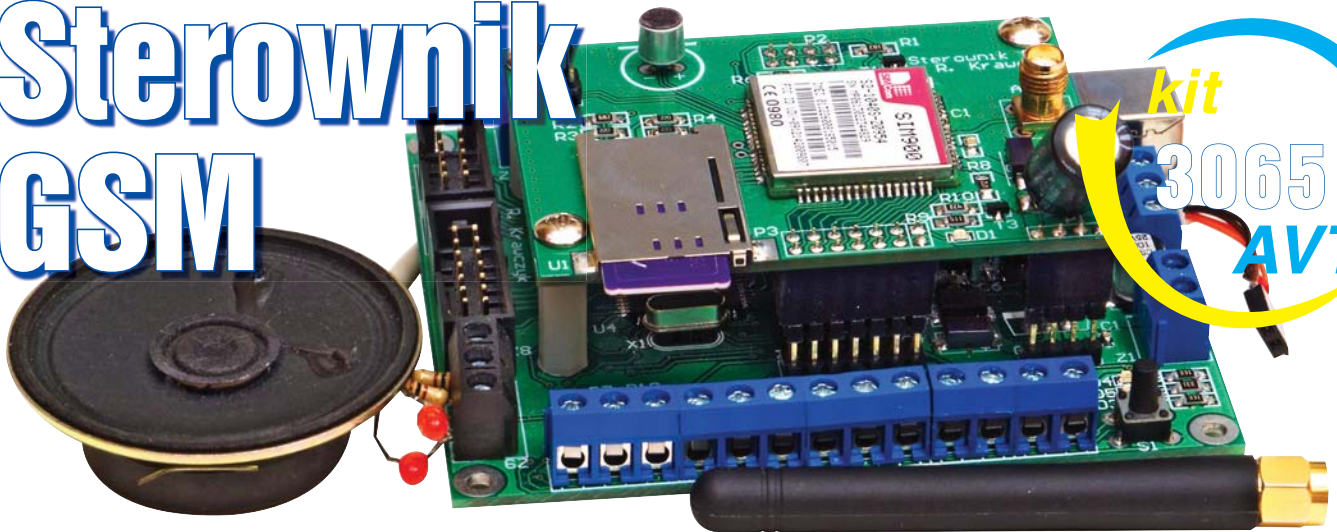




Sterownik GSM



Projekt nie jest wprawdzie przeznaczony dla zupełnie początkujących, jednak powinien zainteresować nawet średnio zaawansowanych z uwagi na ogromne możliwości oraz okazję do praktycznego wykorzystania modułu GSM i zapoznania się ze sterowaniem za pomocą popularnych komend AT. Natomiast do wykonania i wykorzystania urządzenia nie jest potrzebne zrozumienie wszystkich szczegółów.

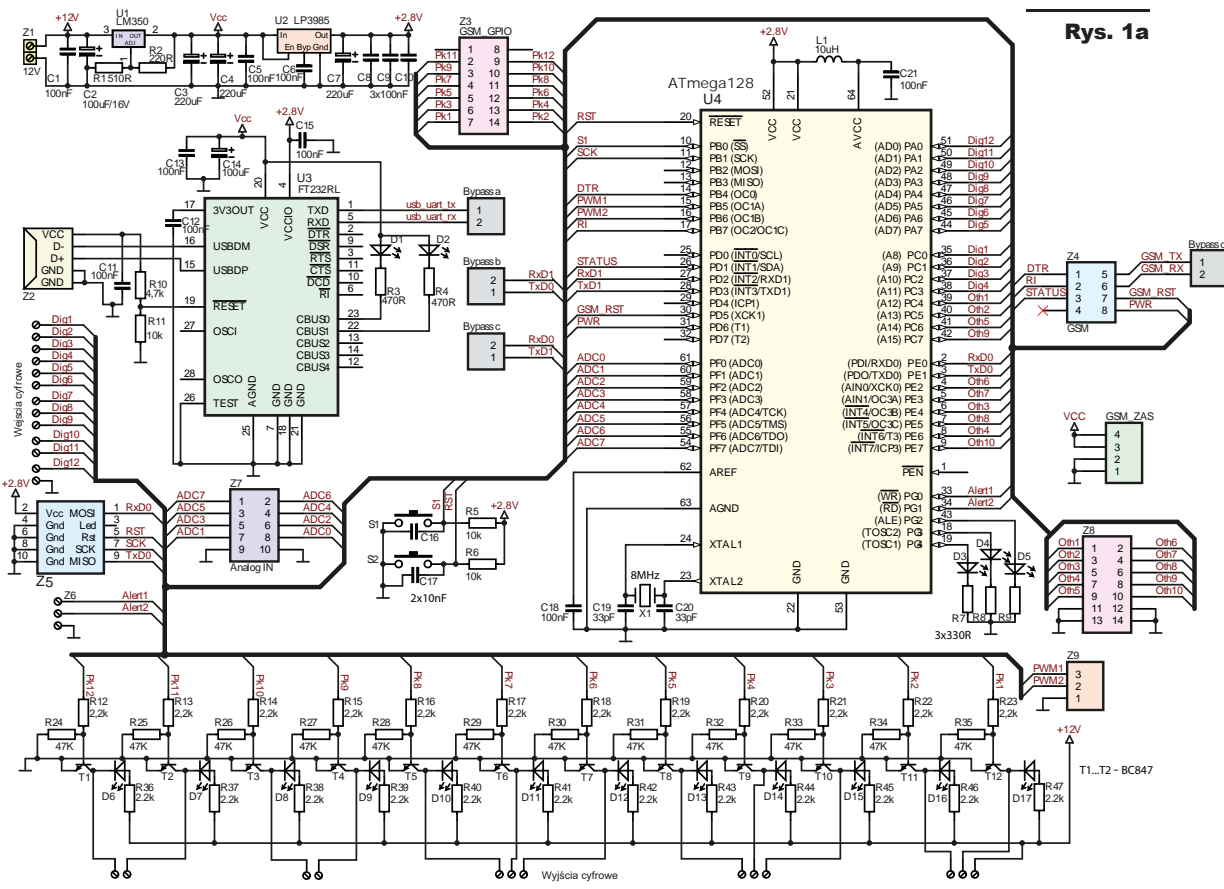
Przez łamy EdW już wiele razy przewijał się temat zdalnego sterowania. Pojawiały się tutaj układy nadające radiowo w paśmie 433MHz, 868MHz, Bluetooth, wykorzystujące podczerwień, światło lasera oraz GSM. Te ostatnie wykorzystywały fabrycznie telefon komórkowy. Najczęściej był to telefon SIEMENS C35, który komunikował się z resztą układu za pomocą popularnego interfejsu szeregowego. Rozwiązanie to sprawdzało się bardzo dobrze, lecz znaczącym mankamentem były duże fizyczne rozmiary urządzeń. Opisywany układ pozbawiony jest tej wady, zawiera bowiem w sobie scalony moduł GSM. Nie jest potrzebne

dotąd dołączenie do niego jakiegokolwiek komórki. Urządzenie pozwala na zdalne sterowanie 12 urządzeniami wyjściowymi (12 wyjść tranzystorowych), sprawdzanie stanu 12 wejść cyfrowych oraz 8 analogowych. Dodatkowo zostało ono wyposażone w dwa kanały PWM oraz dwa wejścia alarmowe.

Do działania wymaga jedynie zasilania oraz zasięgu jednej z sieci komórkowych. Sterowanie odbywa się za pomocą SMS-ów. Sterownik w momencie odebrania wiadomości przeprowadza odpowiednie operacje (wg wiadomości z SMS),

a następnie odsyła potwierdzenie dokonania zmian do nadawcy.

Przykładów zastosowania urządzenia może być wiele. Kanały wejściowe mogą pełnić funkcję monitora sprawdzającego stan oświetlenia, zamknięcia drzwi na klucz itp. Wyjścia z kolei mogą sterować żarówkami, ogrzewaniem, podlewaniem kwiatków itd. Wszystko to oczywiście po dołączeniu odpowiednich układów wykonawczych. Dodatkowo urządzenie ma opcję transparentności, tzn. że sterowanie modułem GSM odbywa się jedynie ze strony użytkownika przez terminal



Rys. 1a

komputerowy. Taka opcja pozwala na dogłębne poznanie możliwości modułu (która jest naprawdę duża), a także na zapoznanie się ze sterowaniem nim za pomocą komend AT.

Opis układu

Rysunek 1a przedstawia schemat ideowy sterownika GSM. Elementy C1–C5, U1, U2, R1, R2 pełnią funkcję stabilizowanego zasilacza o napięciu około 4,14V. Napięcie to służy do zasilenia modułu GSM oraz kolejnego stabilizatora obniżającego napięcie do wartości 2,8V. Wybór takiego napięcia został narzucony przez moduł GSM, który komunikuje się przez interfejs szeregowy UART o poziomach napięć równych 2,8V. W celu uproszczenia konstrukcji – pozbycia się konieczności stosowania konwerterów poziomów napięć – zdecydowano się na zasilenie mikrokontrolera napięciem o tej samej wartości. Elementem zarządzającym pracą urządzenia jest układ AVR ATMEGA 128. Został on wybrany z tego powodu, że ma dwa kanały UART oraz spory zapas pamięci FLASH i RAM.

Pierwszy z kanałów UART jest odpowiedzialny za komunikację z układem FT232RL (U3) zapewniającym wymianę danych z komputerem poprzez port USB. Diody LED D1, D2 sygnalizują nadawanie i odbiór danych przez U3. Drugi kanał UART zapewnia komunikację z modulem GSM. Wszystkie linie sygnałowe służące do komunikacji między mikrokontrolerem a modulem GSM zostały wyprowadzone na złącze Z4. Zasilanie modułu i jego periferii odbywa się poprzez złącze „GSM_ZAS”. Opisujemy moduł wraz z elementami niezbędnymi do jego prawidłowej pracy został umieszczony na drugiej płycie PCB. Jej schemat został przedstawiony na **rysunku 1b**. Dzięki budowie modułowej urządzenie zachowa kompaktowe kształty – płytki łączone są na tzw. kanapkę.

Fot. 1

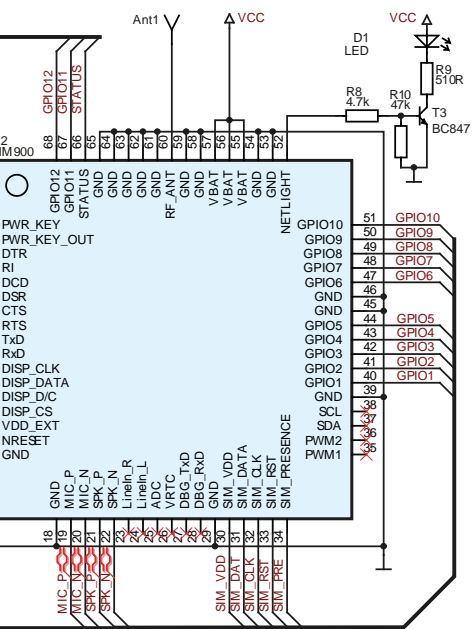


Dołączając do mikrokontrolera. Kanały analogowe pozwalają na pomiar napięć w zakresie 0–2,8V. W przypadku konieczności mierzenia napięć o szerszym zakresie, potrzebny będzie prosty dzielnik. Wejścia cyfrowe oraz ALARM są podciągnięte do plusa zasilania przez wewnętrzne rezystory kontrolera, przez co reagują na zwarcie do masy. Wyjścia cyfrowe sterowane są kluczami tranzystorowymi, zapewniającymi wydajność prądową 100mA. Tranzystory dołączone są do pinów IO modułu GSM. Bieżący stan każdego z wyjść jest sygnalizowany za pomocą diod LED D6–D17. Świecenie diody oznacza, że dany kanał jest aktywny. Ustawienie portu jako wyjście i sterowanie nim sprowadza się do wysyłania komendy:

AT+SGPIO=0,GPIO,1,poziom<CR>
gdzie: GPIO to numer portu (liczba 1–12), poziom: 1 – ustawienie stanu wysokiego na wyjściu, 0 – ustawienie stanu niskiego na wyjściu.

Diody D3–D5 informują o bieżącym stanie pracy urządzenia. Ich dokładne role zostaną opisane w dalszej części artykułu. Złącze Z5 służy do programowania AVR-a. Układ pinów jest typowy jak w złączu ISP-KANDA. Przejdźmy teraz do omówienia części software'owej sterownika.

Kod programu zajmuje około 10% przestrzeni FLASH i 30% pamięci RAM. Zapasu jest jeszcze bardzo dużo, zatem



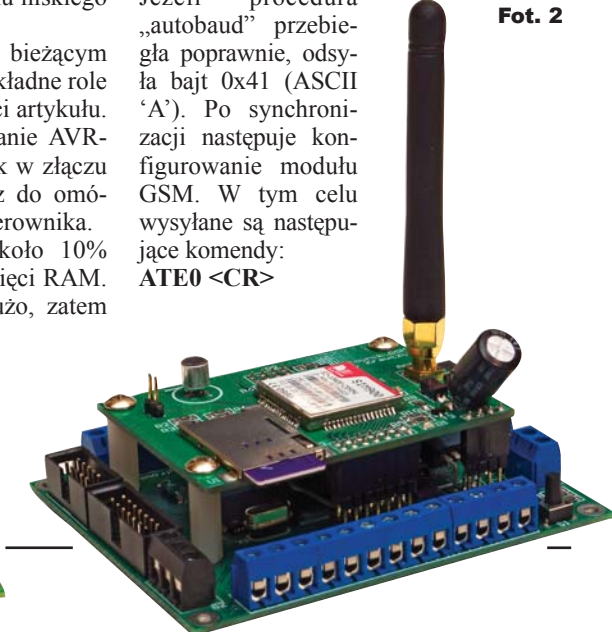
Rys. 1b

można z czasem dowolnie rozwijać funkcjonalność urządzenia.

Po dołączeniu zasilania następuje odczyt niezbędnych danych z pamięci EEPROM (np. kod PIN karty SIM), następnie wywołana jest procedura uruchamiania modułu GSM. Pierwszym wykonywanym krokiem jest zwarcie linii PWR_KEY do masy, co spowoduje, że moduł zostanie załączony. W tym momencie mikrokontroler zawieszona swoje działanie na około 2 sekundy. Następnie przechodzi do procedury synchronizującej wewnętrzny UART SIM900. Jest to bardzo interesująca funkcja modułu. Polega ona na tym, że ustawiamy prędkość UARTU w mikrokontrolerze na zawierającą się w zbiorze (1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200bps), a następnie wysyłamy do modułu bajt 0x41. Moduł, odbierając znak, automatycznie dostraja się do danej prędkości. Jeżeli procedura „autobaud” przebiega poprawnie, odsyła bajt 0x41 (ASCII 'A'). Po synchronizacji następuje konfigurowanie modułu GSM. W tym celu wysyłane są następujące komendy:

ATE0 <CR>

Fot. 2



Powoduje wyłączenie echa tzn. moduł GSM nie będzie odsyłał danych, które odbierze na linii Rx/D.

AT+CLIP=1<CR>

Wyświetlanie informacji na temat połączenia przychodzącego.

AT+CRC=1<CR>

Wyświetlanie szczegółowych informacji na temat rodzaju połączenia przychodzącego.

AT+CNMI=2,2<CR>

Natychmiastowe wyświetlanie przychodzących wiadomości SMS, bez zapisywania ich w pamięci.

AT+CMGF=1<CR>

Uruchomienie trybu SMS Text mode.

AT+CPIN?<CR>

Zapytanie modułu o gotowość SIM. W odpowiedzi oczekujemy jednego z trzech ciągów:

+CPIN: READY

Karta SIM gotowa, zarejestrowana w sieci (PIN został wpisany, lub brak blokady kodem PIN).

SIM not inserted

Brak karty SIM. Wstrzymanie wykonywania programu.

+CPIN: SIM PIN

Karta SIM wykryta, konieczne podanie kodu PIN.

W przypadku wystąpienia ostatniej z opcji następuje wysłanie komendy:

AT+CPIN=xxxx <CR>

(xxxx – oznaczają kod pin)

Po wykryciu komunikatu **+CPIN: READY** program przechodzi do wykonywania pętli głównej.

Wywoływane są w niej cyklicznie cztery funkcje. Dwie mają na celu sprawdzanie buforów. Pierwszy z nich jest buforem GSM. Jest on parsowany pod kątem obecności komend sterujących oraz hasła dostępu do sterownika (bez jego podania żadna z komend nie zostanie wykonana).

W przypadku, gdy hasło zostanie poprawnie podane oraz dostępne są komendy sterujące, nastąpi ich wykonanie, a następnie moduł GSM odeśle do nadawcy informacje o wprowadzonych zmianach. Drugi z buforów przechowuje dane odebrane

przez układ FT232. Podobnie jak poprzedni, jest przeszukiwany w celu znalezienia poprawnie odebranej ramki danych. Jeżeli zostanie ona znaleziona, następuje jej przetworzenie i wykonanie określonych funkcji. Odbierane dane ładowane są do buforów podczas wywoływania funkcji obsługi przerwań. Dzięki temu nie zdarzy się sytuacja, że jakiś bajt/bajty nie zostaną odczytane z powodu wykonywania funkcji delay itp. Dwie pozostałe funkcje sprawdzają, czy nie zostało naruszone któreś wejść typu alarm oraz monitorują układ SIM900 (w przypadku wykrycia, że moduł nie pracuje, następuje ponowna inicjalizacja systemu).

Kilka słów opisu należy się sposobowi komunikacji między komputerem a sterownikiem. Informacja ta będzie niezbędna osobom, które będą próbowały rozbudować urządzenie.

Dane przesyłane są protokołem, którego pola przedstawione są na **rysunku 2**. Każda ramka zaczyna się od bajtów 0x3C3C.

Kolejne pola to adresy nadawcy i odbiorcy. W tym momencie zarówno program jak i sterownik nie weryfikują wartości tych pól – mogą one być dowolne.

Następnie występują dwa pola żądania, pole długości danych oraz dodatkowe dane.

Tabela 1 przedstawia wszystkie

dotychczasowo obsługiwane komendy z dokładnym znaczeniem każdego z pól. XX oznacza, że pole w danym przypadku, nie jest analizowane pod kątem zawartości.

Ostatnie z pól – CRC pełni funkcję 16-bitowej sumy kontrolnej. Do jej wyliczenia służy funkcja **UpdateCRC**, znajdująca się w pliku `uart_protocol.c`

Suma kontrolna liczona jest na podstawie pól, począwszy od Adresu nadawcy do ostatniego bajtu danych włącznie.

Montaż i uruchomienie

Urządzenie składa się z dwóch płytek PCB, pokazanych na **rysunku 3**. Szczegóły można też zobaczyć na **figurach 1...5** oraz tytułowej. Obie płytki są dwuwarstwowe z metalizacją otworów. Kilka elementów to elementy do montażu przewlekane; większa część jest montowana powierzchniowo. Montaż rozpoczynamy od elementów o najmniejszych gabarytach. Jest to bardzo ważne,

Funkcja	Request 1	Request 2	Dane
Wysłanie tekstu do debuggera	0x01	XX	łańcuch znaków
Zażądanie nazwy wejścia cyfrowego	0x02		XX
Zapisanie do EEPROM nazwy wej.c	0x03	Nr wejścia 0x01-0x0C	łańcuch znaków z nazwą wejścia
Wysłanie nazwy wej.c (do PC)	0x04		
Zażądanie nazwy wejścia analogowego	0x05		XX
Zapisanie do EEPROM nazwy wej.a	0x06	Nr wejścia 0x01-0x08	łańcuch znaków z nazwą wejścia
Wysłanie nazwy wej.a	0x07		
Zażądanie nazwy wyjścia cyfrowego	0x08		XX
Zapisanie do EEPROM nazwy wyj.c	0x09	Nr wyjścia 0x01-0x0C	łańcuch znaków z nazwą wyjścia
Wysłanie nazwy wyj.c	0x0A		
Zażądanie nazwy wejścia ALARM	0x0B		XX
Zapisanie do EEPROM nazwy wej. AL	0x0C	Nr wejścia 0x01-0x02	łańcuch znaków z nazwą wejścia
Wysłanie nazwy wej. AL	0x0D		
Zażądanie nazwy wyjścia PWM	0x0E		XX
Zapisanie do EEPROM nazwy wyj. PWM	0x0F	Nr wyjścia 0x01-0x02	łańcuch znaków z nazwą wyjścia
Wysłanie nazwy wyj. PWM	0x10		
Zażądanie numeru głównego	0x11	XX	XX
Zapisanie do EEPROM numeru głów.	0x12	0x01 – informuj o starcie systemu, 0x00 nie informuj	łańcuch znaków z numerem
Wysłanie numeru głów.	0x13		
Ustawienie PINu	0x14	XX	Łańcuch z PINEM
Ustawienie hasła SMS	0x15	XX	Łańcuch z hasłem
Ustawienie stanu wyjść cyfrowych	0xA5	Numer wyjścia	0x01 – zał 0x00-wył
Ustawienie stanu wyjść PWM	0xA6	Numer wyjścia	Wypełnienie 0x00-0xFF

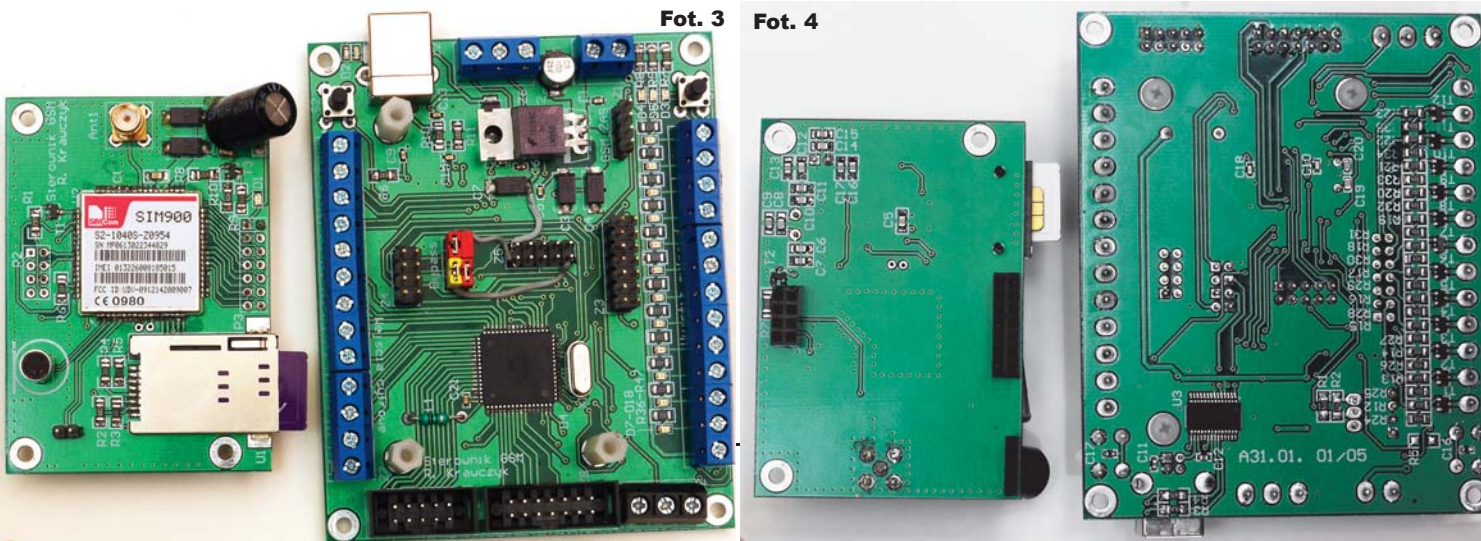
XX – oznacza dowolność

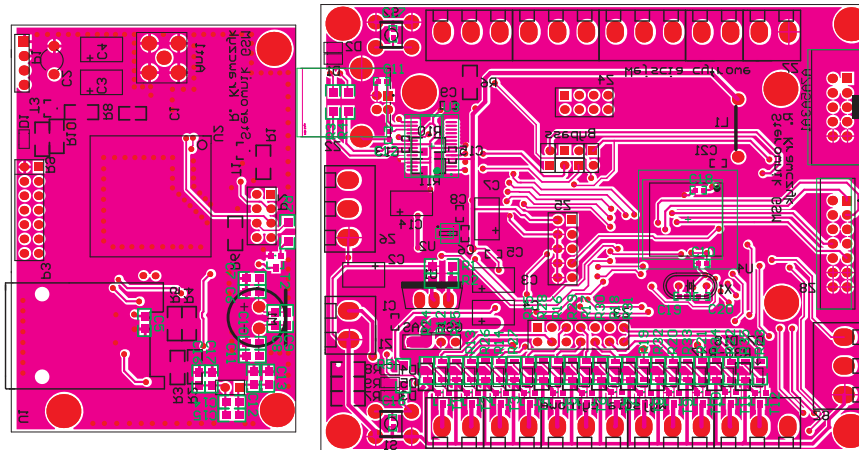
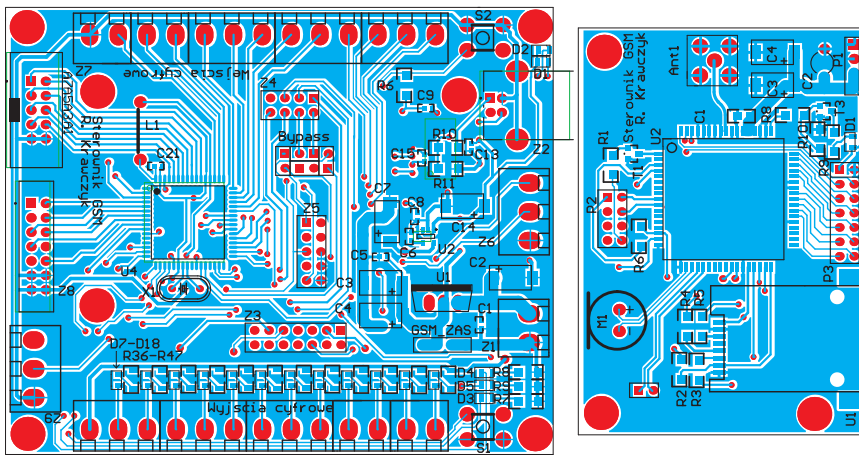
Tab. 1

Rys. 2



Fot. 3 Fot. 4





Rys. 3. Płytki drukowane - skala 75%

ponieważ w przeciwnym wypadku dostęp grota lutownicy do danego obszaru płytki może być bardzo trudny. Każda z płytek ma 3 otwory na śruby M3, służące do ich wzajemnego skręcenia. Dla zachowania sztywności konstrukcji używamy tulejek dystansowych o wysokości 15mm. Zmontowane urządzenie zasilamy napięciem z zakresu 9V–12V przez złącze Z1. Użyty stabilizator (LM350) pozwala na znacznie wyższe napięcie wejściowe. Jednak ze względu na fakt, iż nie ma on radiatora, nie zalecam wartości powyżej 12V. Kolejnym krokiem jest zaprogramowanie AVR-a. Wsad do procesora znajduje się w Elportalu wśród materiałów dodatkowych do tego numeru, w pliku **proc.hex**. Konieczne jest ustawienie fusebitów w następujący sposób:

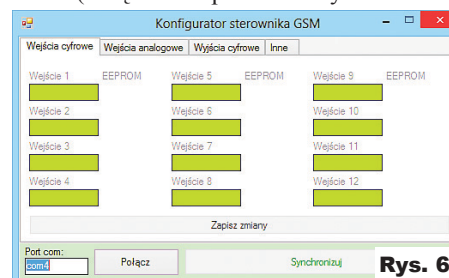
Extended 0xFF,
High Fuse 0xD9
Low Fuse 0xFF

Na czas programowania obowiązkowo należy odłączyć płytkę modułu GSM. W przeciwnym razie programator nie wykryje mikrokontrolera.

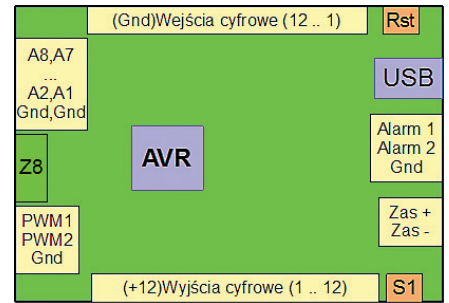
Rysunek 4 przedstawia rozkład wejść-wyjść urządzenia z uwzględnieniem kolejności numeracji kanałów. Dołączamy do nich różnego typu urządzenia wykonawcze wg własnych potrzeb.

Na wejścia cyfrowe (w tym ALARM) i analogowe nie należy podawać napięć wyższych niż 2,8V.

Po podłączeniu dodatkowych urządzeń należy dokonać konfiguracji programowej sterownika. W tym celu zakładamy 4 zworki na złącze bypass w sposób przedstawiony na **rysunku 5**. Jest to układ, kiedy tzw. bypass jest wyłączony – moduł GSM jest sterowany poprzez mikrokontroler. Po umieszczeniu jumperów, łączymy ze sobą obie płytki i podłączamy urządzenie do komputera przewodem USB (urządzenie powinno być zasilane).

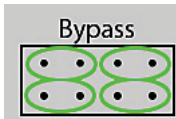


Fot. 5



Rys. 4

Rys. 5



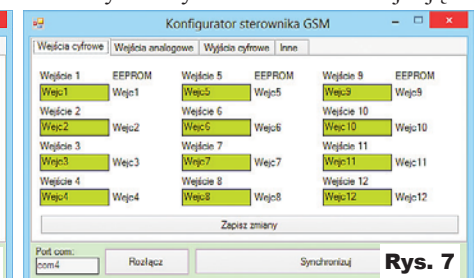
ne). Komputer wykryje nowe urządzenie typu wirtualny port szeregowy i nada mu nazwę COMx (gdzie x należy

do przedziału 1–99).

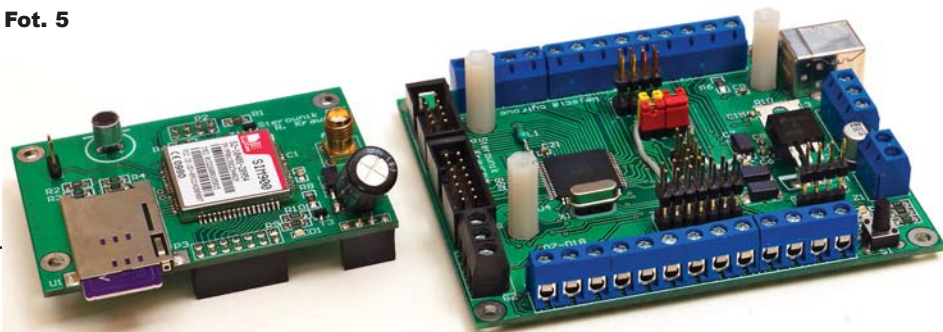
Do obsługi sterownika przygotowano dedykowany program komputerowy (dostępny w Elportalu). Jego głównym zadaniem jest odczyt i zapis przyjaznych nazw wejść i wyjść sterownika. Dodatkowo pozwala on na sterowanie wyjściami cyfrowymi, kanałami PWM oraz proste debugowanie urządzenia.

Rysunek 6 przedstawia okno programu. Po jego uruchomieniu, w pole port COM wpisujemy nazwę wirtualnego portu szeregowego, pod którym zainstalował się sterownik. Po wpisaniu nazwy klikamy przycisk **Połącz**. Jeżeli operacja przebiega poprawnie, zostaną odobkowane główne zakładki oraz przycisk **Synchronizuj**. Przycisk ten rozpoczyna odczyt bieżącej konfiguracji urządzenia. Synchronizacja trwa kilka sekund. W przypadku, gdy przebiegnie ona niepoprawnie (np. któraś z nazw nie zostanie odczytana), należy ponownie wywołać funkcję.

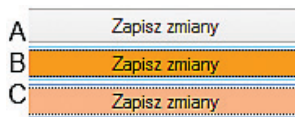
Rysunek 7 przedstawia formę okna po odczytaniu nazw (wejścia cyfrowe). Napisy pod etykietami EEPROM prezentują nazwy zapisane w pamięci urządzenia. Przy każdym z kanałów znajdują



Rys. 7



się pola tekstowe służące do wprowadzania nowych nazw. Kiedy nazwa wpisana w pole jest identyczna jak zapisana w EEPROM, pole to jest podświetlone na



Rys. 8

Wprowadzony numer główny i opcję informacji o ponownym uruchamianiu należy zatwierdzić przyciskiem **Zapisz**.

Sterowanie kanałami PWM odbywa się poprzez wpisanie wartości z zakresu 0–255 w pole PWM1 lub PWM2 i naciśnięcie przycisku **OK** przy używanym polu tekstowym.

Naciśnięcie przycisku **Debugger** pokazuje bądź ukrywa pole debugowania pracy sterownika (rysunek 11). Pole jest bardzo przydatne w przypadku problemów z pracą urządzenia – na podstawie wyświetlanych komunikatów dość szybko można dojść do przyczyny problemów. Podczas pierwszej konfiguracji urządzenia zalecam następującą procedurę obsługi programu. Na początku upewniamy się, czy karta SIM jest **WYJĘTA** z urządzenia. Jeśli nie, wyjmujemy ją przy odłączonym zasilaniu. Jeżeli została wyjęta, łączymy się ze sterownikiem, klikając **Połącz** w programie. Następnie przechodzimy do zakładki **Inne** i otwieramy okno debugowania, po czym naciskamy przycisk **Reset** w urządzeniu. W debugerze powinny pojawiać się kolejno różne komunikaty. W momencie gdy pojawi się tekst „*Brak SIM*”, przystępujemy do wypełnienia przyjaznych nazw, kodu PIN, hasła SMS oraz numeru głównego. Jeżeli jakiś z kanałów nie będzie wykorzystywany – mimo to musimy mu nadać nazwę (może to być np. spacja. Kanały nie muszą mieć unikalnych nazw).

Z racji tego, że fabrycznie EEPROM wypełniony jest nieznanymi danymi, bardzo proszę nie klikać przycisku *Synchronizuj* przed dokonaniem pierwszego zapisu, ponieważ może to spowodować, że program się zawiesi.

Po skonfigurowaniu sterownika zamykamy program, odłączamy zasilanie i umieszczamy w gnieździe kartę SIM. Po tym ponownie dołączamy zasilanie. Diody D3, D4, D5 początkowo powinny świecić. W miarę kolejnych etapów inicjalizacji będą gaśły. W momencie kiedy zgasną wszystkie, urządzenie jest gotowe do odbierania komend SMS. Dioda D3 sygnalizuje odbiór pakietu zgodnego z protokołem z komputera. Błyski D4 informują o wymianie danych między modułem GSM a mikrokontrolerem. Ciągłe świecenie D3–D5 świadczy o wystąpieniu błędu.

Numer główny jest numerem, na który wysyłane są SMS-y informujące o restarcie systemu (o ile opcja jest aktywna) oraz o aktywowaniu któregoś z wyjść ALARM.

Numer główny jest numerem, na który wysyłane są SMS-y informujące o restarcie systemu (o ile opcja jest aktywna) oraz o aktywowaniu któregoś z wyjść ALARM.

Wprowadzony numer główny i opcję informacji o ponownym uruchamianiu należy zatwierdzić przyciskiem **Zapisz**.

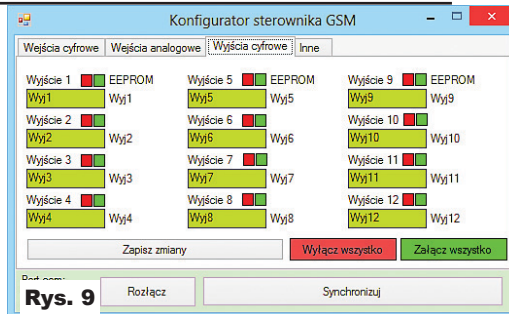
Podczas pierwszej konfiguracji urządzenia zalecam następującą procedurę obsługi programu. Na początku upewniamy się, czy karta SIM jest **WYJĘTA** z urządzenia. Jeśli nie, wyjmujemy ją przy odłączonym zasilaniu. Jeżeli została wyjęta, łączymy się ze sterownikiem, klikając **Połącz** w programie. Następnie przechodzimy do zakładki **Inne** i otwieramy okno debugowania, po czym naciskamy przycisk **Reset** w urządzeniu. W debugerze powinny pojawiać się kolejno różne komunikaty. W momencie gdy pojawi się tekst „*Brak SIM*”, przystępujemy do wypełnienia przyjaznych nazw, kodu PIN, hasła SMS oraz numeru głównego. Jeżeli jakiś z kanałów nie będzie wykorzystywany – mimo to musimy mu nadać nazwę (może to być np. spacja. Kanały nie muszą mieć unikalnych nazw).

Podczas pierwszej konfiguracji urządzenia zalecam następującą procedurę obsługi programu. Na początku upewniamy się, czy karta SIM jest **WYJĘTA** z urządzenia. Jeśli nie, wyjmujemy ją przy odłączonym zasilaniu. Jeżeli została wyjęta, łączymy się ze sterownikiem, klikając **Połącz** w programie. Następnie przechodzimy do zakładki **Inne** i otwieramy okno debugowania, po czym naciskamy przycisk **Reset** w urządzeniu. W debugerze powinny pojawiać się kolejno różne komunikaty. W momencie gdy pojawi się tekst „*Brak SIM*”, przystępujemy do wypełnienia przyjaznych nazw, kodu PIN, hasła SMS oraz numeru głównego. Jeżeli jakiś z kanałów nie będzie wykorzystywany – mimo to musimy mu nadać nazwę (może to być np. spacja. Kanały nie muszą mieć unikalnych nazw).

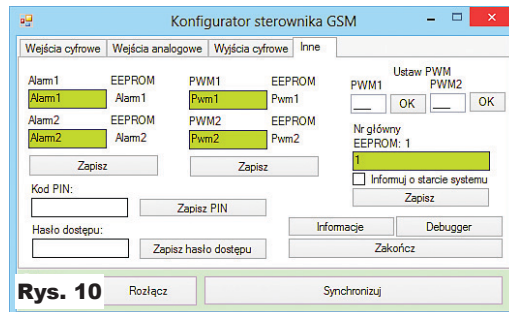
Z racji tego, że fabrycznie EEPROM wypełniony jest nieznanymi danymi, bardzo proszę nie klikać przycisku *Synchronizuj* przed dokonaniem pierwszego zapisu, ponieważ może to spowodować, że program się zawiesi.

Po skonfigurowaniu sterownika zamykamy program, odłączamy zasilanie i umieszczamy w gnieździe kartę SIM. Po tym ponownie dołączamy zasilanie. Diody D3, D4, D5 początkowo powinny świecić. W miarę kolejnych etapów inicjalizacji będą gaśły. W momencie kiedy zgasną wszystkie, urządzenie jest gotowe do odbierania komend SMS. Dioda D3 sygnalizuje odbiór pakietu zgodnego z protokołem z komputera. Błyski D4 informują o wymianie danych między modułem GSM a mikrokontrolerem. Ciągłe świecenie D3–D5 świadczy o wystąpieniu błędu.

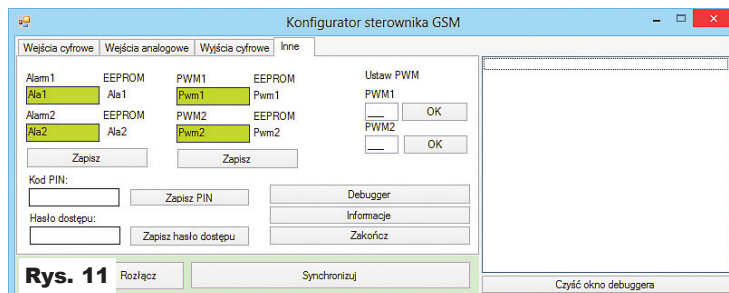
Numer główny jest numerem, na który wysyłane są SMS-y informujące o restarcie systemu (o ile opcja jest aktywna) oraz o aktywowaniu któregoś z wyjść ALARM.



Rys. 9



Rys. 10



Rys. 11

Sterowanie urządzeniem odbywa się za pomocą wiadomości SMS wysyłanych na numer karty SIM umieszczonej w sterowniku. Format wiadomości jest następujący:

Hasło dostępu<spacja>**komenda 1**<spacja>**komenda 2**<spacja>...<spacja>**komenda N**

Tabela 2 przedstawia wszystkie komendy, jakie obsługuje sterownik GSM.

Załóżmy, że hasło dostępu SMS brzmi **elektroNika**. Chcąc załączyć wyjścia 1 i 12 (poprzednio wszystkie były wyłączone), wysyłamy wiadomość o treści:

elektroNika W01:1 W12:1

po odpowiednimysterowaniu kanałów, otrzymamy wiadomość informującą o stanie wyjść.

Aby sprawdzić stan (bez jego zmiany) wyjść i wejść, wysyłamy:

elektroNika d_out d_in a in

Pragnę zauważyć, że komendy **allon** i **alloff** są wykonywane zawsze jako pierwsze. Pozwala to na możliwość zastosowania uproszczonej wiadomości SMS. Przykładowo gdybyśmy chcieli załączyć wszystkie wyjścia prócz pierwszego i dwunastego, nie musimy po kolei pisać: **elektroNika W01:0 W02:1 W03:1 ... W12:0**

tylko prościej poprzez wpisanie: **elektroNika allon W1:0 W12:0**

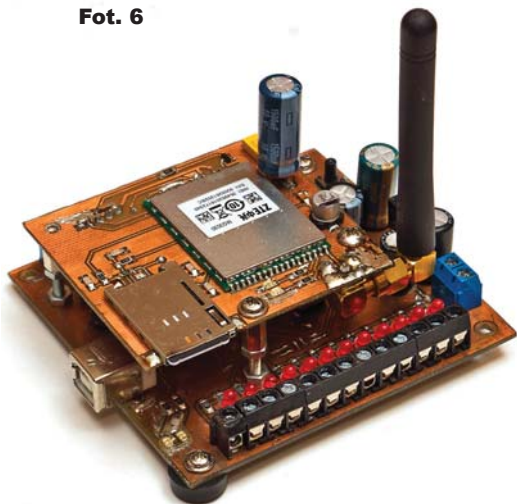
Tabela komend	
Komenda	Działanie
W01:1	załącza wyjście pierwsze
W01:0	wyłącza wyjście pierwsze
W02:1	załącza wyjście drugie
W02:0	wyłącza wyjście drugie
...	...
W12:1	załącza wyjście dwunaste
W12:0	wyłącza wyjście dwunaste
allon	załącza wszystkie wyjścia cyfrowe
alloff	wyłącza wszystkie wyjścia cyfrowe
pwm1:X	steruje wyjściem PWM1
pwm2:X	steruje wyjściem PWM2
	X-jest liczbą z zakresu 0-255
d_out	informacje o stanie wyjść cyfrowych
d_in	informacja o stanie wejść cyfrowych
a_in	informacja o stanie wejść analogowych
oth	Informacja o wejściach alarm i wyjściach PWM
A1_reset	Odblokowuje kanał 1 wejść ALARM
A2_reset	Odblokowuje kanał 2 wejść ALARM

Tabela 2

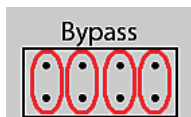
Krótkiego komentarza wymagają komendy odblokowywania wejść ALARM. Domyślnie wejścia te są odblokowane. W momencie zwarcia któregoś z nich do masy następuje automatyczne wysłanie SMS-a pod główny numer z informacją oth. Dodatkowo wejście, które zostało wyzwolone, zostaje zablokowane – kolejne wystąpienie zdarzenia zwarcia do masy nie będzie powodowało wysłania informacji o alarmie do czasu odblokowania kanału komendą A(1,2)_reset.

Tryb bypass. Jest to tryb pozwalający na ręczne sterowanie modułem GSM poprzez terminal komputerowy. Aby rozpocząć pracę w takiej konfiguracji, należy ustawić jumpery złącza bypass wg rysunku 12. Następnie naciskamy przyciski S1 i Reset (S2). Po czym zwalnimy jedynie

Fot. 6



Rys. 12



Reset. Powinna zaświecić się dioda D3. Od tego momentu przycisk S1 służy do sterowania wejściem PWR_KEY modułu GSM.

Na koniec nie pozostaje mi nic innego, jak życzyć wielu pomysłów na wykorzystanie opisywanego urządzenia. Dociekliwych zachęcam do dokładnego zapoznania się z możliwościami modułu GSM. Namawiam również do samodzielnej rozbudowy sterownika o dodatkowe funkcje. Możliwości jest bardzo dużo – może to być np. obsługa bloków AUDIO i GPRS modułu.

Na płycie PCB zostało dodane złącze (Z8) wyprowadzające 10 niewykorzystanych portów mikrokontrolera.

W Elportalu, wśród materiałów dodatkowych można znaleźć schemat, dokumentację płytek, oprogramowanie, rzuty ekranowe, a także schematy wcześniejszej wersji tego urządzenia z modułem MGE3030. Model tej wcześniejszej wersji pokazany jest na fotografiach 6 i 7 (w razie pytań, można pisać do Auto-



Fot. 7



ra, na podany adres). Ponieważ moduł MGE3030 nie jest zalecany do nowych opracowań i stopniowo znika z ofert handlowych, specjalnie została opracowana opisana w artykule nowa wersja z modułem SIM900.

Radosław Krawczyk
radek.radiator@gmail.com

Wykaz elementów

Płytką główna:

R1	510Ω	[1206]
R2	220Ω	[1206]
R3,R4	470Ω	[1206]
R5,R6, R11	10kΩ	[1206]
R7-R9	330R	[1206]
R10	4,7kΩ	[1206]
R12-R23,R36-R47	2,2kΩ	[1206]
R24-R35	47kΩ	[1206]
C1,C5,C6,C8-C13, C15,C18,C21	100nF	[0603]
C2	100uF/16V	[tantal obudowa SMD-D]
C3,C4,C7	220uF/tantalowy	[SMD-D]
C14	100uF/elektrolit	[SMD-D] min 6,3V
C16,C17	10nF	[0603]
T1-T12		BC847
D1,D2, D5		LED żółta [0805]
D3, D6-D17		LED czerwona [0805]
D4		LED zielona [0805]
U1		LM350
U2		LP3985, 2.8V [sot23-5]
U3		FT232RL
U4		ATMEGA128L
Bypass		Listwa goldpin 2x4 + 4 jumpery
GSM_ZAS		Złącze na listwę goldpin żeńskie 1x4
L1		Dławik 10uH
S1, S2		Microswitch
X1		Rezonator kwarcowy 8MHz
Z2		Gniazdo USB-B do montażu przewlekane
Z3		Złącze na listwę goldpin (żeńskie) 2x7
Z4		Złącze na listwę goldpin (żeńskie) 2x4
Z5		Listwa goldpin 2x5
Z7		Złącze IDC do druku 2x5 raster 2,54

Z8 Złącze IDC do druku 2x7 raster 2,54

8 złączy ARK3

5 złączy ARK2

Płytką GSM:

R1,R7	1kΩ	[1206]
R2	6,8kΩ	[1206]
R3-R5	22Ω	[1206]
R6	10kΩ	[1206]
R8	4,7kΩ	[1206]
R9	510Ω	[1206]
R10	47kΩ	[1206]
C1	100nF	[0805]
C2	220uF/min	[6,3V]
C3,C4	220uF/min	6,3V [SMD-D]
C5	220nF	[0805]
C6,C8,C10,C12,C14,C16	10pF	[0805] (*)
C7,C9,C11,C13,C15,C17	33pF	[0805] (*)
D1		LED zielona smd 0805
T1-T3		BC847
U1		Gniazdo na kartę SIM z wyrzutnikiem
U2		SIM900
Ant1		Złącze SMA żeńskie do druku (kątowe lub proste)
M1		mikrofon elektretowy (*)
P1		Listwa goldpin 1x4, raster 2,54, wysokość od 15mm
P2		Listwa goldpin 2x4, raster 2,54, wysokość od 15mm
P3		Listwa goldpin 2x7, raster 2,54, wysokość od 15mm
Antena GSM np. z oferty www.maritex.com.pl		
3 tuleje dystansowe 15mm / średnica 3mm		
Śrubki, podkładki, nakrętki M3		
Opcjonalnie: Głośnik impedancja 32Ω (*)		

Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny AVT-3065.