

11. Alte interfețe: IEEE1394, IrDA, SATA



Cuprins

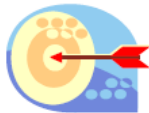
Cuprins modul

- 11.1. IEEE 1394
 - 11.1.1. Caracteristici și istoric
 - 11.1.2. Codificarea datelor
 - 11.1.3. Module IEEE 1394
- 11.2. Transferul de date în infraroșu IrDA
 - 11.2.1. Codificarea datelor
 - 11.2.2. Circuite IrDA
- 11.3. Interfața SATA
 - 11.3.1. Protocolul în interfața SATA



Introducere

Modulul “**Alte interfețe: IEEE1394, IrDA, SATA** “ cuprinde o prezentare generală a trei tipuri diferite de interfețe seriale. IEEE 1394, interfața pentru transmisia datelor în infraroșu IrDA și interfața pentru cuplarea hard discurilor SATA. Interfețele sunt prezentate pe rand, începând cu aplicabilitate, caracteristici și performanțe. Modulul este focalizat pe nivelul fizic și sunt prezentate modulele de codificare a datelor și protocolul de transfer.



Obiective

După parcurgerea acestui modul studenții vor înțelege:

- Legătura dintre partea teoretică parcursă în modulele anterioare și interfețele actuale;
- Diferențele practice dintre codificarea autosincronizabilă și cea neautosincronizabilă;
- Transmisia cu cadre de date, protocole de transfer cu cadre.

De asemenea studenții vor putea să mediteze asupra legăturii între codificare, performanță și aplicațiile acoperite de fiecare tip de interfață.

1. Învățarea unor tipuri de codare digitală. Înțelegerea caracteristicilor cerute codării
2. Cunoașterea unor tipuri de transmisii seriale și interfețe seriale ca structură, protocol și interfețe tipice
3. Învățarea unor interfețe tipice interne din microcontrollere
4. Înțelegerea noțiunilor prin exemplificări practice



Durata medie de studiu individual

Durata medie de studiu individual este de 2 ore.

11.1. IEEE 1394

11.1.1. Caracteristici și istoric

IEEE 1394 este o interfață serială cunoscută sub numele de FireWire (Apple), i.LINK (Sony) și LYNX (TI). Interfața a fost adoptată de HANA (High Definition Audio-Video Alliance) ca interfață standard disponibilă și wireless, pe fibră optică sau pe cablu coaxial. Dezvoltarea interfeței a început în 1980 și a fost încheiată în 1995. IEEE 1394 a fost aplicată și în aviația militară ca magistrală pentru F-22 Raptor și F-35. Navetele spațiale NASA au folosit IEEE 1394 pentru anumiți senzori. În industria auto a fost implementată o versiune numită IDB 1394. Cu toate că IEEE 1394 nu are răspândirea pe care o are USB, majoritatea camerelor digitale sunt echipate cu o astfel de interfață.

Ca și în majoritatea comunicațiilor seriale transferul de date este bazat pe pachete. Canalul comun de date este conceput să poată fi folosit pe rând de fiecare dispozitiv care îl solicită. Există un interval de timp specificat (numit *fairness interval*) în cadrul căruia un dispozitiv are accesul la canalul de date comun. După ce dispozitivul a trimis un pachet de date se așteaptă scurgerea unui timp de separare (numit *sub-action gap*) după care un alt dispozitiv poate trimite un pachet. Dacă după scurgerea timpului de separare nici un dispozitiv nu are de transmis vreun pachet, urmează o secvență de reset.

Pentru a face posibilă funcționarea dispozitivelor care necesită flux de date în timp real, IEEE-1394 folosește un mod special de transfer, modul izocron, ca și USB. Un dispozitiv ce necesită date izocronice emite la fiecare 125μs un pachet special de temporizare prin care asigură prioritatea transferului. Această schemă de arbitrară garantează un minim de buffer-e pentru date audio sau video (1 byte la dispozitive audio, până la 6 bytes la dispozitive video). Perioada de 125μs coincide cu perioada de eșantionare din sistemul de telefonie digitală, astfel interfața IEEE-1394 poate fi plasată în sistemul ISDN (Integrated Service Digital Network).

IEEE 1394 este asemănătoare cu USB, așa încât este utilă o comparație:

- La IEEE 1394 nu este nevoie de un calculator gazdă;
- IEEE 1394 asigură o viteză efectivă de transfer mai mare decât USB (dovedit pe sistemul de operare MAC OS X dar cu rezultate contradictorii sub Windows);
- Implementarea IEEE 1394 are costuri mai mari: licența Apple (0.25\$/sistem) și hardware mai scump cu 1-2\$;
- Ambele standarde pun la dispoziție prin cablul de transmisie de date o tensiune de alimentare, sunt plug and play și admit hot swapping. IEEE 1394 admite tronsoane de cablu de maximum 4.5m și poate alimenta o sarcină cu consum de până la 45W.
- Fiecare dispozitiv IEEE 1394 are un identificator propriu unic, (IEEE EUI-64) care este o adresă asemănătoare cu adresa MAC de 48 de biți.

În decursul timpului au fost realizate mai multe variante constructive:

1.FireWire 400 (IEEE 1394/1995). Versiunea originală poate transfera date cu viteze de 100, 200 sau 400 Mbps (S100, S200, S400) în mod half duplex. Modul de codificare al datelor este data strobe D/S;

2.FireWire 800 (IEEE 1394b/2002). Versiunea a doua asigură o viteză de 800Mbps în mod full duplex. Conectica este diferită față de varianta anterioară. Modul de codificare al datelor este 8B10B.

3.FireWire S800T (IEEE 1394c/2006). Versiunea a treia utilizează cablu Ethernet categoria 5e. Nu există încă implementări în sisteme disponibile pe piață datorită confuziei posibile la o placă de bază echipată cu 2 conectori RJ45, unul cu interfață Ethernet și unul IEEE 1394.

4.FireWire S1600 și S3200 Se lucrează la versiunile de 1.6Gbps și 3.2Gbps, care vor fi concurenți pentru USB 3.0. Conectorii sunt cei de la versiunea FireWire 800.

Se poate implementa o rețea de calculatoare prin legături IEEE 1394 în mod IPv4 sau IPv6. Sistemele de operare care include suport pentru acest tip de rețea sunt MAC OS X, Windows ME, 2000, XP și Server 2003. Windows Vista și Server 2008 nu mai conțin acest suport.

În figura 11.1 este dat un tabel cu conexiunile la conectorii IEEE 1394 cu 4, 6 și 9 pini și structura unui cablu IEEE 1394.

4 pini	6 pini	9 pini	Funcție	Descriere
	1	8	Vcc	30V nestabilizat
	2	6	Masă	Masă pentru tensiune
1	3	1	TPB-	semnal diferențial B
2	4	2	TPB+	semnal diferențial B
3	5	3	TPA-	semnal diferențial A
4	6	4	TPA+	semnal diferențial A
		5	ecran A	
		9	ecran B	

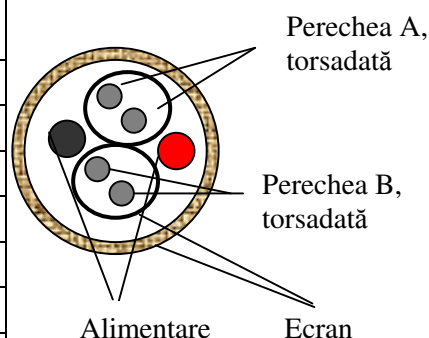


Figura 11.1. Tabel de conexiuni și structura cablului IEEE1394

11.1.2.Codificarea datelor Codificarea datelor D/S este de fapt o codificare NRZ cu transmiterea tactului și necesită 2 linii de semnal, una de date și una de strob. Un SAU Exclusiv între cele 2 semnale reconstituie tactul, figura 11.2. Pentru transmisia datelor este nevoie de ambele perechi FireWire, deci este posibil doar un transfer half duplex. Codificarea este aplicată la FireWire 400.

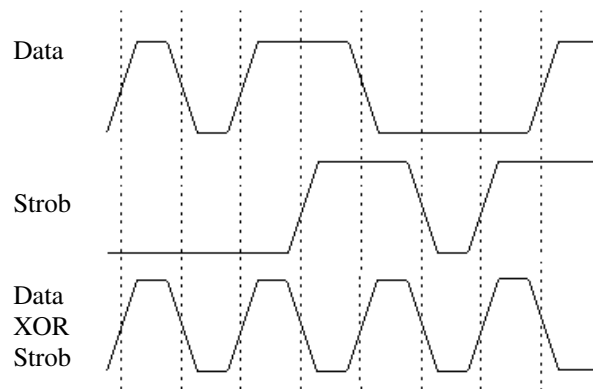


Figura 11.2. Codificarea Data /Strobe

Codificarea 8B10B a fost imaginată de Al. Widmer și P. Franaszek de la IBM în 1983 și IBM a obținut un patent. Răspândirea codificării a luat avânt după expirarea patentului. Aplicațiile dovedesc eficiența codificării: PCI Express, SATA, SAS, Fibre Channel, IEEE 1394b, Gigabit Ethernet (mai puțin la 1000BaseT), DVI, HDMI, USB 3.0 și seamănă cu codificarea folosită la CD (Eight to Fourteen Modulation). În principiu codificarea asigură o componentă DC mică pentru ca șirul de date să poată trece prin transformatorul de separare Ethernet, adică numărul de 0-uri este aproape egal cu numărul de 1-uri. Într-un șir de 20 biți diferența între numărul de 0 și de 1 poate fi maxim 2. Codul este autosincronizabil și se admit maxim 5 valori de 0 sau de 1 succesive. Codificarea atribuie la 8b o entitate de 10b numită simbol sau caracter. La 5b mai puțin semnificativi se atribuie 6b (porțiunea 5b/6b) iar la 3b mai semnificativi se atribuie 4b (porțiunea 3b/4b). Se definesc 12 simboluri speciale de control care marchează începutul cadrului, sfârșitul cadrului, skip, etc. Datorită codării cuvintelor de 8b cu simboluri de 10b anumite valori din cele 1024 pot fi excluse pentru a realiza condiția de a nu exista 5 valori de 0 sau de 1 consecutive. Pe linie se transmite întâi porțiunea 5b/6b apoi 3b/4b. Datele pot fi notate ca D.x.y unde x este porțiunea 5b/6b și poate fi 0-31 iar y este porțiunea 3b/4b și poate fi 0-7 ca valori necodate. Se definește RD (Running Disparity ca diferența între numărul de biți de 1 și numărul de biți de 0. Se urmărește obținerea RD cât mai mic. În acest scop grupurile 5b/6b și 3b/4b se stabilesc în funcție de RD anterior ca valori negate sau nenegate. De exemplu D.00 se codifică ca 100111 (RD inițial este -1 și rezultă RD=+1) sau 011000 (RD inițial este +1 și rezultă RD=-1) La fel, în funcție de RD inițial se codifică și grupul 3b/4b D.x.0 se codifică ca 1011 (RD inițial este -1 și rezultă RD=+1) sau 0100 RD inițial este +1 și rezultă RD=-1). Astfel în ipoteza RD inițial -1, D.00.0 se codifică ca 1001110100 și rezultă RD=0

Topologia unei arhitecturi IEEE 1394 este de tip stea multiplă (arbore) cu posibilitatea de înlănțuire (daisy-chain). În figura 11.3 sunt prezentate două spații de lucru unite cu un bridge. Cele 2 spații sunt izolate din punctul de vedere al traficului de date. Spațiul 1 de lucru ocupă mare parte a benzii din cauza traficului video, dar în spațiul de lucru

2 calculatorul are întregul control al traficului. Este posibil ca și calculatorul 2 să solicite date video, chiar dacă calculatorul 1 este oprit. Este figurat un repetor care mărește distanța de conectare și un splitter care adaugă 2 porturi unui port IEEE 1394.

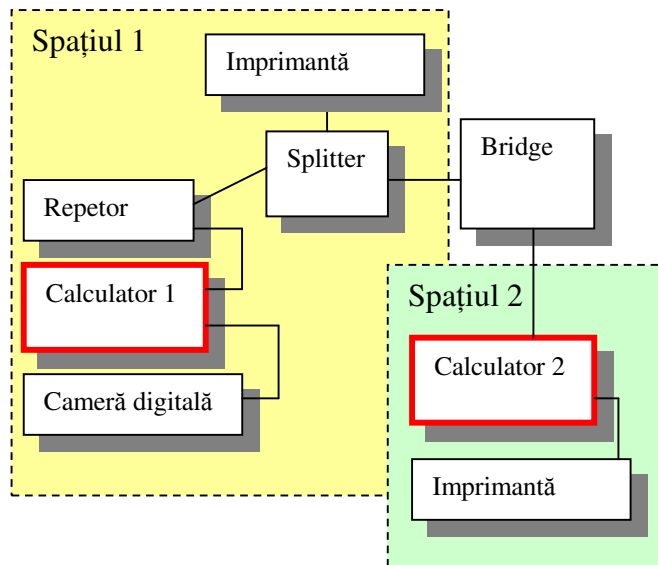


Figura 11.3. Topologia unei arhitecturi IEEE 1394

Pentru a transmite date în mod asincron dispozitivul IEEE 1394 compune un cadru care conține adresele sursei și destinației, apoi date și CRC. Când receptorul acceptă datele un cadru de confirmare este trimis la transmițător. Transmițătorul are posibilitatea să trimită încă 63 de cadre continuu pentru a mări viteza de transfer. Dacă cadrul de confirmare returnează o eroare se aplică o metodă de reacție la eroare.

În mod izocron emițătorul solicită un canal izocron iar dacă receptorul îl acceptă i se asigură un interval de timp de transfer pentru a asigura banda necesară transferului. Se pot defini până la 64 de canale izocrone. În exemplul din figura 11.4. în pachetul de date de 125μs sunt definite 2 intervale de timp pentru 2 transferuri izocrone. Timpul rămas liber se poate folosi la transferuri asincrone.

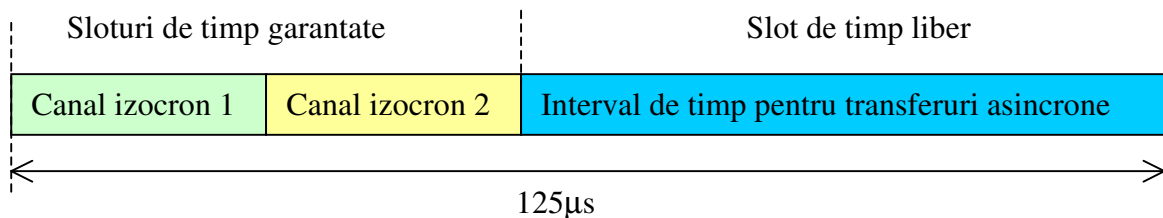


Figura 11.4. Transferuri izocrone pentru a asigura un flux de date în timp real

11.1.3. Circuite IEEE 1394

Nivelele ISO OSI (Open Systems Interconnection) în cazul IEEE 1394 sunt date simplificat în figura 11.5.

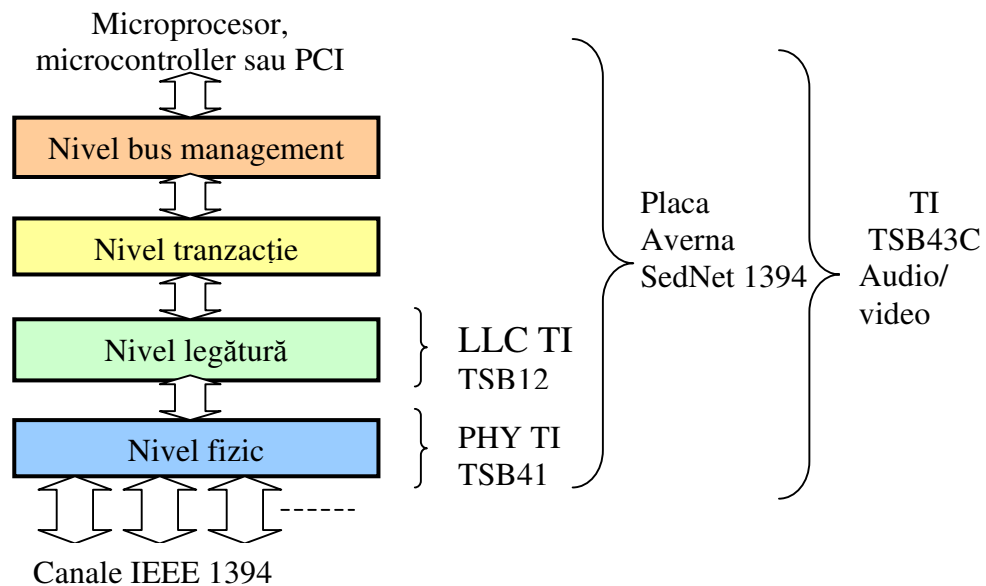


Figura 11.5. structura stivei OSI la IEEE 1394

Nivelul bus-management definește funcțiile de bază de control precum și registrele de control și de stare necesare dispozitivelor conectate pentru a-și face porturile operaționale. Acest nivel se ocupă și de asigurarea canalelor, de arbitrare, mastering și de erori.

Nivelul de tranzacție mediază operațiile de scriere și citire. Standardul permite la acest nivel operații cu cuvinte de lungime variabilă.

Nivelul de legătură realizează controlul logic în legătura IEEE 1394. Acest nivel realizează formarea cadrelor la transmisie și extragerea informației din cadrele recepționate.

Nivelul fizic presupune atât protocolul transferului cât și mediul efectiv de transfer. Partea de protocol controlează accesul la legătură, iar partea de mediu este constituită din cabluri și conectori. La acest nivel se realizează codarea și decodarea datelor, se asigură nivelele de tensiune necesare și se face arbitrarea magistralei.

Circuitele IEEE 1394 au un grad de complexitate mai mare decât cele USB, unul dintre motive fiind acela că pot lucra și independent de calculatorul gazdă. Multe circuite sunt realizate cu magistrală PCI pentru a fi utilizate în calculatoare PC. În figura 11.5. sunt date câteva exemple de circuite și plăci și nivelele OSI pe care le acoperă. A fost figurat un circuit care acoperă nivelul fizic (PHY Physical Layer Controller) și unul care acoperă nivelul de legătură (LLC Link Layer Controller). Unele circuite, așa cum este TSB43C de la Texas Instruments acoperă mai multe

nivele și încorporează blocuri de prelucrare audio video, un alt motiv pentru complexitatea mai mare a circuitelor. Placa SedNet de la Avera este un sistem de dezvoltare IEEE 1394 cu microcontroller Motorola și arată ca în figura 11.6:

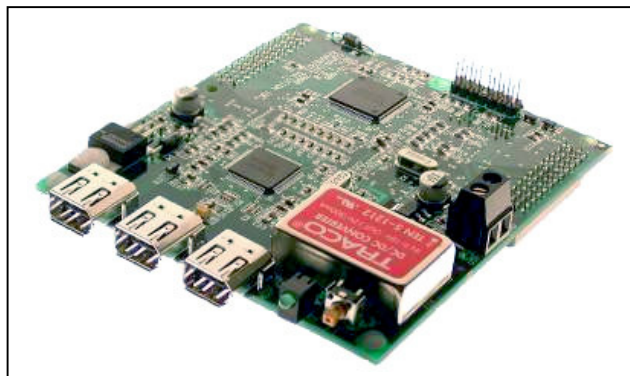


Figura 11.6. placa IEEE 1394 SedNet, sursa

ftp://ftp.adi.com/pub/adiuk/DDS%20V2.0/Avera_1394-OHCI_Boards.pdf

Acest sistem de dezvoltare este o soluție hardware și software completă pentru gestionarea unei comunicații IEEE 1394 între aplicația unui client care rulează pe un microcontroller care se conectează cu această placă prin intermediul unor linii de I/O sau o aplicație client care rulează pe microcontrollerul plăcii SedNet. Schema bloc a plăcii SedNet este dată în figura 11.7:

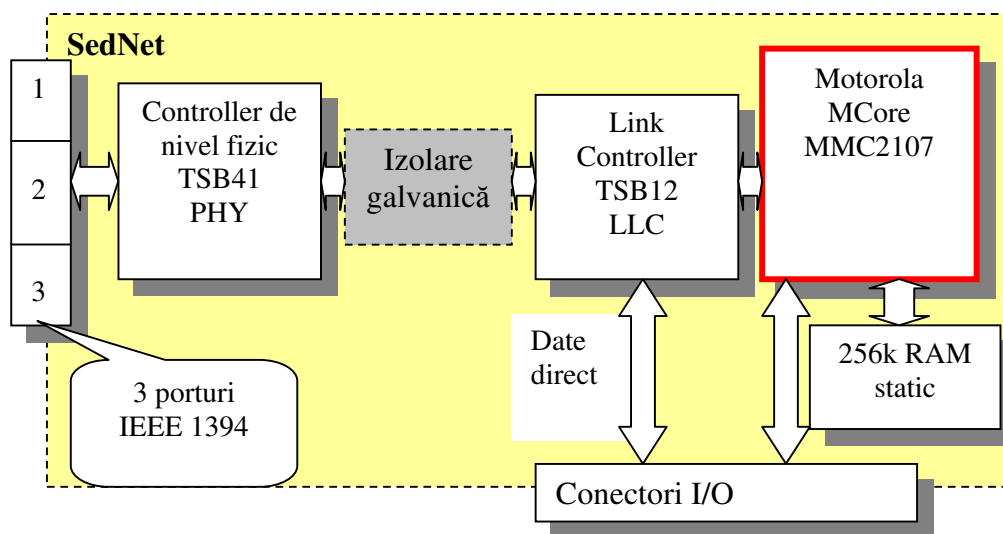


Figura 11.7. schema bloc a plăcii SedNet

Partea software se numește Micro-Stack, rulează pe microcontrollerul plăcii și realizează nivelele bus-management și tranzație din comunicația IEEE 1394. Există și varianta de a cumpăra sursa programului Micro-Stack și de a o porta pe un alt microcontroller pentru a dezvolta o soluție hardware proprie. Alimentarea plăcii este realizată cu alimentator propriu sau prin cablul IEEE 1394, dar în al doilea caz acest caz nu este posibilă izolarea galvanică între controllerul de nivel fizic și restul plăcii. Microcontrollerul MMC2107 are încorporat 128k FLASH iar pentru testare și

debugging portul JTAG este scos la pinii de I/O. Accesul este posibil de la distanță prin IEEE 1394 la toate resursele plăcii.

Se pot conecta maximum 62 dispozitive IEEE 1394, cu viteze de transfer posibile 100, 200 și 400Mbps. Placa suportă atât transferuri izocrone cât și asincrone, datele putând fi preluate direct, fără intermediul microcontrollerului MMC2107 de la controllerul de nivel legătură. Legătura directă este recomandată pentru transferul datelor cu volum mare, cum ar fi cele de la dispozitivele video. Semnalele de legătură cu microcontrollerul sunt cele obișnuite - un port RS232, SPI, JTAG, semnale de întrerupere, reset, tact, linii de I/O etc. Placa a fost special concepută pentru aplicații înglobate care nu conțin PC pentru că nu are interfață PCI.

11.2. Transferul de date în infraroșu IrDA

În 28 iunie 1993, un grup de 120 de reprezentanți din 50 de companii de calculatoare au creat o asociație numită Infrared Developers Association (IrDA) cu scopul de a standardiza comunicațiile în infraroșu. Tehnologia era deja pusă la punct în telecomenzile TV și la transferul de date între Notebook-uri, dar se simțea nevoia unui standard. Primul standard, bazat pe portul serial RS232 a fost aprobat în 1994. Acest standard folosește specificațiile portului serial, aceeași structură de date dar din păcate și limitele vitezei. În 1995 a fost aprobat un nou standard de mare viteză care împinge limita de viteză la 1Mbps.

În cadrul comunicațiilor necablate (wireless), standardul IrDA face parte din categoria transmisiei infraroșu directe, o comunicație punct la punct. Între echipamente trebuie să existe vizibilitate directă. În afara acestui tip de comunicație mai există comunicația infraroșu difuză, o comunicație ce permite legături multiple și care nu necesită vizibilitate directă între echipamente, dar care necesită materiale speciale de construcție a clădirilor.

Avantajele comunicației infraroșu sunt evidente: ușurarea portabilității aparatelor, eliminarea cuplajelor cablate (nu se mai întrerup firele și nu se mai întâmplă ca un conector să nu facă contact), eliminarea perturbațiilor electromagnetice radiate, deci implicit eliminarea interferențelor electromagnetice. Aceste avantaje sunt majore în zonele de lucru cu regim special, cum ar fi centralele nucleare, laboratoarele de cercetare și măsurări de precizie, acceleratoarele de particule etc. Aceste avantaje au dus la răspândirea echipamentelor cu interfață IrDA. Există Notebook-uri care au o astfel de interfață, imprimante pentru Notebook-uri, dar și mouse și tastaturi infraroșu pentru calculatoarele desktop uzuale, sau unele telefoane mobile. Faptul că perifericele în infraroșu se fabrică în multe exemplare a dus firesc și la scăderea prețurilor, care le face accesibile.

11.2.1. Codificarea datelor

La viteze între 2.4 kbps și 1.152 Mbps datele se codifică RZI (Return to Zero Invert). Prin această codificare unui 0 logic îi corespunde un impuls, iar la un 1 logic nu apare nici un impuls, figura 11.8, sus. Impulsul are o durată fixă, mai mică decât durata celulei bit.

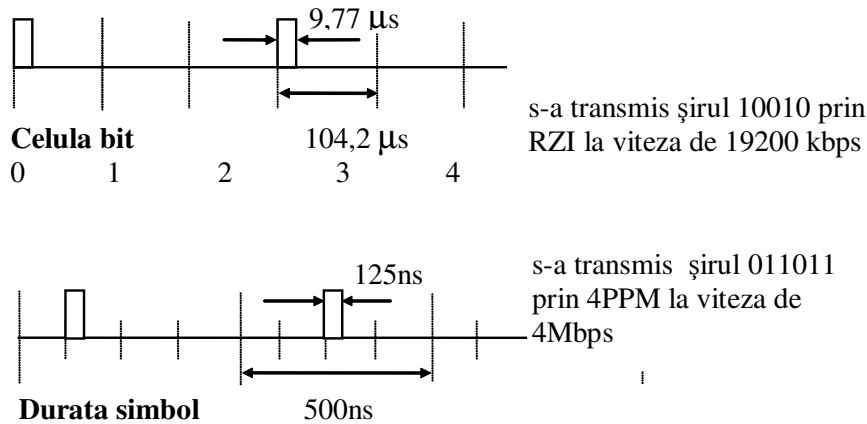


Figura 11.8. Codificarea datelor la IrDA

Un impuls durează $3/16$ din lungimea unei celule bit. Un impuls cu 10% mai lung este acceptabil. De exemplu, celula bit la viteza de 9600 bps durează $104.2 \mu\text{s}$, deci durata impulsului este $18.53 \mu\text{s}$. La viteze mai mari durata impulsului este $1/4$ din durata celulei bit. Durata impulsului în celula bit este astfel aleasă ca receptorul optic să poată distinge impulsurile.

La viteza de 4 Mbps codificarea se face prin modularea impulsurilor în poziție. IrDA implică 4 poziții pentru impuls, de aceea codificarea se numește 4PPM (4 Pulse Position Modulation). Această codificare folosește poziția unui impuls în celula bit pentru a indica o valoare logică. Lungimea celulei bit se numește durata unui simbol (symbol duration) și este împărțită în 4 segmente egale numite *chips*. Un impuls poate apare în unul și numai în unul din aceste segmente. Fiecare impuls în una din 4 poziții poate codifica 2 valori binare. Pentru transmisia unui octet sunt suficiente 4 celule bit. Datele sunt transmise în format pe 8 biți de date, un bit de start, un bit de stop, fără bit de paritate. Se transmit astfel 10 biți/caracter. Cel mai puțin semnificativ bit se transmite primul. Durata impulsului este de 125 ns pentru un impuls singular. Pentru mărirea siguranței transferului se pot folosi impulsuri duble, a căror durată totală este de 250 ns. Codificarea 4PPM este autosincronizabilă, deoarece în fiecare celulă bit există un impuls. Codificarea RZI însă nu este autosincronizabilă, deoarece poate apare un șir lung de valori logice 1 care înseamnă lipsa impulsurilor transmise o perioadă lungă de timp, perioadă în care receptorul poate pierde sincronizarea.

Sistemele IrDA de viteze mici lucrează în mod asincron și la aceste sisteme transmisia se face cu tact standard (cu acest cod neautosincronizabil), pentru că nu pot apare

erori prea mari la transmisia a doar 10 biți. La viteze medii însă, în cazul transmisiei sincrone, este nevoie de autosincronizare.

Există multe firme care produc circuite pentru transferul de date IrDA, așa cum sunt: Texas Instruments, MAXIM, Sharp, Novalog, Agilent Technologies, California Eastern Laboratories, EXAR, Linear Technology etc. Ca variante constructive se poate opta pentru un transceiver IrDA care să se conecteze la un circuit UART existent, se poate alege un circuit UART cu port IrDA sau se poate realiza o interfață IrDA cu microcontroller.

11.2.2. Circuite IrDA

Un transceiver TI de tip TIR1000 poate lucra atât IrDA cât și în standardul de transfer infraroșu al Hewlett Packard HPSIR. Viteza poate fi între 1200 și 115200 bps, iar tensiunea de alimentare între 2,7 și 5V. Este disponibil în capsulă PSOP (Plastic Small Outline Package) cu 8 terminale. Circuitul codează și decodează semnalele IrDA, așa încât el se poate conecta la un UART, figura 11.9.

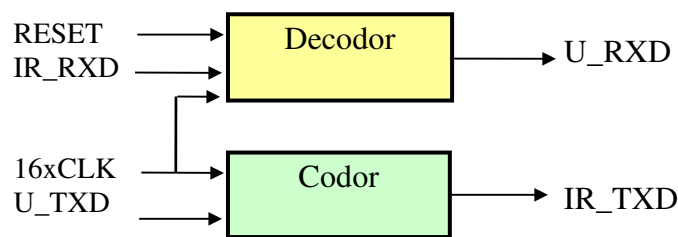


Figura 11.9. Transceiver TIR1000

1. 16xCLK este un semnal de ceas care trebuie să fie de 16 ori mai mare decât rata de transmisie. Tactul maxim este de $16 \times 115200 = 1.843 \text{ MHz}$.
2. IR_RXD intrarea pentru date recepționate IrDA, cu factorul de umplere de 3/16, de la un dispozitiv optoelectronic.
3. IR_TXD ieșire pentru datele emise IrDA către o diodă în infraroșu.
4. RESET inițializare circuit, legat la linia de RESET a circuitului UART.
5. U_RXD date decodate, spre intrarea de date a UART.
6. U_TXD date de transmis de la ieșirea de date a UART.

Un modul care conține un emițător și un receptor IR (ZHX1010) este prezentat în figura 11.10 stânga iar în mijloc este schema de conectare care arată simplitatea soluției. Pentru conectarea la un PC se poate folosi un adaptor USB IrDA (dreapta).

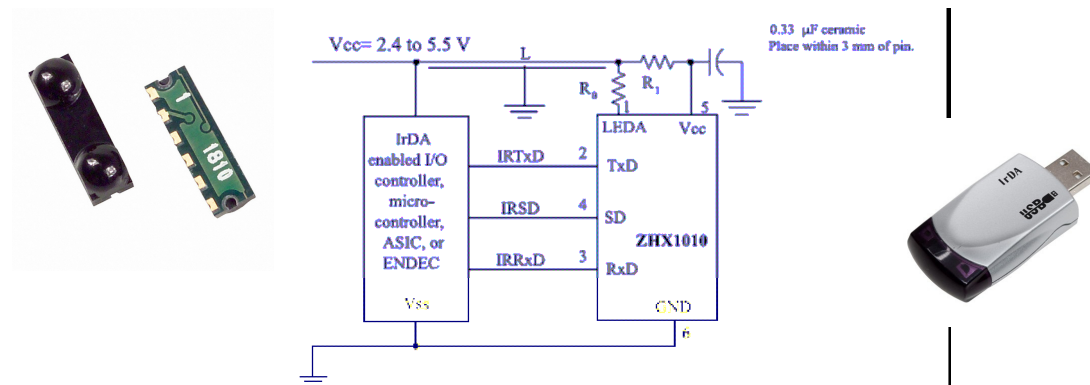


Figura 11.10. Modul IR (sursa <http://www.alldatasheet.com/datasheet-pdf/pdf/113918/ZILOG/ZHX1010.html>) și adaptor USB IrDA (sursa <http://www.i-tec-europe.eu/?t=3&v=20>)

11.3. Interfața SATA

Interfața serială SATA este cea mai răspândită interfață în prezent pentru cuplarea hard discurilor. Interfața este formată din 2 perechi de fire cu transmisie LVDS (Low Voltage Differential Signaling, 250mV), o pereche pentru date emise, una pentru date recepționate, transmisia fiind diferențială. Datele sunt codificate 8B/10B ca și Ethernet Gigabit, PCIe sau Fibre Channel. Legătura SATA este o legătură punct la punct, fiecare drive este conectat printr-un cablu serial la placa de bază. SATA a înlocuit interfața paralelă ATA (numită și PATA) (Advanced Technology Attachment).

Prima generație SATA (SATA 150) comunică cu viteza de 1,5Gbps, ceea ce înseamnă circa 1,2Gbps informație utilă, adică 150Mocțeți/s, ceva mai mult decât ATA133. SATA 300 (SATA 2) asigură o viteză de 3Gbps și are compatibilitate în jos la SATA 150 în urma unei negocieri la inițializare. Acest protocol nu este implementat în toate dispozitivele gazdă, așa încât pe unele hard discuri există un jumper pentru selecția SATA 150 sau SATA 300.

Avantajele SATA sunt:

- Viteza de transfer ridicată
- Posibilitatea implementării Hot Plug In
- Posibilitatea unor porturi externe SATA (eSATA)
- Cablul de date este mai mic deci mai ieftin și asigură o circulație mai bună a aerului în carcasă.

Cablul de alimentare SATA transmite tensiuni de 3,3V, 5V și 12V. Adaptorul de la sursa ATX nu are 3,3V ceea ce împiedică hard discurile astfel alimentate să aibă capacitatea de Hot Plug In.

11.3.1. Protocolul în interfața SATA

Interfața SATA este o interfață serială sincronă cu refacerea tactului din datele citite, codificarea fiind cu adăugare de biți, transmisia fiind realizată cu cadre (blocuri) de date. Sunt definite secvențe de date numite primitive SATA care sunt de fapt cadre de date utilizate pentru comenzi / stări.

- SYNC – Sincronizare
- X_RDY – Interfața este gata să transmită un cadru
- R_RDY – Interfața este gata să recepționeze un cadru
- R_IP – Interfața este în curs de recepție a unui cadru
- WTRM – Interfața a terminat de trimis un cadru și așteaptă terminarea conexiunii
- R_OK – Interfața a terminat de recepționat un cadru și CRC este corect
- R_ERR – Interfața a terminat de recepționat un cadru și CRC nu este corect
- SOF – Început de cadru
- EOF – Terminarea unui cadru

Înainte de transferul de date interfața SATA- negociază viteze de transfer. Pentru aceasta se folosesc cadrele sau primitivele de negociere, figura 11:

- COMINIT – Folosit de dispozitivul SATA pentru a solicita o inițializare
- COMRESET – Folosit de gazdă pentru a forța un Reset
- COMWAKE – Folosit de dispozitiv sau gazdă pentru a scoate interfața din modul adormit.

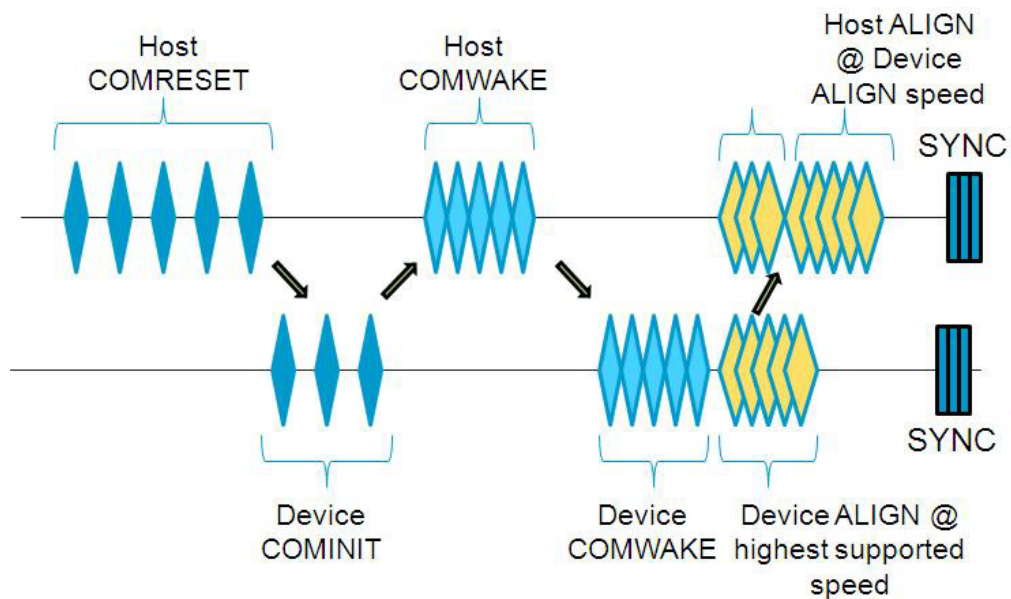


Figura 11.11. Negocierea vitezei (sursa http://www.serialtek.com/sata_protocol_overview.asp)

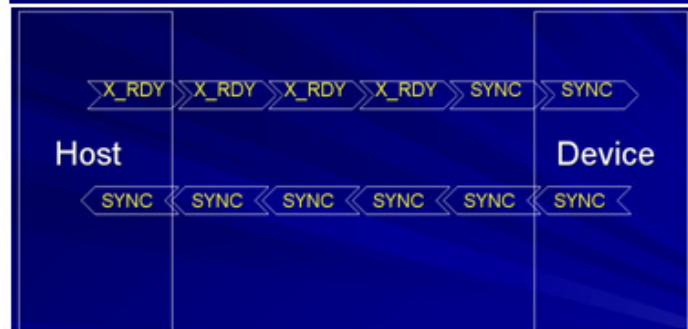
Gazda trimite un COMRESET care resetează drive-ul. Drive-ul solicită cu COMINIT inițializarea comunicației. Gazda trimite COMWAKE care scoate drive-ul din modul adormit. Drive-ul trimite ca răspuns COMWAKE și ALIGN la frecvența cea mai mare de comunicație.

Secvențele SYNC se trimit pentru a permite sincronizarea buclei PLL din receptor atât de la gazdă la drive cât și invers.

După negocierea vitezei se transmit primitive de sincronizare pentru a permite sincronizarea buclei PLL din receptor atât de la gazdă la drive cât și invers, figura 11.12.



Cu primitiva X_RDY gazda comunică că este gata să transmită date.



Cu primitiva R_RDY drive-ul comunică că este gata să recepționeze date.



Figura 11.12. Protocolul de transfer de date SATA (1) (sursa www.eng.auburn.edu)

Gazda începe să transmită date, precedate de primitiva SOF, figura 13.



Cu primitiva R_IP drive-ul anunță că primește date



După terminarea cadrului de date gazda inserează CRC și primitiva EOF. Transmite apoi primitiva WTRM așteptând închiderea conexiunii.



Drive-ul anunță recepția completă a datelor și cu primitiva R_OK comunică un CRC corect.



După terminarea transmisiei de date atât gazda cât și drive-ul transmit primitiva SYNC.

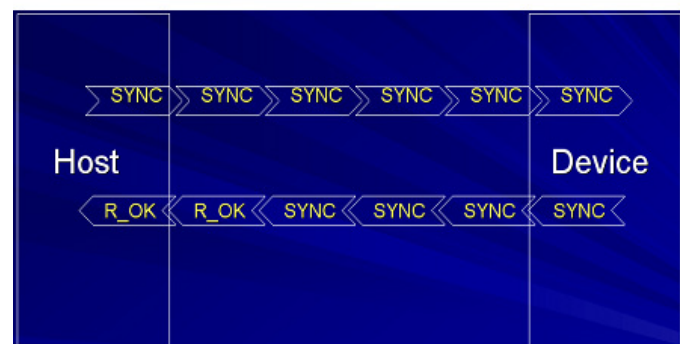


Figura 11.12. Protocolul de transfer de date SATA (2) (sursa www.eng.auburn.edu)



Rezumat

Acest modul cuprinde o prezentare generală a trei tipuri diferite de interfețe seriale. IEEE 1394 este sincronă, cu transmisia tactului (prima versiune) și codificarea autosincronizabilă 8B10B (IEEE1394b). IrDA are o prima variantă de transmisie asincronă cu tact standard, apoi o variantă sincronă cu refacerea tactului. Interfața SATA este sincronă cu refacerea tactului. IEEE 1394 și SATA realizează transferul prin cadre de date iar IrDA prin cuvinte. Sunt descrise la fiecare tip de interfață contextul în care au apărut, caracteristici, codificarea și protocolul de transfer. La IEEE1394 și IrDA sunt prezentate circuite specializate care pot fi utilizate pentru a construi o astfel de interfață.



Bibliografie

1. M. Romanca, P. Ogrutan, *Sisteme cu calculator incorporat. Aplicatii cu microcontrollere*, Editura Universitatii Transilvania Brasov, 2011, pag. 31-42, online la: <http://vega.unitbv.ro/~ogrutan/Microcontrollere2011/3-usb-ieee1394.pdf>
2. C. Gerigan, P. Ogruțan, *Tehnici de interfațare*, Ed. Transilvania Brașov, 2000, ISBN 973-9474-94-2, pag 137-145, online la: <http://vega.unitbv.ro/~ogrutan/>
3. C. Erickson, *Department of Electrical and Computer Engineering*, Auburn University, Auburn, AL 36849 Chris.Erickson@auburn.edu, www.eng.auburn.edu

Test de autoevaluare



R

1 Asemănările dintre USB și IEEE1394 sunt:

- (a) ambele standarde permit alimentarea prin cablu
- (b) în ambele standarde gazda alocă o adresă dinamică dispozitivului la cuplare
- (c) ambele standarde admit decuplarea dispozitivelor în timpul funcționării
- (d) ambele standarde permit transferul și dacă lipsește gazda

I. Vezi pagina 2

R

2. Codificarea D/S este utilizată la transmisii cu refacerea tactului din datele recepționate

- adevărat
- fals

I. Vezi pagina 4

R

3 Codurile 8B10B 1000000100 și 10110111101101111100.:

- (a) sunt ambele valide
- (b) primul valid, al doilea invalid
- (c) primul invalid, al doilea valid
- (d) ambele invalide

I. Vezi pagina 4

4. Avantajele codării 4PPM față de RZI la transmisia IrDA sunt:

R

- (a) codarea 4PPM asigură costuri mai mici
- (b) codarea 4PPM este autosincronizabilă iar RZI nu este
- (c) debitul de informație poate fi mai mare la 4PPM
- (d) codarea 4PPM asigură un număr de erori mai mic

I. Vezi pagina 9

5. Protocolul SATA începe cu negocierea vitezei pentru ca interfața să permită cuplarea hard discurilor cu viteze diferite

R

- adevărat
- fals

I. Vezi pagina 13

6. Protocolul SATA permite transferul datelor half duplex. În timp ce unul dintre partenerii din comunicație trimite date, celălalt trimite o primitivă de confirmare

R

- adevărat
- fals

I. Vezi pagina 15

7. Confirmarea că datele au fost recepționate corect la protocolul SATA se face cu primitiva R_OK.

R

- adevărat
- fals

I. Vezi pagina 15

R

Răspunsuri corecte:

1. a și c, pagina 2 comparația între USB și IEEE1394
2. fals, pagina 4, D/S nu este autosincronizabil
3. d, primul pentru că are o secvență de zerouri de 6 succesive și se permit doar 5, al doilea pentru că numărul de valori de 1 este 14 și cel de zero este 6, deci o diferență mai mare de 2 valori, pagina 4
4. b și c, pagina 9
5. adevărat, pagina 13
6. fals, vezi definiția comunicației half duplex și pagina 15
7. adevărat, pagina 15