

3. Magistrale



Cuprins modul

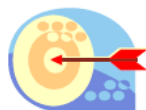
- 3.1. Introducere
- 3.2. Magistrale ierarhizate
- 3.3. Diagrame de semnal la acces
- 3.4. Magistrale multiplexate
- 3.5. Magistralele PCI și PCI Express

Cuprins



Introducere

Modulul “Magistrale“ este o prezentare generală a magistralelor începând de la definiții și aspecte de dezvoltare istorică. Este prezentată o structură tipică de magistrală ierarhizată de la calculatoare PC pentru a pune în evidență structura. Se detaliază prin diagrame de timp de acces o magistrală simplă, tipică pentru microcontrollere. O prezentare a magistralelor PCI și PCI Express arată tendința de trecere la magistrale paralele cu grupuri de canale seriale.



Obiective

După parcurgerea acestui modul studenții vor avea o privire generală asupra magistralelor în evoluția lor istorică și vor înțelege câteva aspecte particulare mai importante:

- Structura unei magistrale ierarhizate exemplificată pentru cazul unui calculator PC;
- Structura și funcționarea în detaliu a unei magistrale simple, implementată în microcontrollere;
- Structura generală a magistralelor PCI și PCI Express.

Obiective specifice:

1. Învățarea principiilor fundamentale de transfer de date
2. Introducere în domeniul transferului de date prin magistrale
3. Înțelegerea noțiunilor prin exemplificări practice



Durata medie de studiu individual

Durata medie de studiu individual este de 2 ore.

3.1. Introducere

Legătura între procesor și EP (Echipamente Periferice) se realizează prin canale I/O (de intrare/ieșire) prin intermediul magistralei. Evoluția în timp a canalelor I/O este în același timp o evoluție a creșterii complexității și performanțelor. Pot fi enumerate următoarele etape:

1. CPU controlează direct EP;
2. Este adăugat un modul I/O (o interfață serială sau paralelă, programabilă). CPU comandă EP prin transfer programat (direct sau prin interogare);
3. Aceeași configurație ca la 2, dar transferul are loc prin întreruperi;
4. Modulul I/O are acces direct la memorie prin DMA. Modulul poate muta informația direct în memorie, accesul CPU fiind necesar doar la începutul și sfârșitul transferului;
5. Modulul I/O folosește un microcalculator sau un microcontroller, cu instrucțiuni proprii. CPU programează procesorul I/O pentru un transfer, pe care acesta îl execută folosind instrucțiunile proprii. Când transferul se termină, procesorul I/O întrerupe CPU pentru a comunica rezultatele transferului;
6. Microcontrollerul are memorie locală. El poate controla astfel mai multe EP cu o intervenție minimă din partea CPU. Memoria locală poate fi folosită și ca buffer de date, realizând astfel o rată de transfer mare.

Definiție: o magistrală este un subsistem cu funcția de comutator universal bidirecțional prin care se transferă date în interiorul unui sistem de calcul sau între sisteme de calcul. Schema bloc a unui sistem bazat pe magistrale este dată în figura 3.1:

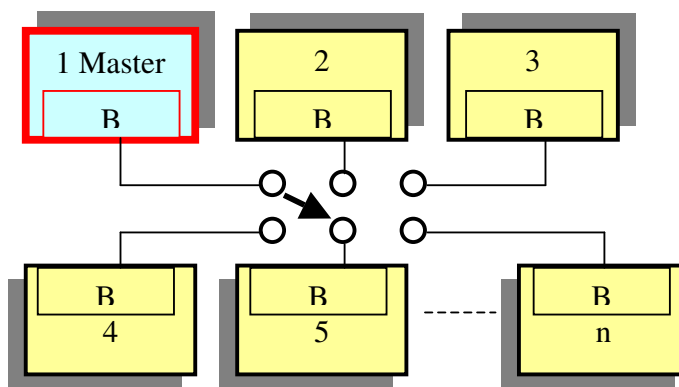


Figura 3.1. Structura unui sistem pe bază de magistrale

Sistemul prezentat în figura 3.1 este format din n subsisteme dintre care unul este Master la un moment dat iar celelalte Slave. Comutatorul este comandat de Master care stabilește subsistemul Slave cu care face transfer de date și stabilește sensul transferului.

Subsistemele pot fi explorate ciclic sau, mai eficient transferul poate fi realizat printr-o logică de priorități. Bufferele B optimizează traficul de date, aceste buffere fiind introduse în structura magistralei începând cu magistrala PCI.. Dacă fiecare subsistem poate transfera date cu viteza v_i și viteza cea mai mare este:

$v_{\max} = \max(v_i)$, atunci viteza magistralei V_M va fi:

$V_M = k \cdot v_{\max}$, adică mai mare (cu un coeficient k) decât viteza celui mai rapid subsistem

Magistralele pot fi de două tipuri:

1. Sincrone, la care există o linie de tact și toate transferurile se fac într-un număr întreg de tacte, ciclul fiind numit ciclu de magistrală. Acest tip de magistrală este cel mai simplu și ca urmare s-a răspândit și la microcontrollere;
2. Asincrone, transferul poate dura oricât, este nevoie însă de un protocol de dialog. Pentru ca un transfer nereușit să nu blocheze sistemul este nevoie de un mecanism de supraveghere a magistralei.

Prima magistrală a apărut în 1984 în structura calculatorului IBM PC și s-a numit **ISA** (Industry Standard Architecture). Magistrala este sincronă, are 16 biți de date, 24 de biți de adresă, subsamblele conectate se configurează manual, viteza maximă este de 16Mbps. Specificațiile tehnice ale magistralei au fost în domeniul public, ceea ce a determinat ca magistrala să fie un succes și ca urmare compatibilele IBM PC să se răspândească în toată lumea.

Specificațiile fiind libere, IBM nu a beneficiat financiar prea mult de pe urma acestei magistrale, așa încât a lansat în 1987 o nouă magistrală, superioară, pentru calculatoarele PS2 numită **MCA** (Microchannel). Caracteristicile ei: 32 de biți de adresă, 32 de biți de date, configurare automată, viteza maximă 32Mbps, cu posibilitatea de a lucra multiprocesor. Pentru a construi dispozitive MCA era nevoie de cumpărarea licenței de la IBM și au fost puține firme care să facă acest lucru, ca urmare calculatoarele cu MCA nu s-au răspândit.

Ca reacție la MCA, un consorțiu de firme (Compaq, Epson, Hewlett Packard, NEC, Olivetti și Zenith) a lansat magistrala **EISA** (Extended ISA) în 1988. Magistrala EISA are performanțe cel puțin atât de bune ca și MCA: 32 de biți de adresă, 32 de biți de date, configurare automată sau manuală, viteza maximă 120Mbps și asigură compatibilitate cu plăcile ISA. Specificațiile au fost în domeniul public și probabil magistrala ar fi avut succes.

EISA nu s-a răspândit pentru că INTEL a lansat în 1990 magistrala **PCI** (Peripheral Component Interconnect) cu specificații în domeniul public, cu un concept nou al arhitecturii. PCI poate lucra cu 32 sau 64 de biți de date la o viteză de maximum

2,112Gbps. Conceptul de ierarhizare a magistralei după viteză permite compatibilitatea cu magistrala ISA. În ultimul timp plachetele ISA au dispărut dar PCI a păstrat intern o magistrală de viteză mică numită LPC (Low Pin Count), de fapt o magistrală ISA cu semnale multiplexate pe aceleași linii pentru economia de pini.

3.2. Magistrale ierarhizate

Magistrala PCI are o arhitectură care permite existența a două magistrale pentru I/O, una de viteză mare și una de viteză mică, figura 3.2.. La magistrala de viteză mare se conectează dispozitivele rapide iar la cea lentă se cuplează canalele care necesită o viteză mică. Culorile sunt sugestive, cele mai calde sugerând o viteză mai mare.

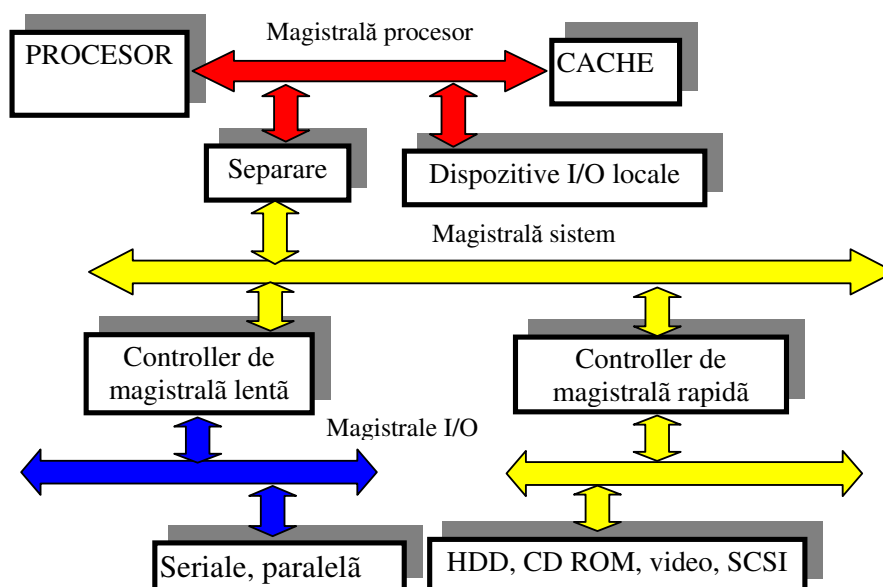


Figura 3.2. Principiul magistralei ierarhizate

Dacă se analizează cazul concret al sistemelor PC lucrurile devin mai clare. Structura ierarhizată a magistrelor este dată în figura 3.3. Se poate vedea structura pe 3 nivele, cel mai de sus fiind constituit pe lângă magistrala procesorului care are viteza cea mai mare, tactul fiind FSB (Front Side Bus). Pe această magistrală sunt conectate controllerul video integrat, controllerul de memorie DRAM și controllerul magistralei PCI. Pe al doilea nivel este situată magistrala PCI cu un tact de 33MHz, la care sunt conectate dispozitivele de I/O rapide, adică canalele USB, interfața SATA, interfața Ethernet 10/100Mbps. AL treilea nivel este reprezentat de magistrala LPC (standard ISA) la care sunt conectate cele mai lente dispozitive de I/O, adică canalele PS2 pentru tastatură și mouse. Probabil că în timp se va renunța la nivelul magistralei LPC, existând tendința ca

dispozitivele de I/O să devină mai rapide și migrarea către USB. În general creșterea vitezei interfețelor face ca acestea să urce pe diagrama magistralelor. Un exemplu clar este interfața Ethernet care la viteza de 10/100Mbps este conectată la magistrala PCI dar la viteza de 1Gbps este conectată la magistrala procesor.

Funcționarea sistemului ierarhizat de magistrale poate fi exemplificat prin modul în care se execută instrucțiunea MOV DX,AL la portul de ieșire 0378H, portul paralel. Instrucțiunea este analizată de controllerele de magistrală ierarhizate. Controllerul PCI are o tabelă de adrese de I/O și verifică dacă adresa din instrucțiune este alocată unui periferic rapid. În acest caz nu este alocată, așa că instrucțiunea este executată pe magistrala LPC. O instrucțiune de citire/ scriere cu hard discul ar fi fost executată pe magistrala PCI.

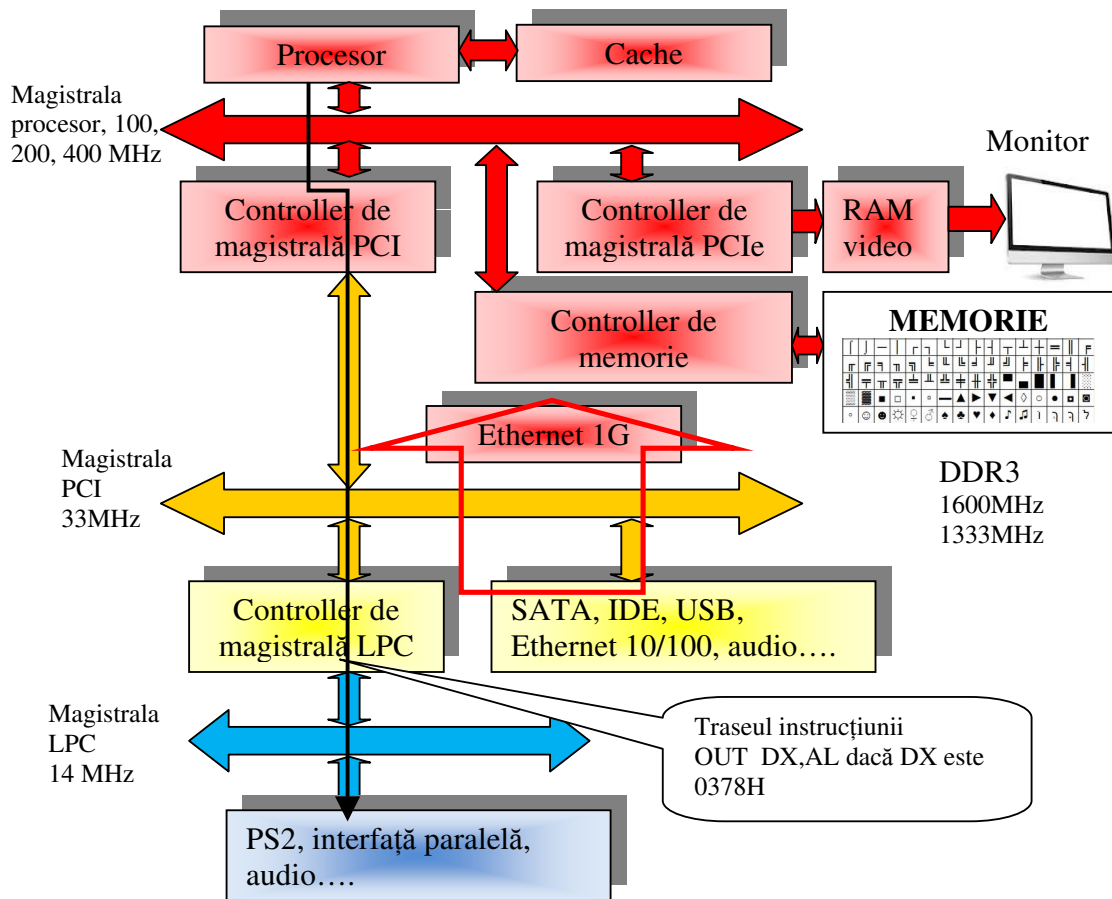


Figura 3.3. Principiul magistralii ierarhizate la PC

3.3. Diagrame de semnal la acces

Diagramele de timp ale accesului pe magistrală arată modul cum se desfășoară în timp transferul de date și rolul semnalelor importante de comandă.

Pe o magistrală sincronă transferul de date durează un anumit număr de tacte de ceas, perioada unui transfer fiind numit ciclu. Ciclurile pot fi de citire sau scriere (din perspectiva procesorului), pot fi cicluri de transfer cu memoria sau cu dispozitivele de intrare ieșire, pot fi cicluri de transfer gestionate de procesor sau cicluri de acces DMA. Un ciclu deosebit care poate exista la anumite sisteme este cel de întrerupere. Sunt exemplificate ciclurile de magistrală în cazul unei magistrale generice simple, asemănătoare cu cea mai simplă magistrală (ISA / LPC) la sistemele x86 și cu magistralele microcontrollerelor.

Ciclul de citire din memorie este reprezentat în figura 3.4. Un astfel de ciclu este declanșat de o instrucțiune de citire din memorie, la x86 este MOV AL, [BX].

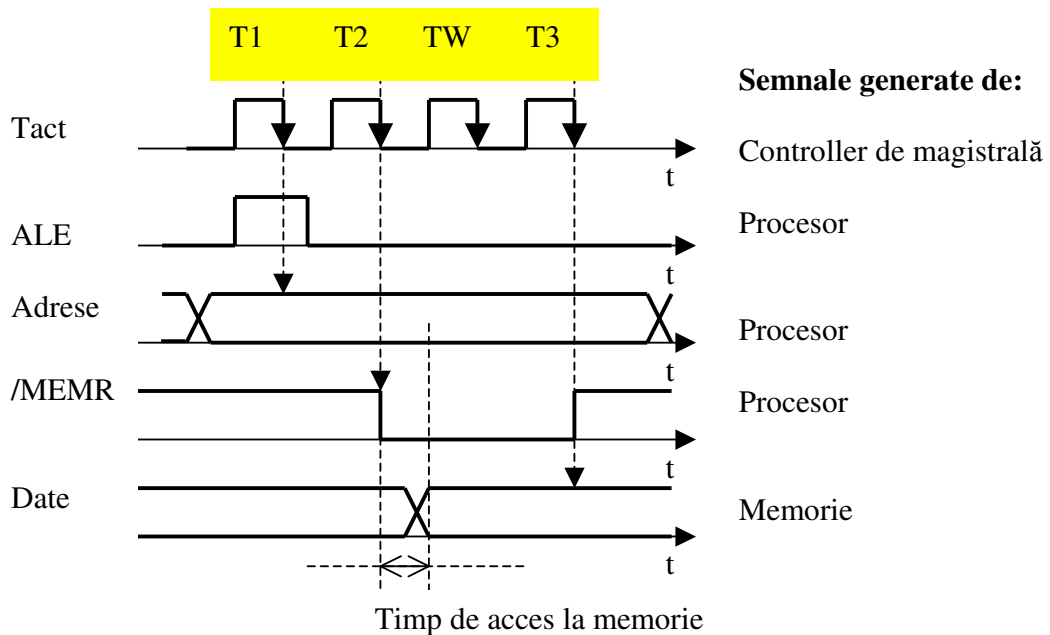


Figura 3.4. Ciclul de citire din memorie

Semnalul ALE (Address Latch Enable) este utilizat pentru memorarea adreselor într-un latch. Memorarea se face pe frontul primului impuls de tact. La executarea instrucțiunii de citire din memorie MOV AL, [BX] procesorul activează semnalul MEMR. După trecerea unui timp de întârziere datele sunt citite din memorie și apar pe magistrala de

date. Ele sunt eșantionate de frontul ultimului impuls de tact. De regulă memoria este mai lentă decât procesorul, de aceea este posibil ca la citire și scriere să se introducă un impuls de tact suplimentar TW numit impuls de *wait* care generează o stare de *wait*. La acest ciclu de citire celelalte semnale de comandă MEMW, IOR și IOW sunt inactive.

Ciclul de scriere în memorie este reprezentat în figura 3.5 și este asemănător cu ciclul de citire. Un astfel de ciclu este declanșat de o instrucțiune de scriere în memorie, la x86 este MOV [BX], AL.

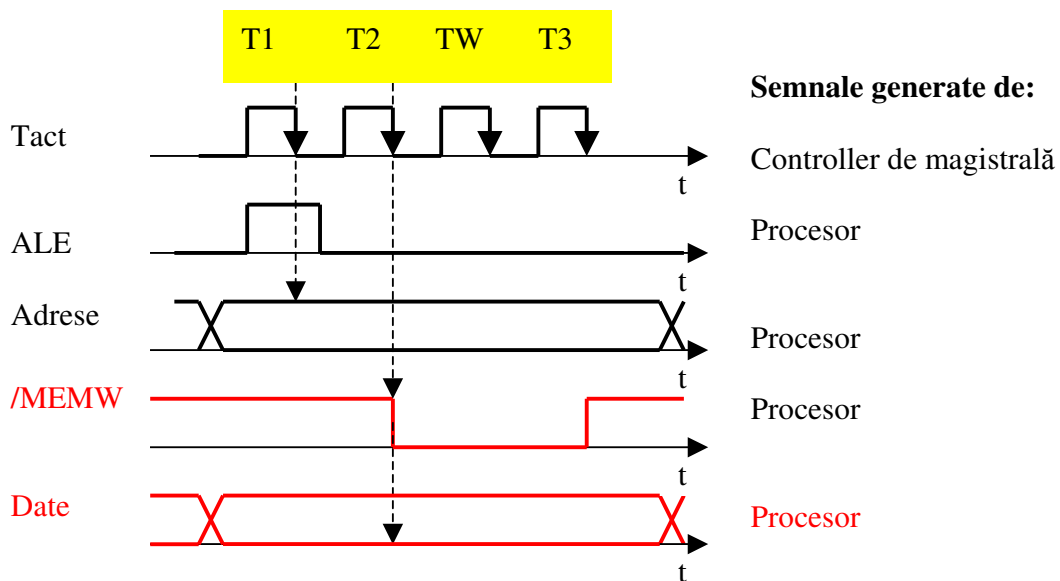


Figura 3.5. Ciclul de scriere în memorie

Instrucțiunea de scriere în memorie MOV [BX], AL activează semnalul MEMW, celelalte semnale de comandă fiind inactive. Modificările față de ciclul de citire au fost reprezentate cu roșu pentru a fi mai vizibile.

Ciclul de citire de la un dispozitiv de I/O este reprezentat în figura 3.6. Un astfel de ciclu este declanșat de o instrucțiune de citire de la un port, la x86 este IN AL, DX. Execuția acestei instrucțiuni activează semnalul IOR. După trecerea unui interval de timp de acces la dispozitivul de I/O datele sunt disponibile pe magistrala de date. Datele sunt eșantionate pe frontul crescător al semnalului /IOR. Dispozitivele de I/O sunt mai lente decât memoria așa încât poate fi nevoie de inserarea mai multor impulsuri de tact TW de așteptare (WAIT)

Ciclul de scriere la un dispozitiv de I/O este reprezentat în figura 3.7. Un astfel de ciclu este declanșat de o instrucțiune de scriere la un port, la x86 este OUT DX, AL.

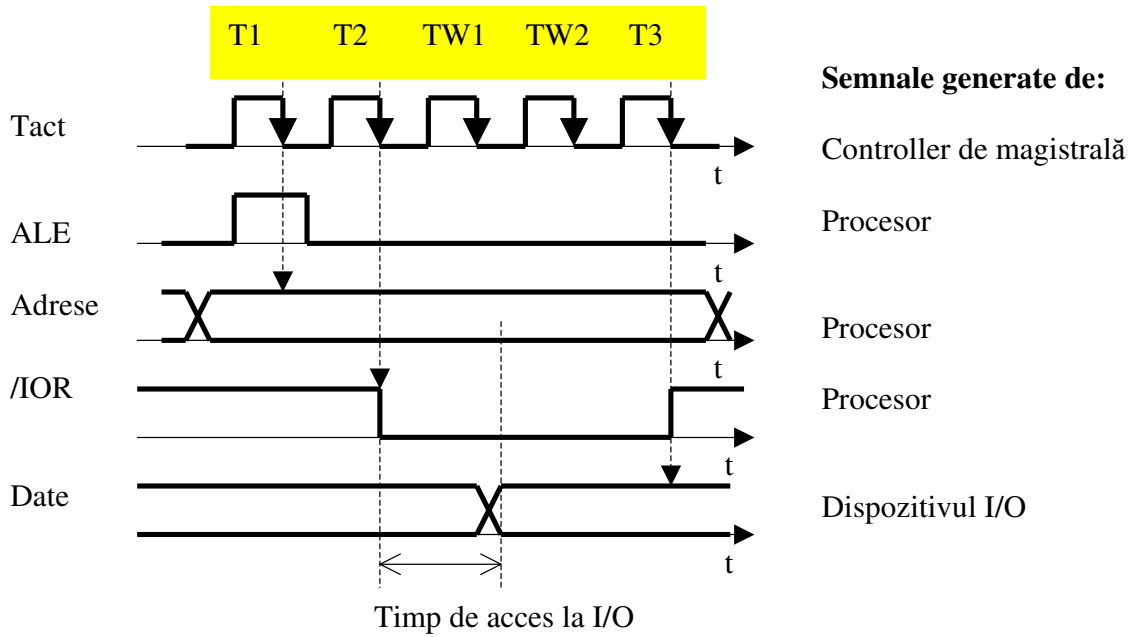


Figura 3.6. Ciclul de citire de la I/O

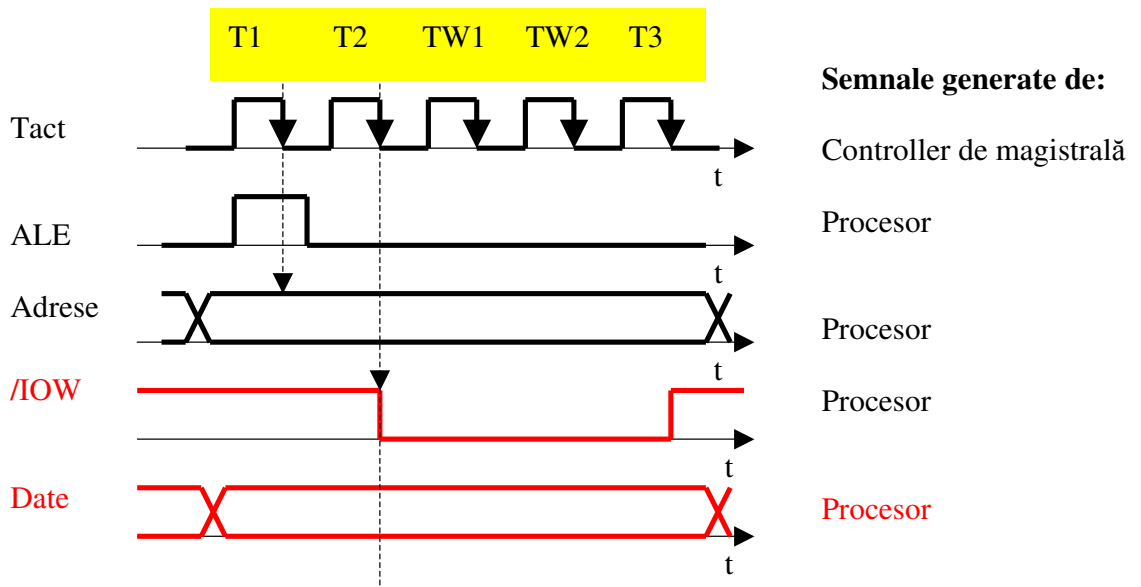


Figura 3.7. Ciclul de scriere la I/O

Modificările la ciclul de scriere față de ciclul de citire sunt marcate cu roșu pentru a ușura înțelegerea.

În ciclurile DMA procesorul nu are controlul magistralelor, controlul fiind preluat de controllerul de magistrală. Ciclul de scriere DMA reprezentat în figura 3.8. înseamnă citirea de la un dispozitiv de I/O și scrierea în memorie.

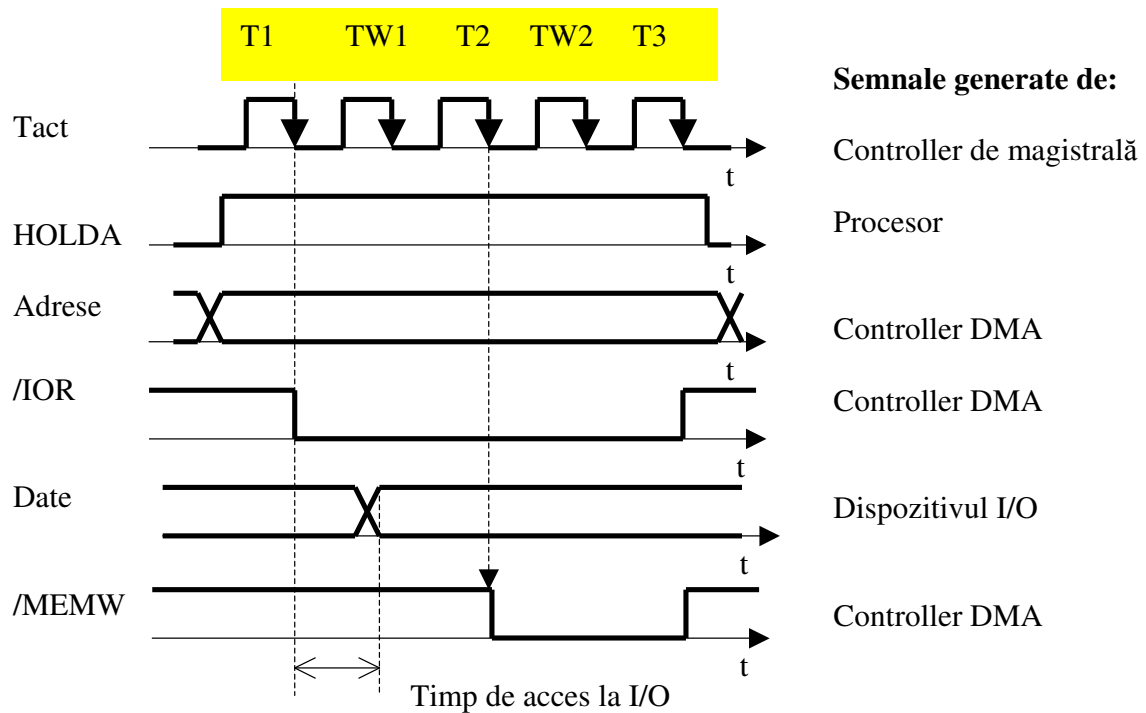


Figura 3.8. Ciclul de scriere DMA

Cedarea magistralelor de către procesor se face în urma unui dialog, după care procesorul confirmă cedarea prin activarea semnalului HOLDA. Controllerul DMA pune adresa de memorie pe magistrală și activează semnalul /IOR. Datele sunt puse de dispozitivul de I/O pe magistrală, după care controllerul DMA activează semnalul /MEMW și datele se înscriu în memorie.

Ciclul de citire DMA reprezentat în figura 3.9. înseamnă citirea din memorie și scrierea la un dispozitiv de I/O. Cu roșu au fost marcate modificările.

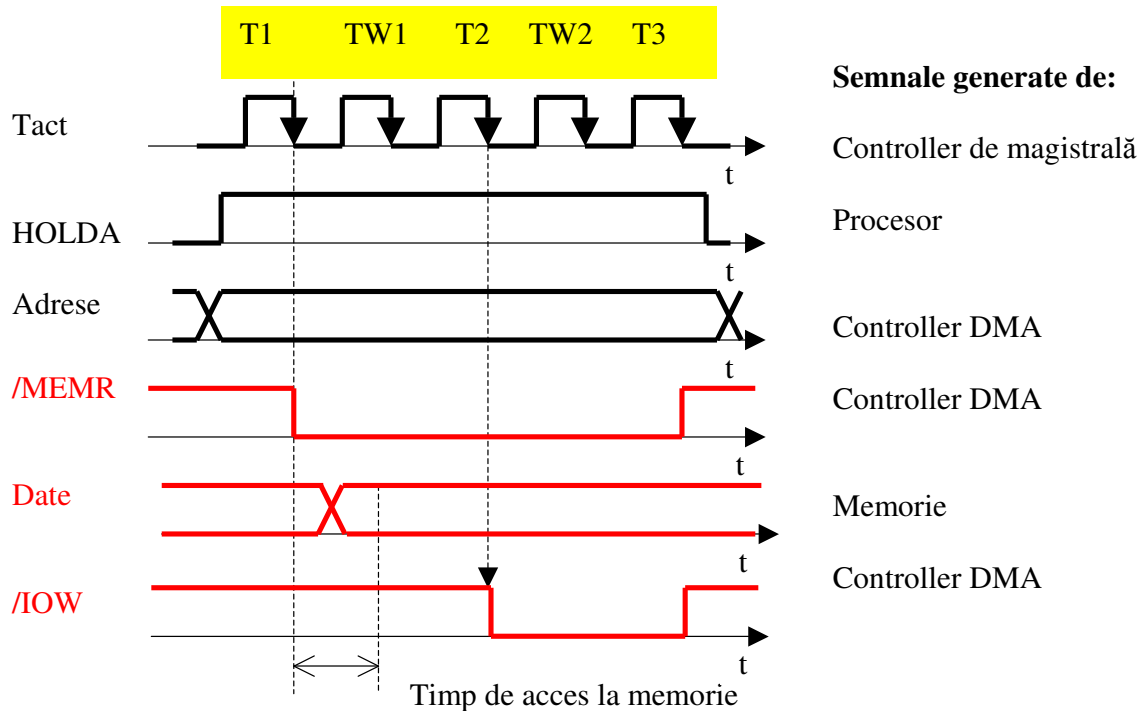


Figura 3.9. Ciclul de citire DMA

Din diagramele de semnal se poate determina aproximativ creșterea vitezei de transfer prin DMA. Un transfer programat înseamnă o instrucțiune de acces la memorie (4 tacte) și una de acces la un dispozitiv de I/O (5 tacte), în total 9 tacte. Un ciclu DMA care face același lucru durează 5 tacte, deci creșterea de viteză este de $9/5$ ori (de 1,8 ori). Desigur că mărirea vitezei de transfer depinde de numărul de tacte de *wait* introduse.

3.4. Magistrale multiplexate

Microcontrollerele au beneficiat la apariție de experiența dezvoltării magistralelor de PC și au împrumutat variantele cele mai simple de magistrale sincrone. Transferul de date pe o magistrală sincronă apare la execuția unei instrucțiuni a microcontrollerului. La execuția instrucțiunii, în funcție de instrucțiunea executată, se generează automat un semnal care stabilește spațiul de adresare și sensul transferului. În cazul în care spațiul de adresare este comun pentru zona de memorie și zona de I/O atunci există două semnale de comandă - RD și WR care stabilesc sensul transferului. Dacă spațiul de adresare este diferit pentru memorie și IO atunci există patru semnale- IOR, IOW, MEMR, MEMW.

Diagrame de timp pentru transferuri tipice pe o magistrală sincronă simplă cu spațiu comun pentru memorie și I/O au fost prezentate anterior.

La un PC la magistralele vechi liniile de adresă și de date sunt diferite pentru că nu s-a pus problema economiei de spațiu. De regulă, în microcontrollere și la magistralele noi de PC nu se poate accepta un număr atât de mare de linii din cauza costurilor de realizare a circuitului, de aceea liniile de adrese și date sunt multiplexate. Aceasta înseamnă cuplarea unui latch de adrese în exteriorul microcontrollerului care să fie încărcat cu adrese, comanda latch-ului fiind realizată cu semnalul ALE (Address Latch Enable). O diagramă de timp în acest caz este dată în figura 3.10:

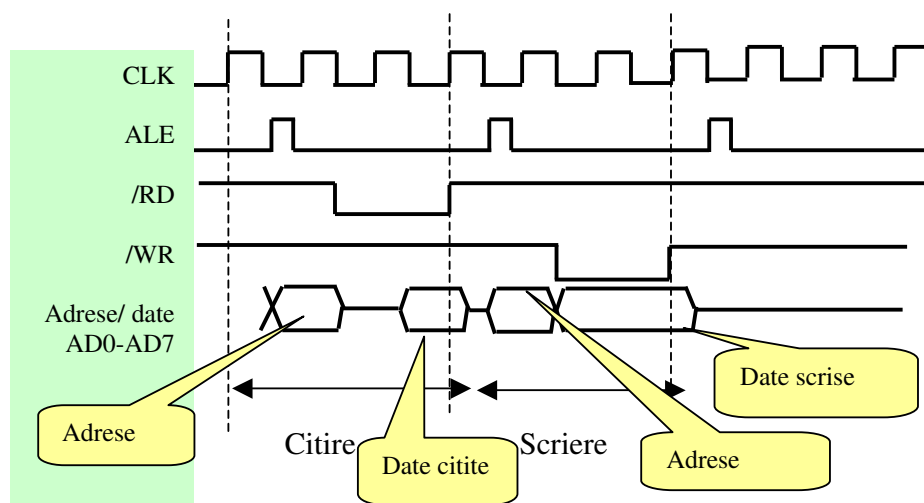


Figura 3.10. Diagrama de timp pentru un ciclu de citire urmat de un ciclu de scriere în cazul unei magistrale de adrese și date multiplexate

3.5. Magistralele PCI și PCI Express

Asociația PCI-SIG (www.pcisig.com) grupează un număr mare de producători (circa 900) interesați în armonizarea structurilor și caracteristicilor magistrelor PCI. Tabelul 3.1 arată evoluția în timp a caracteristicilor magistrelor PCI:

Tabel 3.1.

Magistrala	Anul apariției	Debit maxim
PCI 33MHz	1993	133Mo/s
PCI 66MHz	1995	266Mo/s
PCI-X 133MHz	1999	533Mo/s

PCI-X 266MHz	2002	1066Mo/s
PCI-X 533MHz	2002	2131Mo/s
PCI Express	2002	16Go/s (x32)

Specificațiile PCI permit cuplarea a maximum 32 de dispozitive pe magistrală. Cu toate acestea, din motive de încărcare a magistralei, practic se pot conecta doar 5-10 dispozitive. O diagramă de semnale tipică pentru accesul la magistrala PCI este dată în figura 3.11.

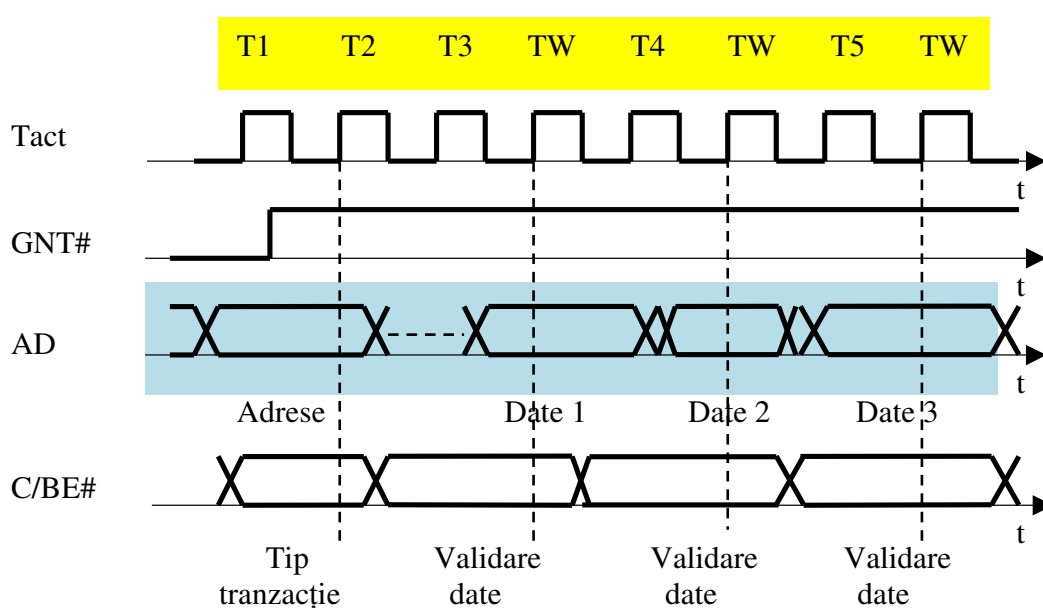


Figura 3.11. O diagramă de semnale pentru accesul la magistrala PCI

Este foarte importantă caracteristica PCI de a permite transferuri în salvă (burst). Magistrala PCI admite 12 tipuri de tranzații (cicluri) printre care tranzații cu memoria, cu dispozitive de I/O, de configurare, etc.

Liniile de date și adrese sunt multiplexate (AD) pentru a reduce numărul de linii. Un dispozitiv master cuplat pe magistrală poate solicita magistrala cu un semnal REQ# și când aceasta este liberă controllerul de magistrală va răspunde cu GNT#. La primul tact pe liniile AD masterul pune adresa, iar pe liniile C/BE# se pune tipul tranzației. După adresă, pe liniile AD urmează datele, cuvânt după cuvânt. Se poate vedea că după fiecare tact urmează un tact de *wait* pentru a permite sincronizarea cu date puse pe magistrală cu diferite întârzieri. Pe liniile C/BE# se pun cuvinte de validare a datelor.

Una dintre problemele majore ale magistrelor paralele este transmisia semnalelor pe mai multe linii (16, 32 sau 64), apropiate între ele, la viteze de transfer din ce în ce mai mari. Efectele perturbatoare care se manifestă la aceste transmisii, în special reflexiile și diafonia nu mai pot fi neglijate și limitează creșterea în continuare a vitezei de transfer. Soluții care micșorează diafonia și reflexiile cum ar fi de exemplu trasee mai late (cu rezistență mai mică), introducerea traseelor de masă între traseele de semnal etc. nu pot fi practic aplicate pentru că dimensiunile noilor echipamente trebuie să fie din ce în ce mai mici. Sunt foarte interesante soluții de codificare a datelor transmise pe magistrala paralelă astfel încât combinațiile de nivele logice pe linii adiacente care produc diafonii să fie eliminate. În figura 3.12. se poate vedea un asemenea sistem care verifică combinațiile care produc diafonii și le înlocuiește, semnalizând acest lucru cu o linie specială care se adaugă la liniile magistralei.

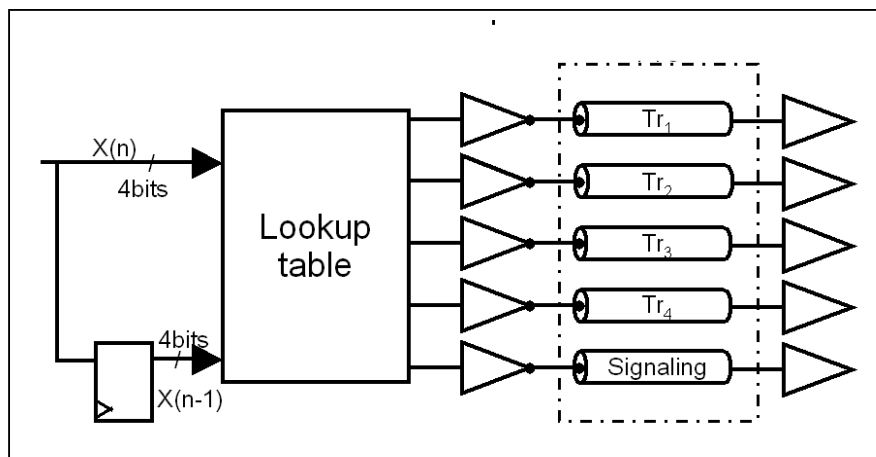


Figura 3.12. Eliminarea combinațiilor care produc diafonii (Sursa: M. Dărăban, Teză de doctorat, Cluj Napoca, 2012, online:

http://www.etti.utcluj.ro/download/988_PhD_Thesis.pdf)

Un răspuns posibil la aceste probleme este apariția unui nou tip de magistrală. PCI Express are o altă filozofie a magistrelor, de la magistrale paralele la care fiecare linie este de semnal este un traseu clasic, la o magistrală paralelă care conține mai multe grupuri de linii de comunicații seriale numite canale. Un canal conține 2 linii de transmisie seriale unidirecționale (simplex), fiecare linie fiind compusă din 2 fire cu transmisie diferențială (High-speed LVDS, Low Voltage Differential Signaling), figura 3.13.

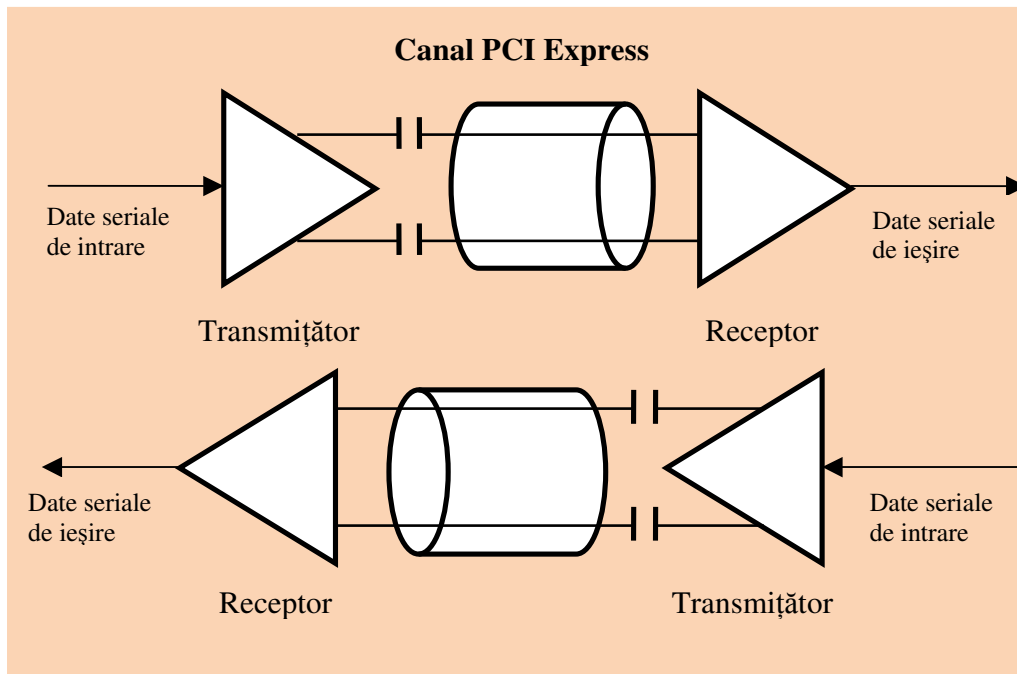


Figura 3.13. Canal PCI Express

Transmisia serială la PCI Express este o transmisie punct la punct, sincronă, cu cadre de date, cu refacerea tactului din datele citite, codarea fiind o codare de grup numită 8b/10b, care va fi tratată într-un modul ulterior, dedicat interfeței IEEE1394.

Magistrala PCI Express poate conține un canal (PCIe x2), 2 canale (PCIe x4), ș.a.m.d. până la 16 canale (PCIe x32).

Controllerul de magistrală conține un comutator (Switch) care comută un dispozitiv master la unul slave, ca în diagrama generală din figura 3.1. O schemă generală a magistralei PCI Express este dată în figura 3.14.

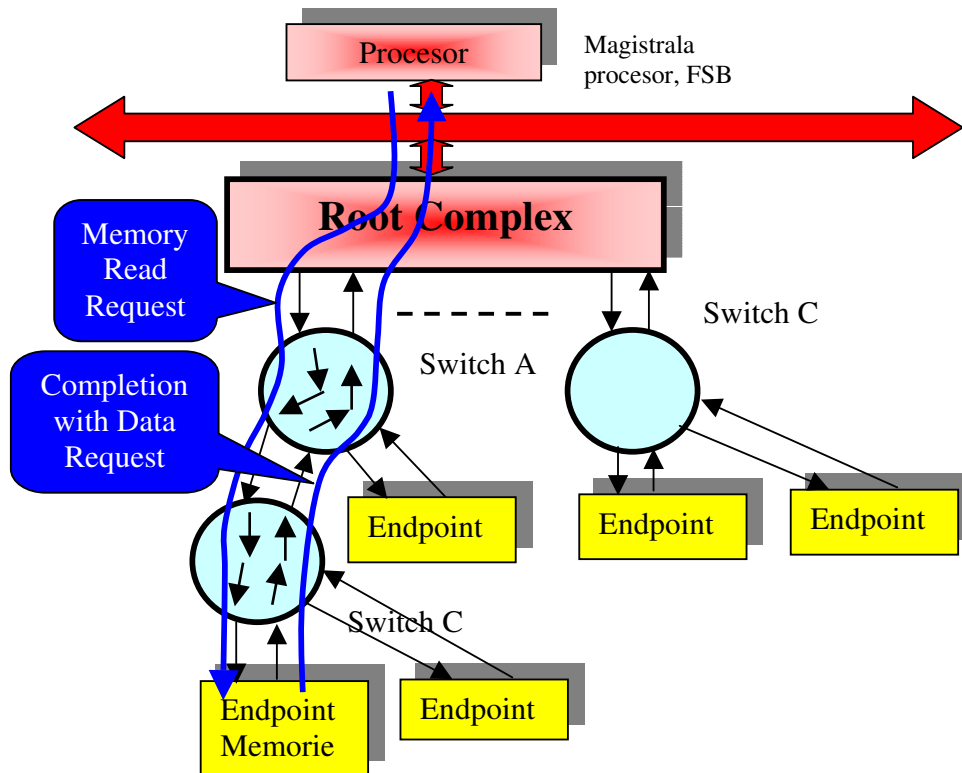


Figura 3.14. Structura magistralei PCI Express

Arhitectura magistralei seamănă cu arhitectura USB, viteza superioară fiind asigurată de un număr mai mare de canale seriale prin care se transferă date simultan. Dacă procesorul are nevoie de un transfer de date de citire de la memorie atunci trimite un cadru de comandă **Memory Read Request**. Aceasta ajunge la **Endpoint-ul memorie** prin trecerea prin **Switch-ul A și C**, comandate pentru a asigura traseul către **Endpoint-ul dorit** de către controllerul de magistrală. Memoria răspunde acestei cereri cu cadrul **Completion with Data Request**. Transferul de date este bidirecțional full duplex prin fiecare canal serial cu câte două linii de transmisie. Toate dispozitivele cuplate la magistrală conțin buffere de transmisie și recepție.



Rezumat

Modulul începe cu definițiile și istoricul magistrelor, punctând caracteristicile care au dus la dezvoltarea unor tipuri de magistrale și la dispariția altora. Principiul ierarhizării magistrelor este ilustrat prin prezentarea pe scurt a magistralei calculatorului PC. Se prezintă apoi detaliat o magistrală simplă, bazată pe structura magistralei ISA, care poate fi acum întâlnită în structura multor microcontrollere. Se prezintă pe scurt o structură care conține linii de adrese și date multiplexate, deseori întâlnită la microcontrollere. Modulul se încheie prin prezentarea structurii magistralei PCI și a noii magistrale PCI Express care acum este cea mai răspândită magistrală la PC-uri. Se discută tendința de trecere de la magistrale clasice la magistrale paralele cu grupuri de canale seriale.



Bibliografie

1. M. Romanca, P. Ogrutan, *Sisteme cu calculator incorporat. Aplicații cu microcontrollere*, Editura Universitatii Transilvania Brasov, 2011, pag. 5-18 online la: <http://vega.unitbv.ro/~ogrutan/Microcontrollere2011/2-magistrale.pdf>
2. P. Ogruțan, C. Gerigan, N. Banciu *Memorii, interfețe și periferice. Interfețe specializate*, Ed. Transilvania Brașov, 2003, 190 pagini, ISBN 973-635-118-1, pag. 1-30, online la: <http://vega.unitbv.ro/~ogrutan/ii/chipset.pdf>
3. C. Gerigan, P. Ogrutan, *Tehnici de interfatare*, Editura Transilvania, Brasov 2000, pag 24-45, online: <http://vega.unitbv.ro/~ogrutan/ti/cap3.pdf>
4. R. Budruk, D. Anderson, T. Shanley, *PCI Express System Architecture*, Addison-Wesley, 2003, online: <http://www.mindshare.com/files/ebooks/pci%20express%20system%20architecture.pdf>

Test de autoevaluare



1. Pe o magistrală pot exista mai multe dispozitive master doar dacă:

R

- (a) numărul de dispozitive slave este mai mare decât o anumită limită (2^4)
- (b) dispozitivele sunt master la momente diferite de timp
- (c) numărul de dispozitive master este mai mic decât 2
- (d) dacă magistrala este sincronă

I. Vezi pagina 2

2. Semnalul ALE este pus de procesor pentru a semnaliza un ciclu de scriere

R

- adevărat
- fals

I. Vezi pagina 6

3. De ce un ciclu DMA este mai lung decât un ciclu de memorie?

R

- (a) pentru că semnalul HOLDA este mai lung decât ALE
- (b) pentru că datele sunt furnizate de memorie cu o întârziere mai mare
- (c) pentru că datele sunt furnizate de memorie cu o întârziere mai mare
- (d) pentru că pe în plus față de citirea/scrierea în memorie face o scriere/citire la I/O

I. Vezi pagina 9

4.Care este motivul pentru care liniile de date și adrese sunt multiplexate?

R

- (a) Pentru a mări viteza de transfer
- (b) Pentru a micșora numărul de linii de magistrală
- (c) Pentru că spațiul de memorie este comun cu cel de I/O
- (d) Pentru a micșora costurile de realizare a circuitului

I. Vezi subcapitolul 3.4.

5.Desenați variația în timp a semnalelor Tact, ALE, adrese, IOR, IOW, MEMR, MEMW, date, pentru instrucțiunea OUT DX,AL

R



I. Vezi subcapitolul 3.3., diagramele 3.4, 3.5, 3.6, 3.7.

6. Magistrala PCI Express conține mai multe canale (1-16), fiecare canal conține 2 linii simplex, fiecare linie fiind compusă din 2 fire cu transmisie diferențială

R

adevărat

fals

I. Vezi pagina 13

R

Răspunsuri corecte:

1. b, vezi definiția magistralei de la pag. 2
2. fals, vezi definiția semnalului de la pagina 6
3. d, analizați figurile 3.6, 3.7, 3.8, 3.9.
4. b, d, subcapitolul 3.4.
5. diagrama 3.7. plus semnalele IOR, MEMR și MEMW inactive (stau la zero logic sau la 1 logic întreg ciclul)
6. adevărat, vezi definiția de la pagina 13