



Ćwiczenie 6

Komunikacja z komputerem (łącze RS232)

1. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z portem szeregowym mikrokontrolera 8051/52 oraz sposobem oprogramowywania tego portu.

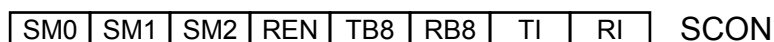
2. Opis portu szeregowego 8051

Klasyczne mikrokomputery 8051/52 mają port szeregowy (UART), umożliwiający szeregowe przesyłanie informacji (8-bitowych słów danych) liniami portu P3: RxD (P3.0 - wejście szeregowe) i TxD (P3.1 - wyjście szeregowe). Port może pracować w jednym z czterech trybów (Tabela 1). Zamiana postaci danych z równoległej na szeregową i odwrotnie oraz sterowanie wysyłaniem słowa odbywa się automatycznie. Dane odebrane przez port szeregowy są zapisywane do rejestru wejściowego transmisji szeregowej, dane wysyłane są pobierane z rejestru wyjściowego transmisji szeregowej. Oba te rejestry są umieszczone w przestrzeni adresowej wewnętrznej pamięci danych jako rejestr specjalny SBUF, pod tym samym adresem 99H. Zapisanie danych do SBUF powoduje wpisanie ich do rejestru wyjściowego, natomiast odczytanie z SBUF powoduje odczytanie zawartości rejestru wejściowego.

Tabela 1. Tryby pracy portu szeregowego w 8051.

Tryb	SM0	SM1	Opis
0	0	0	Transmisja synchroniczna , znaki ośmiobitowe taktowane sygnałem zegarowym.
1	0	1	Transmisja asynchroniczna , znaki ośmiobitowe, szybkość określana programowo.
2	1	0	Transmisja asynchroniczna , znaku dziewięciobitowe, szybkość 1/32 lub 1/64 częstotliwości zegara.
3	1	1	Transmisja asynchroniczna , znaki dziewięciobitowe, szybkość określana programowo.

Do kontroli pracy portu szeregowego służy rejestr sterujący SCON (Rys.1.) znajdujący się pod adresem 98H. Znaczenie poszczególnych bitów tego rejestru przedstawiono w Tabeli 2.



Rys. 1. Słowo sterujące SCON

Tabela 2. Słowo sterujące SCON

Bit	Znaczenie
SM0	definiuje tryb pracy (patrz Tabela 1)
SM1	definiuje tryb pracy (patrz Tabela 1)
SM2	Bit sterujący umożliwiający tworzenie sieci zbudowanej z μC 8051 komunikujących się ze sobą w trybie 2 i 3. Jeżeli SM2 = 1, to maskowane jest odebranie znaku, w którym D8 = 0 (znacznik RI

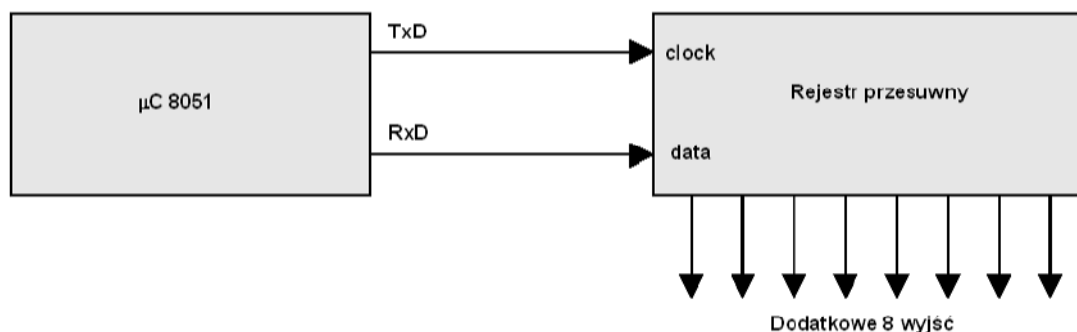
	nie jest ustawiany, znak nie jest zapisywany do rejestru SBUF). Jeżeli SM2 = 0, to wartość dziewiątego bitu (D8) nie jest istotna dla działania układów transmisji szeregowej. Przed wysłaniem znaku wartość tego bitu może być ustalana programowo i wpisywana do znacznika TB8, a po odebraniu znaku odczytywana ze znacznika RB8. Dla trybu 0 bit powinien być wyzerowany.
REN	Ustawienie w stan 1 włącza (uaktywnia) odbiornik – umożliwia odbiór danych. Wartość bitu ustawiana programowo.
TB8	Dziewiąty bit danej, która ma zostać wysłana w trybach 2 i 3, zmieniany programowo.
RB8	Dziewiąty bit odebranego znaku (tryb 2 i 3). W trybie pierwszym bit przyjmuje wartość odebranego bitu stopu (gdy SM2 = 0). W trybie 0 nie jest wykorzystywany. Zerowanie bitu odbywa się programowo.
RI	Znacznik przerwania odbiornika. Znacznik jest ustawiany sprzętowo po odebraniu słowa. Znacznik zerowany programowo.
TI	Znacznik przerwania nadajnika. Znacznik jest ustawiany po wysłaniu słowa. Znacznik zerowany programowo.

2.1. Tryby pracy portu szeregowego

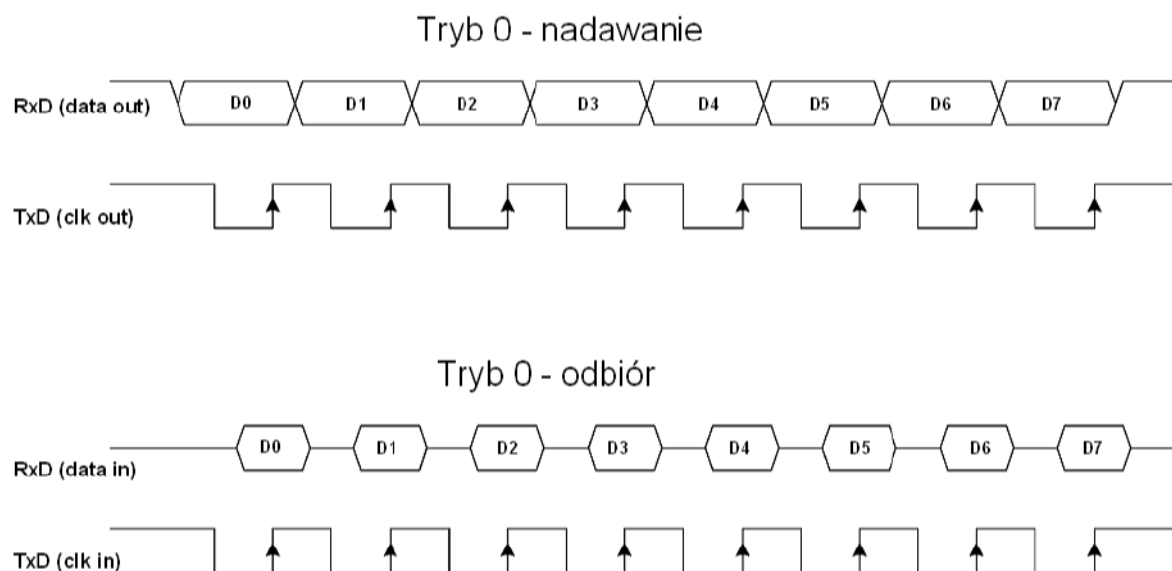
Tryb 0 (synchroniczny)

W trybie 0 port szeregowy pracuje jako nadajnik lub odbiornik 8-bitowych słów szeregowych (pierwszy bit najmniej znaczący). Słowa te są przesyłane po linii P3.0 (RxD) w obie strony, a odbierane lub nadawane przez zewnętrzny rejestr przesuwający, taktowany sygnałem zegarowym wysyłanym na linię P3.1 (TxD) (Rys.2.). Częstotliwość sygnału taktującego wynosi $f_{XTAL}/12$. Wysłanie znaku jest inicjowane przez wykonanie dowolnego rozkazu, który wpisuje dane do rejestru SBUF . Zakończenie wysyłania znaku powoduje ustawienie znacznika TI w rejestrze SCON. Odebranie całego znaku powoduje ustawienie znacznika RI w rejestrze SCON (Rys.3).

Ustawienie znacznika TI lub RI jest sygnałem zgłoszenia przerwania z portu szeregowego. Przyjęcie tego przerwania (gdy nie jest ono zamaskowane) nie powoduje automatycznego wyzerowania znaczników, a zatem ich stan może być w programie obsługi odczytany w celu identyfikacji przyczyny przerwania. Znaczniki powinny zostać wyzerowane programowo.



Rys. 2. Przykładowe wykorzystanie portu szeregowego 8051 w Trybie 0.

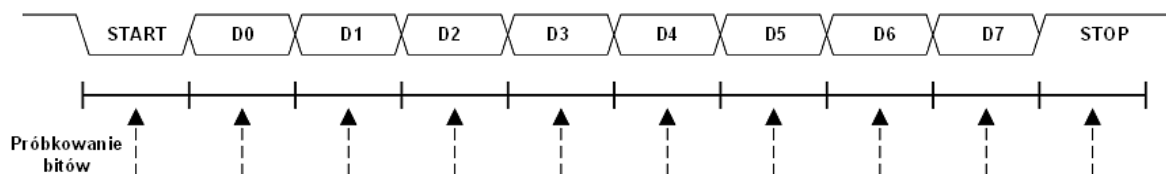


Rys.3. Zależności czasowe portu szeregowego 8051 dla Trybu 0.

Tryb 1 (asynchroniczny)- zapewnia realizację standardowej, dwukierunkowej transmisji asynchronicznej w formacie: bit START, 8 bitów danych (pierwszy LSB), bit STOP (Rys.4.). Dane są wysyłane linią TxD, a odbierane linią RxD (Rys.6.). Podczas odbioru bit stopu zostaje zapisany na bicie RB8 rejestru SCON. Sygnałem taktującym układ transmisji szeregowy (zegarem transmisji) jest sygnał przepełnienia licznika T1 z układu czasowo licznikowego. Szybkość transmisji jest więc programowalna zgodnie z zależnością:

$$f = \frac{2^{SMOD}}{32} \cdot \frac{f_{osc}}{12 \cdot [256 - (TH1)]}$$

SMOD jest najstarszym bitem rejestru specjalnego PCON (nieдоступnego bitowo). Symbol PCON nie jest zdefiniowany w assemblerze, zatem trzeba ten symbol zdefiniować lub posłużyć się adresem tego rejestru (87H). Częstotliwość oscylatora $f_{osc} = 11,059$ MHz. TH1 to jeden z dwóch rejestrów licznika T1, taktującego szeregowy przesył danych.



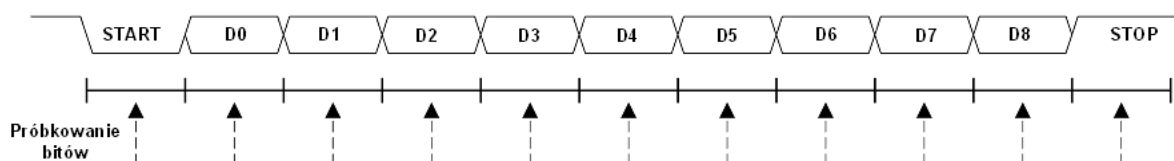
Rys.4. Rys.3. Zależności czasowe portu szeregowego 8051 dla Trybu 1.

Tryb 2 (asynchroniczny)- dane są wysyłane linią TxD, a odbierane linią RxD. Transmisja danych odbywa się w porcjach 11-bitowych: bit startu (0), 8 bitów danych (pierwszy LSB), programowalny 9-ty bit danych, oraz bit stopu (1) (Rys.5.). Przy wysyłaniu danych 9-ty bit danych pochodzi z bitu TB8 rejestru specjalnego SCON (bitowi temu można programowo

nadać odpowiednią wartość). Podczas odbioru 9-ty bit danych jest zapisywany do bitu RB8. Częstotliwość transmisji jest równa 1/32 lub 1/64 częstotliwości oscylatora:

$$f = \frac{2^{SMOD}}{64} \cdot f_{osc}$$

Tryb 3 (asynchroniczny)- dane są wysyłane przez wyprowadzenie TxD, a odbierane z RxD. Przesył danych odbywa się w porcjach 11-bitowych: bit startu (0), 8 bitów danych (najpierw LSB), programowalny 9-ty bit danych, oraz bit stopu (1). Przy wysyłaniu danych 9-ty bit danych pochodzi z bitu TB8 rejestru specjalnego SCON (bitowi temu można programowo nadać odpowiednią wartość). Podczas odbioru 9-ty bit danych jest zapisywany do bitu RB8. Częstotliwość przesyłu jest regulowana tak samo jak w trybie 1.



Rys.5. Rys.3. Zależności czasowe portu szeregowego 8051 dla Trybu 2,3.

Uwaga: tryby 2 i 3 służą do komunikacji wieloprocesorowej.

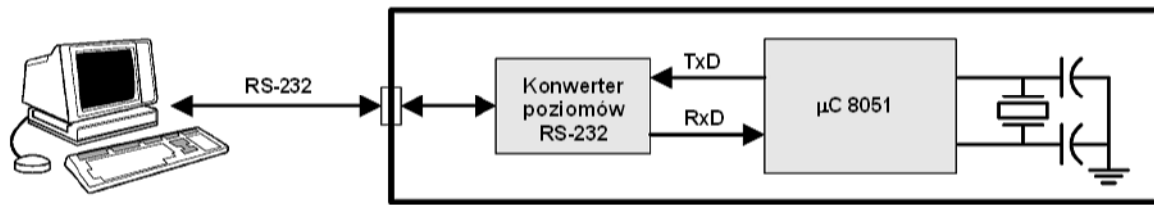
2.2. Przerwanie od portu szeregowego

Port szeregowy zgłasza przerwanie w dwóch przypadkach: kiedy odebrał bajt (znacznik RI) oraz kiedy zakończył wysyłanie bajtu (znacznik TI). Jeżeli przerwanie jest uaktywnione to wówczas następuje sprzętowy skok pod adres 23H i tam powinien się znaleźć skok do procedury obsługi przerwania.

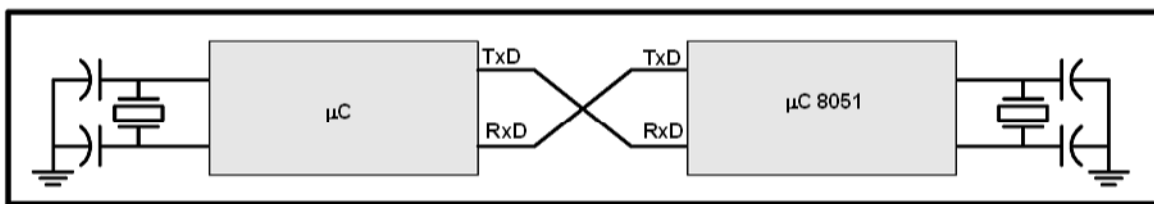
Sens przerwania związanego z odbiorem jest oczywisty - port szeregowy musi bowiem poinformować mikrokontroler, że należy odczytać odebrany bajt (bajt ten znajduje się w rejestrze SBUF). Powód, dla którego zaimplementowano także przerwanie związane z wysyłaniem bajtu jest nieco bardziej skomplikowany. Inicjacji transmisji dokonuje się zapisując bajt, który chcemy wysłać, do rejestru SBUF. Z punktu widzenia szybkości działania mikrokontrolera komunikacja szeregową jest procesem powolnym. Przy typowej szybkości transmisji 9600 bitów na sekundę wysyłanie jednego bajtu trwa ok 1 ms, tymczasem cykl maszynowy 8051 wynosi ok. 1 μ s. Operacja zapisu bajtu do rejestru SBUF trwa jeden (mov SBUF, A) lub dwa (mov SBUF, #'L') cykle maszynowe. Zapisanie do rejestru SBUF kolejnego bajtu, zanim port zakończył transmisję poprzedniego, spowodowałoby przekłamanie. Z zapisem następnego bajtu trzeba zatem poczekać do momentu, kiedy zgłosi się przerwanie związane z wysyłaniem znaku (znacznik TI), które informuje nas o tym, że poprzedni bajt został wysłany.

Ponieważ portowi szeregowemu przypisano tylko jeden wektor przerwania, programista musi mieć możliwość sprawdzenia w procedurze obsługi przerwania, z którym przypadkiem ma do czynienia (odbior czy transmisja). Z tego względu znaczniki przerwania TI i RI **nie są automatycznie zerowane** po przyjęciu przerwania. Jeżeli RI = 1, to zgłosiło się przerwanie związane z odbiorem, a jeżeli TI = 1, to zgłosiło się przerwanie związane z wysłaniem bajtu. Może się zdarzyć, że oba przerwania zgłosiły się jednocześnie. Przed wyjściem z procedury obsługi przerwania należy wyzerować znaczniki przerwania, gdyż w przeciwnym razie

program wejdzie z powrotem do procedury obsługi przerwania (znaczniki przerw są testowane w każdym cyklu maszynowym).



lub



Rys.6. Schemat połączenia portu szeregowego dla trybów 1,2,3.

3. Programowanie portu szeregowego

3.1. Programowanie rejestru SCON

Komunikacja z komputerem narzuca następujące założenia:

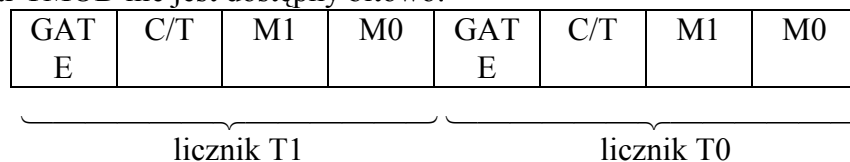
- port szeregowy musi pracować w trybie 1 lub 3, bo potrzebna jest regulowana częstotliwość przesyłu danych, wybieramy tryb 1, zatem SM0=0, SM1=1
- chcemy, aby przerwanie związane z odbiorem zostało zgłoszone tylko wtedy, kiedy bit stopu jest poprawny (RB8=1), zatem SM2=1
- chcemy mieć możliwość odbierania znaków, zatem REN=1
- należy wyzerować znaczniki przerwania, zatem TI=RI=0
- wartości bitów TB8 i RB8 nie mają znaczenia

Wykonujemy zatem instrukcję:

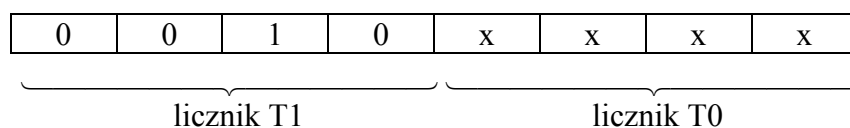
```
mov SCON, #01110000B
```

3.2. Programowanie licznika T1

Rejestr TMOD nie jest dostępny bitowo:



- bramkowanie licznika T1 nie jest potrzebne, zatem GATE=0
- licznik powinien zliczać impulsy zegara, zatem C/T=0
- licznik powinien pracować w trybie 2 (8-bitowy z automatycznym przeładowaniem), zatem M1=1, M0=0



Jeżeli do rejestru TMOD uprzednio wpisano wartość określającą konfigurację licznika T0 (przeoglądanie klawiatury), to należy wykonać instrukcje:

```
and    TMOD,#0FH
orl    TMOD,#20H
```

Przerwanie związane z licznikiem T1 nie jest nam potrzebne, więc trzeba je zdezaktywować (mogłoby się przydać tylko wtedy, gdyby potrzebna była tak niewielka szybkość przesyłu, że licznik musiałby pracować jako 16-bitowy, konieczne byłoby wtedy programowe przeładowanie licznika w procedurze obsługi przerwania), zatem ET1=0 (bit IE.3)

```
clr    ET1
```

3.3. Określenie szybkości transmisji

Zakładamy, że ustalona szybkość transmisji wynosi 9600 bitów/s, zatem SMOD=0, a wartość rejestru TH1 wynosi FDH:

```
mov    87H,#0
mov    TH1,#0FDH
```

Licznik musi zostać uruchomiony:

```
setb   TR1
```

Aby możliwe było przyjęcie znaku, trzeba uaktywnić przerwanie związane z portem szeregowym:

```
setb   ES
setb   EA
```

3.4. Procedura inicjacji portu szeregowego

Ostatecznie procedura inicjacji portu szeregowego ma następującą postać:

Sport_Init:

```
mov    SCON,#01110000B ; zaprogramowanie portu szeregowego
and    TMOD,#0FH       ; wyzerowanie programowania T1
orl    TMOD,#20H       ; zaprogramowanie licznika T1
mov    TH1,#0FDH       ; określenie szybkości transmisji
mov    TL1,#0FDH
mov    87H,#0          ; SMOD=0
clr    ET1             ; dezaktywacja przerwania od T1
setb   TR1             ; uruchomienie T1
setb   TReady         ; flaga definiowana przez użytkownika, informuje
                        ; o tym czy port zakończył transmisję bajtu
setb   EA              ; globalna aktywacja przerwania
setb   ES              ; aktywacja przerwania od portu szeregowego
```

```
ret
```

3.5. Procedura obsługi przerwania

Procedura obsługi przerwania znajduje się pod adresem 4023H

```

org    4023H
jmp    RSInt
...
...
RSInt:
                                ;zachowanie kontekstu
...
...
jb     TI, transmisja
                                ; obsługa odebranego znaku
mov    A, SBUF                  ; odczytanie znaku
...
...
clr    RI                      ; kasowanie znacznika przerwania związanego
                                ;z odbiorem (receive)
jmp    RS_Exit

transmisja:
setb   Tready
...
...
clr    TI                      ; kasowanie znacznika przerwania związanego z transmisją

RS_Exit:
                                ; odtworzenie kontekstu
...
...
reti
```

4. Zadanie do wykonania

Wykorzystując uprzednio napisany program obsługujący klawiaturę i wyświetlacz ciekłokrystaliczny zrealizować dwukierunkową komunikację systemu uruchomieniowego 8051 z komputerem poprzez port szeregowy.

Na początku program użytkownika przesyła do komputera napis początkowy (zdefiniowany przez użytkownika). Napis ten pojawi się na ekranie komputera jeżeli uprzednio załadujemy program Tera Term.

Po wyświetleniu napisu początkowego program użytkownika przechodzi do normalnego trybu komunikacji, tj. znaki wprowadzone z klawiatury komputera PC mają być wyświetlone na wyświetlaczu LCD systemu uruchomieniowego (Tera Term automatycznie wysyła te znaki przez port szeregowy komputera, można je zatem odebrać przez port szeregowy 8051), natomiast znaki wprowadzone z klawiatury systemu mają być wyświetlone zarówno na

wyświetlaczu LCD, jak i na ekranie komputera (trzeba wysłać te znaki przez port szeregowy 8051, wówczas Tera Term automatycznie odbierze je i pokaże na ekranie komputera).

Na ocenę bardzo dobrą:

Napisać program wpisywania z terminala stałej 16-bitowej z możliwością edycji i rozpoznawaniem znaku ENTER (S1=). Przekształcić wprowadzoną stałą na postać heksadecymalną i wysłać na terminal.

UWAGA!!!

Wysyłanie napisu początkowego można zorganizować w taki sposób, że pierwszy znak napisu zapisywany jest do rejestru SBUF w programie głównym (może to być jakaś procedura), natomiast wszystkie pozostałe w procedurze obsługi przerwania.

Wysyłanie znaków do komputera w normalnym trybie komunikacji (za pomocą klawiatury) jest nieco bardziej skomplikowane, bo procedura obsługi przerwania „nie wie” czy jest jakiś znak do wysłania. Dlatego trzeba wprowadzić flagę Tready, której stan informuje program główny czy transmisja poprzedniego bajtu została zakończona, czy nie. W ten sposób decyzję o tym, czy jest jakiś znak do wysłania podejmuje procedura obsługi klawiatury, a w przypadku gdyby należało taki znak wysłać, trzeba tylko przetestować flagę Tready, żeby sprawdzić czy port jest wolny. Po zapisaniu znaku do SBUF należy tę flagę wyzerować.