

Technika Mikroprocesorowa

Laboratorium 10

Realizacja dwukierunkowej transmisji szeregowej za pomocą portu szeregowego UART

Cel ćwiczenia: Celem ćwiczenia jest napisanie programu realizującego odbiór i transmisję strumieni danych z/do urządzeń zewnętrznych np. PC, modem itp.

W tym celu należy wykonać następujące zadania:

- skonfigurować port szeregowy,
- skonfigurować licznik taktujący UART – wybieramy licznik T2,
- skonfigurować kontroler przerwania i zainicjować wektor przerwania
- napisać procedury obsługujące przerwanie od portu szeregowego od nadawania i odbioru.

UWAGI Proszę pamiętać że program składa się z fragmentów kodu:

- inicjacja
- pętla główna

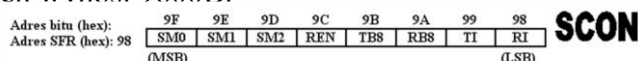
oraz

- procedury do odbioru danych wywoływanej przez przerwanie od odbiornika UART. Procedura odbioru będzie wywoływana przerwaniem od odbiornika.
- procedury do nadawania danych wywoływanej przez przerwanie od nadajnika UART. Procedura nadawania będzie wywoływana dwojako:
 - przerwaniem od nadajnika
 - by nadajnik wygenerował przerwanie musi nadać znak a więc pierwsze nadawanie musi być zrealizowane poprzez CALL.

W części inicjacyjnej wykonujemy następujące czynności

ad a) Konfiguracja portu szeregowego

Proszę skonfigurować port szeregowy do transmisji szeregowej 8 bitowej, asynchronicznej, prędkość transmisji ustawiana programowo. Prędkość transmisji niech wynosi 9600B.



Rys 2.18. Słowo sterujące SCON

- SCON.0 (RI) – znacznik odebrania znaku; zgłoszenie przerwania
- SCON.1 (TI) – znacznik wysłania (zakończenia transmisji) znaku; zgłoszenie przerwania;
- SCON.2(RB8) – dziewiąty bit odebranego znaku;
- SCON.3 (TB8) – dziewiąty bit znaku wysłanego w trybie 2 i 3;
- SCON.4 (REN) – uaktywnienie odbiornika transmisji szeregowej;
- SCON.5 (SM2) – znacznik maskowania odbioru transmisji;
- SCON.6 (SM1)
- SCON.7 (SM0) – uaktywnienie trybu pracy (tabl.2.2).

TABLICA 2.2 Tryby pracy portu szeregowego

Tryb	SM0	SM1	Opis
0	0	0	Transmisja szeregową synchroniczną: znaki 8-bitowe taktowane sygnałem zegarowym, (do dołączenia rejestru przesuwającego)
1	0	1	Transmisja szeregową asynchroniczną: znaki 8-bitowe, szybkość określona programowo
2	1	0	Transmisja szeregową asynchroniczną: znaki 9-bitowe, szybkość 1/32 lub 1/64 częstotliwości zegara
3	1	1	Transmisja szeregową asynchroniczną: znaki 9-bitowe, szybkość określona programowo

ad b) Konfiguracja licznika T2

Taktowanie portu UART zrealizować w oparciu o T2. Ponieważ port szeregowy UART jest już znany i podstawowa obsługa tego portu jest opanowana więc w tym ćwiczeniu należy na podstawie poniższych informacji samemu wybrać tryb pracy portu i odpowiednio go skonfigurować. Do sterowania licznika T2 służą słowa **T2MOD (C9H)** i **T2CON (C8H)** (wystarczy skonfigurować **T2CON**) impulsy zliczane są w rejestrach.

- TH2 (CDH)** – Bardziej znaczący bajt licznika T2
- TL2 (CCH)** – Mniej znaczący bajt licznika T2

rejstry przeładowania

- RCAP2H(CBH)**
- RCAP2L(CAH)**

Asembler „nie rozumie” nazw TH2, TL2 itp. bo nie są predefiniowane w takim przypadku do konfiguracji licznika można zastosować dwa rozwiązania

- użyć adresowania bezpośredniego komórek – np. MOV 0CCH, #001H
- zdefiniować potrzebny symbol w następujący sposób: TH2 SET 0CDH a potem go używać np. MOV T2H, #0FH. Analogicznie postępować z pozostałymi rejestrami TL2, RCAP2H, RCAP2L

Asembler będzie zgłaszał komunikaty w postaci ostrzeżeń – należy je zignorować.

Dane potrzebne do konfiguracji T2 zamieszczono poniżej
Tryby pracy licznika T2:

- capture,
- autoreload
- baud rate generator

Tryby pracy ustawiamy zgodnie z tabelką (x – oznacza dowolną wartość)

RCLK + TCLK	CP/RL2	TR2	Mode
0	0	1	16-bit auto-reload
0	1	1	16-bit capture
1	X	1	baud rate generator
X	X	0	(off)

Opis rejestru T2CON

Bit	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
Nr bitu	7	6	5	4	3	2	1	0

Bit	Funkcja
TF2	Znacznik (flaga) przepełnienia ustawiany przez T2, zerowany programowo. Jeżeli RCLK = 1 lub TCLK = 1 to nie jest ustawiany.
EXF2	Znacznik opadającego zbocza sygnału na wejściu T2EX gdy EXEN2=1, zerowany programowo. Nie generuje przerwania gdy DCEN=1.
RCLK	Przypisanie licznika T2 do sygnału zegarowego dla odbiornika portu szeregowego RCLK=1 w trybach pracy 1 i 3. RCLK=0 przypisanie licznika T1.
TCLK	Przypisanie licznika T2 do sygnału zegarowego dla nadajnika portu szeregowego TCLK=1 w trybach pracy 1 i 3. TCLK=0 przypisanie licznika T1.
EXEN2	Uaktywnienie wejścia wyzwalającego T2EX, reakcja na opadające zbocze EXEN2=1 (gdy T2 nie jest używany jako zegar taktujący dla portu szeregowego). EXEN2=0 licznik T2 ignoruje zdarzenia na wejściu T2EX.
TR2	Start/stop licznika T2. TR2=1 uruchamia zliczanie.
C/T2	Przełączanie między funkcjami licznika i czasomierza. C/T2=0 czasomierz, C/T2=1 - licznik.
CP/RL2	Ustawienie trybu pracy licznika. CP/RL2=1 praca z zatrzymywaniem zawartości licznika. CP/RL2=0 praca z automatycznym wpisaniem wartości początkowej.

Opis rejestru T2MOD

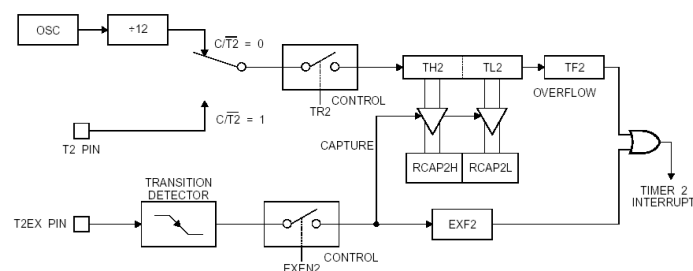
Nazwa bitu	---	---	---	---	---	---	T2OE	DCEN
Nr bitu	7	6	5	4	3	2	1	0

Bit	Funkcja
T2OE	Ustawienie wyjścia T2. T2OE=1 sygnał zegarowy na T2 (P1.0).
DCEN	Zezwolenie na zliczanie w dół i/lub w górę, DCEN=1.

Tryb pracy **capture**,

W tym trybie licznik/czasomierz zlicza impulsy z oscylatora lub z pinu T2. Bieżącą wartość można zatrzasnąć przy pomocy opadającego zbocza na T2EX. Przy przepełnieniu ustawiany jest znacznik TF2 i może zostać wygenerowane przerwanie.

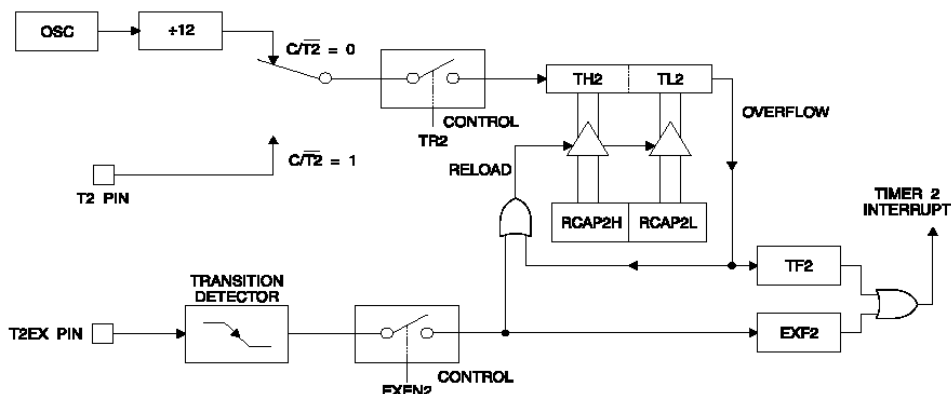
Figure 1. Timer 2 in Capture Mode



Tryb pracy autoreload

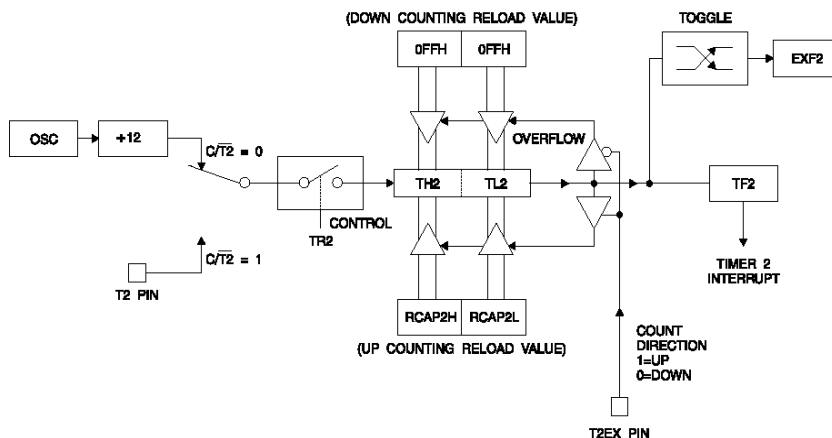
W tym trybie licznik działa z automatycznym przeładowaniem po przepełnieniu. Przy przepełnieniu ustawiany jest znacznik TF2 i może zostać wygenerowane przerwanie.

Figure 2. Timer 2 in Auto Reload Mode (DCEN = 0)



W tym trybie licznik działa z automatycznym przeładowaniem po przepełnieniu lub wyzerowaniu. Kierunek zliczania jest ustawiany przez wartość na pinie T2EX. Przy przepełnieniu lub wyzerowaniu ustawiany jest znacznik TF2 i może zostać wygenerowane przerwanie.

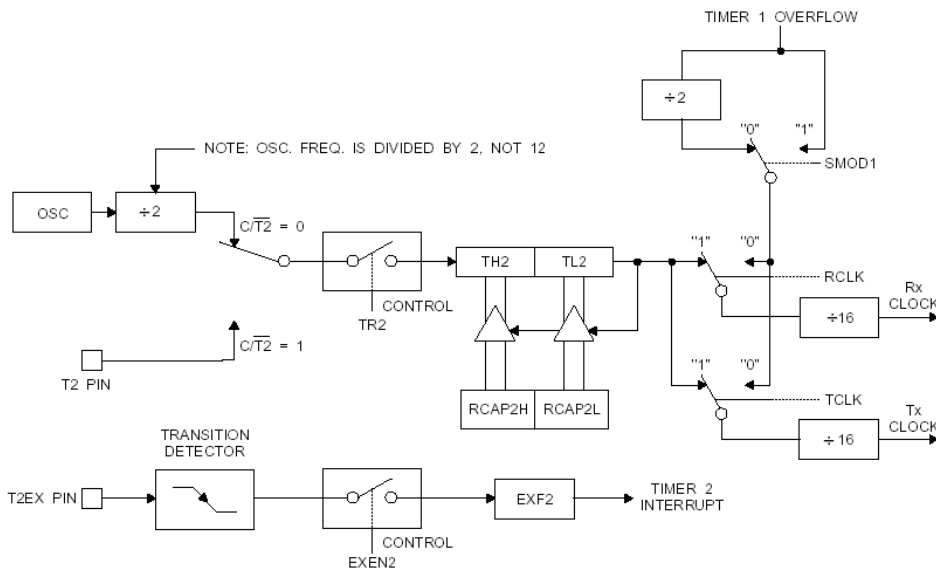
Figure 3. Timer 2 Auto Reload Mode (DCEN = 1)



Tryb pracy baud rate generator

Generator fali prostokątnej o wypełnieniu 50%. Częstotliwość taktowania portu szeregowego UART można wyliczyć ze wzoru lub wyciągnąć na podstawie schematu blokowego.

Figure 4. Timer 2 in Baud Rate Generator Mode



$$TxRxclock = \frac{f_{osc}}{32 \times (65536 - Rcap2HRcap2L)}$$

ad c) Kontroler przerwania i wektor przerwania

Konfiguracja kontrolera dotyczy tylko portu szeregowego, pozostałe przerwania nie aktywujemy.

SYSTEM PRZERWAŃ

a)

Adres bitu (hex):	AF	AD	AC	AB	AA	A9	A8	IE
Adres SFR (hex): A8	EA	ET2	ES	ET1	EX1	ETO	EXO	
	(MSB)						(LSB)	

b)

Adres bitu (hex):	BD	BC	BB	BA	B9	B8	IP
Adres SFR (hex): B8	PT2	PS	PT1	PX1	PT0	PX0	
	(MSB)					(LSB)	

c)

IE	IP	Przerwanie	Priorytet
IE.0 - EX0	IP.0 - PX1	zewnętrzne INT0	najwyższy
IE.1 - ET0	IP.1 - PT0	od licznika/czasomierza T0	
IE.2 - EX1	IP.2 - PX1	zewnętrzne INT1	najwyższy
IE.3 - ET1	IP.3 - PT1	od licznika/czasomierza T1	
IE.4 - ES	IP.4 - PS	od portu szeregowego SPI, UART	
IE.5 - ET2	IP.5 - PT2	od licznika/czasomierza T2	
IE.6 -	IP.6 -		
IE.7 - EA	IP.7 -	system przerwania	

0003H – dla przerwania zewnętrznego INT0,
 000BH – dla przerwania z licznika-czasomierza T0,
 0013H – dla przerwania zewnętrznego INT1,
 001BH - dla przerwania z licznika-czasomierza T1,
 0023H – dla przerwania z portu szeregowego,
 002BH - dla przerwania z licznika-czasomierza T2,

Sterowanie systemem przerwania: a) słowo sterujące IE;

ad d) napisać procedury obsługujące przerwanie od portu szeregowego od nadawania i odbioru

Pamiętamy że struktura programu to część inicjacyjna, pętla główna i procedury.

Ponieważ port szeregowy generuje jedno przerwanie, które dotyczy nadajnika i odbiornika należy sprawdzić która część UARTu wygenerowała przerwanie. Pamiętamy, że nadajnik po wysłaniu kompletu bitów (słowo 8 lub 9 bitowe; znak) ustawia znacznik TI i generuje przerwanie, dla odbiornika po odebraniu kompletu bitów ustawiany jest znacznik RI i jest generowane przerwanie. Ponieważ w obydwu przypadkach procesor „skacze” pod ten sam adres w wektorze przerwania więc przyjmujemy, że źródłem przerwania jest UART. Po przyjęciu przerwania należy określić która część UART je wygenerowała (nadajnik czy odbiornik) i odpowiednio wywołać rozkazem CALL procedurę nadawczą lub odbiorczą.

Proponuję następującą strukturę:

przerwanie wywołuje procedurę obsługującą przerwanie od UARTu zakończoną rozkazem RETI, w której sprawdzany jest znacznik RI i TI. Jeśli jest ustawione RI wywoływana jest procedura odbiorcza rozkazem CALL jeśli ustawiony jest TI nadawcza (obie zakończone RET). Takie rozwiązanie pozwoli łatwo sterować nadawaniem i odbieraniem.

Procedura odbiorcza

Procedura odbiorcza powinna odebrać znak/bajt i umieścić go w pamięci wewnętrznej RAM. Z wykorzystaniem rejestru indeksowego R0 i adresowania pośredniego znaki umieszczane mają być w wektorze (buforze) od adresu 40H do 50H. Czyli inicjujemy R0 wartością 40H a potem po każdym wypełnieniu (wskazywanej rejestrem R0) komórki inkrementujemy R0. Po osiągnięciu wartości 50H należy zresetować R0 do wartości 40H (czyli zawijamy wektor).

Procedura nadawcza

Procedura nadawcza wysyła (z wykorzystaniem rejestru indeksowego R1 i adresowania pośredniego) znaki z wektora od adresu 40H do 50H. Czyli inicjujemy R1 wartością 40H a potem po każdym wysłaniu znaku z komórki (wskazywanej rejestrem R1) inkrementujemy R1. Po osiągnięciu wartości 50H należy zresetować R1 do wartości 40H (czyli zawijamy wektor). Zrealizowaliśmy kolejkę FIFO.

UWAGA

Procedura odbiorcza umieszcza znaki w buforze o adresach od 40H do 50H. Znaki w odbiorniku pojawiają się z terminala, a procedura nadawcza z tego bufora pobiera znaki i je wysyła powrotem do terminala. Procedura nadawcza ma nadać pierwszy znak gdy bufor się zapełni czyli gdy procedura odbiorcza wpisze znak do adresu 50H.

Procedura nadawcza będzie wywoływana w dwojaki sposób:

- Jeśli R0 wskazuje na wartość z przedziału 40H-4FH to nic się nie dzieje. Wartość R0 sprawdzana jest w pętli głównej.
- Jeśli R0 wskazuje wartość 50H należy z pętli głównej wywołać (CALL) procedurę nadawczą procedurą zainicjuje nadawanie.
- Po nadaniu znaku nadajnik wygeneruje przerwanie i ustawi RI co spowoduje wywołanie procedury nadawczej z procedury obsługi przerwania by nadać kolejny znak, aż do momentu gdy R1 osiągnie 50H po tym fakcie należy R1 zresetować do wartości 40H (bez nadawania znaku). Rozpoczyna się oczekiwanie na następne wypełnienie buforu

Do wysyłania znaków do procesora używamy terminala tak jak w laboratorium 8