

# Moduły GSM w praktyce, część 1

Typową konfigurację systemu wykorzystującego moduł GSM przedstawiono na rys. 1. Moduł pełni tu rolę zwykłego modemu, sterowanego przez zewnętrzny mikrokontroler stanowiący serce całego urządzenia. Takie podejście zapewnia dużą elastyczność, ponieważ możemy wybrać najbardziej dla nas odpowiedni mikrokontroler. Również oprogramowanie możemy stworzyć wykorzystując posiadane dla niego biblioteki i własne doświadczenie. Zastosowanie mikrokontrolera oraz oddzielnego modułu GSM, o ile jest w miarę wygodne, o tyle nie zawsze jest uzasadnione ekonomicznie. Decyduje o tym często konieczność zastosowania wielu dodatkowych elementów. Wydaje się oczywiste, że w każdym module GSM znajduje się mikrokontroler sterujący, posiadający dość dużą moc obliczeniową. Nasuwa się więc pytanie, czy nie dałoby się z tej mocy skorzystać na cele własnej aplikacji? W standardowych modułach, a tym bardziej w telefonach GSM nie jest to praktycznie możliwe, chociażby z powodu braku odpowiednich linii IO nadających się do sterowania zewnętrznymi urządzeniami. Także oprogramowanie modułu najczęściej nie pozwala na wykonanie kodu użytkownika. Producenci modułów GSM szybko dostrzegli jednak zapotrzebowanie rynku i jak grzyby po deszczu zaczęły powstawać rozwiązania modułów GSM umożliwiających wykonywanie programu

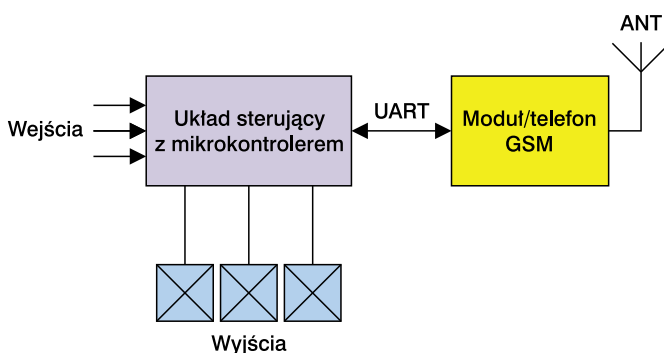
*Telefonia GSM w układach automatyki i telemetrii już od kilkunastu lat jest powszechnie wykorzystywana na szeroką skalę. Na łamach Elektroniki Praktycznej wielokrotnie publikowano projekty wykorzystujące do komunikacji telefon lub moduł GSM. We wszystkich spotykanych do tej pory rozwiązaniach rolę układu sterującego najczęściej pełnił niezależny mikrokontroler komunikujący się z telefonem/modułem GSM przez port szeregowy wykorzystując odpowiednie komendy AT.*

użytkownika. Dodatkowo zostały wyposażone w porty IO i magistrale spotykane w standardowych mikrokontrolerach, takie jak: I<sup>2</sup>C, SPI, UART itp. Wykorzystanie takiego modułu umożliwia realizację większości najczęściej spotykanych zadań przez sam moduł bez konieczności używania dodatkowego mikrokontrolera. Po prostu moduł GSM pełni rolę mikrokontrolera sterującego całym urządzeniem. W związku z tym dla tego typu modułów zwykło się używać określenia „Wireless CPU”, czyli bezprzewodowy procesor. Wykorzystanie modułu do wykonywania własnego programu wiąże się z koniecznością dostosowania programu do systemu operacyjnego wbudowanego w moduł GSM. Musimy w takim przypadku pamiętać, że zadaniem procesora modułu jest przede wszystkim wykonywanie swoich własnych zadań, jak np. obsługa stosu GSM. Do tworzenia oprogramowania zgodnie z powyższą koncepcją konieczne jest korzystanie z oprogramowania SDK (*Software Development Kit*) dostarczanego przez producenta modułu. W zależności od modułu najczęściej używane języki programowania to: Java (Motorola), Python (TELIC), C (WaveCom). Dwa pierwsze języki są bardzo wygodne dla programisty, ponieważ zapewniają wiele mechanizmów

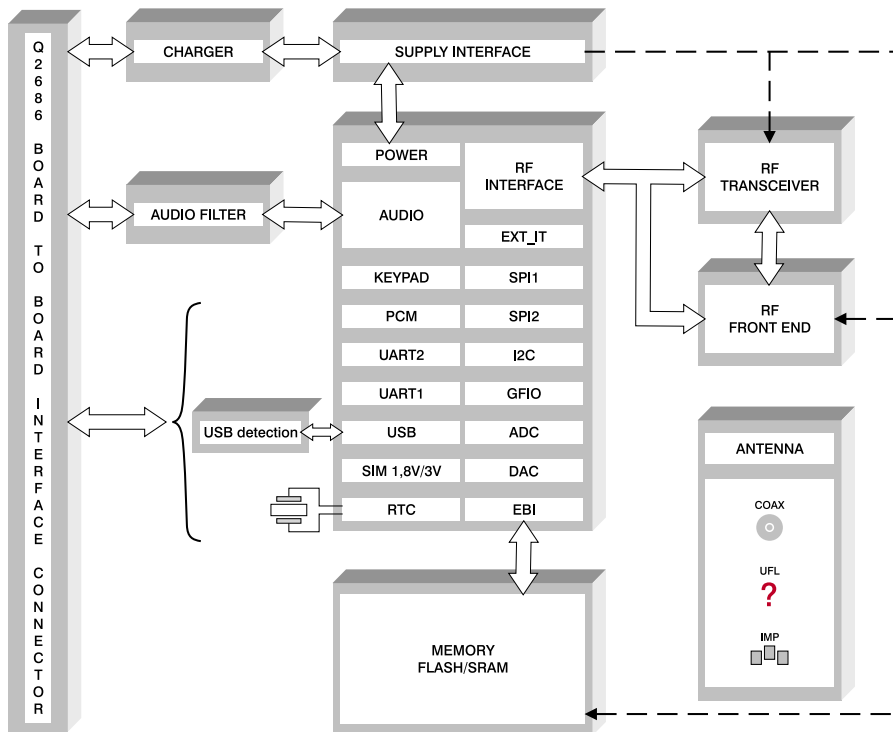
i bibliotek, uwalniając programistę od troszczenia się o szczegóły. Jednak są one stosunkowo wolne, ponieważ Python jest językiem interpretowanym, natomiast Java jest kompilowana do kodu bajtowego i wykonywana przez maszynę wirtualną. Największą wydajność naturalnie pozwala uzyskać rozwiązanie wykorzystujące język C, który jest kompilowany bezpośrednio do kodu maszynowego procesora modułu. Jednak również i w tym przypadku nie uzyskujemy maksymalnej wydajności aplikacji z uwagi na dzielenie czasu procesora pomiędzy naszą aplikacją, a obsługą zadań systemowych modułu. W prostych zastosowaniach nie jest jednak potrzebna duża wydajność, natomiast liczy się wygoda oraz szybkość tworzenia aplikacji.

W niniejszym cyklu będziemy się opierać na module GSM WaveCom Q2686, umożliwiającym tworzenie własnych rozwiązań aplikacyjnych bazujących na samym module oraz dodatkowych układach peryferyjnych. Moduł ten można programować z wykorzystaniem środowiska OpenAT bezpośrednio w języku C, a zastosowany w module mikroprocesor ARM9 umożliwia wydajne wykonywanie własnego programu.

Na wstępie zapoznamy się z charakterystyką i możliwościami modułu, a następnie ze specyfikacją elektryczną oraz sposobem projektowania własnego układu wykorzystującego moduł WaveCom. Następnie poznamy sposób instalacji środowiska OpenAT oraz metody wgrywania oraz uruchamiania oprogramowania do modułu OpenAT. Na za-



Rys. 1. Typowa konfiguracja systemu wykorzystującego moduł GSM



Rys. 2. Budowa wewnętrzna modułu Q2668

kończenie napiszemy kilka przykładowych programów zapoznających Czytelnika ze środowiskiem. Na początek będzie to prosty program zapalający diody LED, następnie pokażemy, w jaki sposób możemy tworzyć własne komendy AT oraz przykładową aplikację umożliwiającą sterowanie za pomocą SMS-ów. Po zapoznaniu się z całością materiałów zainteresowani Czytelnicy będą mogli bardziej szczegółowo zgłębić tajniki środowiska OpenAT, co pozwoli im pisać bardziej zaawansowane programy.

### Charakterystyka modułu WaveCom Q2668 oraz środowiska OpenAT

Procesor zastosowany w module Q2668 to ARM946E-S taktowany zegarem 100 MHz, zapewniający dużą moc obliczeniową zarówno dla aplikacji użytkownika, jak i systemu operacyjnego. W zależności od wersji moduł jest wyposażony w 4 lub 8 MB pamięci Flash oraz 2 lub 4 MB pamięci RAM. Zapewnia to odpowiednią ilość pamięci nawet dla bardzo zaawansowanych aplikacji użytkownika. Oczywiście część pamięci Flash i RAM jest zarezerwowana dla systemu operacyjnego WaveCom OS w wersji 6.60, tak więc do dyspozycji aplikacji użytkownika pozostaje 832 kB pamięci Flash i 256 kB pamięci RAM.

Aby moduł mógł być wykorzystany do sterowania różnymi urządzeniami peryferyjnymi konieczne stało się wyprowadzenie odpowiednich portów IO. Budowę wewnętrzną modułu przedstawiono na rys. 2.

Moduł wyposażono w szereg układów peryferyjnych ułatwiających komunikację z otoczeniem oraz tworzenie własnych aplikacji. Posiada on 36 linii wejścia-wyjścia (IO) ogólnego przeznaczenia, do których można podłączyć dowolne sygnały. Zaimplementowano w nim również szereg sprzętowych interfejsów, które są obsługiwane całkowicie sprzętowo. Są to:

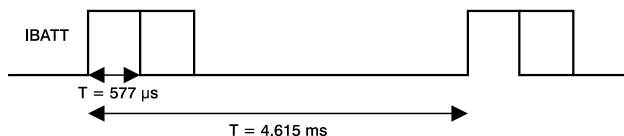
- interfejs USB Device w wersji 1.1,
- 2 porty szeregowy UART,
- interfejs dźwięku cyfrowego DAI,
- 2 interfejsy dźwięku analogowego,
- wyjście sygnalizatora dźwiękowego (buzera),
- 2 wejścia przerwań,
- sprzętowy interfejs klawiatury 5x5,
- interfejs karty SIM w standardzie 1,8 V lub 3 V,
- interfejs I<sup>2</sup>C,
- 2 interfejsy SPI,
- 10-bitowy przetwornik A/C,
- 10-bitowy przetwornik C/A,
- zegar czasu rzeczywistego RTC.

Należy tutaj zaznaczyć, że pierwszy port szeregowy jest wy-

korzystywany tak, jak klasyczny interfejs RS232 w zwykłych modułach GSM, czyli za jego pośrednictwem można wydawać komendy AT do modułu. Interfejs ten jest również wykorzystywany do wgrывania oraz uruchamiania własnych aplikacji. Jedno z wejść dźwięku analogowego umożliwia dołączenie za pośrednictwem kilkunastu elementów mikrofonu elektretowego oraz głośnika, umożliwiając w ten sposób bezpośrednią realizację połączeń głosowych. Interfejsy SPI i I<sup>2</sup>C powszechne w typowych mikrokontrolerach, takich jak AVR czy LPC2000 w znaczący sposób ułatwiają komunikację z różnymi układami zewnętrznymi. Wszystkie wymienione wyżej elementy sprawiają, że moduły WaveCom mogą zastąpić niemal dowolny mikrokontroler.

Moduł WaveCom jest przystosowany do pracy w 4 pasmach 850/900/1800/1900 MHz, tak więc może być bez problemów wykorzystywany zarówno w krajach europejskich jak i w Stanach Zjednoczonych. Q2668 zapewnia cyfrową transmisję danych CSD z prędkością 14,4 kbps oraz pakietową transmisję danych GPRS w klasie 10, co w najbardziej sprzyjających warunkach pozwala uzyskać prędkości transmisji na poziomie 80 kbps podczas odbioru oraz 40 kbps podczas nadawania danych. Jak więc widzimy, moduł jest pełnym terminalem GSM, który oprócz możliwości wykonywania wcześniej wspomnianych programów użytkownika może być wykorzystywany jak klasyczny moduł GSM sterowany za pomocą odpowiednich komend AT wysyłanych poprzez pierwszy port szeregowy. Moduł posiada bardzo małe wymiary 40x32,2x4 mm i waży tylko 9 g, może więc być wykorzystywany nawet w urządzeniach o bardzo niewielkich wymiarach.

Znamy już podstawowe możliwości sprzętowe modułu, jednak sam sprzęt to nie wszystko. Aby uruchomić własną aplikację musimy mieć przygotowany program, który następnie trzeba jakoś wgrać do modułu. Producent modułu WaveCom dostarcza zintegrowane środowisko programistyczne OpenAT, które umożliwia wygodne pisanie oprogramowania w języku C. W większej części środowisko OpenAT jest zbudowane



Rys. 3. Prąd zasilania pobierany przez moduł Q2668 w trakcie pracy

w oparciu o oprogramowanie Open Source, na które składa się kompilator GCC dla mikrokontrolerów ARM oraz środowisko IDE Eclipse. Pozostałe elementy, takie jak biblioteki systemowe oraz automatyczne kreatory projektów stanowią rozwiązania autorskie firmy WaveCom. Niestety, pomimo iż środowisko korzysta z programów Open Source, to działa ono tylko w systemie Windows i producent nie dostarcza oficjalnie wersji przeznaczonych dla Linuxa. Jednak mam dobrą wiadomość: wykorzystując kompilator GCC oraz Eclipse w wersji dla Linuxa, po przeniesieniu plików nagłówkowych i bibliotek oraz wykorzystując środowisko emulacji Windows Wine do uruchamiania kreatorów Open AT, możemy również bezproblemowo tworzyć oprogramowanie dla Modułów WaveCom pod Linuxem. Środowisko OpenAT wspiera pisanie oprogramowania w IDE Eclipse tylko w sposób opcjonalny i jest ono nastawione głównie na wykorzystywanie IDE Microsoft Visual Studio. Możliwe jest tu pełne debugowanie i praca krokowa, jednak nie każda firma, czy tym bardziej osoba zajmująca się tematem hobbystycznie, ma ochotę płacić niemałe pieniądze firmie z Redmond na zakup środowiska Microsoft Visual Studio, tylko po to, żeby można było jednym kliknięciem uruchamiać i debugować aplikację OpenAT. Ponadto, tak naprawdę cała kompilacja i tak przebiega z wykorzystaniem Open Source'owego kompilatora. Dalsza część praktyczna artykułu będzie więc bazowała na wykorzystywaniu alternatywnego, otwartego środowiska IDE Eclipse. Samo wgranie i uruchamianie skompilowanego programu w module Q2668 jest niezależne od platformy, ponieważ moduł możemy programować poprzez pierwszy port szeregowy w standardowym protokole X-Modem, wykorzystując dowolny program terminalowy, np. Minicom czy Hyperterminal.

### Parametry elektryczne – moduł Q2668 we własnym układzie aplikacyjnym

Jak już wspomniano na wstępie, moduł Q2668

dzięki szeregowi portów IO oraz innych urządzeń peryferyjnych pozwala zrezygnować z dodatkowego mikrokontrolera w urządzeniu. Aby można było skorzystać ze wszystkich sygnałów, potrzebne jest dodatkowe złącze zewnętrzne. Do wyprowadzenia sygnałów z modułu służy 100-pinowe złącze NAIS AXK600354J, które posiada raster wyprowadzeń 0,5 mm, co nie powinno stwarzać problemów nawet przy ręcznym montażu. Moduł posiada osobne złącze U.FL służące do podłączenia anteny GSM pod spodem modułu.

Jeżeli nie mamy odpowiedniego złącza lub złącze anteny chcemy wyprowadzić na zewnątrz obudowy urządzenia, to kabel antenowy (RG178) możemy również podłączyć do pola lutowniczego znajdującego się na górnej stronie modułu.

Moduł powinien być zasilany napięciem nominalnym 3,6 V (3,2...4,8 V), przy czym maksymalny pobór prądu w impulsie podczas nadawania może wynosić aż 1,8 A, o czym należy pamiętać projektując obwód zasilania. Średni pobór prądu podczas nadawania wynosi jednak tylko kilkaset mA, ponieważ ma on charakter impulsowy. Impuls w zależności od rodzaju transmisji trwa 577  $\mu\text{s}$  lub 1154  $\mu\text{s}$  na 4,615 ms przerwy (rys. 3). W związku z powyższym nie musimy stosować stabilizatora napięcia o dużej wydajności prądowej, wystarczy zespół kondensatorów o dużej pojemności znajdujący się jak najbliżej modułu. Świetnie do tego celu nadają się kondensatory CAP-X posiadające dużą pojemność oraz bardzo małą rezystancję szeregową. Należy również pamiętać o prawidłowym zaprojektowaniu płytki drukowanej PCB, ze szczególnym uwzględnieniem odpowiednich grubości ścieżek. W układzie będą bowiem płynąć prądy impulsowe o dużej wartości. Porty IO modułu Q2686 są współdzielone z układami peryferyjnymi na przykład UART, czy SPI, podob-

nie jak ma to miejsce w przypadku klasycznych mikrokontrolerów. Pewnym utrudnieniem w przypadku stosowaniu modułu może być fakt, że niektóre porty modułu pracują w standardzie 2,8 V, natomiast druga część posiada porty w standardzie 1,8 V. W niektórych przypadkach jest to dość kłopotliwe i wymaga stosowania układów pośredniczących – przesuwników poziomów (*level-shifters*). Przekroczenie napięcia ponad określony standard może bowiem spowodować uszkodzenie portów. Wszystkie porty IO są w pełni symetryczne i ich wydajność prądowa zarówno w stanie wysokim, jak i niskim wynosi 4 mA, tak więc w większości przypadków dołączenie np. diody LED będzie wymagało zastosowania dodatkowego układu, podobnie jak np. mikrokontrolerach 8051. W tab. 1 przedstawiono wszystkie sygnały dostępne na złączu modułu Q2686 wraz z ich krótką funkcją oraz opisem.

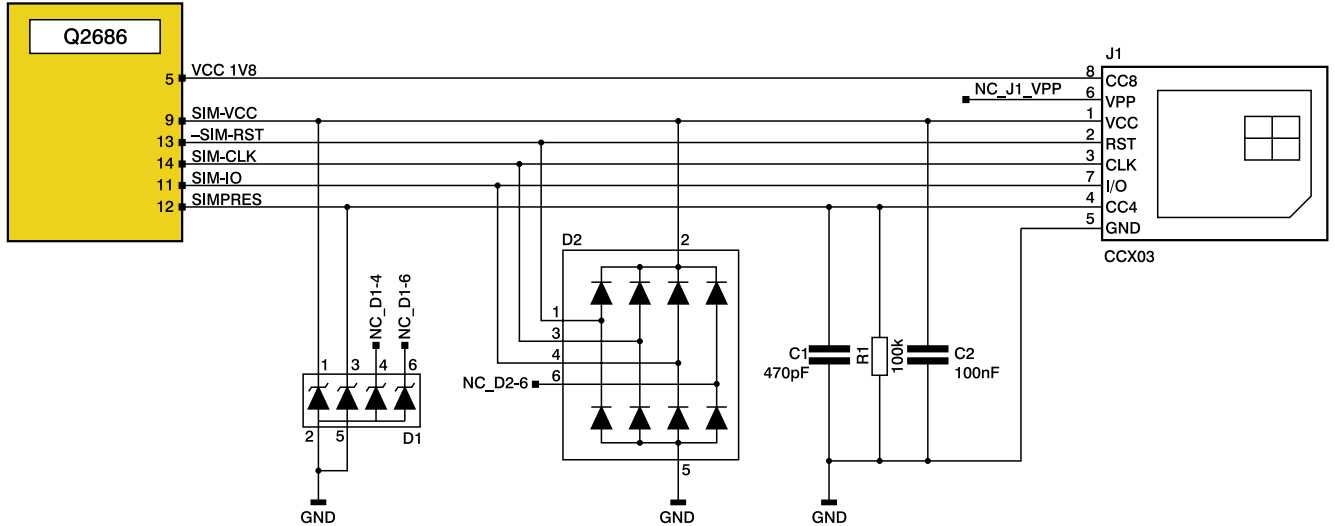
Poszczególne porty IO różnią się pełnionymi funkcjami. Niektóre z nich mogą być skonfigurowane tylko jako linie wejściowe, inne mogą pracować tylko jako wyjścia. Do prawidłowego działania, oprócz zasilania i anteny, moduł wymaga podłączenia karty SIM, która nie jest dostępna w module. Dlatego gniazdo takiej karty musi być umieszczone na płycie projektowanego urządzenia. Na rys. 4 przedstawiono sposób dołączenia gniazda karty SIM do modułu.

Do poszczególnych linii dołączono specjalne diody zabezpieczające moduł przed przepięciami elektrostatycznymi. Układ ten może współpracować zarówno z kartami zasilanymi napięciem 3 V, jak i kartami na napięcie 1,8 V. Możliwe jest zatem używanie dowolnych karty SIM, jakie są dostarczane obecnie przez operatorów. Aby móc programować i uruchamiać własne programy, musimy jeszcze obowiązkowo do pierwszego portu szeregowego dołączyć układ dopasowujący sygnały modułu do standardu RS232. Ponieważ do poprawnej transmisji potrzebny jest pełny interfejs RS232, musimy zastosować układ posiadający więcej linii niż klasyczny MAX2323. Przykładowy interfejs dopasowujący dla pierwszego portu szeregowego przedstawiono na rys. 5.

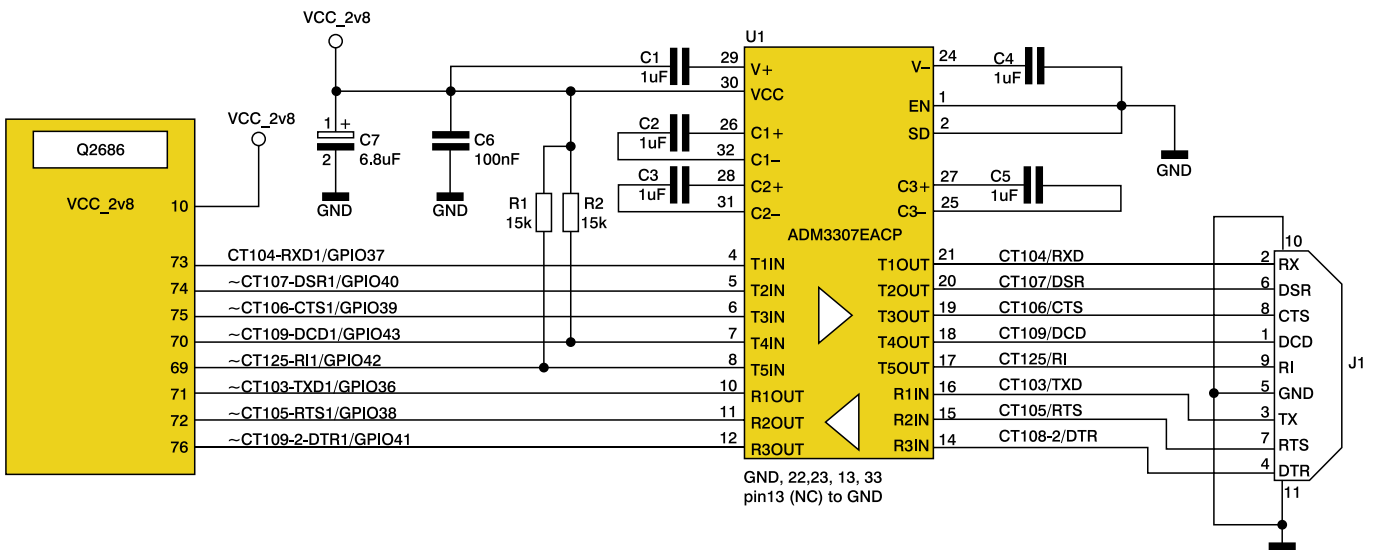
Tab. 1. Sygnały dostępne na złączu modułu Q2686

Opis	I/O	Napięcie	Nazwa sygnału		Numer pinu		Nazwa sygnału		Napięcie	IO	Opis
			Mux	Normal			Normal	Mux			
Zasilanie	I	VBATT		VBATT	1	2	VBATT		VBATT	I	Zasilanie
Zasilanie	I	VBAT		VBAT	3	4	VBAT		VBAT	I	Zasilanie
Zasilanie 1,8 V	0	VCC_1V8		VCC_1V8	5	6	CHG-IN		CHG-IN	I	Wejście ładowania
Bateria RTC	I/O	BAT_RTC		BAT_RTC	7	8	CHG-IN		CHG-IN	I	Wejście ładowania
Zasilanie karty SIM	0	1,8 lub 3 V		SIM-VCC	9	10	VCC_2V8		VCC_2V8	0	Zasilanie 2,8 V
Sygnał danych SIM	I/O	1,8 lub 3 V		SIO-IO	11	12	SIMPRES	GPIO18	VCC_1V8	I	Detekcja włożenia karty SIM
Wyjście SIM-RESET	0	1,8 lub 3 V		~SIM-RST	13	14	SIM-CLK		1,8 lub 3 V	0	Sygnał CLK SIM
Wyjście buzzera	0	Otwarty Dren		BUZ-OUT	15	16	BOOT		1,8	I	Nie używane
Wyjście diody FLASH LED	0	Otwarty Dren		FLASH-LED	17	18	~RESET		1,8	I	Wejście zerujące
Sygnał On/Off	I	Vbatt		ON/~OFF	19	20	BAT-TEMP		Analogowe	I	Wejście analogowe temperatury
Wejście A/C	I	Analogowe		AUX-ADC	21	22	~SPI-CS	GPIO31	2,8	0	SPI1 Chip Select
SPI1 Clock	0	2,8	GPIO32	SPI1-CLK	23	24	SPI1-I	GPIO30	2,8	0	SPI1 Data Input
SPI1 Data I/O	I/O	2,8	GPIO29	SPI1-IO	25	26	SPI2-CLK	GPIO32	2,8	0	SPI2 Clock
SPI2 Data I/O	I/O	2,8	GPIO33	SPI2-IO	27	28	~SPI2-CS	GPIO35	2,8	0	SPI2 Chip Select
SPI2 Input Data	I	2,8	GPIO34	SPI2-I	29	30	RXD2	GPIO15	1,8	0	RS232-2 RXD
RS232-2 TX	I	1,8	GPIO14	TXD2	31	32	CTS2	GPIO16	1,8	0	RS232 CTS
RS232-2 RTS	I	1,8	GPIO17	RTS2	33	34	MIC2N		Analogowe	I	Mikrofon zewnętrzny, wejście odwracające
Wyjście głośnika nieodwracające	0	Analogowe		SPK1P	35	36	MIC2P		Analogowe	I	Mikrofon zewnętrzny, wejście nieodwracające
Wyjście głośnika odwracające	0	Analogowe		SPK1N	37	38	MIC1N		Analogowe	I	Mikrofon 2 zewnętrzny, wejście odwracające
Wyjście głośnika 2 nieodwracające	0	Analogowe		SPK2P	39	40	MIC1P		Analogowe	I	Mikrofon 2 zewnętrzny, wejście nie odwracające
Wyjście głośnika 2 nieodwracające	0	Analogowe		SPK2N	41	42	-	-	-	-	-
	I/O	2,8		GPIO44	43	44	SCL	GPIO26	Otwarty dren	I/O	I <sup>2</sup> C Clock
	IO	2,8		GPIO19	45	46	SDA	GPIO27	Otwarty dren	I/O	I <sup>2</sup> C Data
	I/O	2,8		GPIO21	47	48	GPIO20		2,8	I/O	
Wejście INT1	I	2,8	GPIO25	INT1	49	50	INT0	GPIO3	2,8	I/O	Wejście INTO
	I/O	1,8	GPIO1		51	52	VPAD-USB		VPAD-USB	0	Wejście zasilania USB
	I/O	1,8	GPIO2		53	54	USB-DB		VPAD-USB	I/O	USB data
	I/O	2,8	GPIO23		55	56	USB-DM		VPAD-USB	I/O	USB data
	I/O	2,8	GPIO22		57	58	GPIO24		2,8	I/O	
Klawiatura kolumna 0	I/O	1,8	GPIO4	COL0	59	60	COL1	GPIO5	1,8	I/O	Klawiatura kolumna 1
Klawiatura kolumna 2	I/O	1,8	GPIO6	COL2	61	62	COL3	GPIO7	1,8	I/O	Klawiatura kolumna 3
Klawiatura kolumna 4	I/O	1,8	GPIO8	COL4	63	64	ROW4	GPIO13	1,8	I/O	Klawiatura wiersz 4
Klawiatura wiersz 3	I/O	1,8	GPIO12	ROW3	65	66	ROW2	GPIO11	1,8	I/O	Klawiatura wiersz 4
Klawiatura wiersz 1	I/O	1,8	GPIO10	ROW1	67	68	ROW0	GPIO9	1,8	I/O	Klawiatura wiersz 0
RS232 RI	0	2,8	GPIO42	~RI1	69	70	DCD	GPIO43	2,8	0	RS232 DCD
RS232 TX	I	2,8	GPIO36	TXD1	71	72	RTS1	GPIO38	2,8	I	RS232 RTS
RS232 RX	0	2,8	GPIO37	RXD1	73	74	DSR1	GPIO40	2,8	0	RS232 DSR
RS232 CTS	0	2,8	GPIO38	CTS1	75	76	DTR1	GPIO41	2,8	I	RS232 DTR
Ramka PCM	0	1,8		PCM-Sync	77	78	PCM-IN		1,8	I	Wejście PCM
Zegar PCM	0	1,8		PCM-CLK	79	80	PCM-OUT		1,8	0	Wyjście PCM
					81	82					
					83	84					
					85	86					
					87	88					

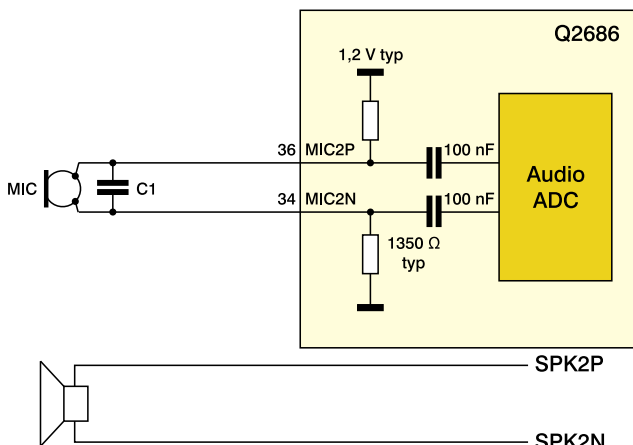




Rys. 4. Sposób dołączenia gniazda karty SIM do modułu



Rys. 5. Przykładowy interfejs RS232 dla pierwszego portu szeregowego modułu



Rys. 6. Sposób dołączenia do modułu mikrofonu elektretowego i słuchawek

nego układu, konieczna jest przetwornica indukcyjna, na przykład LTC2804IGN-1. W przypadku, gdy moduł będziemy wykorzystywać tylko do transmisji cyfrowej (GPRS, SMS), to mamy już w zasadzie wszystkie elementy potrzebne do pracy z modułem. Jeżeli istnieje potrzeba realizacji rozmów głosowych, wówczas konieczne jest dołączenie mikrofonu elektretowego i słuchawek odpowiednio do wejścia i wyjścia analogowego. Potrzebnych będzie do tego jeszcze kilkanaście ele-

mentów, tak jak to przedstawiono na rys. 6.

Wejście analogowe MIC2 posiada układ polaryzacji mikrofonu, tak więc umożliwia bezpośrednie dołączenie mikrofonu elektretowego. Podobnie jest i ze słuchawką lub głośnikiem, którą również możemy podłączyć bezpośrednio do złącza SPK2. Minimalna impedancja głośnika wynosi 8 Ω, a wzmacniacz wyjściowy dostarcza kilkudziesięciu miliwatów mocy wyjściowej, co umożliwia bezpośrednie wysterowanie małego głośniczka. Jak więc widzimy, moduł bardzo prosto można przystosować do realizacji klasycznych połączeń głosowych.

**Lucjan Bryndza SQ7FGB, EP**  
**lucjan.bryndza@ep.com.pl**