

# Urządzenia, czujniki, schematy, sposób programowania

Wykonał: Arkadiusz Cichy

Sprawdził: Szymon Cichy

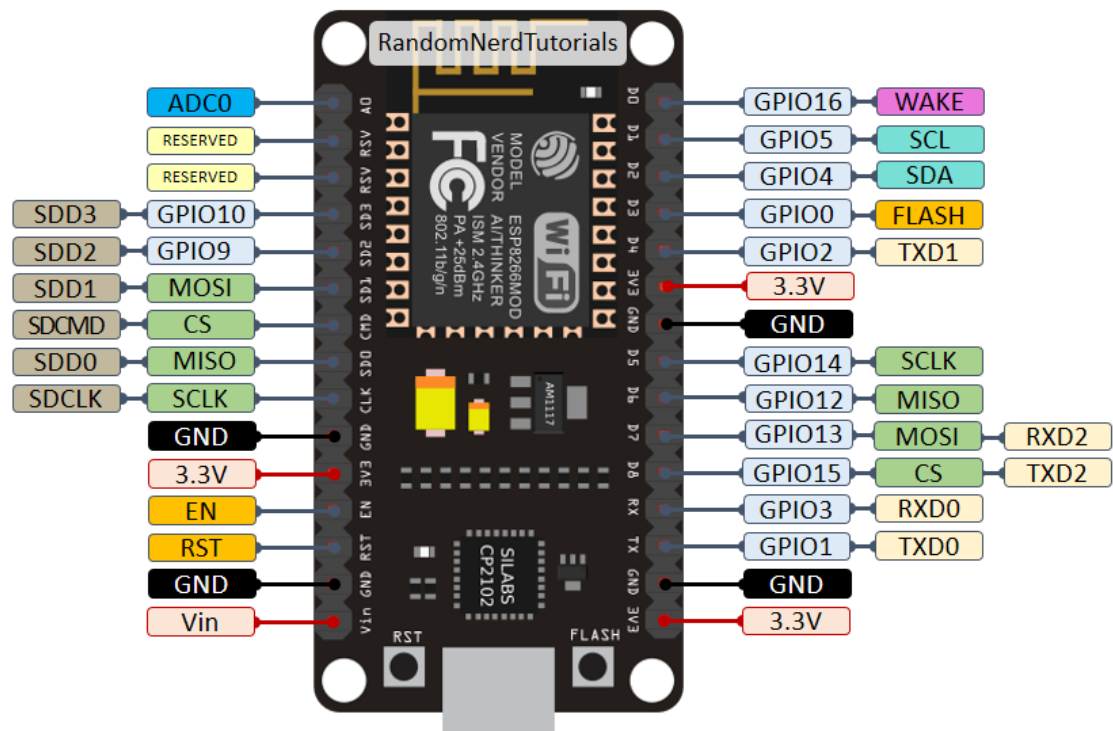
Zatwierdził: Adam Krizar

## 1. Wybrane urządzenia/czujniki/schematy

Założenia projektowe sugerują wybór elektroniki o jak najmniejszym poborze mocy. Zadanie ułatwia fakt, że urządzenia nie muszą mieć dużej mocy obliczeniowej. Jedynym aspektem, który działa na naszą niekorzyść jest poziom skomplikowania programowania/łączenia wybranego sprzętu. Dysponując jedynie taką mocą przerobową, nasze wybory powinny uwzględniać czas nauki obsługi danego sprzętu dodatkowo do czasu zaprogramowania go lub czasu potrzebnego na zbudowanie działającego układu.

### ESP8266

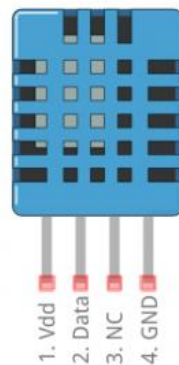
- Komunikacja WiFi:
  - standard 802.11 b/g/n 2,4 GHz,
  - prędkość transmisji do 72,2 Mb/s,
  - zabezpieczenia: WPA/WPA2,
  - szyfrowanie: WEP/TKIP/AES,
  - protokoły: IPv4, TCP/UDP/HTTP.
- Zasilanie:
  - napięcie pracy: 2,5 – 3,6 V,
  - napięcie zasilania: 4,8 – 12 V,
  - średni pobór prądu: 80 mA,
  - maksymalny pobór prądu: 800 mA.
- Aktualizacja oprogramowania:
  - UART,
  - OTA.
- CPU:
  - Tensilica L106 32-bit 80 MHz,
  - obudowa: QFN32-pin (5 mm × 5 mm),
  - interfejsy: UART/SDIO/SPI/I2C/I2S/IR (zdalne sterowanie),
  - dostępne 10 GPIO,
  - 1 wyprowadzenie ADC (0 – 3,3 V).
- Konwerter USB-TTL (UART): CH340.
- Raster wyprowadzeń: 2,54 mm.
- Wymiary modułu: 58 × 30 mm.



Rysunek 1 ESP8266 PinOut

## DHT11

- Ogólne:
  - Napięcie zasilania: 3 V do 5,5 V
  - Pobór prądu: 0,2 mA
  - Częstotliwość próbkowania: 1Hz
- Wbudowany termometr
  - Zakres pomiarowy: 0 - 50 °C
  - Dokładność:  $\pm 2^{\circ}\text{C}$
- Czujnik wilgotności:
  - Zakres pomiarowy: 20 - 95%RH
  - Dokładność:  $\pm 5\%\text{RH}$



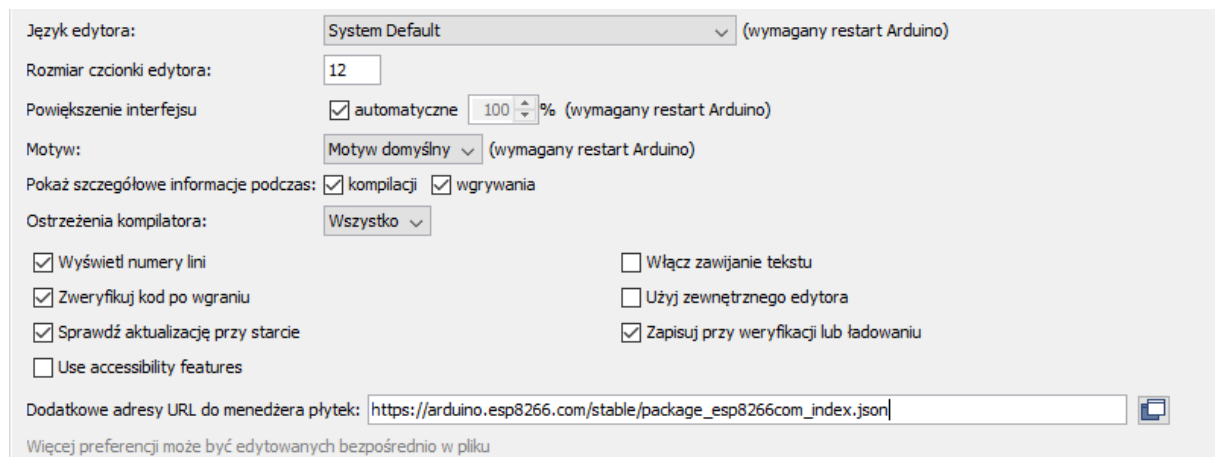
Rysunek 2 DHT11 PinOut

## 2. Sposób programowania

Programowanie ESP8266 przez Arduino IDE jest obecnie najprostszym i najbezpieczniejszym sposobem programowania tego kontrolera. Aby środowisko poprawnie rozpoznało inny kontroler niż Arduino należy pobrać pakiet bibliotek i informacji na temat wybranego przez nas urządzenia.

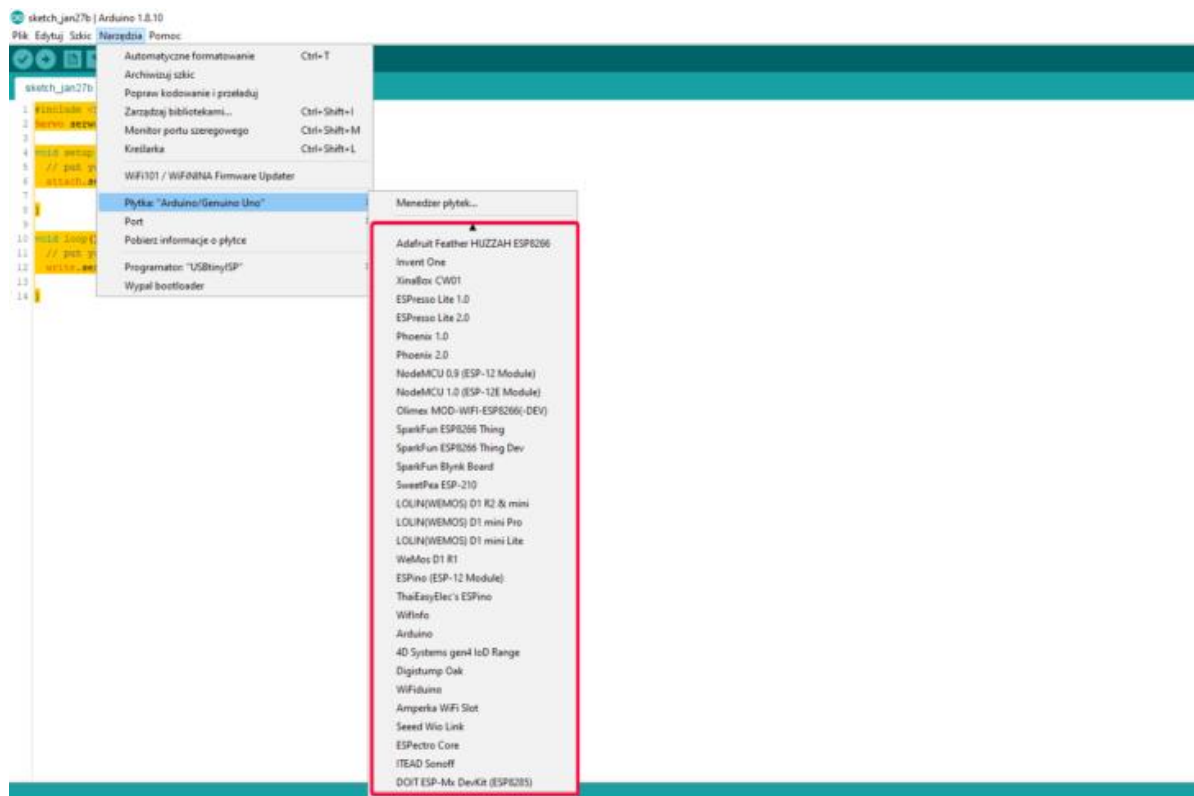
W Arduino IDE wybieramy opcję *Plik > Preferencje* i w polu *Dodatkowe adresy URL do menedżera płytek* wpisujemy poniższy adres:

[https://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](https://arduino.esp8266.com/stable/package_esp8266com_index.json)



Rysunek 3 Dodanie informacji o ESP8266 do Arduino IDE

W kolejnym kroku wybieramy opcję *Narzędzia > Płytki > Menedżer płytek*, w wyszukiwarce wpisujemy hasło "ESP8266" i instalujemy paczkę nazwaną "esp8266 by ESP8266 Community". Od tej pory podczas wyboru płytki dostępne będą różne modele modułów z ESP8266 na pokładzie.



Rysunek 4 Wybór płytek z ESP w Arduino IDE

### 3. Oficjalna dokumentacja

ESP8266: [https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf)

DHT11: <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>