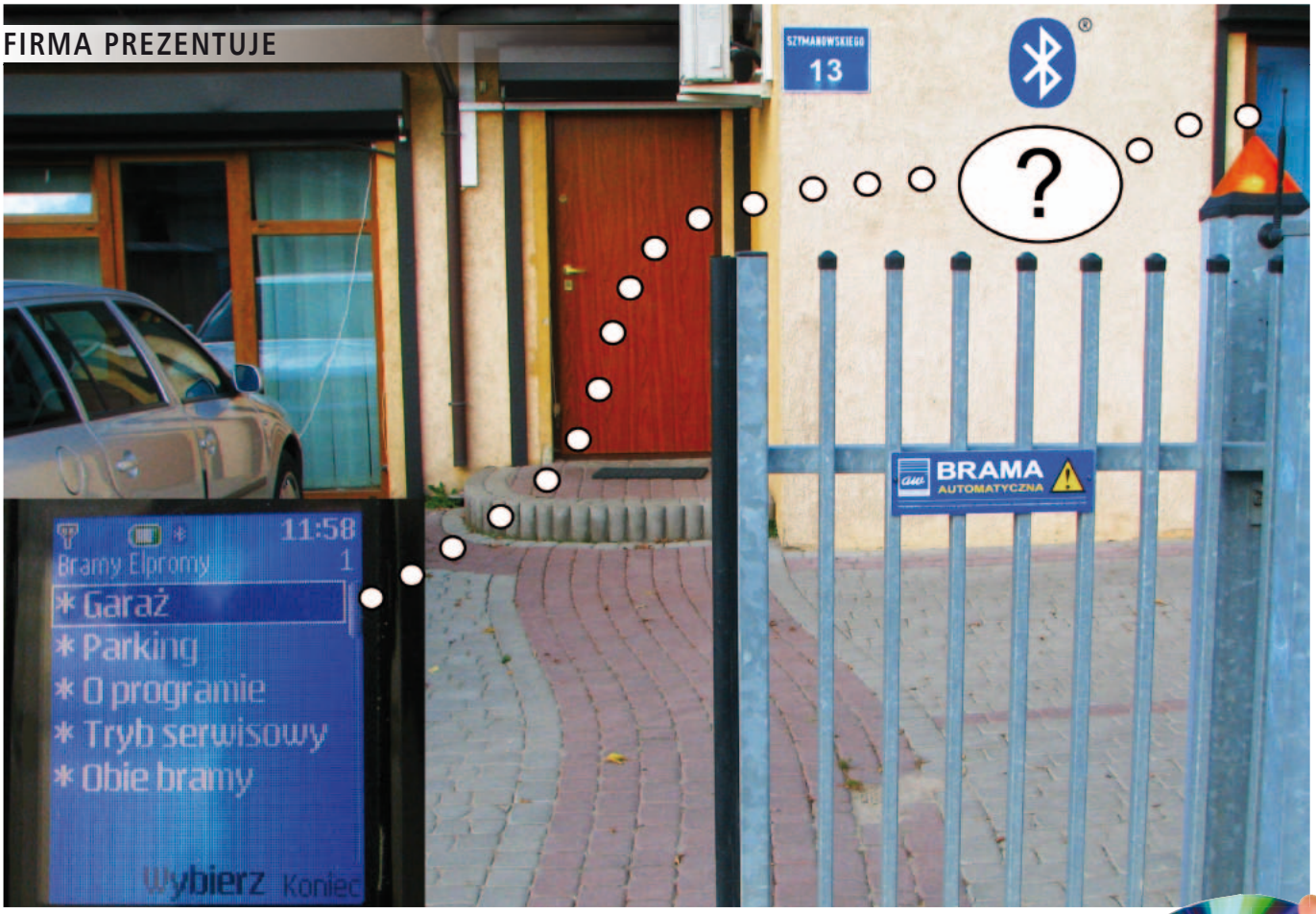


Bramo otwórz się

Zalety standardu Bluetooth

Jedną z zalet Bluetooth jest jego rzeczywista standaryzacja, zgodność ze starszymi wersjami oraz obecność w niemal każdym współczesnym telefonie komórkowym. Naturalne jest więc pytanie o to, czy można jakoś zmusić np. telefon komórkowy do bezprzewodowego sterowania urządzeniami, które nas otaczają? Prezentujemy jak można to wykonać używając jedynie modułu Bluetooth, zasilacza oraz kilku elementów dyskretnych.

W pierwszym kroku załóżmy, że chcemy sterować cyfrowymi liniami wyjściowymi albo ustawiając jeden z dwóch stanów, albo generując na nich impuls dla układu wykonawczego. W ten sposób będziemy mogli kontrolować przekaźniki włączające lampkę w pokoju, uzbrajać alarm, opuszczać rolety okienne czy sterować bramą ogrodzenia. Obecnie w tzw. inteligentnych domach króluje standard ZigBee. W naszym konkretnym przypadku przegrywa on jednak swoją niepopularnością wśród „mobilnych smyczy".

Zastępowany pilot jest dość unikatowy, bo otwiera nasze drzwi, a nie sąsiada, więc kolejny wymóg to bezpieczeństwo oraz unikalny identyfikator. Bluetooth (dalej będzie stosowany skrót BT) oferuje nam tutaj swój unikatowy numer BDA (*Bluetooth Device Address*), który jest odpowiednikiem numeru MAC kart sieciowych. O ile pod numer BDA można się podszyć, o tyle stos BT pozwala nam dodatkowo na autoryzację połączenia poprzez podanie zgodnego kodu PIN urządzeń lub stałe ich sparowanie (wymianę i zapamiętanie kluczy połączenia). Wówczas pojawia się również możliwość szyfrowania przesyłanych danych kluczem np. 56-bitowym, a jeśli zaistnieje taka potrzeba, nawet 128-bitowym. Ponadto, urządzenie nie musi być publicznie widoczne dla innych (wyłączony tryb „*page inquiry*" tzn. bez nasłuchiwania pakietów „*inquiry*" innego urządzenia), aby można było dokonać z nim połączenia i wymiany danych. Wystarczy, że już znamy jego numer BDA. Możliwość połączenia się z urządzeniem typu *slave* wymaga aktywnego trybu „*page scan*" tj. okresowego nasłuchiwania połączeń przychodzących. Dlatego w tym trybie pobiera ono średnio więcej energii, niż urządzenie typu *master* i jest lepiej, aby np. nasz zestaw *Hands Free* czy *Head Set* łączył się z telefonem, a nie na odwrotnie.

Jak widać z powyższego, bezpieczeństwo jest zagwarantowane na wybranym przez użytkownika poziomie.

Zasięg

Kolejnym zagadnieniem jest oczekiwany zasięg połączeń. Jeśli już wybraliśmy Bluetooth, to wiadomo, że będziemy korzystali z pasma ISM (*Industrial, Scientific, Medical*) w technice FHSS (*Frequency Hopping Spread Spectrum*). Aby być w zgodzie z prawem obowiązującym w Polsce nie można przekraczać normy zdefiniowanej w Dz. U. Nr 138 poz. 972 z 2007 r. wraz z aktualizacją w Dz. U. Nr 47 poz. 277 z 2008 r. Dopuszcza ona maksymalną efektywną izotropową moc wypromieniowaną o wartości 100 mW e.i.r.p. i nakłada ograniczenie nie tylko na moc nadajnika, ale również na efektywność anteny (patrz EP+ 1/2007, „Technologie M2M", str. 14 oraz 97). W przypadku standardu Bluetooth (klasyfikacja do wersji 2.1 + EDR

włącznie, z wykluczeniem *Ultra Low Power* oraz *BT 3.0 High Speed*), rozróżniane są trzy klasy, z czego najpopularniejsze są dwie:

– kl. 1 o mocy promieniowanej do 100 mW (20 dBm) i zasięgu do 100 m,
– kl. 2 o mocy promieniowanej do 2,5 mW (4 dBm) i zasięgu do 10 m.

Podawany maksymalny zasięg jest orientacyjny i zależy m.in. od warunków propagacji w otoczeniu anteny.

Ze względu na ograniczoną pojemność akumulatora niemal wszystkie telefony komórkowe mają Bluetooth klasy 2. Zatem, aby maksymalnie zwiększyć dystans transmisji między telefonem komórkowym a naszym urządzeniem, powinniśmy ze swojej strony wybrać optymalny moduł Bluetooth pod względem mocy transmitowanej oraz czułości odbiornika (jak najwyższa czułość, tj. jak najniższa wartość poziomu szumów własnych wyrażona w jednostkach dBm), a także dopuszczalnie efektywną antenę (o ile nie użyjemy zintegrowanej z modułem *chip antenna*). Należy tutaj zauważyć, że moduły o mocy 20 dBm wcale nie będą najlepszym wyborem, jeśli nie będziemy mogli jej programowo ograniczyć. Naszym problemem jest bowiem słaby sygnał pochodzący z telefonu komórkowego. Mniejsza moc nadajnika pozwala dobrać bardziej efektywną antenę, a ta jest urządzeniem działającym dwukierunkowo, zatem poprawi także poziom sygnału odbieranego.

Z drugiej strony musimy też pamiętać, że zmniejszenie mocy nadajnika powoduje spadek mocy sygnału docierającego do odbiornika telefonu. Zatem, gdy projektujemy urządzenie mające współpracować z radiem BT różnych telefonów komórkowych, należy założyć, że czułość ich odbiorników nie będzie zbyt duża, gdyż antena będzie pracować w złych warunkach odstrajana dłonią użytkownika.

Ponadto, można użyć anteny kierunkowej. Ta będzie charakteryzowała się znacznie większą efektywnością dla danego kąta bryłowego w porównaniu do np. anteny dookólnej, ćwierćfalowej. Zminimalizowane wówczas zostaną odbierane zakłócenia sygnału. One także są wzmacniane wraz z efektywnością anteny, co nie poprawia odstępu mocy sygnału od mocy szumów na wejściu odbiornika. Ograniczenie się do mniejszego kąta bryłowego kierunków odbioru skutkuje eliminacją części szumów.

Podsumowując – nie ma idealnego sposobu na zwiększenie zasięgu. Jedyne co możemy robić, to użyć wysokoefektywnej anteny, czułego odbiornika oraz dobrego nadajnika. Na zwiększenie odstępu poziomów mocy sygnału od mocy zakłóceń niestety trudno znaleźć rozwiązanie (nie chodzi tutaj o interferencję z np. siecią WiFi, gdyż technika *Adaptive Frequency Hopping* dość skutecznie sobie z nią radzi pod warunkiem, że bliskich czy „silnych" sieci WiFi nie ma zbyt wiele). Można jednak pozbyć się elementów zakłócających i tłumiących z sąsiedztwa anteny, bliskich przetwornic impulsowych (szczególnie tych dużej mocy) i prawidłowo zaprojektować obwód drukowany (stosować pola masy, wiele drobnych przelotek na brzegach pól masy, itp.).

Realizacja

Do realizacji projektu wybrano moduł Bluetooth WT11-A-AI3 firmy Bluegiga. Ma on certyfikaty (EMC/CE, ETSI EN 300 328, FCC, RoHS, Bluetooth Qualification), które pozwalają na legalne używanie go w praktycznie każdym zakątku naszego globu. Jest to moduł klasy 1 (moc nadajnika +15 dBm, czułość odbiornika –82 dBm), z wbudowaną anteną (ceramiczna „chip" antena o maksymalnej efektywności 0,5 dBi przy 2,4 GHz) i stosem BT o nazwie iWRAP w wersji 3.0.0. To właśnie dzięki zastosowaniu tego stosu można było uniknąć zastosowania mikrokontrolera i pozwolić, aby moduł po jednorazowym skonfigurowaniu pracował autonomicznie.

Dla zmniejszenia liczby modyfikacji sterownika bramy użyto jej oryginalnego pilota, a dołączony układ Bluetooth emuluje naciśnięcie odpowiedniego przycisku (**fot. 1**).

Schemat

Schemat przykładowej bramy sterowanej poprzez Bluetooth przedstawiono na **rys. 2**. Jest on drobną modyfikacją modułu WT11SM produkcji firmy Elproma Elektronika.

Działanie obwodu jest stosunkowo proste. Do złącza JP1 lub CON1 należy doprowadzić zasilanie (napięcie stałe, niekoniecznie stabilizowane). Układ IC1 jest stabilizatorem LDO napięcia, którym zasilany jest moduł Bluetooth BT1. Rezystory R1 i R2 ustalają wartość napięcia wyjściowego. W celu uzyskania większej mocy nadawczej można je nieznacznie zwiększyć (**rys. 3**), nie przekraczając jednak granicy 3,6 V i uwzględniając margines bezpieczeństwa. Kondensatory C5, C6, C8 filtrują napięcie zasilania. Podobną rolę pełnią dwa filtry złożone z elementów C1...C4 oraz dławików L1 i L2.

Gniazdo CON1 służy przede wszystkim do wstępnej konfiguracji modułu za pomocą interfejsu UART, w którym napięcie wysokiego poziomu logicznego jest równe 3,3 V (linie RX, TX, RTS, CTS).

Sercem układu z rys. 1 jest moduł Bluetooth. Do jego wyjścia PIO7 dołączono bramkę tranzystora Q1. Pozwala to nie obciążać prądowo modułu (a dokładniej układu CSR BlueCore04, który znajduje się wewnątrz pokrywy ekranu modułu) elementami wykonawczymi. Pojawienie się na linii PIO7 stanu wysokiego powoduje, że kanał tranzystora Q1 przewodzi i załącza tranzystor Q2. Rezystor R3 ogranicza prąd bazy. Tranzystor Q2 jest zasilany napięciem z baterii oryginalnego pilota bramy (oznaczonego ramką *Gate Remote Control*), którą chcemy sterować. Oryginalny pilot posiada dwa przyciski (S1, S2), które łączą odpowiednie wyprowadzenie układu scalonego pilota bramy z zasilaniem bateryjnym poprzez rezystor RX. Włączenie tranzystora Q2 emuluje wciśnięcie przycisku S1. Rezystor R5 jest opcjonalny i ma za zadanie tłumić ewentualne zakłócenia. Oczywiście konkretne rozwiązanie zależy od konstrukcji użytego oryginalnego pilota – do niego należy dostosować układ wykonawczy (np. zmienić wartość rezystora R4 lub pominąć tranzystor kluczujący Q2, używając wprost Q1).

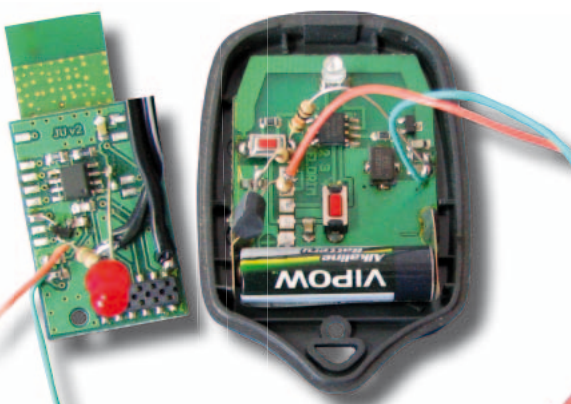
Dla zapewnienia wizualnej kontroli nad przesyłanymi do pilota bramy impulsami można opcjonalnie użyć diodę LED (z rezystorem R6). Dioda ta jest również widoczna na fot. 1.

Konfiguracja modułu

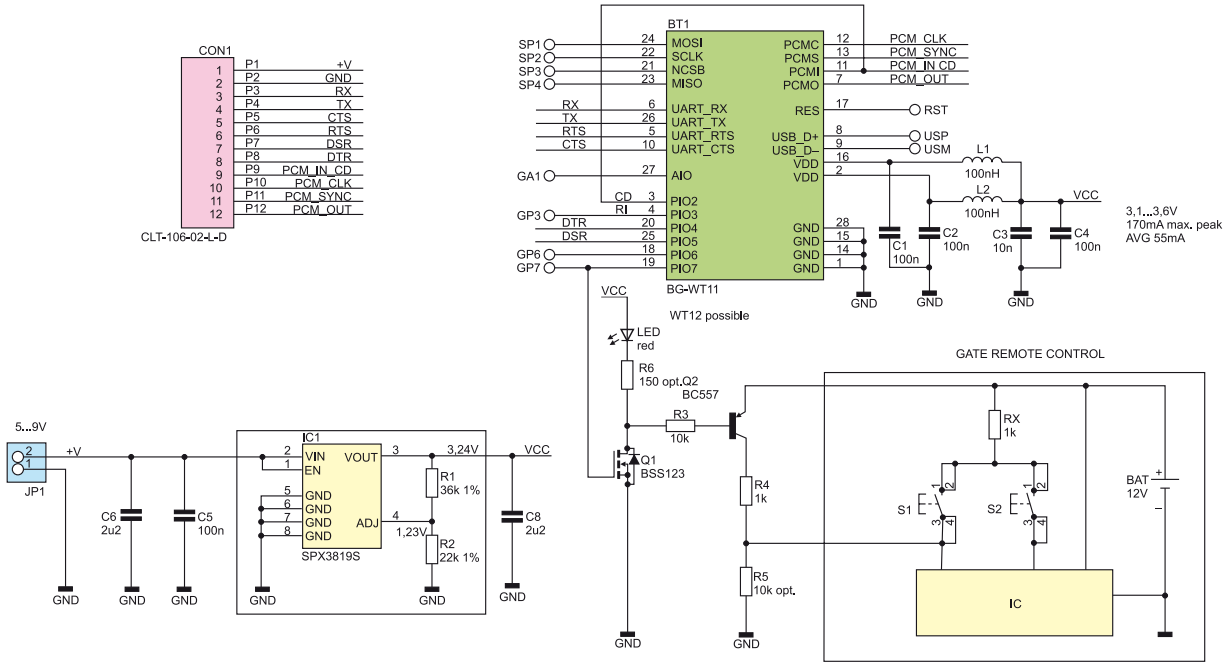
Przed przystąpieniem do zdalnego sterowania modułem trzeba go odpowiednio skonfigurować. Do tych czynności należy przede wszystkim włączenie profilu OTA (*Over The Air*) poleceniem stosu iWRAP „*SET BT PROFILE OTA <hasło>*", gdzie *<hasło>* to pożądany ciąg znaków. Pierwsze testy poprawności konfiguracji można wykonać posługując się płytką ewaluacyjną z modułem WT11 lub WT12, (**fot. 4**). Jeśli mamy do czynienia z nowym, fabrycznie skonfigurowanym modułem, to podłączamy komputer (host) do portu szeregowego RS232 płyty ewaluacyjnej (wyposażono ją w odpowiedni konwerter poziomów logicznych) i otwieramy odpowiedni port szeregowy w terminalu (np. Windows: *HyperTerminal*/*PuTTY*/*Tera Term*/*S3Term*/*Terminal by Br@y++*, Linux: *picocom*/*GtkTerm*/*minicom*, FreeBSD: *cu*) z parametrami 115200, 8n1 i włączoną sprzętową kontrolą przepływu.

Poleceniem „*set*" możemy odczytać bieżącą konfigurację modułu (**list. 1**). Podobnie po włączeniu zdalnej konfiguracji OTA (**list. 2**). Przedstawiono tam również działanie polecenia „*reset*", które jest wymagane po pierwszym aktywowaniu profilu. Z kolei polecenie „*info config*" zwraca szczegółową konfigurację, przydatną przy komunikacji ze wsparciem technicznym producenta podczas rozwiązywania problemów.

Poza wspomnianym włączeniem profilu OTA moduł Bluetooth możemy nazwać (Friendly Name) poleceniem „*SET BT NAME <nazwa>*", ustawić żądanie kodu PIN i parowania poleceniem „*SET BT AUTH*



Fot. 1. Prototyp sterownika bramy Bluetooth



Rys. 2. Schemat zdalnej bramy Bluetooth

* *<PIN>*”, wyłączyć dany profil „*SET BT PROFILE SPP/OTA*” czy wreszcie włączyć go ponownie „*SET BT PROFILE SPP on/<nazwa>*”. Jeśli moduł ma być wykrywalny w otoczeniu przez inne urządzenia, możemy tę wykrywalność ograniczyć do tylko pewnej grupy. Mamy tutaj do dyspozycji tryb standardowy GIAC (General/Unlimited Inquiry Access Code) oraz LIAC (Limited Inquiry Access Code). Tryb możemy zmienić poleceniem „*SET BT LAP 9e8b33* (GIAC)/*9e8b00* (LIAC)”. Przykłady użycia wymienionych poleceń zawiera **list. 3**.

Czasami istotną cechą dla rozpoznawania funkcjonalności innych urządzeń Bluetooth jest tzw. klasa urządzenia (Class of Device). Również i tę możemy ustawić. Służy do tego polecenie „*SET BT CLASS <CoD>*”. Odpowiednią wartość <CoD> ustalimy korzystając np. ze strony Bluetooth SIG *(https://www.bluetooth.org/Technical/AssignedNumbers/baseband.htm)*. Wymagana jest jedynie darmowa rejestracja, niezbędna jednocześnie do stosowania loga i nazwy Bluetooth na produkowanych urządzeniach.

Testy konfiguracji

Jeśli przebrnęliśmy już przez fazę podstawowej konfiguracji, czas sprawdzić działanie układu. Od strony standardu Bluetooth, profil OTA jest widziany praktycznie jak kolejny wirtualny port szeregowy w klasycznym profilu SPP (Serial Port Profile). To, co go odróżnia, to nazwa „Bluegiga iWRAP” usługi SPP/OTA w SDP (*Service Discovery Protocol*) oraz konieczność zdalnego wprowadzenia ustalonego wcześniej hasła poprzez wirtualny port szeregowy. Tuż po nawiązaniu połączenia Bluetooth i otwarciu portu należy wysłać „<hasło>\r”. Przykład komunikatu otrzymanego w konsoli fizycznego portu szeregowego jest na **list. 4**.

List. 1. Odczytanie bieżącej konfiguracji modułu

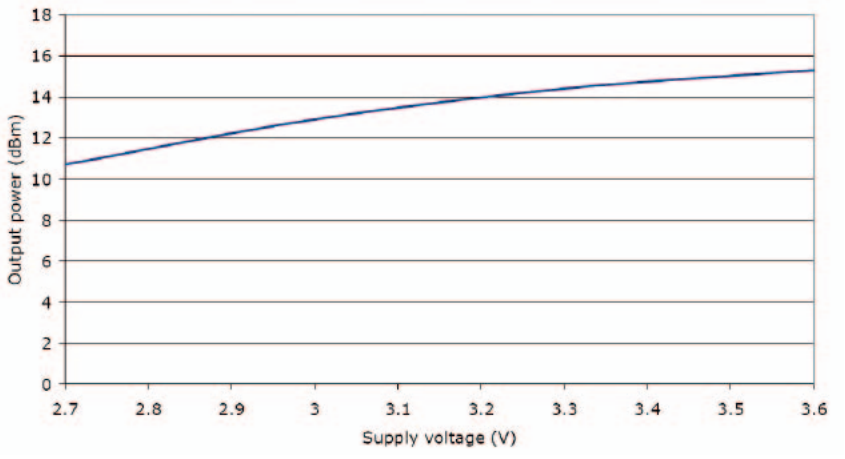
```
WRAP THOR AI (3.0.0 build 148)
Copyright (c) 2003-2008 Bluegiga Technologies Inc.
READY.
set
SET BT BDADDR 00:07:80:88:48:f1
SET BT NAME WT11
SET BT CLASS 001f00
SET BT IDENT BT:47 f000 3.0.0 Bluegiga iWRAP
SET BT LAP 9e8b33
SET BT PAGEMODE 4 2000 1
SET BT POWER 12 12 12
SET BT ROLE 0 f 7d00
SET BT SNIFF 0 20 1 8
SET CONTROL BAUD 115200,8n1
SET CONTROL CD 00 0
SET CONTROL ECHO 7
SET CONTROL ESCAPE 43 00 1
SET CONTROL MSC DTE 00 00 00 00 00 00
SET PROFILE SPP Bluetooth Serial Port
SET
```

Jeśli otrzymamy komunikat „RING OTA”, fizyczny port szeregowy staje się nieaktywny dla poleceń stosu iWRAP aż do zakończenia połączenia i otrzymania komunikatu „NO CARRIER”. W tym czasie rolę portu sterującego pełni wirtualny port szeregowy dostępny poprzez SPP/OTA.

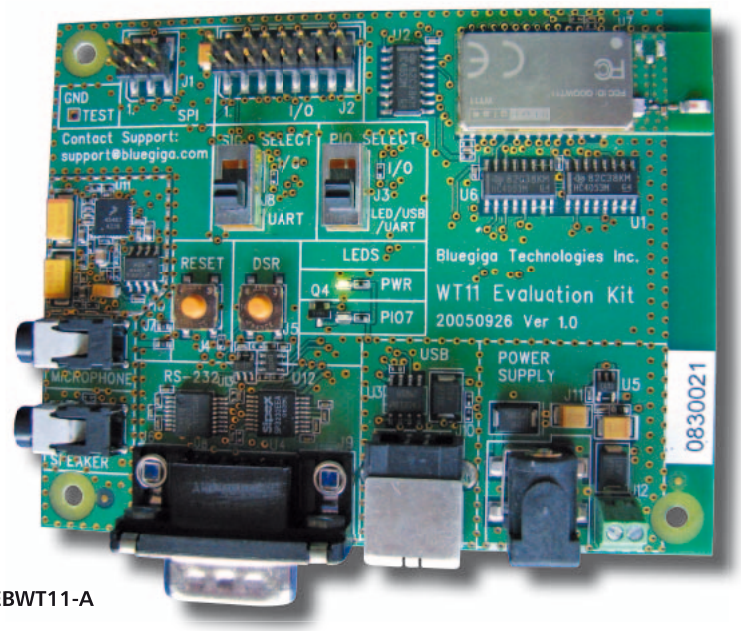
Sterowanie liniami I/O modułu

Stos iWRAP pozwala w prosty sposób sterować liniami wejścia/wyjścia. Służą do tego celu polecenia rozpoczynające się od „pio”. Najważniejszym jest „*pio set <maska> <stan>*”. Maska i stan reprezentują w kodzie szesnastkowym 8 bitów, którym odpowiadają linie I/O. Linii PIO7 odpowiada wartość 80 (**list. 5**).

Może także zaistnieć potrzeba nie tylko sterowania zdalnego, ale także lokalnego przyciskiem. Płyta rozwojowa posiada przycisk „DSR”, który jest dołączony do linii PIO5. Przykład konfiguracji dla włączania linii PIO7 przyciskiem „DSR” przedstawiono na **list. 6**.



Rys. 3. Zależność mocy wyjściowej od napięcia zasilania



Fot. 4. EBWT11-A

List. 2. Włączenie zdalnej konfiguracji OTA

```
set profile ota haslo
set
SET BT BDADDR 00:07:80:88:48:f1
SET BT NAME WT11
SET BT CLASS 001f00
SET BT IDENT BT:47 f000 3.0.0 Bluegiga iWRAP
SET BT LAP 9e8b33
SET BT PAGEMODE 4 2000 1
SET BT POWER 12 12 12
SET BT ROLE 0 f 7d00
SET BT SNIFF 0 20 1 8
SET CONTROL BAUD 115200,8n1
SET CONTROL CD 00 0
SET CONTROL ECHO 7
SET CONTROL ESCAPE 43 00 1
SET CONTROL MSC DTE 00 00 00 00 00 00
SET PROFILE SPP Bluetooth Serial Port
SET PROFILE OTA
SET
reset
WRAP THOR AI (3.0.0 build 148)
Copyright (c) 2003-2008 Bluegiga Technologies Inc.
READY.
info config
WRAP THOR AI (3.0.0 build 148)
Copyright (c) 2003-2008 Bluegiga Technologies Inc.
Compiled on Jul 23 2008 15:31:35, running on WT11 module, psr v15
        PIO=0x00fc
        - BOCK3 version 29-dev (Jun 16 2008 12:11:11) (max acl/sco 7/1)
        - Bluetooth version 2.1, Power class 1
        - Loader 5090, firmware 5090 (56-bit encryption)
        - up 0 days, 00:00, 0 connections (pool 1)
        - User configuration:
&0294 = f1eb 8cc2 851a e547
&02ad = 5457 3131
&02b1 = 000c 000c 000c
READY.
```

List. 3. Zmiana trybów pracy modułu

```
set bt auth * 1234
set bt name brama
set bt lap 9e8b00
set profile spp
set
SET BT BDADDR 00:07:80:88:48:f1
SET BT NAME brama
SET BT CLASS 001f00
SET BT AUTH * 1234
SET BT IDENT BT:47 f000 3.0.0 Bluegiga iWRAP
SET BT LAP 9e8b00
SET BT PAGEMODE 4 2000 1
SET BT POWER 12 12 12
SET BT ROLE 0 f 7d00
SET BT SNIFF 0 20 1 8
SET CONTROL BAUD 115200,8n1
SET CONTROL CD 00 0
SET CONTROL ECHO 7
SET CONTROL ESCAPE 43 00 1
SET CONTROL MSC DTE 00 00 00 00 00 00
SET PROFILE OTA
SET
reset
Wyłączenie żadania kodu PIN, przywrócenie GIAC i profilu SPP:
set profile spp on lub set profile spp moja nazwa portu szeregowego
set bt lap 9e8b33
set bt auth *
```

Jeśli z kolei interesuje nas inny stan tuż po włączeniu zasilania modułu, to możemy posłużyć się poleceniem „*set control init pio set 80 80*”.

Finalnie, dla zastosowania przedstawionego w artykule niezbędne jest generowanie impulsu. Oczywiście można to robić poleceniami ustaw stan wysoki, a następnie ustaw stan niski, jednak w przypadku utraty połączenia istnieje obawa, że stan wysoki może pozostać utrzymany na dłużej. Aby temu zapobiec, moduł będzie realizował funkcję ze sprzężeniem zwrotnym z pewnym opóźnieniem (**list. 7**). Po wydaniu polecenia „*pio set 80 80*” narastające zbocze na linii PIO7 wygeneruje wykonanie polecenia „*pio set 80 00*”.

Oprogramowanie dla telefonu komórkowego

Początkowo oprogramowanie do telefonu komórkowego było pisane i testowanie przed powstaniem układu z fot.1. Do wspomnianych celów posłużyła płytka rozwojowa EBWT11-A. Do linii PIO7 modułu WT11 poprzez tranzystor FDV301N dołączono diodę LED, która ułatwia pracę monitorując stan wspomnianej linii.

Do napisania programu wybrano język, a zarazem platformę, Java Micro Edition (J2ME) wraz z rozszerzeniem dla obsługi stosu Bluetooth JSR82 lub inaczej JABWT (*Java API for Bluetooth Wireless Technology*). Wybór ten niestety wyklucza uruchamianie aplikacji na PDA, Pocket PC czy Smartphone z systemem Microsoft Windows Mobile i domyślną maszyną Javy MIDlet Manager, gdyż nie wspiera ona JSR82. Dla tych urządzeń pozostaje użyć innej wirtualnej maszyny J2ME lub napisać oprogramowanie dedykowane dla Windows Mobile.

Do artykułu dołączono dwie aplikacje wraz z ich pełnym kodem źródłowym pisanym w środowisku *Sun NetBeans 5.5.1* oraz testowanym wstępnie na wirtualnej maszynie *Sun Java Wireless Toolkit 2.5.1*. Są to aplikacje demonstracyjne, przeznaczone do celów niekomercyjnych, dlatego też ich kod nie jest idealny. Pomimo tego sprawdziły się w naszej firmowej społeczności do otwierania i zamykania bramy wjazdowej przed przybyciem osoby posiadającej pilota.

Aplikacja BTgpio1 jest bardzo uproszczona, ale dobrze obrazuje samo sterowanie i pozwala łatwo debugować kod. Aplikacja BTBrama1 posiada bardziej dopracowany interfejs użytkownika oraz rozszerzone możliwości: skanowanie otoczenia Bluetooth zarówno w trybie GIAC jak i LIAC, możliwość zmiany docelowego numeru BDA (po odnalezieniu modułu z profilem OTA) oraz hasła OTA sterowanego modułu, użycie wielowątkowości dla poprawnego działania GUI na różnych testowanych urządzeniach, ograniczony czas utrzymywania połączenia (pozwala na szybszą reakcję urządzenia końcowego w przypadku zmiany decyzji użytkownika telefonu, bez potrzeby ponownego nawiązywania połączenia w krótkim odstępie czasu).

Skompilowane pliki jar i jad wymienionych aplikacji do zainstalowania na telefonie komór-

List. 4. Połączenie i rozłączenie przy wykorzystaniu OTA (komunikat pojawi się po poprawnym, zdalnym wprowadzeniu hasła)
```
RING 0 00:15:de:25:2e:a7 2 OTA
NO CARRIER 0 ERROR 0
```

List. 5. Zmiana stanu linii wyjściowej PIO7
```
pio set 80 80 (włącz)
pio set 80 00 (wyłącz)
```

List. 6. Sterowanie ręczne (przycisk DSR na płycie ewaluacyjnej)
```
set control bind 1 20 fall pio set 80 80
reset
po wciśnięciu ujrzymy w konsoli terminala:
WRAP THOR AI (3.0.0 build 148)
Copyright (c) 2003-2008 Bluegiga Technologies Inc.
READY.
pio set 80 80
wyłączenie powyższego przypisania polecenia do linii I/O:
set control bind 1
```

List. 7. Generowanie impulsu
```
set control bind 0 80 rise pio set 80 00
reset
pio set 80 80
pio set 80 00 (wygenerowane przez sam moduł)
```

kowym znajdują się w katalogach *dist* projektów. Ponieważ każdy może je zainstalować i uruchomić na swoim telefonie komórkowym, poznanie ich nieskomplikowanej obsługi pozostawiam Czytelnikowi.

Co dalej?

Najnowsza wersja stosu iWRAP w wersji 4 (w trakcie pisania artykułu ukazała się wersja beta) oferuje kilka nowych profili Bluetooth, w tym standard medyczny HDP (Health Device Profile) oraz własnościowy BGIO (Bluegiga I/O Profile). Ten drugi z pewnością może zastąpić używany w artykule profil OTA, ale oczywiście wybór należy do Konstruktora. Zachęcam do śledzenia wydarzeń.

Prezentowane w artykule podejście do sterowania jest stosunkowo bezpieczne, jednak niezalecane do stosowania w krytycznych aplikacjach. Wadą jest również podział telefonów na wspierające standard JSR-82 oraz niewspierające, zamiast na posiadające interfejs komunikacji Bluetooth i nieposiadające.

Rozwiązaniem tego problemu może być odwrócenie ról master-slave w relacji bramka dostępowa (np. moduł Bluetooth) – telefon komórkowy poprzez wykrywanie telefonów publicznie widocznych (GIAC) i wyszukiwanie wśród nich znanych numerów BDA. Telefony komórkowe niestety zwykle nie mają możliwości ustawienia trybu widoczności LIAC. Nie ma róży bez kolców. Jak wspomniano na początku, numer BDA można sklonować bez większych trudności.

Inna metoda, to rezygnacja z publicznej widoczności telefonu, a jedynie dokonywanie próby połączenia i rozłączenia z nim na podstawie znanego wcześniej numeru BDA. Jednak takie odpytywanie może trwać zbyt długo przy długiej liście telefonów-kluczy. Pewnym sposobem może być analiza usług (SDP) i/lub użycie profilu DI (*Device Identification*) celem rozpoznania modelu telefonu i utrudnienie jego klonowania. W końcu możemy użyć wymienionych metod w połączeniu z parowaniem urządzeń. Dokonamy tego, podobnie jak w przypadku słuchawek Bluetooth, za pomocą podania jednorazowo oryginalnego kodu PIN w „warunkach bezpiecznych" (bez podejrzenia możliwości podsłuchania komunikacji, używając np. ograniczonego do 1 m zasięgu). „Warunki bezpieczne" są konieczne, aby niepowołany intruz nie uzyskał uprawnień do pobrania naszej prywatnej książki telefonicznej czy w inny, niechciany sposób korzystał z naszego telefonu. Jeśli mamy już utworzone klucze parujące, możemy na chwilę wykonać połączenie szyfrowane do telefonu komórkowego w profilu DUN (*Dial-Up Networking*), odpytać go poleceniem AT o numer IMEI, a następnie rozłączyć. W ten sposób znacznie podniesiemy bezpieczeństwo autoryzacji. Niemniej, długoterminowe sniffowanie pakietów Bluetooth takiej komunikacji, może być również powodem do zmartwień (próba łamania kluczy szyfrujących). Dlatego też należy ostrożnie ustawiać moc nadajnika i przewidywaną odległość między bramką dostępową a telefonem. Nie oznacza to jednak, że mamy przesadzać, gdyż wszystko zależy od tego, co chronimy. To samo dotyczy przecież tradycyjnych zamków, pilotów itp. Można je wreszcie tak samo zapomnieć zabrać ze sobą czy zgubić. Jedno jest pewne - rozładowanie baterii pilota, choć niezbyt częste, oznacza wycieczkę do sklepu, zaś telefon zwykle możemy dołączyć np. do ładowarki samochodowej, co jednak nie będzie zbyt często potrzebne, gdyż i tak dbamy o stan naładowania akumulatora telefonu.

Niestety, wymienione powyżej sposoby wymagają oprócz modułu Bluetooth użycia hosta (np. mikrokontrolera), a to już wykracza poza ramy niniejszego artykułu.

Janusz Użycki
ELPROMA ELEKTRONIKA
http://www.m2mgsm.com
j.uzycki@elproma.com.pl