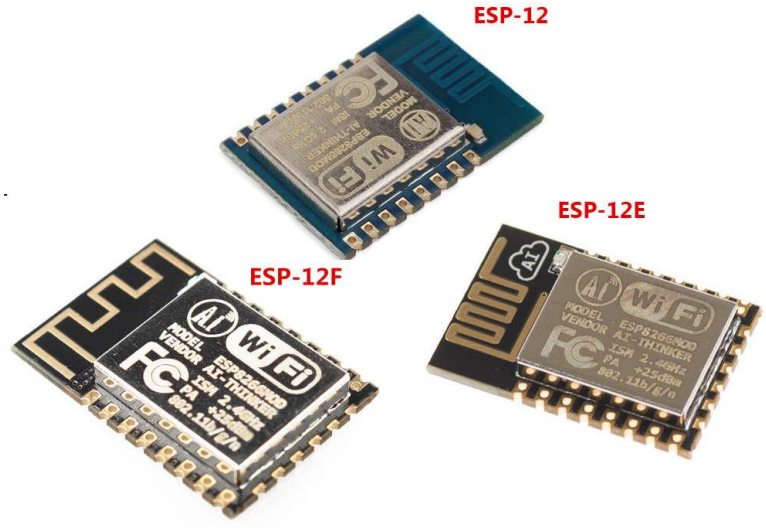
UART (Universal Asynchronous Receiver Transmitter) Pins:

NodeMCU based ESP8266 has two UART interfaces, UART0 and UART1. Since UART0 (RXD0 & TXD0) is used to upload firmware/codes to board, we can't use them in applications while uploading firmware/codes.

Difference in between 1st and 2nd version NodeMCU Board

We can make difference in 1st and 2nd version of NodeMCU Development board by their boards design and ESP modules on it.

- In 1st version of NodeMCU Dev Kit v0.9, CH341SER USB to Serial converter is used whereas in 2nd version of NodeMCU Dev Kit v1.0, CP2102 USB to Serial converter is used.
- 1st version uses ESP-12 and 2nd version uses ESP-12E (Enhanced version).
- Extra 6 pins (MTDO, MTDI, SD_3, MTMS, MTCK, SD_2) brought out on ESP-12E version of ESP-12 modules as shown in below figure. Though Quad SPI pins are brought out, they are internally used for flash memory access.
- Also, there is slight antenna design difference in ESP-12 versions like ESP12-E & ESP-12F as shown in below figure.



ESP-12 Modules

ESP8266 Modules Family

We can also take a look at various ESP modules and their FCC approves till now at esp8266 modules family and summary as shown in below image on same page.

Summary Table

Board ID	#Pins	Pitch	Form factor	LEDs	Antenna	Ant.Socket	Shielded	Dimensions mm	Flash Size in Bytes and (bits)
ESP-01	8	0.1"	2×4 DIL	Yes	Etched-on PCB	No	No	14.3 x 24.8	512KB (4Mb) ××
ESP-02	8	0.1"	2×4 notch	No?	None	Yes	No	14.2 x 14.2	512KB (4Mb) ×
ESP-03	14	2mm	2×7 notch	No	Ceramic	No	No	17.3 x 12.1	512KB (4Mb) ×
ESP-04	14	2mm	2×4 notch	No?	None	No	No	14.7 x 12.1	512KB (4Mb) ×
ESP-05	5	0.1"	1×5 SIL	No	None	Yes	No	14.2 x 14.2	512KB (4Mb) ×
ESP-06	12+GND	misc	4×3 dice	No	None	No	Yes	16.3 x 13.1	512KB (4Mb) ×
ESP-07	16	2mm	2×8 pinhole	Yes	Ceramic	Yes	Yes	21.2 x 16.0	1MB (8Mb) ××
ESP-07S	16	2mm	2×8 pinhole	No	None	Yes	Yes	17.0 x 16.0	4MB (32Mb)
ESP-08	14	2mm	2×7 notch	No	None	No	Yes	17.0 x 16.0	?? *(please fill if you know)*
ESP-08 New	16	2mm	2×8 notch	No	None	No	Yes	18.0 x 16.0	?? *(please fill if you know)*
ESP-09	12+GND	misc	4×3 dice	No	None	No	No	10.0 x 10.0	1MB (8Mb)
ESP-10	5	2mm ??	1×5 notch	No	None	No	No	14.2 x 10.0	512KB (4Mb) *
ESP-11	8	1.27mm	1×8 pinhole	No?	Ceramic	No	No	17.3 x 12.1	512KB (4Mb) *
ESP-12	16	2mm	2×8 notch	Yes	Etched-on PCB	No	Yes	24.0 x 16.0	4MB (32Mb) ??
ESP-12F	22	2mm	2×8 notch	Yes	Etched-on PCB	No	Yes	24.0 x 16.0	4MB (32Mb)
ESP-12E	22	2mm	2×8 notch	Yes	Etched-on PCB	No	Yes	24.0 x 16.0	4MB (32Mb)
ESP-12S	16	2mm	2×8 notch	Yes	Etched-on PCB	No	Yes	24.0 x 16.0	4MB (32Mb)
ESP-13	18	1.5mm	2×9	?	Etched-on PCB	No	Yes	20.0 x 19.9	4MB (32Mb)
ESP-14	22	2mm	2×8 + 6	1	Etched-on PCB	No	Yes	24.3 x 16.2	?? *(please fill if you know)*
ESP-201	22+4	0.1"	2×11 + 4	2	Etched-on PCB ×××	Yes	No	33.5 x 25.5	512KB (4Mb)
WROOM-02	18	1.5mm	2×9	No	Etched on PCB	No	Yes	20.0 x 18.0	?? *(please fill if you know)*
WT8266-S1	18	1.5mm	3×6	1	Etched on PCB	No	Yes	15.0 x 18.6	4MB (32Mb)

× New firmwares can only be flashed on boards with at least 1MB (8Mb) flash. ×× May be different on different editions of the board. ××× Antenna connector is connected by default, to use PCB antenna switch (solder) the 0Ω resistor to the corresponding position.

NodeMCU Dev Kit in Markets

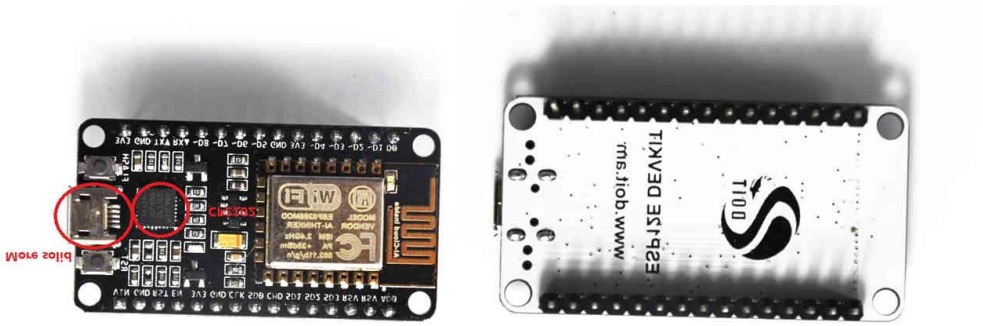
NodeMCU hardware is open source, anyone can edit/modify/produce it and market their modified NodeMCU Development boards. Generally, we can see NodeMCU Dev boards of

Amica

DOIT

Lolin & D1 mini /Wemos etc. in market.

Amica produces NodeMCU ESP8266 Development Boards v1.0(Version2) with designed hardware specifications. Most of the V2 boards are produced by Amica.

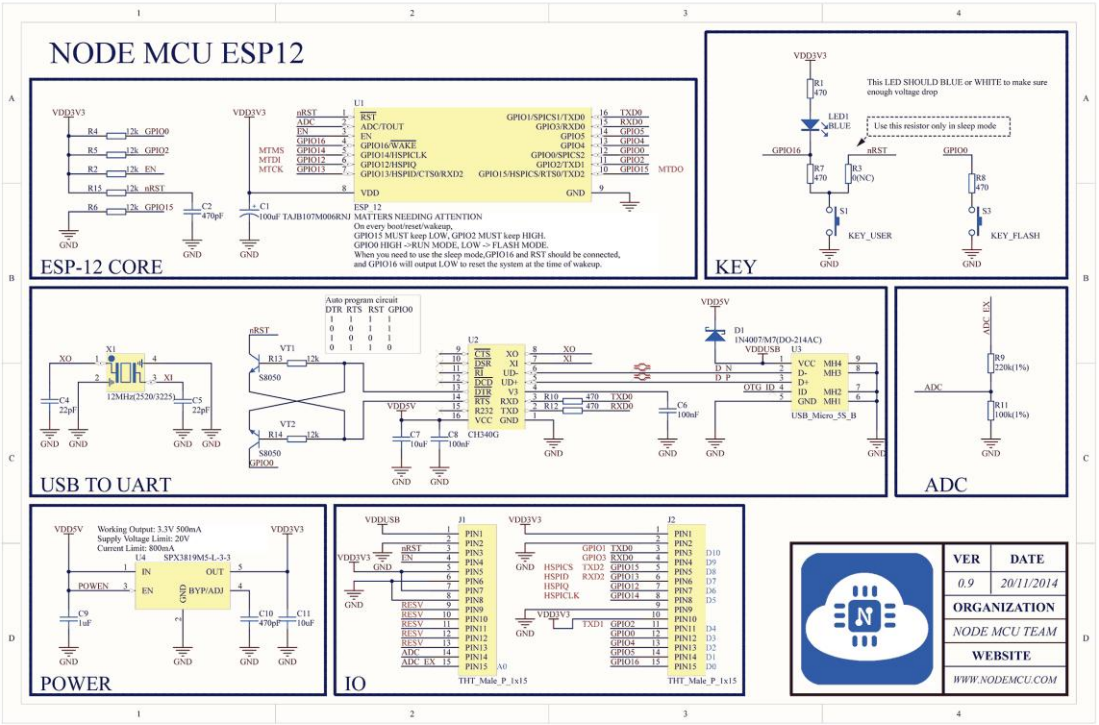


DOIT NodeMCU Dev Kit v1.0(Version2)

NODEMCU HARDWARE SPECIFICATIONS

We can see the NodeMCU Dev Kit v0.9 hardware specifications and design from below link that is open for all.

NodeMCU DevKit v0.9



NodeMCU Dev Kit v0.9 schematic

Also, NodeMCU Dev Kit v1.0 hardware specifications and design is given in below link

NodeMCU DevKit v1.0

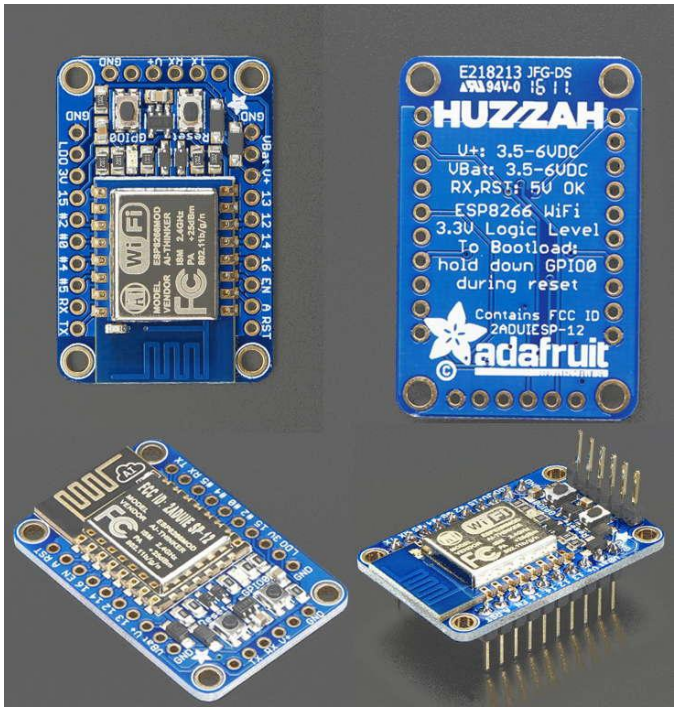
Note: - the ADC block in both version uses resistor divider network (220K and 100K) to scale ESP8266 ADC input voltage range of 0-1V to 0-3.3V. Since the input analog voltage range for ADC pin of ESP8266 is 0–1.0V (while reading external analog voltage), NodeMCU Dev boards uses this resistor divider network to scale it up to 0-3.3V.

ESP8266 Development Boards

Also, we can see ESP8266 development boards/breakouts that are different in design than above NodeMCU Development boards and we can use them with NodeMCU firmware for IoT applications.

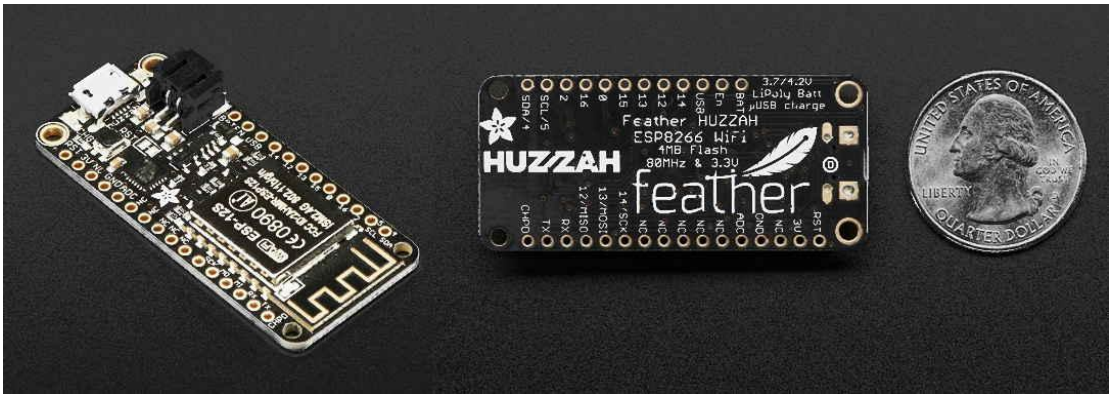
Below are those alternative boards for NodeMCU with different size, pinouts and specifications that are available in market.

Adafruit Huzzah



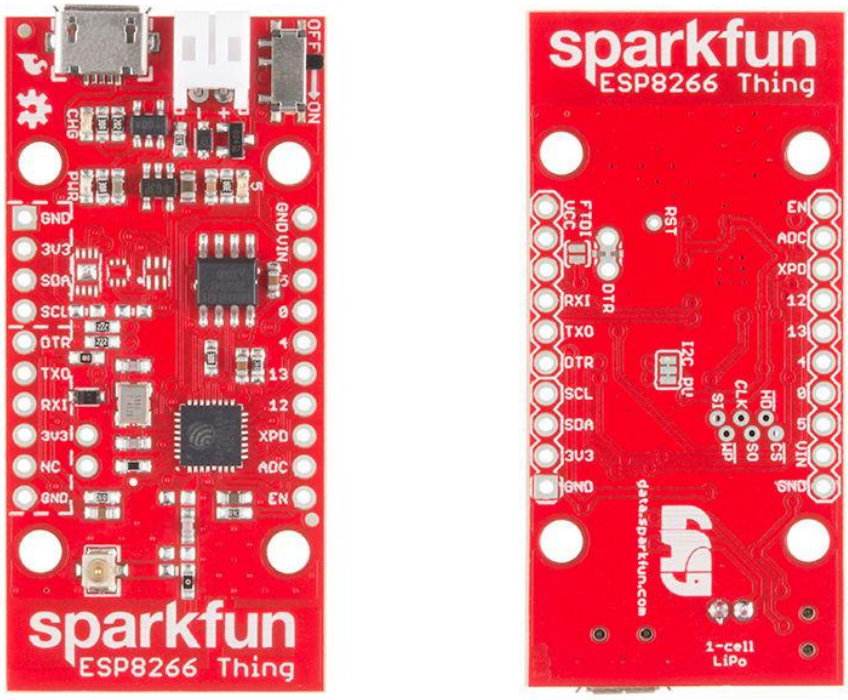
Adafruit HUZZAH ESP8266 Breakout

Adafruit feather Huzzah



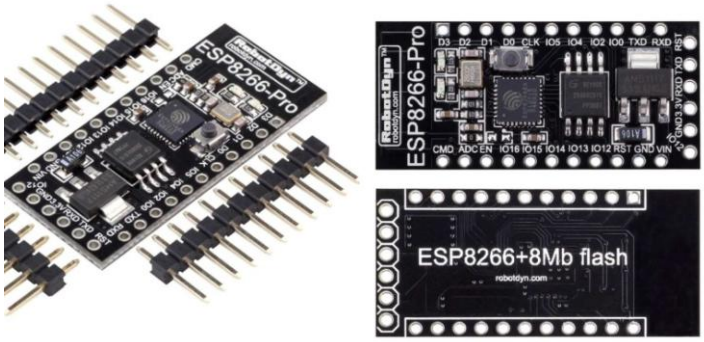
Adafruit Feather HUZZAH with ESP8266 WiFi

Sparkfun thing



SparkFun ESP8266 Thing

RobotDyn NodeM ESP8266

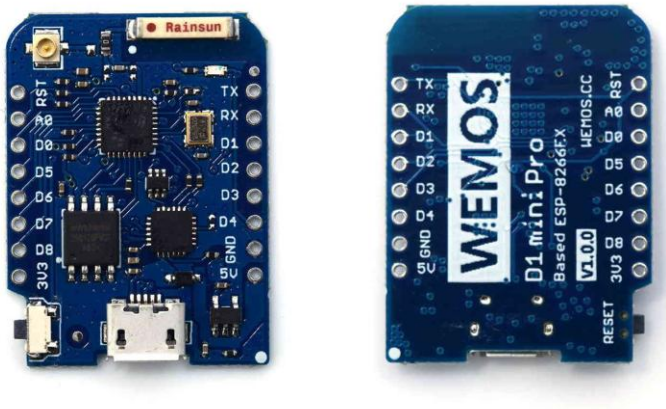


RobotDyn WIFI NodeM ESP8266

WeMos D1 Mini

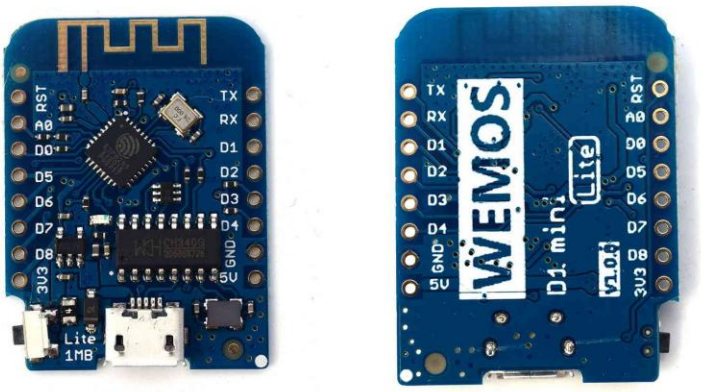
WeMos D1 mini

WeMos D1 mini Pro



WeMos D1 mini Pro

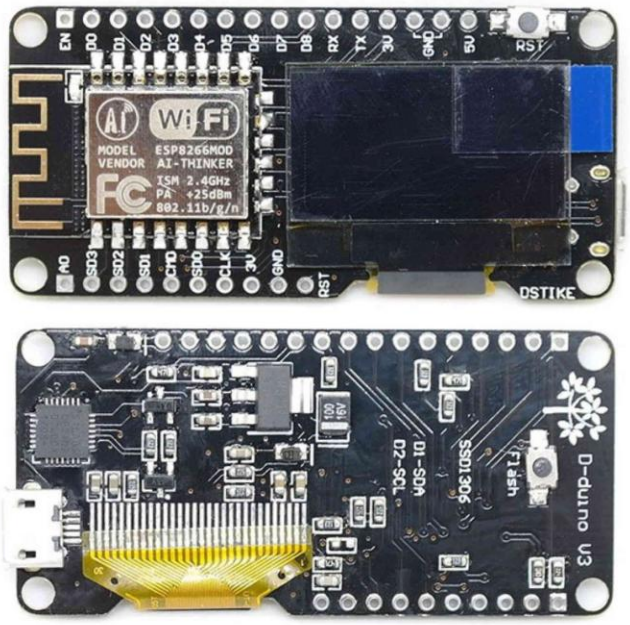
WeMos D1 mini Lite



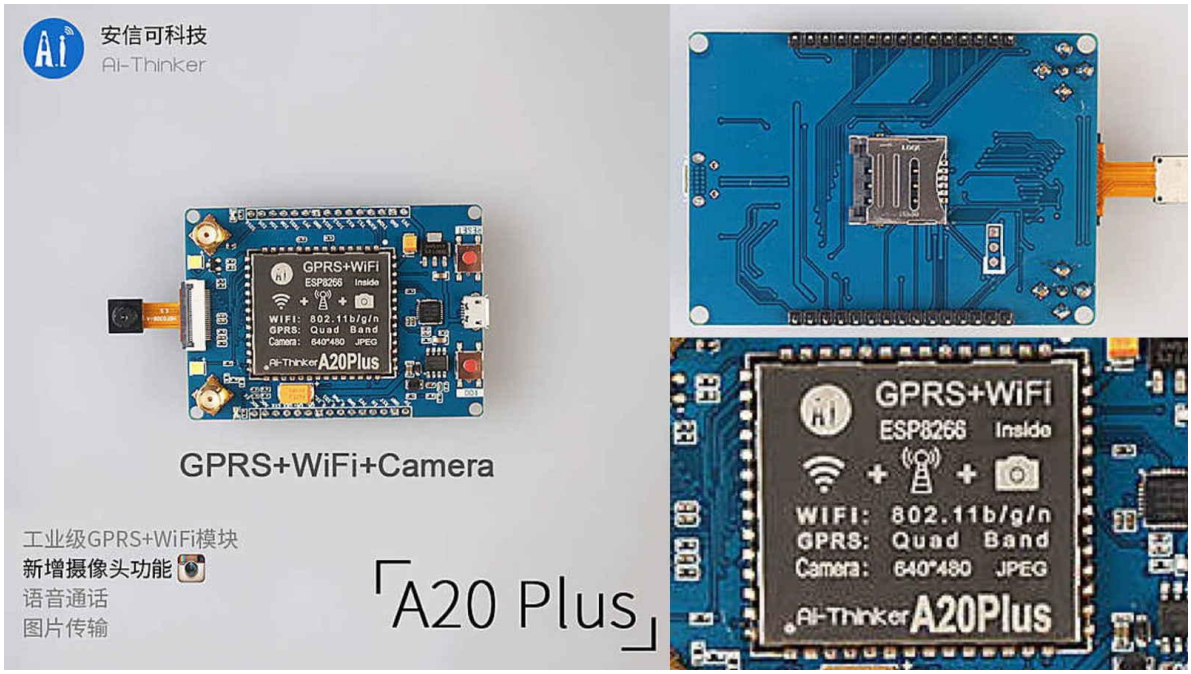
WeMos D1 mini Lite

There are also modified designs available that are application based boards. For example,

- D-duino V3 board which comes with OLED display on board.
- AI-Thinker A20 Plus board comes with GPRS+Camera feature on its board.



D-duino V3 ESP8266 Dev Kit



AI-Thinker A20 Plus GPRS + WiFi + Camera

Here, we can say that there is no unique NodeMCU Development board design in market. If we came across their official boards then we can realize that Amica boards are looks like an official version whereas others not (since they are applications wise designed).

Amica provided some points regarding improving their development boards on their twitter page as shown in below image.