

## 7 Plăcile de sistem. Magistrale și interfețe

### 7.1 Structura plăcii de bază

Una din cele mai importante componente a unui sistem de calcul este placa principală numită și placa de bază sau placa de sistem.

**Placa de sistem** (engleză: *mainboard*, *motherboard*) este o componentă hardware, de obicei complexă, pe care sunt montate toate celelalte componente hardware ale unui PC. Cu evaluarea calculatoarelor se extindeau funcțiile și se modifica structura legăturilor pe placa de bază. Au fost incorporate circuite specializate ce asigurau conectarea diferitelor componente hardware pe placa de bază cu microprocesorul și memoria. Acest grup de circuite integrate, sau cipuri, a primit denumirea de chipset. Chipset-ul constă din 2 (1) circuite, fiecare cu destinație specială, notate - puntea de nord (North bridge) și puntea de sud (South bridge). Chipset-ul conține diferite controllere ce interconectează componentele de pe placa de sistem. Pe placa de bază modernă Intel QM77, puntea de nord este incorporată în circuitul microprocesorului.

Prezentăm câteva structuri ale plăcilor de bază.

#### Arhitectură bazată pe magistrale și punți

Ca exemplu de sistem cu arhitectură bazată pe magistrale și punți poate fi privit chipset-ul AMD-760 (figura 7.1)

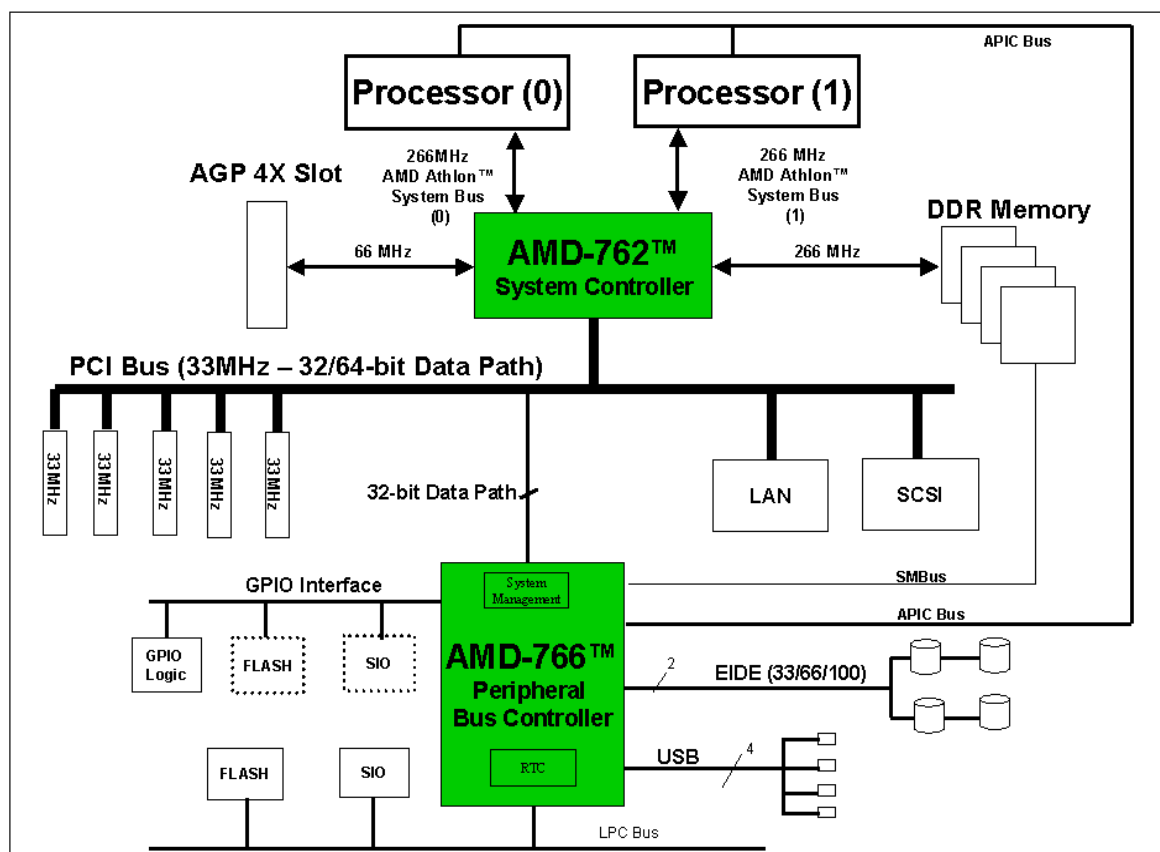


Figura 7.1 Placa de sistem bazată pe chipset-ul AMD-760

Chipset-ul AMD-760 constă din următoarele componente:

- AMD-762 – controllerul de sistem (North bridge)
- AMD-766 – controllerul magistrelor periferice (South bridge)

Controllerul de sistem AMD-762 asigură funcția de legătură (conectare) între magistralele microprocesoarelor (magistrale de sistem), magistrala AGP, subsistemul de memorie DDR și magistrala PCI.

Principalele componente ce încorporează și proprietăți ce asigură controllerul:

- a) 2 magistrale de sistem (conectarea cu microprocesoare) structurate „point-to-point” cu viteza de transfer de 266 MHz, funcționarea monoprosesor sau multiprosesarea simetrică a 2 microprocesoare;
- b) controllerul memoriei ce asigură conectarea a 4 slot-uri de tip DIMM, cu plăci de memorie de tip PC2100 DDR (Double Data Rate), cu volum de până la 4GB;

c) interfața AGP-4x (suportă modurile 1x și 2x);

d) funcționarea interfeței magistralei PCI în mod dual:

1) frecvența 33 MHz, lățimea magistralei pe 32 sau 64 biți (suportă până la 7 dispozitive);

2) frecvența 66 MHz, lățimea magistralei pe 32 sau 64 biți (suportă până la 2 dispozitive).

Controllerul magistralelor periferice AMD-766 integrează subsistemul I/O ce permite comunicarea cu dispozitivele periferice. Controllerul include:

- interfața magistralei PCI pe 33 MHz/32 biți;
- interfața magistralei LPC (Low Pin Count);
- interfața GPIO/Flash: lățimea magistralei pe 8 biți, comunicarea cu memoria Flash BIOS și dispozitivele generale (ex. butoane și întrerupătoare (switch) ) (GPIO – General Purpose I/O Devices);
- controllerul a 2 magistrale EIDE, modurile UDMA 33/66/100;
- controllerul magistralei USB (include host hub cu 4 porturi);
- controllerul de întreruperi și alte componente.

### Arhitectura bazată pe hub-uri

Viteza de transfer a magistralei PCI a devenit insuficientă odată cu introducerea modurilor rapide de transfer a datelor UltraDMA (ATA/66, ATA/100 și ATA/133) între controlerul IDE și memoria de sistem. În plus, au apărut interfețe cu viteze înalte de transfer de date Gigabit Ethernet, FireWire (100/200/400/800 Mb/s) și USB 2.0 (480 Mb/s). Ca rezultat, au fost implementate arhitecturi bazate pe hub-uri. În acest context, hub-uri sunt microcircuite specializate, destinate asigurării transferului de date între interfețele sale externe. Aceste interfețe sunt interfețele ce asigură conectarea microprocesoarelor, modulelor de memorie, magistralelor de extensie și interfețele perifericelor (ATA, SATA, USB, FireWire, Ethernet). Practic, chipset-ul este implementat din două hub-uri (North și South) interconectate printr-un canal (magistrală) cu viteza înaltă de transfer de date. North hub execută aceleași funcții ca și North bridge din arhitectura bazată pe magistrale și punți: interconectează magistralele microprocesorului, memoriei și AGP. În partea de „sud” al acestui hub nu se utilizează magistrala PCI ci o interfață și magistrala DMI (Direct Media Interface) de mare viteză de transfer ce întreconectează ambele hub-uri (North și South) (figura 7.2). În figura 7.2 este prezentat chipset-ul Intel 915 GM.

North hub, notat GMCH (Graphics Memory Controller Hub) determină principalele caracteristici ale plăcii de sistem:

- modelul microprocesorului, frecvența magistralei de sistem (FSB).
- Tipurile de memorie și frecvența magistralei de memorie (pe plăcile de sistem cu microprocesoare cu controller al memoriei încorporat, caracteristicile memoriei (tip, numărul de canale, frecvența) sunt determinate de microprocesor).
- Volumul maximal de memorie.
- Unul sau două canale de memorie.
- PCI Express x16 pentru conectarea cartelei grafice.
- 2 porturi SDVO (Serial Digital Video Out), LVDS Interface, CRT, TV-Out – conexiuni pentru dispozitive seriale și digitale externe cu diferite standarde DVI, LVDS, HDMI, TV-out, etc.

South hub (ICH6 - Intel I/O Controller Hub) conectează magistralele:

- ATA (1 magistrală).
- SATA (2 porturi).
- USB 2.0 (8 porturi)
- 4 porturi PCI-E x1.
- Interfața AC-link (Audio kodek).
- Magistrala PCI.
- Magistrala LPC (memoria CMOS, flash BIOS, mouse, tastatura).

În South hub este încorporat timer-ul (8254), controllerul de întreruperi (APIC), controllerul DMA.

Principiul de funcționare a controllerului DMA este prezentat în figura 7.3.

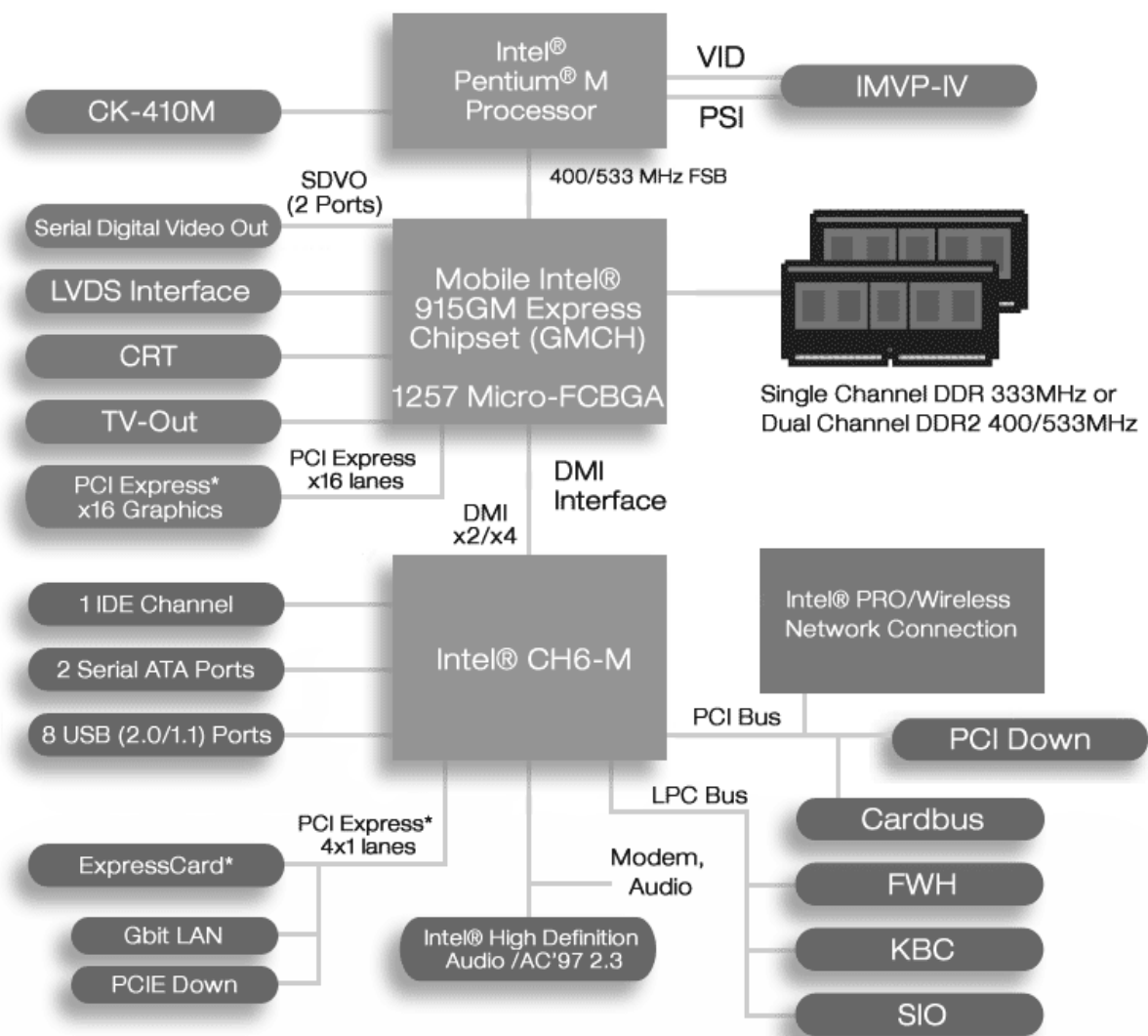


Figura 7.2 Chipset - ul Intel 915 GM

Controllerul DMA (DMA-direct memory access, acces direct la memorie) este un dispozitiv ce asigură accesul direct la RAM a perifericelor, neutilizând registrii interni ai microprocesorului.

Cererea de acces este formulată pe un terminal specializat denumit, de regulă, "BUSRQ" (cerere de magistrală). Are prioritatea maximă și răspunsul așteaptă doar terminarea ciclului mașină curent.

Microprocesorul "îngheață" întreaga activitate internă; magistralele sale trec în "impedanță înaltă" iar controlul transferului de informație este preluat de controllerul DMA. Acesta facilitează transferul informației direct între memorie și echipamente periferice (Figura 7.3).

Controllerul de DMA poate fi extern sau intern unui microprocesor standard. El adresează blocuri de date în memorie (succesiv, între limite prestabilite) și comandă transferul la sau de la un periferic prestabilit.

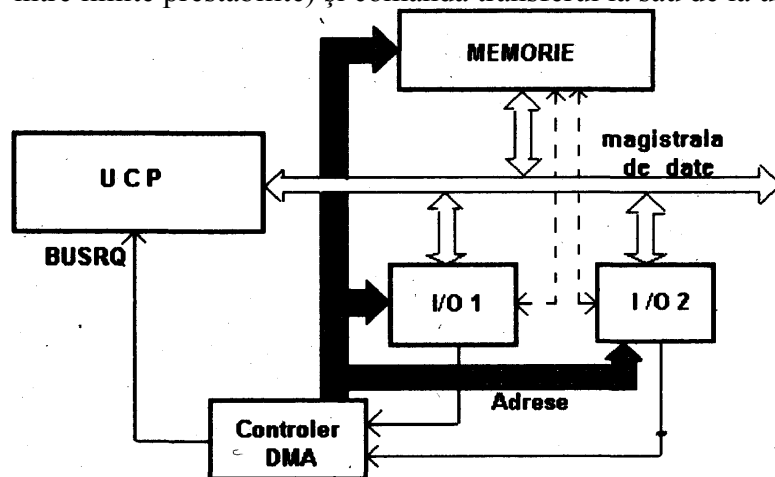


Figura 7.3

Activitatea de DMA este luată drept etalon pentru viteza maximă de circulație a datelor pe magistrala unui calculator (microprocesorul nu participă și deci adresarea memoriei și perifericelor nu este afectată de timpul de calcul al adresei fizice).

### Arhitectura HyperTransport

Tehnologia (arhitectura) HyperTransport (HT) a fost proiectată ca alternativă arhitecturii bazate pe magistrale și punți. Tehnologia a fost elaborată de firmele AMD, Apple Computers, Broadcom, Cisco Systems, NVIDIA, PMC-Sierra, SGI, SiPackets, Sun Microsystems, Transmeta.

Principala inovație – înlocuirea conectării prin intermediul magistrelor a dispozitivelor periferice prin conectarea punct-la-punct (point-to-point) ce presupune transferuri de date concomitent în ambele direcții între 2 noduri (similară cu topologia PCI-e, detaliată ulterior). Prin această tehnologie se majorează considerabil viteza de transfer a datelor. Schema structurală a arhitecturii HT este prezentată în figura 7.4. Puntea de bază (host bridge) asigură legătura cu nucleul sistemului – microprocesorul și memoria. Controllerele perifericelor, ce necesită rate înalte de transfer de date sunt realizate în formă de tuneluri HT (HT Tunnel). Structura conectării controllerului grafic AGP și a magistralei PCI-X la magistrala HT, organizate ca tunel, este prezentată în figura 7.5, 7.6.

Caracteristicile principale ale tunelului sunt:

- Interfața HyperTransport pe 16 biți (Side A) oferă o rată maximală de transfer (lățime de bandă) – până la 6,4 GB/s;
- Interfața HyperTransport pe 8 biți (Side B) oferă o rată maximală de transfer – până la 1,6 GB/s.

În arhitectură este prevăzută și conectarea prin punte cu magistrala PCI.

Arhitectura HT asigură toate tipurile de tranzacții a microprocesorului și dispozitivelor PCI, PCI-X și AGP. Tranzacțiile se efectuează în formă de pachete de diferite tipuri. Inițiatorul transferului transmite pachetul-cerere și datele necesare tranzacției scriere, dispozitivul țintă trimite pachetul-confirmare și date pentru tranzacții de citire.

Semnalizarea întreruperilor în HT este realizată tot pe pachete: dispozitivul trimite mesajul – execută tranzacția scriere pe adresa indicată la configurare. Rutina de tratare a întreruperilor trimite mesajul de finalizare – întrerupere prelucrată (End Of Interrupt, EOI), executând o scriere pe altă adresă aferentă dispozitivului dat.

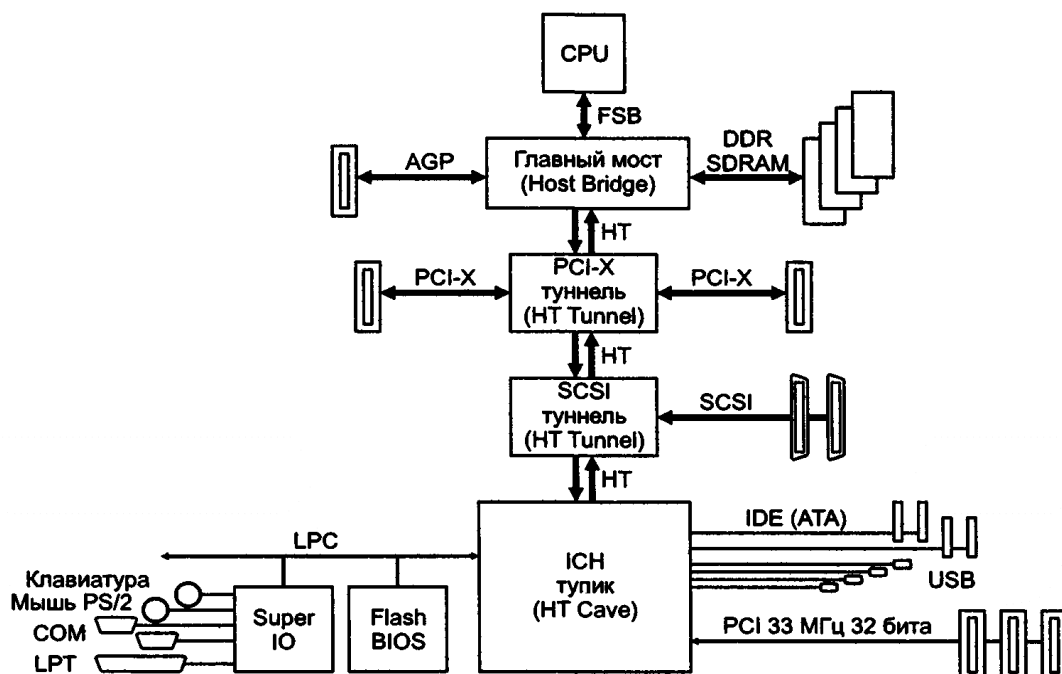


Figura 7.4 Arhitectura HyperTransport

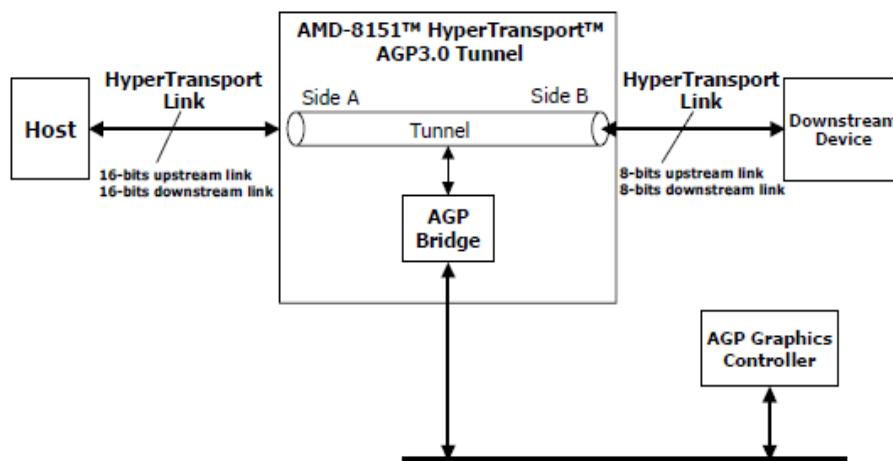


Figura 7.5 Structura conectării controllerului grafic AGP la magistrala Hyper Transport

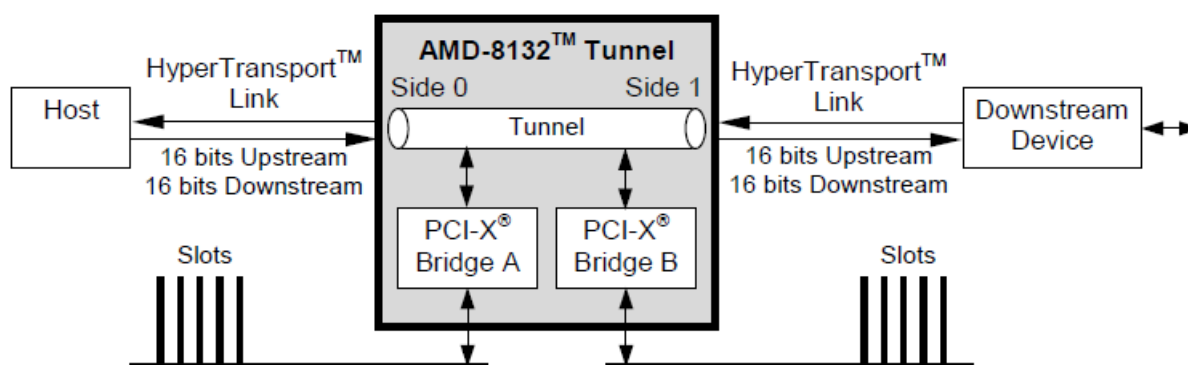


Figura 7.6 Conectarea Magistralei PCI-X la magistrala HT

Arhitectura HT se bazează pe transferuri bidirecționale (pe linii diferite) a pachetelor de date între 2 dispozitive. Dispozitivul HT poate fi ca inițiator sau/și ca ținta tranzațiilor. Sunt diferite tipuri de dispozitive HT din punct de vedere topologic:

- Tunel (Tunnel) – dispozitiv cu 2 interfețe HT; ele pot forma un lanț (daisy chain), organizând o magistrală logică. Lanțul de dispozitive se conectează la host (microprocesorul cu Host bridge), ce asigură configurarea tuturor dispozitivelor și gestionează funcționarea HT.
- Punte (Bridge) – dispozitiv ce conectează o magistrală logică primară cu una sau câteva magistrale logice secundare (lanțuri). Puntea conține un set de registre, informația din care permite gestionarea tranzațiilor între aceste magistrale.
- Comutator (Switch) – dispozitiv cu câteva interfețe HT (analogic magistralei PCI cu câteva punți) conectate la o magistrală internă.
- „Terminus”, sau cavernă (Cave) – dispozitiv cu o interfață HT.

Principala variantă topologică – lanț din dispozitive-tuneluri conectat la Host Bridge. Interfața HT constă din 2 componente independente: emițător și receptor. La configurare fiecărui dispozitiv i se atribuie un spațiu de adresare propriu. Pachetele se livrează în următorul mod: pachetul trimis de un dispozitiv inițiator din lanț este primit, decodificat și retransmis de Host Bridge dispozitivului țintă din lanț, iar dacă se cere acces direct la memoria RAM, Host Bridge formează legătura cu controllerul memoriei.

Din figura 7.5 se observă că magistrala HT (Hyper Transport Link, Side A) constă din 2 submagistrale: o magistrală pe 16 biți – direcția transferurilor spre Host și o magistrală similară destinată transferurilor de date de la Host. Un bit al magistralei reprezintă o pereche de fire diferențiale (16 biți-32 fire) cu impedanța de 100  $\Omega$ , semnale LVDS (Low Voltage Differential Signal) – nivel 1,2 V. Frecvența 200, 300, 400, 500, 600, 800, 1000 MHz.

În microprocesoarele AMD pe 64 biți, ce folosesc arhitectura HT, Host Bridge este incorporat în microprocesor. Ca rezultat microprocesorul posedă 2 interfețe: interfața memoriei și HT în calitate de magistrală de sistem. În chipset-urile VIA, SiS la interfața HT este conectat numai North Hub, ce conectează numai interfața grafică – AGP sau PCI-E. South Hub se conectează la North Hub utilizând o interfață proprie.

În următoarele capitole vom detalia principalele componente de pe placa de bază.

## 7.2 Magistrale

Detaliem noțiunea de magistrală menționată în capitolul 3 (figura 7.7).

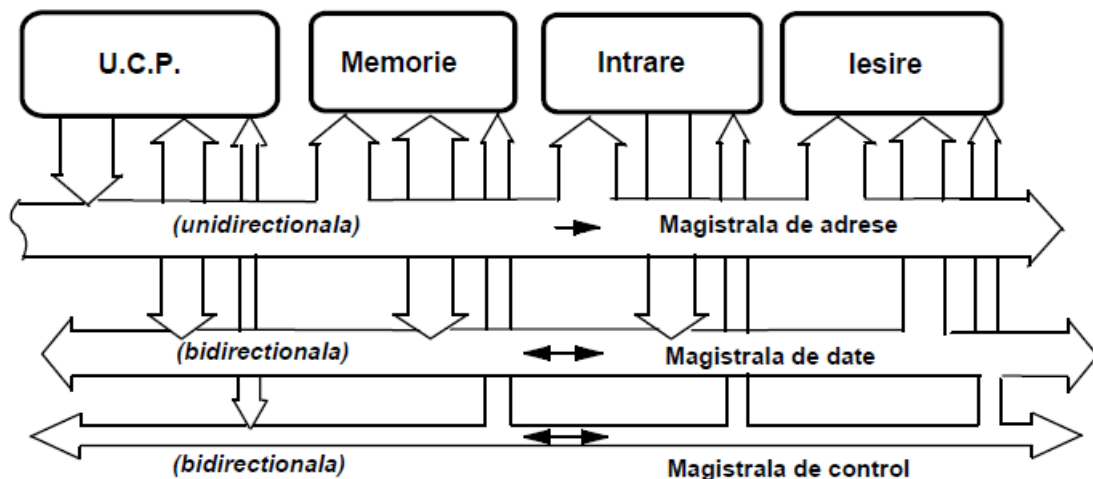


Figura 7.7 Magistralele PC

Magistrala este destinată schimbului de informație între 2 sau mai multe dispozitive. Magistrala conține componente (conectori, sloturi) la care se conectează diferite dispozitive. În arhitectura magistralei intră un controller ce gestionează transferurile de informație pe magistrală.

Sunt 2 tipuri de magistrale: paralele și seriale.

Magistrala paralelă conține un set de linii permițând transferul simultan al unui cuvânt (cuvânt pe 8, 16, 32, 64, 128, 256, 512 biți) de informație. Pe magistrala serială informația se transmite succesiv (un cuvânt este transmis bit cu bit, pe aceeași linie).

Magistrala paralelă poate fi subdivizată în trei categorii, după tipul informației transferate: *magistrale de adrese*, *de date*, și *de control*.

### Destinația liniilor magistralei paralele

Magistrala asigură conectarea paralelă a dispozitivelor periferice și asigură transferurile de informație între ele.

Arhitectura magistralei include următoarele componente:

- Linii pentru transferuri de date (magistrala de date);
- Linii pentru adresarea datelor (magistrala de adrese);
- Linii pentru controlul datelor (magistrala de control);
- Controllerul magistralei.

Controllerul magistralei efectuează controlul procesului de schimb de date, de semnale de control și reprezintă un microcircuit sau se integrează în circuitele chipset-ului.

### Magistrala de date

Magistrala de date este destinată atât transferului *unidirecțional* de instrucțiuni de la memorie (citire a programului), cât și celui *bidirecțional*, de date între memorie, unitatea centrală și/sau interfețele de intrări/ieșiri (informația parcurge magistrala în ambele sensuri, sau de la procesor la una dintre unitățile externe acestei unități, sau de la una dintre aceste unități spre procesor). Direcția transferului de informație este supervizată de către secțiunea de *CONTROL* a microprocesorului, prin generarea de semnale specifice (citire sau scriere). Lungimea cuvântului microprocesorului determină numărul de linii de conexiune din magistrala de date (8, 16, 32, etc.).

### Magistrala de adrese

Magistrala de adrese este o magistrală unidirecțională. Ea vehiculează codul binar reprezentând locația (adresa) datei ce se va utiliza în cadrul operației ce se execută.

Adresabilitatea unui microprocesor este dată de numărul de biți ai magistralei de adrese. Un număr uzual de 65536 (64 KB) celule de memorie vor necesita 16 linii de adresă pe această magistrală. ( $2^{16}=65536$ ). În general, se vor putea adresa  $2^N$  celule de memorie prin intermediul a  $N$  linii de adresă.

### *Magistrala de control*

Această magistrală furnizează informații suplimentare necesare pentru indicarea operației ce se efectuează. Numărul de semnale de pe această magistrală depinde de numărul de semnale de control necesare pentru microprocesorul utilizat. Tipice sunt *semnalul de ceas* al sistemului (care asigură funcționarea secvențială cu o periodicitate fixă a întregului sistem microprocesor), *semnalele de citire/scriere în memorie*, *citire/scriere pentru intrare/iesire din sistem* etc.

Corelate cu magistrala de adrese, semnalele de pe magistrala de control permit selecția unică a echipamentului sau a celulei de memorie căreia de/la care, prin intermediul magistralei de date, se transmite/preia informația, conform operației ce se execută.

Principalele caracteristici ale magistralei:

- Lățimea magistralei – numărul de linii folosite pentru transferuri de informații (dacă nu menționăm lățimea componentei concrete (date, adrese, control), se presupune lățimea magistralei de date). Poate fi pe 8, 16, 32, 64, 128, 256, 512 biți.
- Viteza (rata) de transfer a informației, se determină ca produsul dintre frecvența magistralei și lățimea ei: ex. lățimea magistralei PCI 32 biți și frecvența 33 MHz  
 $(32 \text{ biți} \times 33\text{MHz}) : 8 = (1056 \text{ Mb/s}) : 8 = 132 \text{ MB/s}$

### *Interfața*

Conform concepției clasice un calculator este format dintr-o unitate centrală de prelucrare, memorie și dispozitive de intrare/iesire. Dispozitivele de intrare/iesire, numite și echipamente periferice au în general o structură proprie independentă de structura calculatorului la care se conectează (ex.: imprimanta, plotter, display, mouse, etc.). Cuplarea acestor echipamente la un sistem de calcul presupune adaptarea semnalelor specifice fiecărui echipament periferic la semnalele de pe magistrală și reglarea fluxului de date de intrare/iesire între calculator și periferic și aceste funcții sunt efectuate de interfețe.

## **7.3 Magistrale de sistem**

Magistrala de sistem este destinată schimbului de informație între CPU și alte componente ale PC. Dintre ele fac parte magistralele GTL+(AGTL+), EV6.

Magistrala de sistem GTL+(AGTL+) este magistrala de interacțiune între CPU și Chipset, a fost elaborată de Intel pentru microprocesoarele Pentium III (Pentium IV). Lățimea magistralei – 64 biți, frecvența (ceas, FSB) – 100, 133 (200, 266) MHz, viteza de transfer – 800 MB/s, 1,06 GB/s (3,2 , 4,2 GB/s).

Magistrala EV6 a fost elaborată de firma Digital Equipment pentru CPU Alpha 21264. Ea este folosită de firma AMD pentru sisteme cu microprocesoare K-7. Diferă de alte magistrale prin: validarea informației se petrece pe ambele fronturi ale semnalului de frecvență, specificarea magistralei permite majorarea frecvenței până la 377 MHz.

## **7.4 Magistrala PCI**

### *7.4.1 Magistrala paralelă PCI*

PCI (Peripheral Component Interconnect) local bus – magistrala de conectare a componentelor periferice. Această magistrală de extensie a fost destinată procesoarelor Pentium și prima versiune a apărut în anul 1992. În specificarea PCI versiunea 2.0 (1993) a fost introdusă specificarea conectorilor și cartelelor de extensie. În versiunea 2.1 (1995) a fost introdusă frecvența maximă a ceasului de 66 MHz. Descrierea dată se bazează pe textul standardului „PCI Local Bus Specification. Revision 2.2” de la 18.12. 1998, publicată de organizația PCI SIG (Special Interest Grup).

Magistrala este sincronă – validarea semnalelor se petrece pe frontul pozitiv a semnalului CLK. Frecvența nominală de sincronizare – 33 MHz. Specificarea 2.1, permite mărirea frecvenței până la 66 MHz prin acordul tuturor dispozitivelor conectate la magistrală. La o frecvență de 33 MHz, se poate obține o rată de transfer de 132 MB/s în cazul implementării pe 32 de biți, sau 264 MB/s pentru 64 de biți. Cu o frecvență a ceasului de 66 MHz, se pot obține rate de transfer de 264 MB/s, respectiv 528 MB/s.

### *Configurarea*

Dispozitivul PCI poate fi multifuncțional, adică constă din mai multe funcții. Fiecărei funcții se atribuie un spațiu de configurare de 256 bytes (Standardul PCI determină pentru fiecare slot spațiu de configurare de până la 256 registre (pe 8 bits), ce nu se referă la spațiul de memorie și nici la spațiul de intrare/ieșire). Accesul la acestea registre se petrece în faze speciale ale magistralei *Configuration Read* și *Configuration Write*. Numărul funcției (function number), în tranzațiile de configurare este transmis pe

liniile AD[10:8] (liniile AD[0:7] adresează registrii din interiorul dispozitivului PCI). Deci dispozitivul poate încorpora pînă la 8 funcții. La configurare se adresează funcția cu numărul "0".

La o magistrală PCI pot fi conectate cîteva dispozitive fiecare cu numărul său (device number). În sistemă pot fi cîteva magistrale PCI, fiecare cu numărul său (PCI bus number). Magistralele se numerează consecutiv.

### Protocolul magistralei PCI

La fiecare transfer participă două dispozitive: inițiatorul și ținta. Inițiatorul, sau dispozitivul *master*, este cel care inițiază un transfer. Ținta, sau dispozitivul *slave*, este adresat de inițiator în scopul executării transferului. Dispozitivele inițiator și țintă sunt numite *agenți PCI*.

#### *Transferuri în mod exploziv*

Un asemenea transfer constă dintr-o singură fază de adrese urmat de mai multe faze de date. Arbitrajul de magistrală trebuie executat o singură dată. În timpul fazei de adrese se transmite adresa de început și tipul tranzacției care urmează. Dispozitivul țintă memorează adresa de început într-un contor de adrese, și va incrementa adresa în fiecare fază de date.

În cazul magistralei PCI, cele mai multe transferuri se execută în mod exploziv, cele mai multe dispozitive fiind proiectate astfel încât să permită asemenea transferuri. Dacă un dispozitiv țintă poate efectua numai tranzacții singulare, la încercarea dispozitivului *master* de a executa o tranzacție în mod exploziv, ținta termină tranzacția după prima fază de date. Astfel dispozitivul *master* va fi obligat să solicite din nou magistrala printr-un proces de arbitraj. În acest mod performanțele vor fi mai reduse, dar tranzacțiile singulare se pot utiliza pentru dispozitivele care nu necesită rate ridicate de transfer.

Presupunând că nici inițiatorul și nici dispozitivul țintă nu inserează stări de așteptare, poate fi transferat câte un cuvânt la fiecare front crescător al semnalului de ceas.

### Tranzacții

#### *Faza de adrese*

Toate tranzacțiile pe magistrala PCI constau dintr-o *fază de adrese* urmată de una sau mai multe *faze de date*. Excepția o reprezintă tranzacțiile în care se utilizează adresarea pe 64 de biți, unde adresa este furnizată în două faze de adrese. O fază de adrese are durata unei perioade de ceas. În această fază, inițiatorul identifică dispozitivul țintă și tipul tranzacției. Ținta este identificat prin transmiterea unei adrese de start pe magistrala de adrese/date în cadrul domeniului de adrese asignat dispozitivului.

Tipul tranzacției este identificat prin depunerea tipului comenzii pe liniile *C/BE*. Inițiatorul activează semnalul *FRAME* pentru a indica prezența unei adrese de start și tip de tranzacție valide. Deoarece adresa de start este prezentă numai pe durata unei perioade de ceas, fiecare dispozitiv țintă trebuie să memoreze adresa pentru a putea fi decodificat ulterior.

Atunci când un dispozitiv țintă determină faptul că este adresat, trebuie să activeze semnalul *DEVSEL*, pentru a revendica tranzacția. Dacă acest semnal nu va fi activat într-un timp predeterminat, inițiatorul va abandona tranzacția. După terminarea fazei de adrese, magistrala de adrese/date este utilizată pentru transferul datelor în cadrul unei sau mai multor faze de date.

#### *Faza de date*

Pe durata fazei sau fazelor de date are loc transferul datelor între inițiator și țintă. Atât inițiatorul cât și ținta trebuie să indice faptul că sunt pregătite pentru terminarea unei faze de date prin activarea semnalelor *IRDY*, respectiv *TRDY*. În cazul în care nu sunt activate ambele semnale la începutul unei perioade de ceas, se va insera o stare de așteptare cu durata unei perioade.

#### *Terminarea tranzacției*

Inițiatorul identifică durata totală a unui transfer cu ajutorul semnalului *FRAME*. Acest semnal este activat la începutul fazei de adrese și rămâne activat până când inițiatorul este pregătit pentru a termina faza finală de date (activează *IRDY*). Executarea ultimei faze de date este indicată de inițiator prin dezactivarea semnalului *FRAME* și activarea semnalului *IRDY*. După terminarea ultimului transfer, inițiatorul readuce magistrala în starea inactivă prin dezactivarea semnalului *IRDY*.

Arbitrajul de magistrală are loc în timp ce un dispozitiv *master* utilizează magistrala. Dacă arbitrul de magistrală a acordat în prealabil magistrala unui alt dispozitiv *master*, acest dispozitiv poate detecta starea inactivă a magistralei prin faptul că semnalele *FRAME* și *IRDY* sunt ambele dezactivate.

Se descriu în continuare principalele semnale.

- *AD[31:0] (Address/Data)* - Reprezintă magistrala multiplexată de adrese și de date. În timpul fazei de adrese, pe această magistrală se transmite adresa de start a tranzacției. În timpul fazei de date, pe liniile *AD[31:0]* se transmit date provenite de la inițiator (la o scriere) sau de la ținta adresat (la o citire).



- *C / BE* [3:0] (*Command or Byte Enable*) - În timpul fazei de adrese aceste linii definesc o comandă pe care inițiatorul o utilizează pentru a indica tipul tranzacției necesare. Dintre comenzile posibile se amintesc citirea din memorie, scrierea în memorie, citirea de la un dispozitiv de I/E, scrierea la un dispozitiv de I/E, achitarea unei întreruperi etc. În timpul fazei de date aceste linii sunt utilizate de inițiator pentru a indica octeții care trebuie transferați din cadrul cuvântului dublu adresat și grupele de linii ale magistralei *AD* care trebuie utilizate pentru transferul datelor.
- *FRAME* (*Cycle Frame*) - Este activat de inițiator și indică începutul și durata unei tranzacții pe magistrală. O tranzacție poate consta din una sau mai multe transferuri de date între inițiatorul curent și ținta adresată. Semnalul *FRAME* este dezactivat atunci când inițiatorul este pregătit să încheie faza finală de date.
- *TRDY* (*Target Ready*) - Este activat de dispozitivul țintă adresat atunci când acest dispozitiv este pregătit pentru un transfer de date (poate executa faza curentă de date). Faza de date este terminat atunci când ținta activează *TRDY* și inițiatorul activează *IRDY* la frontul crescător al semnalului de ceas. În timpul unei operații de citire, semnalul *TRDY* activat indică faptul că ținta a depus date valide pe magistrala de date. În timpul unei operații de scriere, semnalul *TRDY* activat indică faptul că ținta este pregătit să accepte datele de la inițiator. Sunt inserate stări de așteptare în faza curentă de date până când ambele semnale *TRDY* și *IRDY* sunt activate.
- *IRDY* (*Initiator Ready*) - Este activat de inițiatorul tranzacției și semnalează momentul în care inițiatorul este pregătit pentru un transfer de date. În timpul unei operații de scriere, semnalul *IRDY* activat indică faptul că inițiatorul a depus date valide pe magistrala de date. În timpul unei operații de citire, semnalul *IRDY* activat indică faptul că inițiatorul este pregătit să accepte datele de la ținta adresată.
- *REQ* (*Request*) - Este activat de inițiator pentru a indica o cerere de magistrală. Această linie este conectată la arbitrul de magistrală. Metoda de arbitrare a magistralei nu este descris în specificațiile PCI; se pot implementa diferite metode. Este specificat doar faptul că arbitrul de magistrală trebuie să utilizeze un algoritm prin care să se evite blocajele; fiecărui dispozitiv *master* potențial trebuie să i se permită accesul la magistrală.
- *GNT* (*Grant*) - Este activat de arbitrul de magistrală pentru a indica acordarea magistralei pentru inițiator. Atunci când detectează acest semnal, inițiatorul trebuie să aștepte terminarea tranzacției în curs de către inițiatorul curent.
- *CLK* (*Clock*) - Reprezintă semnalul de ceas utilizat pentru sincronizarea tuturor tranzacțiilor, inclusiv a arbitrajului de magistrală. Toți parametrii de sincronizare ai magistralei sunt specificați relativ la frontul crescător al semnalului de ceas. Frecvența semnalului de ceas se poate modifica în orice moment, cu condiția să nu existe cereri de magistrală și semnalul *LOCK* să nu fie activ. De asemenea, ceasul poate fi oprit în starea *low* (pentru reducerea puterii consumate).
- *DEVSEL* (*Device Select*) - Este activat de dispozitivul țintă numai când acesta și-a decodificat adresa. Dacă un dispozitiv *master* a inițiat un transfer și nu detectează semnalul *DEVSEL* activ în cursul a șase perioade de ceas, va presupune că ținta nu poate răspunde sau că adresa respectiv nu este utilizată. Va rezulta un abandon din partea dispozitivului *master*.

#### Exemplu de tranzacție

Figura 7.8 prezintă un transfer de date reprezentativ de la un dispozitiv *slave* la un dispozitiv *master*, de exemplu o operație de citire de la un dispozitiv de I/E.

Tranzacția începe în momentul în care inițiatorul activează semnalul *REQ* (*Request*) - pentru a indica o cerere de magistrală, ce este conectat la arbitrul de magistrală. Arbitrul de magistrală pentru a indica acordarea magistralei pentru inițiator, activează semnalul *GNT* (*Grant*), iar inițiatorul trebuie să aștepte terminarea tranzacției în curs de către inițiatorul curent și să activeze semnalul *FRAME* în ciclul de ceas 1.

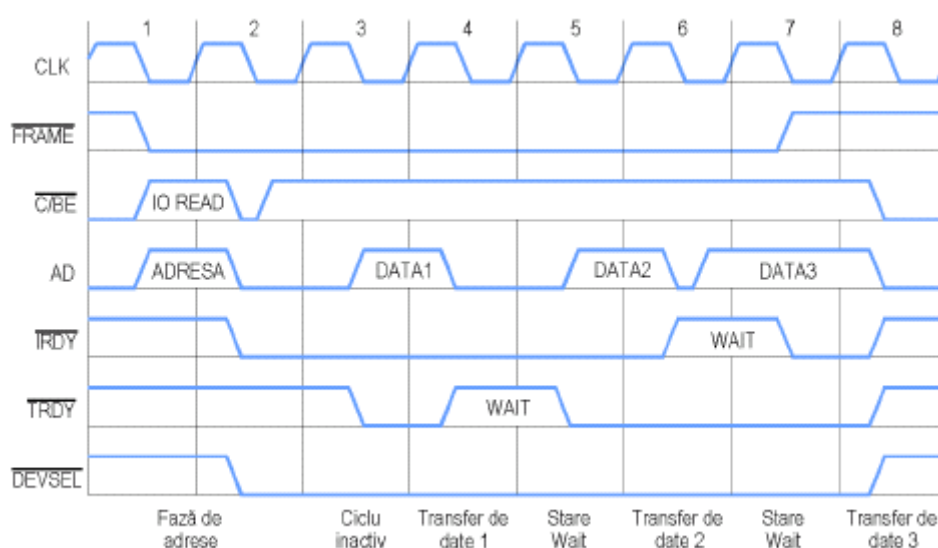


Figura 7.8 - Tranzacție de citire pe magistrala PCI

Ciclul de ceas 1. Activarea semnalului *FRAME* indică începutul unei tranzacții pe magistrală. Inițiatorul plasează o adresă pe liniile *AD* și un cuvânt de comandă pe liniile *C/BE* (IO READ în acest exemplu). Dispozitivul țintă activează *DEVSEL* - "0", pentru a indica faptul că a decodificat cu succes adresa și este ținta tranzacției curente. Nici un transfer de date nu poate avea loc până în momentul în care semnalul *DEVSEL* nu este activ.

Ciclul de ceas 2. Pentru evitarea conflictelor în momentul în care inițiatorul nu mai are controlul asupra liniilor *AD* și acest control este preluat de dispozitivul țintă, faza de adrese este urmat de un ciclu inactiv (*TRDY* - "1"). Condiția tranzacției- liniile *IRDY* și *TRDY* setate "0".

Ciclul de ceas 3. Liniile *IRDY* și *TRDY* setate "0" - se transmite cuvântul de date 1.

Ciclul de ceas 4. Dispozitivul țintă inserează o stare de așteptare (de ex. dispozitivul este mai lent). Cele două dispozitive controlează rata de transfer real prin liniile *IRDY* și *TRDY*, care permit inserarea unui număr oarecare de stări de așteptare după fiecare ciclu de transfer de date.

Ciclul de ceas 5. Dispozitivul țintă activează *TRDY* și se transmite cuvântul de date 2.

Ciclul de ceas 6. Dispozitivul inițiator inserează o stare de așteptare. Transferul de date se întrerupe, transferul cuvântului 3 de date este amânat până în ciclul 7.

Ciclul de ceas 7. Dispozitivul inițiator activează *IRDY* și se transmite cuvântul de date 3. Inițiatorul dezactivează semnalul *FRAME* pentru a indica faptul că următorul ciclu va fi final.

Ciclul de ceas 8. Toate liniile sunt dezactivate, ce marchează sfârșitul tranzacției pe magistrală.

O tranzacție de scriere (în care inițiatorul este sursa datelor) este similar cu cea din Figura 7.3. În acest caz ciclul inactiv după faza de adrese nu este necesar, deoarece inițiatorul menține în continuare controlul asupra liniilor *AD* pe durata tranzacției.

#### 7.4.2 Magistrala serială PCI Express

PCI Express este o magistrală serială ale cărei specificații inițiale au fost preluate de organizația PCI-SIG de la grupul de lucru Arapahoe (*Arapahoe Working Group*), care a elaborat specificațiile preliminare ale interfeței 3GIO (*Third Generation I/O*). Versiunea 1.0 a acestor specificații a fost terminată în aprilie 2002, după care specificațiile au fost transferate la organizația PCI-SIG, interfața 3GIO fiind redenumită PCI Express. Acest nume a fost ales pentru a sublinia compatibilitatea software a noii magistrale cu magistrala PCI convențională. Specificațiile 3GIO au fost revăzute de membrii PCI-SIG, fiind aprobate și publicate în luna iulie 2002 ca PCI Express versiunea 1.0. Scopul magistralei PCI Express este de a pune la dispoziție o interconexiune serială scalabilă, având costuri reduse și utilitate generală, care să unifice mai multe soluții de interconectare din cadrul unei platforme.

Topologia unui sistem bazat pe magistrala PCI Express conține un comutator și mai multe conexiuni punct la punct cu dispozitivele de I/E. În Figura 7.9 se prezintă topologiile conexiunilor magistralelor PCI și PCI Express. Dispozitivele de I/E conectate la magistrala PCI împart lățimea de bandă (fig. 7.9 (a)). Iar la PCI Express (fig. 7.9 (b)) conexiunile multiple punct la punct introduc un nou

element, comutatorul, în cadrul topologiei sistemului de I/E. Comutatorul înlocuiește magistrala partajată pentru conectarea dispozitivelor de I/E și permite comunicația directă dintre două dispozitive de I/E.

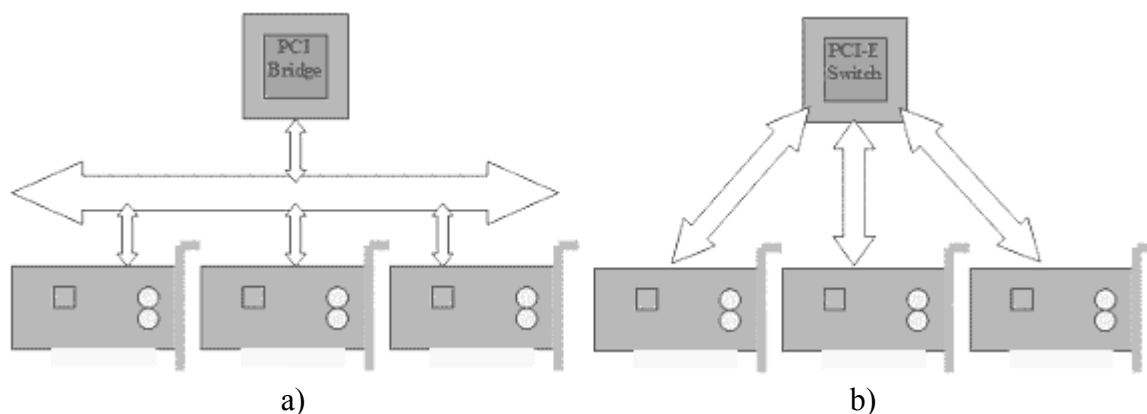


Figura 7.9 - Topologiile PCI și PCI Express

Nivelul fizic are la bază o legătură punct la punct formată din două canale de comunicație simplex, unul pentru transmisie și unul pentru recepție. Legătura fizică de bază (Figura 7.10) utilizează 4 conexiuni între două dispozitive, semnalele transmise fiind diferențiale și având tensiuni reduse (LVDS - Low Voltage Differential Signaling).

Nivelul legăturii de date adaugă numere de secvență și coduri CRC acestor pachete pentru a asigura un transfer fiabil. Semnalul de ceas este codificat împreună cu datele pentru a obține rate de transfer ridicate.

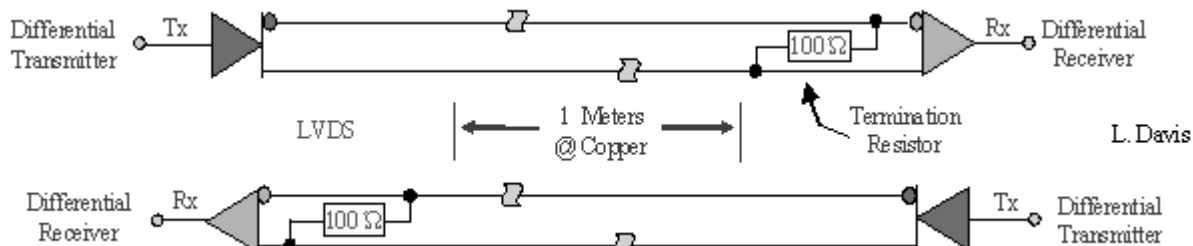
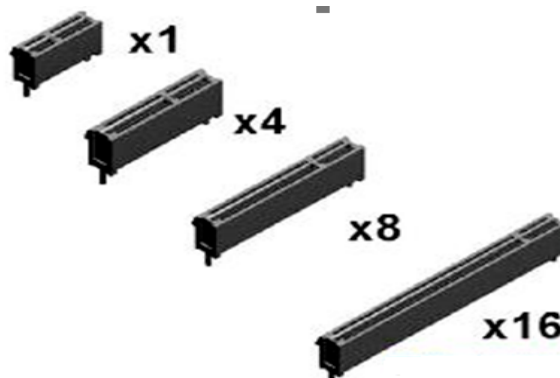


Figura 7.10 – Structura legăturii punct la punct

Rata de transfer inițială este de 2,5 Gbiți/s pentru fiecare direcție, fiind de așteptat ca aceasta să crească ulterior până la 16 Gbiți/s pe direcție.

	Raw Bit Rate	Link Bandwidth (BW)	BW/Lane/Way	Total BW x16
PCIe 1.x	2.5 GT/S	2 GB/S	~250 MB/S	~8 GB/S
PCIe 2.x	5 GT/S	4 GB/S	~500 MB/S	~16 GB/S
PCIe 3.0	8 GT/S	8 GB/S	~1 GB/S	~32 GB/S
PCIe 4.0	16 GT/S	16 GB/S	~2 GB/S	~64 GB/S

Rata de transfer a unei legături poate fi mărită liniar prin adăugarea a noi canale. Numărul de canale poate fi de 1, 2, 4, 8, 12, 16 sau 32. Viitoare îmbunătățiri pentru creșterea frecvenței de funcționare, modificarea tehnicii de codificare sau schimbarea mediului fizic vor afecta numai nivelul fizic.



La transferuri pe mai multe canale se folosește principiul partajării datelor pe canale (data stripping) (Figura 7.11) ce constă în transferul următorului byte pe următorul canal.

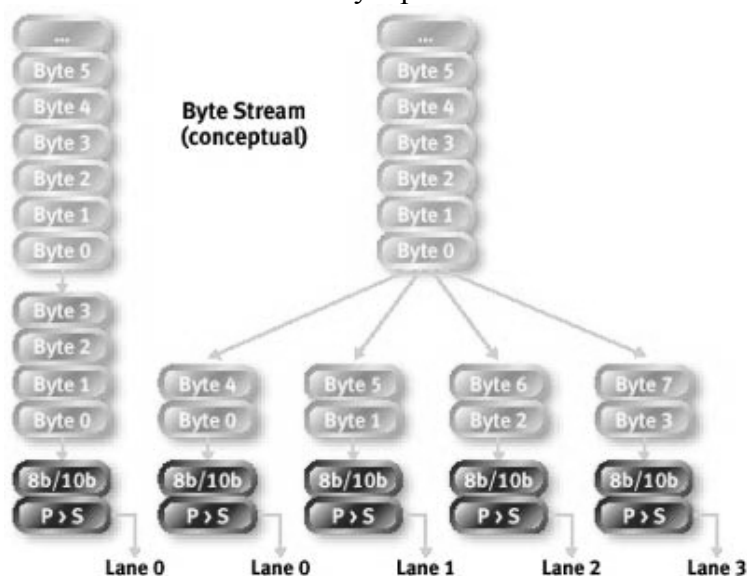


Figura 7.11 - Data stripping

La transferuri pe magistrala PCI Express se folosește algoritmul de codificare 8b/10b – fiecare byte de informație este transmis ca 8 biți + 2 biți de control = 10 biți.

### Protocolul PCI Express

Formatul unui cadru este prezentat în figura Figura 7.12. El constă din: 1byte - Start-of-Frame, 2 byte- Numărul pachetului, 16-20 bytes – identificatorul pachetului, de la 0-4096 bytes - Data field, 0 - 4-bytes câmpul ECRC(End-to-end Cyclic Redundancy Check), 4- bytes LCRC(Local Cyclic Redundancy Check), 1- byte End-of Frame.

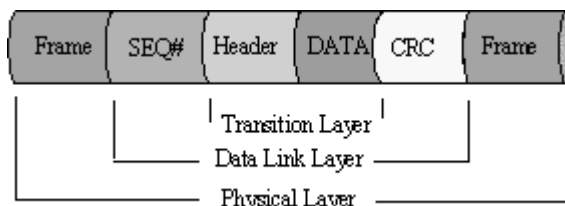


Figura 7.12 - Formatul unui cadru

Urmând acest format, pentru a transfera 4096 bytes, cadrul este format din 4124 bytes.

Nivelul legăturii de date are ca rol principal asigurarea transmiterii fiabile a pachetelor. Acest pachet creat de nivelul tranzacțiilor. Un protocol special asigură ca un pachet să fie transmis numai atunci când este disponibil un buffer pentru recepția acestui pachet la destinație. Aceasta elimină retransmisia pachetelor datorită resurselor limitate. Un pachet va fi retransmis automat de nivelul legăturii de date atunci când a apărut o eroare la transmiterea acestuia.

Nivelul tranzacțiilor primește cereri de citire și scriere de la nivelul software și creează pachetele care vor fi transmise de nivelul legăturii de date. Toate cererile de transfer sunt implementate ca tranzacții divizate, unele pachete de cerere necesitând un pachet de răspuns. Nivelul tranzacțiilor recepționează de asemenea pachetele de răspuns de la nivelul legăturii de date. Fiecare pachet are un identificator unic. Formatul pachetelor permite adresarea memoriei pe 32 de biți sau pe 64 de biți.

## 7.5 Magistrala ATA

### 7.5.1 Magistrala paralelă ATA (IDE)

Versiunea inițială (ATA-1) a standardului ATA (PATA) a fost aprobată oficial de institutul ANSI în anul 1994, deși prima versiune de lucru a acestui standard a fost publicată în anul 1989.

Ultima versiune a standardului ATA/ATAPI 7 definește următoarele caracteristici ale interfeței ATA:

- Conectori cu 40 sau 44 de pini;

- Un canal ATA, care poate fi partajat de două unități de discuri, configurate ca o unitate *master* și o unitate *slave*;
- Utilizarea tehnologiei S.M.A.R.T. (*Self-Monitoring, Analysis, and Reporting Technology*) pentru predicția degradării performanței unităților de discuri;
- Creșterea integrității datelor prin utilizarea unui cod ciclic redundant CRC;
- Definirea unui cablu opțional cu 80 de fire (dintre care 40 de fire sunt de masă), care permite creșterea imunității la zgomote;
- Utilizarea cablului cu 80 de fire este obligatorie pentru funcționarea în modurile UDMA/66, UDMA/100, UDMA/133.

Tehnologia S.M.A.R.T., dezvoltată inițial de firma *IBM*, permite sistemului de operare să monitorizeze parametrii de funcționare ai unei unități de discuri în scopul detectării unor degradări ale performanței acesteia. Această degradare se poate accentua în mod progresiv, conducând în final la o defectare a unității și la pierderea datelor înregistrate. Prin utilizarea acestei tehnologii este posibilă predicția defectării unității și salvarea din timp a datelor. Tehnologia S.M.A.R.T. nu permite însă predicția defectării subite a unei unități de discuri.

#### **Observație**

Rata maximă de transfer a interfeței ATA nu reprezintă principala limitare a vitezei la transferul cu o unitate de discuri. Cel mai important factor este rata susținută cu care unitatea poate transfera datele de pe suport la placa logică a unității, prin intermediul ansamblului capetelor de citire. Chiar și la unitățile de discuri performante, această rată poate fi de sub 80 MB/s. Astfel, principalul criteriu la alegerea unei unități de discuri trebuie să fie rata de transfer susținută de pe suport și nu rata de transfer maximă a interfeței.

#### **Interfața electrică**

Standardul ATA prevede opțiunea de funcționare a sistemului cu două unități conectate în lanț. Prima unitate (unitatea 0) este numită *master*, iar unitatea a doua (unitatea 1) este numită *slave*. O unitate poate fi desemnată fie ca *master*, fie ca *slave*, prin așezarea unui jumper sau comutarea unui microîntrerupător.

Cablul utilizat (Figura 7.13) pentru conectarea unităților IDE este un cablu cu 40 de fire, prevăzut cu trei conectori.

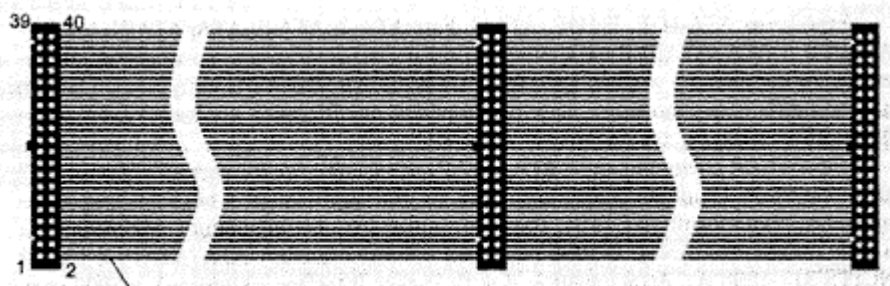


Figura 7.13 - Cablul cu conectori ATA

De obicei, cablul pleacă de la conectorul IDE către o unitate, și apoi în continuare spre a doua unitate (dispunere în lanț). La unul din capete cablul este legat fie la conectorul pentru interfața IDE de pe placa de bază, fie la o placă adaptoare pentru interfața IDE, care este amplasată într-unul din conectorii plăcii de bază.

#### **7.5.2 Tehnologia Serial ATA**

În tehnologia serial ATA (*SATA*) pentru transferuri de date se folosesc două fire prin care se transmit semnale diferențiale cu o amplitudă de 250 mV, rezultând o imunitate crescută la zgomote.

Magistrala constă din 7 fire (figura 7.14), dintre care 4 sunt informaționale și 3 pentru conectare la masă (Ground). O pereche de fire se folosește pentru transferuri, iar alta pentru recepția datelor. Informația se codifică după metoda NRZ (Non Return to Zero).

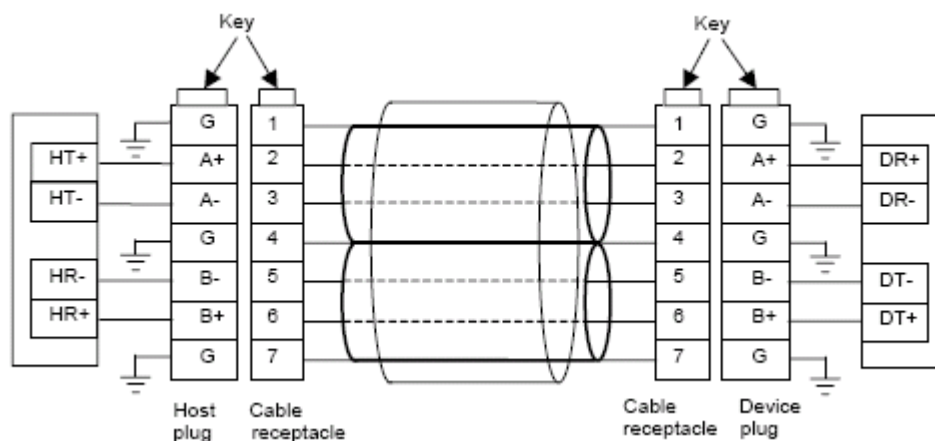


Figura 7.14 - Structura magistralei SATA



Figura 7.15 - Cablul SATA

Lungimea cablului poate fi de 1 m (figura 7.15). La SATA pot fi conectate nu numai dispozitive de memorare interne dar și dispozitive externe blocului de sistem.

Viteza de transfer este de 150 MB/s. Viteza declarată este de 1.5 Gb/s, dar folosirea metodei de codificare 8b/10b, a micșorat viteza de transfer la 1.2Gb/s. Dezvoltarea standardului prevede 3 generații SATA (tabelul 7.3). În SATA II rata maximă de transfer este de 3 Gb/s, iar în SATA III - până la 6Gb/s.

## 7.6 Magistrala SCSI

### 7.6.1 Magistrala paralelă SCSI

Interfața SCSI (*Small Computer System Interface*) provine din interfața SASI (*Shugart Associates System Interface*). SCSI nu este o interfață de disc, deci un anumit tip de controller, ci o interfață la nivelul sistemelor, formată dintr-o magistrală care acceptă mai multe echipamente. Unul din aceste echipamente, adaptorul cu sistemul gazdă, funcționează ca o punte între magistrala SCSI și magistrala sistemului. Magistrala SCSI nu comunică direct cu echipamentele periferice, cum sunt unitățile de discuri, ci cu controllerul care este inclus în aceste unități.

O singură magistrală SCSI poate accepta până la 8 sau 16 unități fizice, numite unități SCSI, dintre care una este adaptorul SCSI. Unitățile fizice pot fi unități de discuri fixe, unități de bandă, unități CD-ROM, scanere, imprimante. Majoritatea sistemelor pot accepta până la patru adaptoare SCSI la sistemul gazdă, fiecare din acestea permițând cuplarea a până la șapte echipamente periferice, conectate în lanț.

SCSI este o interfață destinată în special stațiilor de lucru și serverelor cu performanțe ridicate. La prima versiune a interfeței SCSI, frecvența de ceas a magistralei era de 5 MHz, iar rata maximă de transfer era de 5 MB/s. La versiunile actuale ale acestei interfețe, frecvența de ceas a magistralei este de 80 MHz sau 160 MHz, iar ratele maxime de transfer sunt de 320 MB/s, respectiv 640 MB/s.

Pe lângă aceste versiuni paralele ale interfeței SCSI, a fost elaborată și o versiune serială a interfeței. Această interfață SCSI serială, numită *Serial Attached SCSI*, a înlocuit treptat interfața SCSI paralelă. Atât tehnologia SCSI paralelă, cât și cea serială, sunt promovate de organizația SCSI Trade Association ([www.scsita.org](http://www.scsita.org)).

### 7.6.2 Interfața Serial Attached SCSI (SAS)

Serial Attached SCSI (SAS) reprezintă versiunea serială a interfeței SCSI. Aceasta utilizează un protocol serial punct la punct și setul standard de comenzi SCSI. Interfața serială SAS oferă compatibilitate cu unitățile de discuri SATA de generația a doua, care se pot conecta la plăci de bază SAS. Totuși, unitățile de discuri SAS nu se pot conecta la plăci de bază SATA.



Conectarea a 4 SATA la adaptorul SAS

Standardele SAS au fost elaborate de comitetul tehnic *T10* al *INCITS* (*International Committee for Information Technology Standards*). Prima versiune a standardului interfeței SAS a fost publicată în anul 2003, iar o versiune îmbunătățită (SAS-1.1) a fost publicată în anul 2005. Ambele versiuni specifică a interfață serială cu o viteză maximă de 3 Gbiți/s. Standardul SAS-2, care a fost publicat de comitetul *INCITS* în anul 2009, definește generația a doua a interfeței SAS. Această versiune a standardului introduce o legătură serială cu o viteză maximă de 6 Gbiți/s, un nivel fizic care este compatibil cu interfața SATA, și protocoale pentru transferul comenzilor SCSI la echipamentele SAS și a comenzilor ATA la echipamentele SATA. Versiunea SAS-2.1 a standardului, care a fost aprobată în anul 2010, definește un număr de îmbunătățiri ale standardului SAS-2, printre care conectori suplimentari și facilități de gestiune a energiei consumate. Versiunea SAS-3 a standardului, a fost aprobată în anul 2013, definește generația a treia a interfeței SAS, cu o viteză maximă de 12 Gbiți/s.

Interfața SAS este o arhitectură punct la punct, cu fiecare echipament conectat direct la un port SCSI în loc de a fi conectat la o magistrală partajată.



Protocolul SAS specifică o comunicație duplex între controlerul SAS și o unitate de discuri. Astfel, operațiile de citire și scriere pot fi executate în același timp, ceea ce crește performanța. Comparativ, unitățile SATA utilizează o comunicație semiduplex, astfel încât datele sunt transmise la unitate pentru scriere, datele care trebuie citite de la unitate trebuie să aștepte terminarea comunicației precedente.

Se pot conecta un număr mare de unități de discuri SAS sau SATA la un port al unui controler SAS prin utilizarea unor *expandoare* SAS. Un expander permite ca un singur inițiator să comunice cu un număr de echipamente destinație SAS/SATA. Un expander SAS este similar cu un comutator dintr-o rețea, care permite conectarea mai multor sisteme utilizând un singur port al comutatorului. Costul unui sistem care conține un expander este mult mai redus comparativ cu costul unui sistem care conține un controler SAS cu un număr mare de porturi sau mai multe controlere cu un număr mai mic de porturi. Prin utilizarea expandoarelor, sunt posibile până la 65,535 legături fizice (dispozitive).

Interfața SAS utilizează semnale diferențiale, codificarea 8b/10b a datelor și cifrarea datelor pentru reducerea interferențelor electromagnetice. Se pot combina până la patru porturi cu aceeași adresă într-un port de lățime mai mare, care permite transferuri de date cu până la 24 Gbiți/s.

Unitățile de discuri SAS sunt cu port dual, ceea ce înseamnă că acestea se pot conecta direct și pot fi controlate de două controlere SAS în același timp. Această posibilitate permite construirea unui sistem



redundant. Atunci când unul din controlerele SAS se defectează, celălalt va fi totuși în măsură să acceseze unitățile de discuri SAS și datele memorate pe aceste unități.

Principalele deosebiri dintre interfețele SAS și SATA sunt următoarele:

- Unitățile de discuri SATA sunt identificate prin numărul portului conectat la adaptorul calculatorului gazdă, în timp ce echipamentele SAS sunt identificate prin adresa lor SAS sau *World Wide Name*. *World Wide Name* (WWN) este un identificator de 64 biți care reprezintă adresa SAS și identifică în mod unic echipamentul în domeniul SAS. Din cei 64 de biți, 24 biți reprezintă identificatorul firmei producătoare și 40 de biți reprezintă identificatorul specific al producătorului.
- Spre deosebire de protocolul SATA, protocolul SAS permite existența mai multor inițiatori într-un domeniu SAS.
- Interfața SATA permite conectarea numai a unităților de discuri magnetice și a unităților optice. Interfața SAS permite conectarea și a altor tipuri de echipamente, cum sunt scannere și imprimante. Totuși, aceste echipamente au, de obicei, alte interfețe decât SAS, cum sunt USB, IEEE 1394 sau Ethernet.
- Interfața SAS utilizează nivele mai ridicate de tensiune (0,8-1,6 V) decât interfața SATA (0,4-0,6V).
- Din cauza tensiunilor mai ridicate ale semnalelor, la interfața SAS se pot utiliza cabluri mai lungi (până la 8 m), comparativ cu interfața SATA, la care lungimea maximă a cablului poate fi de 1 m.

Unitățile de discuri SAS au performanțe superioare comparativ cu unitățile de discuri SATA. Viteza de rotație a unităților SAS este cuprinsă între 10.000 și 15.000 rotații pe minut (RPM), în timp ce viteza de rotație a unităților SATA este cuprinsă între 5.400 și 7.200 RPM. Viteza de rotație mai ridicată reduce timpul de acces. Comunicația duplex permisă de unitățile SAS contribuie de asemenea la performanța superioară a acestor unități comparativ cu unitățile SATA. Unitățile SAS sunt cu port dual, iar aceasta le permite să comunice cu două adaptoare ale calculatorului gazdă sau controlere simultan, ceea ce îmbunătățește disponibilitatea datelor. De asemenea, unitățile SAS sunt mai fiabile decât unitățile SATA și sunt proiectate pentru o utilizare mult mai intensă. Pe de altă parte, unitățile SATA sunt mult mai puțin costisitoare decât unitățile SAS. Un alt avantaj al unităților SATA este că acestea au, de obicei, capacitatea semnificativ mai ridicată decât cea a unităților SAS.

## 7.7 Magistrala USB

Unul din scopurile elaborării magistralei USB (Universal Serial Bus sau Magistrala Serială Universală) a fost simplificarea interconexiunilor dintre calculator și periferice, prin reducerea numărului de cabluri care se conectează la calculator și utilizarea aceluiași tip de conector pentru diferite categorii de periferice. Într-un sistem conținând o magistrală USB, diferitele periferice se pot conecta în serie sau într-o topologie sub formă de stea pe mai multe nivele, un singur periferic fiind conectat la un port USB al calculatorului gazdă. Un alt aspect care s-a avut în vedere la elaborarea magistralei USB a fost asigurarea unei rate de transfer mai ridicate decât ratele de transfer permise de porturile seriale și paralele. De asemenea, s-a urmărit ca perifericele să poată fi adăugate în mod simplu la calculator, fără deschiderea carcasei acestuia, fără oprirea tensiunii de alimentare și fără reîncărcarea sistemului de operare.

Specificarea USB 1.0 publicată în ianuarie 1996 determină două viteze de transfer: FS (Full Speed) - 12 Mb/s și LS (Low Speed) - 1.5 Mb/s. În anul 2000 a apărut o nouă specificație pentru USB, versiunea USB 2.0, care a introdus viteza HS (High Speed) - 480 Mbps. Aceasta creștere a vitezei de transfer a făcut posibilă acceptarea magistralei USB ca interfață pentru writer-e CD/DVD, scanere, camere foto digitale, echipament video, etc. În anul 2008 Intel anunță specificația USB 3.0, care a introdus viteza SS (Super Speed) - 4.8 Gbps.

Într-un sistem pot fi prezente și funcționa concomitent dispozitive cu toate trei viteze enunțate mai sus. Magistrala permite conectarea a maxim 127 de dispozitive, la o distanță maximă - 25m (Folosind hub-uri intermediari). Cablul de comunicație constă din 4 fire: două sunt de date și două pentru alimentare (5V și 0V). Pentru a facilita interconectarea între echipamente, se folosesc conectori standard Master și Slave. Lungimea cablului este de maxim 5m. Acesta are la un capăt conectori de tip A, pentru conectarea la calculator (master) și la celălalt - conectori de tip B, pentru conectare la periferic (slave).

## Structura USB

USB asigură schimbul de date între host-calculatorul și diferite echipamente periferice. Distribuirea vitezei de transfer a magistralei între echipamentele periferice sunt planificate de HOST și se realizează de el cu



ajutorul indicatoarelor (token). Magistrala permite conectarea, configurarea, folosirea și deconectarea dispozitivelor în timpul lucrului hostului. Mai jos se aduc variantele de traducere a termenilor din specificațiile «Universal Serial Bus Specification. Revision 1.0, January 15, 1996», publicate de Compaq, DEC, IBM, Intel, Microsoft, NEC și Northern Telecom. Mai multe detalii și informații puteți găsi pe adresa: <http://www.usb.org>.

Ca dispozitive (Device) USB pot fi hub-uri (distribuitoare), funcții sau dispozitive ce îmbină și hub-uri și funcții (compound device, figura 7.17). Hub-urile asigură puncte de conectare suplimentare a dispozitivelor la magistrală. Funcțiile USB sunt niște sisteme, de exemplu mouse ce susține USB. Dispozitive care combină câteva funcții, (compound device, figura mai jos) de exemplu, keyboard cu trackball încorporat.

Elementele principale ale unui sistem care utilizează magistrala USB sunt dispozitivele USB, cablurile USB și programele de sistem. Dispozitivele de pe magistrala USB sunt conectate fizic la calculatorul gazdă utilizând o topologie sub formă de stea, după cum se ilustrează în Figura 7.17.

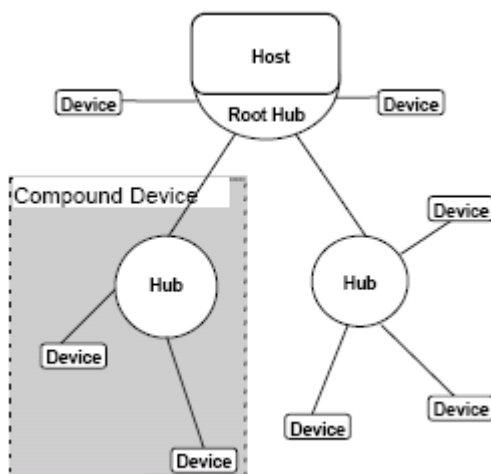


Figura 7.17 – Topologia fizică

În specificațiile versiunea 1.0 a standardului USB magistrala USB este menționată ca host-centrică: Ca Master (dispozitivul ce controlează transferurile pe magistrală) pe magistrala USB întotdeauna este host-calculatorul (cu USB controller incorporat), iar toate dispozitivele periferice conectate la host sunt dispozitive conduse sau Slave. Structura magistralei USB pe mai multe nivele se ilustrează în Figura 7.18.

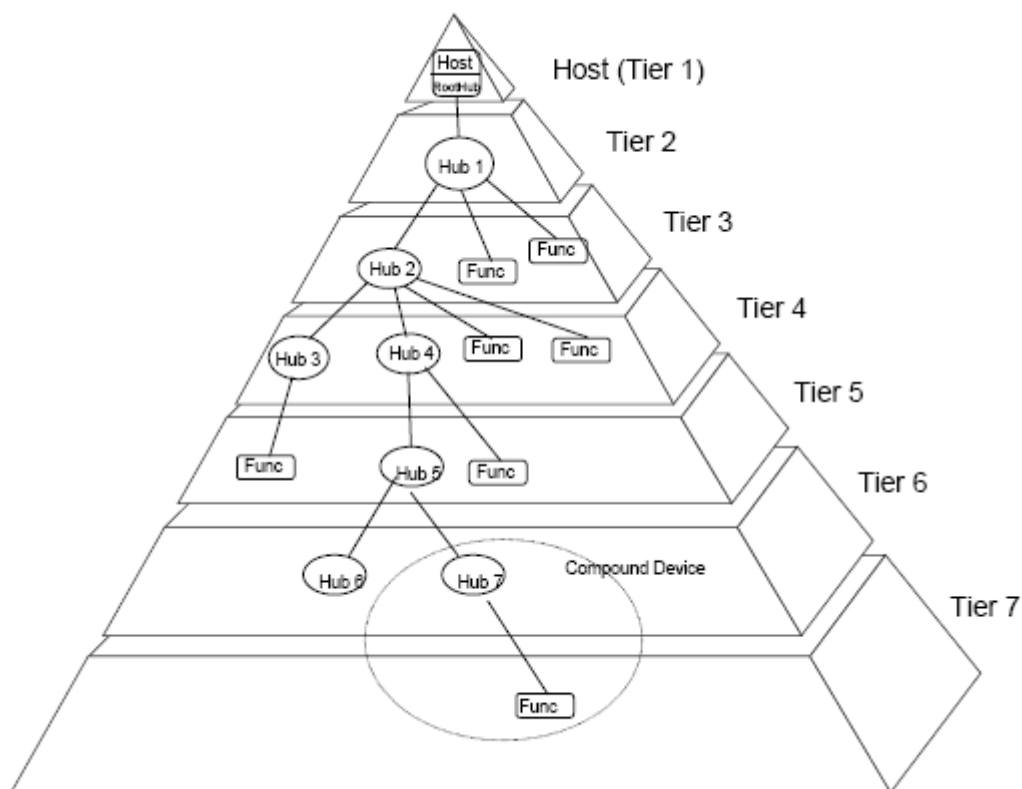


Figura 7.18 – Topologia magistralei pe nivele

Centrul magistralei USB este însăși host-controlerul, care este conectat cu hub de rădăcină (root hub), de regula, prin unul din două porturi ale sale.

Un hub (distribuitor) reprezintă o clasă specială de dispozitiv USB, care asigură puncte de conectare suplimentare pentru alte dispozitive USB. Aceste puncte de conectare se numesc porturi. Calculatorul gazdă conține un hub rădăcină, prin care asigură unul sau mai multe puncte de conectare. În plus, acest hub conține controlerul magistralei USB. Fiecare magistrală USB are un singur controler de magistrală (figura 7.19).

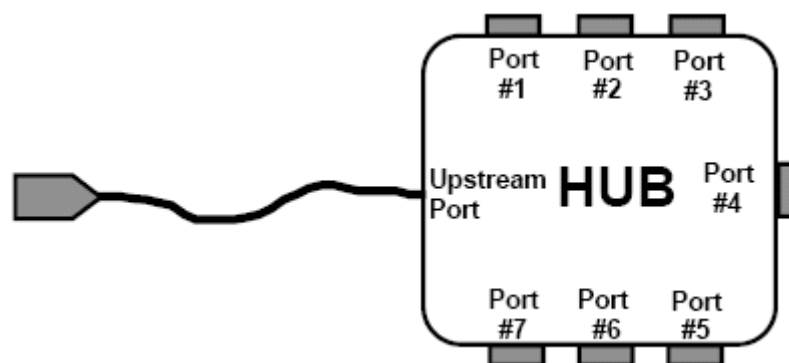


Figura 7.19 – Hub USB tipic

Figura 7.19 prezintă un hub USB tipic. Unul din porturile hub (Upstream port) permite conectarea la calculatorul gazdă sau la un hub de pe nivelul superior al topologiei. Fiecare din celelalte șapte porturi (Downstream ports) permit conectarea la un hub sau la o funcție de pe nivelul inferior. Hub-urile pot fi conectate în cascadă până la cinci nivele. Hub-ul recunoaște conectarea dinamică a unui periferic și asigură o putere de cel puțin 0,5 W pentru fiecare periferic în timpul inițializării. Sub controlul programului de sistem, hub-ul poate asigura o putere suplimentară pentru funcționarea perifericelor, până la 2,5 W (un curent de 0,5 A). Unele periferice, cum este tastatura, mouse-ul sau creionul optic, pot fi alimentate numai cu tensiunea furnizată de cablul magistralei, în timp ce altele pot avea o sursă proprie de alimentare.

Un hub constă din două părți: un controler și un repetor. Controlerul conține registre de interfață pentru comunicația cu calculatorul gazdă. Comenzile de stare și de control permit calculatorului gazdă configurarea hub-ului, monitorizarea și controlul porturilor sale. Repetorul este un comutator controlat prin protocol între portul de nivel superior și porturile de nivel inferior. De asemenea, repetorul monitorizează semnalele de pe porturi și gestionează tranzacțiile care îi sunt adresate. Toate celelalte tranzacții sunt repetate la dispozitivele atașate. Fiecare port de nivel inferior poate fi validat individual și poate fi conectat la dispozitive cu viteză ridicată sau cu viteză redusă. Porturile cu viteză redusă sunt izolate de semnalele cu viteză ridicată.

O *funcție* este un dispozitiv USB care poate transmite și/sau recepționa date sau informații de control pe magistrală. Acest dispozitiv trebuie să răspundă la cererile de tranzacție transmise de calculatorul gazdă. O funcție este implementată în mod obișnuit ca un periferic separat conectat printr-un cablu la un port al unui hub. Un singur dispozitiv fizic poate conține însă funcții multiple. De exemplu, o tastatură și un trackball pot fi combinate într-un singur dispozitiv fizic. În cadrul unui asemenea dispozitiv compus, funcțiile individuale sunt atașate la un hub, iar acest hub intern este conectat la magistrala USB.

Fiecare funcție conține informații de configurație care descriu posibilitățile sale și resursele necesare. Aceste informații sunt transmise calculatorului gazdă ca răspuns la o tranzacție de control. Înaintea utilizării unei funcții, aceasta trebuie configurată de calculatorul gazdă. Această configurare presupune alocarea unei lățimi de bandă în cadrul magistralei USB și selectarea opțiunilor specifice de configurație.

Specificațiile magistralei USB 2.0 (numită și *Hi-Speed USB*) descriu o magistrală cu performanțe îmbunătățite. Conectoarele și cablurile conforme cu specificațiile USB 1.1 permit obținerea ratelor de transfer mai ridicate ale magistralei USB 2.0 fără nici o modificare. Rata de transfer maximă care se poate obține este de 480 Mbiți/s. Perifericele USB 2.0 cu viteze de transfer superioare sunt conectate la un distribuitor USB 2.0. Un distribuitor USB 2.0 acceptă tranzacții de viteză ridicată și furnizează datele cu ratele corespunzătoare perifericelor USB 2.0 și perifericelor USB 1.1. Vitezele de transfer ridicate sunt negociate cu fiecare periferic, iar dacă un periferic nu permite o viteză ridicată, legătura cu acest periferic va funcționa la viteza mai redusă de 12 Mbiți/s sau 1,5 Mbiți/s. Aceasta implică o complexitate mai

ridicată a hub-urilor și necesitatea memorării temporare a datelor recepționate. Un hub USB 2.0 va avea porturi de ieșire pentru transferuri cu viteză ridicată și porturi de ieșire pentru transferuri cu viteză redusă.

Controlerile magistralei USB, aflate pe placa de bază a calculatorului gazdă, au propriile specificații. În cazul versiunii 1.1 a magistralei USB, existau două specificații pentru aceste controlere. Prima dintre ele, *Universal Host Controller Interface* (UHCI), a fost elaborată de firma *Intel* și permitea simplificarea circuitelor, partea mai complexă fiind cea de software. A doua specificație, *Open Host Controller Interface* (OHCI), a fost elaborată de firmele *Compaq*, *Microsoft* și *National Semiconductor*, această specificație permițând simplificarea programelor, partea mai complexă fiind cea de hardware. Odată cu introducerea versiunii 2.0 a magistralei USB, a fost necesară elaborarea unei noi specificații pentru controlerul de magistrală. Această specificație, numită *Enhanced Host Controller Interface* (EHCI), a fost elaborată de mai multe firme, printre care *Intel*, *Compaq*, *NEC*, *Microsoft* și *Lucent Technologies*.

Există specificații separate pentru diferite categorii (clase) de periferice USB. O clasă USB reprezintă un grup de periferice sau interfețe cu atribute sau servicii similare. De exemplu, două periferice sau interfețe sunt plasate în aceeași clasă dacă utilizează șiruri de date cu același format pentru comunicația cu calculatorul gazdă. Dintre clasele de periferice USB se amintesc următoarele: memorii de masă, monitoare, dispozitive de interacțiune cu utilizatorul (HID – *Human Interface Device*), imprimante, dispozitive audio, dispozitive de comunicație (modemuri, telefoane analogice și digitale, adaptoare de rețea), dispozitive de captare a imaginilor fixe (camere digitale).

### Sistema USB pe trei nivele

Sistema USB se dividează pe trei nivele cu anumite reguli de interacțiune. Dispozitivul USB conține interfața (USB Bus Interface), componentele logice (USB Logical Device) și funcția (function). Hostul tot este divizat în trei părți - interfața (USB Host Controller), de sistem (USB System) și soft-ul dispozitivului (Client SW). Fiecare parte are o destinație predefinită, interacțiunile logice și reale între nivele sunt prezentate în fig 7.21.

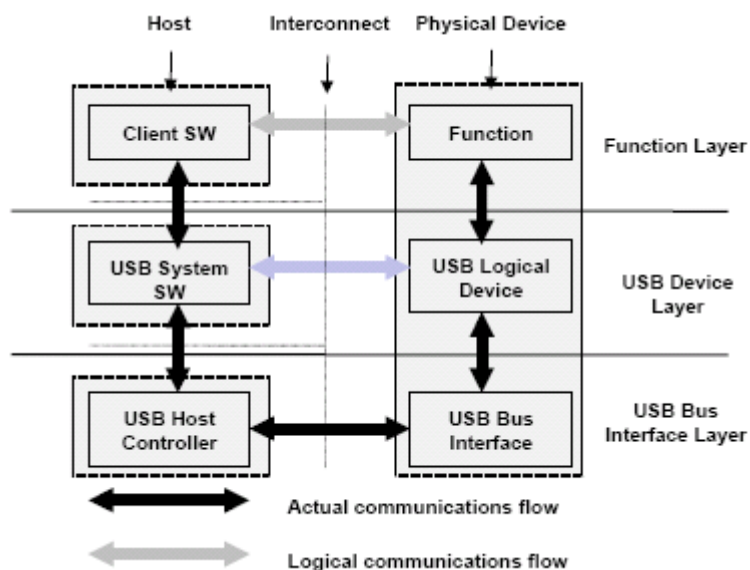


Figura 7.21 - Interacțiunea componentelor USB

În structura prezentată intră următoarele componente:

- Dispozitivul fizic USB- dispozitivul de pe magistrală, ce execută funcțiile, necesare utilizatorului.
- Client SW - soft, ce asigură conectarea și funcționarea a unui dispozitiv periferic concret, se execută în host-calculator.
- USB system SW- softul sistemului USB.
- USB Host Controller – soft și hard ce asigură conectarea dispozitivelor USB la Host.

### Interfața fizică

La interfața USB se utilizează patru tipuri de conectori: două tipuri de fișe, amplasate la capetele unui cablu USB, și două tipuri de mufe, amplasate în cadrul unui hub sau periferic.

Fișele și mufele pot fi de tip A sau de tip B.

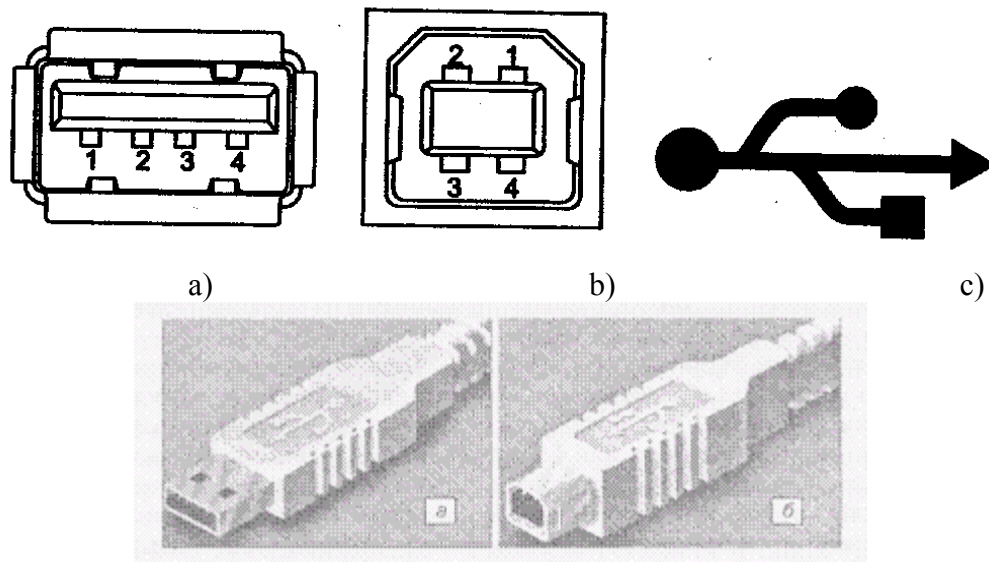


Figura 7.22 - Conector USB: a - tipul «A»(hub), b - tipul «B»(periferic), C - marcare convențională

Hub-rile (de exemplu, cele din calculator) conțin o mufă rectangulară cu patru pini de tip A. Perifericele se conectează la o asemenea mufă dintr-un hub utilizând o fișă rectangulară de tip A (Figura 7.22). Toate cablurile care sunt atașate permanent la periferice conțin o fișă de tip A. De obicei, perifericele se conectează însă printr-un cablu detașabil. Aceste periferice conțin o mufă pătrată de tip B, iar cablul care conectează aceste periferice la un hub conține o fișă de tip B la capătul care se conectează la periferic și o fișă de tip A la capătul care se conectează la hub. În acest fel, nu este posibilă conectarea incorectă a cablului. Specificațiile USB 2.0 au fost modificate ulterior publicării acestora pentru a include o fișă și o mufă de tip B de dimensiuni mai reduse. Acești conectori, numiți *mini-B*, conțin cinci contacte și se utilizează pentru echipamentele mobile de dimensiuni reduse, cum sunt telefoanele mobile. Echipamentele respective conțin o mufă de tip mini-B, iar cablurile utilizate pentru conectarea acestor echipamente la un calculator PC conțin o fișă de tip mini-B la un capăt și o fișă de tip A la celălalt capăt. Figura 7.23 ilustrează o fișă de tip mini-B alăturată cu o fișă de tip A.



Figura 7.23 - Fișă USB de tip mini-B alăturată cu o fișă de tip A

Specificațiile USB “On-The-Go” (OTG), care au fost elaborate ca o extensie a specificațiilor USB 2.0 pentru conectarea directă a unor echipamente mobile, fără utilizarea unui calculator PC, conțin descrierea unor fișe mini-A, mufe mini-A și mufe mini-AB. Utilizarea acestor conectori este necesară deoarece la conectarea directă a unor echipamente mobile unul din echipamente va avea rolul unui calculator gazdă. Specificațiile USB OTG descriu și diferite tipuri de cabluri care utilizează conectori de

dimensiuni reduse sau o combinație între un conector de dimensiuni reduse și unul de dimensiuni normale.

Pentru transferul semnalelor și a tensiunii de alimentare pe magistrala USB, se utilizează un cablu cu patru fire, ilustrat în Figura 7.30. Semnalele diferențiale de date se transmit pe liniile  $D+$  și  $D-$ , formate din două fire torsadate (răsucite). Semnalul de ceas este transmis codificat împreună cu datele. Codificarea utilizată este numită NRZI (*Non Return to Zero Invert*). În cazul acestei metode (fig.7.29),

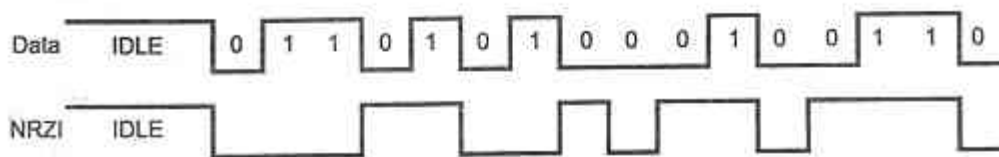


Figura 7.29 - Codificarea prin metoda NRZI

biții de 1 și 0 sunt reprezentați prin tensiuni opuse și alternante înalte și joase, fără a exista revenirea la tensiunea de referință (zero) între biții codificați. Sunt inserați biți suplimentari pentru a asigura tranziții suficiente ale semnalelor transmise, în scopul asigurării sincronizării. Fiecare pachet de date este precedat de un câmp de sincronizare pentru a permite receptorilor sincronizarea ceasurilor de recepție.

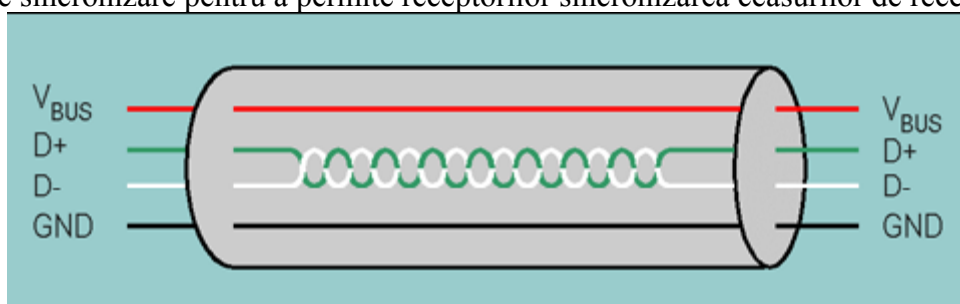


Figura 7.30 - Cablul USB

Cablul USB transmite și tensiunea de alimentare pentru periferice pe liniile  $V_{BUS}$  și  $GND$ . Tensiunea pe linia  $V_{BUS}$  este de +5 V la sursă. Pentru a asigura nivele de tensiune garantate la intrarea perifericelor și o impedanță de terminare corespunzătoare, se utilizează terminatori la fiecare capăt al cablului.

Pentru identificarea simplă a firelor din cablurile USB, standardul specifică utilizarea culorilor din Tabelul 7.5 pentru aceste fire. Acest tabel indică și asignarea pinilor conectorilor la semnalele magistralei USB.

#### Observație

• Nu este posibilă interconectarea a două calculatoare printr-un cablu USB obișnuit. Chiar dacă s-ar utiliza un cablu cu doi conectori de tip A, prin interconectarea a două calculatoare ar exista două controlere USB într-un sistem, ceea ce nu este permis. Există însă cabluri speciale care conțin o punte USB sub forma unui circuit integrat, prin intermediul căruia este posibilă comunicația între cele două calculatoare gazdă.

Tabelul 7.5 - Asignarea pinilor conectorilor la semnalele magistralei USB și culorile firelor cablurilor.

Nr. pin	Semnal	Culoare
1	$V_{BUS}$	Roșu
2	$D-$	Alb
3	$D+$	Verde
4	$GND$	Negru

La planificarea conectării trebuie de luat în considerație necesitățile de alimentare a dispozitivelor. Dispozitivele ce se alimentează de la magistrală de obicei se conectează la hub-uri, care sunt alimentate de la rețeaua electrică. La hub-uri care se alimentează de la magistrală, se conectează doar dispozitive care necesită puțin curent – de exemplu tastatura USB, mouse-ul USB etc.

### Modelul de transfer a datelor

Fiecare dispozitiv USB conține un set de puncte finale (Endpoint), cu care host-controllerul realizează schimb de informație. Endpoint se caracterizează cu următorii parametri:

- frecvența de acces la magistrală;
- lățimea de bandă a canalului;
- numărul Endpoint
- cerințe la prelucrarea erorilor
- mărimea maximală a pachetelor transferate
- tipul transferului
- direcția transferului (pentru transferul izocronic)

Fiecare dispozitiv conține un Endpoint cu numărul 0, folosit pentru inițializare și control. Acest Endpoint este întotdeauna configurat la conectarea dispozitivului la magistrală.

Dispozitivele – funcții mai conțin un set de Endpoint, ce realizează schimburi de date utile. Dispozitivele lente LS pot conține până la două Endpoint, iar dispozitivele FS până la 16 Endpoint de intrare și 16 Endpoint de ieșire (restricție de protocol). Endpoint nu pot fi folosite până ce nu au fost configurate (până la instalarea unui canal coordonat).

Canal (Pipe) la USB se numește modelul de transfer a datelor între host-controller și Endpoint al dispozitivului. EL este un canal logic și uneori întregul canal logic e numit punct final (Figura 7.31). Există două tipuri de canale: fluxuri (Stream) și mesaje (Message).

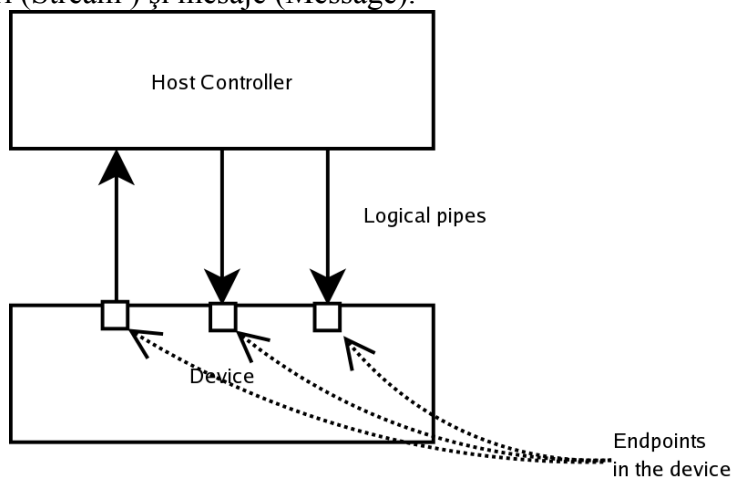


Figura 7.31

Prin canalul de tip flux sunt transmise date într-o direcție. Un Endpoint poate fi folosit pentru două canale de tip flux — pentru intrare sau ieșire. Canalul de tip flux poate realiza următoarele tipuri de transferuri: continuu (date voluminoase), izocron și de întreruperi. Transferul se petrece prin metoda primul intrat – primul ieșit (FIFO). Canalul de tip mesaje susține tipul de transfer de control. Formatul mesajelor este determinat de specificarea USB. Hostul transmite cererea către Endpoint, după care urmează pachetul cu mesaje, după care urmează pachetul cu informație despre starea Endpoint. Următorul mesaj nu poate fi transmis până precedentul nu este prelucrat. Schimbul cu mesaje în ambele direcții se efectuează cu unul și același Endpoint.

Canalele se organizează la configurarea dispozitivului USB. Pentru fiecare dispozitiv conectat există un canal pentru mesaje de tip control (Control Pipe Off), prin care se transmite informația despre configurare, stare și control.

### Tipurile de transfer de date

Arhitectura USB permite patru tipuri de transferuri de date: de control, de întrerupere, de date voluminoase și izocrone.

Transferurile *de control* se utilizează de driverele calculatorului gazdă pentru configurarea dispozitivelor care sunt atașate la sistem. Alte drivere pot utiliza transferuri de control în moduri specifice implementării.

Transferurile *de întrerupere* se utilizează pentru date cu volum redus. Transferul acestor date poate fi solicitat de un dispozitiv în orice moment, iar rata de transfer pe magistrala USB nu poate fi mai redusă decât cea specificată de dispozitiv. Datele pentru care se utilizează transferurile de întrerupere constau din notificarea unor evenimente, din caractere sau coordonate care sunt organizate pe unul sau mai mulți



octeți. Un exemplu îl reprezintă coordonatele de la un dispozitiv indicator (Ex. mouse). Datele interactive pot avea anumite limite ale timpului de răspuns care trebuie asigurate de magistrala USB.

Transferurile *de date voluminoase* (“*bulk*”) se utilizează cu periferice cum sunt memorii de masă, imprimante sau scanere. Aceste date sunt secvențiale. Fiabilitatea transferurilor este asigurată la nivel hardware prin utilizarea unui cod detector de erori și reluarea unui transfer cu erori de un număr de ori. Rata de transfer în cazul acestor transferuri poate varia în funcție de alte activități de pe magistrală.

Transferurile *izocrone* (*isos* – egal, *chronos* – timp) se utilizează pentru datele care trebuie furnizate cu o anumită rată de transfer constantă și a căror sincronizare trebuie garantată. Izocron are semnificația “cu durată egală” sau “care apare la intervale regulate”. Datele izocrone sunt generate în timp real și trebuie furnizate cu rata cu care sunt recepționate pentru a păstra sincronizarea lor. Pe lângă rata de transfer impusă, pentru datele izocrone trebuie respectată și întârzierea maximă cu care acestea sunt furnizate. Furnizarea la timp a datelor izocrone este asigurată cu prețul unor pierderi potențiale în șirul de date. Cu alte cuvinte, erorile de transmisie nu sunt corectate prin mecanisme hardware, de exemplu, prin retransmiterea lor. În concluzie, transferurile izocrone se caracterizează prin furnizarea la timp a datelor și prin lipsa retransmiterii lor în cazul unor erori, deoarece datele întârziate nu mai sunt utile. Spre deosebire de transferurile izocrone, transferurile asincrone se caracterizează prin faptul că fiabilitatea transmiterii datelor este mai importantă decât asigurarea sincronizării. Pentru aceasta se utilizează retransmiterea datelor în cazul unor erori, chiar dacă apar întârzieri din această cauză.

Un exemplu tipic de date izocrone este reprezentat de imaginile video. Dacă rata de transfer a acestor șiruri de date nu este respectată, va avea loc pierderea unor date datorită depășirii capacității bufferelor. Chiar dacă datele sunt furnizate de magistrala USB cu rata adecvată, întârzierile introduse de programe pot afecta negativ aplicațiile care utilizează aceste date, cum sunt cele pentru videoconferințe.

Șirurilor de date izocrone li se alocă o porțiune dedicată a lățimii de bandă a magistralei USB. De asemenea, această magistrală este proiectată pentru o întârziere minimă a transferurilor de date izocrone.

### Protocolul

Similar cu alte interfețe mai recente, interfața USB utilizează un protocol bazat pe pachete. Toate transferurile sunt inițiate de controlerul USB al calculatorului gazdă. Tranzacțiile de pe magistrală implică transmisia a patru tipuri de pachete:

- Pachet antet (simbol – “*token*”);
- Pachet de date;
- Pachet de confirmare (“*handshake*”);
- Pachet special.

Fiecare tranzacție, constă din 3 pachete și începe în momentul în care controlerul USB transmite, pe baza unei planificări, un pachet antet (Token) care descrie tipul tranzacției, direcția acesteia, adresa dispozitivului USB și numărul punctului terminal. Sursa tranzacției transmite apoi un pachet de date (Data) conținând datele care trebuie transferate, sau poate indica faptul că nu are date de transmis prin faptul că pachetul de date nu conține informații utile. Destinația răspunde, în general, cu un pachet de confirmare (Handshake) indicând dacă transferul s-a efectuat cu succes sau dacă punctul terminal nu este disponibil. Succesiunea pachetelor în tranzacții este prezentată în fig. 7.32

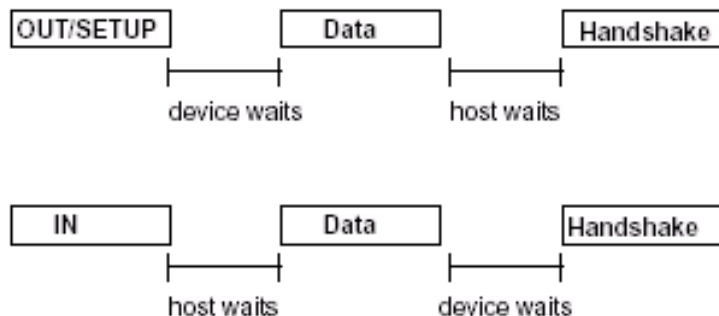


Figura 7.32 - Succesiunea pachetelor

### Formatul pachetelor USB

Toate pachetele încep cu un câmp de sincronizare (SYNC), care este utilizat de circuitele receptorului pentru sincronizarea cu ceasul transmițătorului. Câmpul de sincronizare conține un număr de șase tranziții succesive de la valoarea 1 la valoarea 0 sau invers, urmate de un marcaj de doi biți care indică sfârșitul câmpului de sincronizare.

Câmpul de identificare al pachetului (PID) urmează imediat după câmpul de sincronizare. Câmpul PID conține patru biți care indică tipul pachetului și patru biți de control care confirmă acuratețea biților care conțin tipul pachetului. Biții de control conțin complementul față de 1 al biților care reprezintă tipul pachetului.

Tabelul 7.6 indică diferitele tipuri de pachete, codificarea și descrierea acestora. Pentru simplitate, pachetele speciale nu sunt detaliate.

Câmpul de adresă (ADDR) specifică adresa funcției USB care este sursa sau destinația unui pachet de date. Acest câmp are o lungime de 7 biți, permițând specificarea a până la 128 de adrese. Fiecare adresă definește o singură funcție. Adresa 0 este rezervată ca adresă implicită și nu poate fi asignată în mod explicit unei funcții. La pornirea și resetarea unei funcții, adresa acesteia va avea valoarea implicită 0. Calculatorul trebuie să seteze adresa funcției în timpul procesului de enumerare.

Tabelul 7.6 - Codificarea și descrierea diferitelor tipuri de pachete USB

Tip pachet	Subtip pachet	PID [3..0]	Descriere
Antet (Token)	OUT	0001	Adresa și numărul punctului terminal pentru o tranzacție de ieșire
	IN	1001	Adresa și numărul punctului terminal pentru o tranzacție de intrare
	SOF	0101	Indicator de început al cadrului și numărul cadrului
	SETUP	1101	Adresa și numărul punctului terminal pentru o tranzacție de control în faza de setare
Date	DATA0	0011	Identificator pentru pachetele de date cu număr par
	DATA1	1011	Identificator pentru pachetele de date cu număr impar
	DATA2	0111	Identificator pentru pachetele de date la tranzacțiile izocrone de intrare de viteză ridicată și cu lățimea de bandă ridicată
	MDATA	1111	Identificator pentru pachetele de date la tranzacțiile izocrone de ieșire de viteză ridicată și cu lățimea de bandă ridicată
Confirmare	ACK	0010	Confirmarea recepționării fără erori a pachetului de date
	NAK	1010	Dispozitivul receptor nu poate primi date sau dispozitivul transmițător nu poate transmite date
	STALL	1110	Punctul terminal este oprit
	NYET	0110	Nu s-a recepționat încă un răspuns de la receptor
Special		XY00	Identificator al unui pachet special; XY poate fi 01, 10 sau 11

### Transferuri pe magistrala USB

Fiecare transfer pe magistrală este inițiat de Host controler. Controlerul formează cadre în care sunt plasate transferurile către fiecare dispozitiv conectat la magistrala USB. Pentru sincronizarea întregului sistem USB, calculatorul gazdă transmite câte un pachet SOF (*Start-of-Frame*) la fiecare interval de timp corespunzător începutului unui cadru sau micro-cadru. Un cadru reprezintă un interval de timp de  $1 \text{ ms} \pm 0,0005 \text{ ms}$  și este definit pentru magistrala USB cu viteza normală (12 Mbiți/s). Un micro-cadru reprezintă un interval de timp de  $125 \mu\text{s} \pm 0,0625 \mu\text{s}$  și este definit pentru magistrala USB cu viteza ridicată (480 Mbiți/s). Un pachet SOF constă dintr-un câmp de sincronizare, un câmp PID și un câmp de 11 biți reprezentând numărul cadrului, după cum se ilustrează . După pachetul SOF urmează transferul către primul dispozitiv conectat la magistrala USB (pachetele IN/OUT, DATA, HANDSHAKE, ca pe fig. 7.33), apoi se petrece schimbul de date cu dispozitivul doi (pachetele IN/OUT, DATA, HANDSHAKE), până nu este apelat ultimul dispozitiv conectat la magistrală. Dispozitivele lente pot fi apelate și peste câteva cadre. În următorul cadru interogarea dispozitivelor se repetă.

În cazul magistralei USB cu viteza ridicată, numărul cadrului va fi același pentru opt pachete SOF consecutive, pe durata unei perioade de 1 ms. Succesiunea cadrelor este ilustrată în figura 3.14.



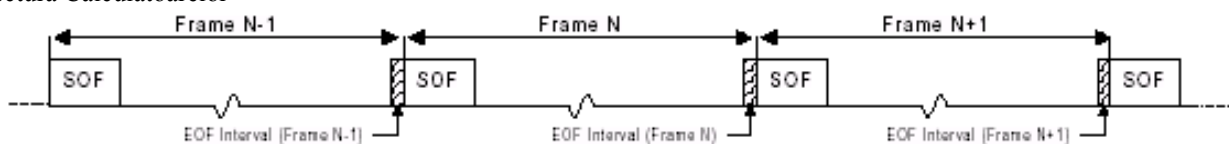


Figura 7.33 - Succesiunea de cadre USB

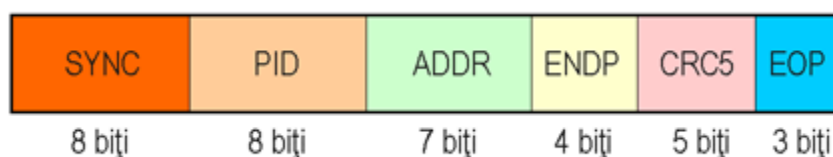
Toate funcțiile și hub USB recepționează pachetele SOF. Recepția acestor pachete nu va determina generarea unui pachet de confirmare din partea receptorului.

#### ***Detalierea pachetelor USB***

În continuare se prezintă formatul pachetelor de antet, SOF, de date și de confirmare.

##### Pachete de antet

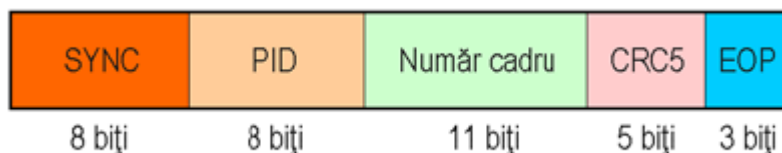
Aceste pachete sunt transmise doar de calculatorul gazdă. Structura unui pachet de antet este următor:



Câmpul de identificare al pachetului, PID, poate specifica un pachet de antet cu subtipul IN, OUT sau SETUP. Pachetele cu subtipul IN sau OUT informează dispozitivul USB asupra direcției transferului care urmează: intrare (citire de către calculatorul gazdă), respectiv ieșire (scriere de către calculatorul gazdă). Un pachet cu subtipul SETUP se utilizează la începutul transferurilor de control. În cazul pachetelor cu subtipul OUT sau SETUP, câmpurile ADDR și ENDP identifică în mod unic punctul terminal care va recepționa următorul pachet de date. În cazul unui pachet cu subtipul IN, câmpurile ADDR și ENDP identifică punctul terminal care va transmite un pachet de date. Câmpul CRC5 conține codul CRC pentru câmpurile ADDR și ENDP.

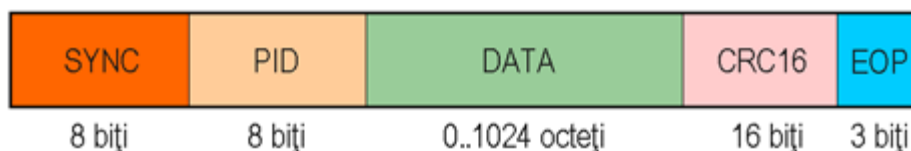
##### Pachet SOF

Un pachet SOF constă dintr-un câmp de sincronizare, un câmp PID și un câmp de 11 biți reprezentând numărul cadrului, după cum se ilustrează. După pachetul SOF urmează transferul către primul dispozitiv conectat la magistrala USB (pachetele IN/OUT, DATA, HANDSHAKE), apoi se petrece schimbul de date cu dispozitivul doi (pachetele IN/OUT, DATA, HANDSHAKE), până nu este apelat ultimul dispozitiv conectat la magistrală.



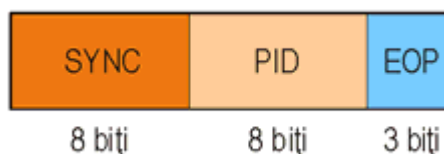
##### Pachete de date

Informațiile propriu-zise sunt transmise pe magistrala USB în pachetele de date. Un pachet de date constă din câmpul de sincronizare SYNC, un câmp de identificare al pachetului PID, un câmp de date, un câmp CRC de 16 biți și câmpul de sfârșit al pachetului EOP. Codul CRC se calculează numai pentru câmpul de date. Datele sunt transmise într-un număr întreg de octeți. Pentru dispozitivele cu viteză redusă, lungimea maximă a câmpului de date este de 8 octeți. Pentru dispozitivele cu viteză normală (12 Mbiți/s) și viteză ridicată (480 Mbiți/s), lungimea maximă este de 1024 octeți.



##### Pachete de confirmare

Pachetele de confirmare constau doar din câmpul de sincronizare SYNC, un câmp de identificare al pachetului PID și câmpul de sfârșit al pachetului EOP. Aceste pachete se utilizează pentru a raporta starea unei tranzacții de date prin subtipul returnat în câmpul PID. Subtipul unui pachet de confirmare poate fi ACK (*Acknowledge*), NAK (*Negative Acknowledge*), STALL (*Stall*) sau NYET (*No Response Yet*). Aceste subtipuri sunt descrise în Tabelul 3.2.



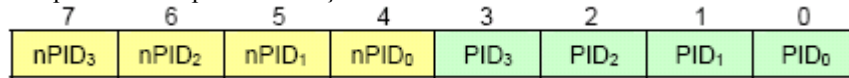
Principalele câmpuri ale pachetelor USB sunt descrise în continuare.

##### Câmpul de sincronizare

Toate pachetele încep cu un câmp de sincronizare (SYNC), care este utilizat de circuitele receptorului pentru sincronizarea cu ceasul transmițătorului. Câmpul de sincronizare conține un număr de șase tranziții succesive de la valoarea 1 la valoarea 0 sau invers, urmate de un marcaj de doi biți care indică sfârșitul câmpului de sincronizare.

#### Câmpul de identificare al pachetului

Câmpul de identificare al pachetului (PID) urmează imediat după câmpul de sincronizare. Câmpul PID conține patru biți care indică tipul pachetului și patru biți de control care confirmă acuratețea biților care conțin tipul pachetului. Biții de control conțin complementul față de 1 al biților care reprezintă tipul pachetului. Formatul câmpului PID este ilustrat în continuare, unde  $nPID_i$  reprezintă complementul față de 1 al bitului  $PID_i$ .



Calculatorul gazdă și toate funcțiile USB decodifică complet toți biții câmpului PID. Dacă un câmp PID este recepționat cu valori incorecte ale biților de control sau cu valori nedefinite ale tipului pachetului, se presupune că a fost recepționat eronat și restul pachetului este ignorat de receptor.

#### Câmpul de adresă

Câmpul de adresă (ADDR) specifică adresa funcției USB care este sursa sau destinația unui pachet de date. Acest câmp are o lungime de 7 biți, permițând specificarea a până la 128 de adrese. Fiecare adresă definește o singură funcție. Adresa 0 este rezervată ca adresă implicită și nu poate fi asignată în mod explicit unei funcții. La pornirea și resetarea unei funcții, adresa acesteia va avea valoarea implicită 0. Calculatorul trebuie să seteze adresa funcției în timpul procesului de enumerare.

#### Câmpul punctului terminal

Câmpul punctului terminal (ENDP) permite o adresare mai flexibilă a funcțiilor cu mai multe puncte terminale. Acest câmp are o lungime de 4 biți, ceea ce permite adresarea a până la 16 puncte terminale. Dispozitivele cu viteză redusă pot avea însă doar două puncte terminale suplimentare pe lângă punctul terminal cu numărul 0.

#### Câmpul de date

Câmpul de date poate conține între zero și 1024 de octeți, în funcție de tipul transferului. Biții de date din cadrul fiecărui octet sunt transmiși pe magistrală începând cu bitul cel mai puțin semnificativ.

#### Câmpurile de control ciclic redundant

Aceste câmpuri conțin codurile de control ciclic redundant (CRC) utilizate pentru verificarea integrității diferitelor câmpuri din pachetele de antet și de date. Câmpul PID nu este inclus în calculul codului CRC. Codurile CRC pentru pachetele de antet și de date asigură detectarea tuturor erorilor de un bit și de doi biți. În cazul în care codul CRC calculat la recepție diferă de codul transmis într-un câmp CRC, receptorul va ignora câmpurile protejate și, în majoritatea cazurilor, întregul pachet.

Standardul USB specifică polinoamele generatoare utilizate pentru calculul codurilor CRC. Pentru pachetele de antet se utilizează un câmp CRC de 5 biți (CRC5), iar pentru pachetele de date se utilizează un câmp CRC de 16 biți (CRC16).

#### Câmpul de sfârșit al pachetului

Acest câmp (EOP) indică sfârșitul unui pachet prin valoarea 0 pe durata corespunzătoare a doi biți, urmată de valoarea 1 pe durata corespunzătoare unui bit.

Standardul USB 3.0 este compatibil cu USB 2.0. Performanțele:

- Viteza maximă de transfer pe USB 3.0 este de 4.8 Gbps (interfața USB 2.0 Hi-Speed: 480 Mbps);
- Transferul informațiilor se face pe canale separate de comunicare (transfer bidirecțional – o pereche de canale LVDS) – astfel este posibilă citirea și scrierea simultană a datelor pe dispozitivele selectate; USB 2.0 suportă doar trafic unidirecțional.
- curentul maximal furnizat - 900 mA, față de cei 500 mA oferiți de USB 2.0.

În figura 7.34 este prezentat conectorul USB3.0

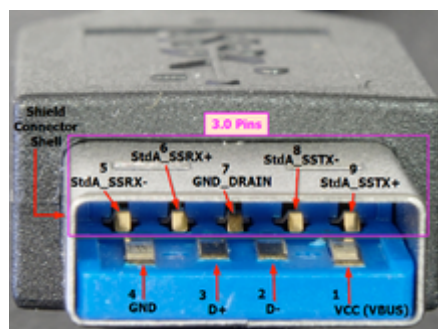


Figura 7.34