

Nadawanie w paśmie radiowym 76...108 MHz, w którym pracuje prezentowane urządzenie wymaga koncesji. Prezentowany projekt ma charakter wyłącznie eksperymentalny i edukacyjny. Redakcja Elektroniki Praktycznej nie bierze odpowiedzialności za skutki jego wykorzystania.

transmitterFM

Miniaturowy nadajnik FM z RDSem



Prezentowany projekt jest nieskomplikowanym nadajnikiem FM pracującym w paśmie 76...108 MHz. Spośród innych projektów tego typu wyróżnia go funkcja RDS, która umożliwia przesyłanie komunikatów tekstowych do radioodbiornika przystosowanego do ich odbioru wraz z transmitowanym sygnałem audio.

Rekomendacje: projekt ma nie tylko charakter edukacyjny, ale również wymiar praktyczny – może np. przydać się do transmitowania sygnału audio z komputera PC do radioodbiornika w obrębie mieszkania.



Coraz częściej utwierdzam się w przekonaniu, że elektronika nigdy nie przestanie mnie zadziwiać i pasjonować. Nie tak dawno na łamach Elektroniki Praktycznej prezentowałem projekt przenośnego radioodbiornika FM ze wsparciem dla RDS, zbudowany z użyciem odbiornika scalonego Si4703 firmy Silicon Labs. Ten projekt uświadomił mi, jak łatwe jest zbudowanie zaawansowanego urządzenia z użyciem niewielkiej liczby komponentów, a to wszystko za sprawą doskonałych układów scalonych. Wystarczy zastanowić się, ile energii i zapału należałoby poświęcić skonstruowaniu podobnego urządzenia używając elementów dyskretnych.

Może się wydawać, że taki sposób konstruowania urządzeń jest nieciekawym, gdyż nie wymaga inżynierskiego podejścia do zagadnienia wyrażającego się potrzebą wykonania dziesiątek czy setek obliczeń popartych sporą wiedzą, lecz moim zdaniem nie jest to prawdą, ponieważ w naszym wypadku cała „zabawa” zaczyna się w momencie tworzenia aplikacji odpowiadającej założeniom, co przecież nie miało miejsca w ujęciu tradycyjnym. Jest to zwyczajnie inna droga do osiągnięcia tego samego celu, niemniej ciekawa czy wymagająca.

Układ scalony nadajnika Si4711

Bazując na swoim doświadczeniu z układem Si4703 oraz wygodzie użytkownika telefonu komórkowego ze zintegrowanym nadajnikiem FM, postanowiłem sięgnąć po inny, bardzo ciekawy element z szerokiej palety układów tego producenta. Mowa o scalonym transmitterze FM o symbolu Si4711/4713 wyposażonym w funkcjonalność RDS. Charakteryzuje się on następującymi parametrami:

- Zintegrowany nadajnik FM w zakresie częstotliwości 76...108 MHz.
- Cyfrowa synteza częstotliwości z wbudowanym oscylatorem VCO.
- Analogowe i cyfrowe wejścia audio standardu I²S.
- Cyfrowa regulacja mocy emitowanego sygnału.
- Programowana wartość deemfazy.
- Cyfrowy modulator stereofoniczny.
- Pomiar mocy sygnału antenowego.
- Wbudowany układ oscylatora dla rezonatora kwarcowego 32768 Hz.
- Interfejs sterowania i kontroli standardu I²C, SPI oraz 3-Wire.
- Niewielka liczba komponentów zewnętrznych w aplikacji podstawowej.
- Brak konieczności strojenia obwodów radiowych, gdyż w układzie wykorzystano cyfrową obróbkę sygnałów przy wsparciu zaawansowanych technik DSP.

W ofercie AVT*

AVT-5437 A
AVT-5437 UK

Podstawowe informacje:

- Napięcie zasilania: 8...14 V DC.
- Maksymalny prąd zasilania (wyświetlacz załączony/wyłączony): 35 mA/17 mA.
- Zakres częstotliwości nadajnika FM: 87,5...108 MHz.
- Typ i liczba obsługiwanych wiadomości RDS: 4×PS (Program Service, 8 znaków).
- Maksymalny poziom wejściowego sygnału audio: 636 mV.
- Impedancja wejściowa: 60 kΩ.

Dodatkowe materiały na CD lub FTP:

ftp://ep.com.pl, user: 85414, pass: 2nev3854

wzory płytek PCB

- karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

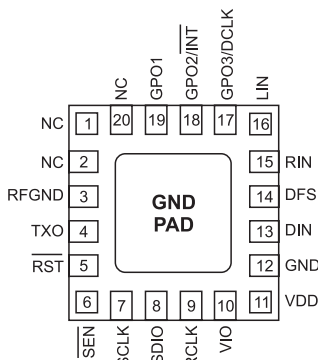
Projekty pokrewne na CD/FTP:

(wymienione artykuły są w całości dostępne na CD)

- AVT-5401 Radioodbiornik kieszonkowy z RDS (EP 6-7/2013)
- AVT-5317 Lampowo-tranzystorowy odbiornik UKF (EP 11/2011)
- AVT-2723 Stereofoniczny nadajnik FM (EdW 5/2004)
- AVT-5016 Amplituner FM z RDS (EP 6-7/2001)
- AVT-2481 Mininadajnik FM UKF (EdW 4-5/2001)
- AVT-2469 Odbiornik UKF FM (EdW 1/2001)
- AVT-2330 Miniaturowy odbiornik FM stereo (EdW 2/1999)

* 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
AVT xxxx C 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
AVT xxxx CD oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można pobrać, klikając w link umieszczony w opisie kitu)
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://sklep.avt.pl>



Rysunek 1. Rozmieszczenie wyprowadzeń układu Si4713

- Szeroki zakres napięcia zasilania (2,7...5,5 V).
- Mały pobór mocy (w tym tryb o ekstremalnie niskim zapotrzebowaniu na energię).
- Wbudowany regulator napięcia.
- Obsługa systemu RDS/RDBS.
- Małe wymiary obudowy (3 mm×3 mm).

Wymienione parametry układu predysponują go do zastosowania również w konstrukcjach amatorskich zwłaszcza, że aplikacja układu ogranicza się do zaledwie kilku elementów zewnętrznych. Wynika to z faktu, że w budowie układu Si4711/4713 zastosowano zaawansowany, cyfrowy tor przetwarzania sygnału. Na **rysunku 1** pokazano wygląd obudowy układu Si4713, natomiast w **tabeli 1** zamieszczono opis i znaczenie poszczególnych wyprowadzeń.

Jak to często bywa w przypadku układów peryferyjnych, obsługa i konfiguracja układu Si4713 jest możliwa dzięki wyposażeniu go interfejs szeregowy 3-wire kompatybilny z I²C i SPI oraz szereg rejestrów kontrolnych, których zapisywanie zmienia bieżący stan nadajnika, natomiast odczytywanie pozwala na monitorowanie stanu pracy układu. Aby jednak rozpocząć właściwą transmisję, jest konieczne poprawne zainicjowanie układu, które ma na celu wybór aktywnej magistrali sterującej (pomiędzy I²C, SPI i 3-wire) oraz uruchomienie wewnętrznego oscylatora niezbędnego z punktu widzenia części radiowej układu.

Rodzaj interfejsu komunikacyjnego wybiera się poprzez ustawienie odpowiednich stanów logicznych na wyprowadzeniach GPIO1 oraz GPIO2/INT. Logika sterująca zapamiętuje go podczas rosnącego zbocza sygnału na wejściu zerowania układu RST. Możliwe ustawienia umieszczono w **tabeli 2**.

W strukturze układu Si4713 zintegrowano rezystory podciągające wyprowadzenia GPIO1 oraz GPIO2/INT, odpowiednio, do napięcia zasilającego (pull-up) oraz do masy zasilania (pull-down) aktywne podczas sygnału zerującego, co oznacza, że te wyprowadzenia należy odpowiednio spolaryzować wyłącznie wtedy, gdy wymagany stan jest inny, niż domyślny. W tab. 2 nastawy domyślne oznaczono symbolem (*).

Po wyjściu z procedury zerowania układu i wyborze aktywnej magistrali sterującej, układ Si4713 pozostaje w trybie *POWER_DOWN*, gotowy na przyjęcie rozkazów sterujących, przy czym pierwszym, dopuszczalnym rozkazem, jaki może na tym etapie wykonać jest *POWER_UP* inicjujący proces aktywacji wewnętrznych peryferiów nadajnika, w tym zintegrowanego oscylatora. Warto przy okazji podkreślić, iż producent układu, firma Silicon Labs, zmieniła i ustandaryzowała interfejs programowy dla całej serii Si47xx, co ułatwiło programowanie tej rodziny układów.

Oprogramowanie

Ramka danych, którą należy przesłać do układu zawiera predefiniowany rozkaz sterujący (1 bajt) oraz krótką listę argumentów (maksymalnie 7 bajtów), których liczba i znaczenie są uzależnione od przesłanego wcześniej rozkazu sterującego. Po wysłaniu takiej ramki danych, układ nadrzędny (mikrokontroler) może zainicjować proces odczytu statusu wykonania rozkazu sterującego, który każdorazowo jest przygotowywany przez układ (maksymalnie 15 bajtów odpowiedzi), lecz jest to operacja opcjonalna. W **tabeli 3** umieszczono przykładową sekwencję sygnałów sterujących w relacji Master (mikrokontroler) <-> Slave (Si4713) w wypadku wysyłania rozkazu sterującego *TX_TUNE_FREQ*, którym jest zmieniana częstotliwość transmisji FM. Specjalnym rodzajem rozkazu sterującego jest rozkaz *SET_PROPERTY*, którym określamy szereg właściwości funkcjonalno-sprzętowych układu transmittera Si4703 posiłkując się listą dołączonych argumentów. Konstrukcję ramki danych w przypadku wysyłania rozkazu *SET_PROPERTY* pokazano w **tabeli 4**.

Na listingach pokazano funkcje odpowiedzialne za realizację wspomnianych

funkcjonalności tj. wysłania rozkazu sterującego wraz z listą niezbędnych argumentów (**listing 1**) i odczyt statusu odpowiedzi oraz funkcji odpowiedzialnej za zmianę właściwości funkcjonalnych układu Si4713 (**listing 2**). W ciele funkcji *Si4713sendCommand* użyto funkcji narzędziowej *Si4713WaitForCTS*, której zadaniem jest sprawdzenie gotowości układu Si4713 na przyjęcie danych (dokładnie, sprawdzenie bitu CTS w rejestrze statusu układu). Funkcję *Si4713WaitForCTS* pokazano na **listingu 3**. Funkcję inicjalizującą układ pokazano na **listingu 4**. Do analizy kodu funkcji inicjalizacyjnej jest niezbędna znajomość zawartości pliku nagłówkowego, który za każdym razem tworzę w taki sposób, aby bez konieczności zaglądania do dokumentacji układu, móc zorientować się w możliwościach peryferium, którego ten kod dotyczy. Zawartość wspomnianego pliku nagłówkowego pokazano na **listingu 5**.

Na koniec przedstawię kod kilku krótkich funkcji narzędziowych pozwalających na ustawienie częstotliwości i mocy transmitowanego sygnału FM oraz dającą możliwość zdefiniowania własnych wiadomości systemu RDS (typu *Program Service*). Kody dwóch pierwszych funkcji pokazano na **listingach 6 i 7**. Na **listingach 8 i 9** pokazano kod dwóch krótkich funkcji pozwalających na ustawienie wiadomości typu RDS *Program Service* (nazwa stacji), którą układ Si4713 będzie transmitował razem z wysyłanym

Tabela 2. Wybór rodzaju interfejsu komunikacyjnego

Magistrala sterująca	GPIO1	GPIO2/INT
I ² C	1	0
SPI	1	1 (*)
3-Wire	0 (*)	0

Tabela 1. Opis i oznaczenie wyprowadzeń układu Si4713

Numer pinu	Nazwa	Opis
1, 2, 20	NC	Niedołączone
3	RFGND	Masa części radiowej układu (należy połączyć z polem masy PCB)
4	TXO	Wyjście sygnału antenowego
5	RST	Wejście zerujące układu (aktywny stan niski)
6	SEN	Wejście aktywacji i wyboru rodzaju magistrali sterującej (aktywny stan niski)
7	SCLK	Wejście zegarowe magistrali sterującej
8	SDIO	Wejście/wyjście danych magistrali sterującej
9	RCLK	Wejście zewnętrznego sygnału zegarowego syntezy częstotliwości
10	VIO	Napięcie zasilania układów wejścia/wyjścia
11	VD	Napięcie zasilania części cyfrowej układu
13	DIN	Wejście danych cyfrowej magistrali audio
14	DFS	Wejście synchronizacji ramki danych cyfrowej magistrali audio
15	RIN	Analogowe wejście audio – kanał prawy
16	LIN	Analogowe wejście audio – kanał lewy
17	GPIO3/DLCK	Uniwersalny, programowalny port IO/sygnał zegarowy synchronizacji bitów danych cyfrowej magistrali audio
18	GPIO2/INT	Uniwersalny, programowalny port IO/sygnał zgłoszenia przerwania
19	GPIO1	Uniwersalny, programowalny port IO
12, GND PAD	GND	Masa (należy połączyć z polem masy PCB)

Listing 1. Funkcja wysyłająca rozkaz sterujący wraz z listą niezbędnych argumentów oraz odczyt statusu odpowiedzi
 /* Funkcja wysyłająca rozkaz Command, towarzyszące mu argumenty *Arguments w liczbie ArgNumbers i zwracająca odpowiedzi *Responses w liczbie RspNumbers (jesli są w ogóle oczekiwane */

```

void Si4711sendCommand(uint8_t Command, uint8_t *Arguments, uint8_t ArgNumber, uint8_t *Responses, uint8_t RspNumber )
{
    Si4711WaitForCTS(); //Sprawdzamy gotowość układu na dane
    TWI_Start();
    TWI_WriteByte(Si4711_WRITE_ADDR);
    TWI_WriteByte(Command);
    while(ArgNumber--) TWI_WriteByte(*Arguments++);
    TWI_Stop();
    if(RspNumber)
    {
        Si4711WaitForCTS(); //Czekamy na gotowosc układu po wysłce
        TWI_Start();
        TWI_WriteByte(Si4711_READ_ADDR);
        /* Jesli oczekiwana była odpowiedź układu Si4711 to odczytujemy ją (liczę bajtów okresloną argumentem RspNumbers) */
        while(RspNumber--)
            *Responses++ = TWI_ReadByte( RspNumber == 0? NACK:ACK);
        TWI_Stop();
    }
}
    
```

dźwiękiem. Przy okazji należy wspomnieć, iż tego typu konfiguracja układu Si4713 w zakresie systemu RDS jest rozwiązaniem najprostszym i najszybszym, jednak sam układ udostępnia ogromną listę możliwości, jeśli chodzi o transmisję wiadomości tego typu.

Dla zrozumienia sposobu działania funkcji narzędziowych, trzeba opisać najważniejsze rozkazy sterujące z listą ich argumentów oraz wybranych właściwości funkcjonalnych układu Si4713. Poświęćmy, zatem chwilę na tą krótką lekturę.

Rozkaz	POWER_UP (0x01)							
Bity	D7	D6	D5	D4	D3	D2	D1	D0
Rozkaz	0	0	0	0	0	0	0	1
Argument1	CTS IEN		GPO2OEN	PATCH	XOSCEN		FUNC[3...0]	
Argument2	OPMODE[7...0]							

Rozkaz *POWER_UP* musi być pierwszym rozkazem, jaki zostanie wysłany do układu nadajnika po ustawieniu sygnału zerowania (RST=1), gdyż umożliwia aktywację peryferiów układu Si4713 oraz konfigurację podstawowych parametrów jego pracy. Zna-

Listing 2. Kod funkcji umożliwiającej zmianę właściwości funkcjonalnych układu Si4713

```

/* Funkcja narzędziowa pozwalająca na zmianę wybranego ustawienia układu Si4713 (tzw. Property) */
void Si4713setProperty(uint16_t Property, uint16_t Value)
{
    Arguments[0] = 0;
    //Wybrane ustawienie - Property
    Arguments[1] = (Property >> 8);
    Arguments[2] = (Property & 0x00FF);
    //Wartosc dla wybranego ustawienia
    Arguments[3] = (Value >> 8);
    Arguments[4] = (Value & 0x00FF);
    //Nie odczytujemy odpowiedzi
    Si4713sendCommand(SET_PROPERTY_CMD, Arguments, 5, NULL, 0);
}
    
```

Listing 3. Kod funkcji sprawdzającej gotowość układu Si4713 na przyjęcie danych

```

/* Funkcja czekająca na ustawienie bitu CTS w rejestrze statusu - Clear to Send - czyli zgłoszenie gotowości na odbieranie danych */
void Si4713WaitForCTS(void)
{
    register uint8_t Status = 0, waitTime = 255;
    while( --waitTime && !(Status & CTS) )
    {
        TWI_Start();
        TWI_WriteByte(Si4713_READ_ADDR);
        Status = TWI_ReadByte(NACK);
        TWI_Stop();
        _delay_ms(1);
    }
}
    
```

Ustawienie fusbiteów:
 CKSEL3...0: 0100
 SUT1...0: 10
 CKOPT: 1
 EESAVE: 0

czenie poszczególnych bitów w kolejnych argumentach przedstawia się następująco:

- CTS IEN: ustawienie tego bitu aktywuje przerwanie CTS (Clear to Send),
- GPO2OEN: ustawienie tego bitu aktywuje wyjście GPIO2, wyzerowanie wymusza stan wysokiej impedancji na wspomnianym wyjściu,
- PATCH: stan tego bitu ma wpływ na przebieg procedury inicjalizacyjnej, jaka wykonywana jest przez firmware transmittera

Tabela 3. Przykładowa sekwencja sygnałów sterujących w relacji Master→Slave w przypadku wysłania rozkazu sterującego TX_TUNE_FREQ

Rodzaj bajtu	Wartość	Kierunek transmisji	Opis
Rozkaz sterujący	0x30	Master→Slave	TX_TUNE_FREQ
Argument	0x00	Master→Slave	Wymagany bajt 0x00
Argument	0x27	Master→Slave	Starsza część 16-bitowego słowa reprezentującego częstotliwość transmisji
Argument	0x7E	Master→Slave	Młodsza część 16-bitowego słowa reprezentującego częstotliwość transmisji
Status operacji	0x80	Master→Slave	Status wykonania operacji – CTS (Clear to Send)

Tabela 4. Przykładowa sekwencja sygnałów sterujących w relacji Master→Slave w przypadku wysłania rozkazu sterującego SET_PROPERTY

Rodzaj bajtu	Wartość	Kierunek transmisji	Opis
Rozkaz sterujący	0x12	Master→Slave	SET_PROPERTY
Argument	0x00	Master→Slave	Wymagany bajt 0x00
Argument	0x02	Master→Slave	PROPERTY (wybrana właściwość funkcjonalna) – bajt MSB
Argument	0x01	Master→Slave	PROPERTY (wybrana właściwość funkcjonalna) – bajt LSB
Argument	0x80	Master→Slave	VALUE (wartość dla wybranej właściwości) – bajt MSB
Argument	0x00	Master→Slave	VALUE (wartość dla wybranej właściwości) – bajt LSB
Status operacji	0x80	Master→Slave	Status wykonania operacji – CTS (Clear to Send)

REKLAMA

Projekty na...Texas

www.stm32.eu

life.augmented

KAMAMI

Listing 4. Kod funkcji inicjalizacyjnej układu Si4713

```
void Si4713init(void)
{
    RADIO_CTRL_PORT |= (1<<SCLK_PIN); //Ustawienie SCLK=1
    //RES=0, SDIO=0, SCLK=1
    RADIO_CTRL_DDR |= ((1<<RES_PIN) | (1<<SDIO_PIN) | (1<<SCLK_PIN));
    delay_us(500);
    //RES=1, SDIO=0, GPO1=1, GPO2=0
    RADIO_CTRL_PORT |= (1<<RES_PIN);
    delay_us(500);
    TWI_Init(); //Inicjalizacja magistrali I2C mikrokontrolera
    /* PowerUP - uruchamiamy oscylator zewnętrzny oraz moduł radiowy w trybie transmittera FM */
    Arguments[0] = USE_CRYSTAL | TX_MODE;
    Arguments[1] = ANALOG_AUDIO_INPÜTS;
    Si4713sendCommand(POWER_UP_CMD, Arguments, 2, NULL, 0);
    _delay_ms(500); //Czas niezbędny na aktywację układu
    /* Uruchamiamy komponenty sygnału FM układu Si4713: pilot Stereo, sygnał L-R, RDS */
    Si4713setProperty(TX_COMPONENT_ENABLE, TX_ENABLE_PILOT | TX_ENABLE_LMR | TX_ENABLE_RDS);
    /* Ustawiamy maksymalny, dopuszczalny poziom wejściowego sygnału audio oraz aktualne maksimum */
    Si4713setProperty(TX_LINE_INPUT_LEVEL, MAX_INPUT_LEVEL_636 | 636);
    //Ustawiamy wartość preemfazy dla Europy
    Si4713setProperty(TX_PREEMPHASIS, PREEMPHASIS_50);
    /* Ustawiamy jaki procent emisji sygnału RDS zajmować będzie nazwa programu (Program Service) */
    Si4713setProperty(TX_RDS_PS_MIX, RDS_PS100percent);
}
```

po ustaniu sygnału zerowania. Wyzerowanie tego bitu inicjuje normalną procedurę inicjalizacji wewnętrznych peryferiów, zaś ustawienie tego bitu umożliwi start procedury aktualizacji wewnętrznego oprogramowania układu,

- Stan tego bitu decyduje o źródle sygnału taktującego układ Si4713. Wyzerowanie tego bitu wymusza konieczność dostarczenia zewnętrznego sygnału taktującego (do wyprowadzenia RCLK), zaś ustawienie, uruchamia zintegrowany oscylator, który wymaga dołączenia do wyprowadzeń RCLK i GPIO3/DCLK rezonatora kwarcowego o częstotliwości 32768Hz (domyślnie, bez potrzeby dodatkowej konfiguracji) lub z zakresu 31.130 kHz...40 MHz (po dodatkowej konfiguracji wartości preskalertera i wartości częstotliwości wynikowej REFCLK).
- Wartość bitów FUNC[3...0] określa dodatkową funkcjonalność rozkazu POWER_UP, według następującej specyfikacji: FUNC[3...0]=0x02 → wymusza tryb nadajnika sygnału FM, FUNC[3...0]=0x0F → wymusza rozrzuconą odpowiedź układu Si4713 zawierającą (poza standardowym bajtem statusu) drobniejsze informacje nt. wersji oprogramowania, które to mogą być przydatne w procesie jego aktualizacji,
- Wartość bitów OPMODE[7...0] ma wpływ na rodzaj użytych wejść sygnału audio według następującej specyfikacji: OPMODE[7...0] =0x50 → aktywuje analogowe wejścia sygnału audio (LIN/RIN), OPMODE[7...0] =0x0F → aktywuje cyfrową magistralę wejściową dla sygnału audio (DIN/DFS/DCLK).

Rozkaz SET_PROPERTY (0x12)								
Bit	D7	D6	D5	D4	D3	D2	D1	D0
Rozkaz	0	0	0	1	0	0	1	0
Argument1	0	0	0	0	0	0	0	0
Argument2	PROPERTY (wybrana właściwość funkcjonalna) – bajt MSB							
Argument3	PROPERTY (wybrana właściwość funkcjonalna) – bajt LSB							
Argument4	VALUE (wartość dla wybranej właściwości) – bajt MSB							
Argument5	VALUE (wartość dla wybranej właściwości) – bajt LSB							

Rozkaz SET_PROPERTY pozwala na określenie nastaw funkcjonalno-sprzętowych układu Si4713. Znaczenie poszczególnych argumentów jest następujące:

- 16-bitowe słowo PROPERTY-MSB/PROPERTY-LSB określa predefiniowany rodzaj właściwości funkcjonalno-sprzętowej układu Si4713,
- 16-bitowe słowo VALUE-MSB/VALUE-LSB określa wartość dla przesłanej wcześniej właściwości funkcjonalno-sprzętowej układu Si4713.

Rozkaz TX_TUNE_FREQ (0x30)								
Bit	D7	D6	D5	D4	D3	D2	D1	D0
Rozkaz	0	0	1	1	0	0	0	0
Argument1	0	0	0	0	0	0	0	0
Argument2	Częstotliwość transmisji FM – bajt MSB							
Argument3	Częstotliwość transmisji FM – bajt LSB							

Rozkaz TX_TUNE_FREQ pozwala na przestrojenie nadajnika FM układu Si4713. 16-bitowe słowo określone argumentami Argument2/Argument3 wyznacza nową częstotliwość transmisji, przy czym dla przykładu: wartość 7610 odpowiada częstotliwości 76,10 MHz zaś wartość 10800, częstotliwości 108,00 MHz itd. Wysłaniu tegoż rozkazu sterującego powinno towarzyszyć następujące bezpośrednio po nim sekwencyjne odczytywanie rejestru statusu układu Si4713 (za pomocą rozkazu sterującego GET_INT_STATUS) w oczekiwaniu na ustawienie bitu STCINT sygnalizującego zakończenie operacji przestrojenia (jeśli nie uruchomiono dedykowanego przzerwania STC).

Rozkaz TX_TUNE_POWER (0x31)								
Bit	D7	D6	D5	D4	D3	D2	D1	D0
Rozkaz	0	0	1	1	0	0	0	1
Argument1	0	0	0	0	0	0	0	0
Argument2	0	0	0	0	0	0	0	0
Argument3	RFdBµV[7...0]							
Argument4	ANTCAP[7...0]							

Rozkaz TX_TUNE_POWER pozwala na określenie mocy transmisji nadajnika FM układu Si4713 oraz pojemności kondensatora w obwodzie strojenia anteny. Wysłaniu tegoż rozkazu sterującego powinno towarzyszyć następujące bezpośrednio po nim sekwencyjne odczytywanie rejestru

Listing 5. Zawartość pliku nagłówkowego sterownika układu Si4713

```
//Adresy układu Si4713, gdy
wyprowadzenie SEN=0
#define Si4713_WRITE_ADDR 0x22
#define Si4713_READ_ADDR 0x23
//Bity statusu układu
#define STCINT 0x01
#define ASQINT 0x02
#define RDSINT 0x04
#define RSQINT 0x08
#define ERR 0x40
#define CTS 0x80

//Używane komendy sterujące
i towarzyszące im argumenty
#define POWER_UP_CMD 0x01
#define USE_CRYSTAL 0x10
#define TX_MODE 0x02
#define ANALOG_AUDIO_INPÜTS 0x50
#define SET_PROPERTY_CMD 0x12
#define GET_INT_STATUS_CMD 0x14
#define TX_TUNE_FREQ_CMD 0x30
#define TX_TUNE_POWER_CMD 0x31
#define TX_RDS_PS_CMD 0x36
//Definicje właściwości (PROPERTY)
oraz dostępnej listy ich wartości
#define TX_COMPONENT_ENABLE 0x2100
#define TX_ENABLE_PILOT 0x0001
#define TX_ENABLE_LMR 0x0002
#define TX_ENABLE_RDS 0x0004
#define TX_LINE_INPUT_LEVEL 0x2104
#define MAX_INPUT_LEVEL_190 (0b00<<12)
#define MAX_INPUT_LEVEL_301 (0b01<<12)
#define MAX_INPUT_LEVEL_416 (0b10<<12)
#define MAX_INPUT_LEVEL_636 (0b11<<12)
#define TX_PREEMPHASIS 0x2106
#define PREEMPHASIS_75 0x00
#define PREEMPHASIS_50 0x01
#define TX_RDS_PS_MIX 0x2C02
#define RDS_PS0percent 0x00
#define RDS_PS125percent 0x01
#define RDS_PS250percent 0x02
#define RDS_PS500percent 0x03
#define RDS_PS750percent 0x04
#define RDS_PS875percent 0x05
#define RDS_PS100percent 0x06
#define TX_RDS_PS_REPEAT_COUNT 0x2C04
#define TX_RDS_PS_MESSAGE_COUNT 0x2C05
```

statusu układu Si4713 (za pomocą rozkazu sterującego GET_INT_STATUS) w oczekiwaniu na ustawienie bitu STCINT sygnalizującego zakończenie operacji zmiany mocy sygnału (jeśli nie uruchomiono dedykowanego przzerwania STC). Znaczenie poszczególnych argumentów jest następujące:

- Bajt RFdB µV[7...0] określa moc transmisji FM nadajnika układu Si4713 w krokach 1 dBµV (maksimum równe 115 dBµV),

Listing 6. Kod funkcji pozwalającej na ustawienie częstotliwości transmisji FM

```

/* Funkcja narzędziowa ustalająca częstotliwość transmisji FM: zakres 87.5...1080 odpowiada czest. 87.5-108 MHz */
void Si4711setTXfrequency(uint16_t Frequency)
{
    Arguments[0] = 0; //Zarezerwowane - zawsze 0
    Arguments[1] = (Frequency*10) >> 8;
    Arguments[2] = (Frequency*10) & 0x00FF;
    Si4711sendCommand(TX_TUNE_FREQ_CMD, Arguments, 3, NULL, 0); //Nie odczytujemy odpowiedzi
    //Czekamy na ustawienie bitu STCINT sygnalizującego zakończenie operacji przestrajania
    while (!(Si4711getIntStatus() & STCINT));
    //Wysyłam rozkaz TX_TUNE_STATUS, którego wykonanie powoduje skasowanie flagi STCINT
    Si4711clearSTCINTflag();
}
    
```

Listing 7. Kod funkcji pozwalającej na ustawienie mocy transmitowanego sygnału FM

```

vvoid Si4711setTXpower(uint8_t Power)
{
    Arguments[0] = 0; //Zarezerwowane - zawsze 0
    Arguments[1] = 0; //Zarezerwowane - zawsze 0
    Arguments[2] = Power; //Moc w dBuV (zakres 88...115)
    Arguments[3] = 0; //Pojemność anteny, 0: zostanie wyznaczona automatycznie przez układ Si4711
    Si4711sendCommand(TX_TUNE_POWER_CMD, Arguments, 4, NULL, 0); //Nie odczytujemy odpowiedzi
    //Czekamy na ustawienie bitu STCINT sygnalizującego zakończenie operacji zmiany mocy nadajnika
    while (!(Si4711getIntStatus() & STCINT));
    //Wysyłamy rozkaz TX_TUNE_STATUS, którego wykonanie powoduje skasowanie flagi STCINT
    Si4711clearSTCINTflag();
}
    
```

- Bajt ANTCAP[7...0] określa pojemność kondensatora w obwodzie strojenia anteny w krokach 0.25 pF (zakres 0...191). Ustawienie ANTCAP[7...0]=0x00 wymusza automatyczny dobór wspomnianej pojemności.

Rozkaz TX_RDS_PS (0x36)	D7	D6	D5	D4	D3	D2	D1	D0
Rozkaz	0	0	1	1	0	1	1	1
Argument1	0	0	0	PSID[4...0]				
Argument2	PSCHAR0[7...0]							
Argument3	PSCHAR1[7...0]							
Argument4	PSCHAR2[7...0]							
Argument5	PSCHAR3[7...0]							

Rozkaz *TX_RDS_PS* pozwala na definicję napisów emitowanych automatycznie przez transmitter FM jako wiadomość typu PS (Program Service) wbudowanego procesora systemu RDS/RDSB. Znaczenie poszczególnych argumentów jest następujące:

- Bity PSID[4...0] określają kolejny numer grupy 4 znaków wiadomości RDS typu PS według następującej specyfikacji:
 - Bity PSID[4...0]=0 → pierwsze 4 znaki pierwszej wiadomości typu PS,

- Bity PSID[4...0]=1 → ostatnie 4 znaki pierwszej wiadomości typu PS,
- Bity PSID[4...0]=2 → pierwsze 4 znaki drugiej wiadomości typu PS,
- Bity PSID[4...0]=3 → ostatnie 4 znaki drugiej wiadomości typu PS,
- Bity PSID[4...0]=22 → pierwsze 4 znaki dwunastej wiadomości typu PS,
- Bity PSID[4...0]=23 → ostatnie 4 znaki dwunastej wiadomości typu PS.

W ramach rozkazu *TX_RDS_PS* możliwe jest zdefiniowanie 12, 8-znakowych wiadomości RDS typu PS, które to będą wysyłane przez wbudowany w układ Si4713 procesor RDS sekwencyjnie.

- Bajty PSCHAR0[7...0]...PSCHAR3[7...0] określają kody ASCII kolejnych znaków poszczególnych wiadomości RDS typu PS, których to numer specyfikowany jest wartością bitów PSID[4...0].

Właściwość TX_COMPONENT_ENABLE (0x2100)									
Bit	D15	D14	D13	D12	D11	D10	D9	D8	D7
Symbol	0	0	0	0	0	0	0	0	0
Bit	D6	D5	D4	D3	D2	D1	D0		
Symbol	0	0	0	0	RDS	LMR	PILOT		

Właściwość *TX_COMPONENT_ENABLE* pozwala określić, które z komponentów sygnału FM znajdują się w widmie transmitowanego sygnału radiowego. Znaczenie poszczególnych bitów jest następujące:

- RDS: ustawienie tego bitu aktywuje nadawanie sygnału RDS w widmie transmitowanego sygnału FM,
- LMR: ustawienie tego bitu aktywuje nadawanie składowej L-R (niezbędnej dla demodulacji Stereo) w widmie transmitowanego sygnału FM,
- PILOT: ustawienie tego bitu aktywuje nadawanie tzw. pilota sygnału Stereo w widmie transmitowanego sygnału FM.

Właściwość TX_LINE_INPUT_LEVEL (0x2104)										
Bit	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6
Symbol	0	0	LIATTEN[1...0]			0	0	LILEVEL[9...0]		
Bit	D5	D4	D3	D2	D1	D0				
Symbol	LILEVEL[9...0]									

Właściwość *TX_LINE_INPUT_LEVEL* pozwala na określenie, maksymalny poziom wejściowego sygnału audio, by w sposób optymalny wykorzystać charakterystykę modulacji FM, co z kolei przekłada się na wynikową jakość sygnału audio po stronie odbiornika radiowego (odstęp sygnału od szumu i inne). Znaczenie poszczególnych bitów przedstawia się następująco:

- LIATTEN[1...0]: wartość tych bitów określa maksymalną, dopuszczalną wielkość wejściowego sygnału audio oraz impedancję wejściową według następującej specyfikacji: LIATTEN[1...0]=0x00 → 190 mV/396 kΩ, LIATTEN[1...0]=0x01 → 301 mV/100 kΩ, LIATTEN[1...0]=0x02 → 416 mV/74 kΩ, LIATTEN[1...0]=0x03 → 636 mV/60 kΩ,

Listing 8. Kod funkcji pozwalającej na ustawienie tekstu będącego nazwą Program Service

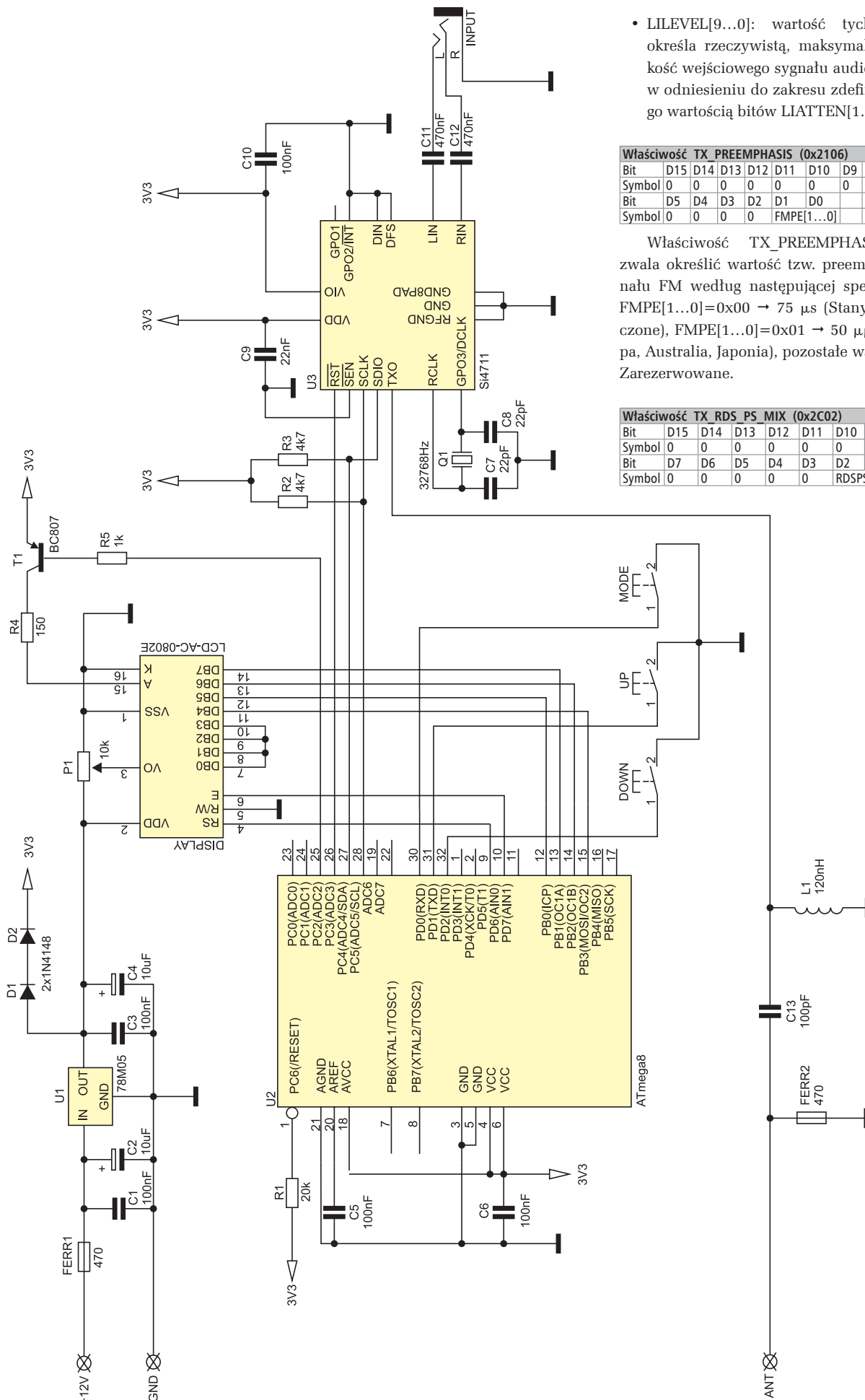
```

/* Funkcja pozwala na definicję tekstu będącego nazwą Program Service: maks. 12 tablic znaków po 8 znaków w każdej (bez terminatora) */
void Si4711setPStext(uint8_t Nr, char *Text)
{
    Arguments[0] = Nr<<1;
    Arguments[1] = Text[0];
    Arguments[2] = Text[1];
    Arguments[3] = Text[2];
    Arguments[4] = Text[3];
    Si4711sendCommand(TX_RDS_PS_CMD, Arguments, 5, NULL, 0);
    Arguments[0] = (Nr<<1)+1;
    Arguments[1] = Text[4];
    Arguments[2] = Text[5];
    Arguments[3] = Text[6];
    Arguments[4] = Text[7];
    Si4711sendCommand(TX_RDS_PS_CMD, Arguments, 5, NULL, 0);
}
    
```

Listing 9. Kod funkcji pozwalającej na ustawienie liczby aktywnych wiadomości typu Program Service

```

/* Funkcja ustawia liczbę aktywnych wiadomości typu Program Service (PS), zakres: 0...11 (dopuszczalnych 12 wiadomości typu PS) */
void Si4711setPSmessageCount(uint8_t Messages)
{
    Si4711setProperty(TX_RDS_PS_MESSAGE_COUNT, Messages);
}
    
```



- LILEVEL[9...0]: wartość tych bitów określa rzeczywistą, maksymalną wielkość wejściowego sygnału audio [w mV] w odniesieniu do zakresu zdefiniowanego wartością bitów LIATTEN[1...0].

Właściwość TX_PREAMPHASIS (0x2106)										
Bit	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6
Symbol	0	0	0	0	0	0	0	0	0	0
Bit	D5	D4	D3	D2	D1	D0				
Symbol	0	0	0	0	0	FMPE[1...0]				

Właściwość TX_PREAMPHASIS pozwala określić wartość tzw. preemfazy sygnału FM według następującej specyfikacji: FMPE[1...0]=0x00 → 75 μs (Stany Zjednoczone), FMPE[1...0]=0x01 → 50 μs (Europa, Australia, Japonia), pozostałe wartości → Zarezerwowane.

Właściwość TX_RDS_PS_MIX (0x2C02)										
Bit	D15	D14	D13	D12	D11	D10	D9	D8		
Symbol	0	0	0	0	0	0	0	0		
Bit	D7	D6	D5	D4	D3	D2	D1	D0		
Symbol	0	0	0	0	0	0	RDSPSMIX[2...0]			

Rysunek 2. Schemat ideowy nadajnika transmitterFM

Właściwość $TX_RDS_PS_MIX$ pozwala na określenie, jaką część emisji danych RDS zajmie transmisja wiadomości typu PS (Program Service). Wartość bitów $RDS_PS_MIX[2...0]$ określa procentowy udział emisji danych PS według następującej specyfikacji: $RDS_PS_MIX[2...0]=0x01 \rightarrow 12.5\%$, $RDS_PS_MIX[2...0]=0x02 \rightarrow 25\%$, $RDS_PS_MIX[2...0]=0x03 \rightarrow 50\%$, $RDS_PS_MIX[2...0]=0x04 \rightarrow 75\%$, $RDS_PS_MIX[2...0]=0x05 \rightarrow 87.5\%$, $RDS_PS_MIX[2...0]=0x06 \rightarrow 100\%$.

Budowa

Schemat ideowy transmitteraFM pokazano na **rysunku 2**. Jest to system mikroprocesorowy zbudowany z użyciem popularnego mikrokontrolera ATmega8 taktowanego wewnętrznym oscylatorem o częstotliwości 8 MHz. Zadaniem mikrokontrolera jest obsługa układu Si4713 za pomocą interfejsu TWI oraz realizacja interfejsu użytkownika zbudowanego z wyświetlacza alfanumerycznego o organizacji 2×8 znaków oraz z 3 przycisków funkcyjnych. Ponadto, urządzenie wyposażono w stabilizator liniowy 78M05 zapewniający napięcie zasilania dla wyświetlacza LCD (+5 V) oraz, poprzez dwie diody 1N4148, napięcie zasilające mikrokontroler i układ Si4713 (ok. 3,4 V). Jest to najprostsze i jednocześnie dopuszczalne rozwiązanie w przypadku, gdy system mikroprocesorowy jest zasilany napięciem 3,3 V a wyświetlacz napięciem 5 V. Wyłącznie w takim przypadku, gdy nie korzysta się z operacji odczytu danych z wyświetlacza (wyprowadzenie R/W podłączone na stałe do masy).

Montaż

Schemat montażowy transmitteraFM pokazano na **rysunku 3**. Całość zmontowano na niewielkiej płytce drukowanej o wymiarach zastosowanego wyświetlacza LCD, ze zdecydowaną przewagą elementów do montażu powierzchniowego, rozmieszczonych – co należy szczególnie podkreślić – po obu stronach płytki. Z uwagi na zastosowanie niewielkich elementów SMD w dość „kłopotliwych” obudowach, montaż układu tego typu najlepiej jest przeprowadzić z użyciem stacji lutowniczej na gorące powietrze, odpowiedniej jakości topników lutowniczych oraz dysponując sporym doświadczeniem w tej kwestii. Dotyczy to zwłaszcza układu scalonego Si4713, którego niewielka obudowa o wymiarach $3 \text{ mm} \times 3 \text{ mm}$ ma 20 wyprowadzeń umieszczonych pod spodem i na obrysie obudowy. Montaż tego układu należy wykonać szczególnie ostrożnie, gdyż praktyka (jak i informacje zgłaszane przez Czytelników) wykazała, że te elementy są bardzo wrażliwe na przegrzanie i elektryczność statyczną, a ich uszkodzenia są dość trudne do wykrycia (np. układ może funkcjonować „prawie” poprawnie poza tym, że nie jest w stanie przestroić wbudowanego

tunera FM, zatrzymując w tym momencie program obsługi urządzenia) sugerując częstokroć inne przyczyny braku poprawnego działania urządzenia.

Wracając jednak do montażu. Rozpoczynamy od przyłutowania wszystkich układów scalonych. Następnie lutujemy diody, rezystory, kondensatory, pozostałe elementy bierne, a na końcu złącze goldpin przeznaczone dla wyświetlacza LCD, gniazdo słuchawkowe oraz mikroprzetwózniki.

Poprawnie zmontowany układ powinien działać od razu po podłączeniu zasilania. Jako antenę należy zastosować kawałek nieekranowanego przewodu podłączonego do punktu lutowniczego na płytce drukowanej oznaczonego jako ANT. Co prawda, producent układu zaleca wykorzystanie jednej z żył przewodu zasilającego (dokładnie masy) jako medium transmisyjnego (tym samym wymuszenie połączenia wyprowadzenia antenowego do masy układu), jednak testy praktyczne wykazały, iż znacznie lepsza propagację sygnału zapewnia zastosowanie oddzielnego kawałka przewodu odseparowanego od masy urządzenia.

Obsługa

Zaprojektowany transmitterFM jest bardzo prostym systemem mikroprocesorowym, o niewielkiej liczbie opcji Menu, zaprojektowano minimalistyczny i przejrzysty interfejs użytkownika. Urządzenie ma dwa tryby pracy: *Przeglądanie* i *Edycja*. Do zmiany trybów pracy służy przycisk umownie oznaczony *Mode*. W trybie *Przeglądanie* możemy zapoznać się z dostępnymi pozycjami menu urządzenia i ustawieniami transmittera dla każdej ze wspomnianych opcji. Do poruszania się po liście pozycji menu służą przyciski umownie oznaczone *UP* i *DOWN*. W trybie *Edycja*, co łatwo się domyślić, mamy możliwość zmiany wartości interesujących nas parametrów urządzenia. Edytowane parametry są wyróżniane (w drugim wierszu wyświetlacza), zaś ich zmiany są możliwe za pomocą *UP* i *DOWN*. Wyjątkowo, dla trybów edycji tekstów transmitowanych w postaci wiadomości RDS typu PS (PStext1...PStext4), krótkie przyciśnięcie wybranego przycisku powoduje zmianę bieżąco edytowanego znaku tekstu (z zakresu ASCII 32...122), zaś długie – przejście do kolejnego/poprzedniego znaku bieżącego tekstu. Dla pozostałych pozycji menu, przyciski *UP* i *DOWN* realizują pojedyncze funkcje służące wyłącznie regulacji wskazywanego parametru.

Każdorazowe opuszczenie trybu edycji, inicjowane ponownym przyciśnięciem przycisku *MODE*, powoduje po pierwsze, wysłanie bieżących ustawień do układu Si4713, natomiast po drugie, zapisanie ich w nieulotnej pamięci EEPROM mikrokontrolera, by ustawienia te stały się aktywne po ponownym włączeniu zasilania. Wygląd

Wykaz elementów

Rezystory: (SMD 0603)

R1: 20 k Ω

R2, R3: 4,7 k Ω

R4: 150 Ω

R5: 1 k Ω

P1: 10 k Ω (potencjometr montażowy)

Kondensatory: (SMD 0603)

C1, C3: 100 nF (SMD 1206, ceram. X5R)

C5, C6, C10: 100 nF (ceram. X5R)

C2, C4: 10 $\mu\text{F}/16\text{V}$ (SMD „B”, tantal., EIA 3528-21R)

C7, C8: 22 pF

C9: 22 nF

C11, C12: 470 nF

C13: 100 pF

Półprzewodniki:

U1: 78M05 (DPAK)

U2: ATmega8 (TQFP32)

U3: Si4713 (QFN20)

T1: BC807 (SOT23)

D1...D2 – 1N4148 (SOD80C)

Inne:

DISPLAY: wyświetlacz AC-0802E-MIG G/K-E6 C PBF

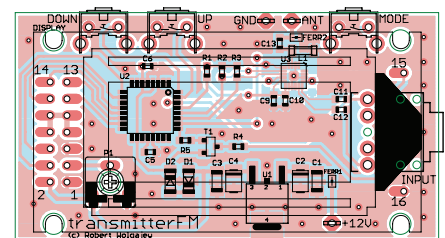
Q1: rezonator kwarcowy, zegarkowy 32768 Hz

INPUT: gniazdo Jack 3,5 mm do montażu przewlekanego

MODE, UP, DOWN: microswitch kątowy 90° do montażu przewlekanego

L1: dławik 120 nH (SMD 1206)

FERR1, FERR2: koralik ferrytowy MURATA BLM18PG471SN1J 470 Ω @100 MHz, 1 A (SMD 0603)



Rysunek 3. Schemat montażowy nadajnika transmitterFM

wszystkich, dostępnych pozycji menu pokazano na **rysunku 4**.

Robert Wołgajew, EP

I would like to thank Mr. Youqing Liu from SiLabs's Support Team for his invaluable support during the startup device.

freq
106.5MHz

Częstotliwość transmisji

Power
95dBuV

Moc nadajnika FM

Audio

Maksymalny poziom wejściowego sygnału audio

Pstext1
denoText

Treść wiadomości RDS typu PS (dla 4 pozycji Menu)

Rysunek 4. Struktura menu nadajnika transmitterFM

