



Sisteme cu microprocesoare

Cursul 9 Proiectarea interfetelor de intrare/iesire (continuare)

Transfer prin acces direct la memorie (DMA- Direct memory access)

- De ce se utilizeaza?
 - Pentru cresterea vitezei de transfer
 - Pentru eliberarea procesorului de sarcina transferului
- Mod de implementare:
 - Circuit specializat pentru operatii de transfer – controlor DMA
 - Controlorul este programat de microprocesor, dupa care efectueaza transferul intre interfata si memorie (sau invers), fara implicarea procesorului
- Fazele transferului:
 - **Initializarea:** - procesorul programeaza transferul unui bloc de date: (controlorul DMA este slave)
 - Adresa zonei de memorie
 - Directia de transfer
 - Numarul de date transferate
 - **Transferul propriu-zis:** controlorul efectueaza transferul si asigura sincronizarea cu dispozitivul periferic (contr. DMA este master)
 - **Finalizarea transferului:** procesorul verifica corectitudinea transferului prin citirea starii controlorului DMA si a interfetei implicate (contr. DMA este slave)

Schema de conectare a controlului DMA in sistem

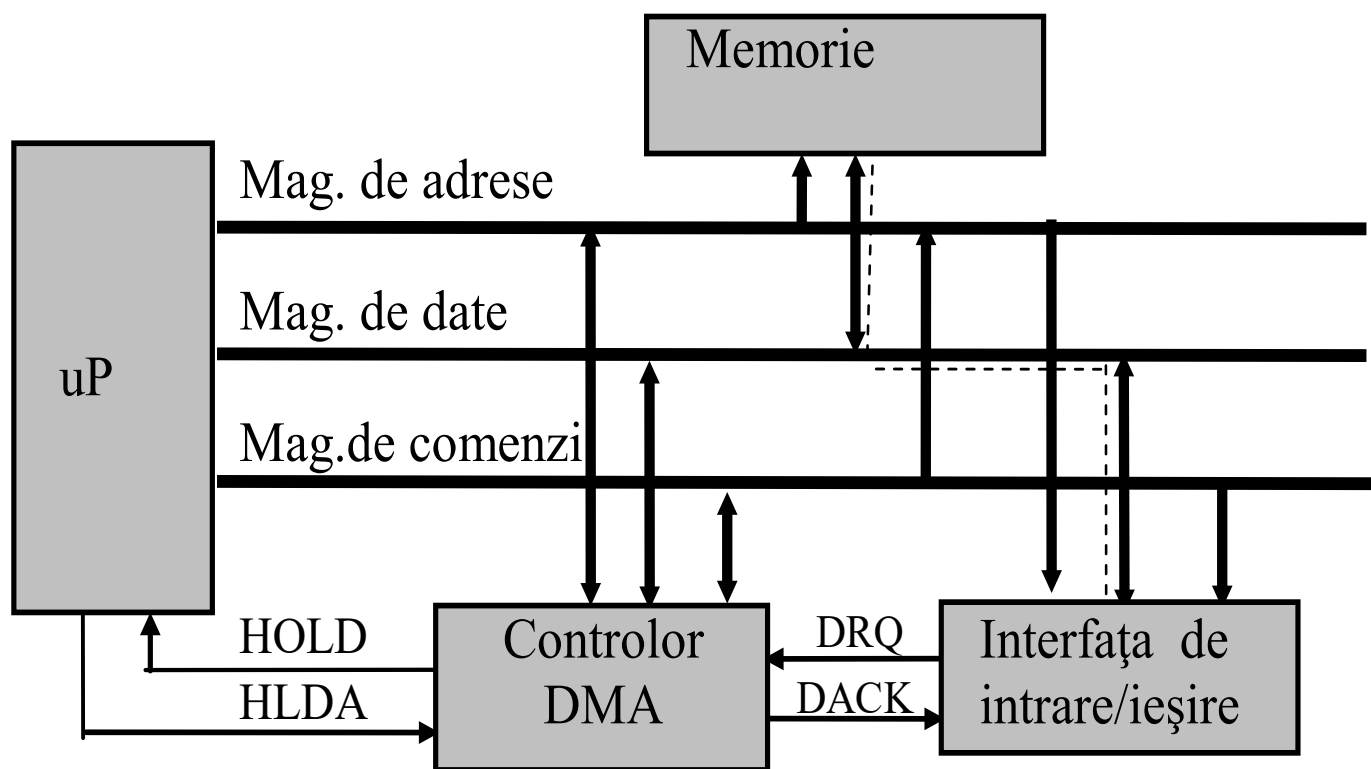


Figura 10-5. Conectarea controlului DMA în sistem

Secventa de deservire a unei cereri DMA

1. O interfata solicita un transfer prin activarea unui semnal DRQi catre controlorul DMA
2. Controlorul DMA verifica daca transferul este valid (programat) si nu este altul mai prioritar in desfasurare
3. Daca cererea este valida se solicita accesul pe magistala prin activarea semnalului HOLD catre procesor
4. Procesorul termina executia ultimului ciclu de transfer dupa care isi invalideaza amplificatoarele de magistrala si semnalizeaza cedarea magistralei prin activarea semnalului HLDA (Hold Acknowledge) catre controlor
5. Controlorul DMA isi activeaza amplificatoarele de adrese si de comanda si genereaza adresa locatiei de memorie unde/de unde se face transferul; activeaza semnalul DACK (DMA acknowledge) catre interfata
6. Controlorul DMA genereaza semnal de citire memorie (MRDC) sau citire interfata (IORC)
7. Se genereaza date de catre memorie sau de interfata
8. Controlorul DMA genereaza semnal de scriere interfata (IOWC) sau de scriere memorie (MWTC); astfel se realizeaza transferul de la memorie la interfata sau invers
9. Controlorul DMA dezactiveaza semnalele de comanda si apoi cele de adresa
10. Controlorul DMA dezactiveaza semnalul HOLD, cedand astfel procesorului controlul magistralei
11. Procesorul dezactiveaza semnalul HLDA, isi reactiveaza amplificatoarele de magistrala si preia controlul magistralei

Secventa de semnale pe magistrala

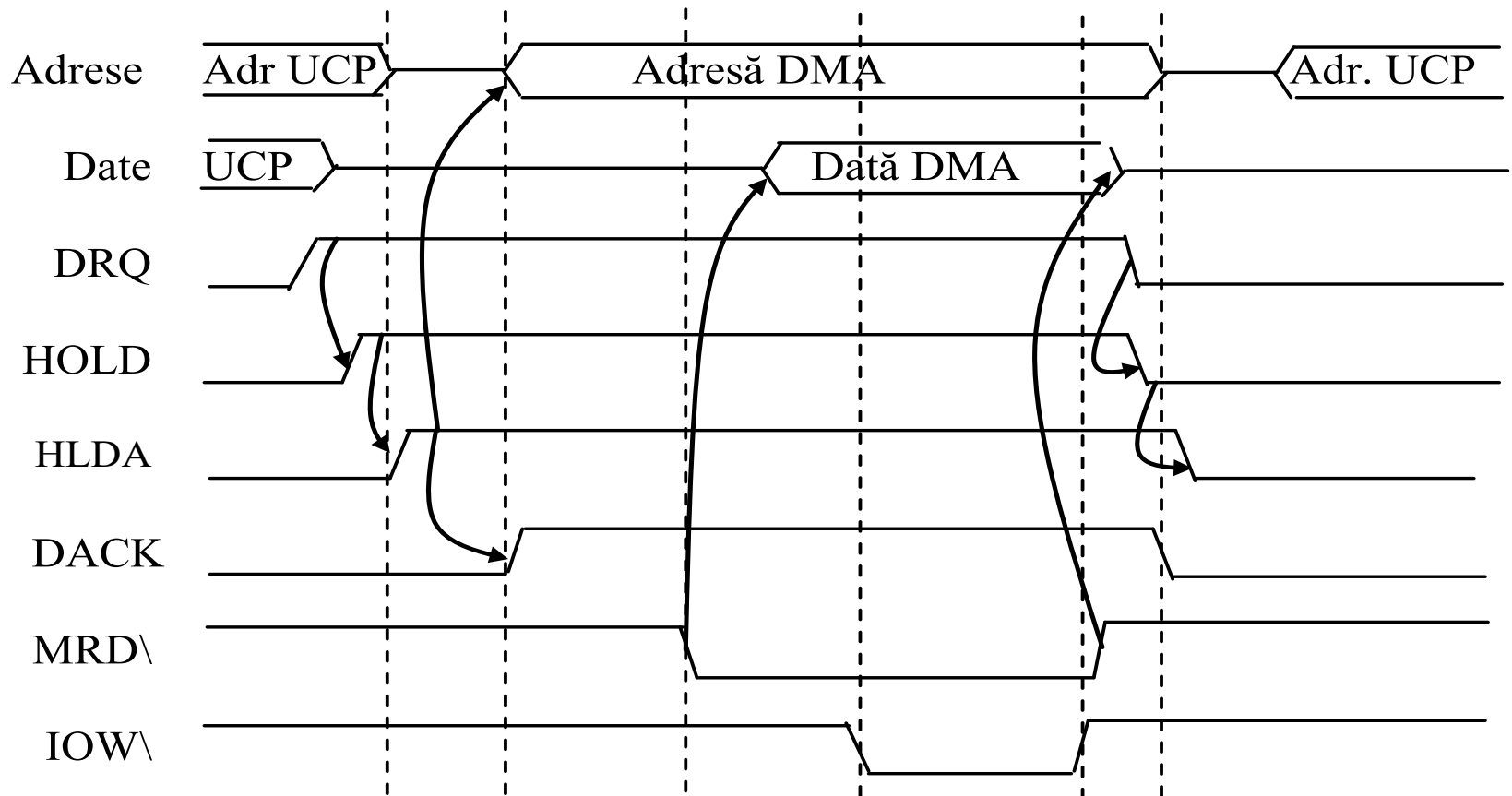


Figura 10-6. Diagrama de timp a unui ciclu de transfer DMA

Schema interna a controlului DMA I8237

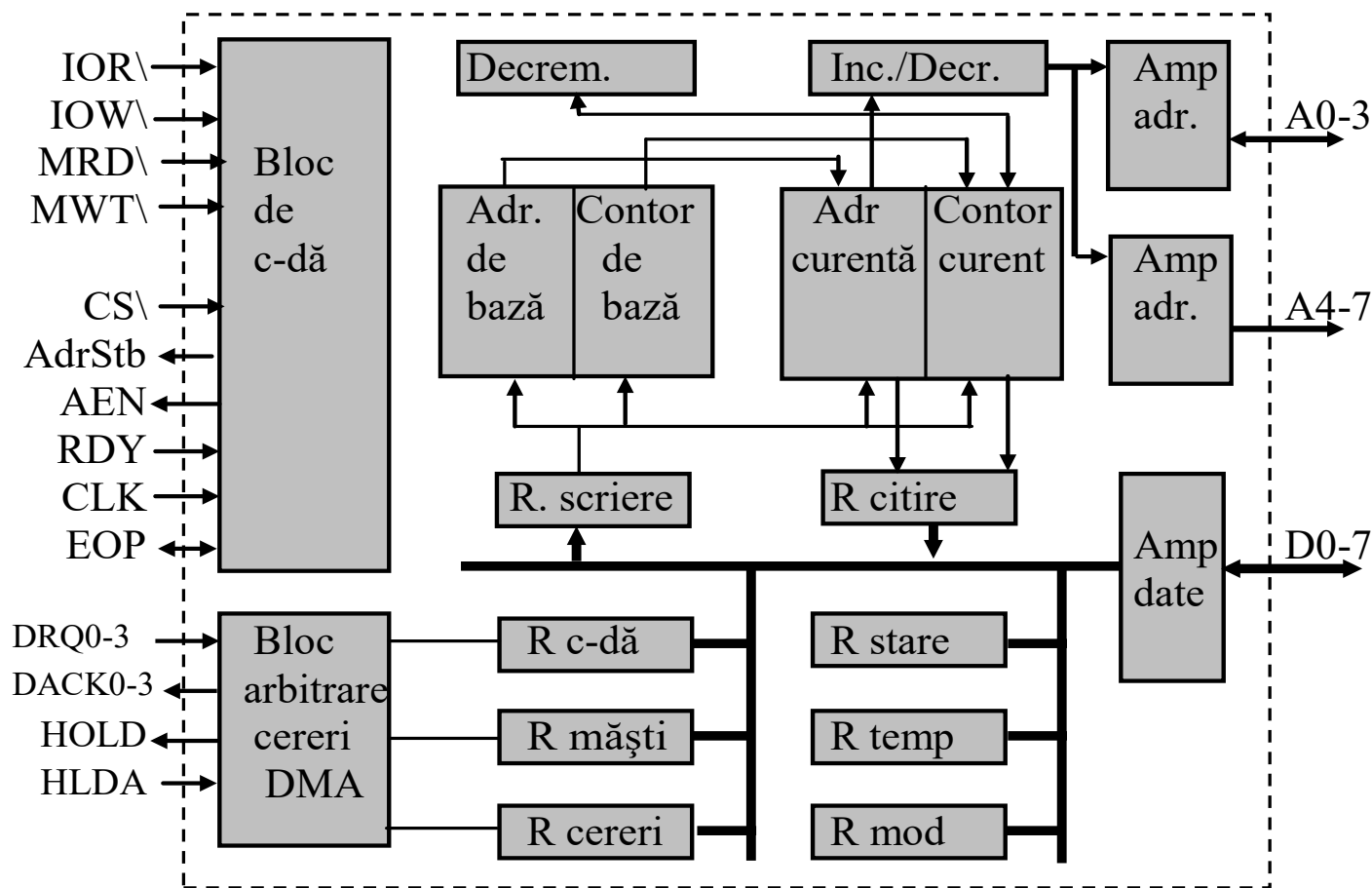


Figura 10-7. Schema bloc a controlului DMA I8237

Caracteristicile controlului DMA I8237

- poate deservi 4 periferice independente (are 4 canale DMA)
- viteza de transfer maximă este de 1,6Mocteți/s
- mai multe controloare pot fi conectate în "cascadă" pentru a mări numărul de canale DMA disponibile
- dimensiunea maximă a blocului transferat este de 64Ko
- poate efectua transfer de tip memorie-memorie
- Mai multe moduri de transfer:
 - Transfer singular
 - Transfer pe bloc (burst)
 - Transfer cu autoinitializare
 - Transfer memorie-memorie

Transfer prin procesor de intrare/iesire

- Cand se foloseste:
 - Se solicita viteza mare
 - Dispozitivul periferic este complex (are un comportament complex)
 - Se doreste eliberarea procesorului central de sarcina transferului
- Implementare:
 - Procesor de intrare/iesire +
 - Program de transfer
 - Se folosesc mecanismele de tip DMA + intreruperi pentru sincronizarea dintre procesorul central si cel de I/E
- Tipuri de procesoare utilizate pentru transfer:
 - Procesoare speciale de I/E
 - Procesoare obisnuite intr-o schema de interfata (ex: Z80)
 - Microcontroloare sau procesoare de semnal (ex: 8048, 8032, etc.)
 - Procesoare specializate pentru un anumit tip de dispozitiv (ex: procesor grafic)

Avantajele si dezavantajele transferului prin procesor de I/E

○ Avantaje:

- Rapid
- Se adapteaza usor la comportamentul dispozitivului (prin program)

○ Dezavantaje:

- Dificil de implementat si testat
- Costuri mai mari
- Apar probleme de concurenta pe magistrala sau de inconsistenta a memoriilor

Comparatie intre modurile de transfer

Transfer	Complexitate	Cost	Viteza	Implicare procesor
Prin program	Mica	Mic	Mica	totala
Prin intreruperi	Medie	Mediu	Medie	Mare
Prin DMA	Mare	Mare	Mare	Mica
Prin procesor de I/E	Foarte mare	Foarte mare	Mare	Foarte mica

Interfata seriala

- Interfete cu dispozitivele periferice:
 - Seriale – numar redus de semnale, transfer bit-cu-bit
 - Paralele – numar mai mare de linii paralele de date (de obicei 8 semnale de date + semnale de control)
- Transmisia seriala:
 - Transferul datelor se face pe un numar restrans de fire (2 sau 3 pe directia de transfer)
 - Transmisie bit-cu-bit
 - Folosita pentru transmisia la distanta a datelor
 - Realizeaza interconectarea unui calculator cu un dispozitiv periferic sau interconectarea a 2 (sau mai multe) calculatoare
 - Tipuri de transfer serial:
 - Dupa modul de realizare a sincronizarii:
 - Transfer serial sincron – controlat de semnal de ceas
 - Transfer asincron
 - Dupa modul de interconectare:
 - Punct-la-punct – conectarea a 2 echipamente
 - Multi-punct – mai multe echipamente pe acelasi tronson
 - Dupa directia de transfer:
 - Transfer unidirectional
 - Transfer bidirectional:
 - Full duplex – 2 directii independente
 - Half duplex – 2 directii, alternativ

Transfer serial sincron

- Problema: sincronizarea între emitor și receptor; când este momentul optim de citire a următorului bit
- Transferul sincron = ritmul de transfer controlat prin semnal de ceas
- Exemple:
 - I2C – magistrala serială pentru microcontroloare
 - Interfata de tastatură la PC-uri
- Avantaje: simplu, ușor de implementat, relativ sigur
- Dezavantaje: fir suplimentar pentru ceas, distanță redusă de transmisie

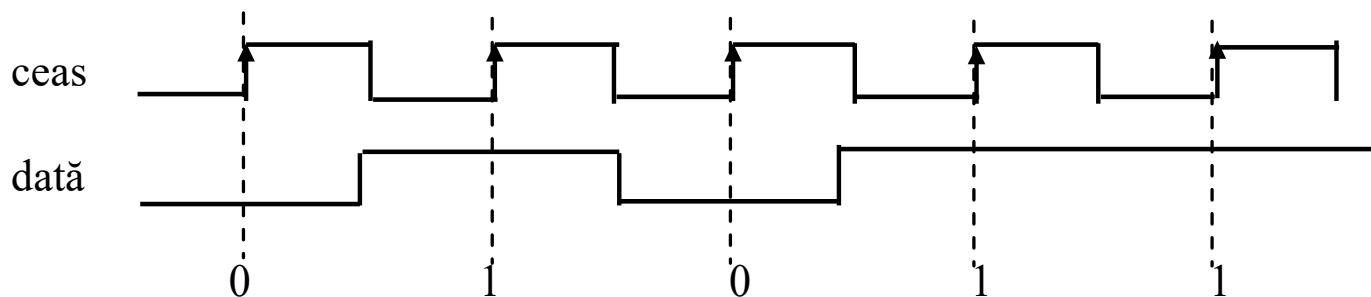


Figura 10-8. Transmisia serială sincronă

Transfer serial asincron

- Sincronizarea se face prin structura specifica a datelor transmise
- Tipuri:
 - Transfer asincron pe caracter
 - Transfer asincron pe mesaj
- Regulile de comunicatie stabilite printr-un protocol (standard) de transmisie
 - Protocolul stabileste:
 - Modul de codificare a informatiilor binare (ex: nivel de tensiune, nivel de curent, etc.)
 - Natura mediului de comunicatie (fire torsadate sau coaxiale, fibra optica, etc.); topologia
 - Structura datelor transmise
 - Metoda folosita pentru sincronizare
 - Metoda folosita pentru detectia erorilor
 - Metoda folosita pentru controlul fluxului de date

Standardul RS232 (V24)

- Codificarea informatiilor: nivele de tensiune
 - "0" – (+3..+15V) ; de obicei 12V
 - "1" – (-3..-15V) de obicei -12V
- Mediu de comunicatie: fire electrice (si telefonice)
- Structura datelor transmise:

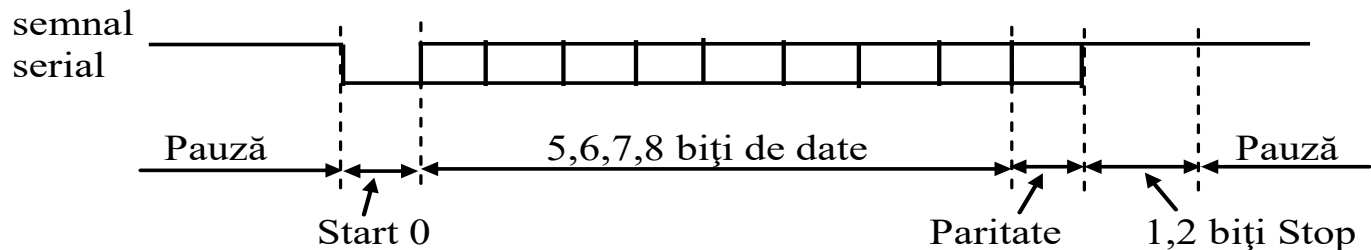


Figura 10-9. Structura unui caracter la transmisia serială asincronă

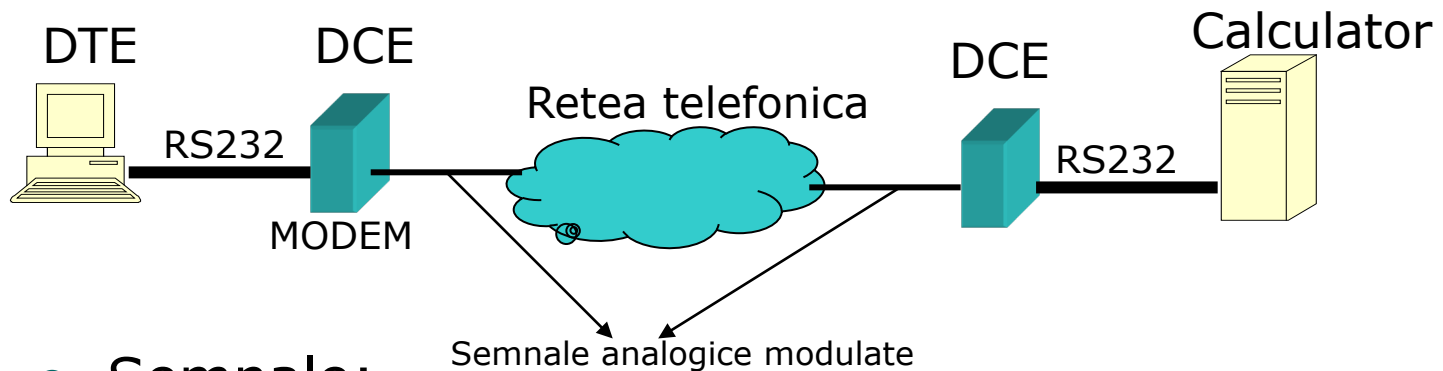
- Sincronizarea: prin bit de start, stop si frecvente prestabilite: 300,600, 1200, 2400,... 19200 Bauds
- Detectia erorilor: prin bit de paritate

Standardul RS232 (V24)

- Controlul fluxului de date:
 - Hardware – prin perechi de semnale de control:
 - DSR-DTR – Dataset ready, Data Terminal Ready
 - RTS-CTS – Request to Send – Clear To Send
 - Software: prin coduri ASCII speciale
 - XON-XOFF – Pornire-oprire transmisie
- Distanța de transmisie maximă: 100 m

Standardul RS232 (V24)

- Transmisia la mare distanta prin MODEM si retea telefonica



- Semnale:

- TXD – transmisie
- RXD – receptie
- GND – masa
- DSR – Data Set Ready
- DTR – Data Terminal ready
- RTS – Request To send
- CTS – Clear To sent
- RI - ring

Standardul RS232 (V24)

- Moduri de transmisie:
 - Transmisie unidirectională,
 - 2 fire
 - Transmisie bidirecțională, cu protocol software de control al fluxului (**XON -XOFF**)
 - 3 fire
 - Transfer bidirecțional cu protocol hard de control al fluxului (**DTR-DSR** sau CTS-RTS)
 - 5-7 fire

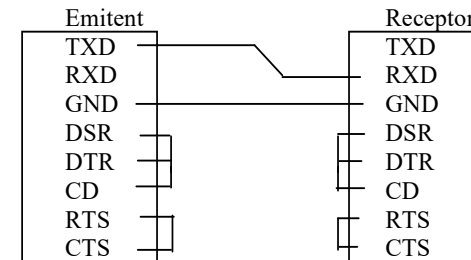


Figura 10-10 Transfer unidirecțional

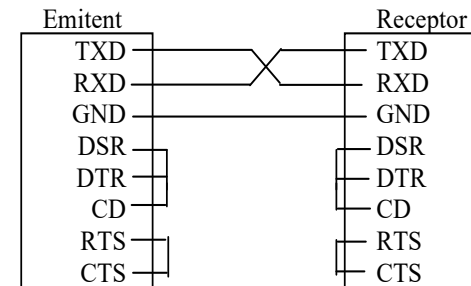


Figura 10-11 Transfer bidirecțional, protocol XO/XOFF

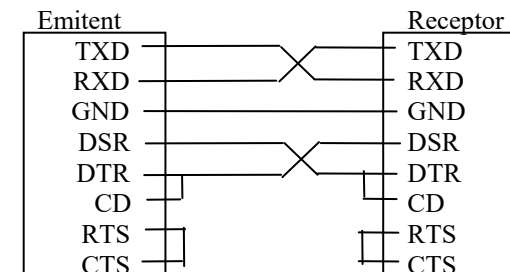


Figura 10 -12 Transfer bidirecțional, protocol DTR-DSR

Controlorul serial asincron (USART) I8251

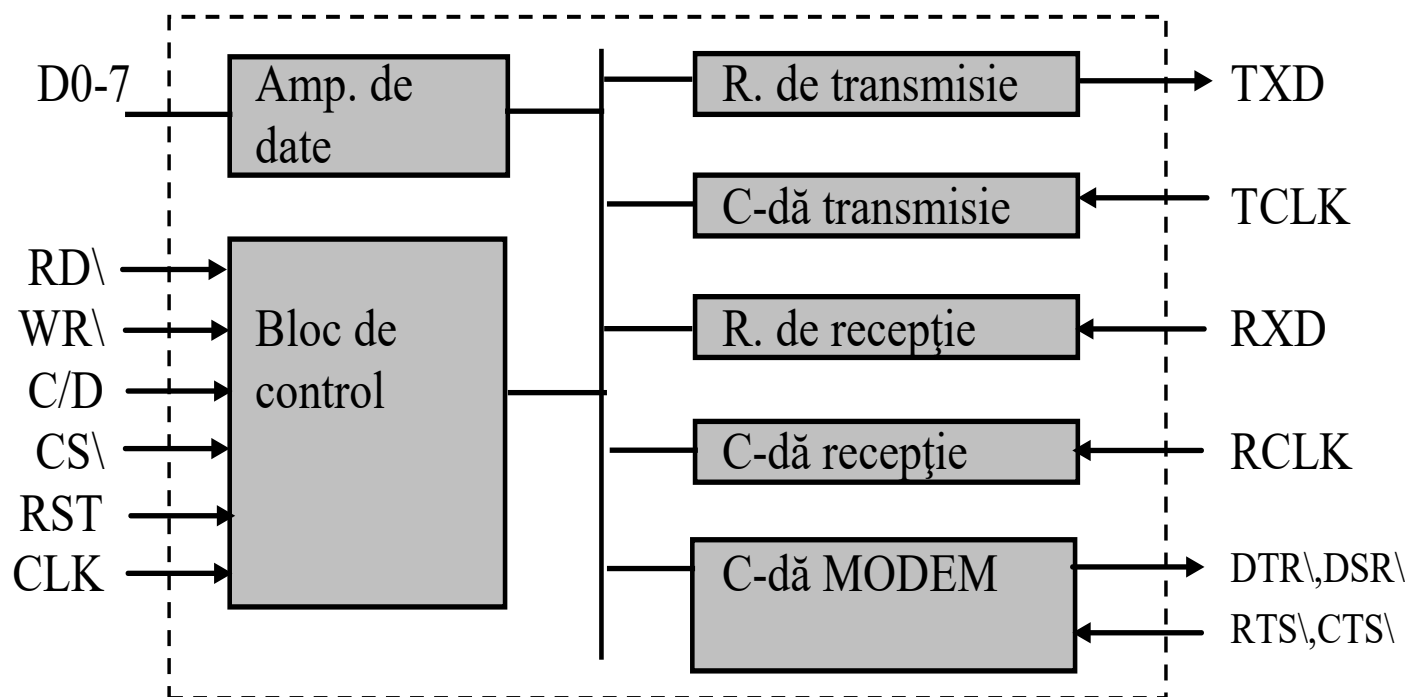


Figura 10-13. Structura internă a circuitului I8251

Alte standarde de transmisie seriala:

○ RS485:

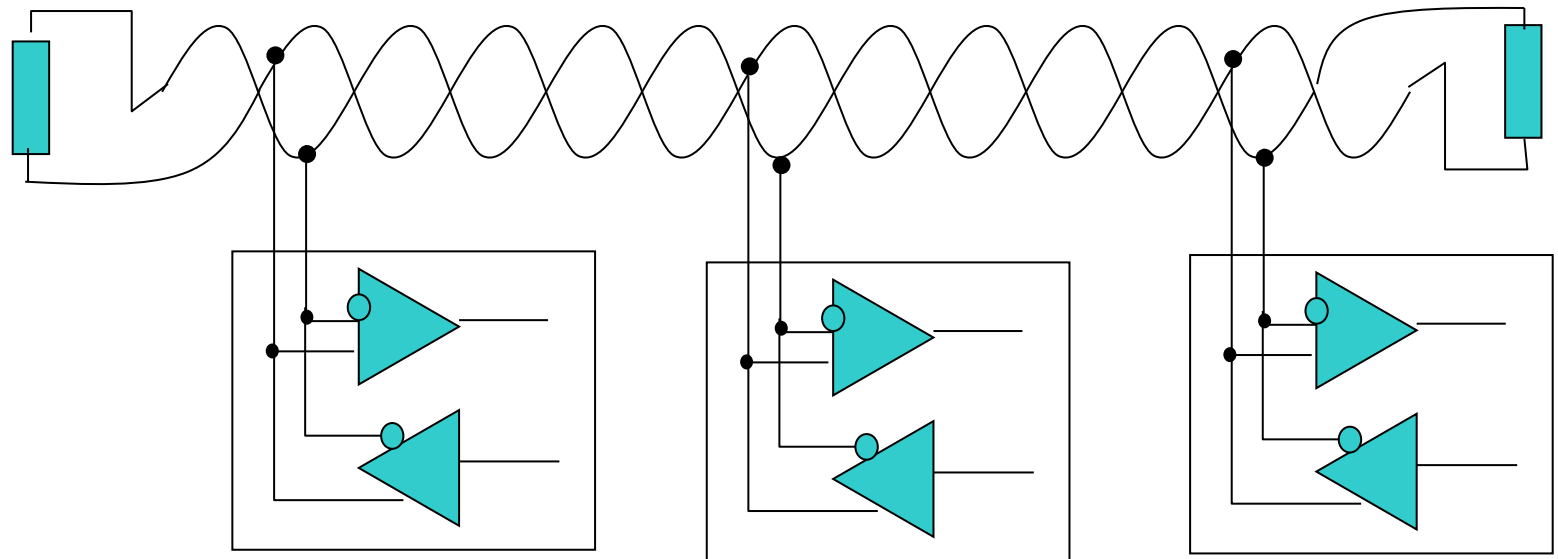
- **Caracteristici:**

- codificarea informatiilor digitale prin diferente de potential (si nu prin nivel)
- se folosesc doua fire de transmisie torsadate
- se pot conecta mai multe unitati pe acelasi tronson de cablu – transmisie multipunct
- se foloseste ca nivel fizic pentru retele industriale de comunicatie (ex: Profibus, CAN)

- **Avantaje:**

- distante mai mari de transmisie (pana la 1 km)
- imunitate mai mare la zgomote (datorita firelor torsadate si a codificari diferentiale)
- se prefera in mediile industriale

Standardul RS485



Alte standarde de transmisie seriala:

○ HDLC/SDLC

- Transmisie seriala asincrona pe mesaj
- Util pentru transmisia in retea
- Transmisie mai eficienta decat transmisia pe caracter (raportul date utile/nr. total de biti este mai bun)
- Mecanism foarte bun de detectie a erorilor (CRC)
- Sincronizare prin mecanism PLL (Phase Lock Loop)
- Structura datelor transmise:- mesaj

Preambul	Adresa	Control	Date utile	CRC
----------	--------	---------	------------	-----