

Łącze radiowe w paśmie 900 MHz

Bezprzewodowa transmisja danych jest obecnie technologią powszechnie stosowaną w każdej dziedzinie techniki. O jej popularności może świadczyć liczba oraz różnorodność obszarów, w których występuje. Wśród licznych aplikacji korzystających z tego rodzaju komunikacji wymienić można między innymi systemy zdalnego sterowania, systemy telemetryczne, sieci czujnikowe, systemy komunikacji głosowej, automatykę budynkową, urządzenia medyczne, robotykę itp. W artykule zaprezentowano budowę łącza bezprzewodowego z użyciem modułów ER900TRS-02. Do ich obsługi wykorzystano zestawy rozwojowe STM32 Discovery z mikrokontrolerem STM32F100RBT6B.

Liczba zastosowań jest przyczyną nieustannie rosnącego zapotrzebowania na kolejne, coraz nowocześniejsze i bardziej zaawansowane rozwiązania z zakresu komunikacji bezprzewodowej. W wyniku tego trendu na rynku pojawia się wiele nowych układów scalonych będących transceiverami (układami nadawczo-odbiorczymi) pracującymi w paśmie ISM bądź mikrokontrolerami z wbudowanym modułem komunikacyjnym pracującym w tym paśmie częstotliwości. Należy jednak pamiętać, że wraz ze zwiększaniem funkcjonalności oferowanych przez producentów układów radiowych rośnie również stopień skomplikowania ich obsługi. Jednocześnie zaimplementowana w nich obsługa rozbudowanych standardów np. stosu ZigBee powoduje duży narzut transmisyjny protokołu. Cechy te powodują, iż w wielu nieskomplikowanych konstrukcjach urządzeń wykorzystujących komunikację radiową optymalnym rozwiązaniem są układy transmisyjne cechujące się przede wszystkim prostotą konfiguracji i realizacji transmisji.

Transceiver ER900TRS-02 firmy LPR

LPRS to firma będąca producentem i dystrybutorem produktów półprzewodnikowych różnego przeznaczenia. Większość jej oferty stanowią moduły do komunikacji bezprzewodowej, przede wszystkim jest seria o nazwie easy-Radio. Jest to grupa złożona z sześciu modułów radiowych, w tym trzy z nich przeznaczone są do pracy w paśmie częstotliwości 433 MHz, a trzy kolejne w paśmie 868 MHz. Do każdej z grup należy moduł odbiornika (ER400RS-02, ER900RS-02), nadajnika (ER400TS-02, ER900TS-02) oraz moduł nadawczo-odbiorczy (ER400TRS-02, ER900TRS-02). Na potrzeby realizowanego

łącza radiowego wybrany został moduł ER900TRS-02 (rysunek 1). Jego główne parametry zamieszczono w tabeli 1.

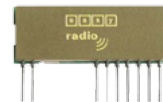
Producent, chcąc jak najbardziej ułatwić realizację projektów wykorzystujących moduły ER900TRS-02, zadbał o ich łatwą implementację zarówno sprzętowo jak i programistyczną. W strukturze modułów zintegrowane zostały wszystkie niezbędne do poprawnej pracy bloki funkcjonalne, dzięki czemu podstawowy schemat aplikacyjny wymaga jedynie zapewnienia napięcia zasilania, dołączenia anteny i połączenia z zewnętrznym układem sterującym (hostem). Główne elementy modułu to układ transmisyjny, stabilizator napięcia oraz mikrokontroler odpowiedzialny za obsługę układu transmisyjnego i udostępnianie interfejsu komunikacyjnego. Budowa struktury wewnętrznej modułu ER900TRS-02 została przedstawiona na rysunek 2.

Moduł ER900TRS-02 został wyposażony w interfejs komunikacyjny UART (domyślny

baudrate wynosi 19200 kbit/s). Za jego pomocą istnieje możliwość zarówno konfiguracji parametrów pracy, jak również obsługa transmisji. Konfiguracja obejmuje głównie zmianę częstotliwości pasma radiowego, parametrów interfejsu komunikacyjnego oraz mocy wyjściowej nadajnika. Zestawienie komend oraz odpowiadającym ich znaczeń przedstawiono w tabeli 2.

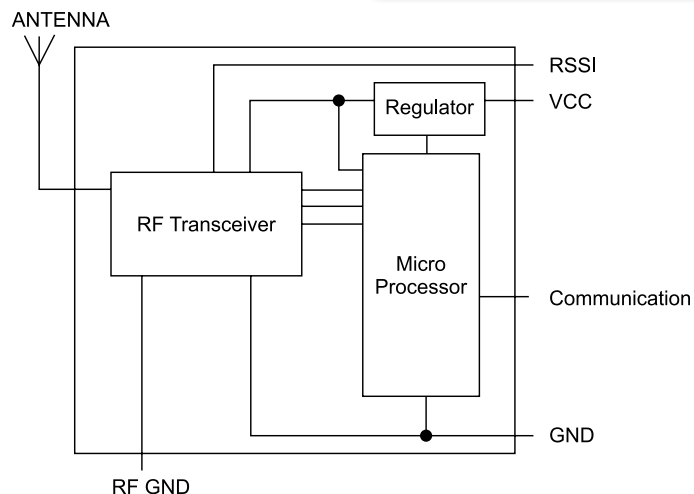
Procedura wykonania komendy odbywa się następująco:

- host (układ sterujący lub komputer) wysła komendę do modułu ER900TRS-02 np. ER_CMD#U5,
- moduł po odbiorze komendy potwierdza jej odebranie zwróceniem echa otrzymanej komendy np. ER_CMD#U5,



Rysunek 1. Moduł komunikacyjny ER900TRS-02 [1]

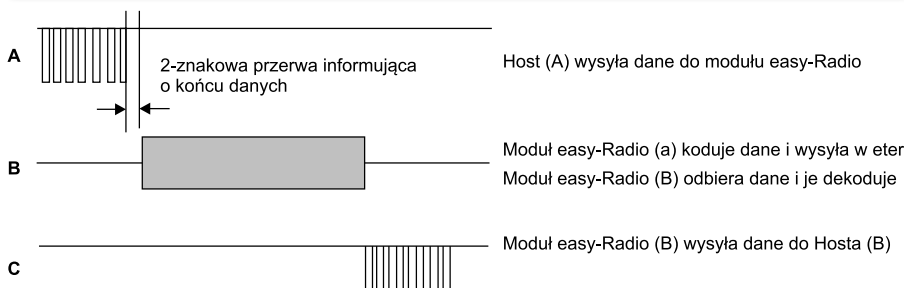
Tabela 1. Parametry układu ER900TRS-02 [1]	
Parametr	Wartość
Częstotliwość pracy	869,85 MHz – 914,65 MHz
Zasięg	około 150 m (max. 300 m)
Zasilnie	2,5 V – 5,5 V
Pobór prądu: - nadawanie - odbiór	25 mA 21 mA
Wymiary	38 mm × 14 mm × 4 mm
Czułość	około -105 dBm
Zakres temperatury pracy	-40 °C – +85 °C



Rysunek 2. Struktura wewnętrzna modułu ER900TRS-02 [1]

Tabela 2. Komendy konfiguracyjne dla modułu ER900TRS-02 [1]

Grupa komend	Komenda	Znaczenie
Parametry interfejsu komunikacyjnego UART	ER_CMD#U0	Domyślna prędkość transmisji
	ER_CMD#U1	Prędkość transmisji: 2400 kbit/s
	ER_CMD#U2	Prędkość transmisji: 4800 kbit/s
	ER_CMD#U3	Prędkość transmisji: 9600 kbit/s
	ER_CMD#U4	Prędkość transmisji: 19200 kbit/s
	ER_CMD#U5	Prędkość transmisji: 38400 kbit/s
	ER_CMD#U?	Zapytanie o aktualną prędkość transmisji
	ER_CMD#H1	Handshaking włączony
	ER_CMD#H2	Handshaking wyłączony
	ER_CMD#A70	Parzystość wyłączona
	ER_CMD#A71	Parzystość typu EVEN
	ER_CMD#A72	Parzystość typu ODD
	ER_CMD#I7	Potwierdzenia komend włączone
	ER_CMD#I8	Potwierdzenia komend wyłączone
Moc wyjściowa	ER_CMD#P0	Wartość mocy wyjściowej: 0.0625 mW
	ER_CMD#P1	Wartość mocy wyjściowej: 0.125 mW
	ER_CMD#P2	Wartość mocy wyjściowej: 0.25 mW
	ER_CMD#P3	Wartość mocy wyjściowej: 0.5 mW
	ER_CMD#P4	Wartość mocy wyjściowej: 1 mW
	ER_CMD#P5	Wartość mocy wyjściowej: 1.2 mW
	ER_CMD#P6	Wartość mocy wyjściowej: 1.5 mW
	ER_CMD#P7	Wartość mocy wyjściowej: 2 mW
	ER_CMD#P8	Wartość mocy wyjściowej: 2.5 mW
	ER_CMD#P9	Wartość mocy wyjściowej: 3.1 mW
ER_CMD#P?	Zapytanie o aktualną wartość mocy wyjściowej	
Częstotliwość kanału transmisyjnego	ER_CMD#C0	Częstotliwość: 869.9 MHz
	ER_CMD#C1	Częstotliwość: 914.65 MHz
	ER_CMD#C4	Częstotliwość: 869.85 MHz
	ER_CMD#C?	Zapytanie o aktualną częstotliwość
Inne	ER_CMD#R1	Przywrócenie ustawień fabrycznych
	ER_CMD#L0	Wejście w tryb uśpienia
	ER_CMD#A00	Wyłączenie możliwości zmiany kanału
	ER_CMD#A01	Włączenie możliwości zmiany kanału
	ER_CMD#A10	Szyfrowanie włączone
	ER_CMD#A11	Szyfrowanie wyłączone
	ER_CMD#A20	Wyłączenie cyklicznego kodu nadmiarowego CRC16 (na rzecz CRC8)
	ER_CMD#A21	Włączenie cyklicznego kodu nadmiarowego CRC16 (zamiast CRC8)
	ER_CMD#A30	Wyłączenie trybu "echo"
	ER_CMD#A31	Włączenie trybu "echo"
	ER_CMD#Fds	Nadawanie i odbiór danych na różnych kanałach (d – kanał odbiorczy, s – kanał nadawczy)
	ER_CMD#T0	Test diagnostyczny 0
	ER_CMD#T1	Test diagnostyczny 1
	ER_CMD#T2	Test diagnostyczny 2
ER_CMD#T3	Zapytanie o wersję oprogramowania	
ER_CMD#T4	Test danych w trybie RAW	



Rysunek 3. Schemat transmisji danych [1]

– host potwierdza otrzymanie echa po przez wysłanie znaków ACK.

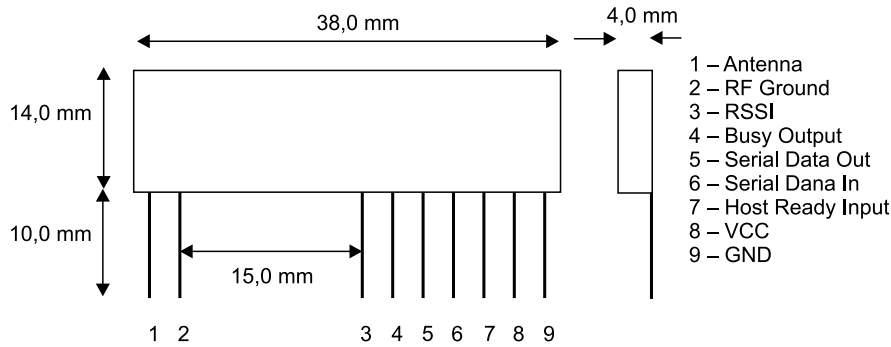
Wszystkie komendy muszą być przesyłane znakami ASCII zgodnie z ustalonymi parametrami transmisyjnymi interfejsu komunikacyjnego.

Interfejs komunikacyjny UART oprócz celów konfiguracyjnych służy również do realizacji transmisji danych. Schemat transmisji przedstawiono na **rysunku 3**.

Dane wysyłane z urządzenia hosta traktowane są jako dane do wysłania w eter radiowy. Istnieje możliwość przesłania w jednym pakiecie od 1 do 180 bajtów. Koniec ramki sygnalizowany jest przerwą o długości dwóch lub więcej znaków. W tym momencie moduł ER900TRS-02 przystępuje do przetwarzania odebranych danych, które są przygotowywane do wysłania, a następnie wysyłane w ustalonym paśmie częstotliwości. Poprawność danych zabezpieczona jest przez sumę kontrolną. Wszystkie moduły ER900TRS-02 będące w zasięgu odbioru modułu, który przesłał dane, odbierają wysłany pakiet, po czym dekodują go i wysyłają interfejsem UART do urządzeń host. Cały mechanizm przesyłania danych usprawniają dodatkowo trzy opcjonalne do wykorzystania wyprowadzenia modułu ER900TRS-02: RSSI, Busy Output, Host Ready Input. Wyprowadzenie RSSI (Received Signal Strength Indicator) informuje o poziomie sygnału odbiorczego. Wartość ta ma postać napięcia i ma wartość z przedziału 0 V (-65 dBm) do 1 V (-115 dBm), co daje dynamikę zmian rzędu 50 dBm/V. Wyprowadzenie Busy Output informuje o gotowości modułu do odbierania danych z urządzenia hosta. Pin Host Ready Input sygnalizuje gotowość urządzenia hosta do odbioru danych z modułu. Kolejność wszystkich wyprowadzeń oraz ich nazwa zostały przedstawione na **rysunek 4**.

Szybkie rozpoczęcie pracy z modułami ER900TRS-02 jest możliwe poprzez skorzystanie z będącego w ofercie firmy LPRS zestawu rozwojowego przygotowanego przez producenta dla zastosowań prototypowych (**rysunek 5**).

Zestaw składa się z dwóch modułów ER900TRS-02, dedykowanych dla nich płytek, anten, przewodów USB oraz dokumentacji i oprogramowania w formie płyty CD [2]. Płytki ewaluacyjne wyposażone zostały w gniazda na moduły, złącza antenowe, diody sygnalizacyjne LED oraz układ interfejsowy UART – USB o nazwie CP2103 (firmy Silicon Labs), który za pomocą interfejsu USB emuluje w komputerze wirtualny port COM. Kompleksowa obsługa modułów możliwa jest dzięki oprogramowaniu easyRadio Companion, które umożliwia przeprowadzanie konfiguracji modułów i transmisji danych (**rysunek 6**). Jako, że dane przesyłane są zgodnie ze standardem RS232 możliwe jest



Rysunek 4. Kolejność wyprowadzeń modułu ER900TRS-02 oraz ich nazwa [1]



Rysunek 8. Wykonany węzeł transmisyjny



Rysunek 5. Zestaw rozwojowy dla modułów ER900TRS-02

również sterowanie modułami z komputera za pomocą zwykłego terminala.

Łącze radiowe STM32 Discovery + ER900TRS-02

Uwzględniając charakterystykę modułów ER900TRS-02 najprostsza realizacja łącza radiowego z ich wykorzystaniem przewiduje dołączenie do nich anten oraz urządzeń host. Przykładowy model łącza

zbudowanego z dwóch tak opisanych węzłów transmisyjnych przedstawiono na rysunku 7.

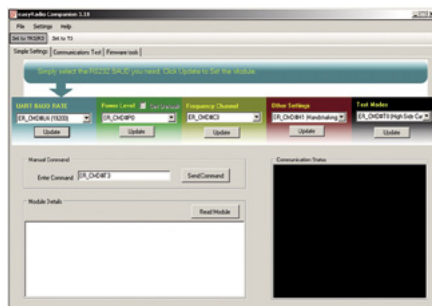
W praktycznej realizacji takiego łącza jako urządzenia host wykorzystano zestawy rozwojowe STM32 Discovery. Są to płytki ewaluacyjne wyposażone w mikrokontroler STM32F100RBT6B, dwie programowalne diody LED, jeden przycisk oraz programator z funkcją debugowania ST-Link [4]. Do-

kładny opis funkcjonalny zestawu STM32 Discovery ukazał się na łamach EP 11/2010.

Aby w sposób wygodny połączyć wszystkie wymagane elementy (zestaw STM32 Discovery, moduł ER900TRS-02 oraz antenę) wykonane zostały płytki, które posiadają gniazdo na moduł komunikacyjny, wejście antenowe oraz złącze szpilkowe z wyprowadzeniem pinów modułu komunikacyjnego. Do połączenia interfejsu komunikacyjnego i zasilania tak skonstruowanych płytek z makietami STM32 Discovery posłużyły złączki. Konstrukcję węzła transmisyjnego przedstawiono na rysunku 8. Przyporządkowanie połączonych wyprowadzeń przedstawiono w tabeli 3.

Oprogramowanie

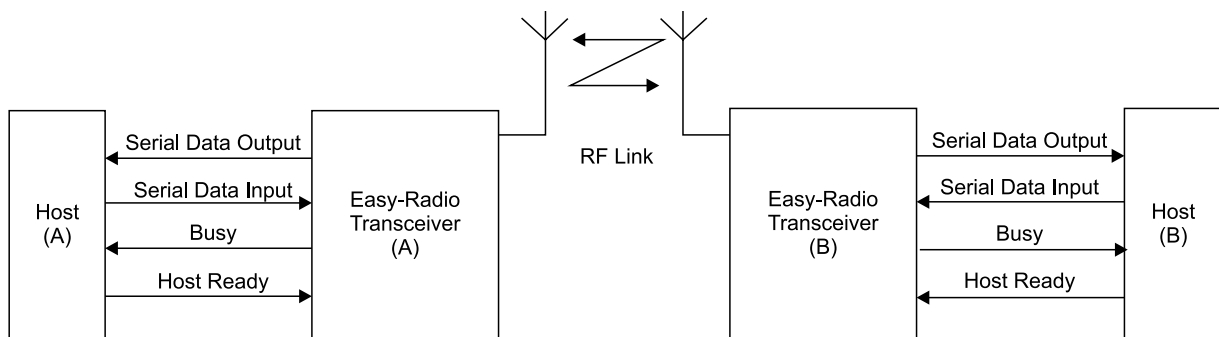
Kod programu realizujący komunikację między dwoma węzłami komunikacyjnymi został napisany za pomocą wysokopoziomo-



Rysunek 6 Oprogramowanie dla zestawów rozwojowych z modułami ER900TRS-02

Tabela 3. Sposób połączenia

Moduł ER900TRS-02		Zestaw STM32 Discovery	
Numer wyprowadzenia	Nazwa wyprowadzenia	Numer wyprowadzenia	Nazwa wyprowadzenia
1	Antenna	-	-
2	RF GROUND	-	GND
3	RSSI	-	-
4	Busy Output	44	PA11 (GPIO)
5	Serial Data Out	43	PA10 (USART1_RX)
6	Serial Data In	42	PA9 (USART1_TX)
7	Host Ready Input	45	PA12 (GPIO)
8	VCC	-	VCC
9	GND	-	GND



Rysunek 7. Model łącza radiowego zbudowanego z dwóch węzłów transmisyjnych [1]

wych bibliotek CMSIS (Cortex Microcontroller Software Interface Standard) dostarczanych przez producenta – firmę ST Microelectronics. Program został stworzony przy wykorzystaniu obsługiwanego przez programator ST-Link środowiska programistycznego IAR Embedded Workbench for ARM, niemniej jednak kod może zostać przeniesiony na dowolne inne środowisko bez dokonywania żadnych zmian.

Program umożliwia komunikację dwukierunkową, czyli zarówno wysyłanie danych, jak również odbieranie. Wykonanie każdej z tych czynności jest każdorazowo sygnalizowane poprzez krótkotrwałe zaświecenie diody LED (odpowiednio: wysyłanie poprzez diodę na porcie PC8 oraz odbieranie poprzez diodę na porcie PC9). Nadawanie jest inicjowane pod wpływem wciśnięcia przycisku (wystąpienie stanu wysokiego na porcie PA0). Schemat blokowy obrazujący działanie programu przedstawiono na rysunku 9.

Program rozpoczyna się od dyrektywy `#include "stm32f10x.h"` dołączającej biblioteki CMSIS. Następnie zdefiniowane zostają zmienne (struktura `GPIO_InitStructure` do konfiguracji portów, struktura `USART_InitStructure` do konfigurowania interfejsu USART, zmienna `UART_Buffer` do przechowywania odebranych danych, zmienna

Listing 1. Definicja struktury do konfigurowania USART

```
#include "stm32f10x.h"
GPIO_InitTypeDef GPIO_InitStructure;
USART_InitTypeDef USART_InitStructure;
u8 UART_Buffer = 0;
u8 button_value = 0;
void Delay(_IO uint32_t nCount)
{
    for(; nCount != 0; nCount--);
}
```

`button_value` do odczytu stanu portu podłączonego do przycisku) oraz funkcja `Delay` umożliwiająca wprowadzenie opóźnienia dla zaobserwowania efektu świecenia diod LED (listing 1).

W następnej kolejności program rozpoczyna wykonywanie funkcji `main`, na początku której włączone zostają zegary dla wykorzystywanych portów GPIOA i GPIOC oraz interfejsu USART1. Po tej czynności następuje skonfigurowanie wszystkich portów zgodnie z ich przeznaczeniem (listing 2).

Następny krok to nastawy interfejsu USART. Domyślne parametry modułu ER900TRS-02 to baudrate 19200, 8 bitów danych, 1 bit stopu, brak parzystości. Ten etap kończy część konfiguracyjną mikrokontrolera. Aby umożliwić odbieranie danych należy wystawić wartość logiczną 0 na porcie odpowiadającym linii Host Ready Input (listing 3).

Obsługa transmisji danych odbywa się w nieskończonej pętli `while(1)`. Odczyt stanu portu od przycisku wykonywany jest za pomocą funkcji `GPIO_ReadInputDataBit()`. Odczytana wartość zapisywana jest do zmiennej `button_value`. Po sprawdzeniu jej wartości instrukcją warunkową `if` jest podejmowana decyzja, czy wysłać bajt danych o wartości 0xAC. Czynność tą realizuje funkcja `USART_SendData()`. Poprawność tego procesu sprawdzana jest za pomocą flagi `USART_FLAG_TXE`. Po wysłaniu danych wykonywane jest krótkotrwałe podanie napięcia na diodę LED (funkcje `GPIO_SetBits()` oraz `GPIO_ResetBits()`), co powoduje mignięcie diodą LED. Następnie wykonywane jest sprawdzenie, czy nowe dane nie oczekują na odbiór z modułu komunikacyjnego. Odbywa się to za pomocą odczytu stanu flagi `USART_FLAG_RXNE` świadczącej o nadejściu danych. W przypadku ustawionej flagi dane są odbierane (funkcja `USART_ReceiveData()`), co jest sygnalizowane chwilowym zaświeceniem diody LED (listing 4).

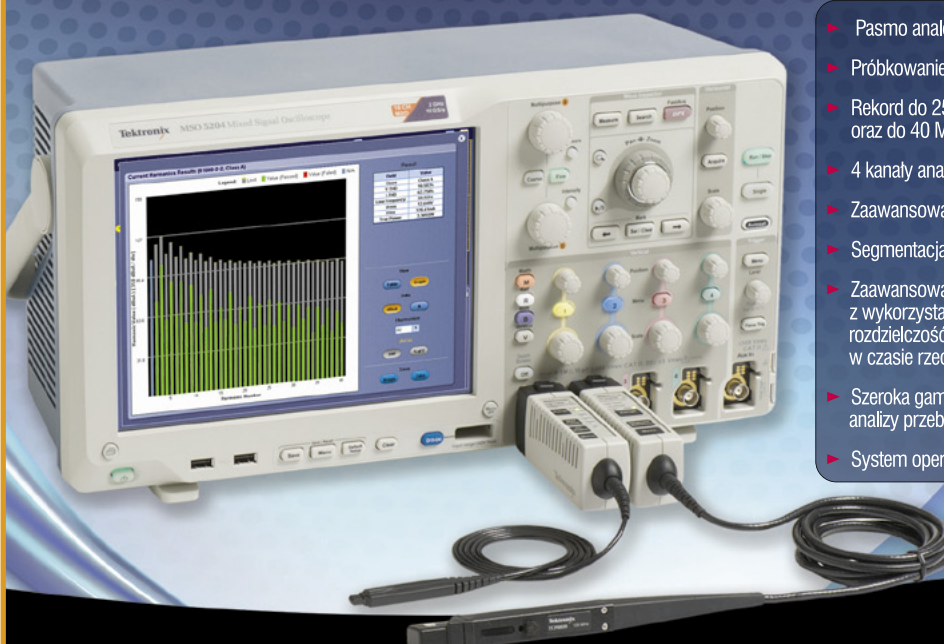
Podsumowanie

Na podstawie przedstawionej w pracy analizy zarówno parametrów modułów ER900TRS-02 jak i zrealizowanego łącza radiowego z ich wykorzystaniem można stwier-

REKLAMA

Oscyloskopy serii DPO5000 oraz MSO5000

to najlepsze rozwiązanie mogące sprostać wyzwaniom, przed jakimi stoją projektanci weryfikując, testując oraz wyszukując błędy w złożonych projektach elektronicznych.



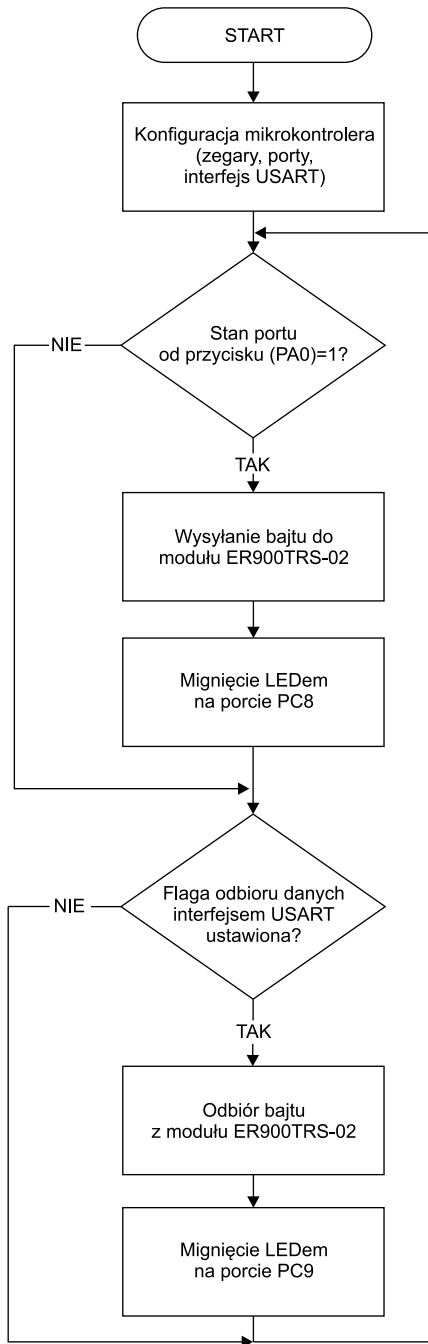
- ▶ Pasma analogowe 350 MHz, 500 MHz, 1 GHz lub 2 GHz
- ▶ Próbkowanie 10 GS/s
- ▶ Rekord do 250 M punktów dla kanałów analogowych oraz do 40 M punktów cyfrowych (wersja MSO),
- ▶ 4 kanały analogowe oraz 16 kanałów cyfrowych (wersja MSO)
- ▶ Zaawansowany system wyzwalania
- ▶ Segmentacja pamięci
- ▶ Zaawansowana analiza sygnałów cyfrowych (wersja MSO) z wykorzystaniem technologii MagniVu zapewniającej rozdzielczość czasową 60,6ps z szybkością pracy 16.5 GS/s w czasie rzeczywistym
- ▶ Szeroka gama wbudowanych narzędzi zaawansowanej analizy przebiegu
- ▶ System operacyjny Windows 7



Tektronix
Enabling Innovation

Siedziba Firmy: 54-413 Wrocław, ul. Klecińska 125, tel. 71 783 63 60, fax 71 783 63 61
Biuro Handlowe: 03-301 Warszawa, ul. Jagiellońska 74, tel. 22 675 75 42

tespol@tespol.com.pl • www.tespol.com.pl



Rysunek 9. Schemat blokowy programu

dzić, że produkt firmy Low Power Radio Solutions to bardzo ciekawe rozwiązanie, które cechuje się łatwością implementacji sprzętowej i programistycznej. Prosta forma komunikacji pozwala w razie potrzeby na rozbudowę transmisji o dodatkową funkcjonalność, którą może być np. dostosowanie modułów do pracy zgodnej z dedykowanym protokołem komunikacyjnym umożliwiającym chociażby wprowadzenie adresacji.

Prostota budowy funkcjonalnego i niezawodnego toru komunikacyjnego jest niewątpliwą zaletą tych modułów, dzięki czemu

Listing 2. Konfigurowanie portów I/O

```

RCC_APB2PeriphClockCmd (RCC_APB2Periph_USART1 | RCC_APB2Periph_GPIOA |
RCC_APB2Periph_GPIOC, ENABLE);

//uart TX
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_9;
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF_PP;
GPIO_Init(GPIOA, &GPIO_InitStructure);

//uart RX
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_10;
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_IN_FLOATING;
GPIO_Init(GPIOA, &GPIO_InitStructure);

//er900trs Host Ready Input
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_12;
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
GPIO_Init(GPIOA, &GPIO_InitStructure);

//STM32 Discovery button
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_0;
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
GPIO_Init(GPIOA, &GPIO_InitStructure);

//STM32 Discovery LEDs
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_8 | GPIO_Pin_9;
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
GPIO_Init(GPIOC, &GPIO_InitStructure);
    
```

Listing 3. Konfigurowanie USART

```

USART_InitStructure.USART_BaudRate = 19200;
USART_InitStructure.USART_WordLength = USART_WordLength_8b;
USART_InitStructure.USART_StopBits = USART_StopBits_1;
USART_InitStructure.USART_Parity = USART_Parity_No ;
USART_InitStructure.USART_HardwareFlowControl = USART_HardwareFlowControl_
None;
USART_InitStructure.USART_Mode = USART_Mode_Rx | USART_Mode_Tx;
USART_Init(USART1, &USART_InitStructure);
USART_Cmd(USART1, ENABLE);

GPIO_ResetBits(GPIOA, GPIO_Pin_12);
    
```

Listing 4. Odbiór i wysyłanie danych

```

while (1)
{
    button_value = GPIO_ReadInputDataBit(GPIOA, GPIO_Pin_0);
    if(button_value == 1)
    {
        //nadawanie danych
        USART_SendData(USART1, 0xAC);
        while(USART_GetFlagStatus(USART1, USART_FLAG_TXE) == RESET) {}

        GPIO_SetBits(GPIOC, GPIO_Pin_8);
        Delay(0xAFFFFF);
        GPIO_ResetBits(GPIOC, GPIO_Pin_8);
        Delay(0xAFFFFF);
    }

    //odbieranie danych
    if(USART_GetFlagStatus(USART1, USART_FLAG_RXNE) == SET)
    {
        UART_Buffer = USART_ReceiveData(USART1);

        GPIO_SetBits(GPIOC, GPIO_Pin_9);
        Delay(0xAFFFFF);
        GPIO_ResetBits(GPIOC, GPIO_Pin_9);
        Delay(0xAFFFFF);
    }
}
    
```

mogą one znaleźć zastosowanie nie tylko w profesjonalnych aplikacjach, ale również tych amatorskich. Dodatkowymi atutami modułów ER900TRS-02 jest ich niewygórowany koszt zakupu (cena u producenta to £25) oraz duża dostępność na rynku (jednym z licznych dystrybutorów jest firma Farnell).

Szymon Panecki
szymon.panecki@pwr.wroc.pl

Bibliografia:

- [www.lprs.co.uk ERx00-02 Series Datasheet](http://www.lprs.co.uk/ERx00-02_Series_Datasheet)
- [www.lprs.co.uk Easy Radio USB Evaluation Kit Datasheet](http://www.lprs.co.uk/Easy_Radio_USB_Evaluation_Kit_Datasheet)
- [www.st.com STM32F100RBT6B Datasheet](http://www.st.com/STM32F100RBT6B_Datasheet)
- [www.st.com STM32 Discovery User Manual](http://www.st.com/STM32_Discovery_User_Manual)

<http://ep.com.pl>