

LUCRAREA 9

Transferul serial

Caracteristici

Transferul serial se definește ca fiind transmisia secvențială a datelor între două puncte de comunicație. Suportul fizic necesar transferului de informații este reprezentat de către magistrale. Caracteristica principală a unei magistrale seriale este transmisia bit cu bit a informațiilor, folosindu-se pentru aceasta un număr redus de semnale sau linii de comunicație.

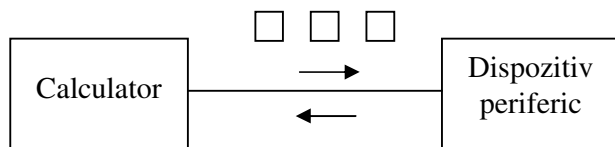


Figura 1. Transferul serial al datelor

Spre deosebire de magistrala serială, o magistrală paralelă permite transferul simultan al mai multor biți pe mai multe linii de date. Transmisia serială, deși asigură o viteză de transfer mai redusă în comparație cu transmisia paralelă, prezintă în plus o serie de avantaje, cum ar fi: un număr redus de linii de transmisie și o distanță de transfer mai mare.

Clasificarea modalităților de transfer serial [1]

Între două puncte de comunicație se pot stabili mai multe moduri de transfer serial. Se poate face o clasificare a acestor moduri în funcție de următoarele criterii :

1. Modul de sincronizare :

- transfer sincron, în care se utilizează un semnal de ceas care să indice momentul în care un anumit bit de dată este valid ;
- transfer asincron, în care nu se utilizează un semnal de ceas, sincronizarea între unitatea emitentă și cea receptoare făcându-se pe baza structurii

specifice a blocului de date transmis (care de exemplu poate include un bit de start, 6 biți de date, un bit de stop).

2. Lungimea blocului de date transmis :

- transfer pe octet
- transfer pe bloc (număr mai mare de octeți)

3. Numărul de participanți în comunicație

- transfer punct-la-punct : legătura se realizează între două puncte de comunicație
- transfer multipunct : legătura se realizează în circuit între mai multe componente, dintre care la un moment dat una transmite și celelalte ascultă

4. Direcția de transfer

- transfer unidirecțional (într-un singur sens)
- transfer bidirecțional sau **full duplex** (simultan în două sensuri)
- transfer bidirecțional pe o singură linie sau **half duplex** (se transmite pe rând în cele două direcții)

De asemeni, magistralele seriale se pot clasifica după modul în care realizează legătura serială :

- magistrale sistem, folosite pentru interconectarea componentelor unui sistem de calcul (microcontroler, memorii, convertoare)
- linii seriale, folosite pentru interconectarea unor calculatoare și dispozitive periferice (imprimantă , consolă) prin legătură punct-la-punct
- rețele de comunicație, utilizate pentru comunicația multipunct între mai multe calculatoare.

Protocolul de comunicație

Protocolul de comunicație reprezintă un set de reguli pe baza căruia se realizează transferul serial. Două echipamente care comunică pe o magistrală serială trebuie să respecte același protocol, respectiv aceleași reguli de transmisie, cerință fără de care nu s-ar putea înțelege reciproc.

Protocolul cuprinde două categorii de informații:

A. Reguli privind modalitatea fizică de efectuare a legăturii, cum ar fi:

- tipul și semnificația semnalelor folosite pentru transmisie,
- mecanismele de sincronizare a echipamentelor comunicante,
- tipul de conectori folosiți,
- natura și parametri fizici ai mediului de transmisie (cablu bifilar torsadat, cablu coaxial, fibră optică, etc.)

B. Reguli privind parametrii transmisiei, ca de exemplu :

- viteza de transfer a datelor (9600 bauds, 19600 bauds, etc.) ;
- tipul de sincronizare (transmisie sincronă sau asincronă) ;
- mecanismele de detecție și corecție a erorilor ;
- structura blocului de date, etc.

Standarde folosite în comunicația serială [1]

Pentru a asigura compatibilitatea între diferite componente realizate de diverși producători, s-au definit mai multe de standarde internaționale. Cele mai cunoscute standarde utilizate în comunicația serială sunt : RS 232, RS 485, HDLC/SDLC, pentru comunicația asincronă, respectiv I2C pentru comunicația sincronă. Standardul RS 232 se folosește pentru comunicația punct la punct pentru transmisia de coduri ASCII sau date în format binar. Standardul RS 485 asigură comunicația multipunct pe cablu bifilar torsadat, iar HDLC/SDLC este un standard pentru comunicare în rețea ce folosește un protocol de comunicație bazat pe cadre. Standardul I2C se utilizează pentru interconectarea pe o magistrală serială a componentelor unui sistem cu microcontroler.

Standardul RS-232

Este cel mai cunoscut standard de **comunicație serială asincronă**. Inițial standardul a fost conceput cu scopul de a permite conectarea unui terminal inteligent la un calculator central printr-o legătură telefonică. Standardul precizează interfața dintre un echipament de calcul (DTE- Data Terminal Equipment) și adaptorul său la linia telefonică – modem (DCE- Data Circuit Equipment) – vezi figura 2 :

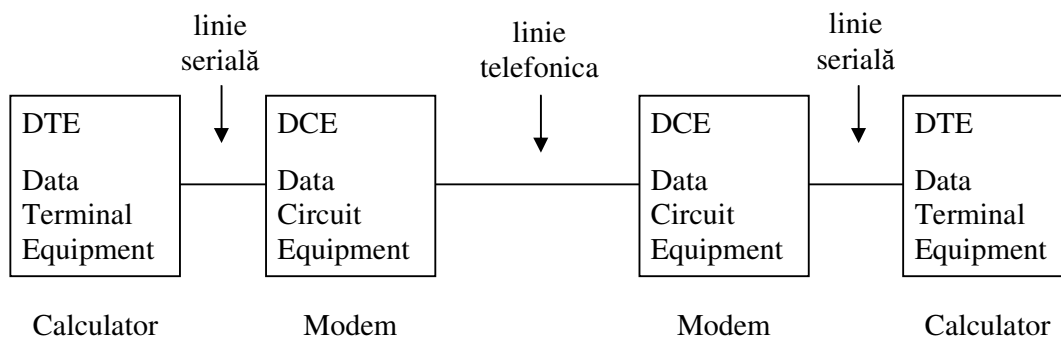


Figura 2. Componentele unui ansamblu de comunicație în standardul RS-232

Interfața permite comunicația serială bidirecțională între calculator (DTE) și componenta de comunicație (DCE), la o viteză de transfer între 110 și 19200 bauds. Standardul se poate folosi și pentru a realiza legături seriale între diverse echipamente fără a mai folosi un modem.

Așa cum am mai spus, modul de transmisie este în acest caz serial asincron, bidirecțional. Codificarea informațiilor binare (1 logic, respectiv 0 logic) se face prin nivele de tensiune sau curent.

În standardul RS-232, avem următoarea structură a informației de transmis (figura 3):

- un bit de start (0 logic)
- 5-8 biți de date
- 0-1 bit de paritate (pentru detecția erorilor)
- 1-2 biți de stop (1 logic)

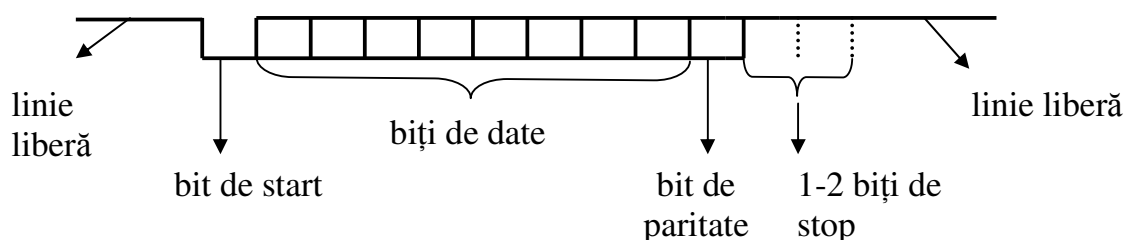


Figura 3. Structura unui bloc de date transmis în standardul RS-232

În cazul transmisiei seriale asincrone, sincronizarea între transmițător și receptor se realizează la începutul fiecărui caracter prin bitul de start (0 logic). De precizat că în repaus linia este în 1 logic. Citirea datelor se face secvențial, la jumătatea intervalelor de bit care urmează bitului de start. Protocolul asigură citirea corectă a datelor chiar și în cazul în care există mici diferențe (până la 2%) între frecvența de emisie și cea de citire a datelor. Această sincronizare nu s-ar păstra în cazul în care lungimea blocului de date ar fi mai lungă.

Protocoloale de comunicație în cazul standardului RS-232

Standardul RS-232 prezintă posibilitatea de specificare a două tipuri de protocoale :

- A. **Protocol hardware**, caz în care controlul transferului de date se realizează prin intermediul semnalelor fizice DTR/DSR. Cu ajutorul acestor semnale, receptorul poate opri fluxul de date transmis, de exemplu din cauza vitezei prea mari a transmițătorului.

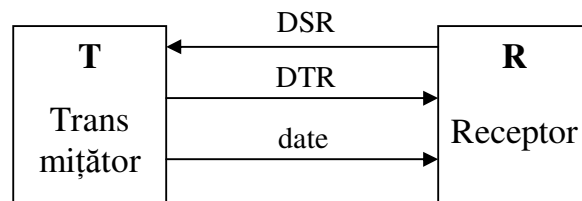


Figura 4. Semnalele de control utilizate în cazul protocolului hardware

Semnificația semnalelor este următoarea:

- DTR (Data Terminal Ready) : transmițătorul este pregătit pentru transmisie
- DSR (Data Set Ready) : receptorul e gata să primească datele.

DTR este generat de către transmițător în momentul în care el poate furniza datele de transmis. Când receptorul este gata, generează semnalul DSR, moment în care transmisia poate începe. Transmisia se oprește ori de câte ori semnalul DSR este dezactivat.

Pentru transmisia și recepția datelor se folosesc semnalele :

- TXD (Transmit Data), la transmițător, respectiv
- RXD (Receive Data), la receptor.

Conectarea semnalelor la capetele de comunicație este ilustrată în figura 5. În cazul transmisiei unidirecționale, se folosesc doar semnalele figurate cu linie continuă, iar pentru transmisia bidirecțională, și semnalele cu linie punctată.

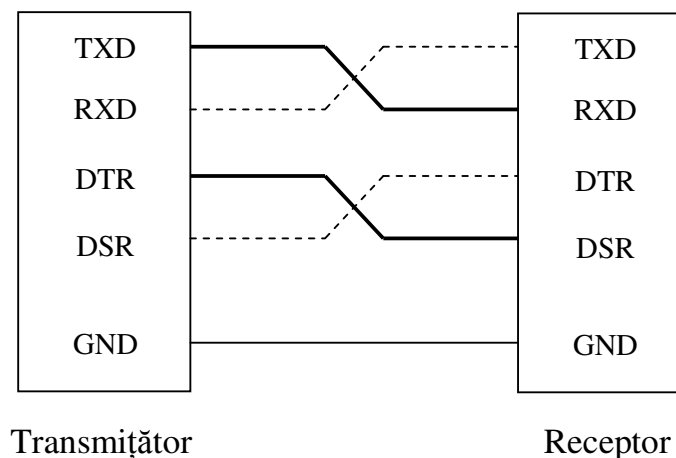
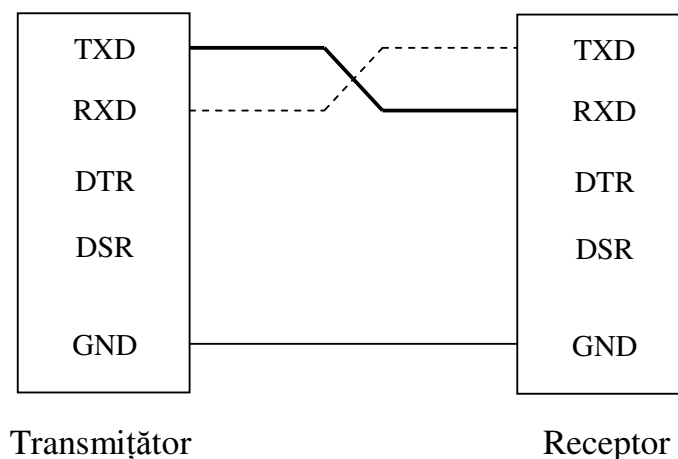


Figura 5. Conectarea punctelor de comunicație în cazul protocolului hardware

B. **Protocol software**, în care se folosesc coduri XON/XOFF integrate în structura blocurilor de date pentru pornirea, respectiv oprirea fluxului de date. XON și XOFF au aceeași semnificație ca și semnalele DTR, respectiv DSR din cazul protocolului hardware, cu deosebirea că ele nu sunt semnale fizice, ci sunt generate prin software.



În tabelul de mai jos s-a indicat numele și semnificația celor mai importante semnale definite de standardul RS-232. De asemenea s-a indicat poziția acestor semnale pe un conector de 25 pini și pe unul de 9 pini. Direcția este indicată între calculator (DTE) și modem (DCE) [1].

RI	Ring – sonerie	←	22	9
CD	Carrier Detect – detecție purtătoare	←	8	1

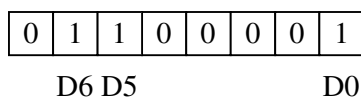
Observație : Pentru controlul fluxului de date transmise se pot utiliza atât semnalele DTR/DSR, cât și semnalele RTS/CTS. Prin ambele modalități, unitatea receptoare poate să oprească temporar fluxul de date transmis, sincronizind frecvența de transmisie a datelor la viteza de prelucrare a receptorului.

Interfața serială la calculatorul IBM PC

Pentru implementarea interfeței seriale la calculatorul IBM PC se folosește un controler specializat care îndeplinește următoarele funcționalități: serializarea datelor de ieșire, generarea informațiilor de control (biți de start, stop, paritate), deserializarea datelor recepționate, generarea semnalelor de control conform standardului utilizat, păstrarea temporară a datelor recepționate și a celor ce urmează a fi transmise, și altele.

Controlerul are în componența sa mai multe registre, dintre care amintim aici trei registre mai importante (lista completă a registrelor apare în tabelul de la sfârșitul lucrării) :

- **registru de date** pentru depozitarea datelor de pe linia de comunicație, atât la transmițător cât și la receptor ;
- **registru de control** pentru setarea parității (pară sau impară), a numărului de biți de stop (1 sau 2) și a numărului de biți dintr-un bloc de date (5-8 biți) ;
- **registru de stare**, ce indică starea registrelor de date. Astfel, bitul D0=1 indică registru de recepție plin (s-a primit un caracter), iar biții D6D5 = 11 indică faptul că registrul de transmisie este gol (s-a transmis caracterul curent) :



În mod uzual un calculator compatibil IBM PC conține două canale seriale plasate începînd de la adresele 2F8 și 3F8. În faza de inițializare a sistemului se testează prezența interfețelor seriale, și în caz afirmativ adresele canalelor detectate se înscriu în memoria RAM, începînd de la adresa 40:00. Sistemul de operare va asigura nume logice COM1 și COM2 adreselor de canale seriale în ordinea în care ele sunt prezente la adresa 40:00. Un transfer serial se poate realiza atât prin controlul direct al programului, cât și prin întreruperi.

Algoritmii de transmisie pe linie serială

Presupunem că avem de transmis pe linie serială o secvență de caractere oarecare, de la un calculator transmițător la un calculator receptor.

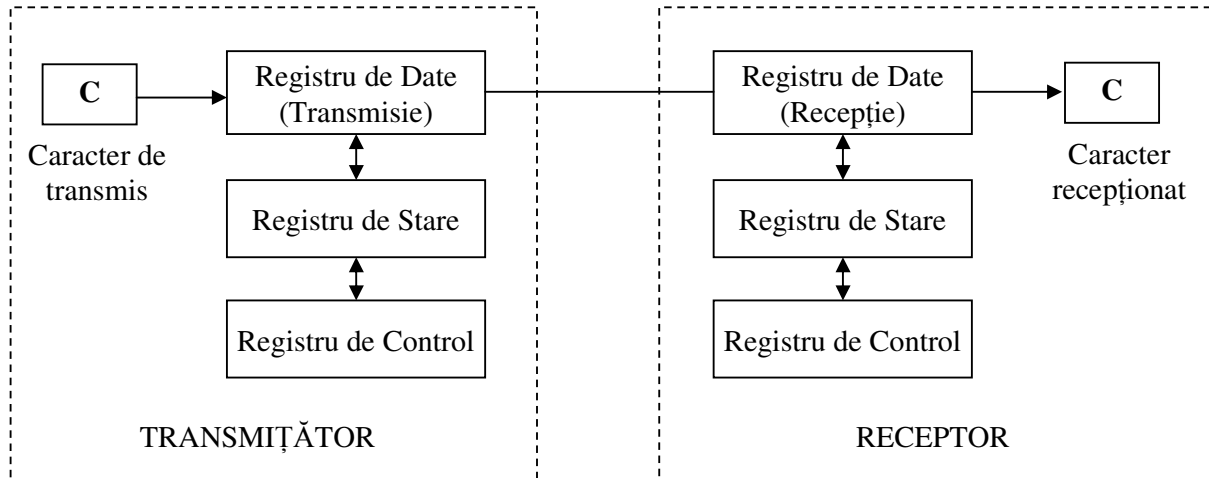


Figura 7. Registrele interfețelor seriale la transmițător și receptor

Algoritmii de transmisie sunt următorii :

A. Algoritmul pentru Transmițător :

```
Inițializare_linie( ) ;
for (each Caracter)
{
    // Bucla de așteptare pentru punerea unui caracter în registrul de transmisie
    do{
        Citire_Registru_Stare( ) ;
    } while(Registru_Transmisie_Plin( ) )

    Registru_Transmisie = Caracter ;
}
```

În algoritmul de mai sus, fiecare caracter trebuie plasat mai întâi în registrul de transmisie. Pentru aceasta, acest registru trebuie să fi trimis deja caracterul precedent, și abia după aceea îl putem încărca cu caracterul următor. Starea registrului de transmisie o putem afla prin interogarea registrului de stare. Când registrul de transmisie e gol, plasăm caracterul curent în acest registru. În același timp, interfața asigură punerea caracterului și pe linia serială, deci caracterul este transmis.

B. Algoritmul pentru Receptor :

```
Inițializare_linie( ) ;  
for (each Caracter)  
{  
    // Bucla de așteptare pentru recepția unui caracter  
    do{  
        Citire_Registru_Stare( ) ;  
    }while(Registru_Recepție_Gol( ) )  
  
    Caracter = Registru_Recepție ;  
}
```

Algoritmul pentru receptor este invers celui pentru transmițător. Acum se așteaptă ca registrul de recepție să fie plin, cu alte cuvinte să recepționeze caracterul curent de pe linie. Din nou este necesară interogarea registrului de stare. Când registrul de recepție a primit caracterul, acesta se va memora în calculatorul receptor.

Bibliografie

1. Gorgan D., Sebestyen G., „Proiectarea Calculatoarelor”, Ed. Albastră, Cluj-Napoca, 2005
2. Lupu, E., „Sisteme cu Microprocesoare”, Ed. Albastră, Cluj-Napoca, 2003
3. Todorean G., „Rețele de calculatoare : îndrumător de laborator”, Ed. Risoprint, Cluj-Napoca, 1999

Anexa 1. Registrele interfeței seriale

În tabelul de mai jos se prezintă registrele interfeței seriale pentru un calculator compatibil IBM PC-AT (s-au indicat adresele pentru portul 3F8).

Adresa	Tip registru	Tip operație	Descriere registru
3F8	Registru de date	citire	conține data recepționată (registru de recepție)
		scriere	conține data ce trebuie transmisă (registru de transmisie) dacă XFB.bit7=1 atunci conține partea mai puțin semnificativă a divizorului de frecvență
3F9	Registru de validare întrerupere	scriere	D3=1- validare întrerupere generată de modificarea stării modemului (CTS, DSR, RI) D2=1 – validare întrerupere generată la modificarea stării liniei (eroare sau break) D1=1 – validare întrerupere generată la golirea registrului de transmisie D0=0 – validare întrerupere generată la recepția unui caracter dacă XFB.bit7=1 atunci conține partea mai semnificativă a divizorului de frecvență
3FA	Registru de identificare a întreruperii	citire	D2,1=00- modificare stare linie (eroare sau break) =01- caracter recepționat =10- registru de transmisie gol =11- modificare stare modem D0=0 – lipsa întrerupere =1 – prezență întrerupere
3FB	Registru de control linie	citire/ scriere	D7- determină modul de utilizare a registrelor XF8 și XF9 (D7=0 registru de date ; D7=1 registre pt. divizarea frecvenței de transmisie) D6- generare semnal de break (se generează zero în mod continuu)

			D4,3 – paritate (x0- fara paritate, 10-pară, 11-impară) D2- biți de stop (0-un bit, 1- doi biți) D1,0 – lungime caracter (00-5biti, 01-6 biți, 10-7 biți, 11-8 biți)
3FC	Registru de control modem	scriere	D4 – activare buclă inversă pentru testare D3- activare OUT2 D2 – activare OUT1 D1- activare RTS D0 – activare DTR
3FD	Registru de stare linie	citire	D6- registru de serializare gol (transmisie) D5- registru de transmisie gol D4- indicator break D3- eroare de cadrare D2- eroare de paritate D1- eroare de supraincărcare D0- registru de recepție plin
3FE	Registru de stare modem	citire	D7- detecție purtătoare D6- indicator sonerie D5- DSR D4- CTS

Anexa 2. Funcții BIOS utilizate în cazul transmisiei seriale

Se utilizează întreruperea BIOS INT 14h. Parametrii de intrare și ieșire sunt următorii:

Intrări: AH – funcția solicitată

DX -- numărul canalului (0-COM1, 1-COM2)

Funcții specifice :

AH=0 - inițializare canal serial

AL – parametrii de inițializare :

-D7,6,5 – viteza de transmisie (000 – 110 bauds ; 001-150 bauds, 010-300 bauds, 011-600 bauds,111-9600 bauds)

-D4,3 – paritatea (x0- fără paritate, 01 paritate pară, 11-impară)

-D2 – biți de stop (0-un bit, 1-doi biți)

-D1,0- lungimea caracterului (10- 7 biți, 11- 8 biți)

AH=1 – transmitere caracter

AL – caracterul de transmis
Ieșire : AH(bit7) !=0 – eroare ; AH(bit6-0)=cod eroare
AH=2 – recepție caracter
Ieșire : AL – caracterul recepționat
AH !=0 - eroare
AH=3 – citire stare canal serial
Ieșire : AL – stare modem
D6- sonerie (ring)
D5- DSR
D4- CTS
AH – stare linie
D7- time-out
D6- registru de serializare gol (transmisie)
D5- registru de transmisie gol
D4- detecție « break »
D3- eroare de cadrare
D2- eroare de paritate
D1- eroare de supraîncărcare
D0- stare registru de recepție (1= registru plin)