

ESP 8266 tutorial

EurOpen 2016, Jindřichův Hradec

Komunitní HW

Arduino, Raspberry Pi, ESP8266

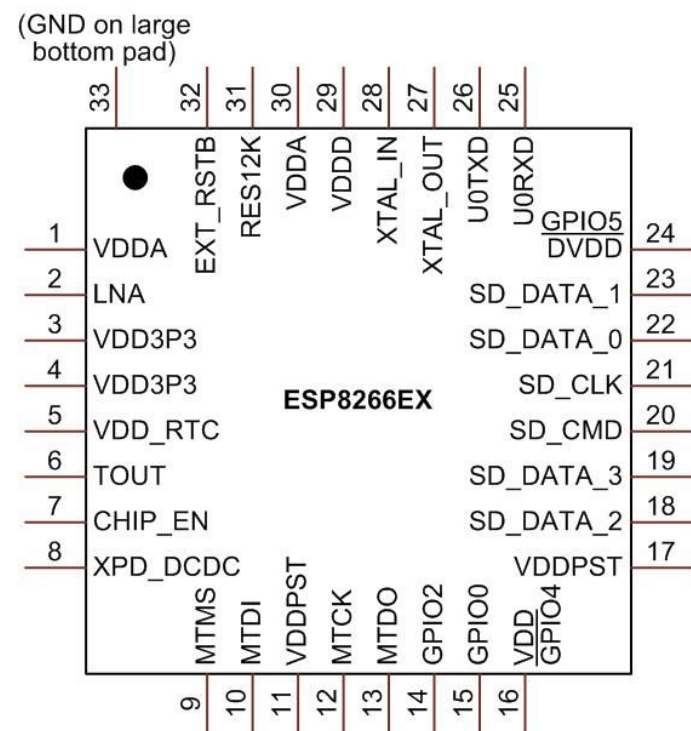
- základ úspěchu
 - funkční návrh HW
 - použitelné multiplatformní IDE
 - GPIO
 - podpora protokolů (SPI, 1wire, ...)

Důležitá je komunita kolem projektu:

- adafruit, github, instructables a další konkrétně zaměřené weby se spoustou návodů

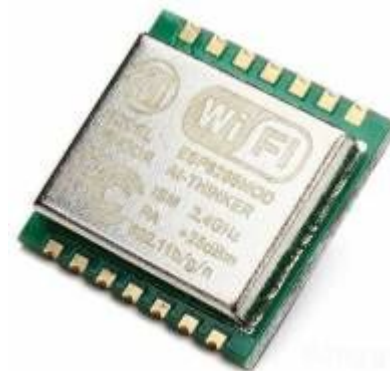
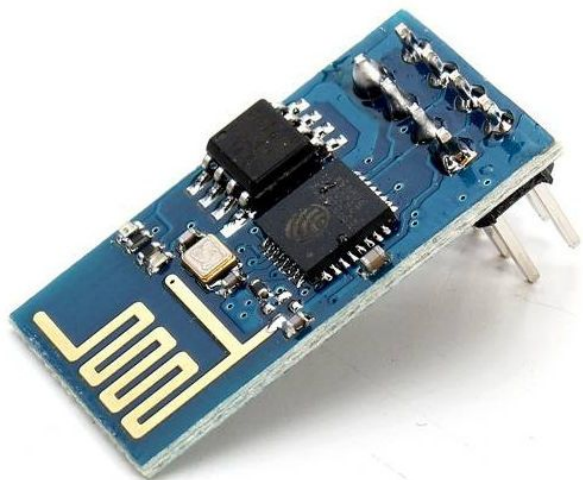
Základní vlastnosti ESP8266

- 3.3 V napájení
- low power 32bit procesor 80-160MHz
- Flash 512kB-16MB
- protokol 802.11 b/g/n (2.4GHz)
- 17 GPIO včetně sériové linky a podpory různých protokolů
- 1 A/D
- vše na čipu 5x5mm,
- minimum externích součástek



Přehled modulů ESP8266

Kompletní seznam: <http://www.esp8266.com/wiki/doku.php?id=esp8266-module-family>

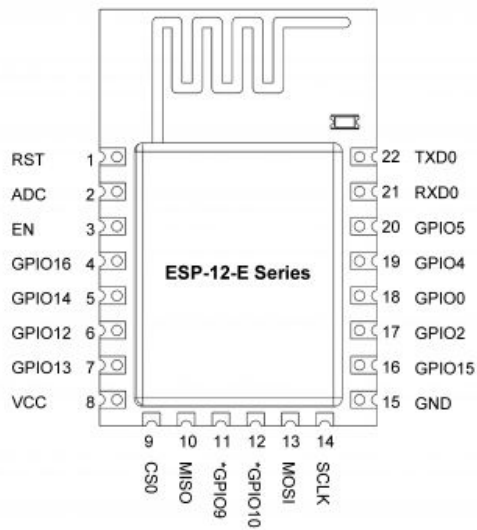


ESP12-E a NodeMCU DEVKIT V1.0

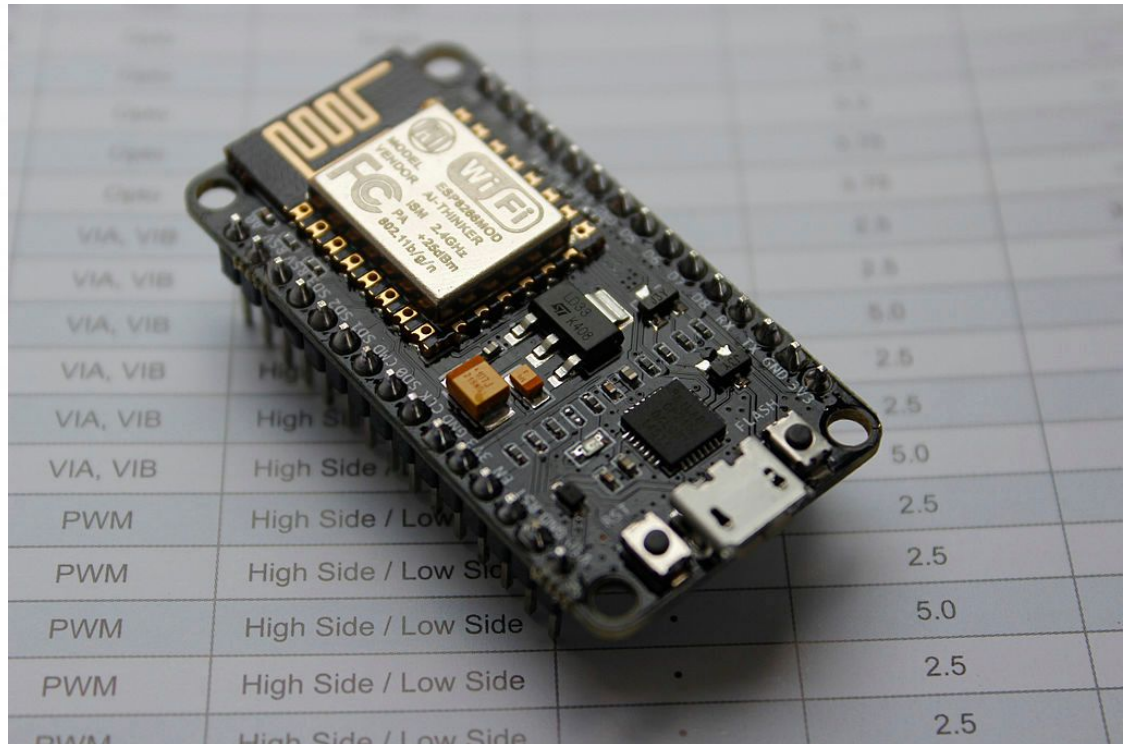
ESP-12-E Series, ESP-12-D/ESP-12-Q 32Mbit

AI-THINKER

Pin Configuration and Functions

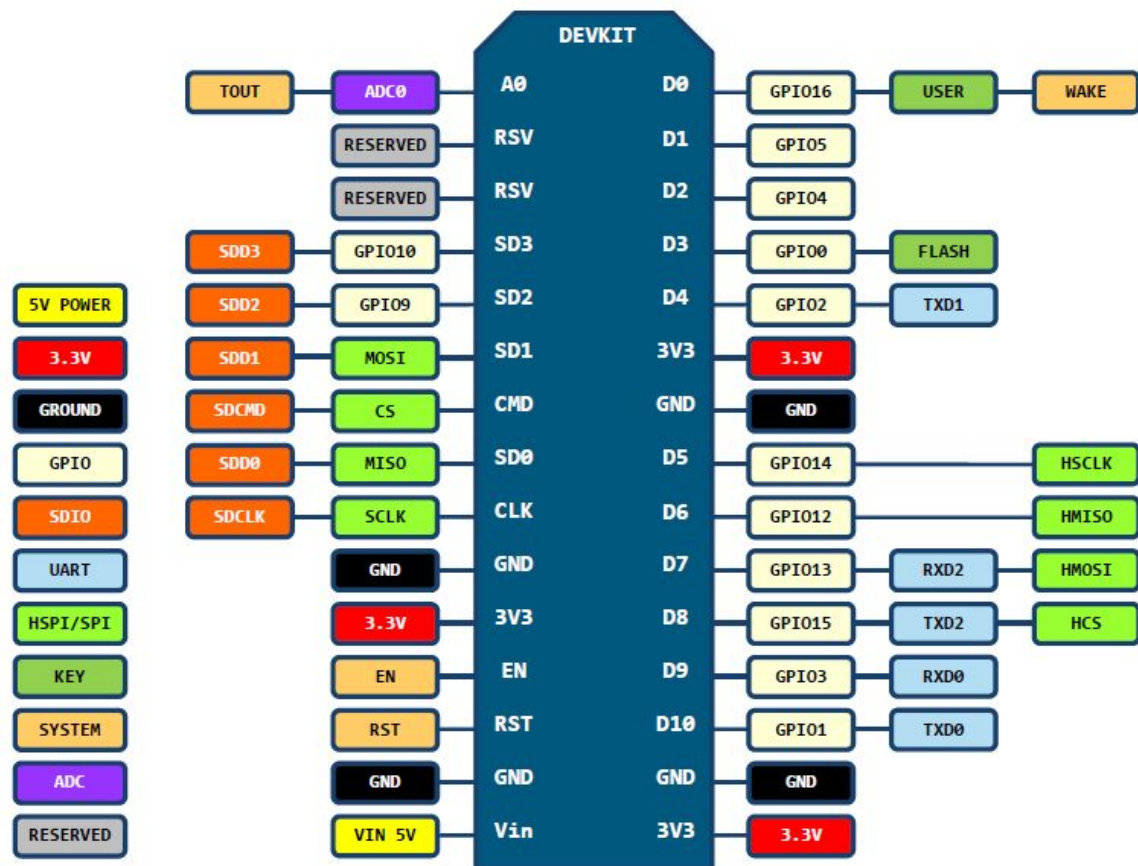


* Can only be used on ESP12-D.



ESP piny

PIN DEFINITION



D0(GPIO16) can only be used as gpio read/write, no interrupt supported, no pwm/i2c/ow supported.

Firmware

- AT command interface <https://github.com/esp8266/at-command-set>
- ESP Easy - web based multifunction sensor device <http://www.esp8266.nu/index.php/ESPEasy>
- Arduino http://www.esp8266.nu/index.php/Tutorial_Arduino_Firmware_Upload
- MicroPython <https://github.com/micropython/micropython/tree/master/esp8266>
- Espruino JavaScript for Microcontrollers <http://www.espruino.com/>
- NodeLua <https://nodelua.org/>
- NodeMCU http://www.nodemcu.com/index_en.html

Flash firmware

GUI

- nodemcu-flasher <https://github.com/nodemcu/nodemcu-flasher>
- esp8266_flasher.exe <http://www.xess.com/blog/esp8266-reflash/>
- Arduino IDE http://www.esp8266.nu/index.php/Tutorial_Arduino_Firmware_Upload

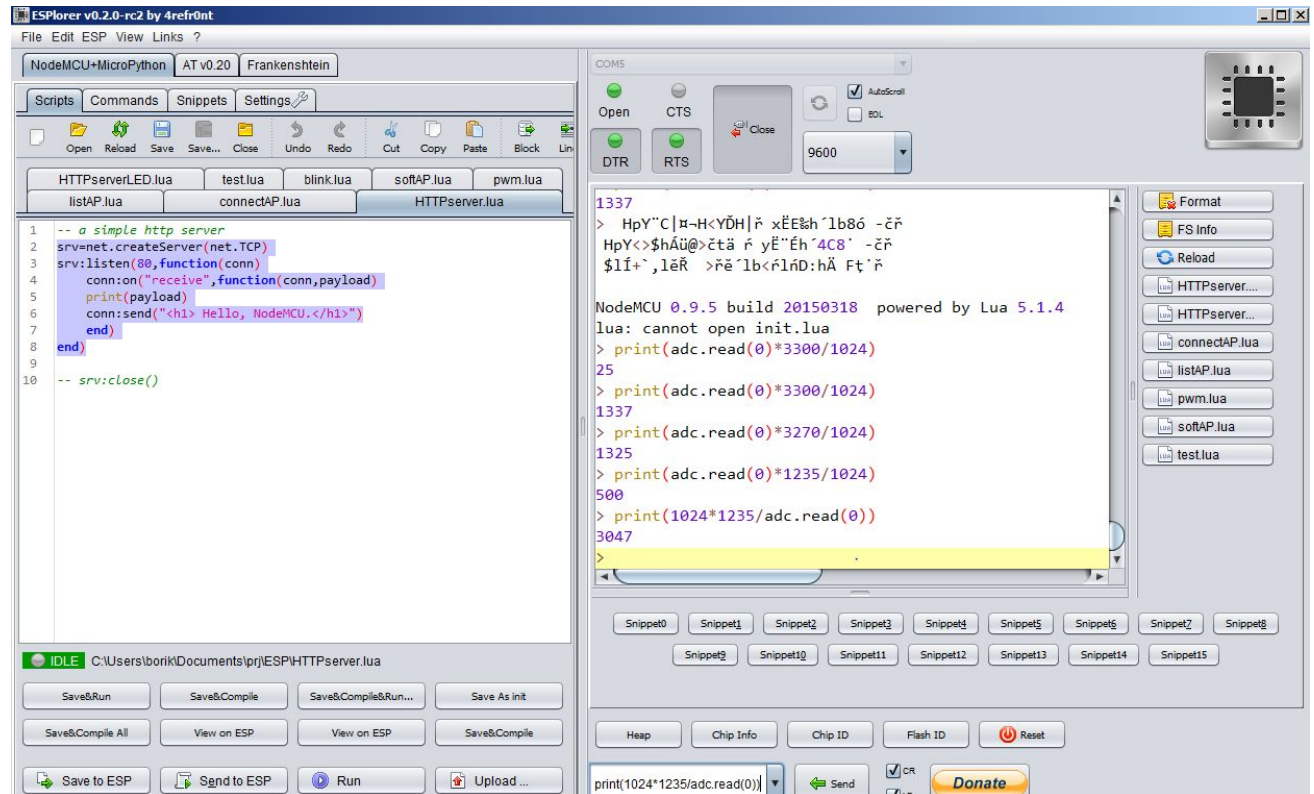
řádkové utility

- esptool-ck <https://github.com/igrr/esptool-ck>
- esptool.py <https://github.com/themadinventor/esptool>

IDE pro ESP

Arduino IDE: <https://github.com/esp8266/Arduino>

ESPLOERER: <https://github.com/4refr0nt/ESPlorer>



Moduly NodeMCU

- systém: node, tmr, perf
 - matematické a logické operce: bit, crypto, encoder
 - porty: adc, gpio, pwm
 - protokoly a senzory: am2320, bme280, bmp085, dht, spi, i2c, ow (1wire)...
 - síťování a komunikace: net, wifi, http, sntp, uart
 - ovládání RGB LED: ws2801, ws2812, apa102
 - grafické displeje SPI, I2C: u8g
-
- podrobně: <http://nodemcu.readthedocs.io/en/dev/>

Jazyk Lua

- “normální” procedurální skriptovací jazyk
- vznikl v Brazílii 1993
- minimální počet datových typů, vše složitější je implementováno pomocí tabulky
- příkazy se oddělují bílými znaky nebo středníkem
- konec bloku: end
- increment/decrement operator není
- nerovnost ~=
- import knihovny: require ("bmp180")

```
for i=1,10 do
    if i==5 then
        print ("pulka")
    else
        print (i)
    end
end
```

Manuál: <https://www.lua.org/manual/5.3/>

Blik & Píp

```
LED_PIN = 4
gpio.mode(LED_PIN, gpio.OUTPUT)
gpio.write(LED_PIN, gpio.HIGH) --
gpio.LOW
```

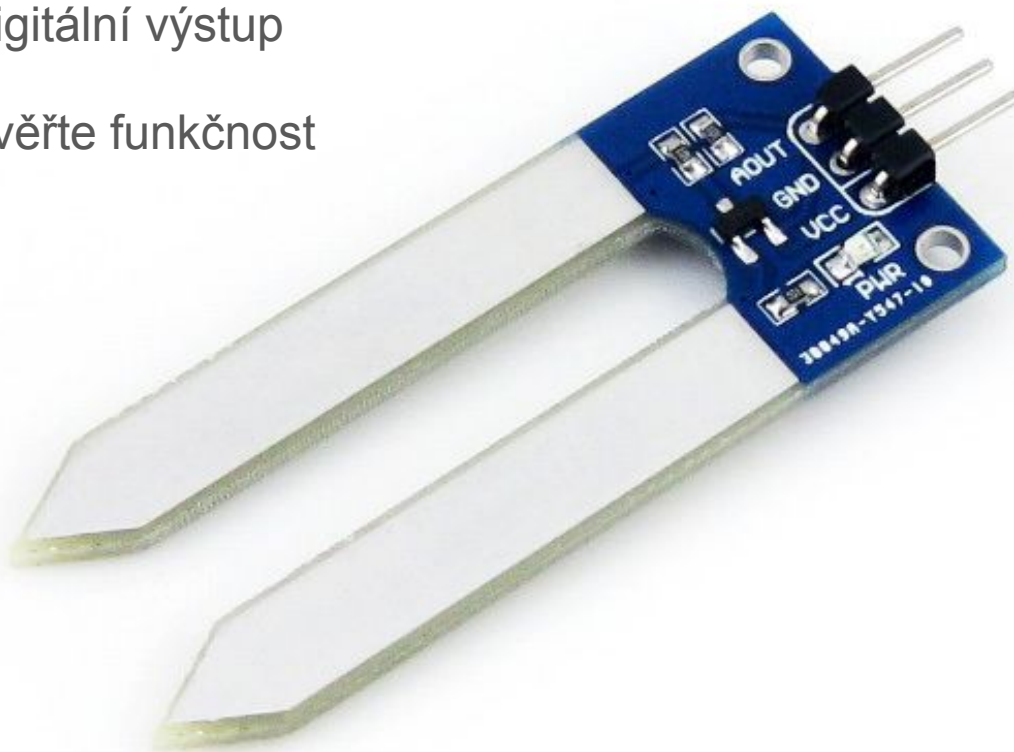


```
LED_PIN = 4
--gpio.mode(LED_PIN, gpio.OUTPUT)
pwm.setup(LED_PIN,2,1023) -- port,clock,duty
pwm.start(LED_PIN)
pwm.setduty(LED_PIN,100)
```

Vlhkost půdy - Soil Moisture Sensor

- měří odpor mezi elektrodami
- pevně nastavená citlivost
- pouze dvoustavový digitální výstup

Úkol: zapojte detektor a ověřte funkčnost



Analogový vstup

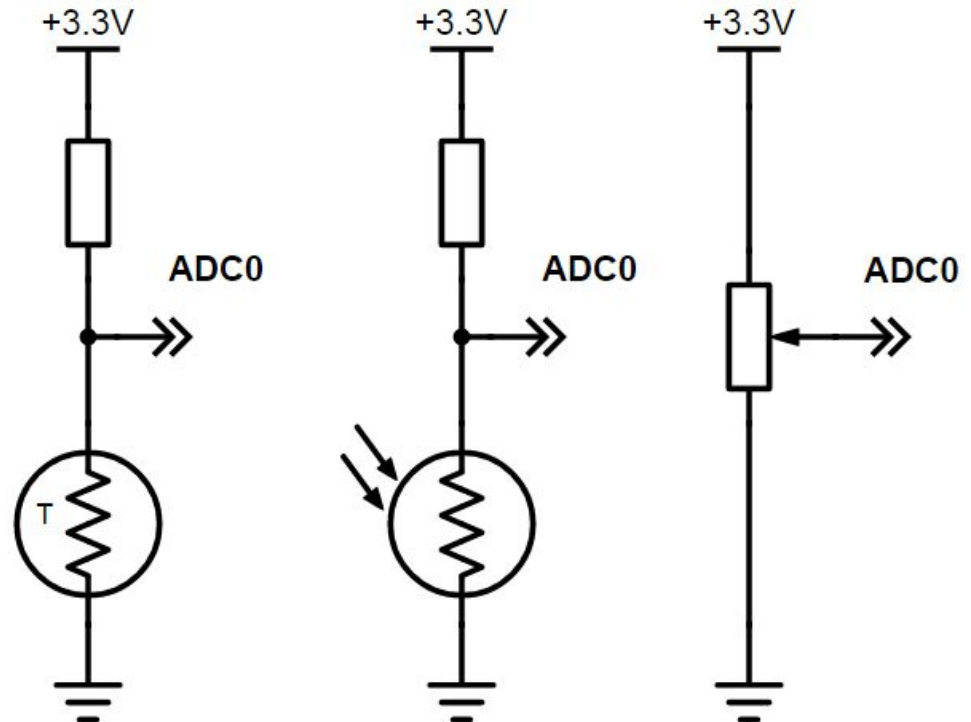
- pouze jeden vstup ADC0
- převodník 10 bitů
- $V_{ref}=V_{cc}$

programování:

```
print(adc.read(0))
```

Úkol: zapojte LED, aby trvale svítila
a změřte její napětí.

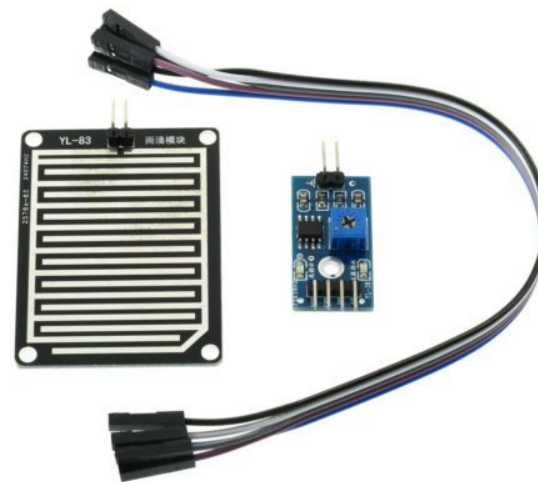
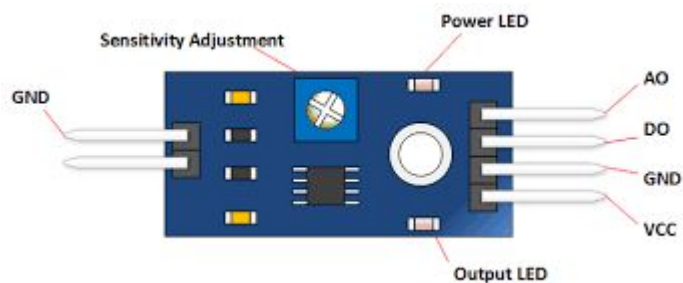
Nezapomeňte na rezistor :-)



Dešťový senzor - Raindrops Detection Sensor

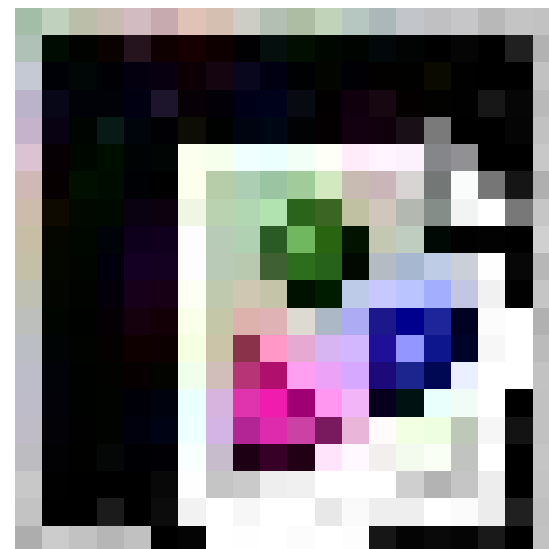
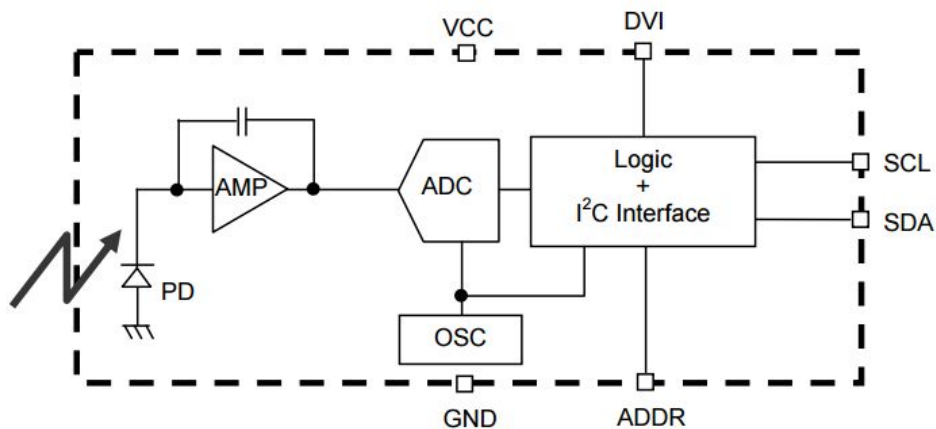
- měří odpor mezi dvěma elektrodami
- nastavení citlivosti trimrem
- analogový i digitální výstup

Úkol: otestujte senzor - analogový i digitální výstup



Osvětlení I2C: BH1750FVI

BH1750FVI is an digital Ambient Light Sensor IC for I2C bus interface.



BH1750FVI - software

čínsky: <http://bbs.nodemcu.com/t/bh1750-lua-for-esp8266/165>

holandsky:

<http://domoticx.com/esp8266-wifi-lichtintensiteit-sensor-bh1750-gy-302-nodemcu/>

normálně:

https://github.com/nodemcu/nodemcu-firmware/tree/master/lua_modules/bh1750

Úkol: sestavte obvod a program pro měření intenzity světla

zapojit GND, VCC, SDA=PIN6, SCL=PIN5

bh1750.lua, lux.lua

Tlak a teplota I2C: BMP180

<https://github.com/javieryanez/nodemcu-modules/tree/master/bmp180>

```
OSS = 1 -- oversampling setting

SDA_PIN = 4 -- sda pin, GPIO2
SCL_PIN = 3 -- scl pin, GPIO0

bmp180 = require("bmp180")
bmp180.init(SDA_PIN, SCL_PIN)
bmp180.read(OSS)
t = bmp180.getTemperature()
p = bmp180.getPressure()
```

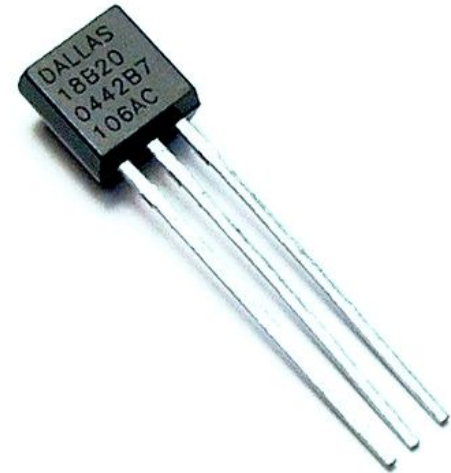


Teplota 1-wire: DS18B20

http://www.nodemcu.com/index_en.html#fr_54747661d775ef1a3600009e

https://github.com/nodemcu/nodemcu-firmware/tree/master/lua_modules/ds18b20

```
-- read temperature with DS18B20
t=require("ds18b20")
t.setup(9)
addrs=t.addrs()
-- Total DS18B20 numbers, assume
it is 2
print(table.getn(addrs))
-- The first DS18B20
print(t.read(addrs[1],t.C))
print(t.read(addrs[1],t.F)) print
(t.read(addrs[1],t.K))
-- The second DS18B20 print(t.
read(addrs[2],t.C))
...
```



Vlhkost a teplota: DHT11, DHT22

<https://github.com/javieryanez/nodemcu-modules/tree/master/dht22>

https://github.com/nodemcu/nodemcu-firmware/tree/master/luam_modules/dht_lib

```
dht22 = require("dht22")  
dht22.read(PIN)  
t = dht22.getTemperature()  
h = dht22.getHumidity()
```



WiFi modul

- **wifi** for overall WiFi configuration
- **wifi.sta** for station mode functions
- **wifi.ap** for wireless access point (WAP or simply AP) functions
- **wifi.ap.dhcp** for DHCP server control
- **wifi.eventmon** for wifi event monitor

Podrobně: <http://nodemcu.readthedocs.io/en/dev/en/modules/wifi/>

Načtení WiFi sítě

listAP.lua:

```
function listap(t)
    for k,v in pairs(t) do
        print(k.." : "..v)
    end
end

wifi.sta.getap(listap)
```

<https://nodemcu.readthedocs.io/en/dev/en/modules/wifi/#wifistagetap>

Stanice v síti

connectAP.lua:

```
wifi.setmode(wifi.STATION)
wifi.sta.setmac(mac) -- !!!
wifi.sta.config("esp", "europen2016")
print(wifi.sta.getip())
```

<https://nodemcu.readthedocs.io/en/dev/en/modules/wifi/#wifista-module>

ESP jako AP

```
wifi.setmode(wifi.SOFTAP)

cfg={ssid="ESPtest99",pwd="20162016"} -- pozor, musi byt
dlouhe heslo!

wifi.ap.config(cfg);

ip_cfg = {}

ip_cfg.ip="192.168.7.1"

ip_cfg.netmask="255.255.255.0"

ip_cfg.gateway="192.168.7.1"

wifi.ap.setip(ip_cfg)
```


HTTP server

HTTPserver.lua, HTTPserverLED.lua

```
srv=net.createServer(net.TCP)
srv:listen(80,function(conn)
    conn:on("receive",function(conn,payload)
        print(payload)
        conn:send("<h1> Hello, NodeMCU.</h1>")
    end)
end)
```

<https://nodemcu.readthedocs.io/en/dev/en/modules/net/#netcreateserver>

HTTP klient

```
-- A simple http client
conn=net.createConnection(net.TCP, false) conn:on
("receive", function(conn, pl)
    print(pl)
end)
conn:connect(80, "121.41.33.127")
conn:send("GET / HTTP/1.1\r\nHost: www.nodemcu.com\r\n"
.."Connection: keep-alive\r\nAccept: */*\r\n\r\n")
```

<https://nodemcu.readthedocs.io/en/dev/en/modules/net/#netcreateconnection>

Autonomní napájení

použitelný rozsah napájení:

- cca 2.3 - 4.7V
- ověřeno experimentálně, dokumentace tyto údaje neuvádí

<http://forum.makehackvoid.com/t/esp8266-operating-voltage-range-and-sleep-current/286>

možné napájecí zdroje:

- vyšší napětí a stabilizátor = 3.3V
- 2*alkaline 1.5V = 3V (2.6 - 3.0V)
- 2*NiCd 1.2V = 2.4V (2.2 - 2.6V)
- 3*NiCd 1.2V = 3.6V (3.3 - 3.9V)
- Li-Ion = 3.7V (3.2 - 4.2V)

úsporné provozní režimy:

- modem sleep
- light sleep - CPU suspended
- deep sleep - CPU off a čítač aktivuje reset

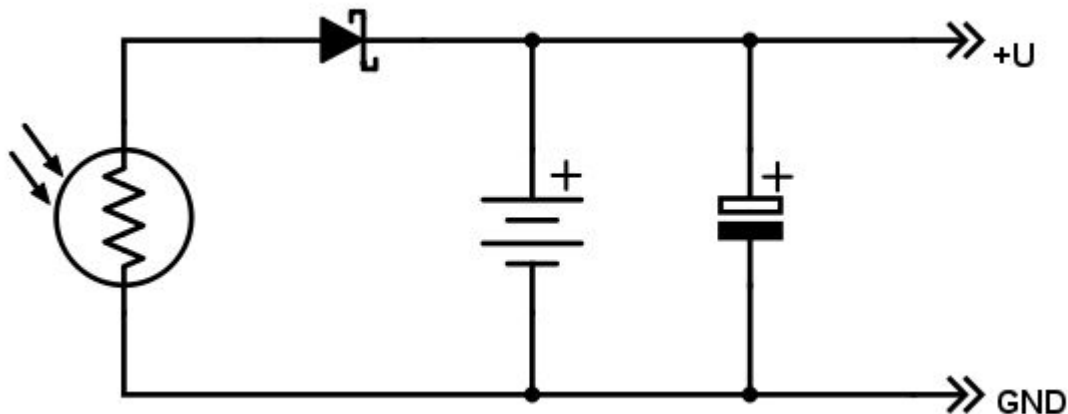
proudový odběr:

- wifi v akci: 60-200 mA
- modem sleep: 15 mA
- light sleep: 0.5 mA
- deep_sleep: 10 μ A

<http://bbs.espressif.com/viewtopic.php?t=133> http://www.esp8266.com/wiki/doku.php?id=esp8266_power_usage

Solární nabíječka

- solární článěk může dobíjet akumulátor - jak řídit nabíjení?
- Ni-Cd a Ni-Mh vydrží trvalé přebíjení do 0.1 kapacity (2500mAh - až 250mA)
- jednoduchá nabíječka - stačí oddělovací dioda (nejlépe Schottkyho)
- námět: <http://www.evilmadscientist.com/2008/simple-solar-circuits/>
- např. pro 3*NiCd akumulátory vyhovuje solární článěk 6V/200mA
- nelze použít pro Li-Ion baterie - nesnášejí přebíjení!
- doplnit napájení kondenzátorem 100M a větším - pokrývá proudové špičky
- ochrana akumulátorů před úplným vybitím
 - zařízení se musí včas vypnout
 - je třeba měřit napětí zdroje



Kontrola napájecího napětí

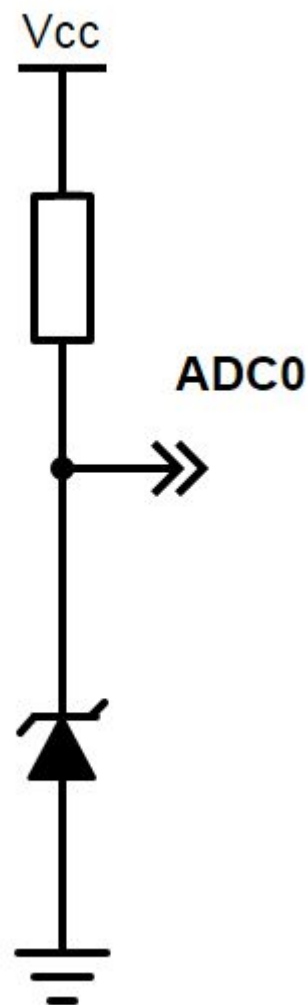
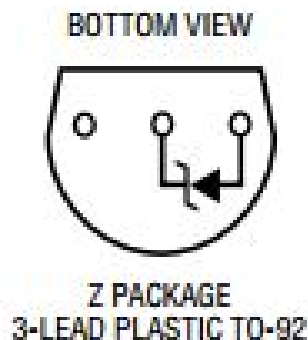
obrácené měření:

- vstupní měřené napětí je konstantní
- referenční (napájecí) napětí kolísá

úkol: změřte hodnotu napájecího napětí

použijte:

- stabilizátor LM385 (1,235V při 10uA-20mA)
- rezistor 47k



Dotazy, komentáře, volná zábava...

Jiří Bořík, jiri@borik.cz