

10. STANDARDELE “PLUG AND PLAY”

10.1. Scopul lucrării

Lucrarea prezintă standardul PnP pentru adaptoarele de extensie conectate la magistrala ISA și specificațiile PnP pentru interfața SCSI. Sunt descrise dificultățile implementării standardelor PnP la calculatoarele IBM PC, procesul de configurare a sistemului și a plăcilor de extensie, protocolul pentru izolarea unei plăci, stările PnP, registrele PnP, citirea datelor de configurare și configurarea resurselor.

10.2. Considerații teoretice

Modernizarea unui calculator IBM PC sau instalarea unor echipamente noi poate constitui o problemă serioasă, mai ales pentru utilizatorii nespecialiști, dar nu numai pentru aceștia. De multe ori, o modernizare necesită nu numai configurarea echipamentului nou instalat, ci și reconfigurarea unuia sau a mai multor echipamente deja existente în sistem. Pe lângă instalarea hardware, este necesară de cele mai multe ori și o procedură de instalare a unor drivere și setarea unor parametri ai perifericului. O instalare necorespunzătoare poate conduce la o funcționare incorectă a sistemului.

O altă problemă a pieței de calculatoare compatibile *IBM* o reprezintă concurența cu calculatoarele *Apple Macintosh* și *Power PC*, care sunt mai ușor de instalat și de modernizat decât calculatoarele PC [8]. Dificultățile legate de instalare și modernizare pot avea un efect negativ asupra industriei de profil, mai ales în segmentul de piață destinat publicului larg. Din acest motiv, producătorii au ajuns la un consens asupra necesității de a simplifica procedurile de instalare și modernizare.

În anul 1994, la simpozionul “*Windows Hardware Conference*”, firmele *Microsoft* și *Intel* au propus un set de standarde noi pentru a rezolva această problemă. Aceste standarde au fost numite standardele “*Plug and Play*” (Conectare și Funcționare), fiind utilizată abrevierea PnP. Scopul acestor standarde este de a permite ca echipamentele noi să fie doar conectate la sistem pentru a funcționa corect, fără a fi necesară

reconfigurarea calculatorului cu ajutorul unor comutatoare și a unor instalări complexe ale unor drivere sau programe utilitare. Aceasta nu înseamnă că sistemul nu trebuie reconfigurat, ci doar că acest proces se realizează automat de către sistemul de operare.

10.2.1. Caracteristici PnP existente la alte sisteme

Ideea că modernizarea unui calculator nu ar trebui să necesite configurarea unor comutatoare nu este nouă [8]. De exemplu, la calculatorul original *Apple II* au fost rezervate blocuri de memorie pentru fiecare conector de extensie, astfel încât informațiile legate de configurație puteau fi memorate pe fiecare placă adaptoare. Astfel, se putea determina echipamentul care era instalat în fiecare conector de extensie și, teoretic, se putea configura automat fiecare adaptor pentru a se evita conflictele la resursele sistemului. Cele mai multe din primele adaptoare utilizate de *Apple II* nu utilizau însă această facilitate, sau o utilizau doar pentru a transmite sistemului de operare informații despre configurația curentă. Doar un număr redus de adaptoare permiteau configurarea prin software. Existau însă conceptele fundamentale și caracteristicile arhitecturii necesare pentru acceptarea viitoarelor standarde PnP.

Calculatoarele *Amiga* ale firmei *Commodore* dispuneau de o caracteristică de PnP, numită *AutoConfig*, care elimina în totalitate comutatoarele de pe adaptoarele utilizate de aceste calculatoare. Specificațiile *AutoConfig* permiteau sistemului de operare să interogheze adaptoarele și să le reconfigureze dacă era necesar, pentru a elimina conflictele sau pentru a asocia niveluri de prioritate care să fie în concordanță cu necesitățile adaptorului. Sistemele *Amiga* au fost probabil primele sisteme care dispuneau de arhitectura hardware și de sistemul de operare necesar pentru standardul PnP.

Magistrala *Micro Channel* (MCA) cu care erau dotate calculatoarele IBM din seria PS/2 dispune de facilitatea *Programmable Option Select* (POS), care elimină necesitatea comutatoarelor și a jumperelor de pe plăcile adaptoare și placa de bază. Prin protocolul specificat era posibilă identificarea și configurarea adaptoarelor aflate în conectorii de extensie. Fiecărui producător de adaptoare îi era alocat un număr de identificare și i se cerea implementarea unui nivel minim de posibilități de configurare. Arhitectura MCA nu a avut însă succes pe scară largă.

Magistralele VL Bus și PCI au rezervat un spațiu de configurare pentru fiecare echipament instalat în conectorii de extensie. Aceste magistrale permit implementarea standardelor PnP. Cu toate acestea, chiar dacă aceste magistrale de extensie implementează o specificație com-

pletă PnP, sistemele utilizează de obicei atât conectori VL Bus/PCI, cât și conectori ISA.

10.2.2. Dificultățile implementării PnP la calculatoarele IBM PC

10.2.2.1. Problemele principale ale implementării PnP

Pentru o funcționare corectă, fiecare adaptor și driverul asociat necesită anumite resurse ale calculatorului. Aceste resurse cuprind adrese ale porturilor de I/E, nivele ale cererilor de întrerupere, canale DMA, precum și adrese ale unor zone de memorie utilizate. Problemele principale cu care se confruntă un calculator PC sunt următoarele [8]:

1. Există un număr finit de resurse ale sistemului care trebuie partajate.
2. Nu există o modalitate standard pentru determinarea resurselor care sunt utilizate de sistem sau a celor disponibile.
3. Nu există o modalitate standard pentru determinarea configurației unui adaptor sau a unui periferic, și nici pentru reconfigurarea utilizării resurselor de către adaptor, astfel încât să se elimine conflictele.

În plus, funcționarea unor adaptoare sau aplicații poate fi incorrectă chiar dacă ele sunt reconfigurabile. De exemplu, pentru o funcționare corectă unele adaptoare necesită anumite priorități pentru nivelul de întrerupere sau canalul DMA. Simpla atribuire a unui nivel de întrerupere sau canal DMA neutilizat până atunci nu asigură neapărat o funcționare corectă. Aplicațiile multimedia care necesită performanțe ridicate pot fi sensibile la configurarea parametrilor. Multe din aceste aplicații necesită o anumită rată a datelor, care poate depinde de prioritatea nivelului de întrerupere și a canalului DMA.

Problema resurselor finite este comună pentru toate calculatoarele. Costurile și alte restricții nu permit existența unor resurse nelimitate. Chiar și în cazul în care resursele ar fi nelimitate, ar putea apărea probleme atunci când adaptoarele ar alege aceleași resurse. Problema principală nu constă în general în lipsa unor resurse suficiente, ci în alocarea resurselor existente astfel încât să se evite conflictele în utilizarea lor.

Pentru implementarea standardelor PnP, trebuie ca sistemul să poată identifica tipul echipamentelor atașate la fiecare conector de extensie sau interfață, și să aibă acces la informațiile de configurare din

fiecare echipament. Dacă aceste condiții sunt îndeplinite, sistemul de operare sau programul de aplicație poate detecta cel puțin problemele potențiale legate de conflictele la resurse. În cazul magistralei ISA, aceasta nu permite accesul la informațiile de configurare ale unui anumit conector de extensie. Deoarece nu există o modalitate simplă de a identifica tipul adaptorului, poziția acestuia și necesarul lui de resurse, controlul alocării resurselor este lăsat la latitudinea utilizatorului. Acest control se realizează manual, prin modificarea poziției comutatoarelor de configurare, astfel încât să se evite conflictele la resursele conectate la magistrala ISA.

Pentru a beneficia de toate avantajele standardelor PnP, trebuie să coopereze toate componentele de pe diferitele nivele ale sistemului. Magistralele și interfețele trebuie să fie proiectate astfel încât să permită identificarea echipamentelor și să aibă acces la informațiile de configurație ale acestora. Adaptoarele și echipamentele trebuie să fie proiectate astfel încât să pună la dispoziție aceste informații. Aceste adaptoare și echipamente trebuie să fie configurabile prin program, pentru a permite rezolvarea eventualelor conflicte la resurse. Componenta BIOS a sistemului trebuie să ofere servicii PnP. Sistemul de operare trebuie să aibă posibilitatea alocării resurselor, să ofere aplicațiilor informații referitoare la resursele alocate și să raporteze conflictele care nu pot fi rezolvate. Driverele pentru echipamente și programele de aplicație trebuie să aibă acces la informațiile de configurare și să se adapteze în mod dinamic la configurația existentă. Chiar dacă nu se pot îndeplini toate aceste condiții și standardele PnP se pot implementa doar parțial, se pot obține avantaje semnificative.

În general, majoritatea interfețelor nu necesită modificări hardware pentru a putea implementa standardele PnP. În schimb, echipamentele conectate la sistem pot necesita unele modificări, pentru a permite accesul la informația de configurare și pentru a accepta reconfigurarea prin program. Este de preferat ca informațiile de configurare să fie obținute direct de la fiecare echipament. Obținerea acestor informații dintr-un fișier de pe disc sau dintr-o memorie nevolatilă care nu aparține echipamentului respectiv prezintă riscul unor date eronate, care nu corespund configurației reale a echipamentului. De exemplu, informațiile despre tipul și parametrii discului fix care se pot obține din memoria CMOS inclusă în circuitul de ceas al sistemului pot fi greșite.

Standardele PnP au în vedere toate tipurile de magistrale de extensie și interfețele din cadrul sistemului. Acestea cuprind magistralele ISA, EISA, VL Bus, PCI, interfețele IDE și SCSI, porturile seriale și paralele, ca și adaptorul video. Problema cea mai dificilă este de a pune la dispoziție facilități PnP pentru magistrala ISA. Chiar dacă sistemul are la

bază magistrala VL Bus sau PCI, există și conectori ISA, la care sunt conectate adaptoare LAN, plăci de fax/modem și plăci de sunet. Toate adaptoarele conectate la magistrala ISA trebuie să utilizeze resursele comune astfel încât să nu apară conflicte.

10.2.2.2. Conflictele generate de adresele de I/E pe magistrala ISA

Procesoarele *Intel* 80x86 permit adresarea a 65.536 porturi de I/E. Pentru selecția porturilor, aceste procesoare utilizează 16 linii de adresă. Calculatoarele originale PC și PC/XT utilizau doar 10 linii de adresă (A9..A0) pentru adresarea porturilor. Chiar dacă magistrala de adrese era de 16 biți, aceste calculatoare nu permiteau decodificarea decât a 10 biți, obținându-se astfel 1.024 de adrese care se repetau la fiecare bloc de 1 KB din spațiul de I/E. Cele 1.024 de adrese de I/E erau împărțite în două blocuri de câte 512 adrese. Adresele între 0 și 1FFh erau rezervate pentru circuitele de pe placa de bază, iar cele cuprinse între 200h și 3FFh erau disponibile pentru adaptoarele sau echipamentele conectate la magistrala de extensie.

Calculatoarele PC/AT au modificat partiționarea adreselor de I/E, astfel încât adresele cuprinse între 0 și FFh erau rezervate pentru placa de bază, domeniul destinat magistralei de extensie fiind astfel suplimentat cu încă 256 de adrese. Deoarece majoritatea adaptoarelor erau proiectate pentru a putea funcționa atât pe magistrala PC/XT, cât și pe magistrala ISA, puține dintre ele au beneficiat de avantajele oferite de extinderea domeniului de adresare.

Pentru alocarea adreselor porturilor de I/E, sunt respectate de obicei anumite standarde de facto. Acest lucru este valabil mai ales în cazul circuitelor de pe placa de bază care sunt conectate la magistrala ISA (ca de exemplu portul IDE), ca și în cazul adaptoarelor de extensie obișnuite (cum sunt adaptoarele porturilor seriale și paralele, adaptoarele video sau plăcile de sunet). Tabelul 10.1 prezintă utilizarea obișnuită a adreselor de I/E de către adaptoarele instalate în calculatoarele IBM PC.

Tabelul 10.1. *Utilizarea adreselor de I/E la calculatoarele IBM PC.*

Adresă hexa	Echipament
000 - 00F	Controler DMA 1 - 8237A-5
010 - 01F	Controler DMA - 8237A-5 (PS/2 Model 60, 80)
020 - 02F	Controler de întreruperi 1, master - 8259A
040 - 05F	Timer - 8254

Adresă hexa	Echipament
060 - 06F	Controler de tastatură - 8042
070	RAM CMOS / Ceas de timp real, validare / invalidare NMI
071	RAM CMOS
080 - 08F	Registre de pagini DMA - 74LS612
090 - 097	Programmable Option Select (PS/2)
0A0 - 0BF	Controler de întreruperi 2 - 8259A
0C0 - 0DF	Controler DMA 2 - 8237A
0F0 - 0FF	Coprocessor matematic
100 - 10F	Programmable Option Select (PS/2)
170 - 177	Controler disc fix 2 (IDE 2)
1F0 - 1F7	Controler disc fix 1 (IDE 1)
200 - 20F	Adaptor pentru jocuri
278 - 27F	Port paralel LPT2
2E2 - 2E3	Adaptor pentru achiziții de date
2E8 - 2EF	Port serial COM4
2F8 - 2FF	Port serial COM2
300 - 31F	Placă pentru experimentare prototipuri
360 - 36F	Placă de rețea
370 - 377	Controler disc flexibil
378 - 37F	Port paralel LPT1
380 - 38F	Adaptor SDLC 2
3A0 - 3AF	Adaptor SDLC 1
3B0 - 3BF	Adaptor de ecran monocrom și imprimantă
3C0 - 3CF	EGA / VGA
3D0 - 3DF	Adaptor grafic color
3E8 - 3EF	Port serial COM3
3F0 - 3F7	Controler disc flexibil
3F8 - 3FF	Port serial COM1

Problema principală este evitarea selectării unor blocuri de adrese de I/E care sunt utilizate deja de un alt adaptor. De obicei adaptoarele sunt setate implicit pentru a utiliza anumite adrese de I/E, pe baza presupunerii că nu vor exista conflicte. Dacă apare un conflict, se poate alege un alt bloc de adrese dintre cele puse la dispoziție de adaptorul respectiv. În cazul în care conflictul persistă, trebuie să se determine cauza acestuia, prin examinarea tuturor adaptoarelor instalate, modificând

selecția adreselor astfel încât conflictul să fie eliminat. Poate fi necesară de asemenea reconfigurarea driverului pentru a se adapta la noul bloc de adrese selectat.

10.2.2.3. Conflictele generate de cererile de întrerupere pe magistrala ISA

O altă problemă care apare la magistrala ISA o constituie modul în care se alocă nivelurile cererilor de întrerupere. Arhitectura originală PC utiliza un singur controler de întrerupere de tip 8259A, având opt intrări pentru cererile de întrerupere.

Tabelul 10.2. Utilizarea cererilor de întrerupere la calculatoarele IBM PC.

Nivel		Funcție
8259A-1	8259A-2	
IRQ0 ¹⁾		Timer
IRQ1 ¹⁾		Tastatură
IRQ2 ¹⁾		Intrare de la 8259A-2
	IRQ8 ¹⁾	Ceas de timp real
	IRQ9	Redirectat prin program către IRQ2
	IRQ10	IRQ de la un conector de extensie ²⁾
	IRQ11	IRQ de la un conector de extensie ²⁾
	IRQ12	IRQ de la un conector de extensie ²⁾
	IRQ13 ¹⁾	IRQ de la coprocesorul matematic
	IRQ14	IRQ de la adaptorul de disc fix
	IRQ15	IRQ de la un conector de extensie ²⁾
IRQ3		Port serial COM2 sau COM4
IRQ4		Port serial COM1 sau COM3
IRQ5		Port paralel LPT2
IRQ6		Controler disc flexibil
IRQ7		Port paralel LPT1
Observații: ¹⁾ Întreruperi utilizate pe placa de bază. ²⁾ Intrări la care sunt direcționate cererile de întrerupere VL Bus sau PCI.		

Arhitectura PC/AT conține două controlere de întrerupere 8259A, ca și calculatoarele actuale. Există un număr de 15 intrări pentru cererile de întrerupere, și o cerere de întrerupere nemascabilă (NMI). Conectorii

magistralei VL Bus și cei ai magistralei PCI au de asemenea intrări pentru cereri de întrerupere. De obicei, aceste cereri de întrerupere sunt direcționate către una sau mai multe intrări de întrerupere ale magistralei ISA. Tabelul 10.2 prezintă utilizarea tipică a cererilor de întrerupere la calculatoarele IBM PC.

Ca și în cazul adresării porturilor de I/E, nu există o modalitate simplă de a determina care sunt intrările utilizate pentru cererile de întrerupere, deși multe nivele de întrerupere au o utilizare standard. Alocarea intrărilor cererilor de întrerupere este controlată manual, prin jumperele și comutatoarele de pe plăcile adaptoare. Alegerea unei anumite intrări implică o problemă suplimentară, care nu apare la alegerea unei adrese de I/E. Cererile de întrerupere au asociate priorități, iar unele aplicații au nevoie de un anumit nivel de prioritate pentru a funcționa corect.

Spre deosebire de numărul adreselor de I/E, care nu reprezintă de obicei o problemă, numărul de intrări pentru cererile de întrerupere este adesea insuficient pentru un sistem cu un număr mare de adaptoare de extensie. Realocarea dinamică a intrărilor cererilor de întrerupere este dificilă, datorită faptului că pe magistrala ISA cererile de întrerupere sunt sesizate pe frontul semnalelor. Toate magistralele noi utilizează sesizarea pe nivel, permițând astfel mai multor echipamente să se conecteze simultan la aceeași intrare de cerere. În cazul sesizării pe nivel, o cerere de întrerupere este considerată activă și după ce semnalul își schimbă starea. Astfel, cererea poate fi detectată oricând, nu numai în momentul tranziției acesteia la o stare activă.

10.2.2.4. Conflictele generate de canalele DMA

O altă resursă deficitară de pe magistrala ISA o reprezintă canalele DMA. Arhitectura originală PC utiliza un controler DMA 8237A-5, care pune la dispoziție patru canale DMA de 8 biți. Trei din aceste canale erau disponibile pentru conectorii de extensie. Începând cu arhitectura PC/AT a fost introdus un al doilea controler 8237A-5, care pune la dispoziție trei canale de 16 biți, al patrulea canal fiind utilizat pentru a recepționa cererile DMA de la controlerul original

Alocarea canalelor DMA este mai dificilă decât cea a nivelelor de întrerupere, din cauza priorităților asociate canalelor DMA. În plus, unele canale DMA sunt destinate transferurilor de date pe 8 biți, iar altele transferurilor pe 16 biți.

10.2.3. Standardul PnP pentru plăcile de extensie ISA

Specificațiile acestui standard descriu un mecanism hardware și software pentru a asigura posibilitatea configurării automate a plăcilor de extensie conectate la magistrala ISA. Aceasta implică rezolvarea conflictelor între plăcile de extensie și alocarea optimă a resurselor sistemului pentru aceste plăci, fără intervenția utilizatorului.

Principalele scopuri urmărite de acest standard sunt următoarele:

1. Posibilitatea adresării unice a plăcilor de extensie, chiar atunci când două sau mai multe plăci sunt configurate pentru a utiliza aceleași resurse ale sistemului.
2. Definirea unui protocol care să asigure citirea informațiilor de configurație a fiecărei plăci, care identifică utilizarea curentă a resurselor și opțiuni pentru fiecare resursă solicitată.
3. Definirea unui mecanism pentru setarea sau modificarea configurației curente a fiecărei plăci.
4. Asigurarea compatibilității cu plăcile mai vechi și cu cele viitoare, după cum urmează:
 - Plăcile ISA PnP vor putea coexista cu plăcile ISA obișnuite în orice sistem bazat pe magistrala ISA. În acest caz există însă posibilitatea ca sistemul să nu fie complet auto-configurabil.
 - Prin adăugarea programelor PnP (BIOS, utilitare și sistem de operare PnP), se va putea îmbunătăți gradul de auto-configurare al sistemului.
 - Un sistem care conține numai plăci ISA PnP și programe PnP adecvate va fi complet auto-configurabil.

10.2.3.1. Secvența de auto-configurare

10.2.3.1.1. Secvența de configurare a sistemului

Figura 10.1 prezintă o secvență posibilă a procesului de configurare a unui sistem care conține plăci ISA PnP [16]. Figura 10.1(a) se referă la un sistem cu BIOS obișnuit, iar Figura 10.1(b) se referă la un sistem cu BIOS PnP. Aceste figuri reprezintă doar exemplificarea modului în care se execută procesul de configurare, deoarece detaliile de implementare pot fi diferite.

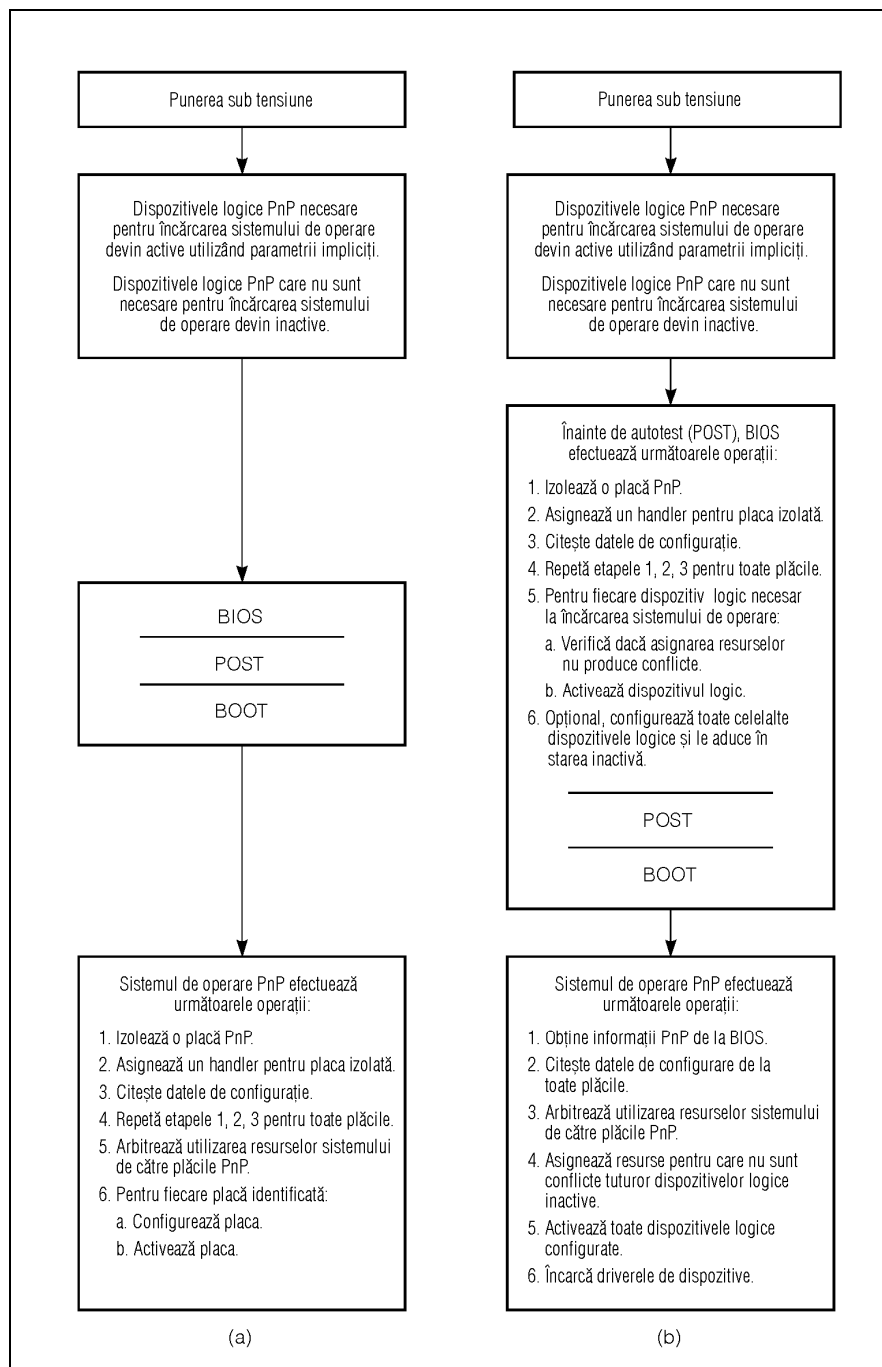


Figura 10.1. Secvența de configurare a unui sistem care conține plăci ISA PnP.

10.2.3.1.2. Secvența de configurare a plăcilor de extensie PnP

Etapele principale ale procesului de auto-configurare sunt următoarele:

1. Trecerea tuturor plăcilor ISA PnP în modul de configurare.
2. Izolarea succesivă a câte unei plăci ISA PnP.
3. Asignarea unui handler și citirea datelor despre resursele plăcii.
4. După determinarea cerințelor de resurse pentru fiecare placă, se utilizează handlerul pentru asignarea fără conflicte a resurselor pentru aceste plăci.
5. Activarea tuturor plăcilor ISA PnP și trecerea lor în modul de funcționare normală.

Programul PnP identifică și configurează echipamentele utilizând un set de comenzi definite de acest standard. Comenzile sunt executate utilizând trei porturi de I/E de 8 biți. Se utilizează o secvență de operații de scriere într-unul din aceste porturi pentru a valida logica PnP de pe plăcile de extensie ale sistemului. Această secvență se numește *cheie de inițializare*.

Toate plăcile PnP răspund la aceleași adrese de porturi, astfel încât este necesar un mecanism de *izolare* pentru a adresa o anumită placă la un moment dat. Protocolul de izolare utilizează un identificator unic pentru fiecare placă pentru a izola succesiv câte o placă PnP. După izolare, programul PnP asignează fiecărei plăci un handler, care se va utiliza pentru selecția plăcii respective.

Fiecare placă dispune de o structură de date pentru descrierea resurselor cerute de funcțiile acesteia. Pot exista mai multe funcții executate de o placă, fiecare funcție fiind definită ca un dispozitiv logic. Pentru fiecare dispozitiv logic sunt prevăzute informații despre resurse, și fiecare asemenea dispozitiv este configurat independent prin intermediul registrelor PnP standard.

După izolare, programul PnP citește de pe fiecare placă structura de date a resurselor. Se invocă în continuare un proces de arbitraj a resurselor pentru a determina alocarea resurselor pentru fiecare placă ISA. După asignarea resurselor, se poate invoca un mecanism pentru detectarea conflictelor de I/E. Acest mecanism asigură ca resursele de I/E asignate să nu fie în conflict cu plăcile ISA standard.

După terminarea configurării, plăcile PnP sunt trecute în modul de funcționare normală. Pentru a revalida modul de configurare, trebuie ex-

ecutată din nou secvența de inițializare. Aceasta previne alterarea accidentală a informațiilor de configurare.

10.2.3.2. Izolare și identificare

Se descrie în continuare asignarea porturilor utilizate pentru auto-configurare și protocolul de izolare [16].

10.2.3.2.1. Porturi de auto-configurare

Se utilizează trei porturi de 8 biți pentru accesul la spațiul de configurare al fiecărei plăci ISA PnP. Porturile sunt descrise în Tabelul 10.3. Aceste porturi au fost alese astfel încât să se evite conflictele cu funcțiile ISA existente, și să se minimizeze în același timp numărul de porturi necesare în spațiul de I/E ISA. Toate cele trei porturi utilizează un decodificator de 12 biți pentru adresele ISA.

Tabelul 10.3. *Porturile de auto-configurare ale plăcilor ISA PnP.*

Nume port	Adresă	Tip
ADDRESS	0279h (Port de stare LPT2)	W/O (Write-Only)
WRITE_DATA	0A79h (Port de stare LPT2 + 0800h)	W/O (Write-Only)
READ_DATA	Relocatibil în domeniul 0203h - 03FFh	R/O (Read-Only)

Spațiul de configurare este implementat ca un set de registre de 8 biți. Aceste registre sunt utilizate pentru transmiterea comenzilor, testarea stării, accesul la informațiile care descriu resursele și configurarea circuitelor PnP.

Pentru accesarea unui registru PnP, se înscrie mai întâi adresa registrului dorit în portul de adresă ADDRESS. Se pot executa apoi un număr arbitrar de comenzi de citire sau scriere cu registrul selectat. Pentru citire se utilizează portul READ_DATA, iar pentru scriere se utilizează portul WRITE_DATA.

Portul ADDRESS reprezintă de asemenea destinația cheii de inițializare. Setarea adresei portului READ_DATA se realizează prin înscrierea valorii corespunzătoare într-un registru de control PnP. Protocolul de izolare verifică faptul că pentru adresa portului READ_DATA nu există conflict.

10.2.3.2.2. Cheia de inițializare

Logica PnP este dezactivată la pornirea sistemului și trebuie validată prin program.

Cheia de inițializare plasează logica PnP în modul de configurare. Aceasta se realizează printr-o serie predefinită de scrieri în portul ADDRESS. Secvența operațiilor de scriere este decodificată de logica de pe placă. Dacă se detectează secvența corespunzătoare, se validează porturile de auto-configurare PnP.

Testarea cheii de inițializare este implementată prin hardware cu ajutorul unui registru de deplasare cu reacție liniară, numit LFSR (*Linear Feedback Shift Register*). Programul generează secvența de inițializare și realizează scrierea acesteia în portul ADDRESS printr-o secvență de cicluri de scriere de 8 biți. La fiecare scriere, logica de pe placă compară octetul înscris cu valoarea din registrul de deplasare. De fiecare dată când valorile sunt diferite, LFSR este resetat prin hardware la valoarea sa inițială. Programul poate reseta LFSR la valoarea sa inițială printr-o secvență de două cicluri de scriere a valorii 00h în portul ADDRESS, înainte de a transmite cheia de inițializare.

10.2.3.2.3. Protocolul de izolare

Se utilizează un algoritm simplu pentru izolarea fiecărei plăci PnP. Caracteristica principală a acestui protocol este faptul că fiecare placă dispune de un număr unic, numit *identificator serial*. Acest identificator are o lungime de 72 biți, fiind format din două câmpuri de câte 32 biți și o sumă de control de 8 biți (Figura 10.2). Primul câmp este un identificator al producătorului. Al doilea câmp poate conține orice valoare, de exemplu un număr de serie, o parte a unei adrese de rețea, sau o valoare fixă, astfel încât în același sistem să nu existe două plăci cu același identificator serial. Identificatorul este accesat bit cu bit de către logica de izolare și este utilizat pentru a diferenția între ele plăcile adaptoare.

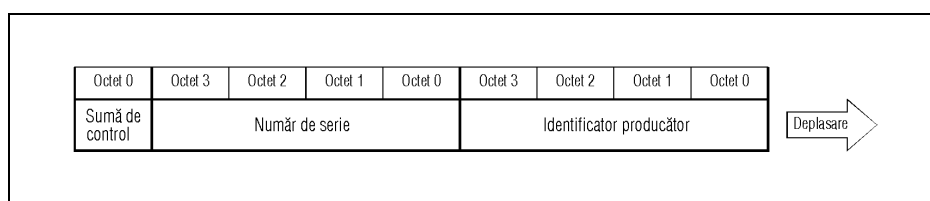


Figura 10.2. Identificatorul serial al plăcilor ISA PnP.

Protocolul de izolare poate fi invocat de programul PnP în orice moment. După trecerea plăcilor în modul de configurare, programul PnP generează 72 de perechi de cicluri de citire din portul READ_DATA. Data returnată de logica de pe fiecare placă depinde de valoarea fiecărui bit al identificatorului serial, care este examinat bit cu bit, în ordinea indicată în Figura 10.2. Pentru primii 64 de biți citiți, programul PnP generează o sumă de control utilizând datele citite. Această sumă de control este comparată cu suma de control citită în cadrul ultimilor 8 biți ai secvenței.

La terminarea unei etape a procesului de izolare, se va cunoaște identificatorul serial al unei singure plăci. Acestei plăci i se asignează un handler numit *număr de selecție al plăcii* (*Card Select Number - CSN*), care va fi utilizat ulterior pentru a selecta placa respectivă. Plăcile cărora li s-a asignat un număr de selecție nu vor mai participa în etapele următoare ale protocolului de izolare. Plăcile trebuie să aibă asignat un număr de selecție înainte de a răspunde la alte comenzi definite în specificațiile PnP.

Fiecare placă ISA PnP conține un registru CSN de 8 biți care păstrează numărul de selecție al plăcii. La pornirea sistemului, toate plăcile inițializează acest registru cu 00h. După ce o placă a fost izolată, conținutul registrului CSN este utilizat pentru selecția plăcii, fără a mai fi necesară execuția protocolului de izolare.

Protocolul de izolare permite ca suma de control de 8 biți să fie păstrată într-o memorie nevolatilă a plăcii sau să fie generată în timp real de către logica plăcii. Pentru generarea sumei de control se utilizează registrul LFSR și același algoritm utilizat în cadrul secvenței de inițiere.

10.2.3.3. Stările PnP

Stările PnP sunt descrise în continuare.

- **Wait for Key:** Plăcile trec în această stare la punerea sub tensiune sau la ca răspuns la comenzile *Reset* și *Wait for Key*. În această stare nu este activă nici o comandă, până când se detectează cheia de inițializare pe magistrala ISA. *Wait for Key* este starea implicită a plăcilor PnP în timpul funcționării normale. După configurare și activare, toate plăcile trebuie aduse prin program în această stare.
- **Sleep:** În această stare, plăcile PnP așteaptă o comandă *Wake* [CSN]. Această comandă validează în mod selectiv trecerea uneia sau a mai multor plăci în starea *Isolation* sau *Config*, în funcție de parametrul comenzii (10.2.3.4.2). Ieșirea din starea *Sleep* se rea-

lizează ca răspuns la o comandă *Wake* [CSN], dacă parametrul acestei comenzi coincide cu valoarea CSN a plăcii.

- **Isolation:** În această stare, plăcile PnP răspund la comenzile de citire ale registrului Serial Isolation, conform protocolului de izolare. După izolarea unei plăci și înscrierea numărului de selecție în registrul CSN, placa va trece în starea *Config*.
- **Config:** O placă aflată în această stare răspunde la toate comenzile de configurare, ca citirea informațiilor de configurare și selecția resurselor. O singură placă se poate afla în această stare la un moment dat.

10.2.3.4. Registrele PnP

10.2.3.4.1. Spațiul registrelor PnP

Spațiul registrelor PnP este divizat în trei părți, pentru controlul plăcii, controlul dispozitivului logic și configurarea dispozitivului logic (Figura 10.3).

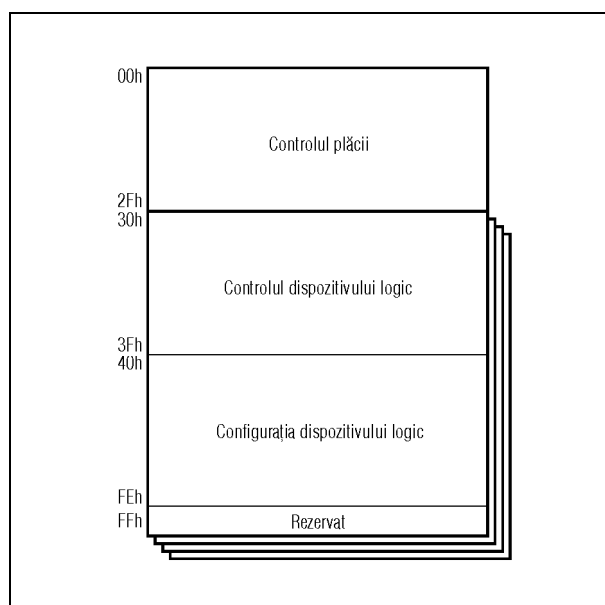


Figura 10.3. Spațiul registrelor PnP.

Fiecare placă conține un singur registru de control al plăcii, care se utilizează pentru controlul funcțiilor globale ale plăcii. Celelalte tipuri

de registre sunt repetate pentru fiecare dispozitiv logic, și ele controlează funcțiile dispozitivului, respectiv utilizarea resurselor magistralei ISA.

10.2.3.4.2. Registrele de control

Sunt descrise în continuare registrele standard de control și comenzile care pot fi utilizate cu aceste registre [16]. Pe lângă aceste registre standard, există un număr de registre definite de producător prin care se pot configura și resursele nestandard ale magistralei ISA prin mecanismul PnP.

- **Set RD_DATA Port (00h).** Scrierea în acest registru setează adresa portului READ_DATA utilizat pentru toate plăcile PnP. Biții [7:0] ai valorii înscrise în acest registru se utilizează ca biți [9:2] ai magistralei de adrese ISA. Biții [1:0] ai acestei magistrale sunt fixați la valoarea binară "11", iar biții [15:10] sunt fixați la valoarea "000000". Scrierea în acest registru se poate realiza numai în starea *Isolation*.
- **Serial Isolation (01h).** O citire din acest registru determină ca plăcile PnP aflate în starea *Isolation* să răspundă la ciclul de citire de pe magistrala ISA în modul descris de protocolul de izolare. Plăcile care nu mai participă la protocolul de izolare trec în starea *Sleep*.
- **Config Control (02h).** Permite activarea a trei comenzi independente, prin setarea la 1 a biților corespunzători ai registrului. Acești biți sunt reșetați automat la 0 după executarea comenzilor.

Comanda Reset. Această comandă este transmisă plăcilor PnP prin înscrierea valorii 01h în registrul Config Control. La această comandă răspund toate plăcile aflate în oricare stare, cu excepția stării *Wait for Key*. Comanda determină reșetarea tuturor dispozitivelor logice. Registrele de configurație ale acestor dispozitive sunt încărcate cu valorile lor implicite din memoria nevolatilă. Starea PnP și conținutul registrelor Set RD_DATA Port și CSN nu se modifică.

Comanda Wait for Key. Această comandă este transmisă plăcilor PnP prin înscrierea valorii 02h în registrul Config Control. La această comandă răspund toate plăcile aflate în oricare stare. Comanda determină trecerea tuturor plăcilor PnP în starea *Wait for Key*. Conținutul registrelor CSN și starea dispozitivelor logice nu se modifică.

Comanda Reset CSN. Această comandă este transmisă plăcilor PnP prin înscrierea valorii 04h în registrul Config Control. Toate plăci-

le PnP aflate în oricare stare, cu excepția stării *Wait for Key*, vor reseta registrul lor CSN la 0.

- **Wake (03h)**. O scriere în acest registru determină execuția comenzii *Wake* [CSN]. Această comandă se utilizează pentru a trece plăcile ISA din starea *Sleep* în starea *Isolation* sau în starea *Config*. O comandă *Wake* [CSN] cu parametrul 0 are ca efect trecerea în starea *Isolation* a tuturor plăcilor care nu au un număr de selecție CSN asignat. O comandă *Wake* [CSN] cu un parametru diferit de 0 are ca efect trecerea în starea *Config* a plăcii cu numărul de selecție identic cu parametrul comenzii. O placă aflată în starea *Isolation* sau *Config* care recepționează o comandă *Wake* [CSN] cu un parametru diferit de numărul ei de selecție va trece în starea *Sleep*.
- **Resource Data (04h)**. Citirea acestui registru va returna octetul următor din datele de configurație ale plăcii aflate în starea *Config*. Înainte de citirea registrului *Resource Data* este necesară citirea registrului de stare pentru a confirma faptul că octetul de configurație este disponibil.
- **Status (05h)**. Bitul 0 al registrului de stare indică prin valoarea 1 faptul că octetul următor al datelor de configurație este disponibil și poate fi citit.
- **CSN (06h)**. Conține numărul de selecție al plăcii. Valoarea 0 a acestui număr indică o placă neidentificată. Pentru plăcile identificate, CSN poate lua valori între 0 și 255. Valorile CSN sunt asignate secvențial începând cu 1. Valoarea CSN se utilizează pentru selecția unei plăci prin comanda *Wake* [CSN].
- **Logical Device Number (07h)**. Se utilizează pentru selectarea dispozitivului logic curent. Numerele dispozitivelor logice sunt determinate de ordinea în care se citesc datele de configurație ale acestor dispozitive. Toate comenzile ulterioare de configurare vor accesa registrele dispozitivului logic selectat.
- **Activate (30h)**. Pentru fiecare dispozitiv logic există câte un registru *Activate*, utilizat pentru activarea dispozitivului respectiv. Dacă bitul 0 al acestui registru este 1, dispozitivul logic este activ.
- **I/O Range Check (31h)**. Acest registru permite programului PnP să determine dacă există un conflict între adresele de I/E care au fost asignate unui dispozitiv logic și adresele asignate altei plăci. Testarea domeniului de adrese se realizează prin returnarea valorilor 55h și apoi AAh ca răspuns la comenzile de citire din toate zonele de adrese utilizate de dispozitivul logic. Programul PnP execută citirea tuturor porturilor care sunt utilizate de dispozitivul logic și verifică dacă se re-

turnează datele corecte. Dacă se detectează un conflict, programul PnP va reloca domeniul adreselor de I/E asignate dispozitivului logic.

10.2.3.5. Configurarea echipamentelor PnP

10.2.3.5.1. Citirea datelor de configurație

Datele de configurație ale unei plăci pot fi citite numai dacă placa se află în starea *Config*. O placă trece în starea *Config* la terminarea procesului în care a fost izolată și i s-a asignat un număr de selecție. Placa poate trece în starea *Config* și la recepționarea comenzii *Wake* [CSN] cu un parametru identic cu numărul de selecție al plăcii.

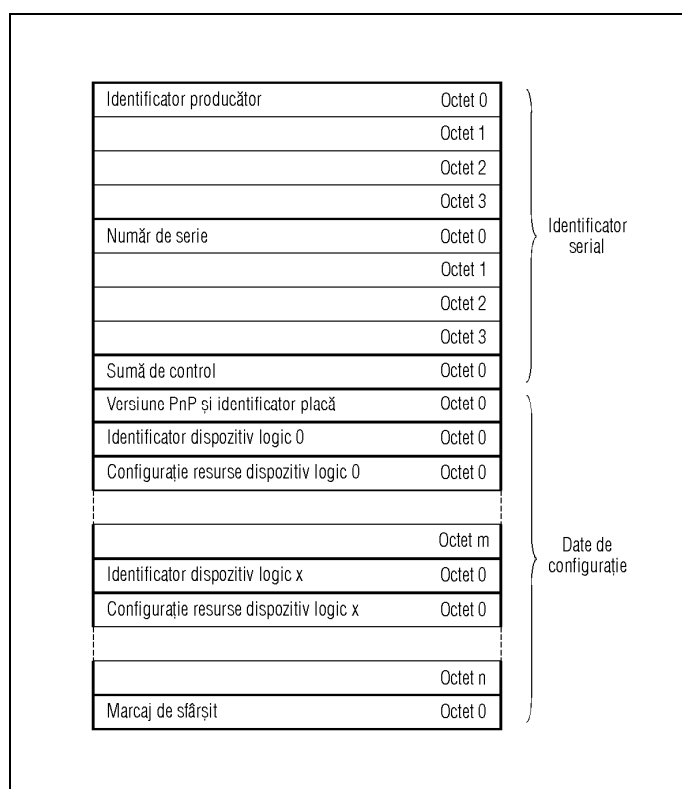


Figura 10.4. Șirul de date format din identificatorul serial și datele de configurație.

Atât identificatorul serial de 72 biți, cât și datele de configurație trebuie considerate ca un singur șir de date care sunt accesibile serial,

octet cu octet (Figura 10.4). Există un pointer la acest șir de date, care indică următorul octet care va fi citit din șir. Acest pointer este resetat ca răspuns la orice comandă *Wake* [CSN]. Ca urmare, în cazul în care placa trece în starea *Config* direct din starea *Sleep* ca răspuns la comanda *Wake* [CSN], trebuie citit mai întâi identificatorul serial de 9 octeți înainte de a putea avea acces la datele de configurație. În cazul în care placa trece în starea *Config* din starea *Isolation* (deci după ce s-a executat protocolul de izolare și toți biții identificatorului serial au fost citiți), prima citire a registrului *Resource Data* va returna primul octet de configurație.

Înainte citirii unui octet de configurație, trebuie să se testeze bitul 0 al registrului de stare. Dacă acest bit este setat, un octet de configurație este disponibil în registrul *Resource Data*. După citirea acestui registru, bitul 0 este resetat automat, și procesul se repetă pentru următorii octeți de configurație.

10.2.3.5.2. Configurarea resurselor

Pentru programarea unei plăci, aceasta este trecută mai întâi în starea *Config* prin comanda *Wake* [CSN]. Celelalte plăci vor trece în starea *Sleep*. Apoi se selectează dispozitivul logic prin înscrierea numărului dispozitivului logic în registrul *Logical Device Number*. Acest număr este determinat de ordinea în care se citesc datele de configurație ale dispozitivelor logice. Primul dispozitiv logic citit va avea numărul 0, al doilea numărul 1, ș.a.m.d. Dispozitivele logice pot fi programate în orice ordine. Adresele registrelor de configurație ale unui dispozitiv sunt determinate pe baza numărului dispozitivului logic.

Plăcile PnP dispun de următoarele registre utilizate pentru configurarea necesarului de resurse standard de pe magistrala ISA pentru fiecare dispozitiv logic:

- Registre pentru adresele de bază ale memoriei (până la patru zone necontigue);
- Registre pentru adresele de bază de I/E (până la opt zone necontigue);
- Registre pentru selecția nivelelor de întrerupere (până la două nivele separate de întrerupere);
- Registre pentru selecția canalelor DMA (până la două canale DMA).

Aceste registre sunt de tip R/W și reflectă întotdeauna starea curentă a dispozitivelor logice ale plăcii. Dacă o resursă nu este programabilă, registrul de configurație respectiv este de tip R/O.

După înscrierea registrelor respective, ultima etapă a configurării este setarea bitului de activare al fiecărui dispozitiv logic. După încheierea configurării, toate plăcile trebuie trecute în starea *Wait for Key*.

10.2.4. Specificațiile PnP pentru interfața SCSI

10.2.4.1. Scopul specificațiilor PnP SCSI

Scopul acestor specificații este de a simplifica utilizarea interfeței SCSI la calculatoarele personale. Deși nu reprezintă un standard propriu-zis, aceste specificații se bazează pe standardul SCSI-2 și pe unele documente ale standardului SCSI-3 (*SCSI-3 Parallel Interface - SPI*). Firmele care au contribuit la dezvoltarea acestor specificații sunt: *Adaptec*, *AT&T Global Information Solutions*, *Digital Equipment Corporation*, *Future Domain*, *Maxtor* și *Microsoft*.

Prin specificațiile PnP SCSI se simplifică utilizarea terminatoarelor. Terminatoarele nu sunt incluse în cadrul perifericelor SCSI, astfel că se elimină confuzia legată de prezența sau absența unui terminator și posibilitatea terminatoarelor suplimentare pe magistrala SCSI. De asemenea, se elimină problema atribuirii unice a identificatorilor SCSI, prin utilizarea unui protocol numit SCAM, care realizează asignarea automată a acestor identificatori. Dispozitivele mai vechi pot coexista cu dispozitivele SCAM mai noi prin utilizarea unei convenții de asignare a identificatorilor pe baza clasei perifericelor.

Toate terminatoarele dintr-un sistem PnP SCSI trebuie să se conformeze cu cerințele definite în specificațiile SCSI-3 *Parallel Interface*. Terminatoarele trebuie să funcționeze corect în gama de tensiuni între 4.0 și 5.25 V pentru semnalul *TERMPWR*. Sunt necesare terminatoare active, realizate cu un regulator de tensiune.

10.2.4.2. Asignarea identificatorilor SCSI

Fiecare dispozitiv de pe magistrala SCSI necesită un identificator SCSI unic. Procesul de configurare a magistralei SCSI este complicat de conflictele între identificatori, setările necorespunzătoare ale comutatoarelor și amplasarea acestora în cadrul dispozitivelor. Soluția la aceste probleme este asignarea automată a identificatorilor SCSI.

În standardul SCSI-3 pentru interfața paralelă a fost inclus un protocol numit SCAM (*SCSI Configured AutoMagically*), care asignează automat identificatorii pentru dispozitivele SCSI. Toate dispozitivele SCSI PnP

trebuie să se conformeze nivelului 1 sau 2 din specificațiile acestui protocol.

Protocolul SCAM este proiectat astfel încât dispozitivele mai vechi pot fi detectate și utilizate. Dacă sunt prezente mai multe asemenea dispozitive, utilizatorul trebuie să asigure ca între acestea să nu existe conflicte.

Pentru dispozitivele SCAM este important să existe setări implicite ale identificatorilor în cazul în care acestea se utilizează pe o magistrală SCSI fără un adaptor SCAM. Tabelul 10.4 prezintă asignările recomandate ale identificatorilor pentru dispozitivele SCSI PnP. Identificatorii 5, 1 și 0 se utilizează în timpul configurării sistemului pentru dispozitive adiționale.

Tabelul 10.4. *Asignările implicite ale identificatorilor pentru dispozitivele SCSI PnP.*

ID SCSI	Dispozitiv
7	Adaptor SCSI
6	Unitate de disc magnetic
5	
4	Bandă magnetică sau disc CD-R/W
3	CD-ROM
2	Scanner/Imprimantă
1	
0	

Pentru asignarea automată a identificatorilor SCSI, se utilizează următorul algoritm [17]:

În timpul procedurii de izolare, adaptorul SCAM izolează câte un dispozitiv SCAM și memorează identificatorul său implicit. Adaptorul asignează dispozitivului SCAM un identificator soft după ce termină procedura de izolare a dispozitivului și înainte de a începe izolarea următorului dispozitiv SCAM.

Adaptorul SCAM asignează identificatori soft care sunt disponibili și care nu sunt utilizați deja de alte dispozitive de pe magistrală. Prima opțiune va fi identificatorul implicit al dispozitivului. Următoarea opțiune va fi identificatorul disponibil cu valoarea mai mică imediat următoare. Dacă identificatorul 0 nu este disponibil, adaptorul SCAM va asigna identificatorul disponibil cu valoarea cea mai mare. Dacă nu este disponibil nici un identificator, nu se va asigna un identificator soft dispozitivului.

10.2.4.3. Configurații de sisteme

Există trei configurații posibile de sisteme SCSI:

- Configurație internă: Toate perifericele SCSI sunt în interiorul sistemului gazdă.
- Configurație externă: Toate perifericele SCSI sunt externe sistemului gazdă.
- Configurație mixtă: Există o combinație de periferice SCSI interne și externe.

Adaptoarele PnP SCSI care permit o configurație mixtă trebuie să aibă în cadrul conectorului din sistemul gazdă un terminator care poate fi validat sau invalidat în mod automat.

Perifericele PnP SCSI nu se pot afla la capătul magistralei SCSI. Pentru perifericele PnP SCSI adăugate la magistrala SCSI, se va instala un terminator extern într-un conector liber al ultimului periferic de pe magistrală.

În cazul unei *configurații interne*, terminarea magistralei este asigurată de adaptorul SCSI și un terminator intern. Magistrala SCSI internă este rutată în cadrul sistemului către una sau mai multe periferice SCSI. La capătul magistralei interne SCSI se află un terminator intern.

În cazul unei *configurații externe*, terminarea magistralei SCSI este asigurată de un terminator intern și unul extern. Perifericele trebuie să dispună de doi conectori cu 50 de poziții. Magistrala SCSI internă este rutată de la un conector SCSI extern la dispozitivele SCSI din cadrul sistemului, și apoi la al doilea conector SCSI extern. La ultimul periferic SCSI trebuie instalat un terminator extern.

În cazul unei *configurații mixte*, terminatorul din conectorul sistemului gazdă va fi validat sau invalidat automat astfel încât magistrala SCSI să fie terminată corect, pe baza detecției configurației sistemului. Aceasta se realizează prin detecția mecanică sau electronică a cablului atașat, sau detecția electronică a dispozitivelor SCSI atașate. Terminatorul trebuie invalidat atunci când există atât un cablu intern cât și unul extern, sau există atât dispozitive SCSI interne cât și externe.

10.2.5. Nivele de implementare ale standardelor PnP

Pentru ca un sistem să se conformeze în totalitate standardelor PnP, trebuie să existe simultan mai multe nivele de implementare a acestor standarde. La nivelul cel mai înalt, *aplicația* trebuie să țină cont de

standardele PnP. Ea trebuie să aibă posibilitatea de a solicita automat resursele necesare de la sistemul de operare și să își ajusteze necesitățile în mod dinamic în funcție de resursele alocate de sistemul de operare.

La următorul nivel, *sistemul de operare* trebuie să permită controlul resurselor pentru a elimina conflictele, să satisfacă necesitățile aplicațiilor și să accepte instalarea și reconfigurarea automată a driverelor pentru echipamente.

Componenta BIOS a sistemului trebuie să aibă posibilitatea de a izola și de a interoga echipamentele PnP, de a raporta eventualele conflicte de alocare a resurselor sau de a încerca rezolvarea acestora prin reconfigurare. De asemenea, componenta BIOS trebuie să accepte solicitările de reconfigurare provenite de la sistemul de operare.

Adaptoarele și echipamentele trebuie să accepte în totalitate specificațiile PnP referitoare la identificarea echipamentului, raportarea configurației și posibilitatea de reconfigurare.

Se prezintă nivelele de implementare necesare a standardelor PnP pentru diferite operații de configurare a sistemului [8].

1. Configurarea automată a echipamentelor conforme cu standardele PnP necesită o componentă BIOS sau un sistem de operare PnP.
2. Configurarea automată atât a echipamentelor PnP, cât și a driverelor acestora necesită atât o componentă BIOS, cât și un sistem de operare conforme cu standardele PnP.
3. Configurarea și reconfigurarea dinamică a echipamentelor PnP și a driverelor acestora (efectuate după pornirea sistemului) necesită o componentă BIOS și un sistem de operare conforme cu standardele PnP.
4. Configurarea și reconfigurarea dinamică a echipamentelor, a driverelor acestora și a aplicațiilor necesită o componentă BIOS, un sistem de operare și aplicații conforme cu standardele PnP.

În practică, există sisteme mixte, parțial conforme cu standardele PnP, în care unele adaptoare și echipamente sunt conforme PnP, iar altele nu. Un sistem conform cu standardele PnP permite utilizarea atât a echipamentelor PnP, cât și a celor non-PnP. Se reduc însă șansele de configurare fără probleme la o modernizare ulterioară a sistemului.

Dacă sistemul de operare nu este de tip PnP, dar există o componentă BIOS conformă cu standardele PnP, aceasta va încerca să configureze sistemul astfel încât să minimizeze riscul conflictelor. Dacă

există adaptoare și echipamente PnP, ele vor fi configurate automat cel puțin la parametrii lor prestabiliți.

Dacă sistemul de operare este conform cu standardele PnP, dar nu există o componentă BIOS adecvată, sistemul de operare preia multe din funcțiile PnP ale componentei BIOS, fiind posibilă configurarea și alocarea corectă a resurselor.

În cazul în care nici sistemul de operare, nici componenta BIOS nu sunt conforme cu aceste standarde, alegerea unui adaptor sau echipament PnP poate simplifica procedura de instalare și configurare. Aceste adaptoare și echipamente dispun în general de utilitare de instalare care permit configurarea lor manuală, cu ajutorul unui meniu de instalare. Astfel se elimină cel puțin confuzia creată de jumpere și comutatoare.

10.3. Desfășurarea lucrării

10.3.1. Se va desena structura registrului LFSR. Care este secvența de valori corespunzătoare cheii de inițializare care trebuie generată și înscrisă în portul ADDRESS ?

10.3.2. Se va descrie modul de generare a sumei de control cu ajutorul registrului LFSR.

10.3.3. Se va descrie în detaliu protocolul hardware și software utilizat pentru izolarea unei plăci.

10.3.4. Se vor pune în evidență printr-o organigramă tranzițiile posibile între diferite stări ale unei plăci ISA PnP și comenzile care sunt active în fiecare stare.

10.3.5. Se va indica secvența stărilor prin care trec plăcile ISA PnP în cadrul protocolului de izolare.

10.3.6. Se vor considera diferite exemple de configurații SCSI PnP, pentru fiecare fiind specificată o listă de echipamente și identificatorii lor implicați. Se va indica modul în care se realizează asignarea identificatorilor soft utilizând algoritmul SCAM. Se presupune că adaptorul SCSI are identificatorul 7.

10.3.7. Se vor prezenta exemple pentru configurațiile SCSI (internă, externă și mixtă), și se va pune în evidență modul de amplasare al terminatoarelor.