

Sprzętowe stosy IrDA firmy Microchip, część 1

Implementacja nowoczesnych interfejsów przesyłu danych w systemach wbudowanych, do niedawna oznaczała konieczność używania rozbudowanych platform sprzętowych zdolnych pomieścić kompletne stosy programowe obsługujące transmisję danych. Użycie w tym celu małych kontrolerów 8-bitowych oznaczało tyle, iż układ taki stawał się jedynie kontrolerem pożądanego interfejsu, niezdolnym (z uwagi na złożoność protokołów poszczególnych warstw) do wykonywania zadań pierwotnie przypisanych projektowanemu urządzeniu.

Konstruktor stawał wówczas przed dylematem – co dalej? Sięgać po kolejny 8-bitowy procesor umieszczony tuż obok omówionego? Czy może pójść krok dalej? Sięgnąć po „większy, szerszy” kontroler, ale czy uda się w krótkim czasie opanować specyfikę jego używania? Co z narzędziami? Czy są darmowe kompilatory? A jaki programator jest wymagany...?

Tego typu dylematy ukazujące się konstruktorowi powodowały, że chęć wdrożenia nowego pomysłu z najnowszymi rozwiązaniami była skutecznie tłumiona. Pozostało zaprojektować urządzenie (kolejne) komunikujące się ze światem zewnętrznym poprzez RS232 (skądinąd genialny w swej prostocie). I nici z satysfakcji. Obecnie nie jest tak źle. Producenci półprzewodników wydają się dostrzegać potrzeby „zwykłych” pasjonatów, ludzi dla których elektronika to całe życie, a nie tylko sucha profesja. W przypadku USB przykładów jest wiele: sztandarowy FTDI, Texas Instruments z serią układów TUSB (np. TUSB3410 – świetny przykład zaangażowania całego 8051 w obsługę stosu USB), Cypress. Jeśli ktoś chciałby wyposa-

żyć swoje urządzenie w transceiver radiowy w standardzie ZigBee, z pomocą przychodzi Chipcon czy RadioCrafts. Ethernet: moduły Digi, Tibbo, Moxa (Mosty Ethernet <-> UART). Microchip natomiast, zajął się dość zapomnianą/niedostrzeganą metodą komunikacji bezprzewodowej – podświetleniem. Transceivery pracujące w standardzie IrDA są obecne niemal we wszystkich telefonach komórkowych oraz laptopach. Są tańsze w implementacji niż Bluetooth, a także wydają się być łatwiejsze w użyciu. Microchip produkuje pięć układów rodziny MCP21xx, różniących się funkcjonalnością oraz głębokością stosu IrDA w nich zaimplementowanego.

Budowa układów MCP212x

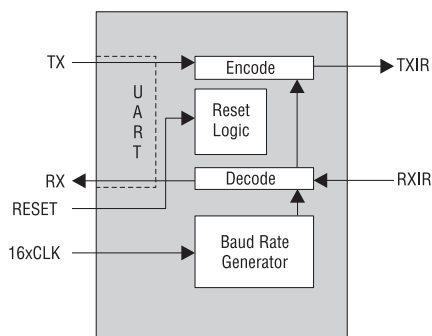
Najmniej złożonymi układami są MCP2120 i MCP2122 (ich podstawowe parametry podano w **tab. 1** i **2**) Obydwa zawierają w swojej strukturze samodzielny enkoder/dekoder zgodny ze standardem IrDA (**rys. 1**). Układy wyposażone są również w UART oraz interfejs pozwalający na bezpośrednie podłączenie optycznych transceiverów IrDA.

MCP212x spełnia rolę tłumacza warstwy fizycznej IrDA na UART, co oznacza, że nie ingeruje on w zawartość przesyłanych informacji, a jedynie zmienia kształt impulsów elektrycznych. Zatem każdemu bitowi obecnemu po stronie UART-a odpowiada bit (przebieg elektryczny) zgodny z IrDA. Typowo układ pracuje jako driver optycznych transceiverów podświetlenia (np. TFDS4500) (**rys. 2**)

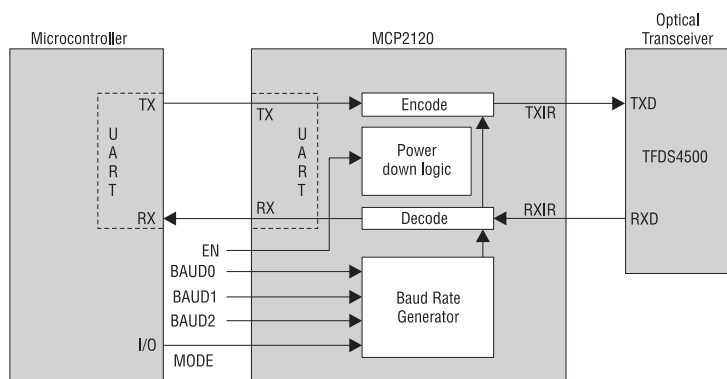
W tym miejscu warto omówić podstawowe różnice pomiędzy układami 2120 i 2122. MCP2120 jest samowystarczającym, rekonfigurowanym sprzętowo, bądź programowo enkoderem/dekoderem. Pozwala użytkownikowi na wybranie właściwej prędkości transmisji poprzez dostarczenie odpowiednich sygnałów logicznych do wyprowadzeń BAUD0 do BAUD2. Jest to tak zwana sprzętowa selekcja

Tab. 1. Cechy układu MCP2120

- Wsparcie dla warstwy fizycznej standardu IrDA (v 1.3)
- UART <-> IR enkoder/dekoder
 - kompatybilny z optycznymi transceiverami przeznaczonymi dla IrDA
 - kompatybilny ze wszystkimi rodzajami UART
- Wspierany format transmisji (tzw. szerokość bitu) – 1,63 μs
- Sprzętowo lub programowo wybór pożądanego prędkości transmisji:
 - max 115,2 kbaud dla aplikacji zgodnych z IrDA – max 312,5 kbaud przy zastosowaniu oscylatora 20 MHz
- Praca z niskimi napięciami zasilania
- Dostępny dla standardowych i przemysłowych zakresów temperatur
- Pobór prądu:
 - <1 mA przy 3,3 V, 8 MHz
 - <3 μA w trybie czuwania dla 5 V



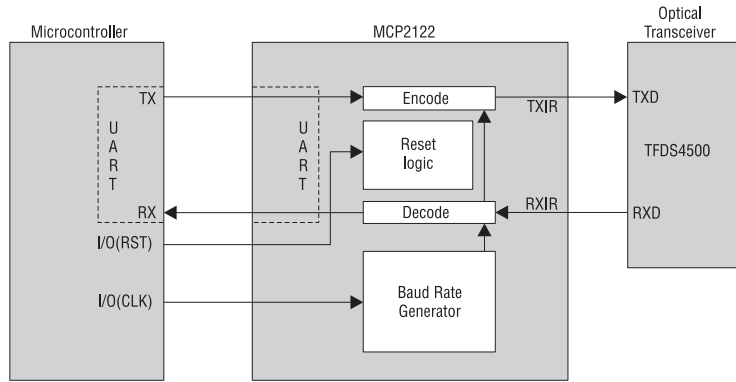
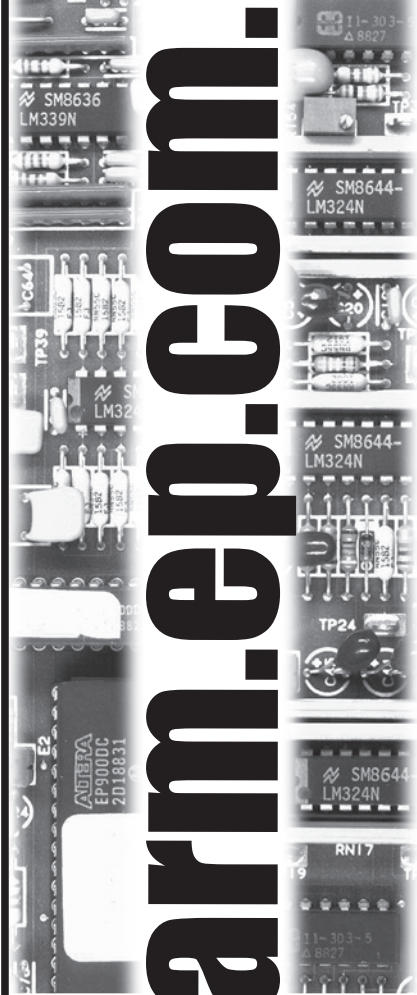
Rys. 1. Schemat blokowy układu enkodera/dekodera MCP2122



Rys. 2. Schemat blokowy typowej aplikacji układu MCP2120

Zapraszamy!

www.arm.ep.com.pl

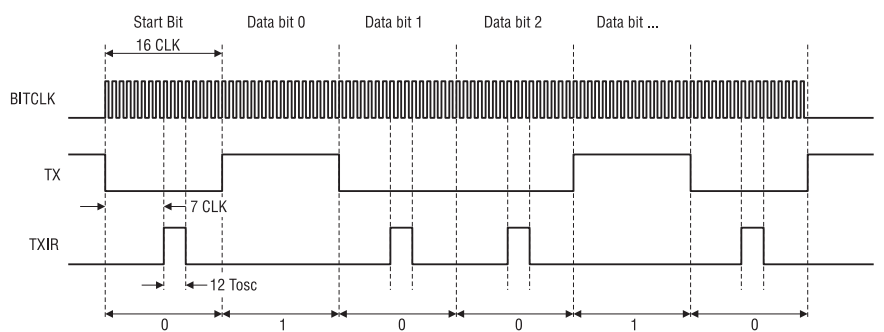


Rys. 3. Schemat aplikacyjny układu MCP2122

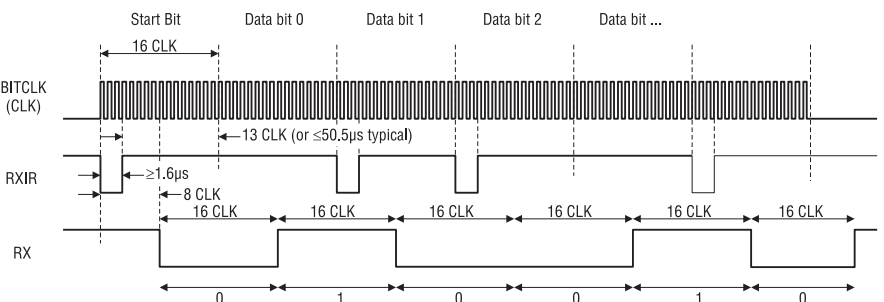
prędkości bodowej UART-a oraz interfejsu IrDA. Ponadto układ został wyposażony w wyprowadzenie MODE, pozwalające na jego rekonfigurację (zmianę prędkości pracy) z poziomu mikrokontrolera z nim współpracującego. Jeśli wybierzemy ten tryb pracy, MCP2120 będzie oczekiwał na swoim porcie UART komend ustalających nową prędkość transmisji. Istnieje zatem możliwość zaimplementowania własnego protokołu transmisji, który w zależności od jakości łącza optycznego (którą to można ocenić np. po ilości błędnych transmisji), będzie zmieniał prędkość pracy układu (co zostało również zaimplementowane w wyższych warstwach stosu IrDA). Do poprawnej pracy układu wymagane jest dołączenie oscylatora kwarcowego, bądź doprowadzenie zewnętrznego sygnału zegarowego.

MCP2122 jest uproszczoną wersją 2120. Z założenia układ miał

być kompatybilny pod względem wyprowadzeń i funkcjonalności z enkoderem/dekoderem HSDL-7000 (Agilent). Do poprawnej pracy bezwzględnie wymaga doprowadzenia sygnału taktującego o częstotliwości 16-krotnie większej niż pożądana prędkość transmisji, co umożliwia zaimplementowanie (w przypadku dostarczenia tego sygnału z wyprowadzenia mikrokontrolera) podobnego mechanizmu dynamicznej zmiany prędkości transmisji, jak wspomniany wyżej. Z punktu widzenia konstruktora jest to rozwiązanie dość niewygodne. Zaletą MCP2122 jest natomiast mniejsza obudowa, mniejsza liczba wyprowadzeń (tylko 8) oraz mniejszy pobór energii. Typowy schemat aplikacyjny przedstawiono na rys. 3. Układ może być wprowadzony w stan czuwania poprzez wymuszenie stanu niskiego na wyprowadzeniu RESET.



Rys. 4. Proces modulacji w układach MCP212x



Rys. 5. Proces demulacji w układach MCP212x

Tab. 2. Cechy układu MCP2122

- Wyprowadzenia kompatybilne z HSDL-7000
- Wsparcie dla warstwy fizycznej standardu IrDA (v 1.3)
- UART <-> IR enkoder/dekoder
 - kompatybilny z optycznymi transceiverami przeznaczonymi dla IrDA
 - kompatybilny ze wszystkimi rodzajami UART
- Prędkość transmisji:
 - max 115,2 kbaud dla transmisji zgodnej z IrDA
- Wspierany format transmisji (tzw. szerokość bitu) – 1,63 μ s
- Tryb czuwania (2 μ A przy 1,8 V)
- Praca z niskimi napięciami zasilania
- Rozszerzony zakres temperatur
- Niski pobór energii

Modulacja/demodulacja w układach MCP21xx

Obydwa układy pracują w trybie half-duplex, a więc w każdym momencie komunikacja odbywa się tylko w jedną stronę (zarówno po stronie UART-a, jak i podczerwieni). Kodowanie (odpowiedniejszym określeniem byłaby modulacja) bitu, który pojawił się na wejściu TX UART-a układu MCP212x przebiega następująco: Jeśli bit ten ma wartość logicznego zera, na wyjściu TXIR układu zostanie ustawiony stan niski. Sytuacja taka potrwa przez 7 cykli zegarowych, po których na trzy cykle zegarowe zostanie ustawiony stan wysoki. Po tych 10 cyklach na wyjściu TXIR ponownie zostanie ustawiony stan niski na 6 cykli zegara. Jeśli z kolei na wejściu TX pojawi się logiczna jedynka, wówczas wyjście TXIR pozostanie w stanie niskim na 16 cykli zegarowych. Ponieważ obsługa każdego bitu pojawiającego się na wejściu wymaga 16 cykli zegarowych, to obliczając prędkość zewnętrznego zegara (w przypadku nie używania oscylatora w MCP2120 i zawsze dla MCP2122), wystarczy pomnożyć wymaganą prędkość bodową przez 16 (wejście zewnętrznego sygnału zegarowego oznaczono 16XCLK)

Dekodowanie, czy raczej demodulacja informacji pojawiającej się na wejściu RXIR przebiega oczywiście w sposób analogiczny. Jeśli odebrany bit to logiczne zero, wówczas na wejściu RXIR zapanuje ten stan przez trzy pierwsze cykle zegarowe, a następnie na wejściu tym pojawi się stan wysoki na pozostały czas 13 cykli zegara, wówczas UART nada logiczne zero. Jeśli natomiast zostanie odebrana logiczna jedynka, na wejściu RXIR zaobserwujemy stan wysoki, który utrzyma się przez 16 cykli zegara. Oczywiście po stronie UART-a pojawi się odpowiedni stan

na całe 16 cykli zegarowych. Odpowiednie przebiegi czasowe ilustrujące omówione sytuacje zostały przedstawione na rys. 4 i 5.

Zastosowanie

Obydwa układy zostały stworzone z myślą o tych aplikacjach, w których nie jest wymagana transmisja w podczerwieni zgodna z protokołami wyższych warstw stosu IrDA, bądź projektowany system posiada wystarczająco dużo zasobów sprzętowych pozwalających na ich programową implementację. Układy idealnie nadają się do zastosowania w systemach zdalnego sterowania, gdzie wymagana jest komunikacja w obu kierunkach (np. dynamiczna zmiana kodu pilota). Innym przykładem zastosowania może być sieć bezprzewodowych czujników temperatury. Chcąc zbierać taką informację np. z hal produkcyjnych można zaimplementować mechanizm przekazywania pakietów (*repeating*) do kolejnych węzłów, aż do węzła dołączonego np. do komputera zapisującego te informacje. W takim przypadku nie jesteśmy ograniczeni niewielkim zasięgiem podczerwieni (kolejne stacje oddalone od siebie o 2-3 m mogą przekazywać informację na dużo większe odległości). Wadą jest oczywiście konieczność zapewnienia widoczności poszczególnych węzłów. Niewątpliwą zaletą omawianych układów jest wspomniany stan czuwania, który ułatwia stosowanie zasilania bateryjnego elementów projektowanego systemu. W podobny tryb wyposażone są też optyczne transceivery dołączane do MCP212x. Przy pomocy 2120 można z powodzeniem zbudować analizator protokołów wyższych warstw. Jeśli skupilibyśmy całą inteligencję takiego urządzenia w oprogramowaniu pracującym na komputerze PC, wówczas do realizacji tego pomysłu potrzebujemy jedynie układu MCP2120, transceivera optycznego np. TFDS4500 oraz układu TTL <-> RS232 (MAX232), bądź np. FTDI232BM (dla komunikacji przy pomocy USB). Możliwości jest wiele. W kolejnej części artykułu zostaną omówione kontrolery MCP2140, 2150 i 2155.

Marcin Chruściel, EP
marcin.chrusciel@ep.com.pl

Literatura

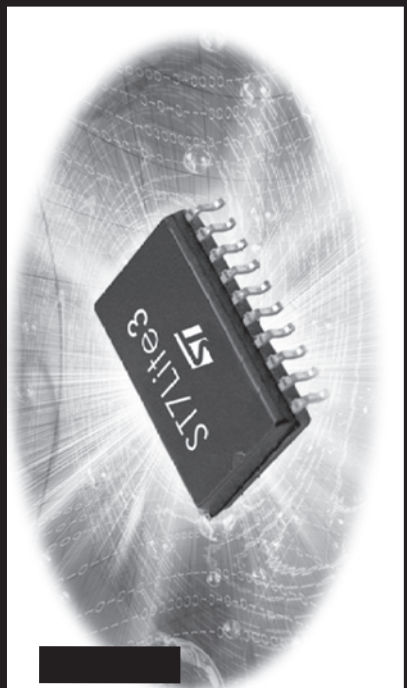
www.microchip.pl

noty katalogowe układów MCP212x:

[1] 21618a.pdf

[2] 21894b.pdf

Zapraszamy
już wkrótce!



>>>st7.ep.com.pl

Dostępne biblioteki SCH/PCB
mikrokontrolerów ST7LITE dla
Protela 99SE/DXP/DXP2004

