

Łączy radiowe w paśmie 2,4 GHz

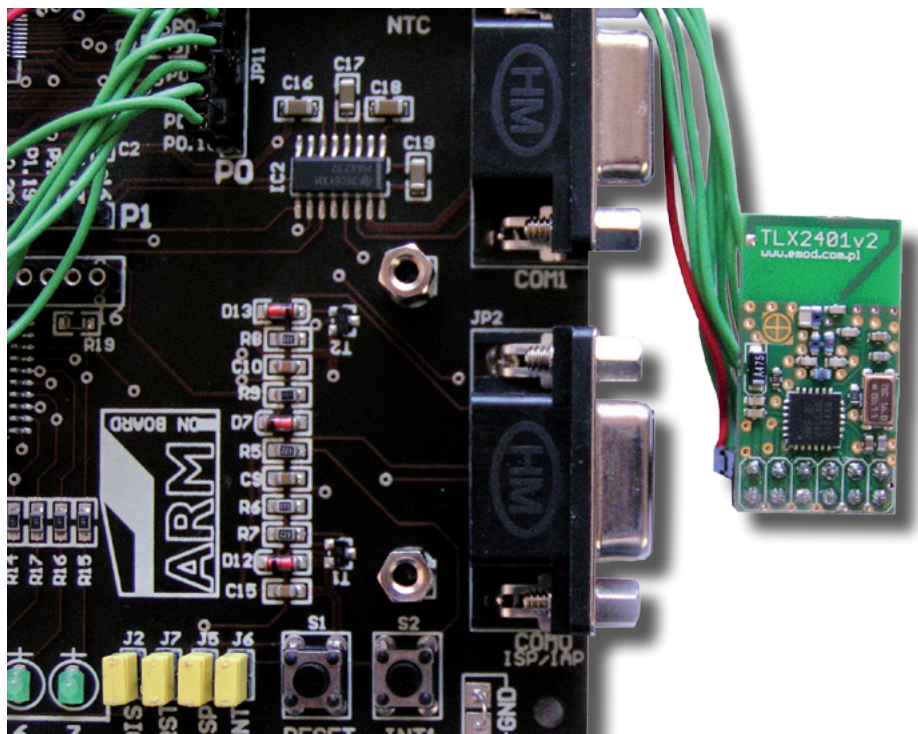
Do niedawna wykorzystywanie łączy radiowych było możliwe po uzyskaniu niezbędnych zezwoleń na używanie nadajników radiowych. Wydzielenie pasm nielicencjonowanych i ograniczenie mocy promieniowania nadajników umożliwiło swobodne korzystanie z połączenia radiowego. Dlatego jest to jeden z segmentów elektroniki, który przeżywa bardzo burzliwy rozwój.

Połączenie bezprzewodowe jest nie tylko „wygodne” w użyciu, ale stało się też „modne”. Dzisiaj wielu internautów nie wyobraża sobie połączenia komputera z Internetem inaczej, niż poprzez bezprzewodowy interfejs WiFi, a połączenia telefonu komórkowego z zestawem słuchawkowym, zewnętrznym odbiornikiem GPS, komputerem lub innym telefonem, za pomocą interfejsu Bluetooth.

W nowoczesnych łączach radiowych jest stosowane pasmo wielkiej częstotliwości (kilka GHz), a nadajniki wypromieniowują bardzo małą moc. Oprócz typowo radiowych parametrów, standard definiuje szereg protokołów sieciowych, ujętych w model warstwowy. Zapewnia to dużą elastyczność przy łączeniu różnych urządzeń, ale wymaga od konstruktora użycia albo specjalizowanych modułów ze sprzętową realizacją niezbędnych protokołów, albo zastosowania mikrokontrolerów o odpowiednio dużych zasobach, aby można było zaimplementować te protokoły programowo.

W praktyce zazwyczaj potrzebujemy łączy radiowego dla konkretnej aplikacji niewymagającej użycia wszystkich protokołów. Nie jest więc istotna, zapewniona przez standard transmisji, kompatybilność z innymi urządzeniami. Inaczej mówiąc – potrzebujemy prostego, pewnego, taniego rozwiązania, zapewniającego łączy punkt-punkt. W takich sytuacjach warto rozważyć zastosowanie modułu radiowego TLX2401 pracującego w paśmie 2,4 GHz. Jego budowę oparto o układ nRF2401 produkcji Nordic Semiconductor.

Przy przesyłaniu danych łączy radiowym należy zastosować modulację analogową.

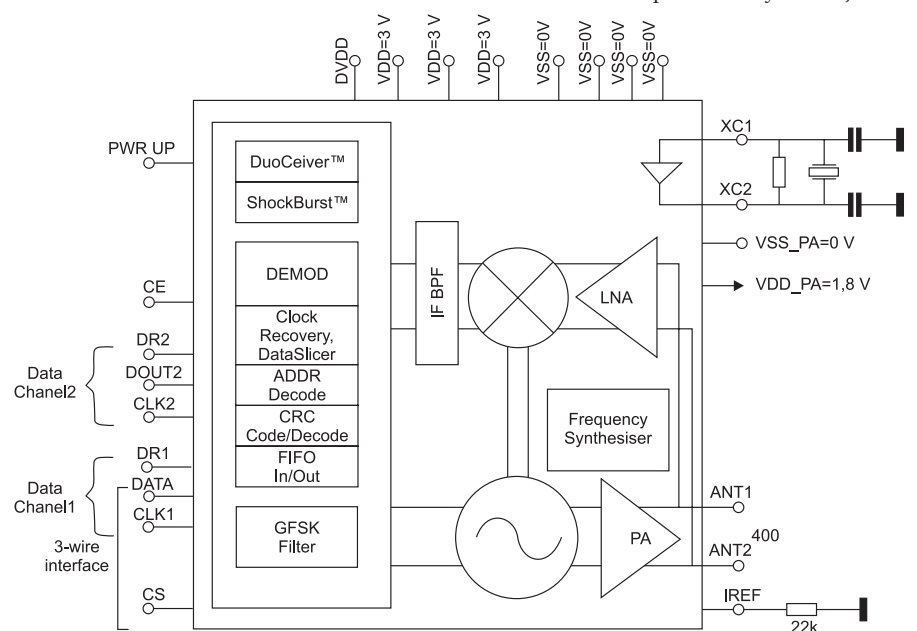


gowego sygnału nośnej. Od rodzaju i parametrów tej modulacji oraz szerokości kanału zależy szybkość transmisji. W układzie nRF2401 zastosowano modulację GFSK. Dzięki temu uzyskano stosunkowo dużą prędkość transmisji – do 1 Mbit/s.

Całe dostępne pasmo radiowe podzielono na 125 kanałów. Tak duża liczba kana-

łów pozwala na bezproblemową pracę wielu urządzeń zlokalizowanych blisko siebie. Zmiana kanału następuje w czasie krótszym niż 200 μs.

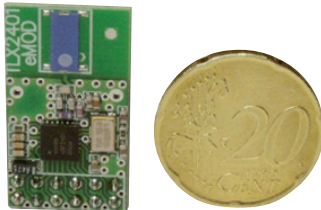
Oprócz kompletnej części radiowej, wymagającej tylko kilku zewnętrznych elementów, układ wyposażono w bloki logiczne sterowania oraz 3-przewodowy interfejs sze-



Rysunek 1. Schemat blokowy układu nRF2401 z elementami zewnętrznymi

Tabela 1. Podstawowe parametry układu nRF2401

Parametr	Wartość	Jednostka
Minimalne napięcie zasilania	1,90	V
Maksymalna moc wyjściowa	0	dBm
Maksymalna prędkość przesyłu danych	1000	kbps
Pobór prądu		
– nadawanie z $P_{wy} = -5$ dBm	10,5	mA
– odbiór	18,0	mA
Zakres temperatury	-40 do +85	°C
Czułość	-90	dBm
Pobór prądu w trybie Power Down	400	nA



Fotografia 2. Moduł TLX2401

regowy. Bloki sterowania mają wbudowane mechanizmy detekcji adresu i generowania wartości wielomianu kontrolnego CRC. Poza tym, można programować dwa tryby transmisji danych: bezpośredni i *ShockBurst*. Jest też możliwość odbierania jednocześnie danych z dwóch kanałów radiowych (oddalonych od siebie o min. 8 MHz). Firma nazwała to rozwiązanie *DuoCeiver*. W tym celu układ wyposażono w dwa niezależne interfejsy szeregowo (rysunek 1).

Układ przeznaczony do nowoczesnych urządzeń mikroprocesorowych powinien być zasilany niskim napięciem i pobierać tak mało prądu, jak to tylko możliwe. Napięcie zasilania nRF2401 jest z przedziału 1,9...3,6 V. Wyraźnie sugeruje to przeznaczenie do zasilania z baterii. Jest korzystne,

ponieważ nRF2401 może być zasilany typowym napięciem +3,3 V. Pobór prądu zależy od tego czy układ nadaje (ok. 10,5 mA), czy odbiera (ok. 18 mA). Niski pobór prądu i niskie napięcie zasilania sprawiają, że układ może być z powodzeniem zasilany z baterii. W tabeli 1 zamieszczono najważniejsze parametry układu nRF2401.

Chociaż układ nRF2401 nie wymaga wielu elementów zewnętrznych, to ze względu na bardzo duże częstotliwości jego użycie wymaga sporego doświadczenia. Producent w nocie katalogowej umieścił przykłady aplikacji łącznie z projektami PCB. Jednak dla kogoś nawet te informacje mogą okazać się niewystarczające. W takiej sytuacji najlepiej jest skorzystać z gotowego minimodułu TLX2401 produkowanego przez polską firmę eMOD.

Moduł ma wymiary 29×16,5 mm (fotografia 2). Na płytce jest umieszczony scalony generator 16 MHz, antena nadawczo/odbiorcza i pozostałe elementy niezbędne do prawidłowej pracy. Linie zasilania i wszystkie sygnałowe niezbędne do połączenia modułu z mikrokontrolerem zostały wyprowadzone na dwurzędowe złącze goldpin (rysunek 3).

Tryb ShockBurst

W trybie *ShockBurst* (rysunek 4) dane są gromadzone w buforze o pojemności 256 bitów. Wielkość bufora musi wystarczyć na przetrzymanie pół: adresu odbiorcy, danych i sumy kontrolnej. Za odpowiednią wielkość pakietu odpowiada mikrokontroler nadrzędny. Suma kontrolna CRC ma długość 8 lub 16 bitów (do wyboru) i może być wyliczana przez nRF2401. Funkcję tę można również

wyłączyć zerując bit CRC_EN w rejestrze konfiguracyjnym. Następnie po wyzerowaniu poziomu na wejściu CE, logika sterująca nadajnikiem dodaje do pakietu preambułę, włącza nadajnik i wysyła pakiet danych z prędkością 1 Mbps lub 250 kbps (do wyboru). W porównaniu z tradycyjną metodą transmisji ciągłej, powoduje to znaczną redukcję poboru energii. Na rysunku 5 pokazano schemat czynnościowy procesu wysyłania danych.

Dane te mogą być prawidłowo odebrane przez inny układ nRF2401 również pracujący w trybie *ShockBurst*. Odbiornik jest aktywowany przez ustawienie sygnału CE. Po czasie 200 µk od ustawienia CE zaczyna on nasłuchiwać na zaprogramowanym kanale. W przychodzących danych są sprawdzane adres, obecność i wartość CRC. Jeżeli odebrane dane są prawidłowe, czyli pakiet jest opatrzony prawidłowym adresem i ma prawidłową wartość CRC, to są z niego usuwane dane nadmiarowe, aby do odbiorcy dotarło tylko pole danych (*PAYLOAD*). Gotowość danych do odbioru jest sygnalizowana poziomem wysokim na wyprowadzeniu DR1.

Mikrokontroler może odczytać dane z wewnętrznego bufora FIFO z dowolną prędkością. W czasie odczytu, jeżeli nie spodziewamy się odbioru kolejnych danych, można wyłączyć układ odbiornika przez wyzerowanie poziomu CE. Po odczytaniu wszystkich danych DR1 zmienia poziom na niski. Schemat blokowy poszczególnych faz odbioru danych w trybie *ShockBurst* pokazano na rysunku 6.

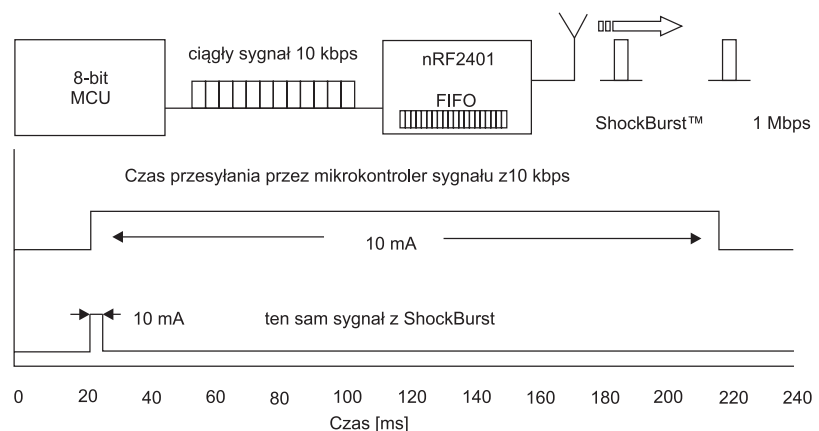
Tryb bezpośredni

W trybie bezpośrednim nRF2401 pracuje jak klasyczny układ nadawczo-odbiorczy. Dane mogą być przesyłane z maksymalną prędkością 1 Mbps ±200 ppm lub 250 kbps ±200 ppm, zależną od wprowadzonych nastaw. Prędkość transmisji odpowiada prędkości przesyłania interfejsu szeregowo. W tym trybie nadawanie rozpoczyna się po ustawieniu linii CE. Po upływie 200 µs, mikrokontroler może rozpocząć transmisję do nRF2401. Uwaga: nadajnik może prze-

GND	⊙ ⊙ ⊙ +3V	⊙ GND
PWR_UP	⊙ ⊙ CE	
DR2	⊙ ⊙ CLK2	
DOUT2	⊙ ⊙ CS	
DR1	⊙ ⊙ CLK1	
DATA	⊙ ⊙ GND	⊙ GND

Opis	Funkcja
GND +3V	Masa zasilania Zasilanie +1,9...+3,5 V
PWR_UP	Tryb uśpienia (0-uśpienie, 1-praca)
CE	Chip Enable, włączenie trybu Rx lub Tx
DR2	Dane z kanału gotowe do odczytu (tylko w trybie ShockBurst)
CLK2	Wejście/wyjście zegara dla kanału 2
DOUT2	Dane wyjściowe z kanału 2
CS	Chip Select 1- włączenie trybu konfigurowania układu
DR1	Dane z kanału 1 gotowe do odczytu (tylko w trybie ShockBurst)
CLK1	Wejście/wyjście zegara dla kanału 1 (tylko w trybie ShockBurst)
DATA	Dane wyjściowe z kanału 1/dane wejściowe nadajnika

Rysunek 3. Wyprowadzenia modułu TLX2401



Rysunek 4. Tryb ShockBurst

śląc pakiet o maksymalnym czasie trwania wynoszącym 4 ms. W tym trybie nie ma wsparcia kontroli adresu, CRC i dodawania preambuły. Wszystkie te elementy powinny być zaimplementowane w oprogramowaniu sterującym.

Większy problem sprawia odbiór. Po ustawieniu linii CE, jeżeli odbiornik odbiera prawidłowo zmodulowaną nośną, to na jego wyjściu danych pojawia się sygnał. Program mikrokontrolera odbiorcy musi wyszukać z tych danych preambułę. Dopiero po odebraniu preambuły, demodulator jest w stanie odtwarzać z sygnału dane i sygnał zegara.

Jedynym sensownym zastosowaniem trybu bezpośredniego jest konieczność przesyłania długich pakietów (do 4000 bitów) z prędkością 1 Mbps lub innych z niewielką prędkością, na przykład z powodu dużych zakłóceń w paśmie 2,4 GHz.

Strukturę typowego pakietu transmitowanego przez nRF2401 pokazano na **rysunku 6**.

Konfigurowanie układu

Wszystkie funkcje nRF2401 są konfigurowane przez nastawy rejestru kontrolnego. Rejestr ma długość 15 bajtów dla trybu ShockBurst i 2 bajtów dla trybu bezpośredniego.

Słowo konfiguracyjne trybu ShockBurst jest dzielone na sekcje:

- długość pola danych. Określa liczbę bitów w polu danych użytkownika (PAYLOAD). Jeżeli jest zdefiniowane obliczanie CRC, to rozmiar pola danych razem z bitami CRC i długością pola adresu nie może przekroczyć 256 bitów (rysunek 6),
- długość adresu odbiorcy,
- adres odbiorcy,
- blokowanie/odblokowanie obliczania CRC przez układ nRF2401.

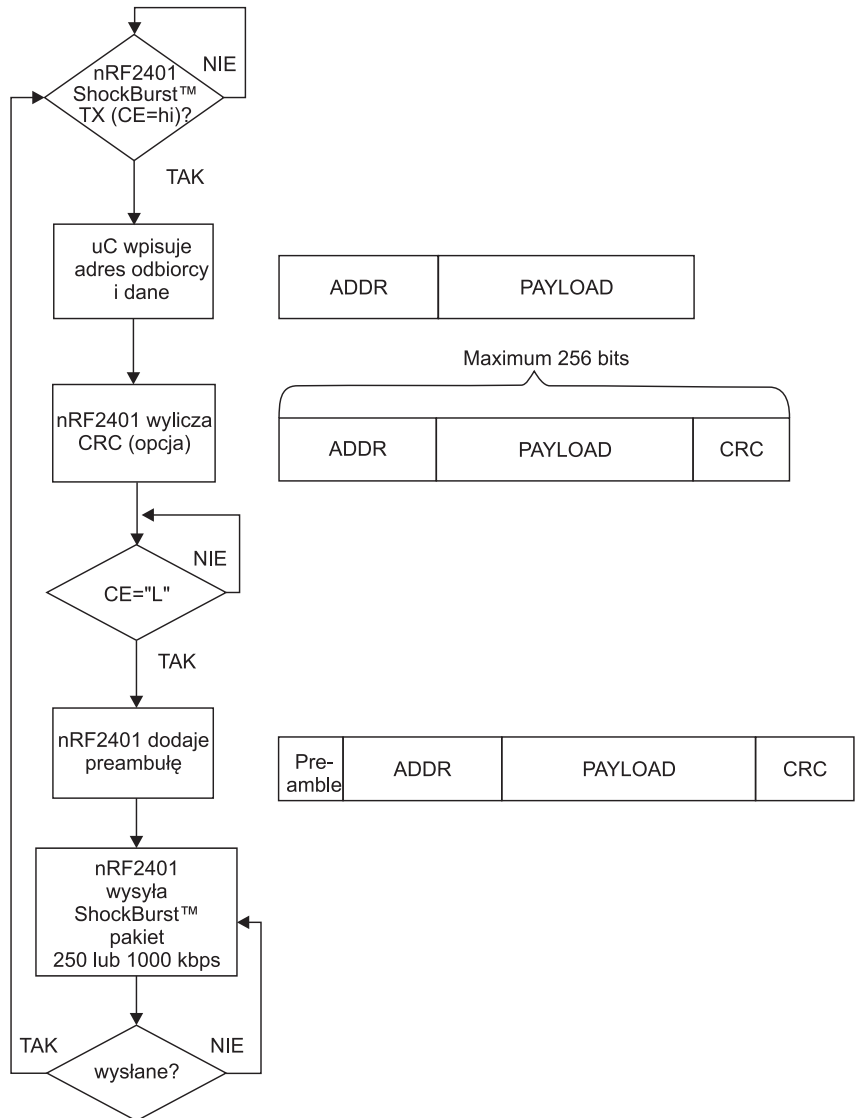
W trybie bezpośrednim są używane 2 pierwsze bajty słowa konfiguracyjnego.

W **tabeli 2** zamieszczono opis nastaw bitów słowa konfiguracyjnego dla obu trybów.

Przykłady wykorzystania TLX2401

Do praktycznych prób łącza radiowego w paśmie 2,4 GHz wykorzystano dwa moduły TLX2401 sterowane przez mikrokontrolery z płytek ewaluacyjnych ZL1ARM z mikrokontrolerami LPC2114 (ARM7). Jeden z zestawów nazwałem umownie „nadajnikiem”, a drugi „odbiornikiem”.

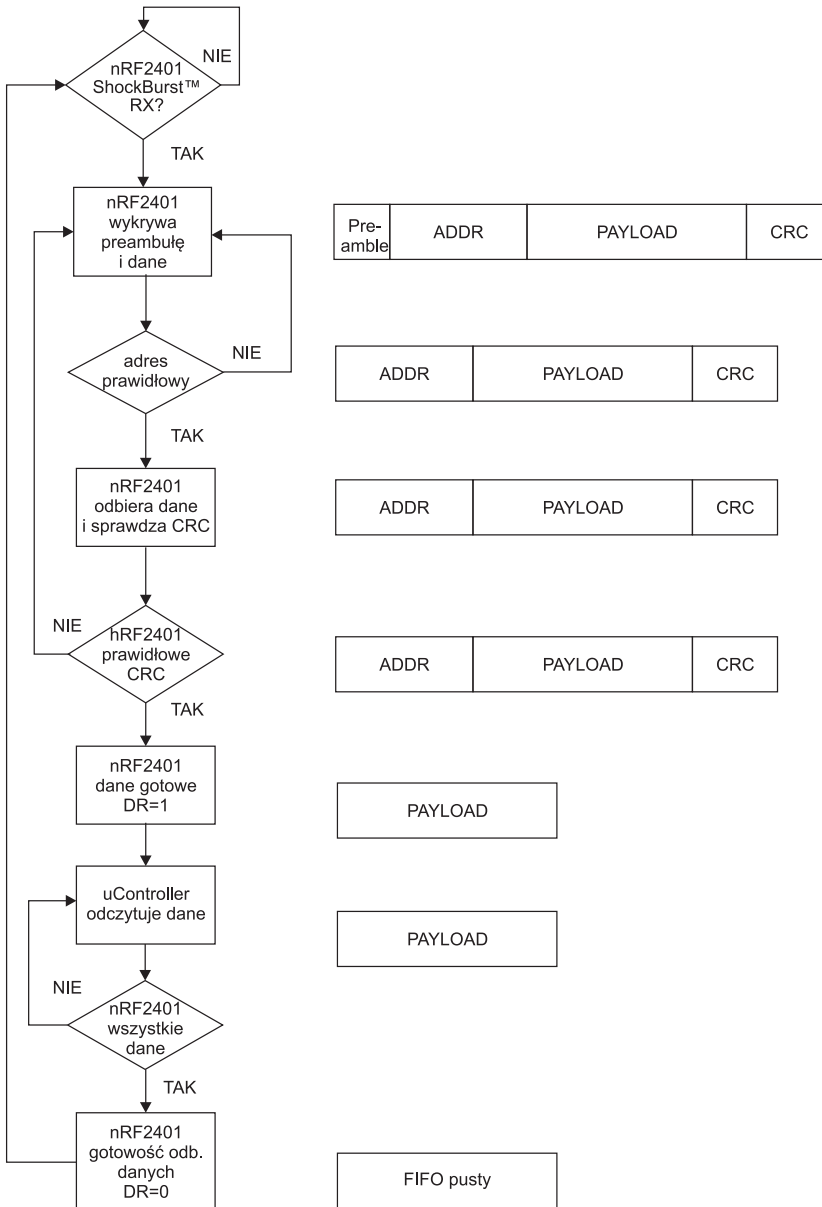
Na **listingu 1** pokazano podprogram wysyłania danych, łącznie z makrami manipulacji poziomami linii: danych i zegarowej. Na **listingu 2** zamieszczono funkcję odbioru danych. Ponieważ dla kanału pierwszego linia danych jest dwukierunkowa, to przed wywołaniem funkcji nadawania lub odbioru, należy określić kierunek transmisji przez linię portu. Funkcję odpowiedzialną za to zawiera **listing 3**.



Rysunek 5. Wysyłanie danych ShockBurst

Tabela 2. Nastawy rejestru konfiguracyjnego układu nRF2401

Tryb	Pozycja	Liczba bitów	Nazwa	funkcja
Tylko Shock-Burst	143:120	24	TEST	Zarezerwowane
	119:112	8	DATA2_W	Długość pola danych (PAYLOAD) odbiornika 2
	111:104	8	DATA1_W	Długość pola danych (PAYLOAD) odbiornika 1
	103:64	40	ADDR2	Adres odbiorcy kanału 2 (maks. 5 bajtów)
	63:24	40	ADDR1	Adres odbiorcy kanału 1 (maks. 5 bajtów)
	23:18	6	ADDR_W	Liczba bitów adresu dla obu kanałów
	17	1	CRC_L	CRC 8 lub 16 bitów (0=8 bitów, 1=16 bitów)
Shock-Burst i tryb bezpośredni	16	1	CRC_EN	Zezwolenie na obliczanie CRC (0=obliczanie CRC zabronione, 1=obliczanie CRC dozwolone)
	15	1	RX2_EN	Włączanie trybu DuoCeiver (1=włączony, 0=wyłączony)
	14	1	CM	Wybór ShockBurst/tryb bezpośredni (0=tryb bezpośredni, 1=ShockBurst)
	13	1	RFDR_SB	Maksymalna prędkość transmisji (0=256 kbps, czułość odbiornika lepsza o 10 dB, 1=1 Mbps, wymaga XO_F=16 MHz)
	12:10	3	XO_F	Nastawa częstotliwości kwarcu podłączonego do nRF2401 (000=4 MHz, 001=8 MHz, 010=12 MHz, 011=16 MHz, 100=20 MHz)
	9:8	2	RF_PWR	Ustawienie mocy wyjściowej (00= -20 dBm, 01= -10 dBm, 10= -5 dBm, 11=0 dBm)
	7:1	7	RF_CH	Określanie numeru kanału
	0	1	RX_EN	Wybór nadawanie (RX_EN=0)/odbior (RX_EN=1)



Rysunek 6. Odbiór danych w trybie ShockBurst

Preambuła	Adres	Dane	CRC
Preambuła (pre-ambule)	Pole preambuły jest wymagane w obu trybach transmisji. W ShockBurst dodawane automatycznie w nadajniku i usuwane w odbiorniku. W trybie bezpośrednim preambuła zależy od wartości pierwszego bitu pola danych Preambuła: = 01010101 dla pierwszego bitu 1 = 10101010 dla pierwszego bitu 0 W tym trybie preambuła jest widoczna w polu danych.		
Adres (address)	Adres (numer) odbiorcy. W trybie ShockBurst ma programowaną długość: od 8 do 40 bitów. W trybie bezpośrednim adres odbiorcy powinien być generowany przez mikrokontroler.		
Dane (payload)	Pole danych użytkownika W trybie ShockBurst całkowita długość pola danych wynosi 256 bitów minus długość adresu (8 do 40 bitów) i opcjonalnej sumy CRC (8 lub 16 bitów) W trybie bezpośrednim długość pola danych zależy od prędkości transmisji, bo nadajnik może pracować przez 4 ms. Dla 1 Mbps można przelać 4000 bitów		
CRC	Opcjonalne pole sumy kontrolnej w trybie ShockBurst, nieużywane w trybie bezpośrednim		

Rysunek 7. Struktura typowego pakietu przesyłanego przez nRF2401

Przed wysłaniem i odbiorem danych trzeba do układu nRF2401 wysłać słowo konfiguracyjne. W testach używano trybu ShockBurst z prędkością 250 kbps i dla porównania – 1 Mbps. W przygotowanym programie słowo konfiguracyjne jest zapisywane w tablicy INIT_nRF[] (listing 4).

Bufory odbiornika mają długość 6 bajtów. Razem z 5 bajtami pola adresu daje to 88 bitów odbieranych w trybie ShockBurst. Ponieważ będziemy używali tylko jednego kanału odbiorczego (DuoCeiver jest wyłączony), to adres kanału drugiego nie ma znaczenia. W polu adresu kanału pierwszego wpisujemy 5-bajtowy adres, który musi być

Listing 1. Funkcja wysyłająca bajt przez SPI do nRF2401

```
//makra manipulacji stanami linii interfejsu SPI
#define DATA_1 {IOSET0|=DATA; delay1();}
#define DATA_0 {IOCLR0|=DATA; delay1();}

#define CLK1_1 {IOSET0|=CLK1; delay1();}
#define CLK1_0 {IOCLR0|=CLK1; delay1();}
//procedura wysyłania danych
void SendSPI(unsigned char data){
    char i;
    CLK1_0
    for(i=0;i<8;i++)
    {
        if(data&0x80) DATA_1 else DATA_0
        CLK1_1
        data<<=1;
        CLK1_0
    }
}
```

Listing 2. Funkcja odbierająca bajt przez SPI z nRF2401

```
unsigned char RecSPI(void){
    char i;
    unsigned char data=0;
    CLK1_0
    for(i=0;i<8;i++)
    {
        data=data<<1;
        CLK1_1 CLK1_1
        if(test_DATA1()==1) data|=1;
        CLK1_0 CLK1_0
    }
    return(data);
}
```

Listing 3. Ustalanie kierunku linii danych mikrokontrolera

```
void DATA_in(void)
{IODIRO = 0x0FDE0000;}//linia danych wejściowa

void DATA_out(void)
{IODIRO = 0x0FDF0000;}//linia danych wyjściowa
```

Listing 4. Bufor ze słowem konfiguracyjnym

```
const unsigned char INIT_nRF[]={
48, //rozmiar bufora odbiornika kanału 2
48, //rozmiar bufora odbiornika kanału 1
0x02, //adres kanału 2
0xaa, 0xaa, 0x55, 0x55,
0xa2, //adres kanału 1
0x23, 0x88, 0x17, 0x6c,
0xA0, //40 bitów pola adresu, 8 bit CRC, bez CRC
0x4F, //moc 00dBm, Fxtal 16MHz, jeden kanał odbiornika,
tryb Shock Burst, 250kBps
0x07, //kanał 3, tryb RX
};
```


Listing 5. Funkcja obliczająca CRC

```
const unsigned int mtab [] =
{0x0000,0x1c1c,0x81c1,0x4001,0x01c3,0xc003,0x8002,0x41c2,
0x01c6,0xc006,0x8007,0x41c7,0x0005,0xc1c5,0x81c4,0x4004,
0x01cc,0xc00c,0x800d,0x41cd,0x000f,0xc1cf,0x81ce,0x400e,
0x000a,0xc1ca,0x81cb,0x400b,0x01c9,0xc009,0x8008,0x41c8,
0x01d8,0xc018,0x8019,0x41d9,0x001b,0xc1db,0x81da,0x401a,
0x001e,0xc1de,0x81df,0x401f,0x01dd,0xc01d,0x801c,0x41dc,
0x0014,0xc1d4,0x81d5,0x4015,0x01d7,0xc017,0x8016,0x41d6,
0x01d2,0xc012,0x8013,0x41d3,0x0011,0xc1d1,0x81d0,0x4010,
0x01f0,0xc030,0x8031,0x41f1,0x0033,0xc1f3,0x81f2,0x4032,
0x0036,0xc1f6,0x81f7,0x4037,0x01f5,0xc035,0x8034,0x41f4,
0x003c,0xc1fc,0x81fd,0x403d,0x01ff,0xc03f,0x803e,0x41fe,
0x01fa,0xc03a,0x803b,0x41fb,0x0039,0xc1f9,0x81f8,0x4038,
0x0028,0xc1e8,0x81e9,0x4029,0x01eb,0xc02b,0x802a,0x41ea,
0x01ee,0xc02e,0x802f,0x41ef,0x002d,0xc1ed,0x81ec,0x402c,
0x01e4,0xc024,0x8025,0x41e5,0x0027,0xc1e7,0x81e6,0x4026,
0x0022,0xc1e2,0x81e3,0x4023,0x01e1,0xc021,0x8020,0x41e0,
0x01a0,0xc060,0x8061,0x41a1,0x0063,0xc1a3,0x81a2,0x4062,
0x0066,0xc1a6,0x81a7,0x4067,0x01a5,0xc065,0x8064,0x41a4,
0x006c,0xc1ac,0x81ad,0x406d,0x01af,0xc06f,0x806e,0x41ae,
0x01aa,0xc06a,0x806b,0x41ab,0x0069,0xc1a9,0x81a8,0x4068,
0x0078,0xc1b8,0x81b9,0x4079,0x01bb,0xc07b,0x807a,0x41ba,
0x01be,0xc07e,0x807f,0x41bf,0x007d,0xc1bd,0x81bc,0x407c,
0x01b4,0xc074,0x8075,0x41b5,0x0077,0xc1b7,0x81b6,0x4076,
0x0072,0xc1b2,0x81b3,0x4073,0x01b1,0xc071,0x8070,0x41b0,
0x0050,0xc190,0x8191,0x4051,0x0193,0xc053,0x8052,0x4192,
0x0196,0xc056,0x8057,0x4197,0x0055,0xc195,0x8194,0x4054,
0x019c,0xc05c,0x805d,0x419d,0x005f,0xc19f,0x819e,0x405e,
0x005a,0xc19a,0x819b,0x405b,0x0199,0xc059,0x8058,0x4198,
0x0188,0xc048,0x8049,0x4189,0x004b,0xc18b,0x818a,0x404a,
0x004e,0xc18e,0x818f,0x404f,0x018d,0xc04d,0x804c,0x418c,
0x0044,0xc184,0x8185,0x4045,0x0187,0xc047,0x8046,0x4186,
0x0182,0xc042,0x8043,0x4183,0x0041,0xc181,0x8180,0x4040 };

unsigned short int CRC(unsigned char lbufor,unsigned char *bufor) //wylczenie
fcs
{
    unsigned short int pomfcs;
    unsigned short int fcs;
    unsigned char ifcs,zfcs;
    fcs=0xffff;
    for (ifcs=0;ifcs<lbufor;ifcs++) {
        zfcs=(bufor+ifcs);
        pomfcs=(mtab+((fcs>>8)^zfcs));
        fcs=((pomfcs&0xff00)^(fcs<<8)|(pomfcs&0x00ff));
    }
    fcs=(fcs^0xffff);
    return (fcs);
}
```

przesyłany w odebranych pakiecie w polu ADDR. Długość tego pola jest w następnym bajcie ustawiana na wartość na 40 bitów. Funkcja obliczenia CRC jest wyłączona.

Tu należy się wyjaśnienie. W tego typu transmisjach CRC jest niezbędne. Wyłączono tę funkcję, ponieważ trakcie wielu prób, gdy obliczanie CRC przez nRF2401 było włączone w formatach 8- lub 16 bitowym, nie udało prawidłowo przesłać pakietu danych. Być może był popełniany jakiś błąd, a może jest jakaś nieścisłość w dokumentacji dostarczonej przez producenta. Bez CRC transmisja radiowa jest trudna do zaakceptowania i dlatego użyto sprawdzonej procedury obliczającej 16-bitowe CRC na podstawie wielomianu $G(x)=x^{16}+x^{15}+x^{2}+1$. Przy wyliczaniu stosowano tablicowane współczynniki 16-bitowe. Dzięki temu podprogram obliczający jest prosty i co ważne - wykonuje się szybko (listing 5).

Następny bajt określa moc wyjściową, częstotliwość taktowania nRF2401, tryb pracy i maksymalną prędkość transmisji. Próby transmisji przeprowadzano z różnymi mocami. Przy odległościach rzędu 1...2 m transmisja z najniższą mocą i 250 kbps działa bez problemów. Docelowo została ustawiono najwyższą moc, a częstotliwość taktowania ustawiono na 16 MHz, bo taką częstotliwość ma scalony generator zamontowany na płycie modułu TLX2401. Prędkość transmisji

Listing 6. Funkcja inicjalizacji

```
//inicjalizacja nRF2401
void InitnRF(void){
    char i;
    CE_0 //CE=0, RX
    CS_1 //CS=1, konfiguracja układu
    for(i=0;i<15;i++) SendSPI(INIT_
nRF[i]);
    CS_0
    DATA_0
}
```

Listing 7. Ustawienie nadawania – włączony nadajnik

```
//ustawienie bitu kierunku
transmisji -> Rx
void Rx_mode(void){
    CE_0; //Rx
    CS_1 //konfiguracja
    DATA_1 //tryb Rx
    CLK1_1 //tick zegara
    CLK1_1
    CLK1_0
    CS_0
    CE_1
}
```

Listing 8. Ustawienie odbioru – włączony odbiornik

```
//ustawienie bitu kierunku
transmisji -> Tx
void Tx_mode(void){
    CE_0; //Tx
    CS_1 //konfiguracja
    DATA_0 //tryb Tx
    CLK1_1 //tick zegara
    CLK1_1
    CLK1_0
    CS_0
    CE_1
}
```

Aby ustawić układ w trybie nadawania lub odbioru, co może się zdarzać w trakcie pracy, trzeba przeprogramować tylko najmłodszy bit słowa konfiguracyjnego (tabela 2). Bloki kontroli nastaw nRF2401 są tak skonstruowane, że można wpisywać do nich niepełną liczbę bitów, w tym również tylko jeden bit. Funkcje pokazane na listingach 7 i 8 wykorzystują tę właściwość i przeprogramowują kierunek przesyłania danych przez układ nadawczo-odbiorczy, wpisując do rejestru konfiguracyjnego tylko jeden, ostatni bit. Wskutek tego przełączanie nadawanie-odbior jest wykonywane tak szybko, jak to możliwe.

Listing 9. Podprogram nadawania danych

```
//wysłanie danej do nRF2401
void SendnRF(unsigned char ld){
    char i;
    unsigned short int crc;
    DATA_out(); // wyjściowa linia danych
    buf_tx[0]=ADDR_SLAVE[0];//0xa1;
    buf_tx[1]=ADDR_SLAVE[1];//0x23; //bajty adresu modułu odb.
    buf_tx[2]=ADDR_SLAVE[2];//0x88;
    buf_tx[3]=ADDR_SLAVE[3];//0x17;
    buf_tx[4]=ADDR_SLAVE[4];//0x6c;
    buf_tx[5]=ld;
    for(i=6;i<5+ld;i++)
    buf_tx[i]=buf_snd_data[i-6];
    crc=CRC(ld,buf_tx+5); //wylczenie CRC
    buf_tx[i++]=crc>>8;
    buf_tx[i]=crc;
    CS_0
    CE_1
    for(i=0;i<ld+7;i++)
    SendSPI(buf_tx[i]); //zapis danych do nRF2401
    CE_0 //włączenie nadawania
    DATA_0
    delay100();
    CE_1
}
```

Listing 10. Podprogram odbioru danych

```
//odbiór danych z nRF
char RecnRF(void) {
    char i;
    unsigned char ld;
    unsigned short crcr, crc;
    for(i=0; i<SIZE_BUF_RX; i++)
        buf_rx[i]=0;
    while(test_DR1()==0);
    DATA_in(); //linia danych wejściowa
    buf_rx[0]=RecSPI();
    ld=buf_rx[0];
    for(i=1; i<ld+2; i++)
        buf_rx[i]=RecSPI();
    //if(test_DR1()==0)
    crcr=(buf_rx[ld]<<8)|buf_rx[ld+1];
    crc=CRC(ld, buf_rx);
    if(crc==crcr)
        return(0);
    else
        return(-1);
}
```

Listing 11. Pętla testowania przesyłania danych

```
InitnRF(); //wyślij sekwencję inicjalizacji
CE_1 //aktywacja odbioru
while(1) {
    Tx_mode();
    SendnRF(25); //wyślij dane
    delay_ms(1); //opcjonalnie
    Rx_mode();
    if(RecnRF()==0) //czekaj na dane do odbioru {
        LED_1
        delay_ms(20);
        LED_0
    } else {
        LED1_1
        delay_ms(20);
        LED1_0
    }
}
while(1);
```

Na **listingu 9** zamieszczono funkcję wysłania pakietu danych w trybie Shock-Burst. Najpierw linia danych jest ustawiana jako wyjściowa. Potem do bufora nadajnika jest wpisywanych 5 bajtów adresu. Ten adres musi być równy adresowi zaprogramowanemu w module TLX2401 – odbiorcy wysyłanych danych. Jeżeli tak nie będzie, to moduł odbiorczy odrzuci dane.

Potem do modułu wpisywany jest parametr *ld* zawierający liczbę danych w polu danych użytkownika. Na jej podstawie mikrokontroler podłączony od modułu odbiorczego może określić, ile ma ważnych danych w polu *PAYLOAD* odebranego pakietu. Ta identyfikacja liczby danych odbywa się na poziomie wymiany informacji pomiędzy mikrokontrolerami nadającymi

i odczytującymi dane z użyciem modułu TLX2401. Po wyliczeniu CRC i wypełnieniu bufora nadawczego, przez wymuszenie poziomu niskiego CE jest włączany nadajnik.

Odbiór danych (**listing 10**) rozpoczyna się od wyzerowania bufora odbiorczego. Następnie poprzez okresowe odpytywanie jest testowany stan linii DR1. Dane są gotowe do odbioru, gdy wyjście DR1 zostanie ustawione. Wtedy wiemy, że układ sprawdził zgodność adresów i odczytał kompletny pakiet danych.

Mikrokontroler, po ustawieniu swojej linii danych jako wyjściowej, odczytuje pierwszy bajt pola *PAYLOAD*. W nim jest zapisana liczba przesyłanych bitów. Po skompletowaniu danych jest wyliczana

suma CRC i porównywana z odebraną. Jeżeli sumy się zgadzają, to funkcja zwraca zero. W przeciwnym przypadku zwraca znacznik błędu –1.

Jak wspomniano, do testowania łącza radiowego użyto dwóch modułów z zestawami ZL1ARM. Testowanie polegało na wysyłaniu przez układ nazwany nadajnikiem do modułu umownie nazwanego odbiornikiem danych, po czym odbiornik przesyłał te dane ponownie do nadajnika – **listing 11**. Sprawdzano w ten sposób rzeczywistą szybkość i niezawodność przesyłania danych.

Uwagi końcowe

Wyniki testów były zachęcające. Jednak przy wykorzystywaniu łącza radiowego trzeba pamiętać o ważnej rzeczy. Nawet przy całkowitej sprawności wszystkich elementów toru nie ma 100% pewności, że wszystkie dane w każdym momencie zostaną prawidłowo przesłane. Zaniki mogą być na przykład spowodowane wyłączeniami atmosferycznymi. Dlatego w aplikacji trzeba wbudować mechanizmy pozwalające na uwiarygodnienie i retransmisję danych. Inaczej będzie wyglądało przesyłanie komend sterujących do zabawki, a inaczej odczytywanie ważnych pomiarów w automatyce.

Pierwszym bardzo ważnym elementem jest wielomian kontrolny CRC. Jeżeli dane są ważne, to można stosować mechanizm potwierdzania odebranych pakietów, a w przypadku braku potwierdzenia stosować mechanizm retransmisji. Wszystko to może spowodować, że rzeczywista prędkość przesyłania danych będzie znacznie niższa, niż oczekiwana.

Zastosowanie modułu TLX2401 jest idealnym rozwiązaniem szybkiego i niezawodnego krótkodystansowego łącza radiowego, zarówno w aplikacjach profesjonalnych jak i amatorskich.

Tomasz Jabłoński, EP
tomasz.jablonski@ep.com.pl

R E K L A M A

RK-SYSTEM
www.rk-system.com.pl

Profesjonalne narzędzia dla elektroników i programistów

- uniwersalne programatory układów scalonych
- analizatory stanów logicznych
- oscyloskopy cyfrowe
- systemy do wyważania i pomiaru drgań
- oprogramowanie CAD, CAM, CAE
- emulatory, symulatory, debuggery dla różnych rodzin procesorów
- kompilatory C/C++ dla różnych rodzin procesorów
- szkolenia w zakresie FPGA, VHDL
- narzędzia na procesory sygnałowe DSP
- projektujemy, produkujemy, szkolimy, dystrybuujemy

05-825 Grodzisk Maz., ul. Chałmońskiego 30, tel. (022) 724 30 33, 792 05 18, fax (022) 724 30 37

RAISONANCE Innovative Development Tools
IAR SYSTEMS
SPECTRUM DIGITAL