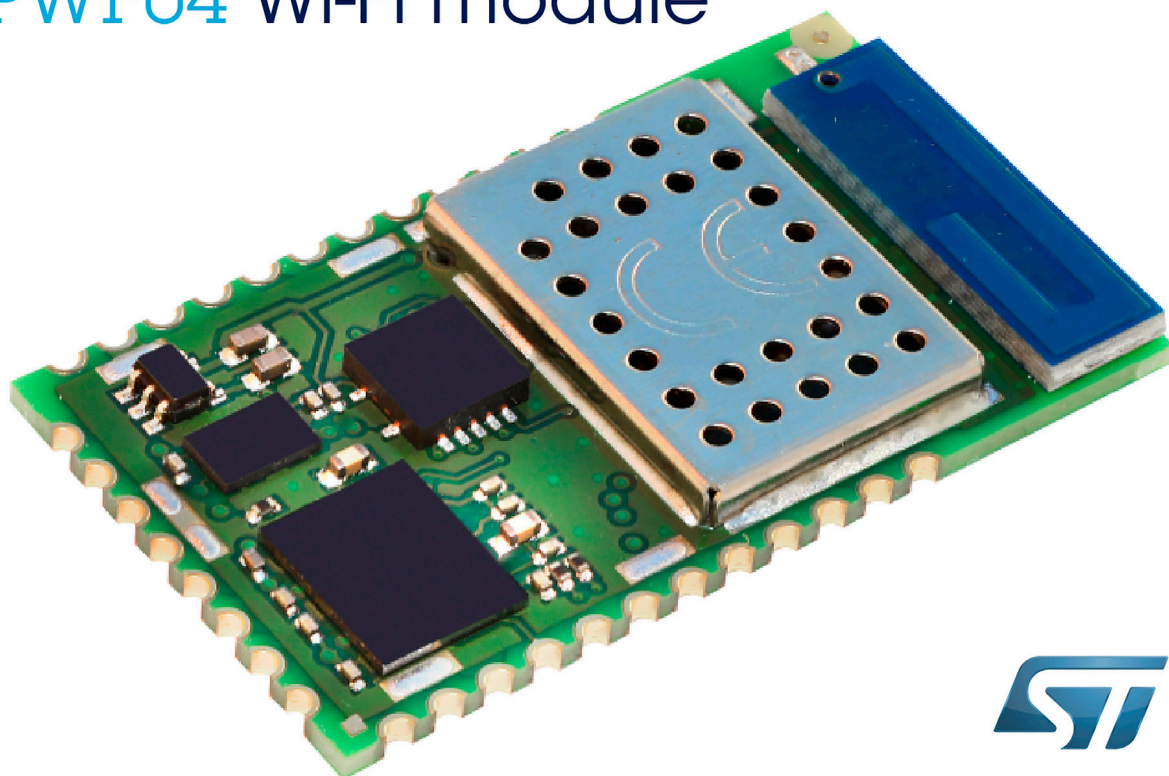


SPWF04 Wi-Fi module



Moduł Wi-Fi typu SPWF04SA

Parametry, możliwości i przykłady zastosowań

SPWF04SA jest modułem WiFi opracowanym i wyprodukowanym przez firmę STMicroelectronics. Na niewielkiej płytce zamontowano część radiową oraz mikrokontroler STM32F4 z rozbudowanym oprogramowaniem do zarządzania pracą modułu.

Dzięki zastosowaniu nowoczesnych komponentów, oprócz funkcji radiowych moduł obsługuje dodatkowe porty oraz interfejsy sprzętowe, takie jak: przetworniki A/C, C/A, generator PWM, interfejsy UART, SPI. Moduł może pracować autonomicznie lub być sterowany za pomocą jednego z interfejsów szeregowych przez urządzenie zewnętrzne.

Budowa mechaniczna modułu

Na jednej płytce drukowanej modułu zamontowano: część radiową, kontroler z dodatkowymi układami peryferyjnymi, antenę (w wersji SPWF04SC jest montowane gniazdo U.FL dla anteny zewnętrznej). Moduł jest przeznaczony do montażu powierzchniowego. Wyprowadzenia są dostępne na krawędzi płytki drukowanej. Ich raster pól wynosi 1,524 mm.

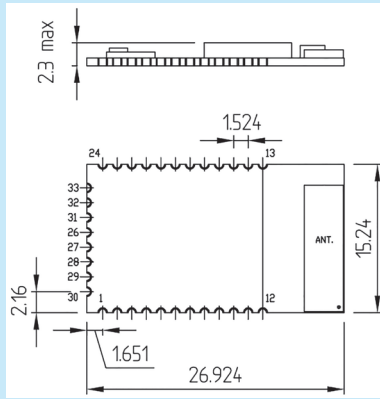
Najważniejsze parametry modułu SPWF04SA:

- Interfejsy komunikacyjne: UART, SPI, obsługa zewnętrznej karty SD.
- Interfejsy sprzętowe: 19 linii GPIO w tym 3 z funkcją wejścia przetwornika A/C, 1 z funkcją wyjścia PWM i 1 z funkcją wyjścia C/A, niektóre z tolerancją na napięcie +5 V.
- Sterowanie rozkazami AT lub językiem skryptowym MicroPython.
- Tryby pracy: Stacja, miniAP, web serwer ze stronami statycznymi lub generowanymi dynamicznie.
- Programowa obsługa stosu TCP/IP w tym protokołów TCP, UDP, HTTPS.
- Czulość odbiornika do -96 dBm.
- Maksymalna moc nadajnika 18,3 dBm.
- Minimalny pobór prądu w trybie standby 43 μ A.
- Średni pobór prądu w trybie nadawania 260 mA przy mocy nadajnika 10 dBm.
- Napięcie zasilania +3,3 V.
- Wymiary modułu: 26,92 mm×15,24 mm×2,35 mm.
- Temperatura pracy: od -40°C do +85°C.
- Certyfikaty: FCC, CE, IC.

Na **rysunku 1** pokazano rozmieszczenie wyprowadzeń wraz z zaznaczeniem najważniejszych wymiarów modułu.

Firmowe oprogramowanie modułu

Za funkcjonowanie modułu SPWF04SA odpowiada zamontowany na płycie mikrokontroler, a właściwie jego oprogramowanie firmowe, które zarządza wszystkimi funkcjami związanymi z obsługą części radiowej, programowego stosu TCP/IP, obsługą protokołów i interfejsów służących do komunikacji z układami zewnętrznymi. O jego złożoności może świadczyć **rysunek 2**. Pokazano na nim kolejne warstwy oprogramowania. Warstwa najniższa „Radio & Lower MAC” zawiera procedury odpowiedzialne bezpośrednio za sterowanie radiową częścią modułu. Poprzez warstwy pośrednie dochodzimy do procedur związanych z obsługą stosu TCP/IP. Wszystko opakowują procedury służące do wymiany danych ze światem zewnętrznym (interfejsy konsoli UART i SPI) oraz procedury do obsługi skryptów MicroPython, gdy moduł pracuje w trybie autonomicznym. Na **rysunku 3** pokazano blokowy diagram obsługiwanych przez programowy stos TCP/IP protokołów.



Rysunek 1. Wymiary i wyprowadzenia modułu SPWF04SA

Sterowanie pracą modułu

Pracą modułu można sterować na kilka różnych sposobów, za pomocą:

- interfejsu UART i tekstowych poleceń AT,
- interfejsu SPI,
- skryptów MicroPythona.

TRANSMISJA ROZKAZÓW AT ZA POMOCĄ UART

Ogólna składnia tak wysyłanych rozkazów jest następująca:

AT+S.<rozkaz><parametry><CR>[dane], gdzie:

- <rozkaz> jest nazwą przesyłanego rozkazu np. AT+S.HELP<cr>,
- <parametry> opcjonalnie zależnie od typu rozkazu przesyłany jest parametr, np. nazwa rejestru lub wartość liczbowa,
- <CR> kod końca rozkazu (heksadecymalnie 0x0D),
- [dane] opcjonalnie zależnie od typu rozkazu przesyłane są dodatkowe dane.

TRANSMISJA ROZKAZÓW ZA POMOCĄ SPI

Moduł pracuje w trybie Slave. Ogólna składnia rozkazów jest następująca:

0x02<rozmiar.przeslania><kod.

roz><ile_param><rozmiar.1param><1.

param>...<rozmiar.n.param><n.param>, gdzie:

- 0x02 bajt startu (heksadecymalnie 0x02),
- <rozmiar.przeslania> 2 bajty określające łączną długość kodu rozkazu,
- <kod.roz> bajt kodu rozkazu odpowiadający rozkazowi typu AT,
- <ile_param> bajt ilości parametrów rozkazu,
- <rozmiar.1param> bajt określający długość (ilość bajtów) 1-go parametru,
- <1.param> bajty 1 parametru,
- ... itd. dla kolejnych parametrów

Przykład:

Rozkaz „AT+S.SCAN=d,/scan.txt” przesyłany za pomocą interfejsu SPI ma postać „0x02 0x00 0x0D 0x33 0x02 0x01 d 0x09 (/scan.txt)”.

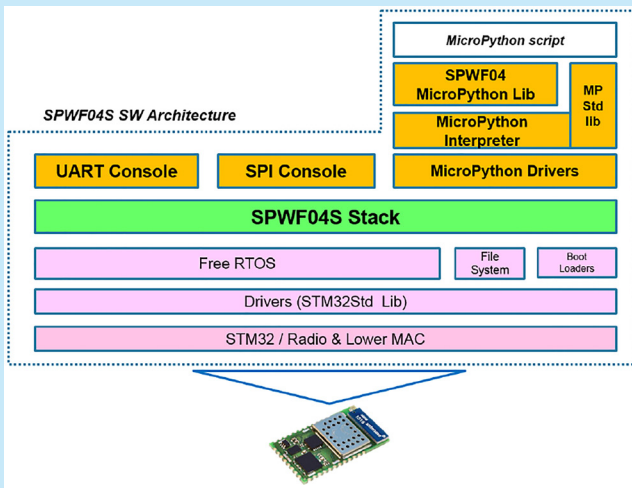
SKRYPTY

W tym trybie pracą modułu sterują polecenia skryptu MicroPython-a, który jest zapisany w pamięci Flash mikrokontrolera. Skrypt może być uruchamiany z poziomu rozkazów AT lub automatycznie. Uruchamianie automatyczne po włączeniu zasilania wymaga wskazania położenia skryptu do uruchomienia (ustawianego w zmiennej python_script opisanej w dokumencie *en.DM00329655.pdf*) i ustawienia wyprowadzenia GPIO(8) w stanie wysokim podczas inicjacji pracy modułu.

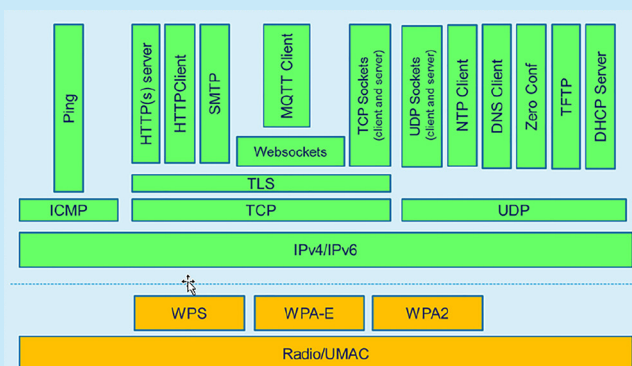
Testowanie modułu w trybie AT

Żeby przetestować sterowanie modułu SPWF04SA rozkazami AT, potrzebna będzie konfiguracja układów pokazana na **rysunku 4**. Do interfejsu UART modułu należy podłączyć konwerter USB⇌UART. Do testów wystarczy zasilanie modułu z portu USB. Ponieważ USB dostarcza napięcia 5 V a moduł pracuje przy zasilaniu +3,3 V, potrzebny będzie stabilizator (na wejściu i wyjściu stabilizatora należy zamontować kondensatory odsprzęgające o wartości 1-10 µF). Do wyprowadzeń SPI modułu warto podłączyć złącze dla karty SD lub mikroSD. Zestawienie wszystkich niezbędnych połączeń w **tabeli 1**.

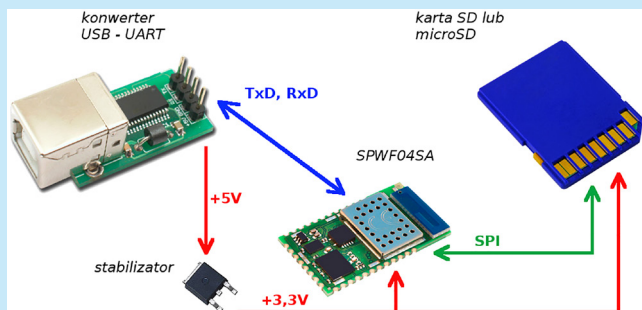
Do testów potrzebny będzie jeszcze uruchamiany na komputerze program typu terminalowego np. Br@y Terminal+. Jest to program terminala tekstowego, który pozwoli na przesyłanie łączem UART modułu rozkazów AT oraz odbieranie z modułu wysyłanych odpowiedzi i komunikatów.



Rysunek 2. Budowa stosu programowego zastosowanego w SPWF04SA



Rysunek 3. Protokoły obsługiwane przez firmware modułu SPWF04SA



Rysunek 4. Konfiguracja sprzętu służąca do testowania modułu SPWF04SA

Pełna lista rozkazów AT dla obecnej wersji oprogramowania modułu, zamieszczona jest w dokumencie *en.DM00329655.pdf*. Jest ona częściowo zgodna z wersjami dla wcześniej produkowanego typu modułu SPWD01xx. Jednak ze względu na nowe funkcjonalności, różni się w pewnych partiach zarówno co do zawartości, jak i formatu poszczególnych rozkazów.

Podane dalej przykłady składni rozkazów AT sterujących wybranymi funkcjami modułu. Rozkazy będą wpisywane ręcznie w edytorze terminala Br@y i przesyłane portem UART do modułu. Tą samą drogą będą odbierane potwierdzenia i komunikaty wysyłane z modułu.

Każdy rozkaz musi być zakończony kodem CR (powrót karetki). Ponieważ w terminalu Br@y można ustawić opcję jego automatycznego dodawania na końcu wysyłanych danych, w przykładach ten kod będzie pomijany.

Manipulowanie stanem GPIO

W przypadku cyfrowych portów GPIO najpierw należy ustawić ich funkcję – wyjście lub wejście. Kolejne rozkazy ustawiają lub zerują wyprowadzenie GPIO2 pracujące jako wyjście:

```
AT+S.GPIOC=2,out – GPIO2 jako wyjście
AT+S.GPIOW=2,1 – GPIO2 ustawione
AT+S.GPIOW=2,0 – GPIO2 wyzerowane
```

Przykład odczytu stanu GPIO13 pracującego jako wejście. Po odebraniu rozkazu odczytu, moduł przesyła komunikat zawierający informację o stanie portu. Kolejne cyfry oznaczają: numer portu, odczytany stan z jego wejścia 1 lub 0, tryb pracy: 1-GPIO jest wejściem, 0-GPIO jest wyjściem.

```
AT+S.GPIOC=13,in – GPIO13 jako wejście
AT+S.GPIOR=13 – odczyt poziomu na wejściu GPIO13
Komunikat o wysokim poziomie logicznym na wejściu GPIO13
– „AT+S.Value:13:1:1”
AT+S.GPIOR=13
```

Komunikat o niskim poziomie logicznym na wejściu GPIO13
– „AT+S.Value:13:0:1”

W kolejnym przykładzie GPIO13 zaprogramowano jako wejście generujące przerwania. Zależnie od ustawienia, wejście będzie reagowało na określone zbocze sygnału wejściowego: „F” – opadające, „R” – narastające, „B” – na obydwie. Detekcja ustawionego stanu zbocza spowoduje wysłanie portem UART do terminala informacji o zaistniałym przerwaniu:

```
AT+S.GPIOC=13,in,F – GPIO13 jako wejście generujące przerwanie od opadającego zbocza sygnału
Komunikat przesłany przez UART po wystąpieniu przerwania
– „+WIND:52:GPIO Interrupt:13:0”.
```

W następnym przykładzie jeden z trzech wyposażonych w taką funkcjonalność portów GPIO1, jest inicjowany do pracy jako wejście przetwornika A/C. Po tym rozkazie moduł w odpowiedzi przesyła komunikat z liczbą wyrażającą w miliwoltach zmierzone na wejściu portu napięcie analogowe. Zakres napięcia wejściowego wynosi 0...3300 mV:

Tabela 1. Konfiguracja połączeń testowych

SPWF04SA wyprowadzenie	Sygnał	Podłączenie
24	VCC +3,3 V	Z wyjścia stabilizatora
23	GND	Masa
8	RxD	Z wyjścia TxD konwertera (tolerowane poziomy 0-5 V)
6	TxD	Z wejścia RxD konwertera (tolerowane poziomy 0-5 V)
1	SPI_CS (GPIO.3)	Karta SD pin 1 (CD/DAT3) Karta microSD pin 2
31	SPI_CLK	Karta SD pin 5 (CLK) Karta microSD pin 5
32	SPI_MISO	Karta SD pin 7 (DAT0) Karta microSD pin 7
33	SPI_MOSI	Karta SD pin 2 (CMD) Karta microSD pin 3
		Karta SD pin 4 (+3,3 V) Karta SD pin 3,6 (GND) Karta microSD pin 4 (+3,3 V) Karta microSD pin 6 (GND)

AT+S.ADC=1 – ustawienie GPIO1 w trybie wejścia przetwornika A/C i inicjacja konwersji

Komunikat o wyniku konwersji – „AT-S.Value:2300” (2300 mV).

Przykład pokazujący jak na porcie GPIO15 mogąym pracować jako wyjście przetwornika C/A, ustawić napięcie analogowe w zakresie 1...3300 mV. Wartość 0 oznacza zakończenie pracy GPIO15 jako wyjścia przetwornika:

AT+S.DAC=500 – GPIO15 jako wyjście przetwornika C/A o napięcie 500 mV

Manipulacja plikami

Moduł umożliwia dostęp do systemu plików znajdujących się w pamięci Flash, pamięci RAM i na karcie SD. Przy czym dodawanie i usuwanie plików jest możliwe tylko dla pamięci RAM i karty SD.

- Przykład utworzenia w głównym katalogu na karcie SD pliku „Test.txt” o długości 13 znaków tekstowych (#013 – polecenie dla terminala Br@y wstawienia liczby dziesiętnej 13): „AT+S.FSC=0:/Test.txt,13#013Napis testowy”.
- Dodanie do istniejącego pliku na karcie SD, 14 znaków tekstu „AT+S.FSC=0:/Test.txt,14#013_kolejny_napis”.
- Odczyt pliku zapisanego pliku na karcie SD i przesłanie jego zawartości z przesunięciem względem początku o 0 bajtów za pomocą UART „AT+S.FSP=0:/Test.txt,0”.
- Usunięcie pliku z karty SD „AT+S.FSD=0:Test.txt”.

Modułu SPWF04SA w trybie stacji przyłączonej do punktu dostępowego

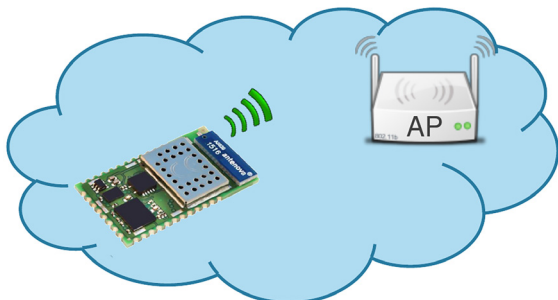
Moduł w trybie stacji może pracować w sieci utworzonej przez znajdujący się w zasięgu punkt dostępowy (AP). Taką sytuację pokazano na **rysunku 5**. Poleceniem AT+S.SCAN można uzyskać listę będących w zasięgu AP. Ustawiamy nazwę (SSID) wybranego punktu dostępu i jeżeli nie jest to AP typu otwartego, także klucz. Po wybraniu trybu pracy modułu „stacja” i włączeniu Wi-Fi moduł powinien zostać podłączony do AP i w trybie DHCP powinien mu zostać nadany numer IP:

AT+S.WIFI=1 – włączenie Wi-Fi

AT+S.SCAN=d, – przesłanie za pomocą UART listy punktów dostępu będących w zasięgu

AT+S.WIFI=0 – wyłączenie Wi-Fi

AT+S.FCFG – przywrócenie ustawień fabrycznych

SPWF04SA
jako StacjaPunkt dostępu
AP**Rysunek 5. Konfiguracja modułu w trybie stacji dołączonej do AP**

AT+S.SSIDTXT=nazwa _ AP – ustawienie SSID punktu dostępu
 AT+S.SCFG=wifi _ wpa _ psk _ text,123456 – ustawienie klucza do punktu dostępu

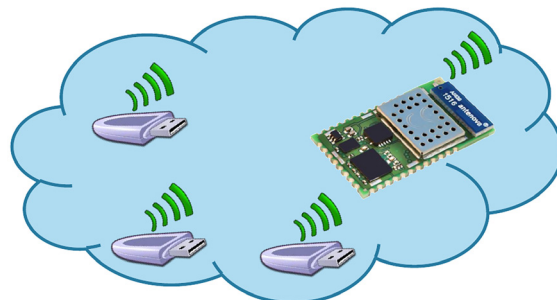
AT+S.SCFG=wifi _ priv _ mode,2 – network privacy mode
 AT+S.SCFG=wifi _ auth _ type,0 – authentication type
 AT+S.SCFG=wifi _ mode,1 – SPWF04SA pracuje jako stacja
 AT+S.WIFI=1 – włączenie Wi-Fi
 AT+S.WCFG – zapis konfiguracji do pamięci nieulotnej
 AT+S.RESET – zerowanie modułu
 AT+S.STS – odczyt statusu

Komunikat przesłany przez moduł – „AT-S.
 Var:ip _ ipaddr=192.168.1.102”

W ustawieniach fabrycznych, moduł pracuje jako serwer HTTP z zaprogramowanymi w pamięci Flash przykładowymi stronami. Po wpisaniu w pasku statusu przeglądarki przydzielonego adresu zostanie wyświetlona strona startowa.

Praca modułu w trybie miniAP (punkt dostępowy)

Moduł SPWF04SA może też pracować w trybie punktu dostępowego, do którego będą się przyłączały inne urządzenia Wi-Fi pracujące w trybie stacji. Taką sytuację pokazano schematycznie na **rysunku 6**. W tym trybie nadaje się modułowi nazwę (SSID) pod jaką będzie widoczny dla innych urządzeń w sieci Wifi. Jeżeli moduł jako AP nie ma pracować w trybie otwartym, należy określić parametry klucza dostępu, który każda stacja będzie musiała podać zanim uzyska status podłączonej. Należy także określić adres IP pod jakim AP będzie dostępny w sieci. Jeżeli włączony będzie tryb DHCP, każdej podłączonej do modułu stacji będzie

SPWF04SA jako
punkt dostępu**Rysunek 6. Konfiguracja modułu jako AP**

nadawany numer IP. Na koniec należy zapisać ustawienia do pamięci nieulotnej tak, aby każdorazowo po włączeniu zasilania modułu, były odtwarzane.

AT+S.WIFI=0 – wyłączenie WiFi

AT+S.FCFG – przywrócenie ustawień fabrycznych

AT+S.SSIDTXT=modul _ AP – nadanie nazwy modułowi pracującemu w trybie AP

AT+S.SCFG=wifi _ priv _ mode,1 – dostęp do AP w trybie WEP po podaniu klucza

AT+S.SCFG=wifi _ mode,3 – tryb MiniAP

AT+S.SCFG=wifi _ wep _ keys[0],76543210ab – klucz dostępu do AP np. „76543210ab”

AT+S.SCFG=wifi _ wep _ key _ lens,05 – rozmiar klucza w bajtach

AT+S.SCFG=wifi _ auth _ type,0

AT+S.SCFG=ip _ ipaddr,192.168.0.50 – adres IP, pod którym będzie dostępny AP

AT+S.SCFG=ip _ use _ dhcp,2 – tryb DHCP, automatyczne przydzielanie numerów IP

AT+S.WIFI=1 – włączenie Wi-Fi

AT+S.WCFG – zapamiętanie konfiguracji w pamięci nieulotnej

AT+S.RESET – zerowanie modułu

W ustawieniach fabrycznych, moduł pracuje jako serwer HTTP z zaprogramowanymi w pamięci Flash przykładowymi stronami. Po podłączeniu do modułu poprzez sieć WiFi np. telefonu, otwieramy jego przeglądarkę internetową i w pasku wyszukiwania wpisujemy ustawiony adres IP (w przykładzie 192.168.0.50). Powinna wyświetlić się strona testowa.

Ryszard Szymaniak, EP

www.ep.com.pl/kap

Klub Aplikantów Próbek to inicjatywa redakcji Elektroniki Praktycznej.

W kontaktach z firmami redakcja często otrzymuje do przetestowania próbki podzespołów, modułów, a nawet całych urządzeń elektronicznych. Są to zwykle najnowsze typy / modele produktów na rynku. Z chęcią podzielenia się z Czytelnikami tymi próbkami zrodziła się inicjatywa pod nazwą Klub Aplikantów Próbek. Członkiem KAP staje się każdy, kto zgłosi chęć przetestowania próbki. Wykaz i krótki opis próbek, którymi dysponuje redakcja EP, można znaleźć poniżej (www.ep.com.pl/KAP). Wystarczy wybrać rodzaj próbek i zwrócić się majlmem (na adres: Szef Pracowni Konstrukcyjnej grzegorz.becker@ep.com.pl) z prośbą o przesłanie bezpłatnych próbek, podając ich nazwę i adres wysyłki. Warto dopisać jaki jest plan zastosowania tych próbek. Nie jest to konieczne, ale może mieć znaczenie przy podziale próbek w przypadku większej liczby zgłoszeń. Mile widziane, choć nieobowiązkowe, jest też przystanie do redakcji EP opisu wykonanej aplikacji próbek, oczywiście po jej wykonaniu z zastosowaniem otrzymanej próbki. Autorom przystanych opisów przyznamy punkty, które będą im dawały pierwszeństwo przy ubieganiu się o kolejne próbki. Najciekawsze opisy aplikacji opublikujemy na forum ep.com.pl lub na łamach Elektroniki Praktycznej. Dla pełnej jasności jeszcze raz podkreślamy, że próbki przekazujemy bezpłatnie i nie trzeba ich zwracać do redakcji.

