

Moduł 4 przekaźników z interfejsem Bluetooth

DODATKOWE MATERIAŁY NA FTP:

<ftp://ep.com.pl>

USER: 22086, PASS: 218655ee

W ofercie AVT*

AVT-5552

Podstawowe informacje:

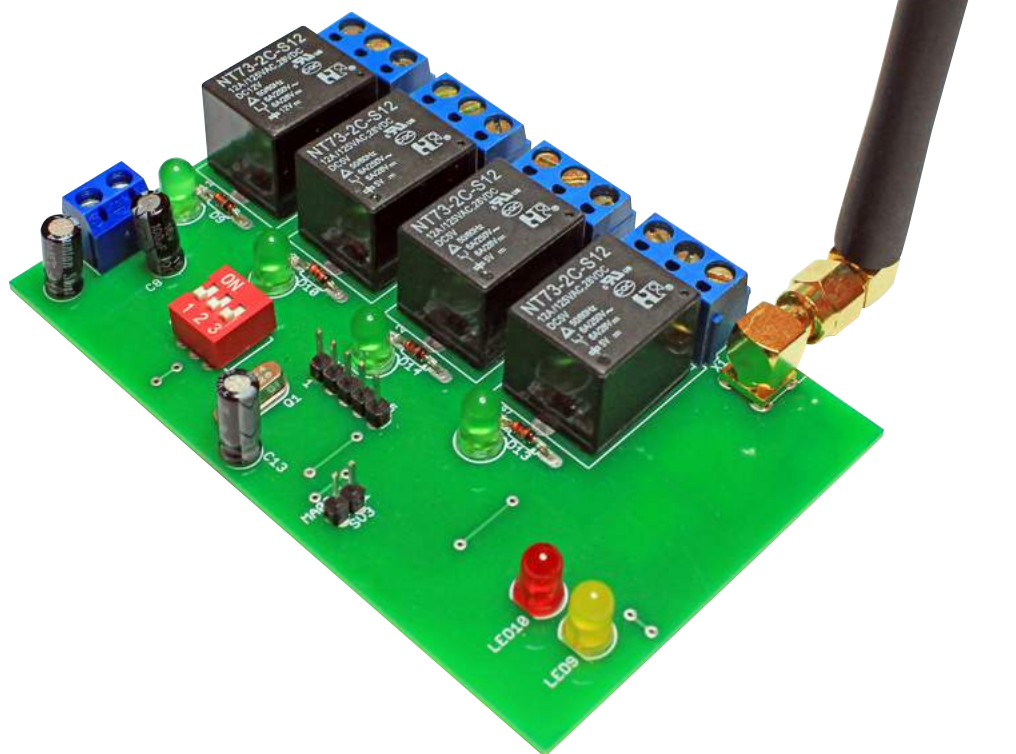
- 4 przekaźniki wykonawcze (125 V AC lub 28 V DC/12 A).
- Sygnalizacja statusu za pomocą diod LED oraz aplikacji sterującej.
- Wyłącznik awaryjny wyłączający wszystkie przekaźniki. Możliwość blokady sterowania.
- Sterowanie za pomocą aplikacji dla systemu Android (lub wykonanej samodzielnie).
- Mikrokontroler ATmega8, moduł Bluetooth BTM222.

Projekty pokrewne na FTP:

(wymienione artykuły są w całości dostępne na FTP)

AVT-5551	Przełącznik z interfejsem Bluetooth (EP 9/2016)
AVT-1916	Konfigurowalny przełącznik 4-kanalowy (EP 8/2016)
AVT-1914	Uniwersalny 2-kanalowy moduł przekaźnikowy (EP 8/2016)
AVT-1890	Moduł przekaźników z USB (EP 6/2016)
AVT-5538	Moduł załączający z triakami (EP 5/2016)
AVT-3130	Moduł I/O sterowany przez USB (EdW 5/2015)
AVT-1815	4-kanalowy przełącznik sterowany dowolnym pilotem IR (EP 8/2014)
AVT-5368	Programowalny moduł przekaźników (EP 11/2012)
AVT-1679	Moduł wykonawczy z triakami (EP 6/2012)
AVT-1659	8-kanalowy miniaturowy moduł przekaźników (EP 1/2012)
AVT-1656	Uniwersalny moduł wykonawczy (EP 12/2011)
AVT-1560	8-kanalowa karta przekaźników (EP 2/2010)
AVT-1481	Przekaźnikowy moduł wykonawczy (EP 8/2008)
AVT-925	Karta przekaźników na USB (EP 4/2006)

* Uwaga: Zestawy AVT mogą występować w następujących wersjach:
 AVT xxxx UK to zaprogramowany układ. Tylko i wyłącznie. Bez elementów dodatkowych.
 AVT xxxx A płytka drukowana PCB (lub płytki drukowane, jeśli w opisie wyraźnie zaznaczono), bez elementów dodatkowych.
 AVT xxxx A+ płytka drukowana i zaprogramowany układ (czyli połączenie wersji A i wersji UK) bez elementów dodatkowych.
 AVT xxxx B płytka drukowana (lub płytki) oraz komplet elementów wymienionych w załączniku pdf to nie innego jak zmontowany zestaw B, czyli elementy wmontowane w PCB. Należy mieć na uwadze, że o ile nie zaznaczono wyraźnie w opisie, zestaw ten nie ma obudowy ani elementów dodatkowych, które nie zostały wymienione w załączniku pdf oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można pobrać, klikając w link umieszczony w opisie kitu).
 AVT xxxx CD Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf. Podczas składania zamówienia upewnij się, którą wersję zamawiasz (UK, A, A+, B lub C). <http://shlep.ovt.pl>



Moduł służy do bezprzewodowego sterowania za pośrednictwem interfejsu Bluetooth dowolnymi 4 urządzeniami podłączonymi do przekaźników wykonawczych. W dowolnym momencie pracy układu można zablokować możliwość bezprzewodowego sterowania wyjść mocy. Ze względów bezpieczeństwa układ został także wyposażony w awaryjny wyłącznik wszystkich przekaźników bez konieczności używania aplikacji.

Rekomendacje: urządzenie jest szczególnie przydatne wszędzie tam, gdzie jest konieczne zdalne włączanie i wyłączanie różnych odbiorników prądu. Może być stosowane w roli sterownika oświetlenia lub urządzeń.

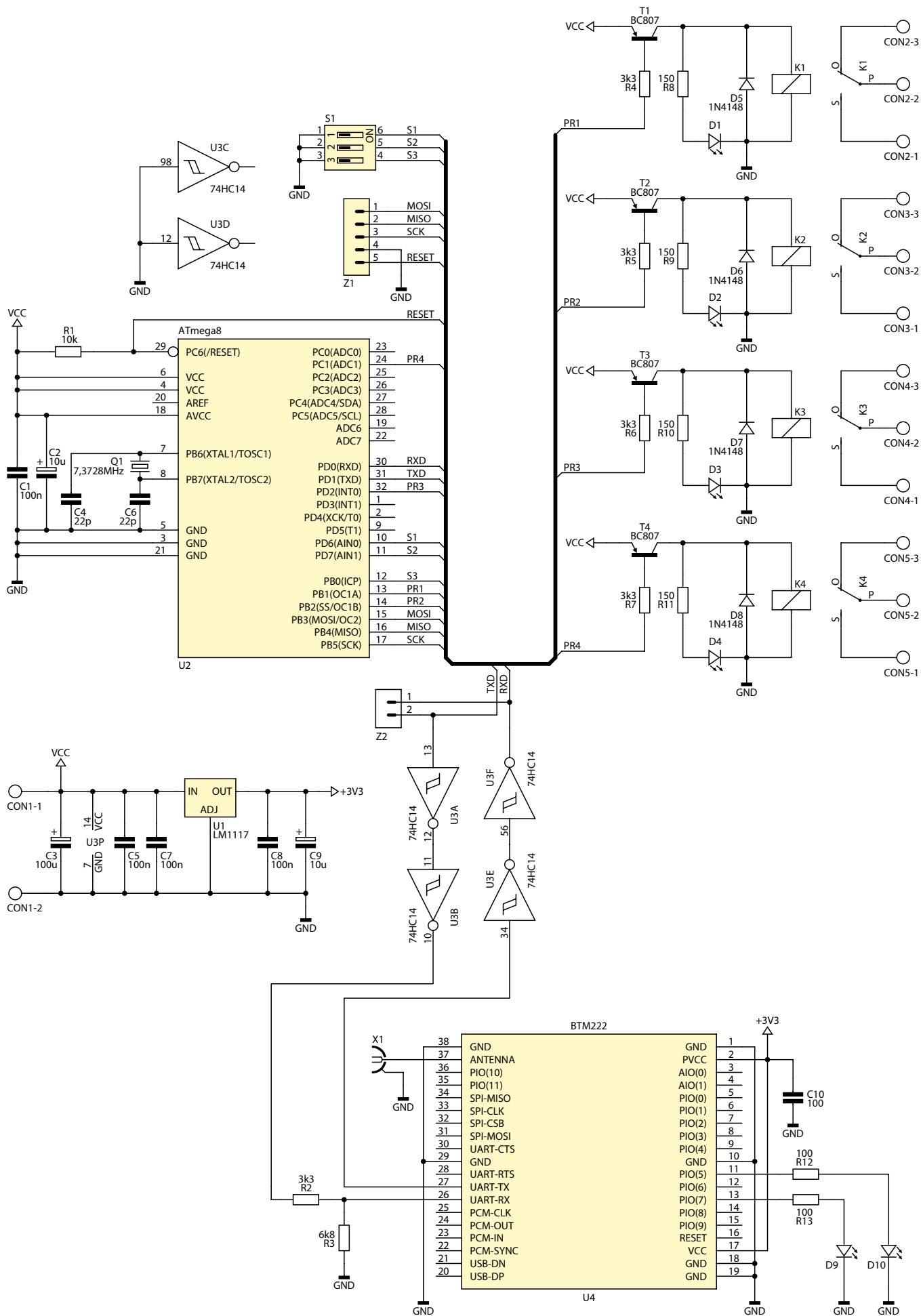
Schemat ideowy układu przedstawiono na **rysunku 1**. Sercem układu jest mikrokontroler ATmega8 pracujący z zewnętrznym rezonatorem kwarcowym 7,3728 MHz. Częstotliwość taktowania mikrokontrolera tak dobrano, aby błąd transmisji układu USART był jak najmniejszy. Rezystor R1 (10 kΩ) zasila wyprowadzenie zerujące, aby mikrokontroler nie został zrestartowany przez zaburzenia mogące wystąpić na doprowadzeniu. Przekaźniki są włączane kluczami

tranzystorowymi. Rezystory R4...R7 ograniczają prąd baz tranzystorów, a diody D5...D8 zabezpieczają tranzystory przed przepięciami, które występują podczas wyłączenia przekaźników. Rezystory R8...R11 ograniczają prąd diod LED D1...D4 wizualizujących stan przekaźników.

Jako moduł radiowy Bluetooth zastosowano BTM-222 firmy Rayson pracujący w klasie 1 i zapewniający zasięg rzędu 100 metrów. Do uzyskania takiego zasięgu jest wymagana antena zewnętrzna. W projekcie wykorzystano pokazaną na **fotografii 2** antenę BLU-ANT01R o impedancji 50 Ω, pracującą w zakresie częstotliwości 2,4...2,5 GHz z zyskiem energetycznym 1 dBi. Antenę dołączono do gniazdka SMA

umieszczonego na płytce PCB. Komunikacja z modułem jest zwyczajną transmisją szeregową RS232, jednak moduł pracuje przy poziomach logicznych CMOS 3,3 V, a mikrokontroler 5 V, dlatego zastosowano konwerter poziomów zbudowany z użyciem bramek 74HC14 oraz dzielnika rezystancyjnego R2/R3. Wejścia niewykorzystanych bramek dołączono do masy, aby uniknąć zakłóceń.

Program sterujący mikrokontrolerem napisano za pomocą Bascoma AVR. W pętli głównej program sprawdza stany przekaźników stykowych bufor wejściowy. Włączaniu i wyłączaniu poszczególnych przekaźników przypisano komendy w postaci kodów ASCII, dlatego zmiana stanu danego przekaźnika odpowiada przesłaniu



Rysunek 1. Schemat ideowy modułu z interfejsem Bluetooth

odpowiedniego znaku z terminalu. Z racji zastosowania możliwości blokowania sterowania bezprzewodowego stan przycisku blokującego jest co kilka sekund przesyłany do odbiornika w postaci wyrazów: „ENABLED” oraz „DISABLED”. Takie rozwiązanie nie jest konieczne, ale zostało wprowadzone,

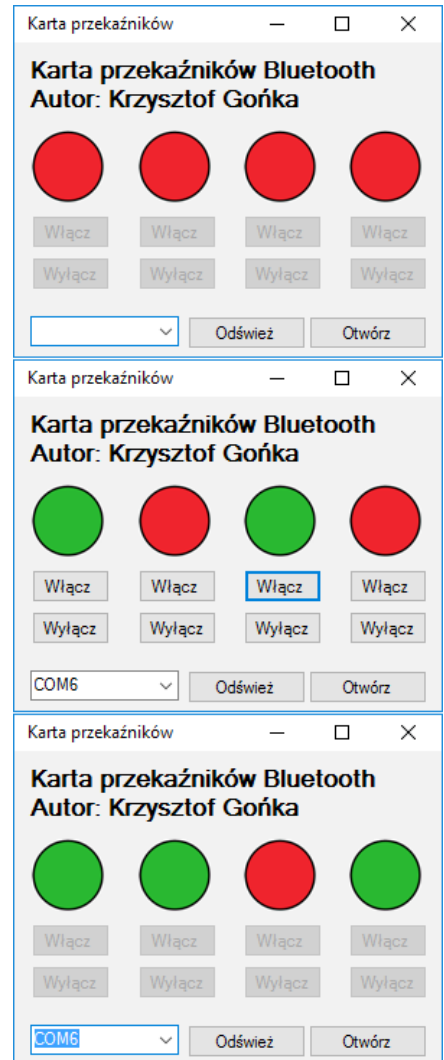
aby nie zakłócić odzwierciedlania stanu przycisków w programie komputerowym.

Oprogramowanie

Przedstawiony powyżej układ jest jedynie przystawką do komputera, laptopa lub telefonu komórkowego i aby w pełni go wykozystać, konieczny jest program komputerowy lub aplikacja dla systemów Android. Dla prezentowanego układu wykonano aplikację dla komputerów PC w C# pracującą na platformie .NET Framework. Dlatego do uruchomienia jest wymagane zainstalowanie środowiska Microsoft .NET Framework w wersji 4.0 lub nowszej. Fragment programu reprezentujący obsługujący zdarzenia odebrania znaków przez port szeregowy pokazano na **listingu 1**. W wypadku odebrania komendy zablokowania odbioru przez Bluetooth przyciski zmiany stanu przekaźników zostają zablokowane do czasu otrzymania komendy oznaczającej odblokowanie sterowania bezprzewodowego. Okno aplikacji w różnych stanach pracy przedstawiono na **rysunku 3**. Pierwsze okno przedstawia wygląd aplikacji po włączeniu bez ustawienia i otwarcia portu. Z kolei kolejne przedstawia aplikację podczas działania. Zrzut aplikacji w przypadku zablokowania sterowania bezprzewodowego obrazuje ostatnie okno. Program przy uruchomieniu pobiera wszystkie dostępne nazwy portów szeregowych i ładuje je do listy rozwijanej. W przypadku, gdy zostanie wyświetlone nowe urządzenie już po włączeniu programu, nie zostanie wykryte. W tym celu zastosowano ikonę odśwież, pobierającą aktualne nazwy portów. Dodatkowo ikona ta służy do zamykania otwartego portu i może być użyta, jeśli użytkownik zechce zmienić nazwę portu w trakcie działania programu.

Montaż i uruchomienie

Na **rysunku 4** pokazano schemat montażowy układu. W projekcie wykorzystano mikrokontroler ATmega8 w obudowie



Rysunek 3. Okno aplikacji w różnych stanach pracy

TQFP32, układ 74HC14 w obudowie SO14 oraz moduł BTM222. Istnieje kilka technik montażu powierzchniowego układów w obudowach TQFP czy SO. Jedną z nich jest użycie stacji na gorące powierze oraz odpowiedniego topnika. Montaż można dokonać również zwykłą stacją lutowniczą wyposażoną w specjalny grot minifala lub zwykły grot szpilkowy,

Wykaz elementów

Rezystory: (SMD 1206)

- R1: 10 kΩ
- R2: 3,3 kΩ
- R3: 6,8 kΩ
- R4...R7: 3,3 kΩ
- R8...R11: 150 Ω
- R12, R13: 100 Ω

Kondensatory:

- C1, C5, C7, C8, C10: 100 nF (SMD 1206)
- C2, C9: 10 μF/16 V
- C3: 100 μF/16 V
- C4, C6: 22 pF (SMD 1206)

Półprzewodniki:

- D1...D4: dioda LED, zielona, 5 mm
- D5...D8: 1N4148
- D9: dioda LED, żółta, 5 mm
- D10: dioda LED, czerwona, 5 mm
- T1...T4: BC807
- U1: LM1117
- U2: ATmega8 (TQFP32)
- U3: 74HC14
- U4: BTM222

Inne:

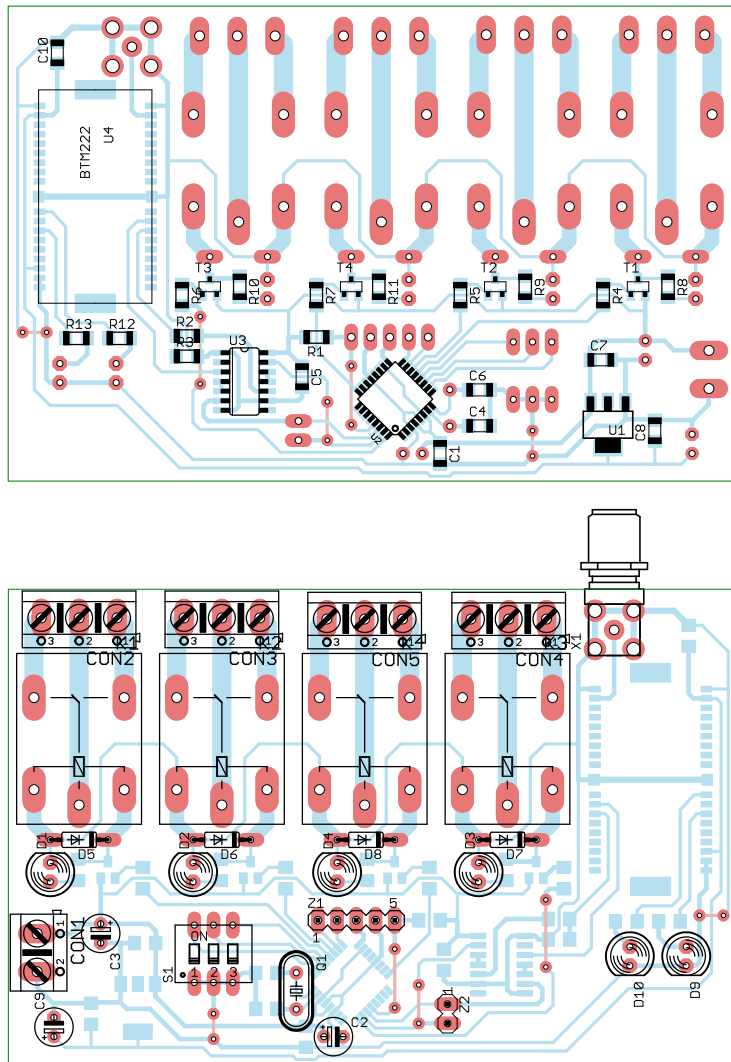
- Q1: kwarc 7,3728 MHz
- X2: złącze SMA, antena 2,4 GHz
- Z1: listwa goldpin 5 pin
- Z2: listwa goldpin 2 pin
- CON1: złącze śrubowe ARK500/2
- CON2...CON5: złącze śrubowe ARK500/3



Fotografia 2. Antena BLUE-ANT001 na pasmo 2,4 GHz

Listing 1. Fragment programu reprezentujący obsługujący zdarzenia odebrania znaków przez port szeregowy

```
private void serialPort1_DataReceived(object sender, SerialDataReceivedEventArgs e)
{
    string s = serialPort1.ReadTo(„\r\n”);
    if (s == „DISABLED”)
    {
        button1.Invoke(new EventHandler(delegate { button1.Enabled = false; }));
        button2.Invoke(new EventHandler(delegate { button2.Enabled = false; }));
        button3.Invoke(new EventHandler(delegate { button3.Enabled = false; }));
        button4.Invoke(new EventHandler(delegate { button4.Enabled = false; }));
        button5.Invoke(new EventHandler(delegate { button5.Enabled = false; }));
        button6.Invoke(new EventHandler(delegate { button6.Enabled = false; }));
        button7.Invoke(new EventHandler(delegate { button7.Enabled = false; }));
        button8.Invoke(new EventHandler(delegate { button8.Enabled = false; }));
    }
    if (s == „ENABLED”)
    {
        button1.Invoke(new EventHandler(delegate { button1.Enabled = true; }));
        button2.Invoke(new EventHandler(delegate { button2.Enabled = true; }));
        button3.Invoke(new EventHandler(delegate { button3.Enabled = true; }));
        button4.Invoke(new EventHandler(delegate { button4.Enabled = true; }));
        button5.Invoke(new EventHandler(delegate { button5.Enabled = true; }));
        button6.Invoke(new EventHandler(delegate { button6.Enabled = true; }));
        button7.Invoke(new EventHandler(delegate { button7.Enabled = true; }));
        button8.Invoke(new EventHandler(delegate { button8.Enabled = true; }));
    }
}
```

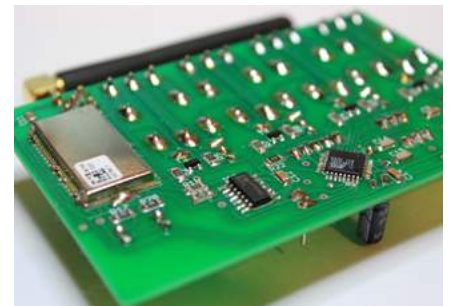


Rysunek 4. Schemat montażowy modułu z interfejsem Bluetooth

stosując cynę o niewielkiej średnicy. Na **fotografii 5** przedstawiono zmontowaną płytkę od strony ścieżek, a na **fotografii 6** od strony elementów. Przy montażu należy zwrócić uwagę, aby nie zrobić

zwarcia między wyprowadzeniami oraz aby nie uszkodzić termicznie układu. Po wlutowaniu mikrokontrolera, układu 74HC14 i modułu BTM222, należy wlutować pozostałe elementy SMD aż po zworki

Tabela 1. Ustawienia bitów bezpieczników	
Fusebit	Stan
BODLEVEL	Niezaprogramowany
BODEN	Niezaprogramowany
SUT1	Niezaprogramowany
SUT0	Niezaprogramowany
CKSEL3	Niezaprogramowany
CKSEL2	Niezaprogramowany
CKSEL1	Zaprogramowany
CKSEL0	Niezaprogramowany
RSTDISBL	Niezaprogramowany
WTDON	Niezaprogramowany
SPIEN	Zaprogramowany
CKOPT	Niezaprogramowany
EESAVE	Niezaprogramowany
BOOTSZ1	Zaprogramowany
BOOTSZ0	Zaprogramowany
BOOTRST	Niezaprogramowany



Fotografia 5. Zmontowany moduł – widok od spodu

i elementy do montażu przewlekane, od najmniejszych po największe.

Wszystkie potrzebne sygnały do programowania mikrokontrolera zostały wyprowadzone na złącze szpilkowe Z1, dzięki

REKLAMA

ELEKTRONIKA PRAKTYCZNA

na tabletach z systemami iOS i Android





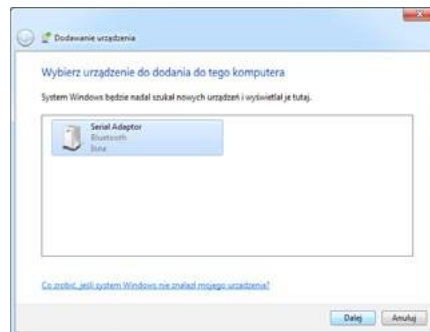
Fotografia 6. Zmontowany moduł – widok od góry

czemu nie trzeba programować mikrokontrolera w podstawie przed wlutowaniem, a można tego dokonać po zmontowaniu układu. Bity konfiguracyjne mikrokontrolera należy ustawić zgodnie z tabelą 1, w systemie heksadecymalnym powinny wyglądać następująco:

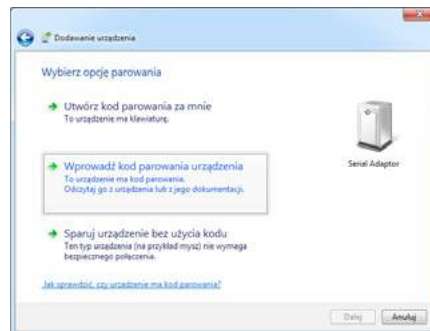
1. Fuse High Byte: 0xD9.
2. Fuse Low Byte: 0xFD.

Podczas tej czynności należy być ostrożnym, gdyż przypadkowe błędne ustawienie bitów konfiguracyjnych może przynieść niekorzystne skutki, a nawet zablokowanie mikrokontrolera. Na płytce znajdują się trzy diody. Zielona reprezentuje pracę układu (zasilanie), żółta stan połączenia, a czerwona sygnalizuje, że moduł odbiera dane. Po wlutowaniu wszystkich elementów i zaprogramowaniu mikrokontrolera układ jest gotowy do użycia. Przed włączeniem programu należy ustawić wirtualny

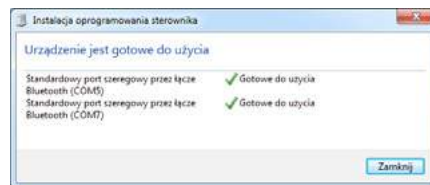
kanal szeregowy COM, do którego będzie dołączony moduł Bluetooth. W tym celu należy kliknąć na ikonę modułu Bluetooth prawym przyciskiem myszy i wybrać *Dodaj urządzenie*. Po ukazaniu się okna, jak na rysunku 7, należy wybrać znalezione urządzenie i wybrać *Dalej*. W kolejnym oknie wybrać opcję *Wprowadź kod parowania urządzenia (rysunek 8)*. Następnie należy wpisać kod „1234” i zakończyć proces parowania. W tym momencie zostanie wyświetlone okno informujące, do którego portu zostało dołączone wybrane urządzenie (rysunek 9) – należy zapamiętać pierwszą nazwę (w omawianym przypadku COM5). Po zakończeniu procesu parowania trzeba włączyć program. W pierwszym kroku należy z listy wyboru wybrać wcześniej otrzymaną nazwę portu (w omawianym przykładzie COM5) i wcisnąć *Otwórz port*. W tym momencie żółta



Rysunek 7. Wybranie parowanego urządzenia Bluetooth



Rysunek 8. Wprowadzenie kodu parowania



Rysunek 9. Wyświetlenie okna informującego o wirtualnym porcie COM

dioda powinna przestać migać i świecić ciągłym światłem, co świadczy o poprawnym zestawieniu połączenia programu z urządzeniem.

Krzysztof Gońka
krzysztof.gonka@interia.pl

REKLAMA

http://m.ep.com.pl

Najlepszy Mobilny Adres w Sieci