

# Moduły komunikacyjne IoT

**W artykule przedstawiono projekt dwóch modułów do komunikacji radiowej umożliwiających transmisję danych w aplikacjach IoT i nie tylko. Pierwszy to popularny moduł ESP8266 umożliwiający łączność przez sieć Wi-Fi. Drugim jest Bluetooth BLE4 z komunikacją radiową w standardzie „niskomocowego” interfejsu Bluetooth, oparty o moduł RN4020. Rekomendacje: moduły są zgodne mechanicznie ze standardem XBee, co ułatwia ich zastosowanie w szeregu płytek uruchomieniowych min.: Raspberry PI (AVT1854 z EP6/15), Launchpad (AVT5476 z EP11/14), Arduino itd. BT41 i Wi-Fi uzupełniają zestaw modułów komunikacyjnych opisanych w EP9/15 (kity AVT5513).**

Internet rzeczy stał się rzeczywistością, więc nie ma na co czekać i warto wprowadzić swoje aplikacje w nową rzeczywistość. Oba prezentowane moduły mają wsparcie programowe skracające czas niezbędny od pomysłu do jego realizacji.

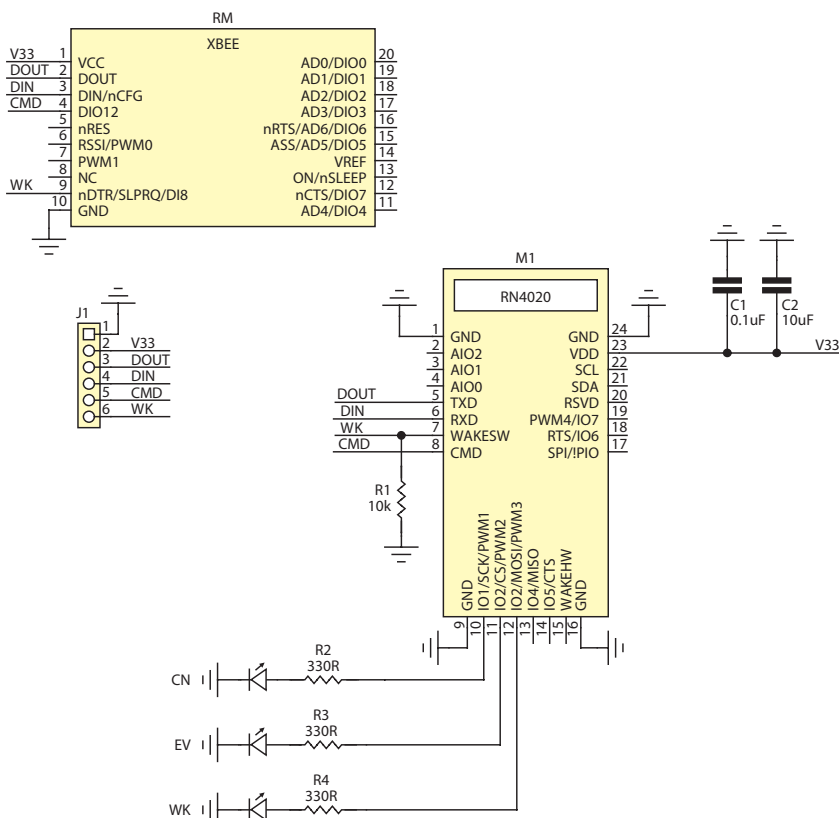
Schemat ideowy płytki rozszerzenia Wi-Fi pokazano na **rysunku 1**. Bazuje on na module ESP8266v7. Wybrana go ze względu na niską ceną, dostępność, wbudowaną antenę oraz możliwość przyłączenia anteny zewnętrznej za pomocą typowego wtyku UFL. Komunikacja z modulem odbywa się poprzez port szeregowy DIN/DOU. Możliwe jest wykorzystanie kilku pinów GPIO wbudowanego procesora (np. w aplikacjach samodzielnych, bez zewnętrznego procesora). Zwora UPD służy do wprowadzenia modułu w tryb aktualizacji oprogramowania (zwarne 2-3).

Schemat płytki rozszerzenia Bluetooth zamieszczono na **rysunku 2**. Jej najważniejszym

elementem jest moduł RN4020 firmy Microchip. Moduł ma certyfikację Bluetooth, wbudowany stos BLE4.0. Komunikacja odbywa się poprzez port szeregowy. Istnieje możliwość aktualizowania oprogramowania przez UART oraz przez interfejs radiowy. Moduł ma API ułatwiające jego zastosowanie. Układ uzupełniają trzy LED sygnalizujące stan pracy: CN połączenie, EV zdarzenie (w trybie MLDP), WK aktywność modułu. Dodatkowo, wyprowadzono sygnały WK (WAKESW) wybudzania modułu z trybu Deep Sleep (aktywny poziom wysoki) oraz CMD włączający tryb Command/MLDP (opadające zbocze wprowadza RN4020 w tryb Command).

Oba moduły zmontowano na dwustronnych płytkach drukowanych. Rozmieszczenie elementów przedstawiają **rysunki 3** oraz **4**. Zmontowane moduły przedstawiono na **fotografiach 5** i **6**.

Dla szybkiego sprawdzenie poprawności działania modułu Wi-Fi możemy połączyć go z płytką bazową konwertera XBee\_PC lub dowolną inną zgodną z 3,3 V oraz zasilaczem 3,3 V/300 mA. Do komunikacji z ESP będzie potrzebny terminal znakowy. Po połączeniu układu, restarcie, ustawieniu terminala (115200, 8, n, 1) wpisujemy komendę AT+RST. Moduł zostanie zrestartowany oraz potwierdzi gotowość za pomocą komunikatu OK. Komendą AT+GMR sprawdzamy wersję oprogramowania (możliwe i polecane jest uaktualnienie np. programem **ESP FLASH Download TOOL**, **XTCOM Utility** lub innymi przeznaczonymi dla ESP). Komenda AT+CWMODE=3, załącza tryb pracy klient i punkt dostępowy Wi-Fi. Komenda AT+CWLAP? wyświetla listę dostępnych sieci Wi-Fi. Zalogowanie do sieci umożliwia komenda AT+CWLAP= „nazwasieci”



**Rysunek 1. Schemat ideowy modułu Wi-Fi**

**DODATKOWE MATERIAŁY NA FTP:**

<ftp://ep.com.pl>

**USER: 22118, PASS: 07764433**

Podstawowe informacje:

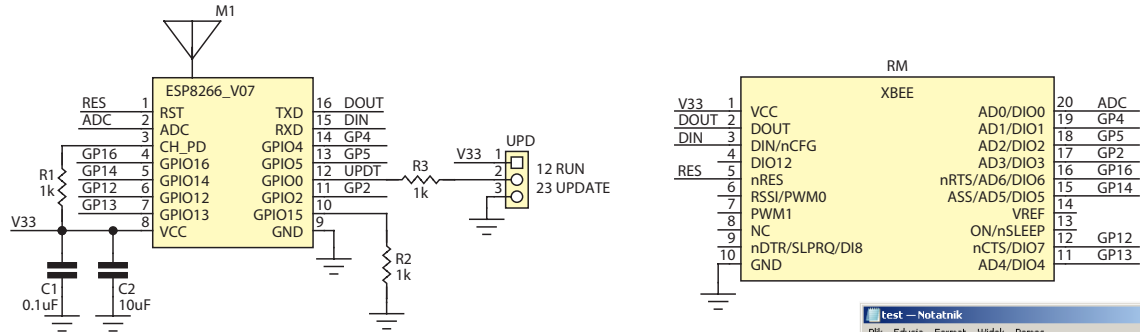
- łączność za pomocą Wi-Fi lub Bluetooth.
- Sterowanie z użyciem komend AT.
- Dołączenie do systemu nadrzędnego przez UART.
- Zasilanie 3,3 V/200 mA.
- Moduły kompatybilne pod względem wyprawadzeń i wymiarów.
- Do zastosowania z Raspberry PI, Launchpad, Arduino itp.

Projekty pokrewne na FTP:  
(wymienione artykuły są w całości dostępne na FTP)

AVT-1773	Xbee Mini – łączność bezprzewodowa (EP 9/2013)
AVT-5332	ZigT system kontrolno-pomiarowy pracujący z użyciem łączności ZigBee (EP 3/2012)
AVT-5313	IntelliDom – System sterowania inteligentnego budynku z interfejsem ZigBee (EP 10-11/2011)

\* Uwaga:  
Zestawy AVT mogą występować w następujących wersjach:  
AVT xxxx UK to zaprogramowany układ. Tylko i wyłącznie. Bez elementów dodatkowych.  
AVT xxxx A płytka drukowana PCB (lub płytki drukowane, jeśli w opisie wyraźnie zaznaczono), bez elementów dodatkowych.  
AVT xxxx A+ płytka drukowana i zaprogramowany układ (czyli połączenie wersji A i wersji UK) bez elementów dodatkowych.  
AVT xxxx B płytka drukowana (lub płytki) oraz komplet elementów wymienionych w załączniku pdf.  
AVT xxxx C to nic innego jak zmontowany zestaw B, czyli elementy wmontowane w PCB. Należy mieć na uwadze, że o ile nie zaznaczono wyraźnie w opisie, zestaw ten nie ma obudowy ani elementów dodatkowych, które nie zostały wymienione w załączniku pdf.  
AVT xxxx CD oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można pobrać, klikając w link umieszczony w opisie kitu).

Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf! Podczas składania zamówienia upewnij się, którą wersję zamawiasz! (UK, A, A+, B lub C). <http://shlep.ep.com.pl>



Rysunek 2. Schemat ideowy interfejsu Bluetooth

„hasło”. Sprawdzenie uzyskanego IP następuje komendą *AT+CIFSR*.

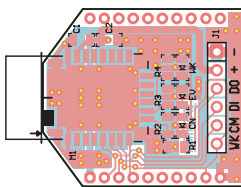
Możliwa jest też aktualizacja oprogramowania przez Wi-Fi. Odpowiada za to komenda *AT+CIUPDATE*. Po pomyślnej aktualizacji moduł jest restartowany i ponownie

automatycznie logowany do sieci określonej przy konfiguracji. Wersję po aktualizacji sprawdzamy komendą *AT+GMR*. Przykładową komunikację testową z ESP przedstawia rysunek 7.

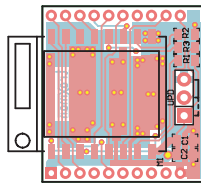
Podobnie jest z modułem Bluetooth. Do wstępnego sprawdzenia ponownie będzie potrzebna płytka XBee\_PC oraz terminal. Po przyłączeniu modułu należy wybudzić go z trybu uśpienia poprzez zwarcie WK z +3,3 V. Zostanie to potwierdzone zaświeceniem się diody WK. Następnie, w terminalu należy wstępnie skonfigurować RN4020 wprowadzając go w tryb poleceń CMD (opadające zbrocze sygnału CMD). W terminalu zostanie to potwierdzone wyświetleniem tekstu „CMD”. Domyślnymi ustawieniami UART są 115200, 8, N, 1. Można je oczywiście przystosować do możliwości systemu, komendą SB, <0-7> (gdzie 0=2400,1=9600...7=921600). Następnie, warto nadać urządzeniu własną nazwę w sieci Bluetooth poleceniem S-<nazwa>, np. S,BT41 oraz nadać nazwę modułu poleceniem SN,<nazwa>, np. SB,BT41. Od tej chwili można sprawdzić za pomocą np. smartfona, czy nasz moduł zgłosił się w sieci. Po wybraniu parowania w terminalu zostanie wyświetlone hasło dostępowe i możliwe będzie połączenie urządzeń (rysunek 8). Poprawne połączenie potwierdzone jest także zaświeceniem diody CN, na płytce modułu.

Jeżeli wszystko wstępnie zadziałało, nie pozostaje nic innego jak wybór medium komunikacyjnego IoT i przejście do stworzenia własnego urządzenia dostępnego z całego świata.

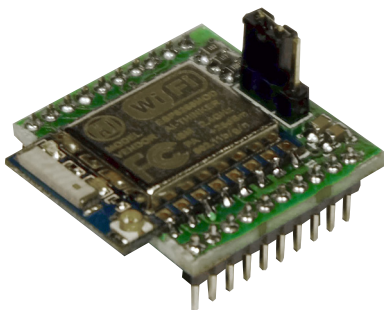
Adam Tatuś, EP



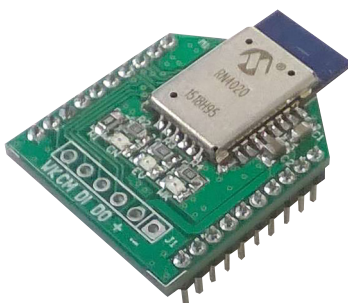
Rysunek 3. Schemat montażowy interfejsu Wi-Fi



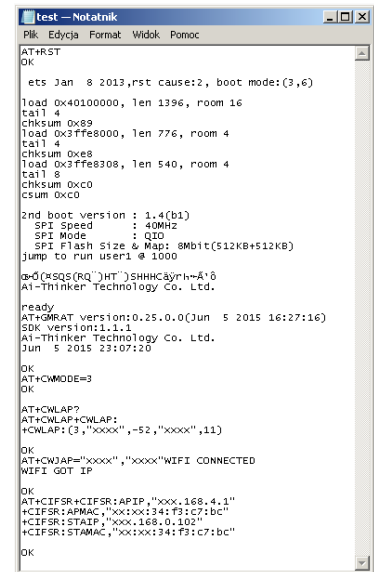
Rysunek 4. Schemat montażowy interfejsu Bluetooth



Fotografia 5. Zmontowana płytka rozszerzenia Wi-Fi



Fotografia 6. Zmontowana płytka rozszerzenia Bluetooth



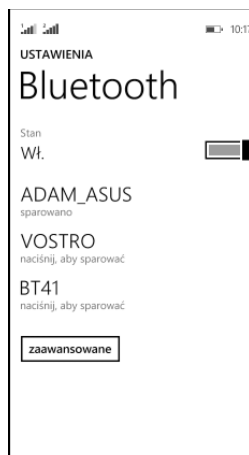
Rysunek 8. Konfigurowanie i parowanie BT41

**Wykaz elementów  
Moduł Wi-Fi**

- R1...R3: 1 kΩ / 5% (SMD 0805)
- C1: 0,1 μF (SMD 0805, X7R)
- C2: 10 μF (SMD 0805, X7R)
- M1: moduł ESP8266 ver 7
- RM: złącza szpilkowe 2 mm, SIP10
- UPD: złącze szpilkowe 2,54 mm, SIP3 + zwora

**Moduł Bluetooth**

- R1: 10 kΩ/5% (SMD 0805)
- R2...R4: 330 Ω/5% (SMD 0805)
- C1: 0,1 μF (SMD 0805, X7R)
- C2: 10 μF (SMD 0805, X7R)
- CN, EV, WK: dioda LED (SMD 0805)
- J1: złącze SIP 2,54 mm
- M1: moduł BLE RN4020
- RM: złącze szpilkowe 2 mm, SIP10



Rysunek 7. Przykładowa konfiguracja XBee Wi-Fi