

Table of Contents

- Custom NodeMCU Builder** 1
- Tasmota** 1
- Basic Info** 1
- Power consumption ESP8266** 1
- What is possible on ESP8266?** 1
- ESP8266 NodeMCU Pinout** 2
- Wifi Deauthentication** 3
- Lua Sources** 3
- Turn internal LED on/off** 3
- HTTP GET** 4

Custom NodeMCU Builder

nodemcu-build.com

Tasmota

<https://tasmota.github.io>

Basic Info

Type	Arduino Uno WiFi	ESP8266 NodeMCU	ESP 32
Digital I/O	13	11	
Analog I/O	6	1	
PWM	6	10	
Frequency	16MHz	80MHz or 160MHz	160MHz or 240MHz
I/O Voltage	5V	3,3V	3,3V
Flash size	32kB (-0,5kB bootloader)	4096kB	
RAM	8kB	96kB	520kB
GPIO Current	40mA	12mA	

Power consumption ESP8266

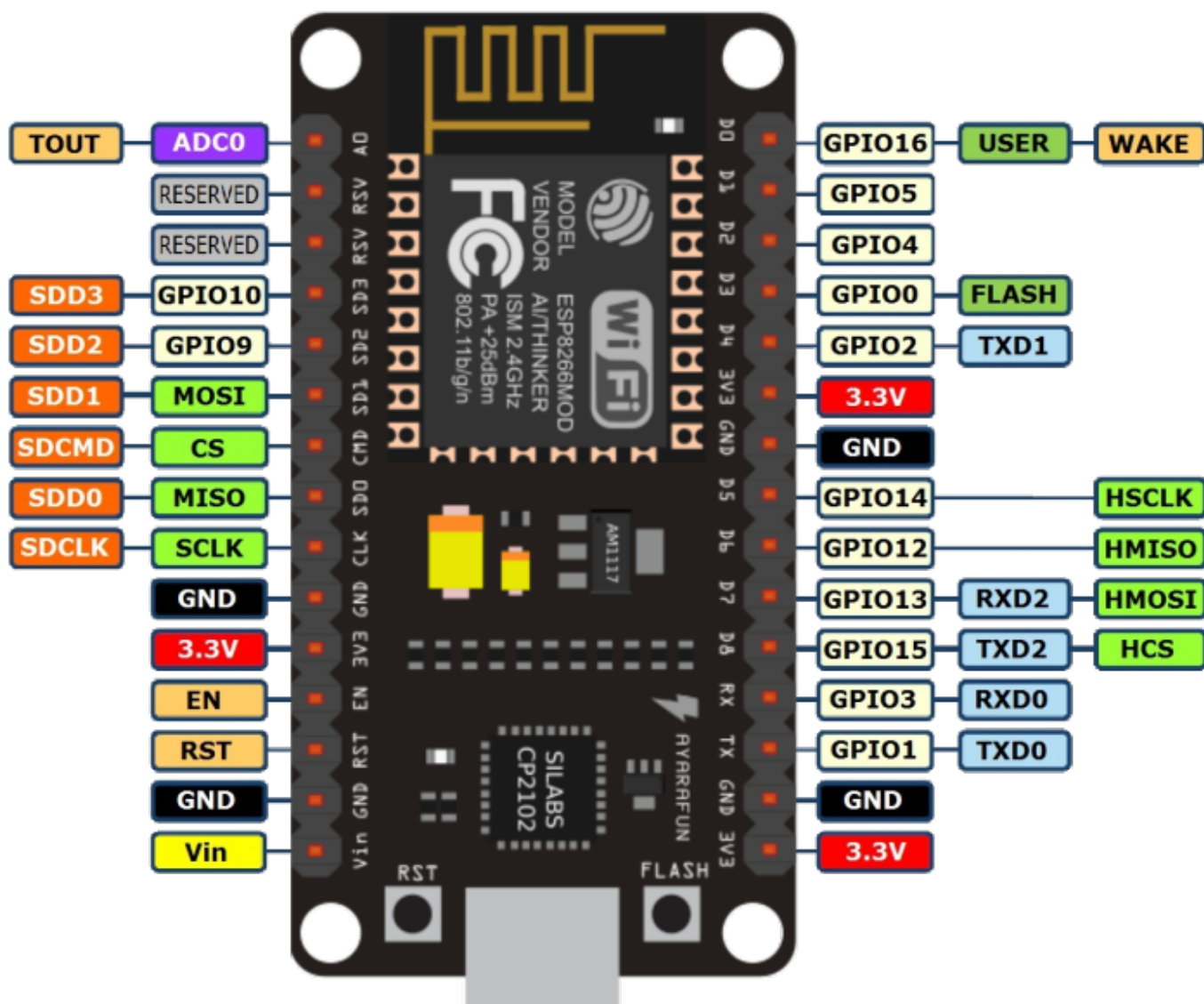
Mode	Current
Wifi Send	120mA
Wifi Receive	56mA
Modem Sleep	15mA
Deep Sleep	10µA*

* Deep sleep ~300s and waking up to connect to the AP (taking about 0.3~1s)

What is possible on ESP8266?

Wifi 2.4 GHz b/g/n WPA2
Internal Analog-Digital-Converter (0-1V)
HX711 - inexpensive 24bit ADC
Read battery voltage
AM2320 humidity and temperature sensor - i2c interface
BME280/BMP280 temperature/air pressure/humidity sensor
ADXL345 triple axis accelerometer
L3G4200D three axis digital gyroscope
BMP085/180 temperature and pressure sensor
HMC5883L three axis digital compass
Audio output 16KHz / mono / 8Bit
Switec X.27 instrument stepper motor
TSL2561 - check illuminance in lux
Serial line communication

ESP8266 NodeMCU Pinout



PIN	IO	Remark	Function
D0	GPIO16	May work! Connected to LED	Wake up from sleep
* D1	GPIO5	OK	
* D2	GPIO4	OK	
D3	GPIO0	Don't use! Turn flashing	
D4	GPIO2	Don't use! Conflict while booting	
* D5	GPIO14	OK	
* D6	GPIO12	OK	
D7	GPIO13	May work!	
D8	GPIO15	Don't use! Conflict while booting	
SDD2	GPIO9	May work but...	
SDD3	GPIO10	May work but...	
RX	GPIO3	Usable, but conflict with USB converter	
TX	GPIO1	Usable, but conflict with USB converter	

The maximum current that can be drawn from a single GPIO pin is 12mA.

Wifi Deauthentication

https://github.com/spacehuhn/esp8266_deauther

Lua Sources

```
--This file is init.lua
wifi.setmode(wifi.STATION)
wifi.sta.config("janforman.dmz","")

local IDLE_AT_STARTUP_MS = 5000;

tmr.alarm(1,IDLE_AT_STARTUP_MS,0,function()
    dofile("status.lua")
end)
```

```
--This file is status.lua

tmr.alarm(0, 1000*60*10, tmr.ALARM_AUTO, function ()

-- loop
conn = nil
conn=net.createConnection(net.TCP, 0)

conn:on("receive", function(conn, payload)
    success = true
    print(payload)
end)

conn:on("connection", function(conn, payload)
    print('\nConnected')
    conn:send("GET /"..(tmr.time() / 3600)
    .." HTTP/1.1\r\n"
    .."Host: janforman.com\r\n"
    .."Connection: close\r\n"
    .."Accept: */*\r\n"
    .."User-Agent: Mozilla/4.0 (compatible; esp8266 Lua; Windows NT
5.1)\r\n"
    .."\r\n")
end)

conn:on("disconnection", function(conn, payload) print('\nDisconnected') end)
conn:connect(80,'janforman.com')
-- loop

end)
```

Turn internal LED on/off

```
gpio.mode(4,gpio.OUTPUT)
gpio.write(4,gpio.LOW)
```

```
LED_PIN = 4
gpio.mode(LED_PIN,gpio.OUTPUT)
value = true
```

```
tmr.alarm(0,500,tmr.ALARM_AUTO, function()  
  gpio.write(LED_PIN, value and gpio.HIGH or gpio.LOW)  
  value = not value  
end)
```

HTTP GET

```
http.get("http://janforman.com/",nil,function(code,data)  
  if(code < 0) then print ("Failed")  
  else print(code,data)  
  end  
end)
```

From:

<https://wiki.janforman.com/> - **wiki.janforman.com**

Permanent link:

<https://wiki.janforman.com/iot>

Last update: **2021/05/10 22:26**

