

4. PORTUL PARALEL

Portul paralel este de departe cea mai populară interfață a calculatorului. Acest port nu a fost multă vreme reglementat prin norme stabilite de vreun organism de standardizare (aproximativ jumătate din viața sa). Ca norme de funcționare erau acceptate regulile propuse de IBM în cadrul definirii calculatorului personal IBM-PC. Începând din 1994 transferul pe la portul paralel este reglementat de standardul IEEE-1284 sub următoarele aspecte:

- definește 5 moduri de operare pentru transferul paralel al datelor;
- definește protocolul de comunicație;
- definește interfața mecanică (cabluri și conectori);
- definește interfața electrică (emițătoare, receptoare terminator, impedențe).

Standardul IEEE-1284 definește comunicația paralelă bidirecțională de mare viteză între calculatorul PC și periferice externe. Viteza la care este posibil transferul în acest standard este de 50 până la 100 de ori mai mare decât în modul paralel definit inițial pentru un calculator PC, fără a pierde compatibilitatea cu modul inițial. Standardul IEEE-1284 nu reglementează decât semnalele (stabilește temporizări și relații între semnale). Nu conține specificații legate de natura datelor vehiculate sau de sistemul de control care elaborează semnalele.

Standardul IEEE-1284 este cunoscut sub numele *Standard Signaling Method for Bi-directional Parallel Peripheral Interface for Personal Computers*.

4.1. ISTORIC

Când IBM a lansat în 1981 calculatorul PC, portul paralel a fost inclus ca o alternativă a portului serial (mai lent) pentru conectarea imprimantelor matriciale mai performante (la acea vreme). Dacă la început imprimanta era singurul periferic conectat la portul paralel, în timp, nevoia de conectare din exterior a crescut și portul paralel a devenit principala cale de conectare a diverselor echipamente periferice. Acum se conectează aici imprimanta, unități de disc portabile, unități de bandă magnetică, unități CD-ROM sau adaptoare locale de rețea. Între timp PC-ul a evoluat foarte mult, iar la portul paralel nu au avut loc nici schimbări de arhitectură, nici creșteri de performanță. Portul paralel, în formula inițială, prezintă următoarele neajunsuri:

- rata de transfer de maxim 150 KBps;
- nu există standard pentru interfața electrică;
- limitare a lungimii cablului exterior la 1.8m.

În 1991 fabricanții de imprimante au luat inițiativa dezvoltării unui standard pentru a conecta în rețea imprimante inteligente. Astfel a luat naștere Network Printing Alliance - NPA. Cerințele formulate pentru noul standard sunt în principal următoarele:

- legătura bidirecțională de mare viteză;
- viteză mai mare de 1 MBps;
- compatibilitate cu portul paralel inițial.

Propunerea a fost înaintată IEEE și standardul a apărut în martie 1994.

4.2. PORTUL PARALEL STANDARD (SPP)

Portul paralel, așa cum este implementat în PC, constă într-un conector cu 17 linii de semnal și 8 linii de masă. Liniile de semnal sunt clasificate în 3 categorii:

- linii de control (4);
- linii de stare (5);
- linii de date (8).

Tabelul 4.1. conține simbolul, descrierea și alocarea la conectorii tipici a fiecărui semnal SPP.

Tabel 4.1. Semnalele portului paralel standard. (Simbolul și descrierea SPP)

| Grupa | Semnal SPP | In/Out | Descrierea semnalului | DB (25) | Centronics |
|---------|------------|--------|--|---------|------------|
| Control | nSTROBE | Out | Activ LOW. Sunt date valide pe liniile de date | 1 | 1 |
| | nAUTOFEED | Out | Activ LOW. Imprimanta va include LF la fiecare CR. | 14 | 14 |
| | nSELECTIN | Out | Activ LOW. Spune imprimantei că este selectată. | 17 | 36 |
| | nINIT | Out | Activ LOW. Resetează imprimanta. | 16 | 31 |
| Stare | nACK | In | Un puls negativ pentru a semnaliza că ultimul byte a fost recepționat. | 10 | 10 |
| | BUSY | In | Nivel HIGH, anunță calculatorul că imprimanta nu poate prelua date. | 11 | 11 |
| | PE | In | Nivel HIGH, imprimanta nu are hârtie. | 12 | 12 |
| | SELECT | In | Nivel HIGH, imprimanta este ON LINE. | 13 | 13 |
| | nERROR | In | Nivel LOW, la imprimantă există o condiție de eroare. | 15 | 32 |
| Date | DATA [1:8] | Out | 8 linii de date (numai ieșire) | 2-9 | 2-9 |
| GND | | - | masă de semnal | 18-25 | 15-30 |

Semnalul nSTROBE este emis de calculator pentru a comunica perifericului că există informație validă pe liniile de date. Pentru aceasta se generează un impuls negativ care apare după 0.5 μ S de la apariția datelor și durează cel puțin 0.5 μ S și cel mult 50 μ S.

Semnalul BUSY comunică cu nivel HIGH că nu este pregătit să preia date, altele decât cele curente sau, că pur și simplu nu poate prelua date dintr-un motiv sau altul (eroare, nu este on-line, nu are hârtie). Semnalul BUSY este elaborat de periferic și comută în “1” imediat după detectarea semnalului nSTROBE activ. Rămâne în această stare până la terminarea recepției datelor semnalizate valide.

Semnalul nACK este elaborat de periferic și comunică printr-un impuls negativ că ultimul Byte a fost recepționat. Durata impulsului este de 0.5 μ S - 10 μ S.

Semnalele nSTROBE, BUSY și nACK controlează fluxul de date, celelalte semnale de interfață sunt ajutătoare în dialogul dintre calculator și o imprimantă. Semnificația lor este descrisă pe scurt în coloana de descriere a semnalului din Tabelul 4.1. Nu toate imprimantele pot lucra cu toate semnalele ajutătoare.

Observație: nSELECTIN permite comutarea imprimantei în regim economic.

Portul paralel este mapat în spațiul I/O al PC-ului. Fiecărei grupe de semnale îi este asociat un registru; în cadrul fiecărui registru, fiecărui semnal îi este precizată poziția. Fiecare registru este apelabil cu o adresă de port. Adresa de bază a portului paralel poate fi 3BCh, 378h sau 278h. La această adresă se apelează registrul de date, iar celelalte (stare și control) se apelează cu un offset. Tabelul 4.2. conține informații legate de registrele portului paralel SPP.

Tabel 4.2. Registre SPP

| Offset | Nume | mod de operare | Descriere |
|--------|----------------------------------|----------------|-----------------------------------|
| 0 | Registru de date | | Conține biți de date; R/W |
| | DB ₀ -DB ₇ | neinversat * | DATA [1:8] |
| 1 | Registru de stare | | Conține biți de stare; R |
| | DB ₃ | inversat * | nERROR |
| | DB ₄ | neinversat * | SELECT |
| | DB ₅ | neinversat * | PE |
| | DB ₆ | inversat * | nACK |
| | DB ₇ | inversat * | BUSY |
| | | | |
| 2 | Registru de control | | Conține biți de control; W |
| | DB ₀ | neinversat * | nSTROBE |
| | DB ₁ | neinversat * | nAUTOFEED |
| | DB ₂ | inversat * | nINIT |
| | DB ₃ | neinversat * | nSELECTIN |
| | DB ₄ | neinversat * | IRQE ** |

* Cu fiecare bit din aceste registre poate fi controlat nivelul unui semnal al interfeței. Specificația inversat și neinversat are următoarea semnificație:

inversat: setarea unui "1" are ca rezultat generarea unui nivel LOW pentru semnalul respectiv.

neinversat: setarea unui "1" are ca efect generarea unui nivel HIGH pentru semnalul respectiv.

A nu se confunda cu starea activă a semnalului. Spre exemplu, nSTROBE este activ LOW; pentru a activa acest semnal se înscrie un "0" în registrul de control, pe rangul D₀. În schimb, pentru a activa nINIT, care este activ tot LOW, se va înscrie un "1" în registrul de control pe rangul D₂.

** Bitul alocat rangului D₄ în registrul de control, IRQE, nu activează vreun semnal de ieșire; înscrierea unui "1" pe această poziție validează acceptarea activării semnalului nACK ca o cerere de întrerupere venită din partea echipamentului periferic. Această facilitate nu este utilizată în general de imprimante, ea a fost prevăzută pentru aplicații de transfer de date pe la portul paralel.

Într-un sistem PC pot să se găsească mai multe porturi paralele, caz în care adresele sunt alocate pe rând. Programatorul poate să nu cunoască adresa portului; acestea sunt referite cu numele generice LPT1 (sau PRN), LPT2 și LPT3. La inițializarea sistemului, la testul hardware, adresele de bază la care sunt apelate porturile paralele (câte au fost identificate) vor fi înscrise în RAM începând cu adresa 0400:008 astfel:

| Adresă | Conținut | Observații |
|-----------|----------|----------------------------------|
| 0040:0008 | 78 03 | LPT1(PRN) se află la adresa 378h |
| 0040:000A | 00 00 | nu există LPT2 |
| 0040:000C | 00 00 | nu există LPT3 |

BIOS-ul preia adresele de bază pentru localizarea registrelor portului paralel din acest tabel aflat în memoria RAM.

BIOS-ul conține rutine de comandă a transferului la portul paralel standard. Acestea sunt apelabile cu întreruperea soft INT 17h. Tabelul 4.3. conține informațiile necesare utilizării acestei întreruperi, iar Tabelul 4.4. conține interpretarea fiecărui bit din cuvântul de stare întors de această întrerupere.

Tabel 4.3. Semnificația registrelor la INT 17h

Tabel 4.4. Semnificația biților pentru stare

| Funcție | Registru | Registru returnat | Bit | Semnificație |
|--------------|---|-------------------|-----|--------------|
| Byte Out | AL=Byte Out AH=00h DX=Nr. port (LPT1=0) | AH=stare | 0 | Timeout |
| | | | 1 | 0 |
| | | | 2 | 0 |
| Reset | AH=01h DX=Nr. port (LPT1=0) | AH=stare | 3 | Error |
| | | | 4 | Select |
| Citire Stare | AH=02h DX=Nr. port (LPT1=0) | AH=stare | 5 | Paper empty |
| | | | 6 | Acknowledge |
| | | | 7 | Busy |

Valoarea biților nu corespunde nivelului activ al semnalului cu același nume, ci biții au valoarea “1” atunci când semnalul respectiv este activ.

4.3. MODURI DE TRANSFER

Standardul IEEE-1284 definește 5 moduri de transferare a datelor. Fiecare mod face posibil transferul înainte (forward; de la PC la periferic), înapoi (reverse; de la periferic la PC) sau bidirecțional (doar în regim half-duplex). Modurile definite sunt sintetizate în Tabelul 4.5. (pagina următoare).

Toate porturile pot realiza legături bidirecționale folosind modurile compatibil sau nibble.

Modul byte poate fi utilizat doar în 25% din porturile paralele existente (porturile calculatoarelor PS/2).

În modurile compatibil, nibble și byte transferul datelor este realizat în întregime sub conducerea unui program - asistat integral de software. Driver-ul va scrie datele, va verifica liniile de handshake (ex. BUSY) și va elabora semnale de comandă corespunzător

situației (ex. STROBE). Datorită ponderii crescute a software-ului viteza este limitată la 50-100 KBps.

Spre deosebire de cele trei moduri descrise anterior, modurile EPP și ECP folosesc hardware-ul pentru a asista transferul de date. Transferarea unui byte se poate face cu o singură comandă OUT. Controlerul I/O (hw) se ocupă de urmărirea semnalelor de handshake și de control. Modurile EPP și ECP sunt implementate de majoritatea circuitelor controler I/O care se produc astăzi.

Tabel 4.5. Moduri de transfer

| Sensul transferului | Mod | Descriere |
|----------------------------------|--|---|
| Forward (PC⇒ periferic) | Modul compatibil (Compatibility mode) | Modul SPP sau “Centronics” |
| Reverse (periferic⇒ PC) | Modul Nibble (Nibble Mode) | Se transferă 4 biți într-o comandă folosind liniile de stare pentru transferul datelor. |
| | Modul Byte (Byte Mode) | Se transferă 8 biți într-o comandă folosind liniile de date. Mod referit ca enhanced bidirecțional. |
| Bidirecțional (PC⇔ periferic) | Modul EPP | Enhanced Parallel Port Este folosit mai ales de periferice care nu sunt imprimante (CD-ROM, unitate de bandă, unitate de disc sau adaptoare de rețea). |
| | Modul ECP | Extended Capabilities Port Folosit în general de noua generație de imprimante. |

4.3.1. Modul Compatibil

Acest mod definește protocolul utilizat de majoritatea PC-urilor pentru transferarea de date unei imprimante. Este numit deseori și mod “Centronics” și este modul utilizat la portul paralel standard (SPP).

Transferul începe prin verificarea stării imprimantei (dacă este BUSY sau dacă este vreo stare de eroare). La o imprimantă BUSY este activ la inițializarea acesteia, în stare off-line, în cazul apariției unei erori, la avansul hârtiei sau la preluarea datelor.

Dacă BUSY nu este activ, sunt plasate datele pe liniile de date prin înscrierea registrului de date și se activează semnalul nSTROBE pentru a valida datele de pe liniile de date. Impulsul nSTROBE trebuie să dureze cel puțin 1 μ S, dar nu mai mult de 50 μ S (pentru a nu încetini transferul).

La sesizarea prezenței datelor valide, perifericul răspunde cu BUSY. Durata semnalului BUSY nu este limitată, dar este condiționat momentul activării. BUSY se activează la cel târziu 0.5 μ S după frontul căzător al semnalului nSTROBE (pentru ca unitatea centrală să poată opri la timp trimiterea altor date).

Preluarea datelor de către periferic este semnalizată cu nACK. Durata acestui semnal trebuie să fie de 5-10 μ S și trebuie să apară cu cel puțin 7 μ S înainte de dezactivarea lui BUSY. Cel târziu la 7 μ S după dezactivarea lui BUSY trebuie să fie dezactivat și nACK. Unitatea centrală și echipamentul periferic sunt acum pregătite pentru transferarea unui alt Byte.

Aceste faze sunt ilustrate de diagrama de semnal din Fig. 4.1.

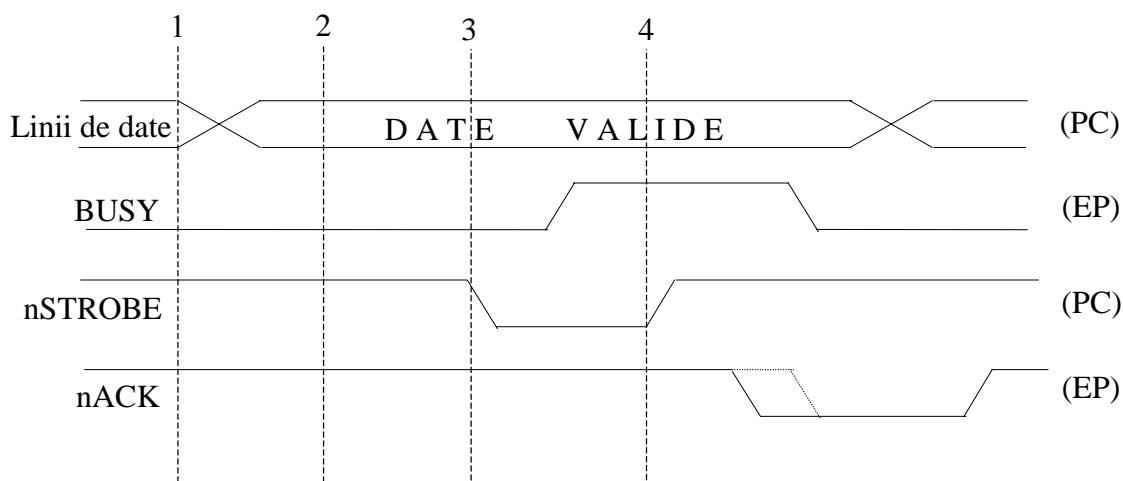


Fig. 4.1. Diagrama unui transfer de date în modul compatibil

Fazele transferului, marcate în diagramă 1, 2, 3, 4 au următoarea interpretare:

1. Înscrierea registrului de date.
2. Programul verifică registrul de stare pentru a se asigura că imprimanta nu este BUSY.
3. Dacă imprimanta nu este BUSY, programul înscrie registrul de comenzi pentru a genera STROBE pe linia de comandă.
4. Se înscrie registrul de comenzi pentru dezactivarea liniei STROBE.

Procedura de transfer descrisă este numită transfer cu protocol pe trei fire. Există și transfer cu protocol pe două fire în care nu se consideră BUSY.

Pentru a transfera un byte de date sunt necesare 4 instrucțiuni I/O. Limitările impuse de timpul consumat în acest fel nu deranjează în cazul imprimantelor obișnuite, dar este o problemă în cazul altor echipamente. Rațiunea pentru care s-a definit acest mod este păstrarea compatibilității cu vechiul port paralel.

Multe circuite controler I/O -1284 au implementat cu ajutorul unui registru FIFO un mod numit "Fast Centronics" sau "Parallel Port FIFO Mode". Când acest mod este activat, datele înscrise în FIFO vor fi transferate imprimantei cu strobe și handshake generate de un hardware propriu. Se pot atinge rate de transfer de peste 500 KBps.

4.3.2. Modul Nibble

Modul Nibble este cea mai simplă soluție de a transfera date de la periferic la PC. Combinând acest mod cu modul compatibil se pot transfera date bidirecțional. Toate porturile paralele standard existente dispun de 5 linii care transportă semnal electric de la periferic spre PC pentru a fi folosite ca indicator de stare a perifericului. Folosind aceste linii, un periferic poate transmite un byte în două cicluri de transfer, câte 4 biți într-un ciclu. Un grup de 4 biți este numit nibble. Numele și definiția semnalelor de interfață se schimbă față de modul compatibil. Corespondența și noua semnificație a semnalelor este prezentată în Tabelul 4.6.

Tabel 4.6. Semnale de interfață în mod nibble

| Semnal SPP | Semnal Nibble | In/Out | Descriere în mod nibble |
|------------|---------------|--------|--|
| nSTROBE | nSTROBE | Out | Nu este folosit în transfer reverse. |
| nAUTOFEED | HostBusy | Out | Semnal handshake în mod nibble. LOW - indica PC este pregătit pt. nibble. HIGH - confirmă recepția unui grup de 4 biți (nibble). |
| nSELECTIN | 1284Active | Out | HIGH - PC este în unul din modurile 1284. |
| nINIT | nINIT | Out | Nu este folosit în transfer reverse. |
| nACK | PtrClk | In | LOW - indică date nibble valide. HIGH - ca răspuns la tranziția HIGH a HostBusy. |
| BUSY | PtrBusy | In | Folosit pentru bitul 3 respectiv bitul 7 la pasul 2. |
| PE | AckDataReq | In | Folosit pentru bitul 2 respectiv bitul 6 la pasul 2. |
| SELECT | Xflag | In | Folosit pentru bitul 1 respectiv bitul 5 la pasul 2. |
| nERROR | nDataAvail | In | Folosit pentru bitul 0 respectiv bitul 4 la pasul 2. |
| DATA [8:1] | Nefolosit | - | - |

La inițializare, un sistem este programat în mod compatibil (d.p.d.v. al portului paralel).

Un transfer în mod nibble parcurge două faze: *Reverse idle phase* și *Reverse data phase*. *Reverse idle phase* este faza de intrare în mod nibble. Perifericul atenționează unitatea centrală că dorește să transfere date în mod reverse (de la periferic la calculator) cu semnalul PtrClk, generând o întrerupere. Calculatorul răspunde cu HostBusy și prin aceasta se trece la faza a doua.

În faza a doua se desfășoară transferul efectiv de date. HostBusy comută HIGH, iar dacă există date valide de la periferic, HostBusy comută LOW. Comutarea LOW a lui HostBusy, fără ca să existe date valide de la periferic, are ca rezultat intrarea în *Reverse idle phase*.

Un ciclu pentru un transfer nibble este reprezentat în Fig. 4.2 .

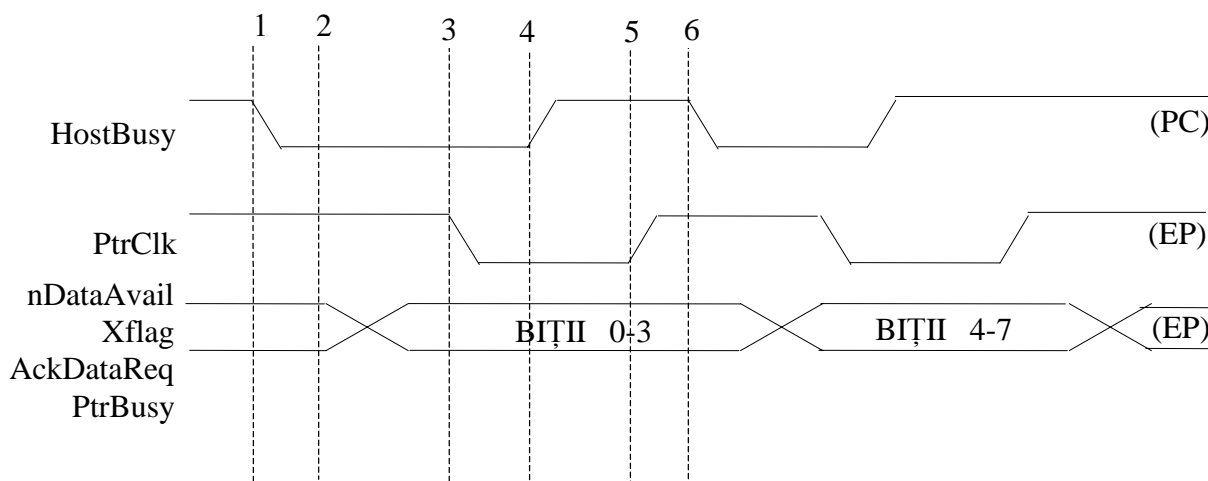


Fig. 4.2. Diagrama unui transfer de date în modul nibble

Semnificația momentelor marcate pe diagramă cu 1, 2, 3, 4, 5, 6 este:

1. PC-ul anunță disponibilitatea de a prelua date (HostBusy trece în LOW).
2. Răspunsul perifericului prin plasarea primului grup de 4 biți pe liniile de stare (dedicate acum transferului de date).
3. Perifericul validează datele trimise pe liniile de stare (PtrClk trece LOW).
4. PC-ul anunță că a recepționat grupul de 4 biți și nu este încă pregătit pentru următorul grup (HostBusy trece HIGH).
5. Perifericul anunță că a sesizat confirmarea recepției.
6. Se repetă stările 1-5 pentru al doilea grup de 4 biți.

Biții 0-3 și 4-7 sunt transferați conform Tabelului 4.7.

Tabel 4.7. Ordinea de transferare a biților în mod nibble

| Semnal | Biții 1-4 (Ciclu I) | Biții 5-8 (Ciclu II) |
|------------|----------------------|----------------------|
| nDataAvail | D ₀ (LSB) | D ₄ |
| Xflag | D ₁ | D ₅ |
| AckDataReq | D ₂ | D ₆ |
| PtrBusy | D ₃ | D ₇ (MSB) |

Ca și în modul compatibil, în modul nibble, protocolul este condus prin program (soft) prin setarea și citirea liniilor de comandă, respectiv de stare. Modul nibble este mai lent decât modul compatibil, rata de transfer este de aproximativ 50 KBps. Principalul avantaj al acestui mod este că poate fi folosit pentru transfer de date de la periferic la PC (reverse) pe orice calculator.

Deoarece este o extensie a modului compatibil, modul nibble folosește aceleași registre, doar denumirea și funcția semnalelor sunt schimbate. Modul de acționare și rangul ce corespunde fiecărui semnal sunt date în Tabelul 4.8.

Tabel 4.8. Registre în mod nibble

| Offset | Nume | mod de operare | Descriere |
|--------|----------------------------|----------------|-----------------------------------|
| 1 | Registru de stare | | Conține biți de stare; R |
| | DB ₃ | inversat | nDataAvail |
| | DB ₄ | neinversat | Xflag |
| | DB ₅ | neinversat | AckDataReq |
| | DB ₆ | inversat | PtrClk |
| | DB ₇ | inversat | PtrBusy |
| 2 | Registru de control | | Conține biți de control; W |
| | DB ₁ | neinversat | HostBusy |
| | DB ₃ | neinversat | 1284Active |
| | DB ₄ | neinversat | IRQE |

Inversat și neinversat au aceeași semnificație ca și în modul standard.
În mod nibble nu se lucrează cu registrul de date.

4.3.3. Modul Byte

La dezvoltarea sistemelor PS/2 de către IBM, portului paralel i s-a adăugat o nouă facilități; liniile de date au fost comandate prin circuite TSL bidirecționale, permițând portului de date să fie folosit și ca port de intrare. Circuitele TSL sunt comandate în starea de înaltă impedanță, și atunci orice nivel pe liniile de date poate fi citit efectiv prin registrul de întoarcere (facilitate implementată inițial pentru a verifica dacă datele au ajuns pe liniile de ieșire). Spre deosebire de modul nibble, modul Byte presupune un hardware special, nu orice interfață poate lucra în mod Byte.

Noua structură, care nu o exclude pe cea veche, face posibil ca un periferic să poată trimite un byte PC-ului într-un singur ciclu de transfer, față de două cicluri necesare în modul nibble. Rata de transfer pentru transfer reverse (de la periferic la PC) este apropiată de rata de transfer în mod compatibil. Acest mod de transfer mai este numit și “enhanced bi-directional” și nu trebuie confundat cu modul EPP (Enhanced Parallel Port).

Tabelul 4.9. conține corespondența semnalelor raportată la modul compatibil, noua denumire și descriere a semnalelor.

Tabel 4.9. Semnale de interfață în mod byte

| Semnal SPP | Semnal Byte | In/Out | Descriere în mod byte |
|------------|-------------|--------|---|
| nSTROBE | HostClk | Out | Impuls Low - este semnal de confirmare. PC-ul anunță că a sesizat anunțul perifericului că la rândul său a sesizat confirmarea de recepție a byte-lui de către PC. |
| nAUTOFEED | HostBusy | Out | LOW - PC-ul este pregătit să preia un byte. HIGH - PC-ul a recepționat byte-ul. |
| nSELECTIN | 1284Active | Out | HIGH - PC-ul este în unul din modurile 1284. |
| nINIT | nINIT | Out | Nefolositor. Este HIGH tot timpul. |
| nACK | PtrClk | In | LOW - există date valide pe liniile de date. HIGH - ca răspuns a tranziției în HIGH a lui HostBusy |
| BUSY | PtrBusy | In | Stare Busy a canalului forward (PC-periferic). |

| Semnal SPP | Semnal Byte | In/Out | Descriere în mod byte |
|------------|-------------|--------|--|
| PE | AckDataReq | In | Urmărește semnalul nDataAvail |
| SELECT | Xflag | In | Fanion pt. extensii. Nefolosit în modul byte. |
| nERROR | nDataAvail | In | LOW - perifericul anunță că sunt valide date pe canalul reverse. |
| DATA [8:1] | DATA [8:1] | Bi-Di | Vehiculează date de la periferic spre PC. |

Fig. 4.3. redă diagrama de semnale pentru un ciclu de transferare a unui byte de la periferic la calculator în mod byte.

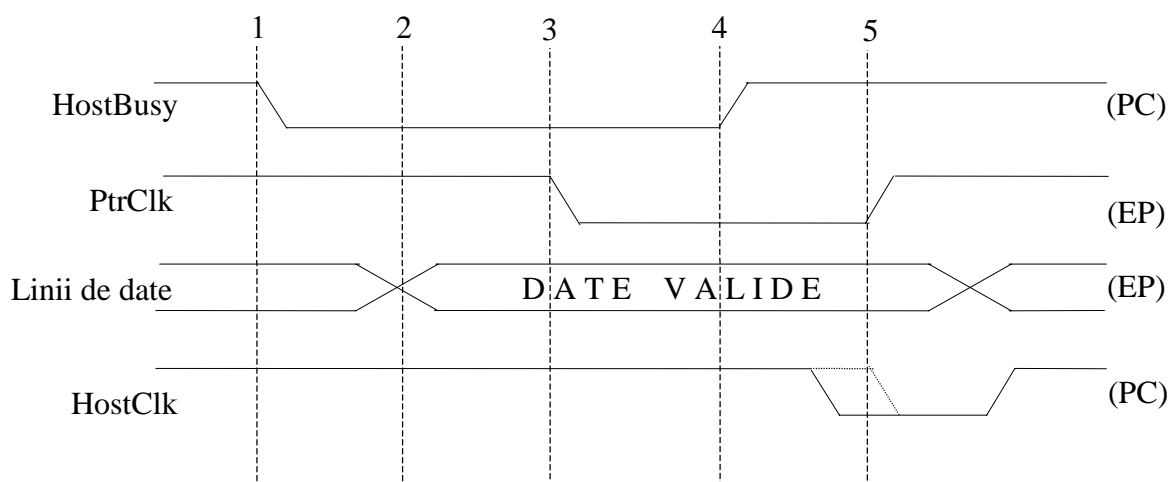


Fig.4.3. Diagrama unui transfer de date în modul byte

Semnificația momentelor marcate pe diagramă cu 1, 2, 3, 4, 5 este următoarea:

1. PC-ul anunță disponibilitatea de a prelua date (HostBusy trece LOW).
2. Perifericul răspunde prin plasarea unui byte pe liniile de date.
3. Perifericul anunță că byte-ul de pe liniile de date poate fi citit, este valid (PtrClk trece LOW).
4. PC-ul a citit datele și nu este gata pentru o nouă citire (HostBusy trece HIGH).
5. Perifericul comută PtrClk HIGH pentru a confirma PC-ului că i-a sesizat starea. Totodată PC-ul generează un impuls LOW pe HostClk ca răspuns spre imprimantă.

Pentru următorul byte se repetă pașii 1-5.

Ca și modul nibble, modul byte este o extensie a modului standard, deci folosește același set de registre pentru a stabili semnalele de interfață. Adresele din spațiul I/O rămân aceleași ca și în mod standard. Tabelul 4.10. conține alocarea fiecărui semnal în cadrul registrelor și modul de acționare.

Tabel 4.10 Registrele în mod byte

| Offset | Nume | mod de operare | Descriere |
|--------|-----------------------------------|----------------|---------------------------------|
| 0 | Registru de date | | Conține biți de date |
| | DB ₁ - DB ₈ | neinversat | D ₁ - D ₈ |
| 1 | Registru de stare | | Conține biți de stare; R |
| | DB ₃ | inversat | nDataAvail |
| | DB ₄ | neinversat | Xflag |
| | DB ₅ | neinversat | AckDataReq |
| | DB ₆ | inversat | PtrClk |
| | DB ₇ | inversat | PtrBusy |
| 2 | Registru de control | | Conține biți de control; W |
| | DB ₀ | neinversat | HostClk |
| | DB ₁ | neinversat | HostBusy |
| | DB ₃ | neinversat | 1284Active |
| | DB ₄ | neinversat | IRQE |
| | DB ₅ | neinversat | Sensul datelor |

Sensul transferului pe liniile de date este stabilit prin valoarea bitului D₅ din registrul de control.

$$\begin{aligned} D_5 = 1 & \quad \text{mod SPP} \\ D_5 = 0 & \quad \text{mod byte} \end{aligned}$$

4.3.4. Modul EPP

Enhanced Parallel Port a fost dezvoltat și adoptat inițial de un grup de producători (Intel, Xircom și Data Systems) ca o soluție pentru o legătură paralelă performantă, fără a neglija compatibilitatea cu portul paralel standard. Datorită avantajelor sale a fost adoptat și de alți producători ca o metodă adițională pentru transferarea datelor și ulterior a fost inclus în standardul IEEE-1284 ca unul din modurile avansate de transfer paralel. Datorită acestui curs de desfășurare a evenimentelor există mici diferențe între portul EPP inițial și protocolul EPP reglementat prin standardul IEEE-1284.

Protocolul EPP definește 4 tipuri de transferuri de date:

- ciclu de scriere de data (Data Write Cycle);
- ciclu de citire de date (Data Read Cycle);
- ciclu de scriere de adrese (Address Write Cycle);
- ciclu de citire de adrese (Address Read Cycle).

În cadrul ciclurilor de date se transferă date între PC și periferic, iar ciclurile de adrese vehiculează adrese sau informații de comandă și control.

Tabelul 4.11. conține simbolurile și definițiile semnalelor EPP.

Tabel 4.11. Semnale de interfață în modul EPP

| Semnal SPP | Semnal EPP | In/Out | Descriere în mod EPP |
|------------|-----------------|--------|--|
| nSTROBE | nWRITE | Out | LOW - indică o operație de scriere. HIGH - indică un ciclu de citire. |
| nAUTOFEED | nDATASTB | Out | LOW - indică faptul că este în curs de desfășurare o operație de scriere sau citire de date. |
| nSELECTIN | nADDRSTB | Out | LOW - indică faptul că este în curs de desfășurare o operație de scriere sau citire de adrese. |
| nINIT | nRESET | Out | LOW - resetează perifericul. |
| nACK | nINTR | In | LOW - întrerupere de la periferic. |
| BUSY | nWAIT | In | Semnal de handshake. LOW - indică PC-ului că poate începe un ciclu (să activeze Strobe). HIGH - indică PC-ului că se poate încheia un ciclu (să dezactiveze Strobe). |
| DATA [8:1] | AD [8:1] | Bi-Di | Linii bidirecționale pentru vehicularea datelor sau a adreselor. |
| PE | def. utilizator | In | Folosit de periferic conform definiției particulare. |
| SELECT | def. utilizator | In | Folosit de periferic conform definiției particulare. |
| nERROR | def. utilizator | In | Folosit de periferic conform definiției particulare. |

În mod EPP sunt folosite 8 linii de date și 6 semnale pentru controlul transferului pe liniile de date. Adresele corespund unui registru aflat la periferic Adresa stabilește cui îi sunt destinate datele transferate pe liniile de date.

Semnalul nWRITE exprimă sensul în care se transferă datele. Pentru semnal activ, (nivel LOW), sensul de transfer este forward (de la calculator la periferic). Pentru semnal inactiv (nivel HIGH) datele sunt transferate în sens reverse, de la periferic la calculator.

Semnalul nDATASTB are aceeași funcție ca și semnalul STROBE la modul compatibil. Semnalul nADDRSTB validează adresele aflate pe liniile de date ale interfeței.

Când semnalul nWAIT este activ, nivel LOW, perifericul este gata pentru un transfer de date. Când semnalul este inactiv, perifericul nu este gata pentru transfer de date, deci calculatorul trebuie să aștepte.

La activarea semnalului nRESET sistemul este inițializat, deci iese din mod EPP și revine în mod SPP.

Prin semnalul nINTR perifericul semnalizează o cerere de întrerupere.

În Fig. 4.4. este reprezentată diagrama de semnal a unui ciclu de scriere de date. În diagramă este reprezentat și semnalul nIOW al UCP pentru a sublinia că întreg transferul are loc într-un singur ciclu I/O.

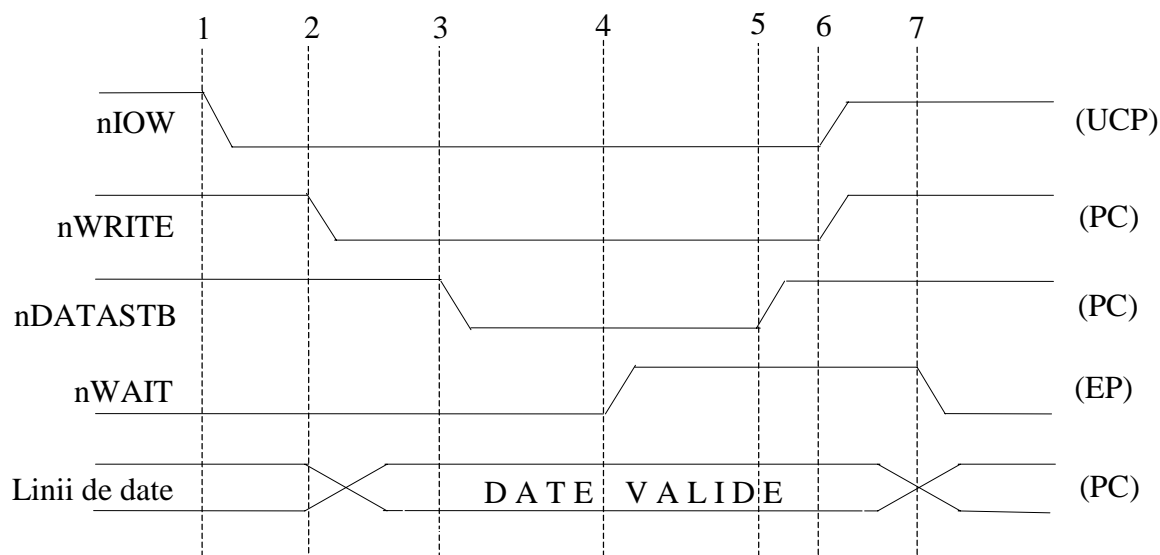


Fig. 4.4. Diagrama unui ciclu de scriere de date în modul EPP

Momentele ciclului marcate în diagramă 1, 2, 3, 4, 5, 6, 7 au următoarea semnificație:

1. Programul execută o comandă I/O de scriere la portul de date EPP (port 4).
2. Se activează linia de comandă nWRITE și sunt trecute datele pe liniile de date.
3. Deoarece nWAIT este LOW, se generează nDataStrobe activ.
4. Portul așteaptă confirmarea de la periferic (dezactivarea nWAIT).
5. Se dezactivează nDataStrobe și ciclul EPP se încheie.
6. Se încheie ciclul I/O al magistralei de sistem.
7. Se comută nWAIT LOW pentru a face posibilă începerea unui nou ciclu.

Cea mai importantă particularitate a acestui mod este că întregul transfer are loc în cadrul unui ciclu I/O al unității centrale, ceea ce face ca rata de transfer posibilă în acest mod să poate fi cuprinsă între 500 Kbps și 2 Mbps. Ratele de transfer sunt comparabile cu cele atinse de un modul conectat direct la o magistrală ISA.

Un ciclu de citire de adrese se desfășoară conform diagramei de semnal din Fig. 4.5.

Protocolul este integral comandat de calculator, terminalul poate doar să valideze sau să confirme un transfer de date.

Protocolurile pre-1284 EPP diferă de protocolul 1284 EPP prin faptul că perifericul nu poate bloca începerea unui transfer prin semnalul nWAIT. În modul pre-1284 EPP nDATASTB sau nADDRSTB pot fi activate (tranzitate LOW) la începutul unui ciclu fără a ține cont de starea semnalului nWAIT. Modul pre-1284 este cunoscut și ca EPP 1.7, după versiunea Xircon 1.7. Un periferic 1284 EPP va lucra corect cu un PC cu adaptor EPP 1.7, dar este posibil ca un periferic EPP 1.7 să nu lucreze corect cu un PC cu adaptor 1284 EPP.

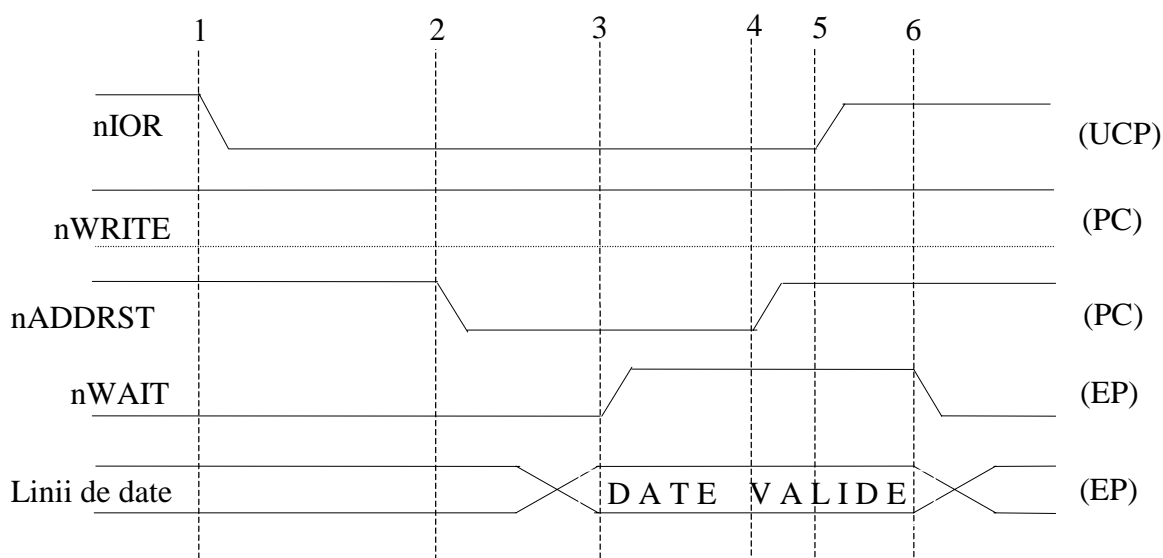


Fig. 4.5. Diagrama unui ciclu de citire de adrese în modul EPP

Performanța modului EPP este posibilă prin extinderea numărului registrelor de interfață. Registrele portului în varianta inițială sunt păstrate ca semnificație și ca offset față de adresa de bază a portului. Lista registrelor folosite în mod EPP este redată de Tabelul 4.12

Tabel 4.12. Registre EPP

| Nume port | Offset | Mod | R/W | Descriere |
|---------------------|-----------|---------|-----|---|
| Port de date SPP | +0 | SPP/EPP | W | Portul standard de date. |
| Port de stare SPP | +1 | SPP/EPP | R | Citește liniile de stare ale interfeței. |
| Port de control SPP | +2 | SPP/EPP | W | Stabilește starea liniilor de control. |
| Port de adrese EPP | +3 | EPP | R/W | Generează un ciclu de citire sau scriere de adrese. |
| Port de date EPP | +4 | EPP | R/W | Generează un ciclu de citire sau scriere de date. |
| Nedefinit | +5,+6, +7 | EPP | N/A | Definiție proprie aplicației particulare. |

O instrucțiune I/O la adresa de bază sau la adresele [adresa_de_bază+1], [adresa_de_bază+2], va determina funcționarea în modul SPP; se realizează astfel compatibilitatea cu perifericele construite pentru a fi conectate la portul paralel standard. O instrucțiune I/O la portul ce are adresa [adresa_de_bază+4] va activa controlerul EPP, care va genera semnalele de comandă și de dialog necesare realizării transferului de date. Pentru a declanșa un ciclu de read/write adrese, este necesară execuția unei instrucțiuni I/O la portul cu adresa [adresa_de_bază+3].

Porturile cu offset +5, +6, +7 sunt folosite diferit de aplicațiile hardware. Pot fi folosite pentru a implementa interfețe software pe 16 sau 32 de biți, pot fi folosite ca registre de configurare sau se poate să nu fie folosite deloc. Există interfețe care mapează 4 porturi I/O pentru registrul de date EPP. Transferul de date se face pe 32 de biți din punct de vedere al utilizatorului. Practic, controlerul portului paralel va genera prin hardware-ul de care dispune, 4 cicluri I/O - câte unul pentru fiecare grupă de 8 biți:

- primul ciclu va transfera biții mai puțin semnificativi (0-7) asociați portului cu adresa [adresa_de_bază+4],
- al doilea ciclu va transfera biții 8-15 asociați portului cu adresa [adresa_de_bază+5],
- al treilea ciclu va transfera biții 16-23 asociați portului cu adresa [adresa_de_bază+6] și
- al patrulea și ultimul ciclu va transfera biții cei mai semnificativi (24-31) asociați portului cu adresa [adresa_de_bază+7].

Prin acest mecanism implementat hard în interfață se obține o viteză de transfer de 10 ori mai bună, în medie.

Ciclurile de adrese sunt în continuare limitate la transferuri pe 8 biți.

Facilitatea de a transfera date la sau de la PC prin executarea unei singure instrucțiuni (IN sau OUT) face posibil transferul datelor pe la portul paralel la viteza magistralei. Depinzând de tipul particular de adaptor paralel și de periferic, rata de transfer poate atinge 2 MBps sau la limita inferioară, 500 KBps.

Modul EPP este susținut de rutine BIOS

Versatilitatea conectării în mod EPP face ca această soluție să fie potrivită pentru periferice programabile la nivel de registru; astfel de periferice sunt adaptoarele de rețea, modulele de achiziție de date, hard-discuri portabile, ș.a.

4.3.5. Modul ECP

Protocolul Extended Capability Port, sau ECP, a fost propus de Hewlett Packard și Microsoft ca un mod evoluat de comunicație cu periferice de imprimare sau scanare.

Protocolul ECP definește două tipuri de transferuri de date, valabile atât pentru sensul direct (forward) cât și pentru sensul invers (reverse):

- Ciclu de date (Data cycle);
- Ciclu de comandă (Command cycle).

Ciclurile de comandă sunt la rândul lor de două categorii:

- Run-Length Count
- Channel address.

În mod ECP adaptorul are capacitatea de a face compresie de date, RLE (Run_Length_Encoding), sau de a utiliza de registre FIFO pentru ambele canale (forward și reverse) sau posibilitatea de a transfera date în mod DMA/programat, însușiri care stau la baza vitezelor mari de transfer ce pot fi atinse în mod ECP.

Pentru compresia datelor este folosită metoda RLE. Comprimatele acționează în cazul trimiterii unui caracter de mai multe ori succesiv; în acest caz se transmite caracterul urmat de numărul de apariții a caracterului respectiv.

Exemplu

șir de caractere: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAACC
se transmite: A[33]CC

Cu compresie de date în timp real -compresie de tip RLE- se pot atinge rapoarte de comprimare de 64:1, ceea ce este eficient mai ales în cazul imprimantelor sau scanner-elor care vehiculează informație ce permite o comprimare bună. Pentru a fi exploatată capacitatea de a lucra cu compresie de date, atât adaptorul paralel din PC cât și perifericul trebuie să aibe implementată această facilitare.

Adresarea unui canal este conceptual diferită de adresarea în mod EPP. Este posibilă adresarea mai multor dispozitive logice în cadrul unui singur dispozitiv fizic. Un simplu exemplu justifică utilitatea noului concept de adresare: un dispozitiv Fax/Modem/Printer atașat la portul paralel. Trei dispozitive sunt atașate printr-un singur conector la portul paralel. Prin adresarea canalelor specifică ECP, se poate ca în timp ce canalul de date al imprimantei este ocupat cu procesarea unei imagini pentru imprimare, de la modem să recepționăm date.

Ca și pentru celelalte moduri 1284, protocolul ECP redefineste semnalele SPP. Semnalele ECP sprijină un transfer de date bidirecțional fără intervenția unității centrale. Noile nume și semnificații, raportate la semnalele SPP, sunt prezentate în Tabelul 4.13.

Tabel 4.13. Semnale de interfață în modul ECP

| Semnal SPP | Semnal ECP | In/Out | Descriere în mod ECP |
|------------|-----------------|--------|--|
| nSTROBE | HostClk | Out | Folosit împreună cu PeriphAck pentru a transfera date sau adrese în sens forward. |
| nAUTOFEED | HostAck | Out | În transfer forward - stabilește ciclu date/comenzi. În transfer reverse - folosit împreună cu PeriphAck pentru a transfera date. |
| nSELECTIN | 1284Active | Out | HIGH - PC-ul este într-un mod 1284 |
| nINIT | nReverseRequest | Out | Comută LOW pentru a stabili sensul reverse. |
| nACK | PeriphClk | In | Folosit împreună cu HostAck pentru a transfera date în sens reverse. |
| BUSY | PeriphAck | In | În transfer forward - folosit împreună cu HostClk pentru a transfera date sau adrese. În transfer reverse - stabilește ciclu date/comenzi |
| PE | nAckReverse | In | Comută LOW pentru a confirma sesizarea cererii nReverseRequest. |
| SELECT | Xflag | In | Fanion pentru extensii. |
| nERROR | nPeriphRequest | In | Comutat LOW de către periferic pentru a semnaliza că datele pentru transfer reverse sunt valide. |
| DATA [8:1] | Data [8:1] | Bi-Di | Vehiculează date între PC și periferic. |

Modul ECP operează cu 8 linii de date și 7 semnale care sprijină transferul de date.

La pornire interfața operează în mod compatibil. Pentru a trece în mod ECP se negociază întâi această tranziție.

Semnalul HostClk comută HIGH, ceea ce înseamnă că PC-ul a trimis date pe liniile de date ale interfeței și acestea sunt valide.

Transferul nu începe până nu vine acordul perifericului prin comutarea HIGH a semnalului PeriphAck. După ce citește datele, perifericul comută LOW semnalul PeriphAck. Prin această acțiune, ambele semnale HostClk și PeriphAck sunt pregătite pentru un nou transfer.

nPeriphRequest este comutat LOW de către periferic atunci când dorește să transmită date la calculator. Activarea semnalului determină generarea unei întreruperi în calculatorul gazdă (doar acesta poate inversa un transfer de date).

nReverseRequest este un semnal prin care calculatorul anunță perifericul că îi este permis să trimită date unității centrale sau altui periferic.

nAckreverse este semnalul răspuns la nReverseRequest prin care perifericul spune că este pregătit.

PeriphClk exprimă faptul că datele puse de periferic pe liniile de date ale interfeței sunt valide.

HostAck este răspunsul la PeriphClk prin comutare High, comutare care atrage după sine și comutarea PeriphClk în stare HIGH. După ce calculatorul a recepționat date, HostAck comută LOW, ceea ce înseamnă că s-a încheiat transferul.

Data [8:1] sunt liniile de date ale interfeței. Acestea pot vehicula date sau comenzi. În transfer forward este folosit HostAck pentru a specifica dacă sunt date sau comenzi pe linii. În transfer reverse este folosit PeriphAck pentru a specifica dacă sunt date sau comenzi pe linii.

În Fig. 4.6. este redată diagrama de semnal a unui transfer forward, un ciclu de date urmat de un ciclu de comandă. Un transfer reverse este redat în următoarea figură, Fig. 4.7, de asemenea un ciclu de date urmat de un ciclu de comandă.

Conform definiției semnalelor în mod ECP, într-un transfer forward, dacă HostAck este HIGH, se desfășoară un ciclu de date. Dacă HostAck este LOW, se desfășoară un ciclu de comandă, iar informația de pe liniile de date reprezintă fie un număr RLE fie o adresă de canal. Mai precis, dacă MSB (bitul 8) este 0, atunci biții 1-7 reprezintă Run_Length_Count (0-127). Dacă MSB este 1, atunci biții 1-7 reprezintă o adresă de canal (0-127).

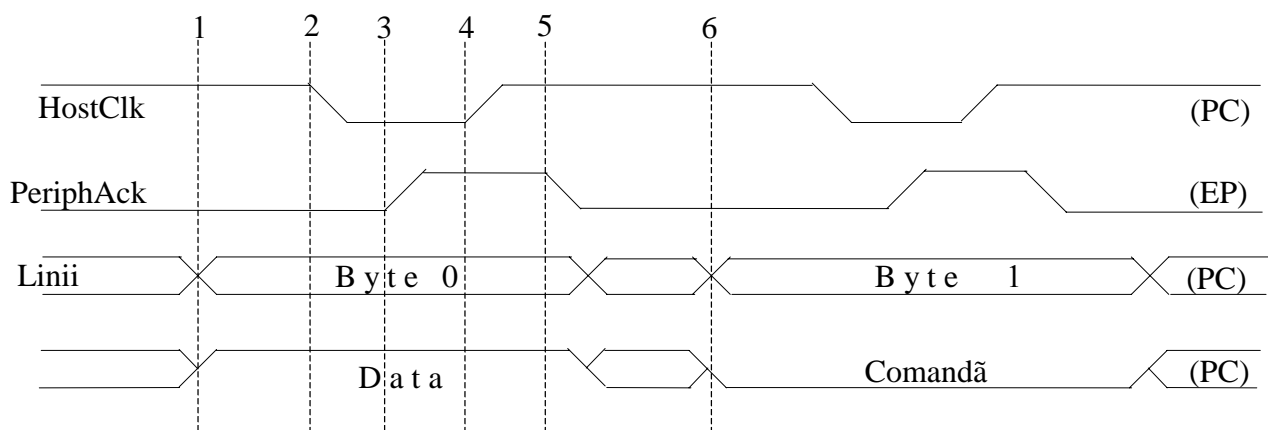


Fig. 4. 6. Diagramă de semnal pentru transfer ECP forward;
un ciclu de date și un ciclu de comandă

Momentele marcate în diagrama de semnal au următoarea semnificație:

1. PC-ul plasează date pe liniile de date și totodată semnalizează că urmează un ciclu de date prin comutare HIGH a semnalului HostAck.
2. PC-ul comută linia HostClk LOW pentru a indica date valide pe liniile de date.
3. Perifericul comută PeriphAck HIGH pentru a confirma că a luat la cunoștință că datele sunt valide.
4. PC-ul comută HostClk HIGH. Acest front înscrie datele la periferic.
5. Perifericul comută PeriphAck LOW pentru a semnaliza că este pregătit pentru un nou ciclu.
6. Ciclul se repetă, dar de data aceasta este un ciclu de comandă pentru că HostAck este LOW.

Transferul reverse este descris în continuare de diagrama din Fig.4.7.

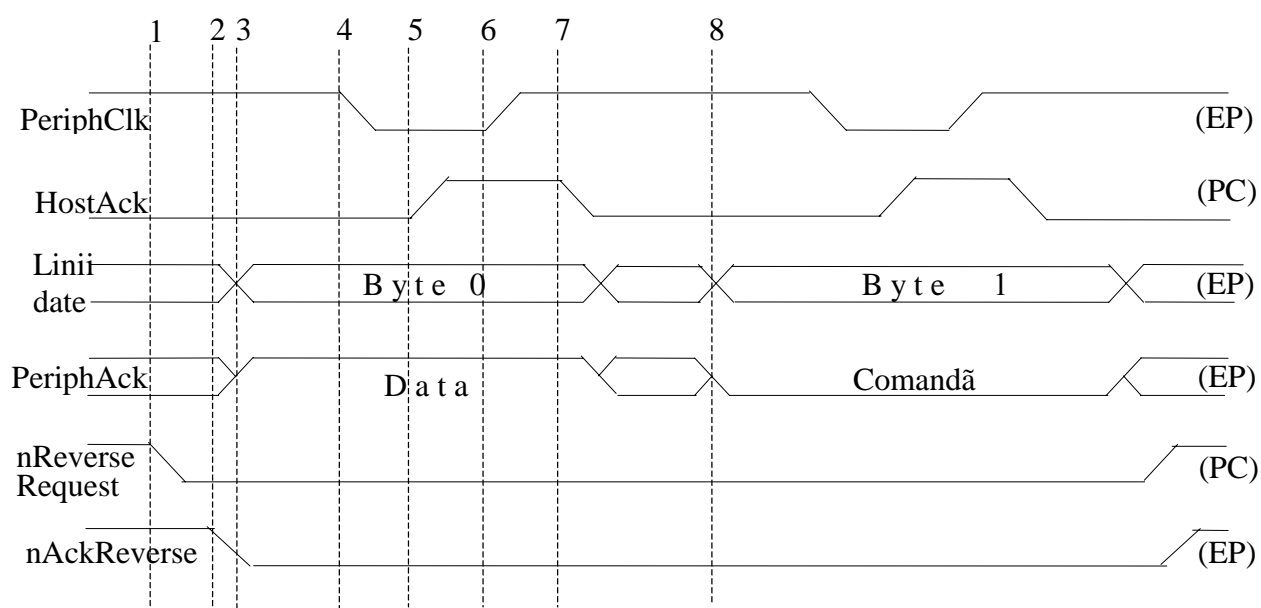


Fig. 4. 7. Diagramă de semnal pentru transfer ECP reverse;
un ciclu de date și un ciclu de comandă

Momentele marcate în diagrama de semnal definesc următoarele faze:

1. PC-ul solicită un canal de comunicație de tip reverse prin comutarea LOW a semnalului nReverseRequest.
2. Perifericul confirmă disponibilitatea de a transfera date spre PC prin comutarea LOW a semnalului nAckReverse.
3. Perifericul plasează date pe liniile de date și semnalizează că urmează un ciclu de date prin comutarea HIGH a semnalului PeriphAck.
4. Perifericul trece semnalul PeriphClk LOW pentru a anunța ca datele sunt valide.
5. PC-ul anunță că a luat la cunoștință că datele sunt valide prin comutarea lui HostAck HIGH.
6. Perifericul generează un front pozitiv la PeriphClk, front folosit pentru înscrierea datelor la PC.
7. PC-ul comută HostAck LOW pentru a semnaliza că este pregătit de un nou ciclu.
8. Ciclul se repetă, dar de data aceasta este un ciclu de comandă pentru că semnalul PeriphAck este LOW.

Spre deosebire de modul EPP, modul ECP comută între transfer forward și reverse doar în urma unei negocieri. PC-ul cere transfer reverse prin semnalul nReverseRequest și așteaptă acordul perifericului care răspunde cu semnalul nAckReverse. La modul EPP driver-ul software poate intercala transferuri forward cu transferuri reverse fără a mai negocia într-un protocol handshake.

Specificațiile Microsoft, “The IEEE 1284 Extended Capabilities Port Protocol and ISA Interface Standard”, definesc un registru pentru adaptoare paralele ISA în care se stabilește modul de operare al adaptorului. Registrul este numit Extended Control Register (ECR). Posibilitățile sunt prezentate în Tabelul 4.14.

Tabel 4.14. Registrul ECR - moduri de operare

| Mod | Descriere |
|-----|-------------------------------|
| 000 | Mod SPP |
| 001 | Mod Byte |
| 010 | Fast Centronics (SPP cu FIFO) |
| 011 | Mod ECP |
| 100 | Mod EPP |
| 101 | rezervat |
| 110 | Mod test |
| 111 | Mod configurare |

Magistrala ISA are o particularitate în adresarea porturilor; se folosesc doar 10 biți de adresă, deci se pot apela doar 1024 de porturi (3FFh). Decodificând mai mulți biți se pot obține mai multe pagini de câte 1024 de adrese. Adăugând 400h se găsește o adresă din pagina următoare. Astfel, apelând adresele 378h și 778h se pot accesa două registre aflate pe pagini diferite cu garanția că 778h nu va apela un alt dipozitiv instalat pe magistrală. Modul ECP exploatează această particularitate și cu 3 adrese propriu-zise se apelează 6 registre. Numele, adresa și descrierea pe scurt a acestor registre este făcută în Tabelul 4.15.

Tabel 4.15. Descrierea registrelor ECP

| Offset | Nume | Read/Write | Mod ECP | Funcție |
|--------|-----------|------------|---------|--------------------------|
| 000 | Data | R/W | 000-001 | Registru de date |
| 000 | ecpAfifo | R/W | 011 | FIFO adrese ECP |
| 001 | dsr | R/W | toate | Registru de stare |
| 002 | dcr | R/W | toate | Registru de control |
| 400 | cFifo | R/W | 010 | FIFO date port paralel |
| 400 | ecpDfiffo | R/W | 011 | FIFO date ECP |
| 400 | tfifo | R/W | 110 | FIFO test |
| 400 | cnfgA | R | 111 | Registru configurare A |
| 401 | cnfgB | R/W | 111 | Registru configurare B |
| 402 | ecr | R/W | toate | Extende Control Register |

În cazul în care portul este SPP sau bidirecțional, primele 3 registre sunt tratate în mod standard.

Informații despre funcțiile registrelor ECP și definirea acestora la nivel de bit sunt accesibile fie în documentul Microsoft fie în foile de catalog ale producătorilor de controlere I/O cu port paralel ECP.

Utilizarea acestui mod este similară cu a modului EPP. Se înscrie modul de operare în registrul ECR, iar apoi transferul de date este realizat prin scrieri și citiri la portul adecvat. Tot protocolul conversațional este generat automat de controlerul de interfață prin resurse hardware.

În mod ECP interfața este prevăzută cu registre FIFO atât pentru date cât și pentru adrese, ceea ce îmbunătățește mult transferul sub aspect timp.

Modul ECP prezintă o însușire esențială care, pe lângă celelalte amintite, contribuie semnificativ la atingerea vitezelor mari de transfer; este vorba despre suportul pentru transfer DMA. În timpul unui transfer DMA datele din memorie sunt transferate într-un registru FIFO (sau invers) sub controlul circuitului controller-DMA.

4.4. INTERFAȚA ELECTRICĂ

Pentru portul paralel inițial nu erau prevăzute reglementări sub aspect electric. Nu exista nici o specificație care să caracterizeze circuitele emițătoare sau receptoare, terminatorii sau capacitățile de linie, astfel încât să poată fi garantată compatibilitatea între diferitele dispozitive.

Standardul 1284 definește două nivele de compatibilitate de interfață; Nivelul I și Nivelul II. Nivelul I este definit pentru dispozitive care nu urmează să opereze la viteze mari, dar care operează atât cu canale forward cât și cu canale reverse. Nivelul II este definit pentru dispozitive care vor opera în moduri avansate, cu cabluri lungi și cu rate de transfer ridicate.

Pentru Nivelul II, cerințele pentru circuitele emițătoare la nivel de conector sunt:

- Nivelul HIGH în gol să nu depășească +5.5V;
- Nivelul LOW în gol să nu fie mai mic decât -0.5V;
- $V_{OHmin} = 2.4V/14mA$;
- $V_{OLmax} = 0.4V/14mA$;
- $R_O = 50 \pm 5\Omega$;
- Slew rate = 0.05-0.40V/nS.

Pentru Nivelul II, cerințele pentru circuitele receptoare la nivel de conector sunt:

- Receptorul suportă vârfuri accidentale de -2.0V și +7.0V fără a opera greșit și fără a se defecta;
- $V_{IHmin} = 2.0V$;
- $V_{ILmax} = 0.8V$;
- $I_{IH} = 20nA$ la +2.0V;
- $I_{IL} = 20nA$ la +0.8V;
- Capacitatea circuitului să nu depășească 50pF.

În Fig. 4.8. este reprezentată o legătură emițător/receptor pentru Nivel II.

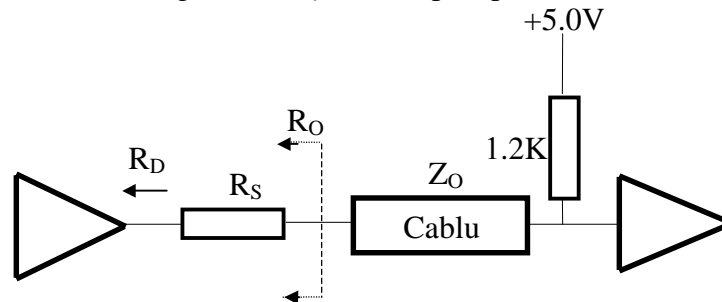


Fig. 4.8. Legătură emițător-receptor de Nivel II

R_O reprezintă impedanța de ieșire la conector. Pentru unele tipuri de emițătoare, pentru adaptare de impedanță, este necesară inserierea lui R_S .

În Fig. 4.9. este reprezentată legătura recomandată în cazul în care linia este bidirecțională (o linie de date). Circuitele de legătură spre cablu sunt transceivere.

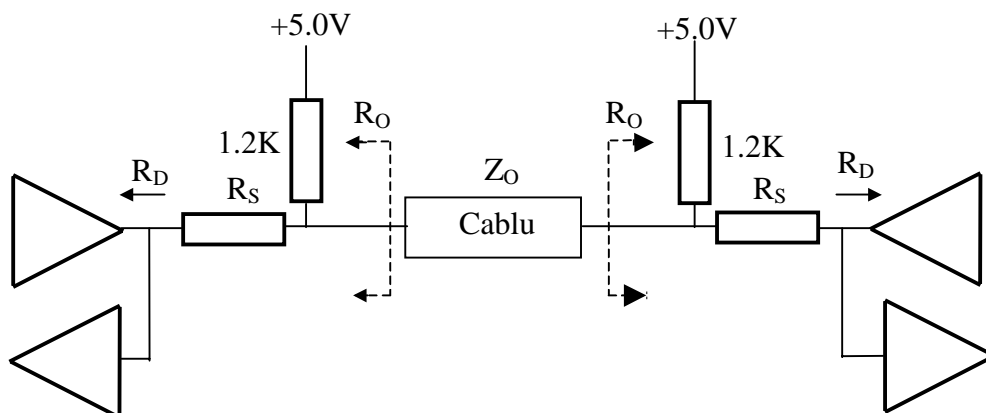


Fig. 4.9. Legătură bidirecțională de Nivel II

Inițial interfețele portului paralel erau realizate cu circuite L-TTL (74LS374 - octal latch). Acestea puteau să asigure la ieșire următorii curenți: $I_{OH} = 2.6 \text{ mA}$ și $I_{OL} = 24 \text{ mA}$. Cu circuite 74ACT374 se putea asigura $I_{OH} = 48 \text{ mA}$ și $I_{OL} = 64 \text{ mA}$.

Interfața la portul de control are ieșirile realizate cu circuite open collector inversor (SN7405) cu câte o rezistență de 4.7 K legată la $+5\text{V}$. Toate liniile în afară de D_2 sunt inversate, iar D_2 este dublu inversat (deci neinversat ca valoare). Registrul de întoarcere al portului de control poate fi citit, deoarece la ieșirile open collector putem forța cu alte semnale.

Se programează toate ieșirile HIGH. Orice HIGH conectat la acest nivel va pastra ieșirea circuitului. Orice LOW conectat la acest nivel va trage ieșirea open collector în LOW. Această stare poate fi citită prin registrul de întoarcere.

Open collector poate asigura $I_{OH} = 1 \text{ mA}$ și $I_{OL} = 7 \text{ mA}$.

4.5. INTERFAȚA MECANICĂ

Standardul 1284 prevede reglementări și pentru cabluri și conectică, asigurând astfel interoperabilitatea între configurații și periferice diferite. Lungimea cablului poate fi de 10m. Există un cablu paralel standard; acesta are la un capăt un conector mamă de tip DB25 și la celălalt capăt un conector Champ de 36 de pini (Centronics). În interior, cablul poate avea de la 18 la 25 fire conductoare, din care 1 până la 8 sunt fire de masă. Cablurile pot fi ecranate. Acest tip de cablu va opera la 10 Kbps cu o lungime de 1.8m, dar nu va opera la 2 Mbps cu o lungime de 9m.

Pentru rate de transfer mai mari, sunt restricții suplimentare legate de construcția cablului. Cablul pentru rate de transfer mari este obligatoriu inscripționat “*IEEE Std. 1284-1994 Compliant*”.

Lungimile standard pentru cablu sunt de 3m, 6m și 9m.

În ceea ce privește conectorii, și aceștia sunt definiți prin standardul 1284. Există trei tipuri: conector 1284 Tip A (DB25), 1284 Tip B (Centronics, 36 pini) și 1284 Tip C (Mini-Centronics, 36 pini). Este reglementată și poziționarea semnalelor la pinii conectorilor.