

Wykorzystanie modułów Wi-Fi ESP32 i ESP8266 w aplikacjach Internetu rzeczy

Spis treści

- Moduły ESP8266 i ESP32
- Protokół MQTT
- Programy (gotowe) do komunikacji z modułami ESP
- Zastosowania modułów ESP

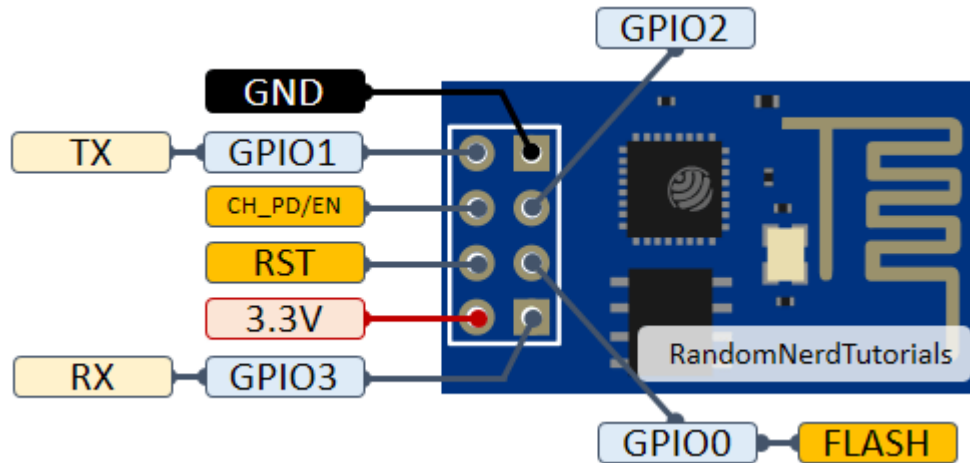
ESP8266

30 grudnia 2013r chińska firma Espressif Systems wypuściła na rynek tani układ scalony ESP8266

Parametry ESP8266

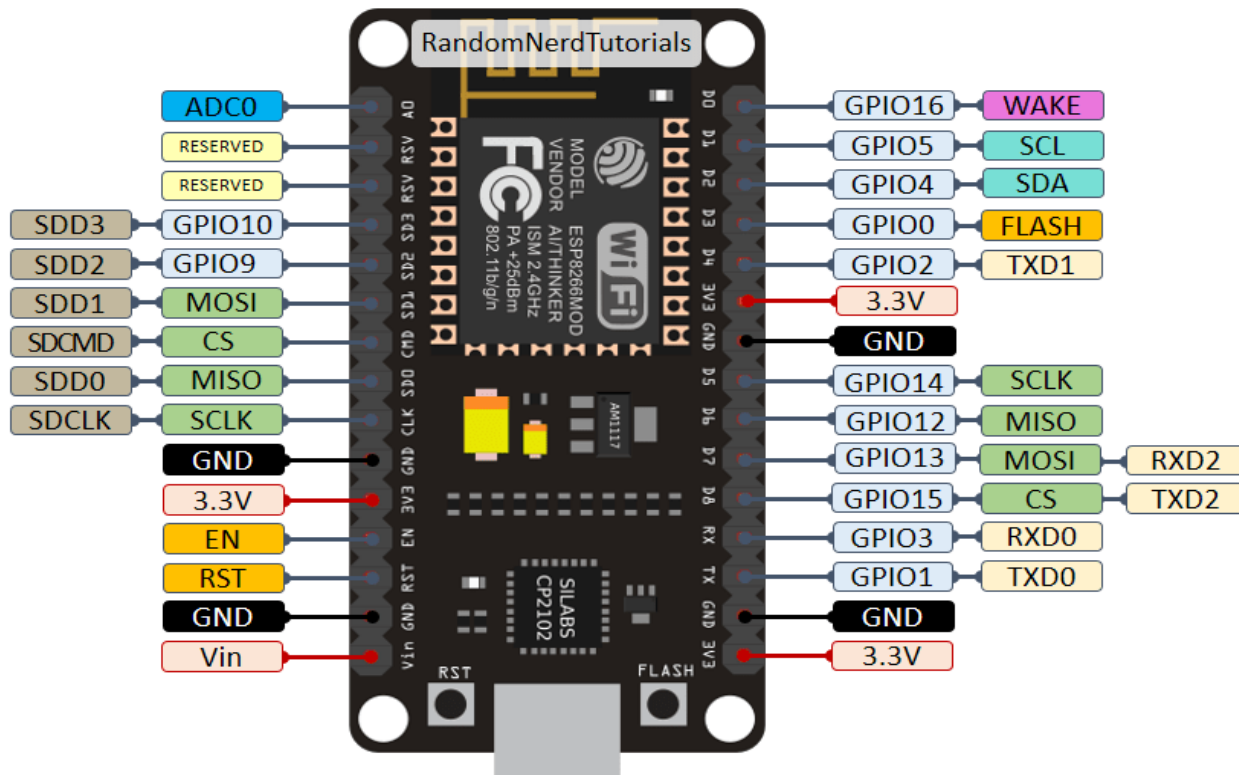
- Napięcie zasilania 3,3 V
- Wbudowany 32 bitowy mikroprocesor RISC Tensilica L106 o częstotliwości taktowania 80 lub 160 MHz
- Pamięć:
- 32 KiB instruction RAM
- 32 KiB instruction cache RAM
- 80 KiB user-data RAM
- 16 KiB system-data RAM
- Zewnętrzna pamięć Flash do 16 MiB (typowo od 512 KiB do 4 MiB)
- **Wi-Fi IEEE 802.11 b/g/n**
- 16 pinów ogólnego przeznaczenia GPIO
- Interfejsy: SPI, I²C (realizacja softwarowa), I²S i UART
- 10 bitowy przetwornik A/C

ESP-01 – pierwszy moduł wykorzystujący ESP8266



ESP-01 dał początek wielu projektom Internetu rzeczy, świetnie się nadaje do montowania w urządzeniach, ale jest on mało przyjazny dla do nauki i rozwijania oprogramowania. Oprogramowanie modułu jest wgrywane do pamięci Flash za pomocą konwertera USB-UART, co jest dość kłopotliwe w warunkach laboratoryjnych. Wygodniejsze w eksperymentowaniu i nauce są płytki rozwojowe zawierające już w sobie konwerter USB-UART.

NodeMCU WiFi DevKit

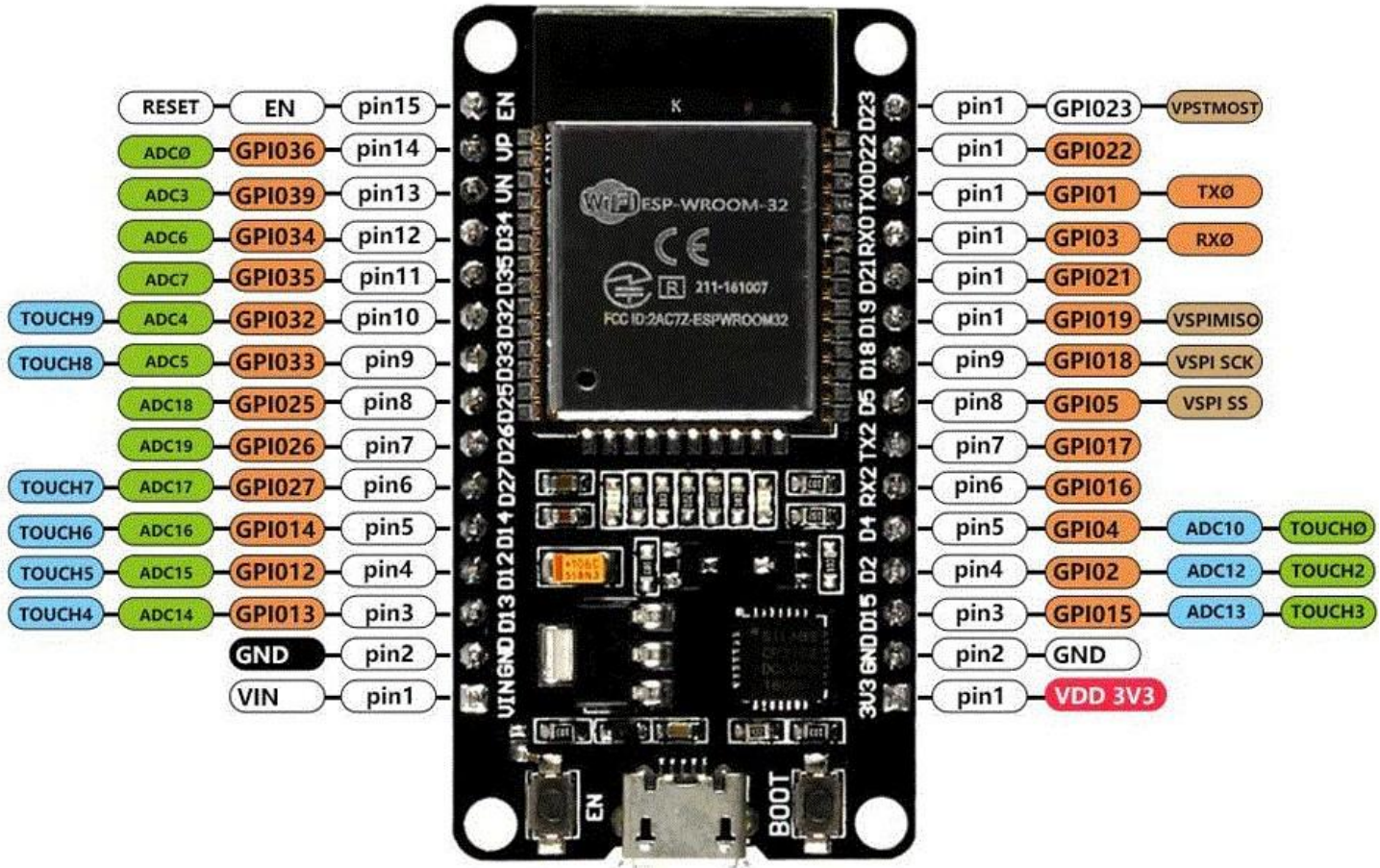


W sierpniu 2016r. na rynku pojawił się mikrokontroler ESP32, następca ESP8266

Podstawowe parametry ESP32

- 32-bitowy mikroprocesor LX6, o częstotliwości 160 lub 240 MHz
- Pamięć 520 KiB SRAM
- Zewnętrzna pamięć Flash do 32 MiB (typowo 4 MiB)
- Wi-Fi IEEE 802.11 b/g/n
- **Bluetooth v4.2**
- Interfejsy: 4 × SPI, 2 × I²C, 2 × I²S i 3 × UART
- 12-bitowy przetwornik A/C – 18 kanałowy
- **2 8-bitowe przetworniki C/A**
- 10 czujników dotykowych (piny GPIO)
- **Czujnik Halla**

ESP32-DevKit

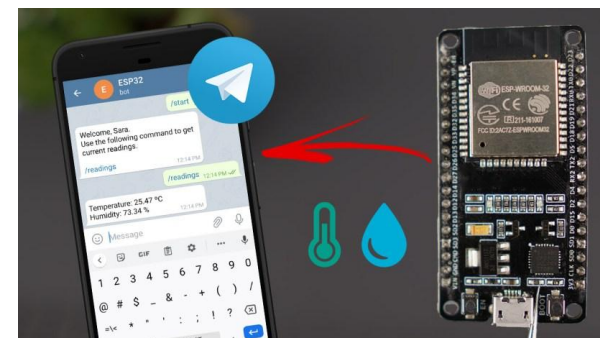
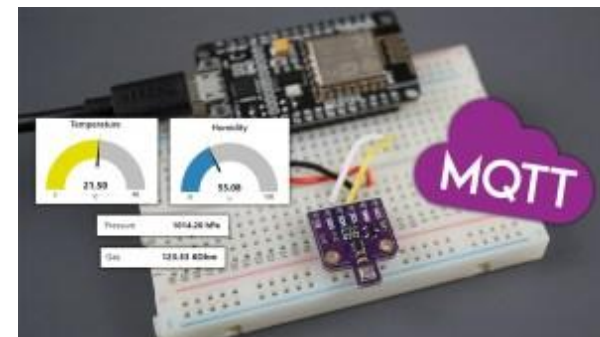


Zastosowania

- Internet rzeczy
- Automatyka domowa
- Systemy telemetryczne
- <https://randomnertutorials.com/>
- <https://hackaday.io/projects?tag=ESP8266>

Stosowane protokoły

- HTTP – serwery, usługi REST, klienci usług,
- MQTT – automatyka domowa, systemy telemetryczne,
- ESP-NOW – protokół opracowany przez firmę Espressif,
- WebSocket.



Protokół MQTT

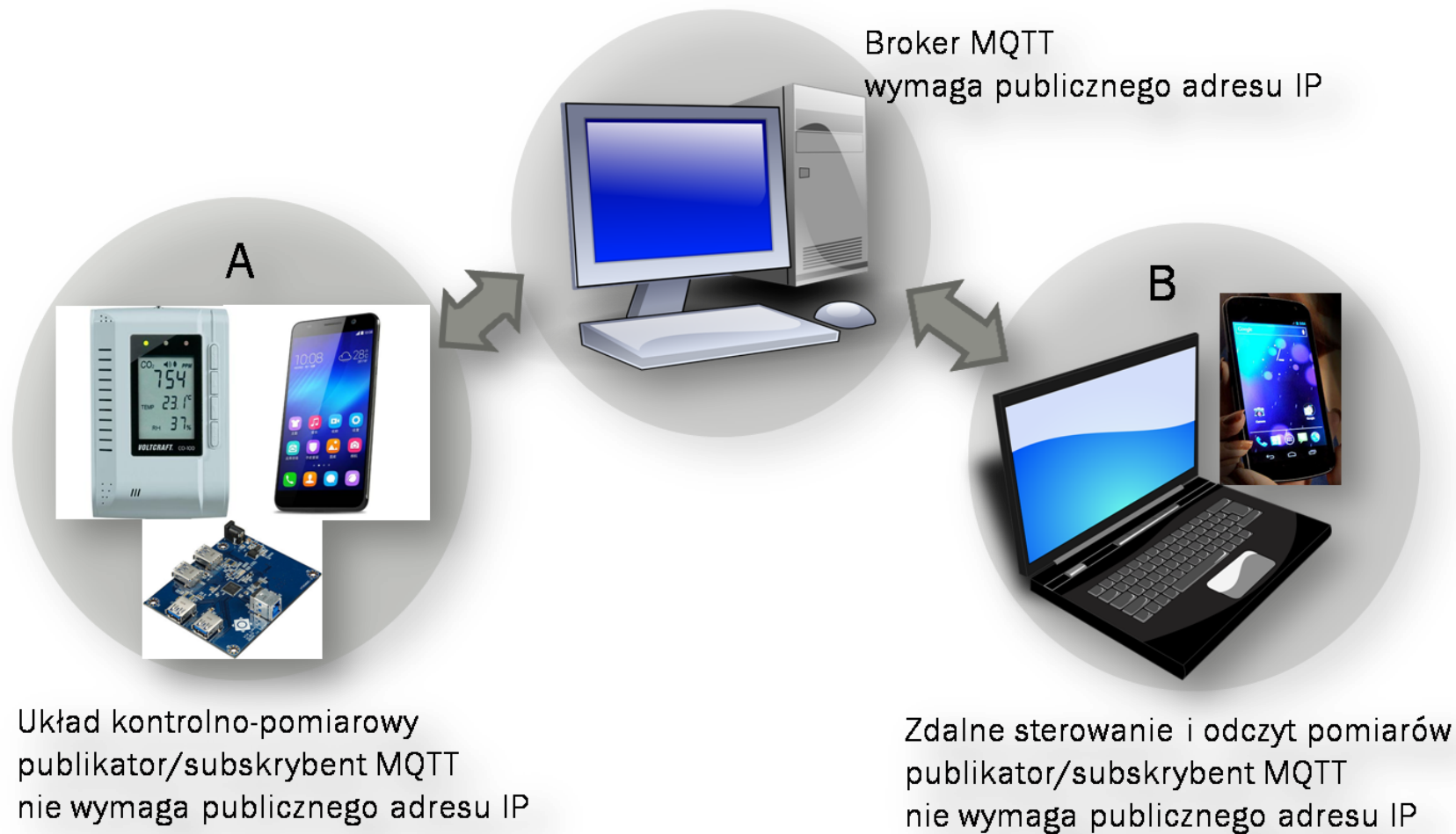
Message Queuing Telemetry Transport

- protokół MQTT jest prostym i lekkim protokołem warstwy aplikacji,
- protokół MQTT jest oparty o wzorzec publikacja/subskrypcja,
- jest przeznaczony do transmisji dla urządzeń niewymagających dużej przepustowości,
- poprzez ograniczenie prędkości transmisji, protokół zapewnia większą niezawodność,
- protokół MQTT idealnie sprawdza się przy połączeniach maszyna-maszyna (M2M), w Internecie rzeczy, w urządzeniach mobilnych, oraz tam, gdzie wymagana jest oszczędność przepustowości, oraz energii,
- przesyłanie wiadomości odbywa się za pośrednictwem brokera MQTT.

Broker MQTT pełni rolę serwera, z którym łączą się klienci.

- Klienci mogą występować w dwóch rolach, publikatorów i subskrybentów.
- Publikatory mogą publikować wiadomości w danym temacie, a subskrybenci mogą subskrybować tematy.
- Broker jest łatwy w instalacji, można korzystać także z bezpłatnych brokerów publicznych.

Łączność z wykorzystaniem protokołu MQTT odbywa się za pośrednictwem Brokera MQTT



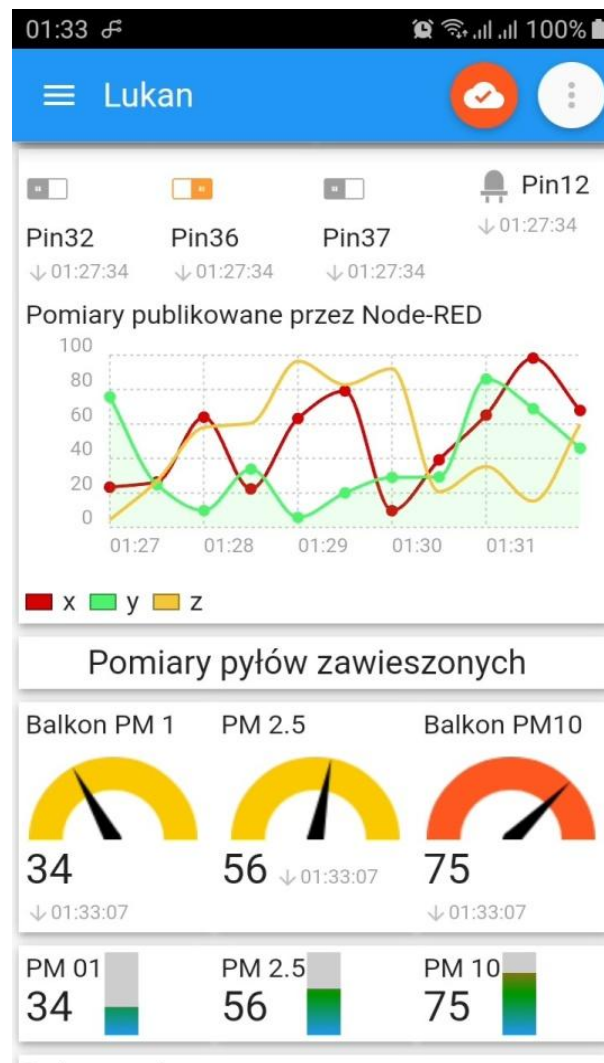
Porównanie protokołów MQTT i HTTP

	MQTT	HTTP
Przesyłana treść	Dane lub krótkie wiadomości	Głównie dokumenty, często bardzo duże
Wzorzec komunikacji	Publikacja / Subskrypcja	Żądanie / Odpowiedź
Złożoność	Mała	Większa
Rozmiar wiadomości	Mały, jeden mały nagłówek	Duży, kilka nagłówków
Jakość usługi	Trzy poziomy, opcja Retain zachowania ostatniej wiadomości w temacie	Jeden poziom
Dystrybucja wiadomości	Jeden do wielu (1:0,1,n) Jedna publikacja jest rozsyłana do wielu subskrybentów, albo do żadnego jeśli żaden nie subskrybuje tematu	Jeden do jednego
Biblioteki programistyczne (klienckie)	Małe	Duże

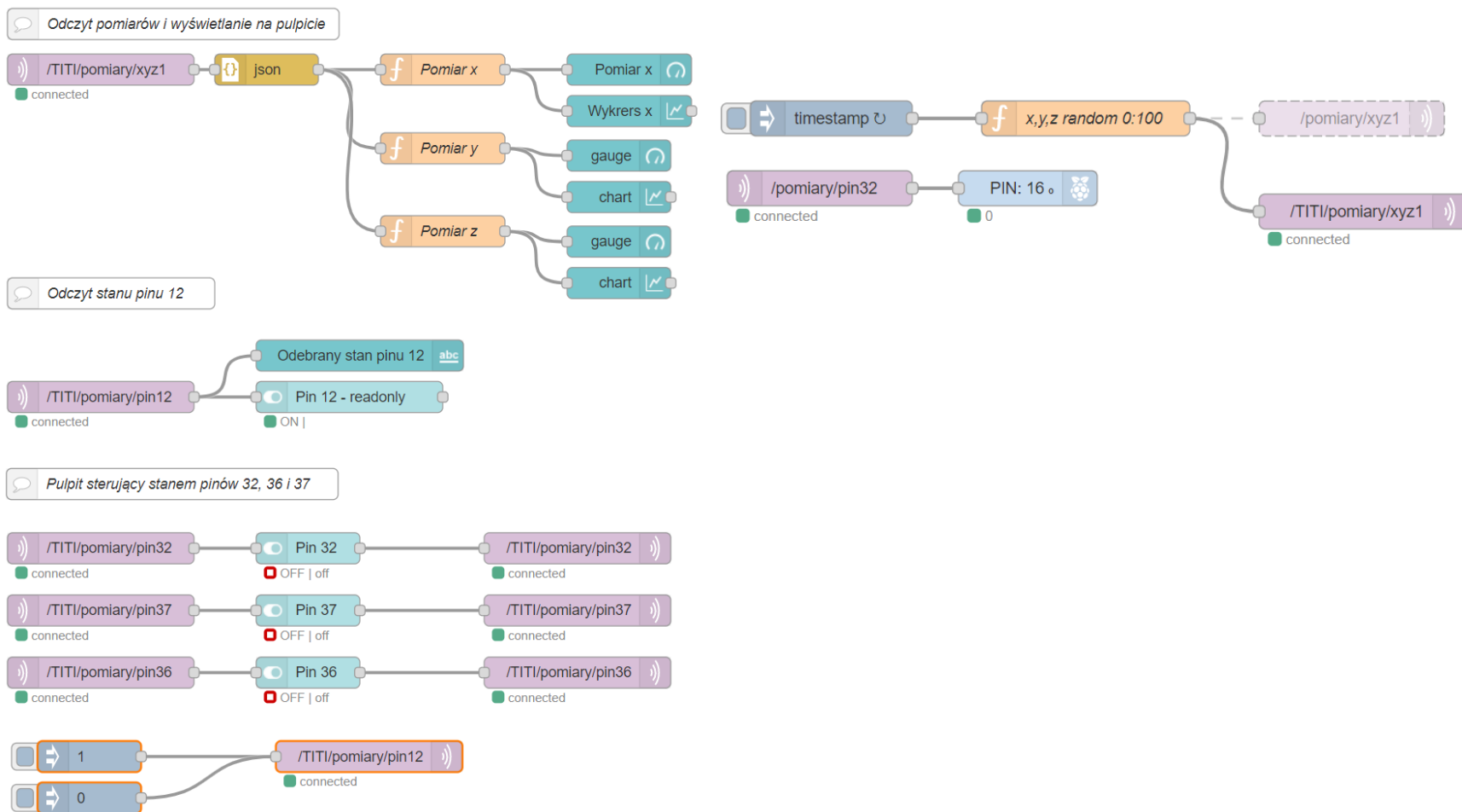
TEMATY

- Tematy nie muszą być wcześniej tworzone, i mogą mieć dowolną nazwę.
- Tematy mają wielopoziomową strukturę hierarchiczną.
- / (slash) – rozdziela poziomy.
- # (hash character) – multi level wildcard.
- + (plus character) – single level wildcard.
- Przykład: house/+/main-light.
- \$SYS/broker/# – komunikaty brokera.
- Symbole + i # można stosować subskrybując tematy, nie można ich stosować publikując.

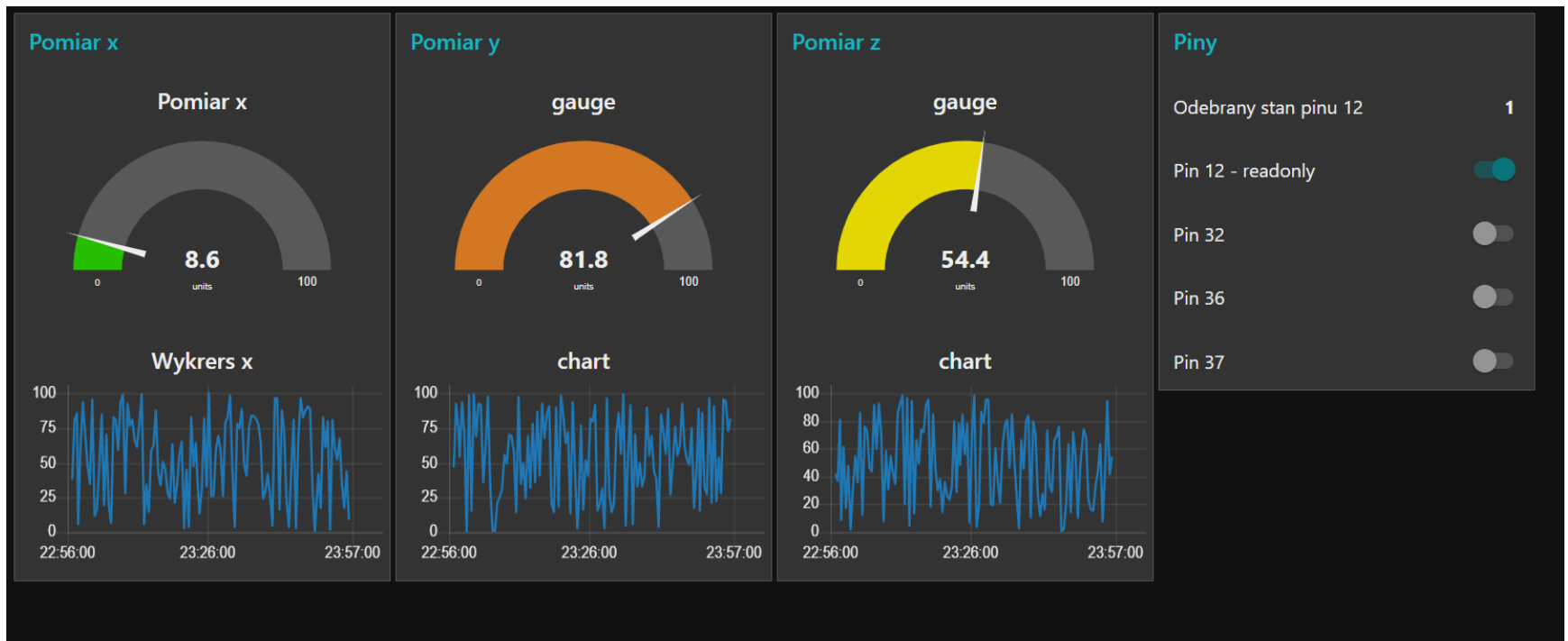
IoT MQTT Panel



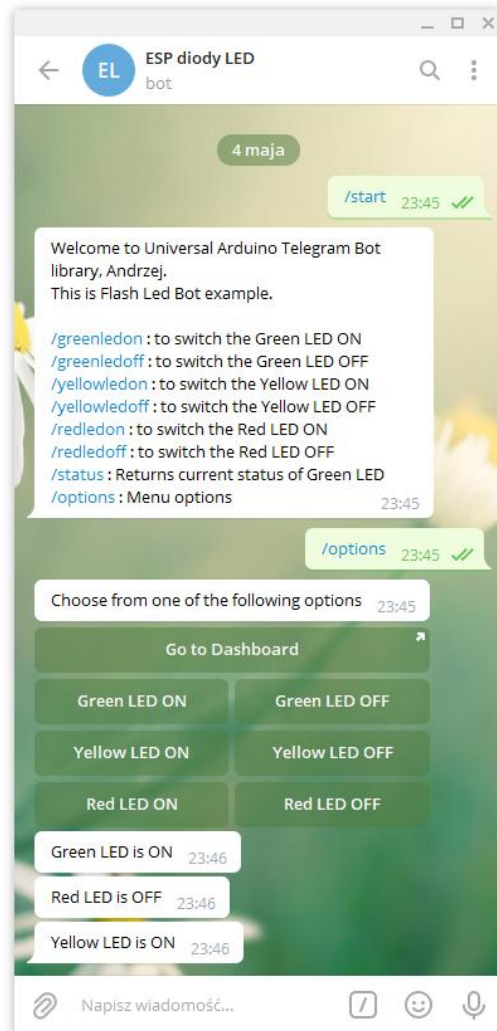
Graf Node-RED do sterowania diodami



Pulpit stworzony przez graf



Telegram - Bot



Boty w programie Telegram tworzy się b. łatwo