

## Capitolul 2: Magistrala externă

### 2.1. Magistrale

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 [1]. Schema bloc a unui sistem bazat pe magistrale este dată în figura 2.1.

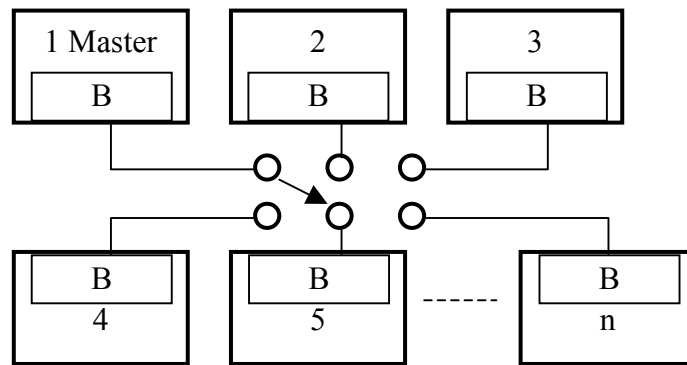


Figura 2.1: Structura unui sistem pe bază de magistrală

Sistemul prezentat în figura 2.1 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.

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 IO (In/Out) 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- IORD, IOWR, MEMRD, MEMWR. O diagramă de timp pentru un transfer tipic pe o magistrală sincronă simplă cu spațiu comun pentru memorie și IO în care sunt prezentate un ciclu de citire și unul de scriere [2] este dată în figura 2.2.

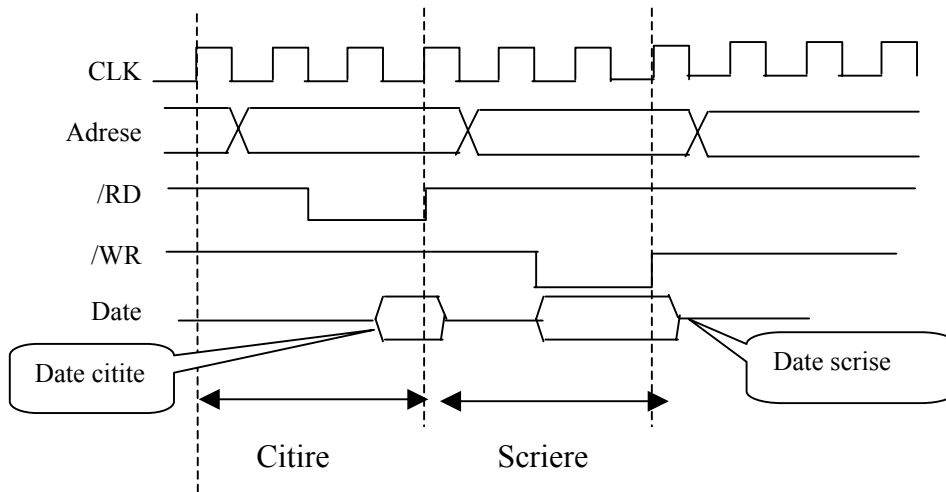


Figura 2.2: Diagrama de timp pentru un ciclu de citire urmat de un ciclu de scriere

Semnalul de tact este reprezentativ pentru magistralele sincrone. Activarea semnalului de /RD înseamnă că datele sunt citite de pe magistrală într-un registru al UC iar activarea semnalului /WR înseamnă că datele sunt scrise în memorie sau într-un dispozitiv de I/O. În această diagramă se remarcă faptul că liniile de adresă și de date sunt diferite.

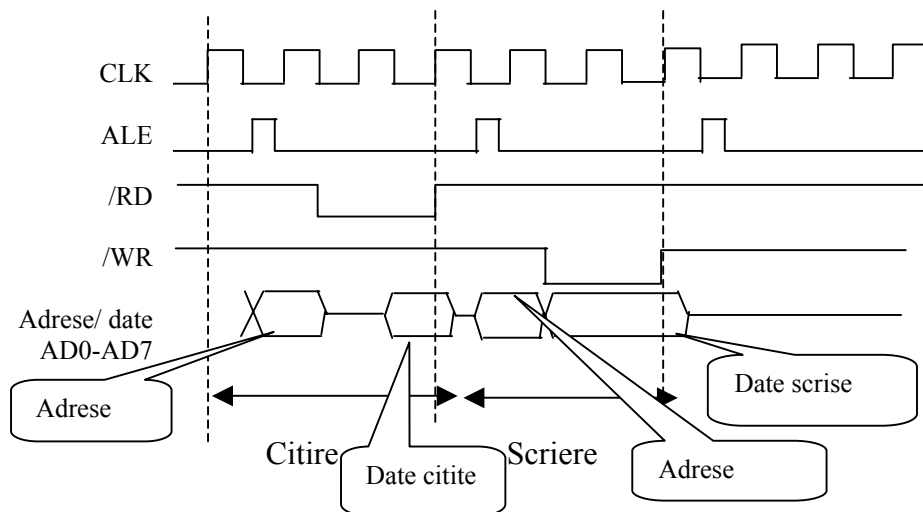


Figura 2.3: Diagrama de timp pentru un ciclu de citire urmat de un ciclu de scriere în cazul unei magistrale de adresă și date multiplexate

De regulă, în microcontrollere nu se poate accepta un număr atât de mare de linii din cauza costurilor, 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 2.3.

## 2.2. Selectarea unui dispozitiv pe magistrală

La un moment dat pe o magistrală sunt active 2 subsisteme, dintre care unul este Masterul. Iar celălalt este subsistemul Slave selectat de Master, figura 2.4.

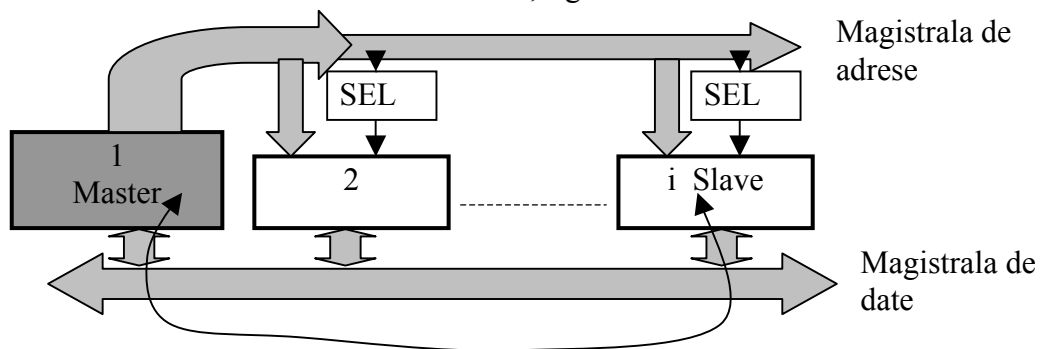


Figura 2.4: Selecția prin adrese

Transferul de date are loc între Master și subsistemul  $i$  pe magistrala de date. Liniile de adresă necesare pentru adresarea locațiilor sau registrelor interne ale subsistemelor sunt conectate direct la acestea. Cu liniile rămase libere se selectează subsistemul cu care Masterul transferă date. Subsistemul selectat devine activ, celelalte subsisteme intră în înaltă impedanță la interfața cu magistrala. Selecția poate fi:

1. Selecția liniară se poate realiza dacă numărul liniilor de adresă rămase disponibile este mai mare sau egal cu numărul circuitelor care trebuie selectate. Fiecare circuit este selectat cu o linie de adresă, avantajul soluției fiind simplitatea iar dezavantajul fiind faptul că se pierde din spațiul de adresare;
2. Selecția decodificată, în care liniile de adresă rămase libere sunt introduse într-un decodificator DCD, iar ieșirile DCD selectează fiecare un circuit. Se poate astfel folosi întregul spațiu de adresare;
3. Selecția mixtă, în care unele linii de adresă libere selectează direct câte un circuit în timp ce alte linii selectează prin intermediul unui DCD.

În primul exemplu se propune conectarea pe o magistrală de adrese de 16 biți a unor circuite de memorie de 16k. Selecția liniară și harta memoriei sunt date în figura 2.5.

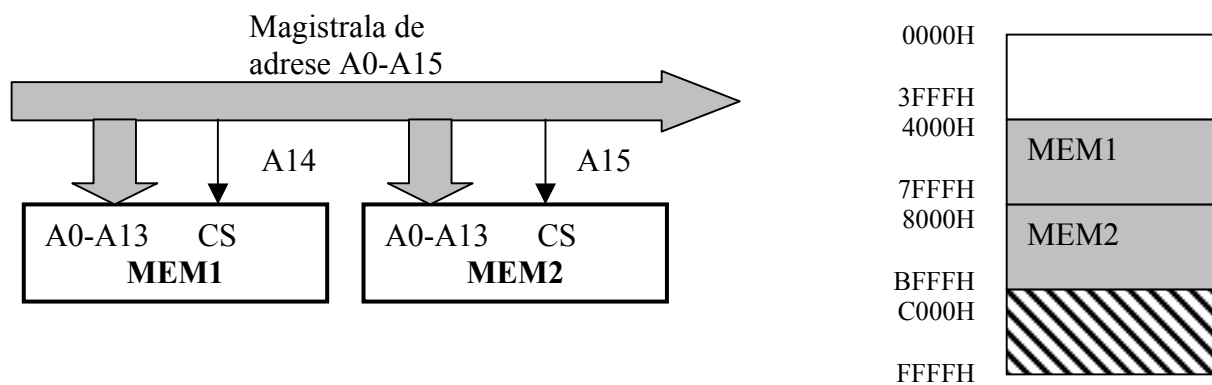


Figura 2.5: Selectarea liniară în cazul memoriilor (CS activ în 1)

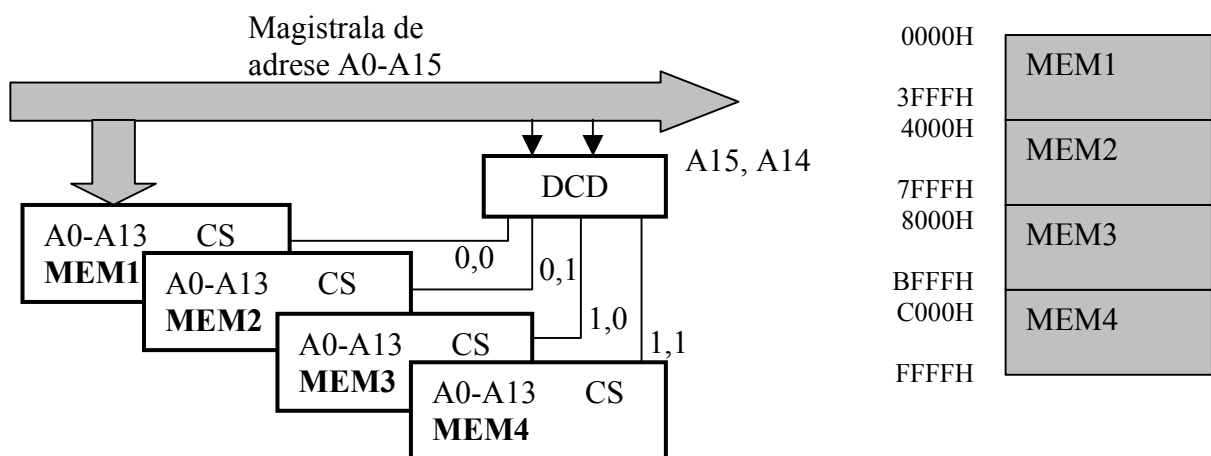


Figura 2.6: Selectarea decodificată în cazul memoriilor

Se poate observa că în cazul selectării decodificate se folosește întreaga zonă de adresare. În cazul conectării unui dispozitiv de IO se aplică aceleași reguli, diferența fiind dată de faptul că dispozitivele de IO au mai puține linii de adresare pentru regiștrii interni, ca urmare rămân mai multe linii de adresă libere. În al doilea exemplu se conectează o interfață paralelă de tip Intel 8255 pe o magistrală de adrese de 16 biți. I8255 utilizează 2 linii de adresă pentru selectarea celor 3 regiștrii interni. Să presupunem că adresa de bază la care dorim conectarea circuitului este 0180H. În figura 2.7 este dată schema bloc de conectare și harta memoriei.

Selecția prin decodificare a fost realizată cu 4 DCD cu 4 intrări. Schema bloc este detaliată în partea de selecție. După programarea circuitului I8255 se pot trimite /recepționa date de un octet cu o singură instrucțiune sau două, dacă datele sunt extrase din memorie, ceea ce înseamnă, de exemplu la un microcontroller RISC cu instrucțiuni executate într-un singur tact o rată de transfer de ordinul de mărime a tactului. La MC complexe, așa cum sunt de exemplu MC pe 32 de biți de la Fujitsu se pun la dispoziția utilizatorului semnale de CS pentru diferite zone de memorie liberă pentru a ușura implementarea aplicațiilor care necesită lucrul pe o magistrală externă.

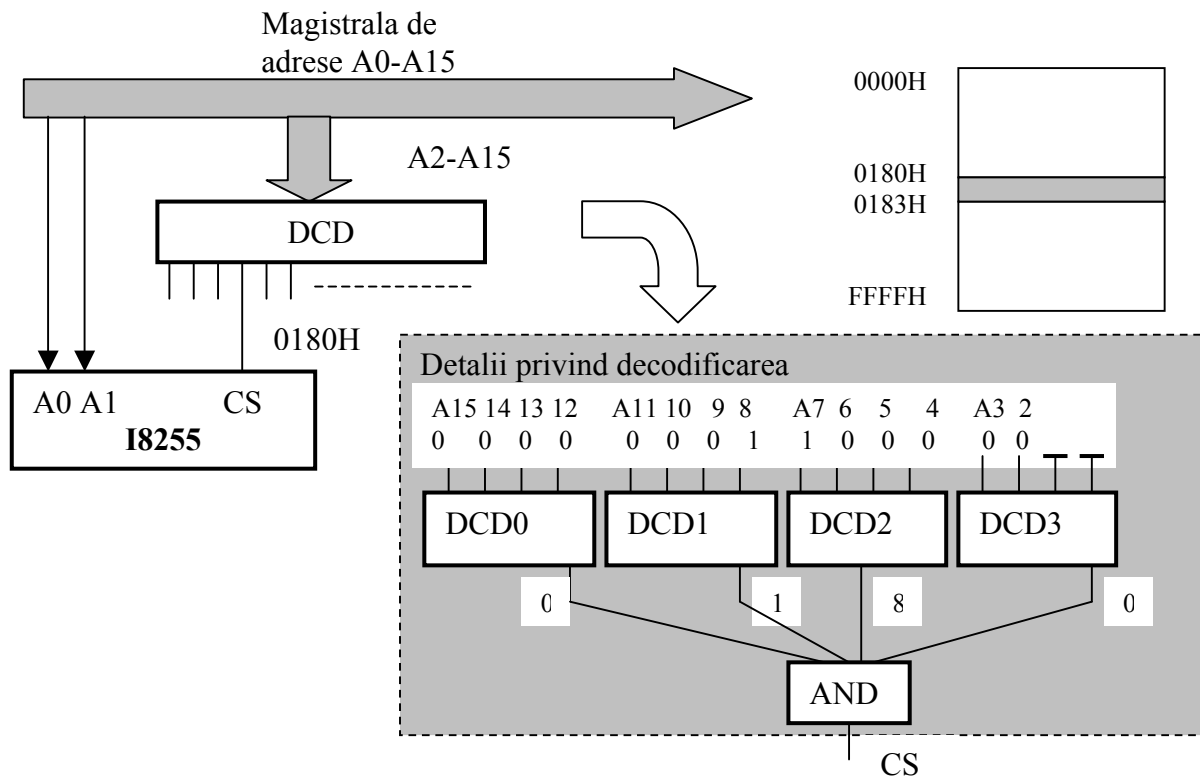


Figura 2.7: Conectarea unui circuit interfață paralelă I8255 pe magistrală

Dacă conectăm același circuit la porturi paralele obținem schema bloc din figura 2.8.

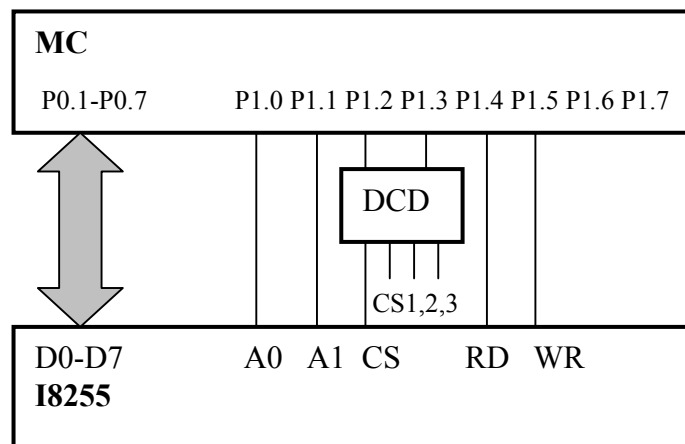


Figura 2.8: Conectarea unui circuit interfață paralelă I8255 la porturi paralele

Portul 0 al MC este folosit pentru transferul de date pe 8 biți, bidirecțional. Liniile 0 și 1 din portul P1 sunt folosite pentru selectarea registrelor interne ale I8255. Liniile 2 și 3 prin decodificare pot selecta unul din patru circuite conectate cu liniile de date la portul 0 al MC. Liniile 3 și 4 ale portului 1 stabilesc sensul transferului. Se poate observa din schema bloc faptul că o conectare la porturi paralele este mai simplă decât una pe magistrală, mai ales dac numărul de circuite conectate este mic.

Pentru un transfer de un octet se pune octetul pe magistrala de date apoi se selectează portul intern al I8255 prin punerea unui cuvânt pe portul 1, apoi se trimite un RD sau WR, ceea ce înseamnă un cuvânt pe portul 1 care face RD sau WR 1, urmat de un cuvânt care face RD sau WR 0. Din această succesiune de cuvinte de comandă se poate deduce că transferul este de cel puțin 4 ori mai lent decât dacă I8255 este conectat pe magistrală.

### 2.3.Exemplu la microcontrolerele MCS 51

Familia MCS 51 prevede magistrala externă chiar și la modelele mai ieftine, așa cum este de exemplu MC Atmel AT89LS51 [3]. Pe magistrala externă poate fi conectată memoria de date și memoria de program, spațiile de adresare fiind separate. Portul 0 poate fi configurat ca port de adrese și date multiplexate pe 8 biți astfel: P0.0 – AD0... P0.7 – AD7. Portul 2 generează adresele superioare pentru adresarea pe 16 biți (MOVX @ DPTR). Dacă se folosesc adresări pe 8 biți la portul 2 (MOVX @ RI) este generat conținutul registrului special P2. Portul 2: P2.0 – A8... P2.7 - A15. Semnalele de comandă generate sunt ALE, /RD, /WR și /PSEN (Program Store Enable) care este un strob pentru memoria externă de program. O diagramă de timp de acces la magistrală în cazul unui transfer de citire cu memoria de date (/PSEN inactiv):

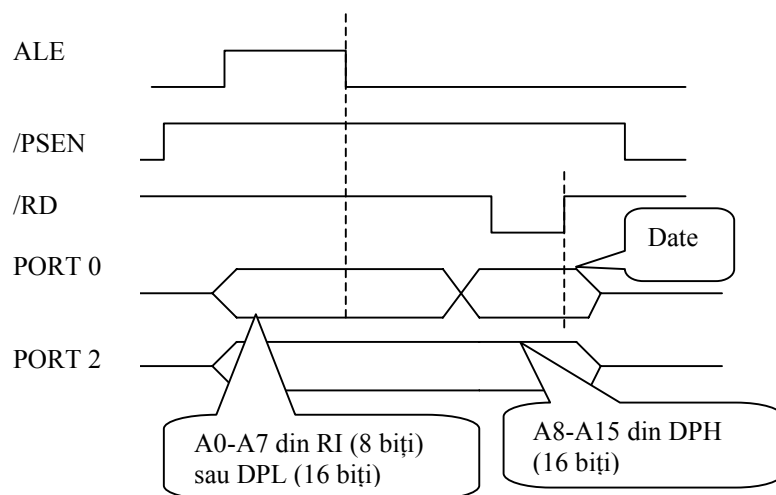


Figura 2.9: Diagrama de timp la un ciclu de citire dintr-o memorie externă de date

O aplicație de conectare la un MC din familia MCS 51 a unei memorii externe EPROM de 64koceteți pe magistrală și a unui afișaj LCD pe un port paralel [5] este dată în figura 2.10.

Pentru a adresa memoria EPROM se folosesc 16 biți de adresă, cei mai puțin semnificativi la portul P0, cei mai semnificativi la portul P2. Un impuls pe linia ALE încarcă biții de adresă din portul P0 în latch-ul 74HC573. Se activează PSEN pentru că în EPROM sunt stocate instrucțiuni (memorie externă de program), datele fiind pe liniile de date din portul P0 putând fi citite (/RD activ) sau scrise (/WR activ).

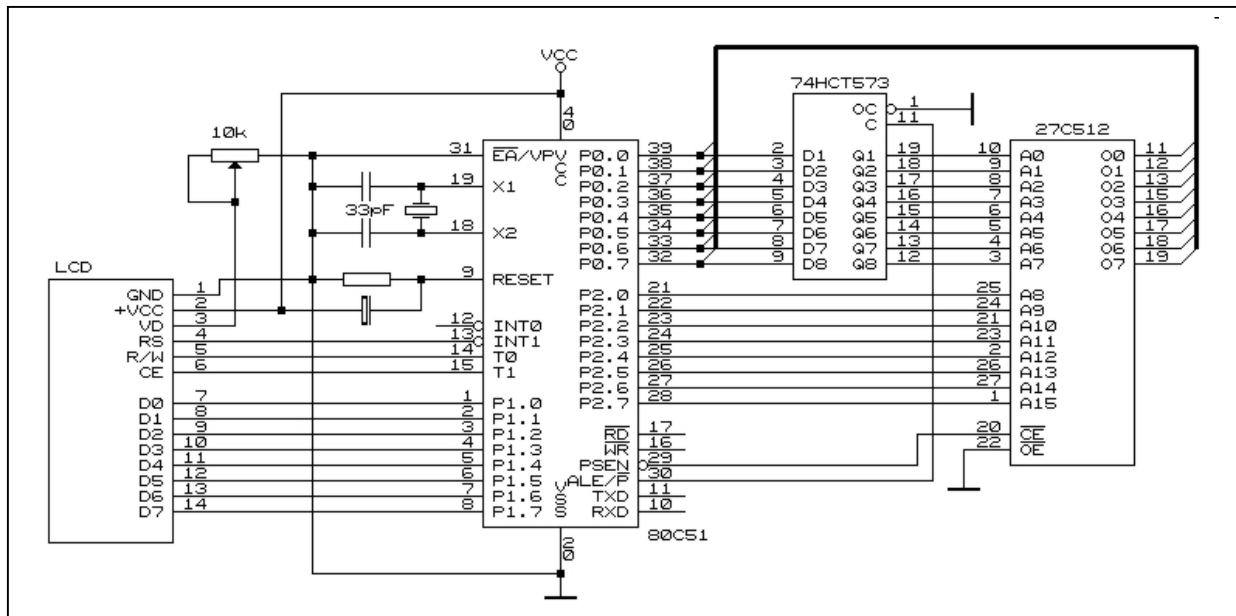


Figura 2.10: Conectarea unei memorii externe și a unui LCD la un MC 80C51

Pentru ca MC să execute codul stocat în EPROM trebuie ca semnalul EA să fie 0. Afișajul LCD este conectat cu liniile de date la portul P1 și semnalele de comandă astfel: RS – datele transmise pot fi un caracter de afișat sau o comandă, RW – sensul de transfer al datelor, CE – comanda de afișare. Un exemplu de programare:

MOV P1,A	datele de afișat sunt trimise la portul P1
SETB RS	datele reprezintă un caracter de afișat
CLR RW	sensul este de scriere în afișaj
NOP	întârziere
SETB CE	comanda de afișare
NOP	întârziere
NOP	
CLR CE	linia de comanda de afișare este readusă la 0
RET	întoarcere în programul principal

Înainte de a se lucra cu afișajul LCD este nevoie de trimiterea unor date pentru inițializare, funcție de tipul afișajului.

## 2.4.Exemplu la microcontrollerele ATMEL RISC

MC AVR pe 8 biți Atmega64 admite lucrul cu magistrală externă pentru cuplarea unei memorii externe printr-o interfață internă specială numită XMEM. Magistrala externă este pe 8 biți de date și 16 biți de adresă astfel: PA0- AD0... PA7-AD7, PC0-A8...PC7-A15, PG0-/WR, PG1-/RD, PG2-ALE. Datele sunt multiplexate cu adresele pe portul PA. O schemă bloc de conectare a unei memorii externe este dată în figura 2.11.

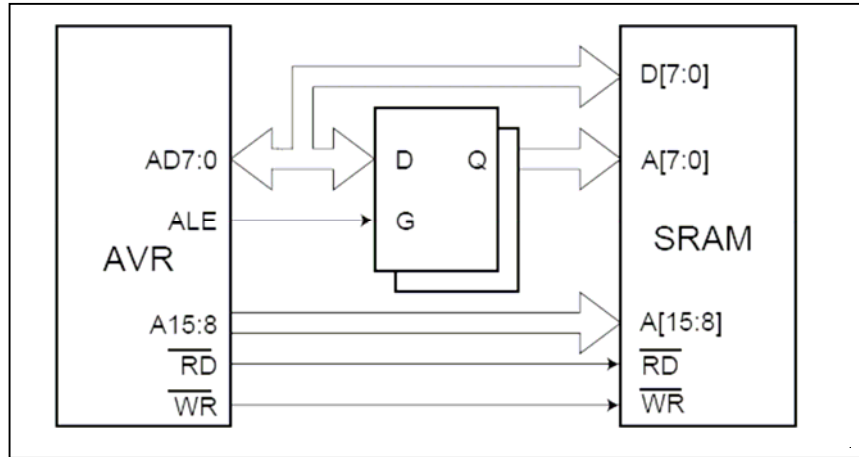


Figura 2.11: Schema bloc de conectare a unei memorii externe la un Atmega64

XMEM va detecta un acces la memoria externă și nu va ține cont de setările de direcție ale porturilor PA și PC. Viteza de transfer a datelor este mare, de aceea se impun condiții de viteză pentru latch-ul de adrese. Programarea și controlul modului de lucru se face cu registrele: MCUCR (MCU Control Register), XMCRA și XMCRB (External Memory Control Register A și B). Pentru că memoriile au timpi diferiți de acces se pot defini 4 variante de lucru cu memoria externă prin inserarea a 0, 1 sau 2 stări de WAIT.

O diagramă de timp pentru accesul la memoria externă pe 16 biți, la citire este dată în figura 2.12.

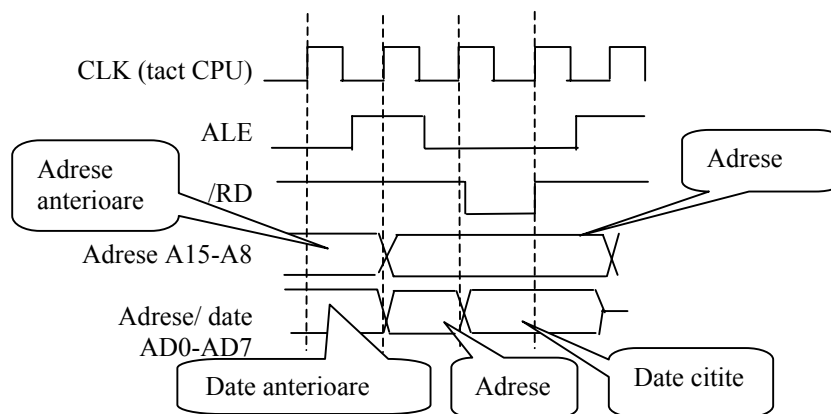


Figura 2.12: Diagrama de timp pentru un acces la citire

O aplicație de conectare a unui LCD la un MC RISC pe magistrala externă [6] este dată în figura 2.13.

Afișajul este validat (Enable) cu linia A15 când are loc o scriere /WR sau o citire /RD. Liniile de date / adrese inferioare sunt folosite doar ca date. Selecția zonei de adresare este simplă pentru că nu se folosește decât un singur circuit conectat pe magistrala externă.



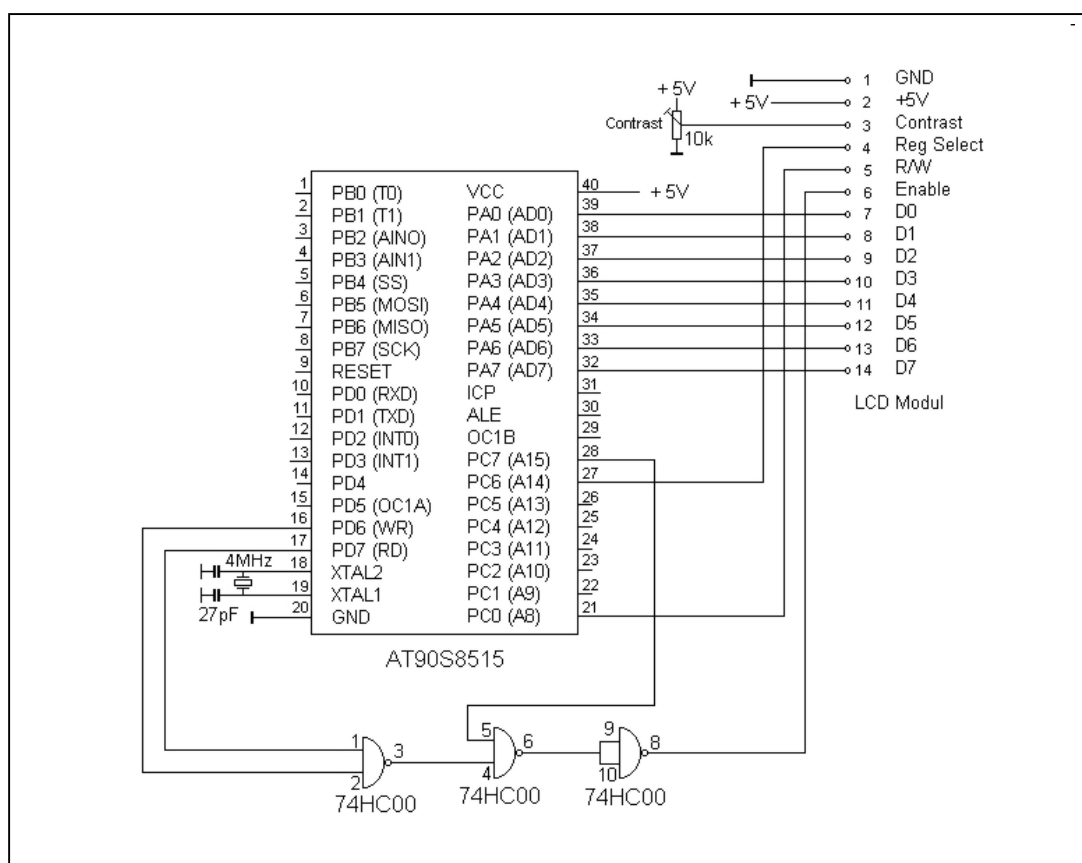


Figura 2.13: Conectarea unui LCD la un MC RISC

## 2.5.Exemplu la microcontrollerele Fujitsu

### 2.5.1. Magistrala externă la MC Fujitsu pe 16 biți F<sup>2</sup>MC-16LX

Magistrala externă permite adresarea pe 16 biți și transferuri pe 16 biți de date. Zona de adresare a magistralei externe este mai mică (22 biți) decât zona de adresare totală care este pe 24 de biți. De asemenea MC poate fi comandat în înaltă impedanță pentru a permite un transfer DMA pe magistrală. Memoria Flash din MC poate fi programată prin intermediul magistralei externe, dar există și alte moduri de programare. Totuși, programarea pe magistrală este cea mai rapidă. Schema bloc este dată în figura 2.14.

Semnalele magistralei externe au următoarea semnificație:

**AD07-AD00** pe portul P07-P00, 8 biți de date multiplexate cu adrese

**AD15-AD08** (16 biți) sau **A15-A08** pe portul P17-P10, 8 biți de date multiplexate cu adrese sau doar 8 biți de adresă

**A21-A16** pe portul P25-P20, 6 biți de adresă. Nu este adresabilă spre exterior toată zona de adresare pe 24 de biți.

**ALE** pe linia P30, Address Latch Enable

**/RD** pe linia P31, Read

**/WR** (8 biți) sau **/WRL** (16 biți) pe linia P32, Write

Port I/O (8 biți) sau **/WRH** (16 biți) pe linia P33, activ doar la magistrala pe 16 biți, Write.  
**HRQ** pe linia P34 cerere către MC de intrare în HOLD.  
**/HAK** pe linia P35 răspuns de confirmare la cererea de intrare în HOLD.  
**RDY** pe linia P36  
**CLK** pe linia P37.

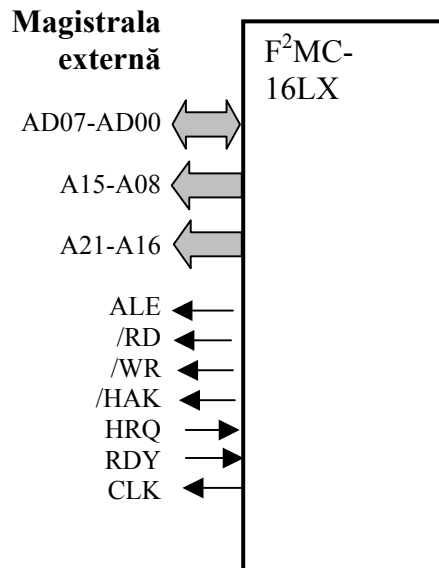


Figura 2.14: Schema bloc a MC pe 16 biți Fujitsu cu magistrală externă

Regiștrii de comandă pentru lucrul cu magistrală externă selectează dacă liniile porturilor au semnificația de linii de magistrală sau linii IO și comandă inserarea a 1, 2 sau 3 stări de WAIT pentru accesul pe magistrală.

### 2.5.2. Magistrală externă la MC Fujitsu pe 32 de biți

În cazul microcontrollerelor cu putere de calcul mai mare, magistrala externă are mai multe linii [4], astfel că lucrul cu magistrala externă este mai complicat, transferurile de date pot fi pe 8, 16 sau 32 de biți. În cazul MC din familia MB91F care sunt MC RISC, schema bloc, prezentată în figura 2.15, arată o structură complexă în care magistralele interne sunt de mai multe tipuri. Între CPU și memorie magistrala este Harvard pe 32 de biți, legătura cu dispozitivele de IO fiind realizată cu o magistrală pe 16 biți, iar în exterior MC este prevăzut cu o magistrală externă Von Neuman.

Caracteristicile magistralei externe sunt:

- Spațiul extern este adresabil cu 32 de biți fiind de 4GB.
- Se pot defini până la 8 bank-uri independente cu semnale de CS generate de MC. Bank-urile pot fi de minimum 64K și pot ocupa orice poziție logică în aria externă.

- Magistrala de date poate fi pe 32/16/8 biți, lățime care poate fi diferită pentru bank-uri diferite.
- Se pot transfera date prin DMA.

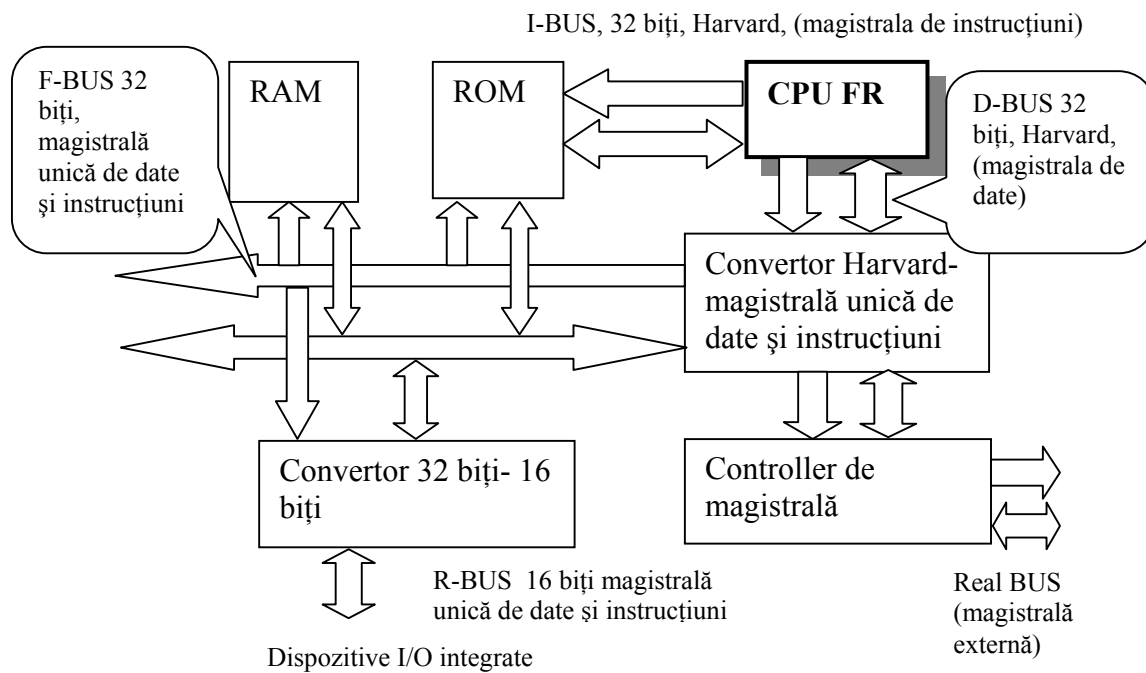


Figura 2.15: Structura internă a MC pe 32 de biți Fujitsu MB91F

O schemă bloc a interfeței cu magistrala externă este dată în figura 2.16.

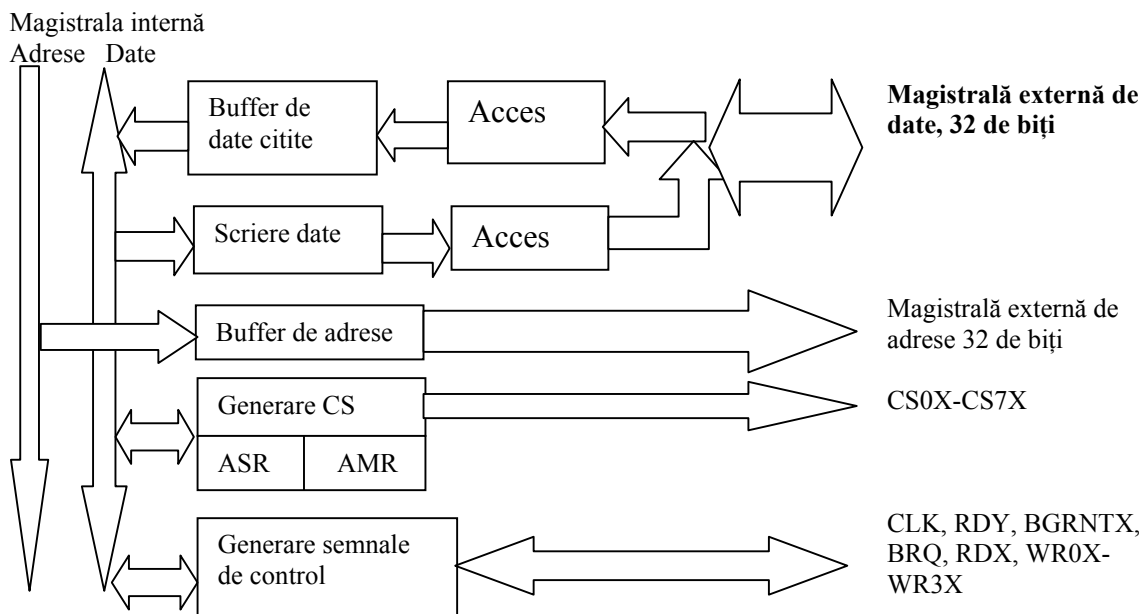


Figura 2.16: Interfața cu magistrala externă

Generarea semnalelor de CS poate fi activată sau invalidată, zonele de adrese selectate fiind programabile prin înscrierea unui registru de programare. Semnalele WR0X, WR1X, WR2X, WR3X sunt semnale de strob de scriere, cu legătură cu lărgimea magistralei de date. Astfel la transferul pe octet se activează WR0X, la un transfer pe 16 biți WR0X și WR1X, iar la transferul pe 32 de biți se activează toate 4. Ordinea în care se transferă datele din regiștrii interni este “Big Endian”, figura 2.17.

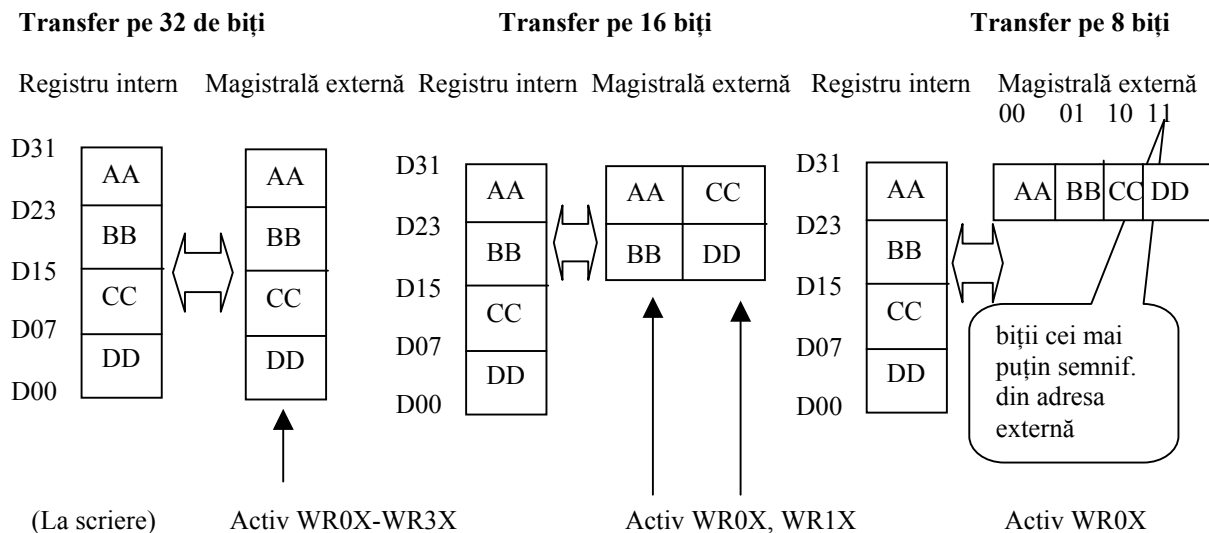


Figura 2.17: Transferul de date prin magistrala externă

**AS-** strob de adrese

**BRQ-** cerere de magistrală

**BGRNTX-** acceptarea cererii de magistrală

**RDX-** strob de citire

**RDY-** semnal de sincronizare (Ready)

**ALE-** strob de memorare a adresei într-un latch extern (Address Latch Enable).

Diagrama de timp pentru un acces de 32 de biți în 2 bank-uri diferite, un ciclu de citire și un ciclu de scriere este dată în figura 2.18. Între ciclurile de scriere / citire se introduc cicluri de sincronizare (a tactelor interne) numite *Idle* al căror număr depinde de tact și de tipul de acces. Pentru sincronizarea externă se pot introduce cicluri de WAIT programând corespunzător un registru de programare sau se poate utiliza pinul extern RDY.

Un exemplu de conectare a unui GDC (Graphic Display Controller) la un microcontroller pe 32 de biți este prezentat pe scurt în continuare. Aplicațiile de afișare și prelucrare de imagini necesită o putere de calcul și o viteză mare.

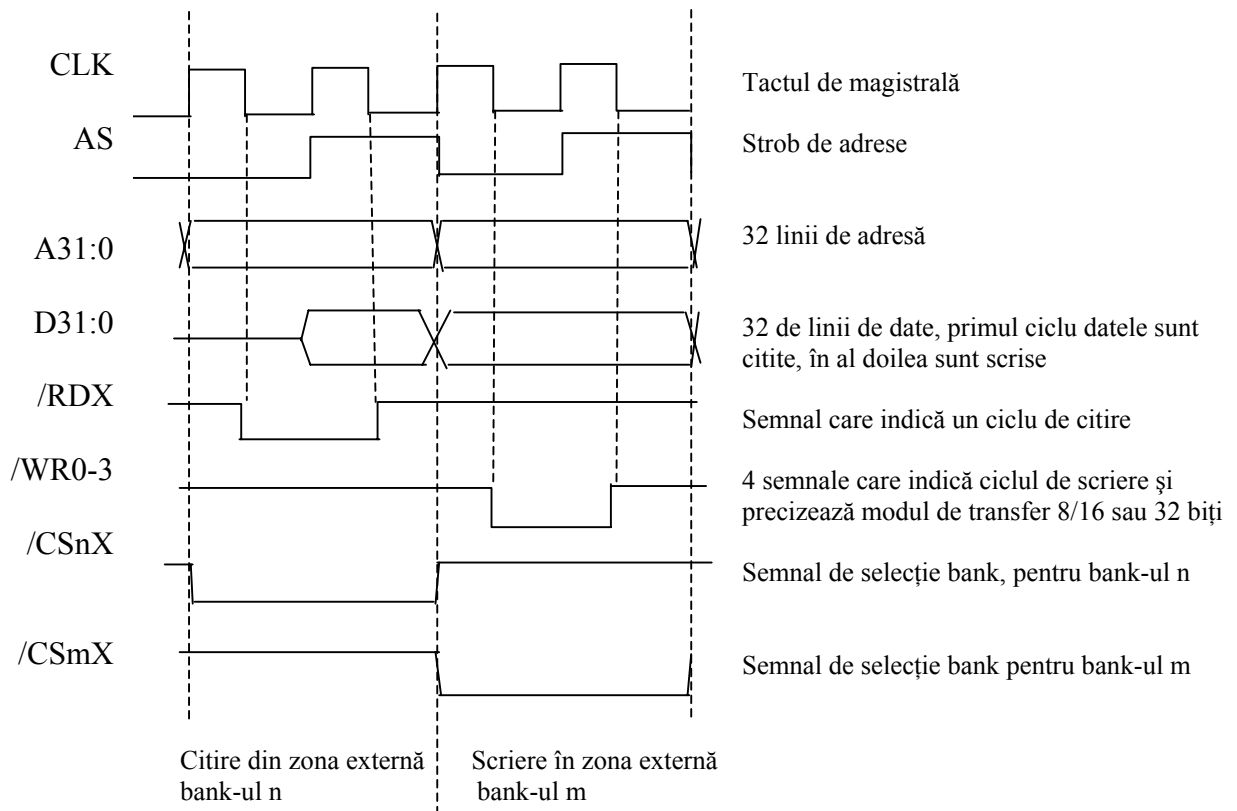


Figura 2.18: Diagrama de timp pentru un transfer pe 32 de biți

În figura 2.19 este prezentată conectarea unui GDC din familia Fujitsu la un MC pe 32 de biți din aceeași familie. Conectarea este pe magistrala externă deoarece este singurul mod de conectare care asigură viteza suficientă de afișare.

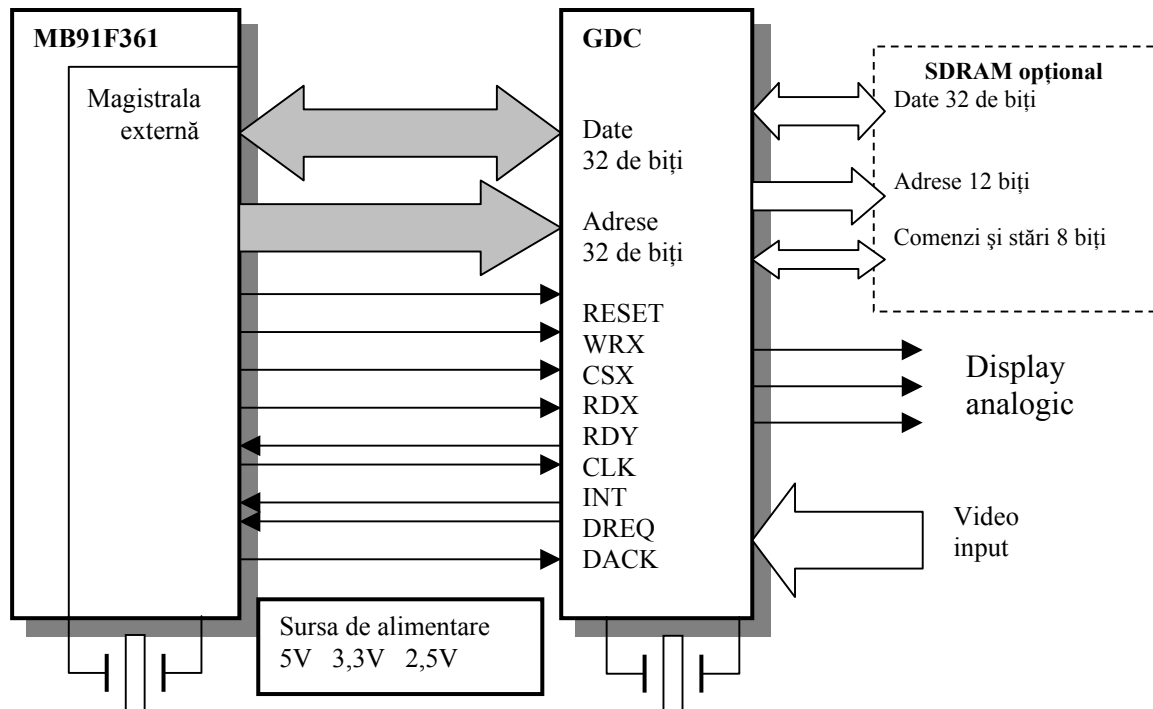


Figura 2.19: Conectarea unui GDC Fujitsu la un MC pe 32 de biți

## 2.6.Scurt istoric al magistralelor standardizate

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 plăchetele 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.

Microcontrollerele au beneficiat la apariție de experiența dezvoltării acestor magistrale și au împrumutat variantele cele mai simple de magistrale sincrone, bazate pe structura ISA.

## Bibliografie

- [1] Gerigan C., Ogruțan P., *Tehnici de interfațare*, Ed. Transilvania Brașov, 2000, 315p., ISBN 973-9474-94-2
- [2] Borza P., Gerigan C., Ogruțan P., Toacșe Gh., *Microcontrollere. Aplicații*, Editura Tehnică București, 2001, ISBN973-31-1577-6, 220p
- [3] Ogruțan P., Gerigan C., Banciu N., *Memorii, interfețe și periferice. Interfețe specializate*, Ed. Transilvania Brașov, 2003, 190 pagini, ISBN 973-635-118-1
- [4] Ogrutan P., *Microcontrollere si controllere grafice Fujitsu*, Ed. Universitatii Transilvania Brasov, 2006, 182 pag, ISBN 973-635-621-3
- [5] Lipnharski W., *8051+LCD+EPR0M*, [online], [www.ustr.net](http://www.ustr.net)
- [6] Fleury P., *Interfacing a HD44780 Based LCD to an AVR*, 2003, [online], <http://jump.to/fleury>