

Arhitectura calculatoarelor 2. Laborator 11,12.

Studiul și experimentarea interfeței seriale programabile Z80SIO

1. Obiectivul lucrării

Lucrarea are ca obiectiv însușirea structurii interfeței seriale Z80-SIO și experimentarea unor posibilități de funcționare în transmisia de date de tip asincron. Se urmăresc atât aspecte hardware cât și software.

2. Problematika lucrării

2.1. Transmisia asincronă

Un canal de comunicație de tip **duplex** presupune că transmisia datelor între emițător și receptor are loc simultan în ambele sensuri.

În sistemele care folosesc modul **asincron**, linia de transmisie se găsește în repaus în starea '1' logic sau MARK. Emisia unui caracter este precedată de trecerea liniei în starea '0' logic sau SPACE, pe durata unui bit, numit bit de START. Această trecere indică receptorului că urmează emisia unui caracter. Dispozitivul receptor detectează bitul de START și, în continuare, recepționează biții de date. După transmiterea caracterului, linia este trecută în starea MARK pe durata a 1, 1 și jumătate sau 2 biți, numiți biți de STOP, pregătindu-se astfel pentru emisia unui nou caracter. Procesul de emisie va fi astfel repetat caracter cu caracter până la terminarea transmiterii întregului mesaj, momentele de început ale fiecărui caracter succedându-se în *mod asincron*, la intervale diferite de timp.

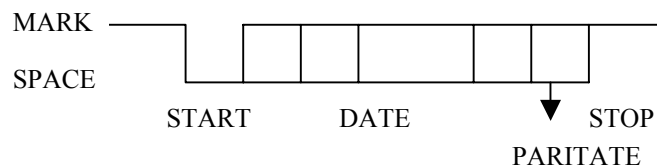


Fig.1 Structura unui caracter în transmisia serială asincronă

2.2. Structura circuitului și conexiunile externe

Schema bloc a circuitului este prezentată în figura 2. iar conexiunile externe în figura 3.

Circuitul este realizat în jurul unei magistrale interne la care se conectează blocurile funcționale. Acestea realizează interfața cu magistrala unității centrale, comanda internă, comanda întreruperilor, precum și programarea și exploatarea a 2 canale seriale duplex A și B. Fiecare canal este constituit din blocul de transmisie/recepție serială, interfața cu un echipament extern de tip modem, precum și din registrele de comenzi și stări.

Semnalele magistralei de date D0–D7, precum și liniile de comenzi RESET/, M1/, IORQ/, RD/ au semnificațiile cunoscute. Linia CE/ selectează global circuitul, iar liniile B/A/ și C/D/ selectează canalul (B sau A) respectiv registrul de comenzi sau de date al fiecărui canal.

RxD (Receive Data) este intrarea pentru datele seriale care se eșantionează cu frontul pozitiv al semnalului RxC (Receive Clock).

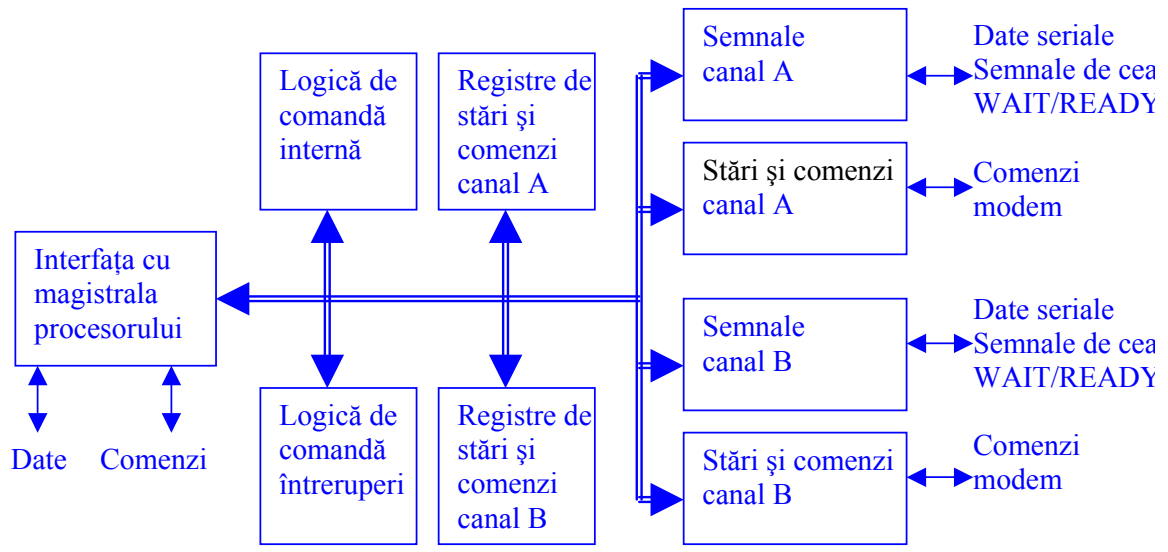


Fig. 2.

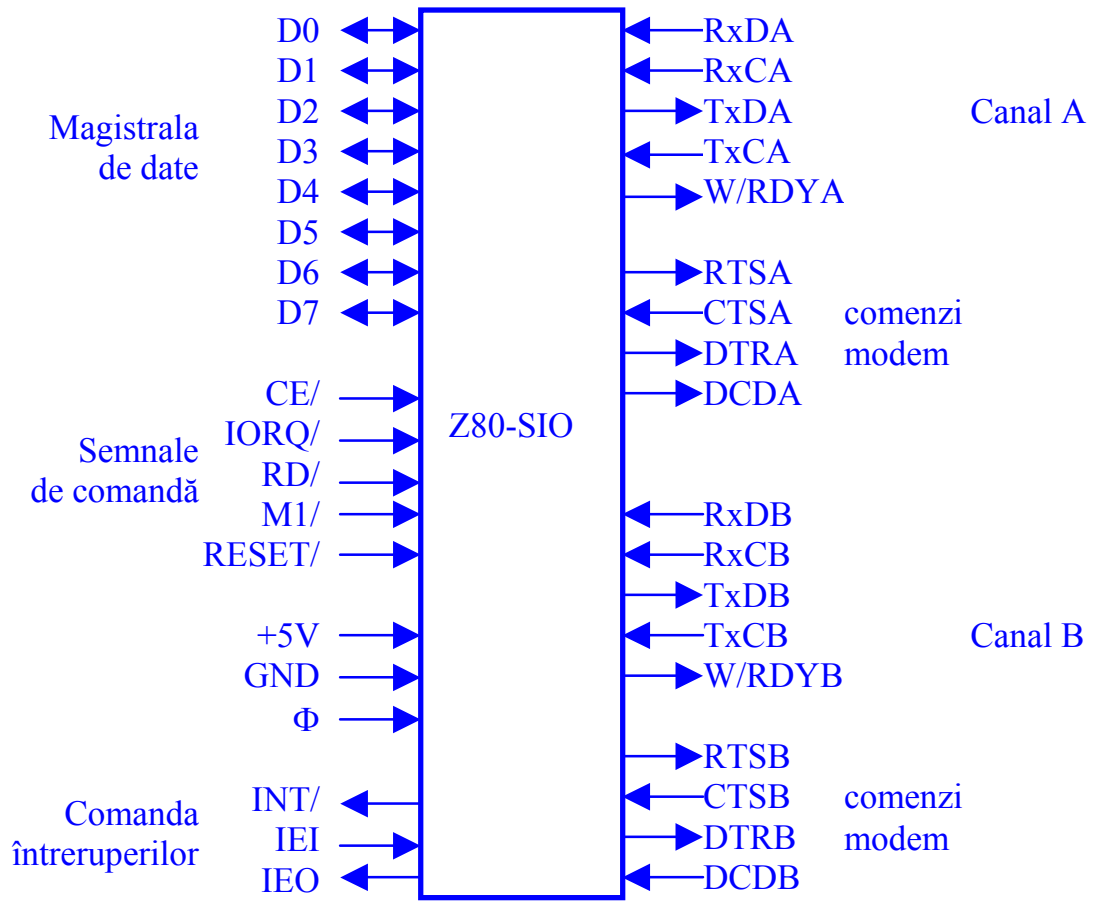


Fig. 3

TxD (Transmit Data) este ieșirea pentru datele seriale care se modifică pe frontul negativ al semnalului TxC (Transmit Clock). Semnalele RxC și TxC se obțin, în general, de

la un timer programabil (Z80-CTC). Ele sînt divizate intern cu un factor egal cu 1, 16, 32 sau 64.

Z80- SIO poate lucra în întreruperi în modul vectorizat al procesorului Z80 și poate fi integrat într-un sistem de priorități serial. Linia INT/ (Interrupt Request) este o cerere de întrerupere adresată procesorului .

3. Funcționarea circuitului

Funcționarea circuitului este determinată de conținutul registrelor de comandă, care trebuie programate înainte ca SIO să execute vreo funcție. Registrele de stare se pot citi în orice moment, dar numai unele comenzi și moduri pot fi modificate pe durata unei funcționări deja programate.

3.1. Moduri de lucru cu unitatea centrală

În modul de lucru “polling“ (testare prin software) este necesară verificarea permanentă a parametrilor ce caracterizează transferul de date prin citirea registrelor de stare RR0 și RR1.

În modul de lucru bazat pe întreruperi, se programează un vector de întrerupere care va putea fi returnat modificat de către circuit, astfel încît acesta poate “puncta“ direct 8 rutine de tratare diferite. Modul în care se modifică vectorul depinde de sursa întreruperii. Întreruperile se împart în trei categorii: de recepție, de transmisie și externe/de stare, cu această ordine de prioritate, canalul A fiind prioritar față de canalul B.

4. Programarea circuitului

Funcționarea fiecărui canal se definește prin software cu ajutorul a 8 registre de comandă ce pot fi programate, notate WR0 – WR7 și care au următoarele funcții principale :

- WR0 – adresarea celorlalte registre, inițializarea CRC, inițializări pentru diverse moduri de lucru
- WR1 – definirea modului de transfer și întreruperi transmisie/recepție
- WR2 – vector de întrerupere (scris numai pentru canalul B)
- WR3 – comenzi și parametri de recepție
- WR4 – moduri și parametri diverși de transmisie/recepție
- WR5 – comenzi și parametri de transmisie
- WR6 și WR7 sînt utilizați doar pentru transmisia sincronă

Pentru a cunoaște starea fiecărui canal, programatorul poate citi trei registre de stare notate RR0, RR1 și RR2 care dau informații despre condițiile de eroare, vectorul de întrerupere sau starea unor semnale de interfață :

- RR0 – starea bufferului de transmisie/recepție, stare de întrerupere, stare externă
- RR1 – stare condiție de recepție specială
- RR2 – vector de întrerupere modificat (citit numai pe canalul B)

4.1. Registrele de comandă

Din structura registrelor de comandă și de stare vom ignora biții care nu ne interesează în această lucrare.

Cu excepția lui WR0, programarea celorlalte registre de comandă necesită scrierea a 2 octeți. Biții D0–D2 din WR0 specifică adresa registrului în care se va înscrie octetul de comandă următor.

Registrul de comandă WR0

	D7	D6	D5	D4	D3	D2	D1	D0
WR0	0	0	0	0	0	Adresa registrului următor		

Registrul de comandă WR1

	D7	D6	D5	D4	D3	D2	D1	D0
WR1	0	0	0	Mod de întrerupere la recepție		Starea afectează vectorul	Validare întreruperi transmisie	0

D1 = 1 permite activarea întreruperii atunci când buferul de transmisie devine gol

D2 = 1 permite selectarea facilității ca vectorul trimis de SIO într-un ciclu de achitare a întreruperii să fie modificat (3 biți) conform tabelului :

V3	V2	V1	Semnificația modificării	
0	0	0	Buffer de transmisie gol	Canal B
0	1	0	Caracter recepționat disponibil	
1	0	0	Buffer de transmisie gol	Canal A
1	1	0	Caracter recepționat disponibil	

Dacă D2 = 0 vectorul rămâne cel programat în registrul WR2.

Selectarea modului de întrerupere la recepție se face conform tabelului :

D4	D3	Semnificația modificării
0	0	Întrerupere recepție invalidată
0	1	Întrerupere la primul caracter recepționat
1	0	Întrerupere la fiecare caracter recepționat/ Paritatea afectează vectorul
1	1	Întrerupere la fiecare caracter recepționat/ Paritatea nu afectează vectorul

Registrul de comandă WR2

Conține vectorul de întrerupere al circuitului și există doar în setul de registre al canalului B.

	D7	D6	D5	D4	D3	D2	D1	D0
WR2	V7	V6	V5	V4	V3	V2	V1	0
	Se returnează nemodificați				Modificați numai dacă D2/WR1 = 1			

Registrul de comandă WR3

Conține comenzi și parametri referitori la logica de recepție :

	D7	D6	D5	D4	D3	D2	D1	D0
WR3	Lungime caracter la recepie		0	0	0	0	0	Validare recepie

Biții D7 și D6 stabilesc lungimea caracterului la recepie:

D7	D6	Număr de biți/caracter
0	0	5
0	1	7
1	0	6
1	1	8

Registrul de comandă WR4

Conține comenzi care afectează atât transmisia cât și recepția

	D7	D6	D5	D4	D3	D2	D1	D0
WR4	Rata datelor		0	0	Număr de biți STOP		Paritate pară/impară	Paritate

D0 = 1 – la transmisie se adaugă un bit de paritate, iar la recepie se verifică paritatea

D1 = 1 – paritate pară D1 = 0 – paritate impară (atât pentru transmisie cât și pentru recepie)

D3	D2	Număr de biți STOP
0	0	-
0	1	1 bit de STOP pe caracter
1	0	1 și jumătate biți de STOP
1	1	2 biți de STOP pe caracter

Biții D7 și D6 se utilizează pentru specificarea ratei de transmisie a datelor, în funcție de frecvența ceasului de emisie/recepție.

D7	D6	Rata datelor
0	0	Rata datelor x 1 = frecvența ceasului
0	1	Rata datelor x 16 = frecvența ceasului
1	0	Rata datelor x 32 = frecvența ceasului
1	1	Rata datelor x 64 = frecvența ceasului

Registrul de comanda WR5

Conține comenzi și parametri care afectează transmisia.

	D7	D6	D5	D4	D3	D2	D1	D0
WR5	0	Lungime caracter la transmisie		0	Validare transmisie	0	0	0

D3 = 1 – valideaza emisia unui caracter.

Bitii D6 și D5 stabilesc lungimea caracterului la recepie:

D6	D5	Număr de biți/caracter
0	0	5
0	1	7
1	0	6
1	1	8

Biții care se emit se presupun aliniați la dreapta octetului, transmisia începînd cu bitul cel mai puțin semnificativ.

3.2 Registrele de stare

Sînt disponibile 3 registre pentru memorarea stării fiecărui canal: RR0, RR1, RR2 (RR2 nu poate fi citit decît pe canalul B). Vom prezenta doar registrul RR0, care va fi utilizat în aplicații.

Registrul de stare RR0

	D7	D6	D5	D4	D3	D2	D1	D0
RR0	0	0	0	0	0	Bufer transmisie gol	0	Caracter recepționat disponibil

D0 = 1 – cînd în buferul de recepție se află cel puțin un caracter care poate fi citit de microprocesor.

D2 = 1 – cînd buferul de transmisie devine gol și microprocesorul poate scrie un nou caracter.

5. Desfășurarea lucrării

5.1 Portul B al SIO este folosit pentru comunicația între sistemul de dezvoltare și calculatorul PC. Prin urmare, acest port este programat de către programul monitor la lansarea acestuia în funcțiune după RESET. Se va scrie un program care să exploateze în funcționare acest port astfel:

- programul va avea o buclă în care va testa intrarea SIO (apăsarea unei taste).
- codul citit va fi trimis în retur pentru afișare, urmat de un caracter "spațiu".

În acest timp, calculatorul PC trebuie să funcționeze în regim de emulator de terminal de comunicații, adică în regimul "Run the Monitor" al mediului de dezvoltare.

5.2 Se va realiza o legătură între două sisteme de dezvoltare. Pentru aceasta se vor lega porturile SIO A ale celor două module între ele. Pe fiecare modul se va rula un program care va îndeplini următoarele funcții:

- După lansare va programa portul A să lucreze la rata de transfer de 1200 baud, cu 7 biți pe caracter, fără paritate, cu emiterea unei cereri de întrerupere la recepționarea unui caracter.
- Va intra într-o buclă de polling pe intrarea B. Caracterele citite de la această intrare vor fi trimise către ieșirea portului A.
- Rutina de întrerupere activată de recepția unui caracter de la portul A va citi caracterul și îl va trimite către ieșirea portului B.

În acest fel, cu cele două sisteme legate între ele prin porturile A, și fiecare dintre ele legat la un terminal prin portul B, ceea ce se tastează la terminalul unui sistem va fi afișat pe displayul celuilalt și reciproc.