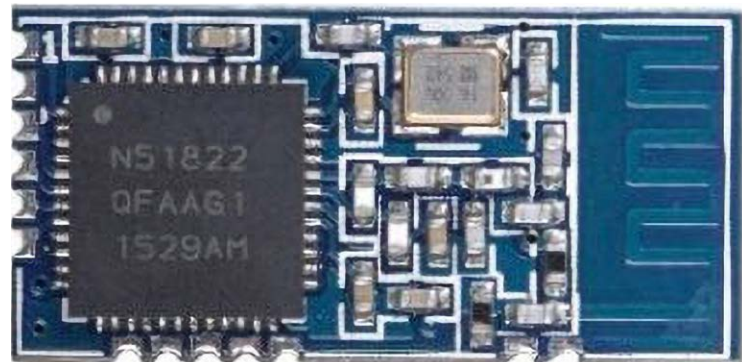
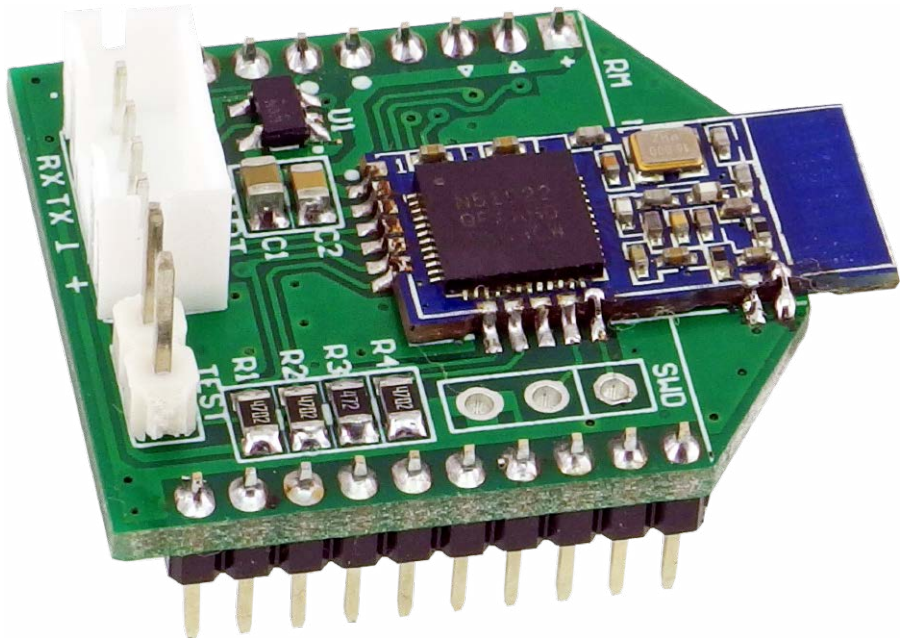


Moduł XBee Bluetooth BLE4.1

Prezentowany moduł Wireless-Tag WT51822-S4AT jest jednym z najmniejszych i łatwiejszych w aplikacji rozwiązań komunikacyjnych BLE4.1. Wyposażony jest w układ Nordic nRF51822 i dzięki niewielkim rozmiarom (18,5×9,1×2 mm), małej ilości niezbędnych do pracy elementów zewnętrznych oraz szybkiej konfiguracji jest ciekawą alternatywą dla tanich, słabo udokumentowanych modułów lub rozbudowanych modułów renomowanych producentów, które w prostszych aplikacjach nie wykorzystują nawet części swoich możliwości.

Płytkę rozszerza zestaw opisanych w EP rozwiązań komunikacyjnych zachowujących zgodność z podstawką XBee. Zastosowany moduł WT51822-S4AT, którego wygląd pokazano na fotografii 1, odpowiada za realizację



Fotografia 1. Moduł WT51822-S4AT (za notą Wireless-tag)

Dodatkowe materiały do pobrania ze strony www.media.avt.pl

W ofercie AVT* AVT-5754

Podstawowe parametry:

- konwerter UART-Bluetooth 4.1,
- domyślne parametry transmisji 9600,8,N,1,
- konfiguracja przy pomocy komend AT,
- zasilany napięciem 3,3 V.

Wykaz elementów:

Rezystory:

R1, R2, R4: 47 kΩ SMD0805
R3: 4,7 kΩ SMD0805

Kondensatory:

C1: 10 µF/10 V SMD0805
C2: 100 nF SMD0805

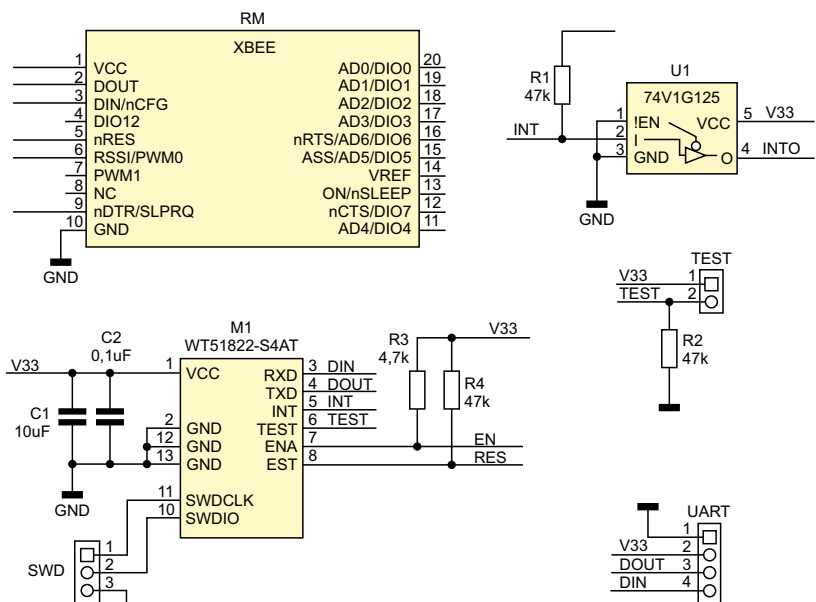
Półprzewodniki:

M1: Moduł BT WT51822-S4AT
U1: SN74LVC1G125SBV SOT-23-5

Pozostałe:

RM: Podstawka XBee 2x10 pin SIL 2 mm
TEST: Listwa SIP2 + Zwora
UART: Złącze PH 4 pin 2 mm

Uwaga! Elektroniczne zestawy do samodzielnego montażu. Wymagana umiejętność lutowania!
Podstawową wersją zestawu jest wersja [B] nazywana potocznie KIT-em (z ang. zestaw). Zestaw w wersji [B] zawiera elementy elektroniczne (w tym [UK] - jeśli występuje w projekcie), które należy samodzielnie wzlutować w dołączoną płytkę drukowaną (PCB). Wykaz elementów znajduje się w dokumentacji, która jest podlinkowana w opisie kitu. Mając na uwadze różne potrzeby naszych klientów, oferujemy dodatkowe wersje:
• wersja [C] - zmontowany, uruchomiony i przetestowany zestaw [B] (elementy wzlutowane w płytkę PCB)
• wersja [A] - płytkę drukowaną bez elementów i dokumentacji Kity w których występuje układ scalony wymagający zaprogramowania, mają następujące dodatkowe wersje:
• wersja [A+] - płytkę drukowaną [A] + zaprogramowany układ [UK] i dokumentacja
• wersja [UK] - zaprogramowany układ
Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf! Podczas składania zamówienia upewnij się, którą wersję zamawiasz!
<http://sklep.avt.pl>. W przypadku braku dostępności na <http://sklep.avt.pl>, osoby zainteresowane zakupem płytek drukowanych (PCB) prosimy o kontakt via e-mail: kity@avt.pl.



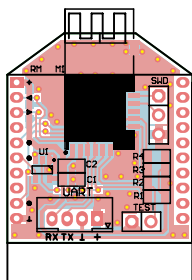
Rysunek 1. Schemat elektryczny układu

transmisji szeregowej UART z wykorzystaniem BLE4.1. Można śmiało nazwać go „radiowym” odpowiednikiem konwerterów UART/USB produkcji FTDI. Dzięki zastosowaniu WT51822-S4AT, w łatwy sposób można zwiększyć funkcjonalność istniejącego projektu, opartego o procesor bez komunikacji radiowej o monitorowanie lub sterowanie poprzez Bluetooth, wprowadzając aplikację w świat IoT i to dosłownie w ciągu kilku minut!

Budowa i działanie

Schemat układu pokazano na **rysunku 1**. Moduł WT51822-S4AT zasilany jest napięciem 3,3 V z podstawki zgodnej z Xbee. Interfejs UART, dodatkowo oprócz złącza Xbee, razem z zasilaniem wyprowadzony jest na złącze PH UART, co ułatwia zastosowanie w systemach pozbawionych podstawki Xbee. Bramka U1 buforuje sygnał przerwania generowany przez M1 po odebraniu danych.

Zwora TEST służy do szybkiego przetestowania modułu, który w trybie tym niezwłocznie odsyła odebrane dane do sparowanego urządzenia, czyli realizuje funkcję loop-back, ale korzystając z interfejsu radiowego BLE – bez konieczności sterowania przez UART. Sprawdzenie działania modułu może więc odbyć się przy wykorzystaniu terminala szeregowego Bluetooth, bez napisania linijki kodu. Sygnał EN aktywuje



Rysunek 2. Schemat płytki PCB wraz z rozmieszczeniem elementów

```
COM7:9600baud - Tera Term v1.7
File Edit Setup Control Window Help
AT+RX
Name: WT-0001
Baud: 9600
MAC: F7CDF29E5FC6
AT+VERSION
V1.1
AT+NAME=ATWT01
OK
teacoffemilk
herbatakawamleko
```

Rysunek 3. Konfiguracja modułu w programie Tera Term

UART modułu M1. Do wyprowadzeń Xbee oprócz sygnałów DIN/DOOUT doprowadzone są sygnały EN i RES.

Montaż

Układ zmontowany jest na dwustronnej płycie drukowanej, której schemat i rozmieszczenie elementów pokazano na **rysunku 2**. Montaż modułu nie wymaga opisu, po zmontowaniu wygląda jak na fotografii tytułowej.

Uruchomienie

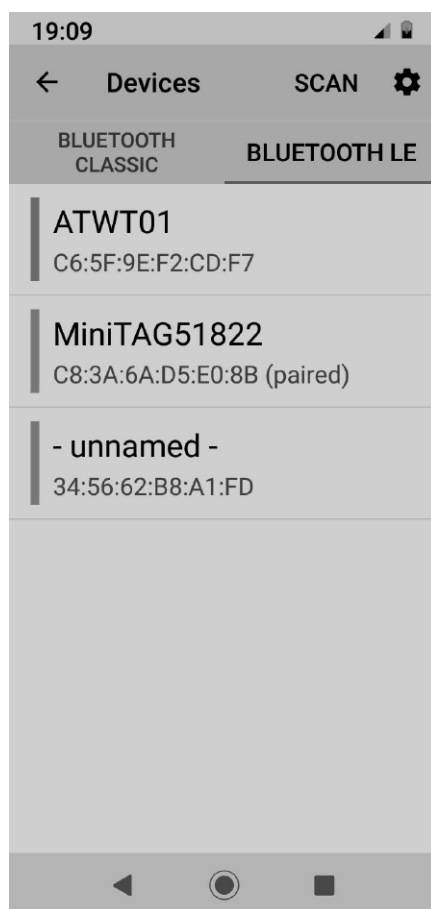
Do uruchomienia modułu przydatny będzie konwerter Xbee-PC (AVT5513), po zamontowaniu modułu **Xbee Bluetooth** w płytkę Xbee-PC, możemy podłączyć ją do komputera PC. Następnie należy uruchomić terminal komunikacji szeregowej np.: Tera Term lub inny umożliwiający wysyłanie komend AT wykorzystywanych do konfiguracji modułu. Domyślnie WT51822-S4AT ma ustawione parametry transmisji 9600,8,N,1 i tak należy skonfigurować terminal.

Podstawową komendą jest odczyt parametrów modułu AT+RX, który zwraca nazwę, prędkość transmisji oraz adres MAC, natomiast komenda AT+VERSION zwraca wersję modułu. Każda komenda AT wysłana do modułu wymaga zakończenia \r. Aby zmienić nazwę (maksymalnie 20 bajtów) wydajemy komendę AT+NAME=NowaNazwaModułu, jeżeli operacja przebiegnie poprawnie

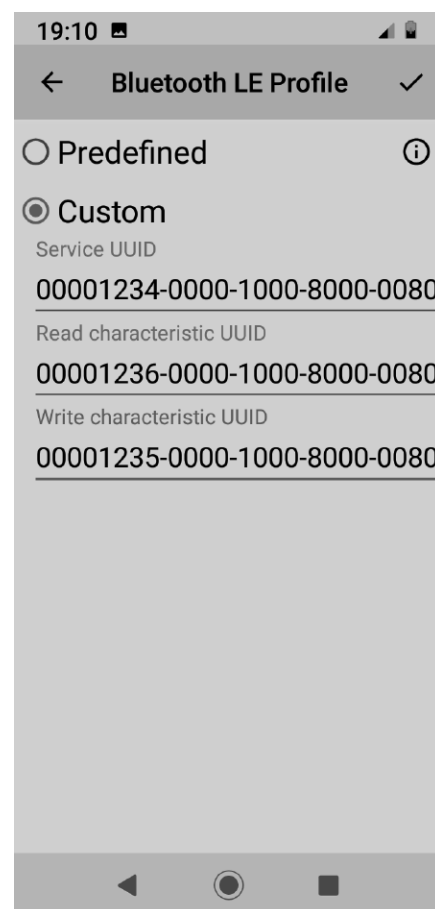
otrzymamy potwierdzenie OK. Od tego momentu moduł gotowy jest do pracy.

Przebieg konfiguracji w Tera Term przedstawiono na **rysunku 3**. Moduł obsługuje prędkości transmisji pomiędzy 1200-230400 bps, producent jednak nie zaleca ustawiania większej niż 57600 bps ze względu na możliwe przekłamania w transmisji. Komendą AT+BAUD=xxxx można ustawić wymaganą prędkość transmisji. Jeżeli nie jest wymagana pełna moc nadajnika możemy dostosować ją do wymogów aplikacji (zasięgu) komendą AT+RFPM=x gdzie x=0-8 ustawia moc +4 dBm do -40 dBm zgodnie z dokumentacją. Komenda AT+RESET restartuje moduł, a AT+DEFAULT przywraca ustawienia domyślne.

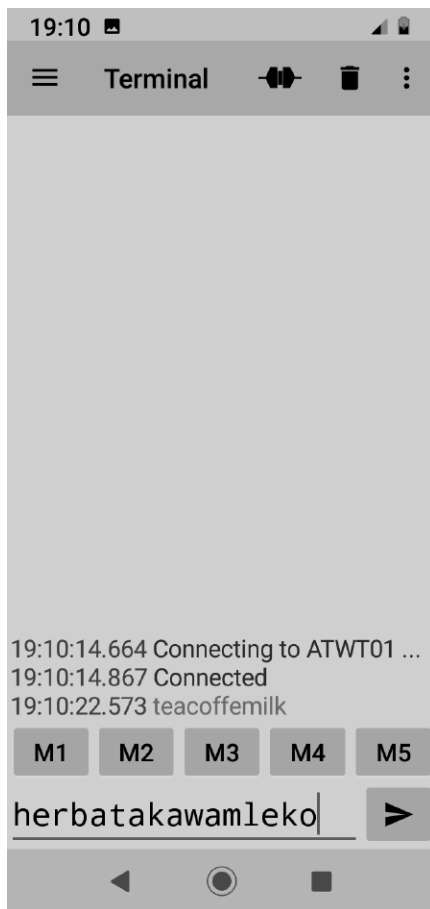
Po konfiguracji moduł gotowy jest do sparowania. Do tego celu użyjemy smartfona z systemem Android i zainstalowanym monitorem portu szeregowego Bluetooth. W moim przypadku jest to Serial Bluetooth Terminal 1.29 dostępny w sklepie Google Play, którego autorem jest Kai Morich. Pierwszą czynnością jest włączenie interfejsu Bluetooth w smartfonie, następnie w aplikacji terminala należy zeskanować dostępne urządzenia (**rysunek 4**). Na liście dostępnych modułów znajduje się także ATWT01, dla którego należy zdefiniować profile usług. Konfiguracji dokonujemy poprzez wybór i przytrzymanie nazwy



Rysunek 4. Wyszukiwanie urządzeń BT



Rysunek 5. Definiowanie UUID



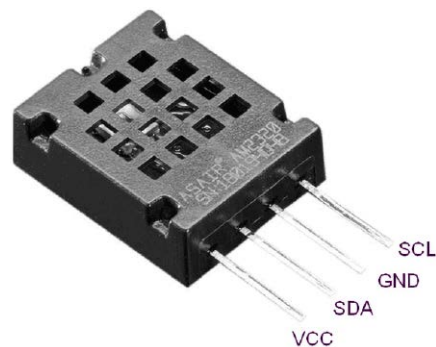
Rysunek 6. Transmisja testowa z terminalem

modułu ATWT01, co wywoła opcje Connect i Edit. Należy zdefiniować profile (Edit) zgodnie z **rysunkiem 5**. Następnie łączymy się z modulem (Connect), co potwierdzone zostanie także w okienku terminala. Od tego momentu można sprawdzić komunikację pomiędzy modulem **XBee Bluetooth** podłączonym do PC, a terminalem BLE w smartfonie (**rysunek 6**).

Dla bardziej użytecznego wykorzystania WT51822, podłączymy go do portu szeregowego Arduino MKR Zero (sygnały i zasilanie w standardzie 3,3 V), do magistrali I2C podłączymy czujnik temperatury i wilgotności AM2320. Wyprowadzenia czujnika AM2320 pokazuje **rysunek 7**. Konieczne jest podwieszenie magistrali I²C rezystorami 2,2 kΩ i odsprężnienie zasilania kondensatorem 100 nF. Po wgraniu szkicu z **listingu 1** i nawiązaniu połączenia w terminalu szeregowym BLE, co 5 sekund będzie odczytywana wartość zmierzonej temperatury i wilgotności (**rysunek 8**).

Z pewnością udało się zmieścić w kilku minutach. Jak dobrze, że dzisiaj dostępne są tak zaawansowane technicznie, a jednocześnie tak proste w zastosowaniu podzespoły. Powodzenia w tworzeniu własnych aplikacji z modulem **XBee Bluetooth**.

Adam Tatus
adam.tatus@ep.com.pl



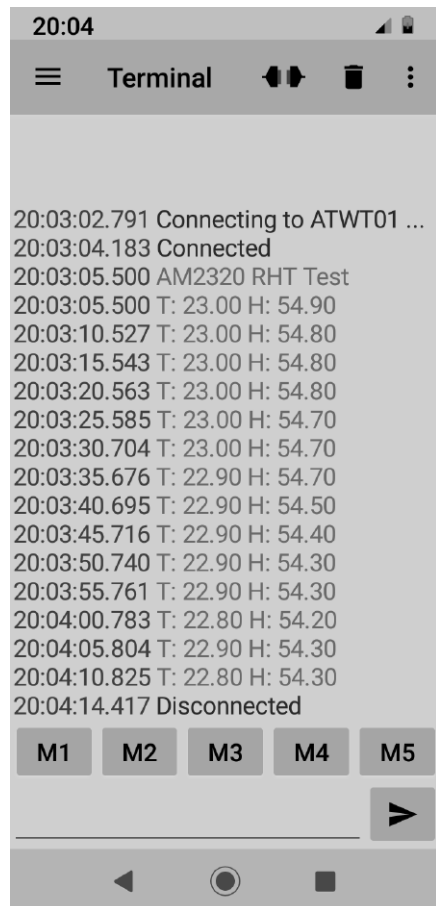
Rysunek 7. Czujnik AM2320

```
Listing 1. Przykładowy szkic w Arduino
#include „Adafruit_Sensor.h”
#include „Adafruit_AM2320.h”

Adafruit_AM2320 am2320 = Adafruit_AM2320();

void setup() {
  Serial1.begin(9600);
  while (!Serial1) {
    delay(100);
  }
  Serial1.println(„AM2320 RHT Test”);
  am2320.begin();
}

void loop() {
  Serial1.print(„T: "); Serial1.print(am2320.readTemperature());
  Serial1.print(„ H: "); Serial1.println(am2320.readHumidity());
  delay(5000);
}
```



Rysunek 8. Działanie aplikacji przykładowej