

3.5.3. Magistrala VME

3.5.3.1. Principii

Magistrala VME (*Versa Module Eurocard*) provine din magistrala Versabus a firmei *Motorola*, care a fost utilizată la primele sisteme bazate pe procesorul 68000. Ulterior, magistrala Versabus a fost adaptată pentru formatul plăcilor dublu *Eurocard* (160 x 200 mm), format devenit standard pentru calculatoarele industriale. Noua magistrală, proiectată în colaborare cu firmele *Mostek*, *Signetics/Philips* și *Thompson CSF*, a fost denumită VME, fiind anunțată în anul 1981.

Este o magistrală de 32 biți cu performanțe ridicate, utilizată pe scară largă, mai ales pentru aplicații industriale. Există mii de plăci de extensie bazate pe această magistrală, produse de sute de firme. Are specificații bine definite, care descriu funcționarea magistralei și regulile care trebuie respectate de plăcile VME.

Performanțele ridicate sunt asigurate prin asincronismul magistralei, ceea ce permite diferitelor componente să funcționeze la viteza corespunzătoare tehnologiei utilizate. În practică, limita superioară este de ordinul a 100 ns pentru un ciclu de magistrală, deoarece peste această limită apar nesimetrii (*bus skew*) și alte probleme similare. Cu un transfer de 4 octeți la fiecare 100 ns, rezultă o rată de transfer de 40 MB/s.

Fiabilitatea magistralei este asigurată prin proiectarea mecanică și protocolul logic. Spre deosebire de conectorii calculatoarelor IBM PC, se utilizează conectori formați din pini metalici. Deși soluția este mai scumpă, se elimină astfel conexiunile necorespunzătoare, care reprezintă una din principalele surse de probleme la sistemele de calcul. Există linii ale magistralei care se pot utiliza pentru autotest și raportarea stării.

Magistrala VME face parte dintr-o familie de trei magistrale proiectate pentru o gamă largă de sisteme de calcul, de la mici sisteme de dezvoltare la sisteme multiprocesor. De exemplu, Figura 3.17(a) prezintă un *sistem minimal*, format din trei plăci VME, câte una pentru UCP, memorie și un controler de I/E.

Figura 3.17(b) prezintă un *sistem multiprocesor* [18]. Fiecare procesor are o memorie locală la care se conectează printr-o magistrală VSB. Dacă se păstrează programul și datele locale în memoriile locale, magistrala VME va fi utilizată numai de instrucțiunile care fac acces la memoria globală partajată. Astfel, rata de transfer globală poate depăși limita de 40 MB/s impusă de tehnologia VME. De exemplu, un sistem multiprocesor cu 16 UCP, fiecare UCP solicitând un cuvânt de 4 octeți la fiecare 200 ns, necesită o rată de transfer de:

$$(4 \cdot 1/200 \cdot 10^9) \cdot 16 \text{ B/s} = 320 \text{ MB/s}$$

Dacă 90% din aceste accesuri se fac la memoria locală (pentru încărcarea instrucțiunilor, citirea/scrierea datelor locale), cu o magistrală VME și 16 magistrale VSB se poate asigura această rată.

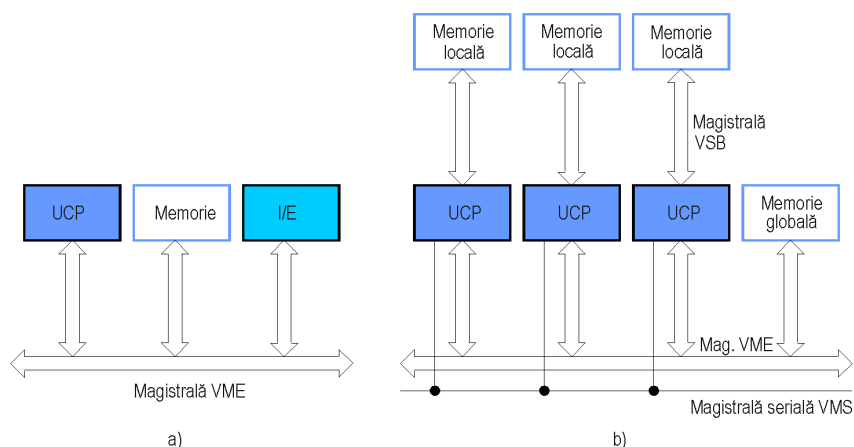


Figura 3.17. Sisteme cu magistrală VME: a) Sistem minimal; b) Sistem multiprocesor.

Magistrala serială VMS, care funcționează independent de celelalte două magistrale, poate fi utilizată pentru comunicare și sincronizare între procesoare, în paralel cu transferurile de date pe magistrala principală.

3.5.3.2. Structura funcțională

Această structură constă din logica de interfață, patru grupuri de linii numite magistrale, și o colecție de module funcționale asociate cu acestea.

Magistrala pentru transferul de date

Constă din 32 linii de date, 32 linii de adrese, 6 linii de modificare a adresei, care definesc tipul ciclului de magistrală, și 5 linii de control. Modulele funcționale ale acestei magistrale cuprind modulele *master* și *slave*, un modul *monitor* și un modul de tip *ceas*.

Modulul master inițiază cicluri de transfer pentru a transfera datele cu modulul *slave*. *Modulul slave* detectează ciclurile de transfer inițiate de un master, și dacă acele cicluri specifică participarea sa, transferă datele la sau de la modulul master.

Monitorul urmărește transferurile de date, în scopul detectării accesurilor la locațiile care i-au fost asignate pentru a fi urmărite. În cazul în

care apare un acces la o asemenea locație, monitorul poate genera o întrerupere. Dacă, de exemplu, procesorul plăcii *A* scrie într-o locație a memoriei globale care este urmărită de monitorul plăcii *B*, procesorul plăcii *B* va fi întrerupt.

Ceasul măsoară durata fiecărui transfer, și termină transferul dacă acesta durează un timp prea mare. Este util în cazul în care se adresează o locație inexistentă sau un modul slave inexistent. Se evită astfel blocajul.

Magistrala pentru transferul de date dispune de linii nemultiplexate pentru adrese și date. Aceasta asigură creșterea performanțelor la executarea ciclurilor de transfer singular, când se transmite o adresă, după care se citește sau se scrie data.

Magistrala pentru întreruperi

Constă din 7 linii de cerere de întrerupere, $\overline{IRQ1}.. \overline{IRQ7}$, o linie de achitare a întreruperilor, \overline{IACK} , și o conexiune de tip *daisy chain* pentru achitarea întreruperilor. Deoarece pot exista mai multe UCP, liniile de cerere pot fi partiționate între mai multe controlere de întreruperi, fiecare controler gestionând anumite linii și generând semnalele de întrerupere pentru anumite UCP.

Sursa de întrerupere generează o cerere de întrerupere pe una din liniile de cerere. Atunci când cererea este achitată de un modul de gestiune a întreruperii, sursa de întrerupere furnizează 1, 2 sau 4 octeți de stare sau identificare în cadrul unui ciclu special de magistrală, octeți care permit servirea întreruperii.

Conexiunea de tip *daisy chain* pentru achitarea întreruperilor va fi activată atunci când un modul de gestiune a întreruperii achită o cerere de întrerupere. Această conexiune asigură ca un singur modul să răspundă cu octeții de stare sau identificare atunci când au fost generate mai multe cereri.

Magistrala de arbitraj

În cadrul procesului de arbitraj este selectat modulul master sau modulul de gestiune a întreruperilor care va putea utiliza magistrala pentru transferul de date. Magistrala de arbitraj constă din 4 linii de cerere a magistralei, $\overline{BR0}.. \overline{BR3}$, 4 linii de acordare, $\overline{BG0IN}.. \overline{BG3IN}$, și alte două linii numite *Bus Busy* (\overline{BBSY}) și *Bus Clear* (\overline{BCLR}).

Există un modul de cerere a magistralei, aflat pe aceeași placă cu un modul master sau modul de gestiune a întreruperii. După efectuarea cererii, se așteaptă achitarea acesteia de către arbitru, după care modulul

de cerere preia controlul asupra magistralei prin activarea liniei *Bus Busy*, dezactivează cererea și indică modulului master faptul că magistrala este disponibilă.

Arbitrul poate avea o facilitate de *time-out*, care permite retragerea utilizării magistralei dacă modulul solicitant nu începe utilizarea magistralei în timpul prestabilit. Aceasta asigură că magistrala nu va fi blocată ca rezultat al unui semnal tranzitoriu pe linia de cerere. Arbitrul poate activa linia *Bus Clear* dacă detectează o cerere de prioritate mai ridicată față de cea a modulului care utilizează magistrala în acel moment, acordând magistrala modulului mai prioritar.

Magistrala utilitară

Conține două linii pentru semnale de ceas, o linie de resetare a sistemului, o linie de date pentru magistrala serială, o linie de detectare a căderii tensiunii de alimentare și o linie pentru indicarea a unei erori de sistem.

Modulul de ceas al sistemului furnizează un semnal de ceas *SYSCLK* de 16 MHz, util pentru diferite funcții care necesită o referință de timp. Acest semnal nu se utilizează pentru delimitarea ciclurilor de magistrală.

Pentru magistrala serială VMS există un semnal de ceas *SERCLK* care asigură sincronizarea operațiilor pe această magistrală. Viteza de transfer pe magistrala serială este de 32 Mbiți/s, dar se poate utiliza și o viteză de 16 sau 8 Mbiți/s. Protocolul acestei magistrale este complet independent de magistrala VME.

Starea sursei de alimentare este urmărită de un modul monitor, iar atunci când se detectează o tensiune în afara limitelor admise, monitorul utilizează linia prevăzută în acest scop (*ACFAIL*) pentru a avertiza toate plăcile conectate la magistrală în vederea pregătirii căderii tensiunii (“*graceful shutdown*”).

La detectarea unei erori de către o placă conectată la magistrala VME, aceasta poate utiliza o linie specială (*SYSFAIL*) pentru a indica celorlalte module apariția acestei erori.

3.5.3.3. Cicluri de magistrală

Specificațiile VME definesc cinci tipuri de cicluri de transfer și două cicluri suplimentare. Ciclurile de transfer se pot utiliza pentru transferul datelor de 8, 16 sau 32 de biți.

Ciclurile de citire și scriere

Încep prin transmiterea unei adrese și a unui cod de modificare a adresei, care indică tipul ciclului de magistrală. Sunt permise și transferuri nealiniate, de exemplu 4 octeți începând de la o adresă impară.

Ciclurile de transfer pe blocuri

Se utilizează pentru citirea sau scrierea unui bloc de maxim 256 octeți. Acestea permit unui modul *master* să adreseze o singură locație și apoi să acceseze acea locație și locațiile următoare, fără a transmite informații suplimentare de adresare. La inițierea unui asemenea ciclu de către un *master*, modulul *slave* memorează adresa într-un numărător de adrese, ea fiind incrementată ulterior și utilizată la transferurile ulterioare.

Lungimea blocurilor transferate este limitată la 256 de octeți. Motivul este simplificarea proiectării plăcilor de memorie și reducerea timpului de acces la modulele de memorie. Dacă nu ar exista această limitare, toate modulele de memorie ar trebui să memoreze adresa de început a blocului, și apoi să incrementeze această adresă în fiecare ciclu. Adresa incrementată ar trebui apoi decodificată de fiecare modul, pentru a testa dacă adresa a ajuns în propria zonă de adrese. Această decodificare ar crește timpul de acces.

Prin limitarea introdusă, dacă un modul de memorie are o dimensiune de cel puțin 256 octeți, iar prima adresă se află în propria zonă de memorie, următoarele se vor afla de asemenea în această zonă. Deoarece numai cei 7 biți c.m.p.s. ai adresei se vor modifica în timpul unui transfer, liniile superioare de adresă pot fi decodificate o singură dată, la începutul ciclului.

Ciclurile de citire-modificare-scriere

Asigură o operație indivizibilă de citire și scriere a unei locații, fără a permite unui alt *master* accesul la acea locație înainte de terminarea operației. Aceste cicluri sunt utile în sistemele multiprocesor, unde trebuie partajate anumite resurse, unele locații de memorie fiind utilizate pentru *funcții semafor*. Trebuie să se asigure ca o resursă utilizată de un anumit proces să nu fie utilizată de un alt proces în același timp. Un asemenea ciclu este similar cu un ciclu de citire urmat de unul de scriere, dar nu se permite acordarea magistralei de date unui alt modul între cele două cicluri.

Ciclurile fără transferuri de date

Constau doar din transmiterea unei adrese. Rolul acestui ciclu este de a permite unui modul *master* să anunțe că va solicita în scurt timp conținutul memoriei de la o anumită adresă. La o cerere ulterioară, memoria va putea răspunde fără întârziere, evitându-se stările de așteptare. Se poate suprapune astfel un ciclu cu cel următor.

Ciclurile de achitare a unei întreruperi

Sunt inițiate de modulele de gestiune a întreruperilor în scopul citirii informațiilor de stare sau identificare de la un modul care a solicitat o întrerupere.

În Figura 3.18 se prezintă un ciclu de citire de 32 de biți [18].

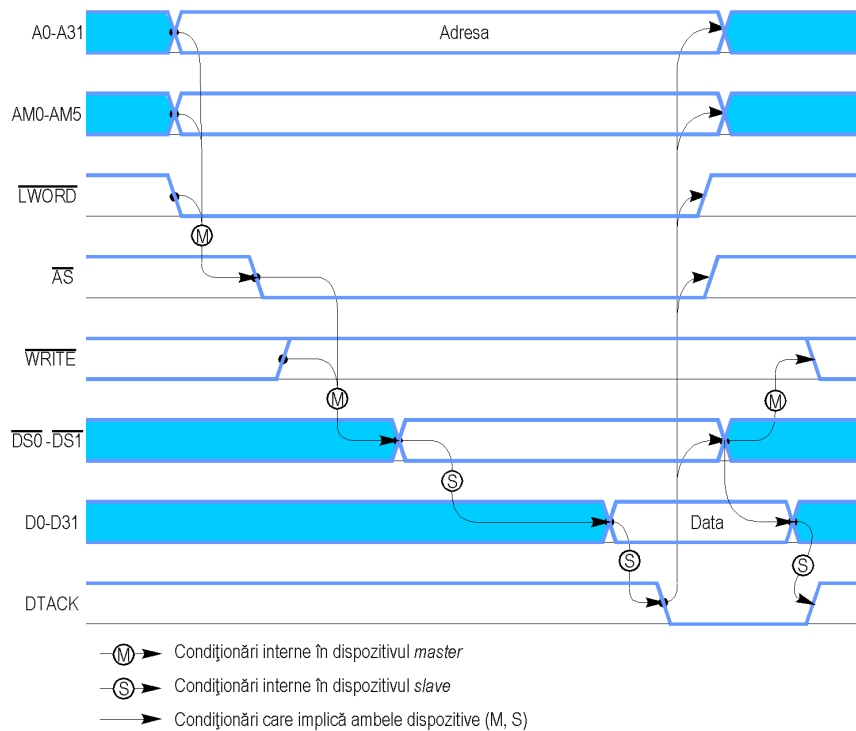


Figura 3.18. Ciclu de citire la magistrala VME.

La începutul unui transfer, dispozitivul *master* depune adresa pe liniile de adrese $A0-A31$ și activează semnalele de modificare a adreselor $AM0-AM5$, care definesc tipul ciclului de magistrală. Dacă se efectuează un transfer complet de 32 de biți, se activează de asemenea semnalul $LWORD$. Acest semnal definește, împreună cu tipul ciclului, dimensiunea datelor transferate.

După stabilizarea semnalelor $A0-A31$, $AM0-AM5$ și $LWORD$, dispozitivul *master* activează semnalul AS (*Address Strobe*) pentru a indica dispozitivului *slave* faptul că adresa este validă și poate fi memorată. Semnalul $WRITE$ va fi dezactivat dacă se efectuează o citire, și va fi activat dacă se efectuează o scriere. În final, dispozitivul *master* activează semnalele $DS0-DS1$ (*Data Strobe*) pentru a indica dispozitivului *slave* că așteaptă datele.

Pe baza semnalelor $DS0$, $DS1$, $LWORD$ și $A1$ se determină octetul sau octeții necesari din cadrul cuvântului selectat. Dispozitivul *slave* sesizează activarea semnalelor $DS0-DS1$ și depune datele pe liniile $D0-D31$. Apoi activează semnalul $DTACK$ (*Data Acknowledge*) pentru a semnala depunerea datelor.

Dispozitivul *slave* sesizează activarea semnalului $DTACK$, preia datele și dezactivează semnalele care au fost activate la începutul ciclului. Dezactivarea semnalelor $DS0-DS1$ determină ca dispozitivul *slave* să dezactiveze semnalul $DTACK$, terminându-se ciclul.

3.5.3.4. Arbitrajul de magistrală

Se utilizează o combinație între arbitrarea centralizată și cea distribuită. Există un arbitru de magistrală amplasat pe prima placă, cu rolul de a acorda magistrala pentru o cerere de un anumit nivel de prioritate. În cadrul nivelului respectiv, un modul va fi selectat prin utilizarea unei conexiuni în lanț între module.

Dintre algoritmi de arbitraj posibili, specificațiile VME descriu trei:

- arbitraj pe bază de *priorități*;
- arbitraj “*round robin*”;
- arbitraj cu un *singur nivel*.

Arbitrajul pe bază de priorități

Atribuie magistrala pe baza unei scheme de priorități fixe, fiecărei linii de cerere fiindu-i asignată o prioritate.

Arbitrajul “round robin”

Atribuire magistrala pe baza unor priorități rotitoare. Fiecare din cele 4 linii are aceeași prioritate. Dacă magistrala este atribuită modulului care a activat cererea de nivel n , prioritatea cea mai mare va fi asignată liniei de cerere $n-1$, și cea mai mică liniei n .

Arbitrajul cu un singur nivel

În cazul arbitrajului *cu un singur nivel*, sunt acceptate numai cereri pe linia de nivel 3. Conexiunea de tip *daisy chain* a acestui nivel va asigura ca un singur modul să utilizeze magistrala.

Deși nu sunt descriși de specificațiile VME, se pot utiliza și alți algoritmi. De exemplu, un algoritm poate asigna prioritatea maximă cererii de nivel 3, pentru celelalte cereri atribuirea realizându-se în modul “*round robin*”.

Eliberarea magistralei se poate realiza în două moduri:

- Eliberare la terminare (RWD - *Release When Done*);
- Eliberarea la cerere (ROR - *Release On Request*).

În primul caz, modulul care a solicitat magistrala semnalează arbitralului terminarea utilizării magistralei imediat după ce modulul său *master* a indicat că nu mai necesită magistrala. În cazul eliberării la cerere, după terminarea utilizării magistralei arbitralul va fi anunțat numai dacă un alt modul a efectuat o cerere de magistrală. Pentru aceasta, modulul trebuie să monitorizeze toate liniile de cerere. Această metodă permite implementarea procedurii de parcare a magistralei.

3.5.4. Magistrala Micro Channel

În anul 1987, firma IBM a lansat pe piață calculatoarele din familia PS/2. Acestea dispuneau de unele îmbunătățiri, ca: un controler pentru adaptorul video VGA integrat pe placa de bază, adaptorul de disc flexibil integrat pe placa de bază, noi porturi pentru tastatură și mouse, un adaptor și o interfață mai performante pentru discul fix.

Caracteristica cea mai importantă a acestor sisteme a fost utilizarea unei noi magistrale de extensie, magistrala *Micro Channel* (MCA - *Micro Channel Architecture*). Modelele 50Z, 55SX și 60 ale calculatoarelor PS/2 utilizează o versiune de 16 biți a acestei magistrale, iar modelele 70, 80 utilizează o versiune de 32 de biți pentru adrese și date.