

Jak „oswoić” i skłonić do pracy transceiver ISM CC1100, część 1

Układ CC1100 firmy Texas Instruments (która wchłonęła pierwotnego producenta, czyli firmę Chipcon) jest miniaturowym układem nadawczo-odbiorczym służącym do przesyłania danych w postaci cyfrowej. Jest to układ typu SRD i ISM, co oznacza, że umożliwia dwustronną transmisję danych na krótkim zasięgu w zastosowaniach przemysłowych, naukowych lub medycznych.

Podstawowe parametry

CC1100 odznacza się dużymi możliwościami. Najbardziej podstawowe i interesujące parametry to:

- Praca w pasmach od 300 MHz do 928 MHz. Częstotliwość pracy jest ustawiana programowo. Dla zakresów powyżej 800 MHz należy zmienić wartości niektórych elementów w obwodach wejściowych. Możliwości techniczne pozwalają na pracę w pasmach częstotliwości używanych zarówno w Europie, jak i w USA oraz Japonii. W Polsce dostęp do określonych częstotliwości reguluje Krajowa Tablica Przeznaczeń Częstotliwości. Jest to akt prawny dość często nowelizowany i zainteresowanych jego aktualną wersją zachęcam do poszukiwań w Internecie.
- Szybkość transmisji danych ustawiana programowo w zakresie od 1,2 kb/s do 500 kb/s.
- Duża czułość odbiornika, nawet -110 dBm przy szybkości transmisji 1,2 kb/s dla BER nie przekraczającej 1%.
- Moc wyjściowa nadajnika do 10 dBm ustawiana programowo w szerokim zakresie.
- Dwa oddzielne 64-bajtowe bufora nadawczo-odbiorcze. Dzięki znacznemu zautomatyzowaniu nadawania i odbierania danych układ może współpracować nawet z bardzo wolnymi mikro-

Wartykule znajdują się wskazówki jak wykorzystać i oprogramować układ radiowego transceivera firmy Texas Instruments. Poza tym będzie też garść praktycznych porad dla tych Czytelników, którzy po raz pierwszy stykają się z opisywanym układem. W tej części artykułu podajemy kilka podstawowych pojęć i przydatnych w późniejszej pracy procedur.

kontrolerami lub w systemach, w których mikrokontroler jest silnie obciążony innymi zadaniami.

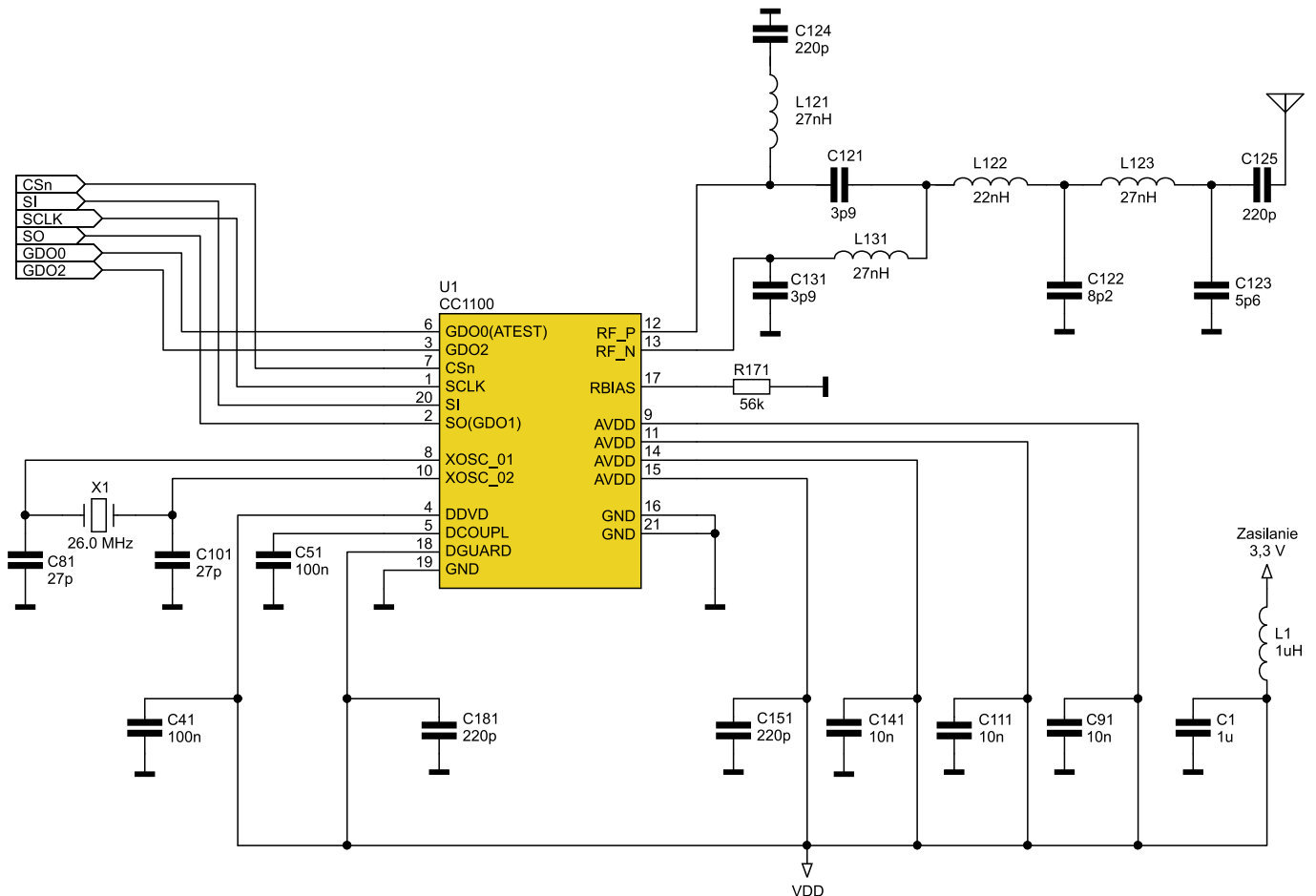
- Wewnętrzny układ automatycznej korekcji błędów pozwalający eliminować lub częściowo naprawiać błędy transmisji.
- Wybierany programowo format transmisji radiowej: FSK, OOK, ASK. Pozwala to lepiej dopasować się do warunków, w jakich pracuje układ i polepszyć jakość transmisji zmniejszając stopę błędów BER.
- Wybierany programowo numer kanału radiowego z możliwością ustawienia do 255 kanałów.
- Możliwość wprowadzenia układu w tryb czuwania, w którym średni pobierany prąd redukowany jest do wartości mniejszej od 1 μ A. W tym trybie układ po zidentyfikowaniu początku transmisji radiowej automatycznie przechodzi do normalnej pracy, może także wybudzić współpracujący mikrokontroler, jeżeli także został uśpiony. Ze względu na oszczędność w poborze prądu jest to tryb pracy idealny przy zaawansowanych konstrukcjach z zasilaniem bateryjnym.
- Magistrala SPI używana zarówno do programowania aktualnych parametrów pracy układu, jak i do transmisji i odbioru danych.
- Zasilanie pojedynczym napięciem stałym w zakresie od 1,8 V do 3,3 V.

Dla wielu czytelników kluczowym parametrem będzie z pewnością zasięg transmisji. Zależy on od wielu czynników, między innymi od warunków zewnętrznych (przeszkody pomiędzy nadajnikiem a odbiornikiem, poziom zakłóceń pochodzących np. od linii energetycznych), zastosowanej anteny, mocy nadajnika, szybkości transmisji radiowej (im wyższa szybkość, tym zasięg maleje, a rośnie pobór prądu). CC1100 jest układem SRD i należy przyjąć, że przy niskiej szybkości transmisji i niskim lub średnim poziomie zakłóceń w terenie otwartym można osiągnąć zasięg 300...350 metrów. Jednak przy bardzo sprzyjających okolicznościach może się zdarzyć zadawalająca transmisja nawet na dystansie 1000 metrów. W budynku wielopiętrowym można uzyskać transmisję w obrębie 10 pięter.

Pożyteczne dodatki

Pracując z układem CC1100 dobrze jest zaopatrzyć się w kilka dodatkowych pomocy. Przede wszystkim jest to jego dokumentacja techniczna w języku angielskim, którą w formacie PDF można pobrać ze strony producenta <http://focus.ti.com/lit/ds/symlink/cc1100.pdf>.

Wszystkie funkcje i parametry układu można zmienić poprzez zapis odpowiednich wartości do wewnętrznych rejestrów układu. Niezbędnym narzędziem do generacji gotowych plików ustawień rejestrów jest program „SmartRF Studio”. Wygenerowane pliki moż-



Rys. 1. Schemat typowej aplikacji transceiwera dla pasma 433 MHz

na w prosty sposób wykorzystać we własnym oprogramowaniu mikrokontrolera. „SmartRF Studio” w postaci pliku *swrc046g.zip* jest dostępny pod adresem: <http://focus.ti.com/docs/toolsw/folders/print/smarttrftm-studio.html>. Także na tej samej stronie internetowej znajduje się plik instrukcji do programu *swru070c.pdf* w języku angielskim. O sposobie posługiwania się programem „SmartRF Studio” będzie mowa w dalszej części artykułu.

Schemat przykładowej aplikacji

Układ CC1100 do normalnej pracy potrzebuje niewiele elementów dodatkowych. Głównie są to indukcyjności i pojemności obwodów wejściowych i kilka pojemności odsprężających.

Na **rys. 1** został pokazany schemat typowej aplikacji transceiwera dla pasma 433 MHz. Układ może być fragmentem większego urządzenia lub może stanowić osobny moduł. Sposób wykonania takiego modułu został opisany w innym artykule. Do prób z CC1100 potrzebny

będzie albo samodzielnie zmontowany układ, albo kupiony moduł.

Połączenie CC1100 z mikrokontrolerem

Układ CC1100 komunikuje się z mikrokontrolerem przy pomocy standardowej magistrali SPI. W skład magistrali wchodzi następujące sygnały:

- CSn - (wejście) stan niski uaktywnia magistralę SPI,
- SI - (wejście) szeregowe dane wejściowe,

- SCLK - (wejście) zegar taktujący magistrali,
- SO - (wyjście) szeregowe dane wyjściowe.

Dodatkowym, użytecznym wprowadzeniem nie wchodzącym jednak w skład magistrali SPI jest linia GDO0. W typowych aplikacjach zmiana stanu na tej linii może powiadamiać mikrokontroler o zakończeniu transmisji danych drogą radiową lub o rozpoczęciu odbioru danych. Wszystkie opisane wyżej linie są widoczne na **rys. 1**, linia GDO2 jest niewykorzystana.

Połączenie wyprowadzeń układu z mikrokontrolerem zasilanym tym samym napięciem nie przedstawia żadnych trudności. Jeżeli mikrokontroler ma wewnętrzne mechanizmy sprzętowe wspierające transmisję SPI, należy odpowiednie linie CC1100 połączyć z odpowiadającymi im liniami mikrokontrolera lub z dowolnymi portami, które powinny być odpowiednio skonfigurowane jako wejścia i wyjścia (sygnały wyjściowe modułu czyli SO i GDO0 muszą być dołączone do portów wejściowych mikrokontrolera, po-

Skróty, które pojawiają się w tekście

ISM - Industrial Scientific and Medical (układ do zastosowań przemysłowych, naukowych i medycznych)

SRD - Short Range Device (układ do łączności w bliskim zasięgu)

dBm - jednostka mocy odniesiona do 1 mW, mówi o ile moc w decybelach jest mniejsza lub większa od 1 mW. I tak 10 dBm=10 mW, a -10 dBm=0,1 mW.

BER - Bit Error Rate (stopa błędów transmisji, czyli inaczej jaki jest procent błędnie przesłanych bitów w transmisji).

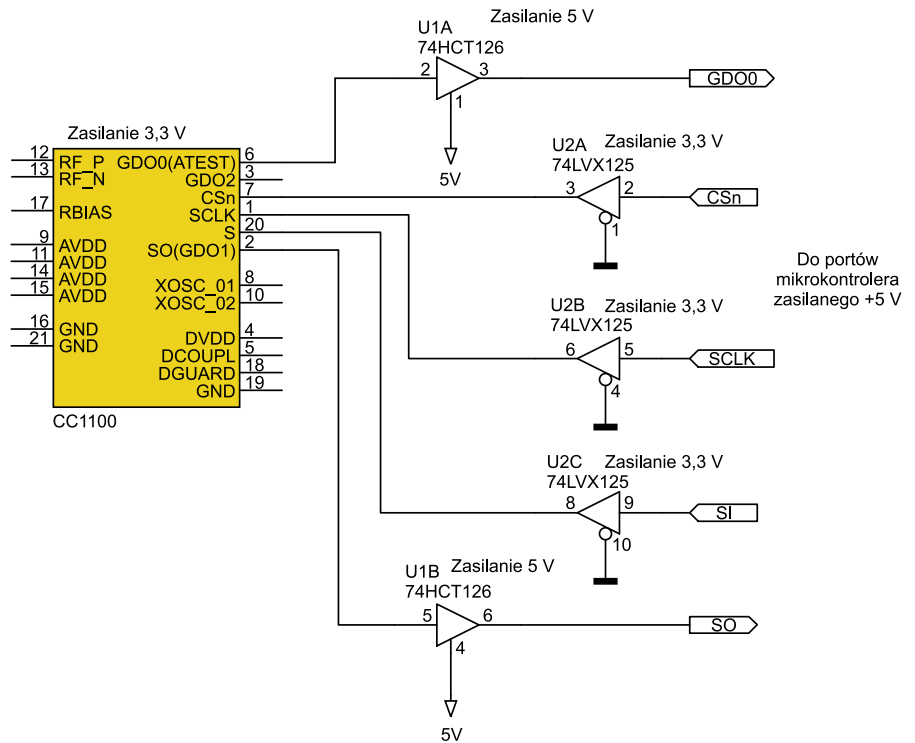
Krajowa Tablica Przeznaczeń Częstotliwości - akt prawny przydzielający do wykorzystania poszczególne częstotliwości radiowe.

zostałe do portów wyjściowych). Może okazać się korzystnym połączenie wyjścia GDO0 z wejściem przerwać mikrokontrolera.

Dla sygnałów wejściowych CC1100 najprostszym rozwiązaniem jest zastosowanie dzielników oporowych, tak aby poziom sygnałów wysyłanych przez mikrokontroler nie przekroczył wartości napięcia zasilającego CC1100. Z kolei sygnały wyjściowe powinny zostać wzmocnione do poziomu akceptowalnego przez porty mikrokontrolera. Można to zrobić przy pomocy tranzystorów, co jednak spowoduje odwrócenie fazy sygnałów. Dlatego być może lepszym rozwiązaniem będzie zastosowanie bramek odpowiednich serii układów MOS, które dopasują poziomy sygnałów. Jeżeli sygnały wyjściowe modułu o poziomie 3,3 V podłączymy do bramek np. 74HCT126 zasilanych napięciem +5 V, to ich poziom zostanie przekonwertowany na odpowiedni dla mikrokontrolera bez pogorszenia ich jakości. Z kolei jeżeli sygnały wejściowe będą podawane na CC1100 poprzez bramki 74LVX125 lub 74LVX125 zasilane takim samym napięciem jak transceiver, to także uzyskamy prawidłową konwersję sygnału z wyższego poziomu na niższy. Na rys. 2 pokazano taki właśnie układ konwersji poziomów.

Przykłady kodu oprogramowania

Zamieszczone w tekście przykłady kodu zostały napisane w języku C dla mikrokontrolera z rodziny AVR. Zdają sobie sprawę, że nie wszystkich czytelników może to zadowolić, ale przykłady są na tyle proste, że mogą być pomocne przy pisaniu oprogramowania



Rys. 2. Układ konwersji poziomów

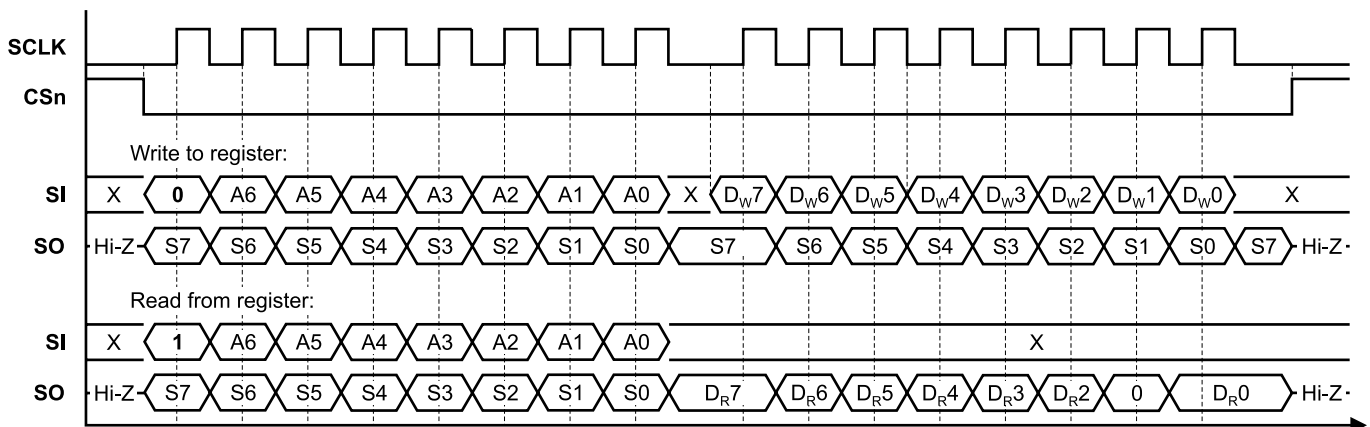
w BASIC-u czy nawet assemblerze. Wybrany typ mikrokontrolera narzucił konwencję składni jeśli chodzi o fragmenty związane z bezpośrednim dostępem do portów. Po niewielkich modyfikacjach kod ten również pracował w wersji dla mikrokontrolerów ARM.

Oprogramowanie transmisji poprzez magistralę SPI

W przypadku mikrokontrolerów wspierających transmisję magistralą SPI należy zwrócić do ich dokumentacji technicznej i postępując zgodnie z zawartymi tam informacjami uruchomić transmisję do układu CC1100. Jeżeli jednak mikrokontroler nie posiada takiego wsparcia lub nie można z niego skorzystać,

należy napisać własne procedury transmisji. Nie jest to trudne. Na początku należy się jednak zapoznać z samym formatem wymiany danych magistralą SPI. Przebiegi impulsów na poszczególnych liniach w czasie zapisu i odczytu pojedynczego bajtu danych pokazano na rys. 3.

Możliwość wymiany danych z układem CC1100 poprzez magistralę SPI następuje z chwilą ustawienia linii CSn w stanie niskim. W trakcie zapisu każde zbocze narastające zegara SCLK powoduje zapis kolejnego bitu podanego na wejście SI do CC1100. W trakcie odczytu każde zbocze narastające zegara SCLK powoduje odczyt kolejnego bitu z wyjścia SO układu CC1100.



Rys. 3. Przebiegi impulsów na poszczególnych liniach w czasie zapisu i odczytu pojedynczego bajtu danych

W trakcie zapisu jako pierwszy powinien być wysłany bit o wartości „0”, a po nim 7 bitów adresu rejestru (w kolejności od najstarszego bitu A6 do najmłodszego A0). Następnie wysyłanych jest 8 bitów bajtu danych, także z najstarszym bitem D7 jako pierwszym.

Podczas odczytu zawartości rejestru CC1100 jako pierwszy powinien być wysłany bit o wartości „1”, a po nim 7 bitów adresu rejestru (w kolejności od najstarszego bitu A6 do najmłodszego A0). Następnie z wyjścia SO można odczytać 8 bitów bajtu danych, przy czym bit D7 jest odczytywany jako pierwszy. Po zakończeniu transmisji należy zamknąć magistralę SPI podając stan wysoki na wyprowadzenie CSn.

Tryby zapisu i odczytu rejestrów CC1100

Wewnętrzne rejestry CC1100 odgrywają kluczowe znaczenie w sposobie jego funkcjonowania. Zrozumienie kilku niezbyt skomplikowanych zasad określających

R E K L A M A

Rozwiązania przemysłowego Ethernetu do stacji energetycznych, systemów kontroli ruchu oraz kolei

MOXA PT-7728

redundantny, modułowy, gigabitowy switch przemysłowy

- 24x Fast Ethernet, 4x 1Gb/s, skrętka/światłowód/SFP(mini-GBIC)
- IEC 61850-3, IEEE1613 (stacje energetyczne), NEMA TS2 (systemy kontroli ruchu), EN50121-4 (kolej)
- Protokoły redundantnych sieci Ethernet: MOXA Turbo Ring oraz RSTP/STP
- Uniwersalny zakres napięcia zasilania: 24, 48 VDC lub 110/220 VDC/VAC
- Modułowa obudowa umożliwiająca elastyczny wybór ilości oraz rodzaju portów sieciowych
- Zakres temperatury pracy: -40°C do +85°C

28 portów
-40°C do +85°C



ELMARK
Automatyka

Elmark Automatyka Sp. z o.o.
ul. Radna 12, 00-341 Warszawa
tel. 022 821 30 54
fax. 022 821 30 55

www.elmark.com.pl

MOXA

www.moxa.com.pl

Telic
Polska Sp. z o.o.

Terminale GSM

Urządzenia Telemetryczne

Czterozakresowe
Plug&Play
Port Szerokowy RS232
Programowalne
Interfejs USB/AUDIO
Niezawodne
Szeroki zakres zasilania
SIM 1,8 i 3V
Temp. pracy -30°C...+75°C
Embedded!!!



GT864-PY Terminal
GT864-QUAD Terminal

GT64Terminal
GS64Terminal

Sterowanie przez telefon GSM
Plug&Play
Prosta instalacja
Uruchamianie zdalne aplikacji
Sterowanie 2x6A na wyj.
2x wyj./wej.



STD32

Otwieranie bram wjazdowych i garażowych
Alarmy domowe i miejscach oddalonych
Stacje pogodowe i meteorologiczne
Instalacje domowe



STD35

Sterowanie przez telefon oraz RS232
Uruchamianie zdalne aplikacji
Sterowanie 3x6A na wyj.
Plug&Play
Prosta instalacja
5x wyj./wej.



SYSTEMY
PARKINGOWE



BRAMY
WJAZDOWE



SYSTEMY
PRZEMYSŁOWE



WĘZŁY GAZOWE I
CIEPŁOWNICZE



ENERGETYKA



AUTOMATY DO
GIER I NAPOJÓW

Telic
Polska Sp. z o.o.

Telic Polska Sp. z o.o.
ul. Świeradowska 51-57/103
50-559 Wrocław

Tel.: +48 71 7812788
Fax: +48 71 7812789
E-mail: poland@telic.pl

www.Telic.pl

sposób ich zapisu i odczytu pozwoli kontrolować działanie całego układu.

Większość rejestrów można zarówno odczytywać, jak i zapisywać. Jak to pokazano na rys. 3, o tym czy mamy do czynienia z zapisem, czy z odczytem decyduje najstarszy 7 bit adresu, który jest wyzerowany przy zapisie i ustawiony przy odczycie. Z kolei adres rejestru, do którego chcemy mieć dostęp ustawiany jest bitami A5... A0. Przyporządkowanie adresów rejestrów wraz z opisem ich wpływu na sposób działania układu CC1100 można znaleźć w tabeli 37 „SPI Address Space” danych technicznych.

Sprawę komplikuje fakt, że zapis i odczyt rejestrów można zrealizować dwoma sposobami. Pierwszy jest oczywisty i polega na podaniu najpierw adresu rejestru, do którego chcemy mieć dostęp, a następnie na odczytaniu lub zapisaniu bajtu danych do rejestru. Ten tryb zapisu/odczytu nosi w dokumentacji nazwę „single byte”.

Drugi tryb zapisu/odczytu danych jest trybem grupowym i nosi nazwę „burst”. Jak to sugeruje sama nazwa oznacza on hurtowy zapis lub odczyt rejestrów. Po podaniu adresu pierwszego rejestru z grupy następuje cykliczny zapis lub odczyt kolejnych rejestrów bez konieczności podawania ich kolejnych adresów.

O tym czy zapis/odczyt jest typu „single byte”, czy „burst” decyduje 6. bit adresu. Jeżeli bit ten jest wyzerowany, mamy do czynienia z dostępem typu „single byte”. Jeżeli bit A6 jest ustawiony, chodzi o dostęp grupowy, czyli „burst”.

Zastosowanie konkretnego trybu dostępu do rejestru ma znaczenie, o czym można się przekonać oglądając tabelę 37 „SPI Address Space”. Wynika z niej, że rejestry o adresach od 00h do 2Eh (litera „h” oznacza zapis liczby w kodzie hexadecymalnym) można zapisywać i odczytywać w obydwu trybach.

List. 1. Procedura SpiReadReg

```
//odczyt zawartości pojedynczego rejestru
//-----
unsigned char SpiReadReg(unsigned char adres)
//we: adres -adres rejestru
{
    unsigned char x, maska, wartosc;

    /*01*/ SCLK_RF_PORT =0;
    /*02*/ CS_RF_PORT =0;
    /*03*/ while (SO_RF_PIN !=0); //oczekiwanie na stan niski na wyjściu SO
    /*04*/ adres =adres & 0b00111111; //wyzerowany b.7 i b.6
    /*05*/ adres =adres | 0b10000000; //b.7=1, b.6=0 - odczyt w trybie „single byte”
    /*06*/ maska =0b10000000;
    //wysłanie adresu rejestru
    /*07*/ for (x=0; x<8; x++)
    {
        /*08*/ if ((adres &maska) ==0) SI_RF_PORT =0;
        /*09*/ else SI_RF_PORT =1;
        /*10*/ #asm(„nop”);
        /*11*/ SCLK_RF_PORT =1;
        /*12*/ #asm(„nop”);
        /*13*/ SCLK_RF_PORT =0;
        /*14*/ maska =maska >>1;
    }
    /*15*/ wartosc =0;
    /*16*/ maska =0b10000000;
    //odczyt zawartości rejestru
    /*17*/ for (x=0; x<8; x++)
    {
        /*18*/ SCLK_RF_PORT =1;
        /*19*/ („nop”);
        /*20*/ if (SO_RF_PIN ==1) wartosc =wartosc |maska;
        /*21*/ #asm(„nop”);
        /*22*/ SCLK_RF_PORT =0;
        /*23*/ maska =maska >>1;
    }
    /*24*/ CS_RF_PORT =1;
    /*25*/ return wartosc;
}
```

List. 2. Procedura SpiWriteReg

```
//zapis wartości do pojedynczego rejestru
//-----
void SpiWriteReg(unsigned char adres, unsigned char wartosc)
//we: adres - adres rejestru, wartosc - wpisywana do rejestru wartość
{
    unsigned char x, maska;

    SCLK_RF_PORT =0;
    CS_RF_PORT =0;
    while (SO_RF_PIN !=0); //oczekiwanie na stan niski na wyjściu SO
    adres =adres & 0b00111111; //wyzerowany b.7 i b.6
    adres =adres | 0b00000000; //b.7 i b.6 =0 zapis do rejestru w trybie „single byte”
    maska =0b10000000;
    //wysłanie adresu rejestru
    for (x=0; x<8; x++)
    {
        if ((adres &maska) ==0) SI_RF_PORT =0;
        else SI_RF_PORT =1;
        #asm(„nop”);
        SCLK_RF_PORT =1;
        #asm(„nop”);
        SCLK_RF_PORT =0;
        maska =maska >>1;
    }
    //wysłanie kolejnych bitów bajtu zapisywanego do rejestru
    maska =0b10000000;
    for (x=0; x<8; x++)
    {
        if ((wartosc &maska) ==0) SI_RF_PORT =0;
        else SI_RF_PORT =1;
        #asm(„nop”);
        SCLK_RF_PORT =1;
        #asm(„nop”);
        SCLK_RF_PORT =0;
        maska =maska >>1;
    }
    CS_RF_PORT =1;
}
```

Sprawa komplikuje się dla rejestrów o adresach od 30h do 3Dh. W zależności od trybu dostępu pod tym samym adresem uzyskujemy dostęp do rejestrów statusu, do których zapis jest niemożliwy lub do rejestrów komend. Rejestry komend wpływają na zasadnicze funkcje transceivera, np. przełącza-

ją go w tryb odbioru lub nadawania. Do rejestrów komend w trybie zapisu nie ma potrzeby przesyłania żadnych danych, wystarczy podanie ich adresu, by uaktywnić funkcję za którą odpowiadają.

Ryszard Szymaniak, EP
ryszard.szymaniak@ep.com.pl