

ESP8266, czyli kontrola na odległość z dowolnego miejsca w zasięgu sieci telefonii komórkowej

Zainspirowany szeregiem artykułów, które zostały publikowane na łamach Elektroniki Praktycznej, zainteresowałem się modułem Wi-Fi ESP8266. Zacząłem zastanawiać się czy można go użyć do sterowania na odległość, chociażby diodami LED. Okazało się, że bez trudu! W dodatku korzystając z infrastruktury lokalnego Ethernetu i telefonii komórkowej, można osiągnąć imponujący zasięg.

Sterowanie na odległość ma swoje miejsce w zastosowaniach zarówno zabawowych jak i tych całkiem na poważnie. Jak chociażby w idei inteligentnego domu. Zdalna kontrola temperatury, włączanie lub wyłączenie ogrzewania, zdalny dozór nad mieszkaniem lub domem, otwieranie bramy wjazdowej lub drzwi garażu... Wszystkie możliwe zastosowania wiążą się z transmisją danych na odległość. Miniaturowe moduły Wi-Fi ESP8266 doskonale się będą do tego nadawały. I to, w zależności od potrzeb, do połączeń na niewielką odległość lub bardzo dużą.

Na krótkim dystansie kilkunastu metrów moduł Wi-Fi może być użyty jako część sterownika np. bramy wjazdowej, otwieranej przez aplikację zainstalowaną w telefonie komórkowym. Dzięki wbudowanemu w oprogramowaniu modułu ESP8266

mechanizmom ułatwiającymi komunikację z ruterami, można go zastosować do sterowania poprzez lokalną sieć Ethernet, łączącą komputery na określonym obszarze takim jak blok, szkoła, laboratorium, biuro itp. Jeżeli lokalny Ethernet ma połączenie z siecią WAN (Wide Area Network, rozległą siecią komputerową), dostęp do sterowanych urządzeń poprzez moduł ESP8266 jest możliwy z każdego zakątka Internetu. A poprzez sieć telefonii komórkowej, ściśle powiązanej przecież z Internetem, możemy uzyskać połączenie z naszym modułem bez konieczności użycia kabla.

ESP8266 w wielu odmianach

Moduły Wi-Fi z układem ESP8266 są oferowane w różnych wariantach wykonania. Główne różnice polegają na odmiennej liczbie dostępnych

wyprowadzeń I/O, obecności zintegrowanej anteny, różnych wymiarach płytki i wykonaniu. W tekście będą odwoływał się do najprostszego, oznaczonego jako ESP01, ze zintegrowaną anteną wykonaną w postaci ścieżki na płycie drukowanej. Na **rysunku 1** pokazano rozmieszczenie wyprowadzeń modułu ESP01 w widoku z góry.

Funkcje poszczególnych wyprowadzeń są następujące:

- VCC – zasilanie +3,3 V.
- GND – masa.
- UTXD – wyjście danych interfejsu szeregowego modułu.
- URXD – wejście danych interfejsu szeregowego modułu.
- RST – wejście zerujące, w czasie normalnej pracy musi być dołączone do napięcia zasilającego 3,3 V.
- CH_PD – wejście wprowadzające moduł w tryb obniżonego poboru mocy, w czasie normalnej pracy musi być dołączone do napięcia zasilającego 3,3 V.
- GPIO0, GPIO2 uniwersalne porty wejścia/wyjścia, można pozostawić niepodłączone.

Wszystkie wyprowadzenia sygnałowe pracują z poziomami CMOS 0/3,3 V.

Oprócz wcześniej wymienionych, jest jeszcze jedna różnica pomiędzy modułami nawet tego samego typu: wielkość zamontowanej pamięci Flash. Aktualnie dostępne są moduły z pamięcią Flash o pojemności od 4 Mb do 32 Mb. Różnica wpływa na rodzaj oprogramowania wewnętrznego modułu i sposób jego aktualizacji. Jeżeli nie wiadomo, jaka jest pojemność zamontowanej pamięci Flash, można spróbować się tego dowiedzieć analizując oznaczenie na obudowie 8-nóżkowego układu scalonego, który jest pamięcią Flash. I tak, pamięci o pojemności 4 Mb mogą mieć oznaczenia 25Q40, 25Q41 lub podobne, a 8 Mb – 25Q80 itd.

Aktualizacja oprogramowania modułu

Co pewien czas producent modułów udostępnia aktualizację ich firmware. W ten sposób poszerzane są możliwości, usuwane zauważone błędy, poprawiana stabilność działania. Aktualizację przeprowadza użytkownik. Jeżeli moduł znajduje się w zasięgu rutera mającego połączenie z Internetem, aktualizację można przeprowadzić poprzez sieć Wi-Fi. Innym sposobem jest aktualizacja poprzez interfejs szeregowy modułu, który dołączamy przez konwerter do złącza USB komputera. W skrócie taka aktualizacja będzie przebiegała następująco:

- Należy ustawić wyprowadzenia sygnałowe modułu w trybie gotowości do aktualizacji oprogramowania:
 - UTXD – połączyć z wyprowadzeniem RxD konwertera USB,
 - URXD – połączyć z wyprowadzeniem TxD konwertera USB.
 - RST, CH_PD – podać poziom wysoki (bezpośrednio lub przez rezystor połączyć z napięciem +3,3 V).
 - GPIO0 – podać poziom niski (bezpośrednio lub poprzez opornik połączyć z masą).
- Po włączeniu zasilania moduł znajdzie się w trybie gotowości do aktualizacji. Dla pewności po włączeniu można dodatkowo wyzerować moduł, przez krótkotrwałe wyzerowanie wejścia RST.

UWAGA! NIEKTÓRE TYPY MODUŁÓW ESP, O WIĘKSZEJ LICZBIE WYPROWADZEŃ GPIO, MOGĄ DODATKOWO WYMAGAĆ PODANIA NA NIEKTÓRE Z NICH KOMBINACJI POZIOMÓW LOGICZNYCH – NALEŻY ZAPOZNAĆ SIĘ Z DOKUMENTACJĄ TECHNICZNĄ MODUŁU.

- Ze strony producenta procesora – firmy Espressif Systems – pobrać oprogramowanie firmowe **Esp Flash Download Tool** służące do aktualizacji oprogramowania modułu za pośrednictwem interfejsu szeregowego modułu podłączonego do komputera. Nowsze oprogramowanie znajduje się pod adresem <https://goo.gl/OBVQms>. Starszą wersję FLASH_DOWNLOAD_TOOLS_v0.9.3.1_141118.rar – uproszczoną, ale zupełnie wystarczającą można pobrać z <https://goo.gl/1CM58V>.

- Ze strony producenta pobrać pliki z aktualizowanym oprogramowaniem. W tym miejscu sprawa się komplikuje dla posiadaczy modułów z mniejszą (4 Mb) pamięcią Flash. Ze względu na ograniczoną pojemność nie ma miejsca na pełną aktualizację. Ograniczenie polega na braku możliwości późniejszej kolejnej aktualizacji poprzez Wi-Fi.

Aktualizacja dla modułu z Flash 4 Mb

- Pobrać spod adresu <https://goo.gl/j7sSOM> plik aktualizacyjny **esp_iot_sdk_v1.5.0_15_11_27.zip**. Po jego zainstalowaniu będzie dostępna wersja rozkazów AT v.0.51, zupełnie wystarczająca do dalszych eksperymentów z modułem ESP8266.
- Po rozpakowaniu należy znaleźć w podkatalogu *esp_iot_sdk_v1.5.0/bin/at/* plik **readme.txt**. Po otwarciu do odczytu w sekcji **NON-BOOT MODE** można znaleźć zestawienie plików potrzebnych do aktualizacji dla pamięci 4 Mb.
- Uruchomić program **Esp Flash Download Tool**. Jeżeli pojawi się takie pytanie, należy wybrać aktualizację dla układu ESP8266.
- W sekcji „Download Path Config” należy wskazać pliki do zapisu do pamięci FLASH wraz z adresami. Informacje na ten temat znajdziemy w pliku *readme.txt*. Potrzebne pliki i adresy do zapisu w pamięci są następujące:
 - **esp_iot_sdk_v1.5.0/bin/at/noboot/eagle.flash.bin**: 0x00000,
 - **esp_iot_sdk_v1.5.0/bin/at/noboot/eagle.irom0text.bin**: 0x40000,
 - **esp_iot_sdk_v1.5.0/bin/blank.bin**: 0x3e000,
 - **esp_iot_sdk_v1.5.0/bin/esp_init_data_default.bin**: 0x7c000.
- Wybrać: *Crystal Freq: 26M, SPI SPEED 40MHz, SPI MODE QIO, FLASH SIZE 4Mbit*.
- Nacisnąć przycisk START i poczekać do końca zapisu.

Aktualizacja dla modułu z Flash 8 Mb

- Pobrać plik ESP8266_NONOS_SDK_V1.5.3_16_04_18.zip np. spod adresu <https://goo.gl/vA59zk>. Po jego zainstalowaniu będzie dostępna wersja rozkazów AT v.1.0 dla pamięci 8 Mb.
- Uruchomić program **Esp Flash Download Tool** i przejść do sekcji **Download Path Config**.
- Zgodnie z zestawieniem podanym w pliku „README.MD”, należy wskazać pliki do zapisu do pamięci Flash modułu z następującymi adresami:
 - **/bin/boot_v1.2+.bin**: 0x00000,
 - **/bin/at/512+512/user1.1024.new.2.bin**: 0x01000,
 - **/bin/esp_init_data_default.bin**: 0xf0000,
 - **/bin/blank.bin**: 0x7e000.
- Wybrać: *Crystal Freq: 26M, SPI SPEED 40MHz, SPI MODE QIO, FLASH SIZE 8Mbit*.
- Nacisnąć przycisk START i poczekać do końca zapisu.

Oprogramowanie użytkownika

Oprogramowanie użytkownika może sterować modułem ESP na dwa sposoby. W module jest zaszyty mikrokontroler ESP8266, dla którego można wykonać program i zapisać go w obszarze pamięci Flash. Moduł będzie działał jak mikrokomputer wyposażony w interfejs Wi-Fi. Innym sposobem jest sterowanie modułu komendami AT z dodatkowego zewnętrznego kontrolera. Ten sposób wydaje się łatwiejszy w realizacji i bardziej odpowiedni dla modułów z małą liczbą dostępnych portów GPIO, tak jak w przypadku ESP01. Będzie on wykorzystywany w dalszych opisach.

Schemat połączenia układu testowego

Do testów wykorzystana została płytka KA-NUCLEO-F411CE, miniaturowy wyświetlacz SSD1306 i oczywiście moduł ESP01. Płytkę KA-NUCLEO-F411CE okazała się wygodna w użyciu, ze względu na zamontowaną potrójną diodę LED, którą będziemy sterować. Jednak można użyć dowolnej płytki ewaluacyjnej z kontrolerem STM.

Wykaz użytych wyprowadzeń GPIO oraz dodatkowych połączeń umieszczono w **tabeli 1**, a na **rysunku 2** pokazano schemat ideowy połączeń między poszczególnymi elementami układu.

Tryby pracy modułu ESP01

Zanim przejdę do opisu przykładów wykorzystania modułu do zdalnego sterowania diodami LED, najpierw kilka słów o jego trybach pracy.

TRYB AP LUB STACJI (STACJI). W pierwszym trybie moduł pełni rolę punktu dostępowego Wi-Fi (AP), z którym mogą się łączyć inne urządzenia. W trybie AP moduł może obsługiwać komunikację z maksymalnie 4 urządzeniami. W trybie Stacji żeby mieć możliwość komunikacji z modułem, on sam musi być połączony z jakimś AP, najczęściej ruterem.

TRYB SERWERA. Ustawiony w tym trybie moduł będzie oczekiwał na nawiązanie połączenia Wi-Fi przez inne urządzenia – klientów.

PROTOKÓŁ TCP. Moduł można ustawić do pracy z protokołem TCP, sieciowym sposobem komunikacji, zapewniającym niezawodne przesyłanie danych. Protokół TCP jest podstawą http, czyli np. wymiany danych pomiędzy klientem a serwerem za pomocą wygodnej w użyciu przeglądarki internetowej. Korzystający z protokołu TCP serwer, nasłuchuje na wybranym porcie oczekując na transmisję od klienta. Po jej odebraniu może odpowiedzieć. Moduł ESP01 może także pracować z protokołem UDP. W ogólnych zarysach jego działanie jest podobne do TCP jednak pozbawiony jest dodatkowych mechanizmów gwarantujących dotarcie transmisji do adresata.

KOMENDY AT. Jest to standardowy zestaw rozkazów sterujących pracą modułu, np. wyborem trybu AP lub Stacji, wyborem protokołu, transmisją danych itd. Komendy AT do modułu będą wysyłane za pośrednictwem portu szeregowego przez zewnętrzny kontroler realizujący program użytkownika. Parametry transmisji po restarcie, jeżeli nie zostały wcześniej zmienione przez użytkownika, to: 115200 b/s, 8 bitów danych, bez parzystości. We wcześniejszych wersjach modułu domyślna szybkość transmisji wynosiła 9600 b/s.

ESP01 – praca lokalna

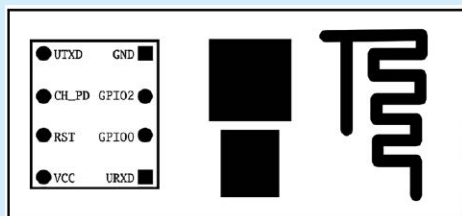
Na **rysunku 3** pokazano konfigurację, w której moduł komunikuje się bezpośrednio z urządzeniem sterującym. Moduł skonfigurowany jako AP pracuje w trybie serwera. Do sterowania jest używany protokół http. Po odebraniu transmisji od klienta (np. z przeglądarki telefonu) sterownik połączony z modułem generuje dynamicznie stronę internetową zawierającą aktualny stan diod LED (zapalone lub zgaszone). Poprzez moduł ESP01 dane strony przesyłane są do telefonu. Tam wyświetlane na uruchomionej przeglądarce internetowej. Z poziomu przeglądarki można sterować włączeniem lub zgaszeniem wybranych diod.

Sekwencja rozkazów AT, przygotowująca moduł ESP8266 do pracy w takiej konfiguracji jest następująca:

```
AT+CWMODE_CUR=3\r\n – mieszany tryb pracy modułu jako AP i Stacja.
```

```
OK\r\n – potwierdzenie wykonania rozkazu przez moduł.
```

```
AT+CIPMUX=1\r\n – moduł będzie pracował z wieloma klientami Wi-Fi.
```



Rysunek 1. Rozmieszczenie wyprowadzeń modułu ESP01 w widoku z góry

```
OK\r\n – odpowiedź modułu.
```

```
AT+CIPAP?\r\n – odczyt adresu IP modułu pracującego jako AP.  
+CIPAP:<IP> – odczytany adres może być użyty do wyświetlenia na wyświetlaczu.
```

```
OK\r\n – odpowiedź modułu.
```

```
AT+CIPSERVER=1,80\r\n – moduł będzie pracował jako serwer TCP nasłuchujący na porcie 80.
```

```
OK\r\n – odpowiedź modułu.
```

```
AT+CIPSTO=10\r\n – serwer zerwie połączenie po 10 sekundach nieaktywności klienta.
```

```
OK\r\n – odpowiedź modułu.
```

```
AT+CWSAP_CUR=\"Ki_ESP8266\", \"\", 1, 0, 4, 0\r\n – konfiguracja modułu jako AP: SSID „Ki_ESP8266”, brak hasła dostępu, kanał 1, tryb OPEN, możliwość pracy z maksymalnie 4 stacjami jednocześnie.
```

```
OK\r\n – odpowiedź modułu.
```

Od tej chwili moduł pracujący jako serwer oczekuje na transmisję od klienta.

ESP01 – odbiór danych

Klient chcący nawiązać połączenie z modułem w trybie AP i przelać do niego dane, powinien skorzystać z adresu IP, który został odczytany rozkazem *CIPAP?*. Jeśli klient korzysta z przeglądarki i protokołu http, nie musi wpisywać za IP: numeru portu, transmisja będzie domyślnie korzystała z portu 80 (ustawionego w module rozkazem *CIPSERVER*). Po odebraniu transmisji od klienta, moduł wyśle swoim portem szeregowym do zewnętrznego kontrolera parametry `+IPD,<ID>,<len><data>`, gdzie:

- `<ID>` identyfikator przyłączonego klienta, dziesiętnie 0..3,
- `<len>` rozmiar odebranych bajtów danych, dziesiętnie,
- `<data>` po znaku ‘:’ odebrane bajty danych.

ESP01 – transmisja odpowiedzi

W odpowiedzi na transmisję klienta, sterownik poprzez port szeregowy modułu przesyła dane przygotowanej strony do wyświetlenia na przeglądarce klienta. Sekwencja transmisji jest następująca `AT+CIPSEND=<link ID>,<length>`, gdzie:

- `<link ID>` identyfikator klienta, dziesiętnie
- `<length>` rozmiar bajtów danych do wysłania, dziesiętnie 1-2048

Po odebraniu z modułu znaku zachęty ‘>’, sterownik może odebrać dane do wysłania.

Sposoby sterowania za pomocą http diodami LED

Przy dwustronnej wymianie danych klient przesyła nowe ustawienia stanu diod do serwera, a ten – zwrótnie do klienta – przesyła ich aktualny stan. Korzystając z protokołu http najłatwiej posłużyć się w tym celu generowanymi przez sterownik serwera stronami do wyświetlenia na przeglądarce klienta, zawierającymi formularze w formie przycisków. Kod HTML takiego formularza może wyglądać następująco:

```
<form method=\"get\" action=\"index.shtml\">
  <input type=\"submit\" name=\"tag_f_out1\"
  value=\"R\" style=\"background-color:red;color:beige\">
  <input type=\"submit\" name=\"tag_f_out2\"
  value=\"G\" style=\"background-color:white;color:black\">
  <input type=\"submit\" name=\"tag_f_out3\"
  value=\"B\" style=\"background-color:blue;color:beige\">
</form>
```

Na ekranie będą wyświetlone 3 pola „input” typu „submit”, odpowiadające 3 diodom LED: R, G, B. Jeżeli na płytce sterownika

Tabela 1. Połączenia pomiędzy płytką KA-NUCLEO-F411CE a modułem ESP01

| Port kontrolera na płytce | Sygnat | Podłączenie |
|---------------------------|-----------------------------------|---|
| PB12 | Przycisk | Zamontowany na KA-NUCLEO-F4 |
| PB13 | Katoda diody B (niebieskiej) | Zamontowana na KA-NUCLEO-F4 |
| PB14 | Katoda diody R (czerwonej) | Zamontowana na KA-NUCLEO-F4 |
| PB15 | Katoda diody G (zielonej) | Zamontowana na KA-NUCLEO-F4 |
| PA9 | Wyjście TxD kontrolera | ESP01 wejście URXD interfejsu szeregowego |
| PA10 | Wejście RxD kontrolera | ESP01 wyjście UTXD interfejsu szeregowego |
| PB4 | Sterowanie zerowaniem modułu | ESP01 wejście RST |
| PB5 | Sterowanie obniżonym poborem mocy | ESP01 wejście CH_PD |
| - | 3,3 V | ESP01 zasilanie VCC |
| - | GND | ESP01 masa |
| PB8 | I ² C SCL | SSD1306 linia SCL |
| PB9 | I ² C SDA | SSD1306 linia SDA |
| - | AREF | SSD1306 zasilanie VCC |
| - | GND | SSD1306 masa |

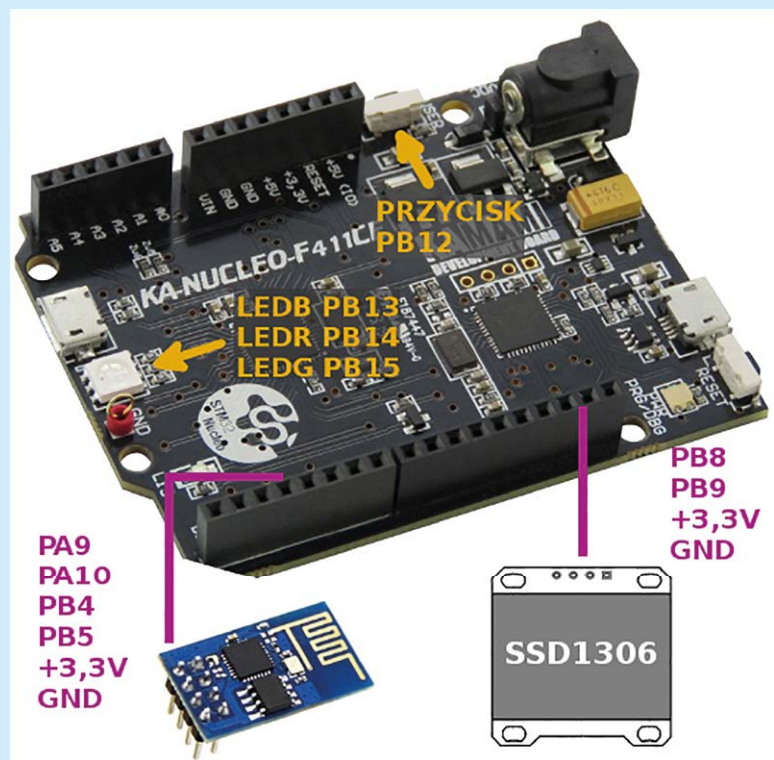
KA-NUCLEO są zaświecone diody R i B, odpowiadającym im polom „input” przypisane zostaną kolory „color:red” i „color:blue”. Ponieważ dioda G (zielona) jest zgaszona, środkowemu polu „input” odpowiadającemu tej diodzie przypisany zostaje kolor nieaktywny „color:white”. Dlatego kod strony musi być generowany dynamicznie, ponieważ zależy od aktualnego stanu diod.

Korzystając z formularzy, po każdorazowym dotknięciu któregoś z pól następuje ponowna transmisja w trybie GET od klienta do serwera z nazwą dotkniętego pola formularza. Np. w przypadku dotknięcia pola odpowiadającemu zmianie stanu diody LED-R wysłany zostanie komunikat widoczny na pasku stanu przeglądarki http://192.168.4.1/index.shtml?tag_f_out1=R. W otrzymanym komunikacie sterownik zidentyfikuje nazwę formularza „tag_f_out1=R”, zmieni stan diody na przeciwny i odeśle do przeglądarki utworzoną stronę do wyświetlenia.

Przesyłanie do sterownika SSID i hasła rutera

W podobny sposób korzystając z formularzy, prześlemy do sterownika nazwę SSID i hasło dostępu do rutera, z którego usług skorzystamy testując kolejną opcję trybu pracy modułu jako Stacji. W tej konfiguracji dostęp do modułu a poprzez niego sterowanie diodami będzie możliwe w obrębie sieci Ethernet, do której zostanie włączony za pośrednictwem rutera (rysunek 4). Ruter ethernetowy pracuje w sieci, do której są włączone również inne urządzenia. Do tej sieci poprzez, łącze Wi-Fi rutera, jest włączony także moduł ESP01. Moduł pracuje w trybie Stacja, a ruter pełni funkcję punktu dostępowego AP.

Aby móc się połączyć z ruterem, moduł musi „znać” jego SSID oraz hasło dostępu. Te informacje mogą być na stałe zaszyte w oprogramowaniu sterownika lub przekazane w inny sposób, przed połączeniem modułu z ruterem, np. poprzez formularze. W przygotowanym oprogramowaniu testującym, przełączenie trybu pracy modułu pomiędzy AP a Stacją, dokonuje się w momencie restartu. Jest wtedy sprawdzany stan przycisku na płytce KA-NUCLEO-F4. Jeżeli przycisk nie jest naciśnięty, moduł będzie pracował w połączeniach lokalnych jako punkt dostępowy AP. Jeżeli przycisk będzie wciśnięty, moduł przełączy się w tryb Stacja. Korzystając z przesłanych wcześniej informacji o SSID i hasle, będzie usiłował połączyć się ze wskazanym ruterem (AP). Jeżeli mu się to uda,



Rysunek 2. Schemat ideowy połączeń między poszczególnymi elementami

ruter przyzna modułowi adres IP. Posługując się nadanym adresem IP od tej chwili możliwa będzie komunikacja z modułem ESP01 w obrębie lokalnej sieci.

Sekwencja rozkazów AT realizująca opisane powyżej działania wygląda następująco:

```
AT+CWMODE=1,333\r\n – moduł będzie pracował z wieloma podłączonymi urządzeniami Wi-Fi.
OK\r\n
AT+CIPMUX=1,333\r\n – moduł będzie pracował jako serwer TCP nasłuchujący na porcie 333.
OK\r\n
AT+CWMODE_CUR=3\r\n – mieszany tryb pracy modułu jako AP i Stacja.
OK\r\n
AT+CWJAP_CUR=<SSID>,<PAS>\r\n – połączenie z ruterem gdzie <SSID> rutera w formacie "abcd", <PAS> hasło dostępu do sieci w formacie "1s56x".
```

OK\r\n

AT+CIPSTA?\r\n – odczyt IP nadanego przez ruter modułowi pracującemu jako Stacja.

+CIPSTA:<IP> – nadany IP np. 192.168.6.100 można pokazać na wyświetlaczu.

OK\r\n

Od tej chwili moduł pracujący jako serwer oczekuje na transmisję od klienta w ramach lokalnej sieci Ethernet.

Przydzielanie modułowi ESP01 adresu IP przez ruter

Najczęściej ruter przydzieli modułowi adres IP w trybie dynamicznym (gdy włączona jest w routerze opcja DHCP). Oznacza to, że za każdym razem IP może być inny. Gdybyśmy chcieli żeby adres IP za każdym połączeniem modułu do rutera był ten sam, trzeba sprawdzić, czy ruter nie dysponuje pulą adresów statycznych. Często tak jest. Wtedy na podstawie numeru MAC modułu należy wybrać i przypisać modułowi jeden z wolnych adresów statycznych. Zaznaczam, że przeprowadzenie tej operacji zależy od oprogramowania samego rutera.

Transmisja danych pomiędzy modułem ESP01 w trybie Stacja a terminalem TCP

W programie testowym, gdy moduł pracuje w trybie Stacja i jest dołączony do sieci Ethernet, zrezygnowałem z protokołu http do przesyłania danych sterujących diodami. Zamiast tego są przesyłane rozkazy sterujące w formacie transmisji TCP. Sterowanie za pomocą przeglądarki jest wygodne, jednak oznacza konieczność przesłania dużej ilości dodatkowych danych. „Czysta” transmisja TCP jest szybsza i bardziej stabilna. Zwłaszcza, jeżeli moduł będzie pracował w rozległej sieci. Zamiast przeglądarki do zdalnego sterowania należy użyć terminala TCP z funkcją klienta. Do tego celu wypróbowałem darmową aplikację **TCP/UDP Terminal**, pracującą pod kontrolą Androida. Z kolei, dla Windowsa i Linuksa wystarczył **Hercules_3-2-8**. Do sterowania użyłem uproszczonej składni rozkazów: R-T\r\n, G-T\r\n, B-T\r\n. Po odebraniu i rozpoznaniu każdej z 3 sekwencji, program sterujący przełącza na stan przeciwny odpowiednio diodę czerwoną, zieloną i niebieską. Dodatkowy rozkaz LED-T przełącza na stan przeciwny wszystkie diody. Potwierdzeniem wykonania rozkazu jest odsyłana przez sterownik i moduł sekwencja OK\r\n. Rozkaz RGB-? służy do odczytu aktualnego stanu diod.

Przesyłanie danych pomiędzy klientem a serwerem przebiega w sposób identyczny do wcześniej opisanego. Moduł odbiera dane poprzedzone sekwencją +IPD,<ID>,<len>:<data>. Z kolei transmisja odpowiedzi do klienta wymaga użycia rozkazu AT+CIPSEND=<link ID>,<length>.

ESP01 w rozległej sieci internetowej

Dostęp do modułu i sterowanie za jego pośrednictwem diodami z poziomu rozległej sieci internetowej pokazano schematycznie na **rysunku 5**. Jak poprzednio moduł pracuje w trybie Stacja i łączy się za pośrednictwem Wi-Fi z ruterem. Sekwencja rozkazów sterujących AT będzie taka sama, jak dla połączenia w obrębie sieci Ethernet. Na czym więc polega różnica?

Po pierwsze, ruter powinien mieć połączenie z Internetem. Po drugie, oprogramowanie rutera musi pozwalać na utworzenie tzw. „serwera wirtualnego”. W opcjach modemu należy odnaleźć taką, która będzie się nazywała „Virtual Server” (lub podobnie), a w jej ramach „Port Range Forwarding”. Jej działanie polega na przypisaniu do zadeklarowanego numeru portu określonego adresu IP w sieci Ethernet, w której pracuje ruter. Tym sposobem dowolna transmisja odwołująca się do zadeklarowanego numeru portu zostanie automatycznie przekierowana pod powiązany adres IP.



Rysunek 3. Konfiguracja, w której moduł komunikuje się bezpośrednio z urządzeniem sterującym

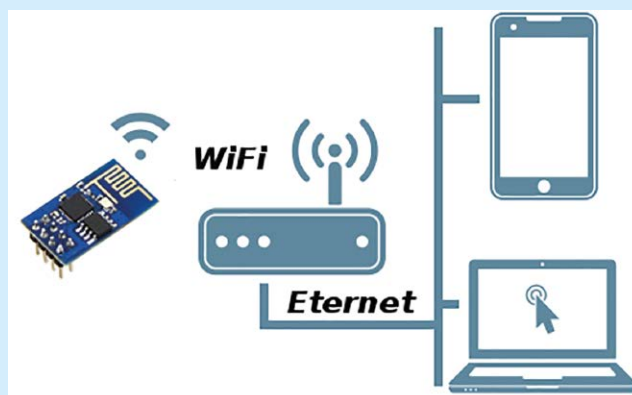
Ostatnią kwestią jest poznanie numeru IP, pod jakim w Internecie jest widziany nasz ruter. Zazwyczaj jest to możliwe po odszukaniu w ustawieniach statusu rutera pozycji WAN IP. Przykładowo, jeżeli nasz ruter otrzymał adres WAN IP 89.74.100.200, a w ramach serwera wirtualnego odwołanie do portu 333 będzie przekierowywane do modułu ESP01, zaadresowanie transmisji 89.74.100.200:333 spowoduje, że dotrze ona do modułu z dowolnej części Internetu. Tym samym stanie się możliwa transmisja danych sterujących diodami na sterowniku połączonym z modułem.

Ustawienia rutera

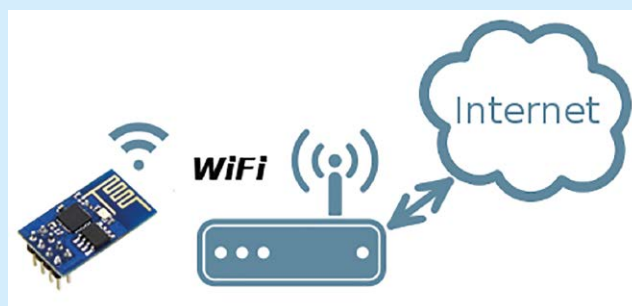
Aby była możliwa komunikacja pomiędzy modułem a ruterem, zwykle wystarczą ustawienia fabryczne tego ostatniego. Na wszelki wypadek podaję najważniejsze z nich:

- Wireless Working Mode: Wireless Access Point(AP).
- Network mode – tryb pracy sieci: 802.11g lub 11g mode lub 11/b/g/n mixed mode.
- Enable Wireless Security – włączone zabezpieczenia sieci bezprzewodowej.
- Security Mode: WPA2-PSK lub WPA-PSK/ WPA2-PSK.
- Security Option: Automatic.
- Encryption: AES.
- MAC address filter: Disabled.

Ryszard Szymaniak, EP



Rysunek 4. Włączenie modułu ESP01 do sieci Ethernet za pomocą rutera



Rysunek 5. Sposób włączenia modułu ESP01 do sieci Internet