

7.CONTROLLERE PENTRU USB

Magistrala USB cunoaște o răspândire din ce în ce mai mare deoarece noile echipamente periferice sunt echipate cu interfață USB. Au apărut camere digitale cu USB, majoritatea imprimantelor sunt echipate cu USB și toate chipseturile pentru plăci de bază au interfețe USB. Pentru cuplarea pe USB se construiesc microcontrollere cu USB, așa cum este un microcontroller de la Anchor Chips format dintr-un nucleu 8051 și o interfață USB, numit EZ-USB [1]. O altă variantă este folosirea unui microcontroller de uz general cuplat cu un controller USB, așa cum este USBN 9603.

Atunci când a fost scrisă lucrarea [1] existau foarte puține chipuri USB acum există un număr foarte mare, create atât de producători de prestigiu cum ar fi Motorola sau INTEL dar și de nou veniți pe piață cum ar fi Mitsubishi. O simplă căutare pe Internet dovedește abundența de soluții USB. Mai mult, pentru a susține vânzarea de chipuri USB producătorii acompaniază producerea chipurilor cu seturi de unelte de dezvoltare, cum ar fi kit-urile USB [9].

În [1] se prezintă magistrala USB căreia îi este dedicat un capitol special și care se recomandă de a fi citit înainte de a parcurge acest capitol. În această lucrare se respectă terminologia din [1] referitoare la USB. Suplimentar sunt prezentate câteva aspecte ale transmisiei USB, care este destul de complexă (specificațiile USB au circa 600 pagini).

7.1.Controller de USB USBN 9603 (producător National Semiconductor)

USBN este un controller de nod USB compatibil USB 1.0 și 1.1. El integrează un transceiver USB cu regulator de tensiune de 3,3V, un motor (Engine) serial numit SIE (Serial Interface Engine), buffere FIFO pentru USB, o interfață paralelă pe 8 biți multifuncțională, un generator de tact și o interfață MICROWIRE PLUS. Tot în [1] se descrie pe scurt interfața MICROWIRE care echipează microcontrollerele COP.

Capsula circuitului este de 28 de pini în diferite variante, inclusiv DIP.

Schema bloc a circuitului este dată în figura 7.1.:

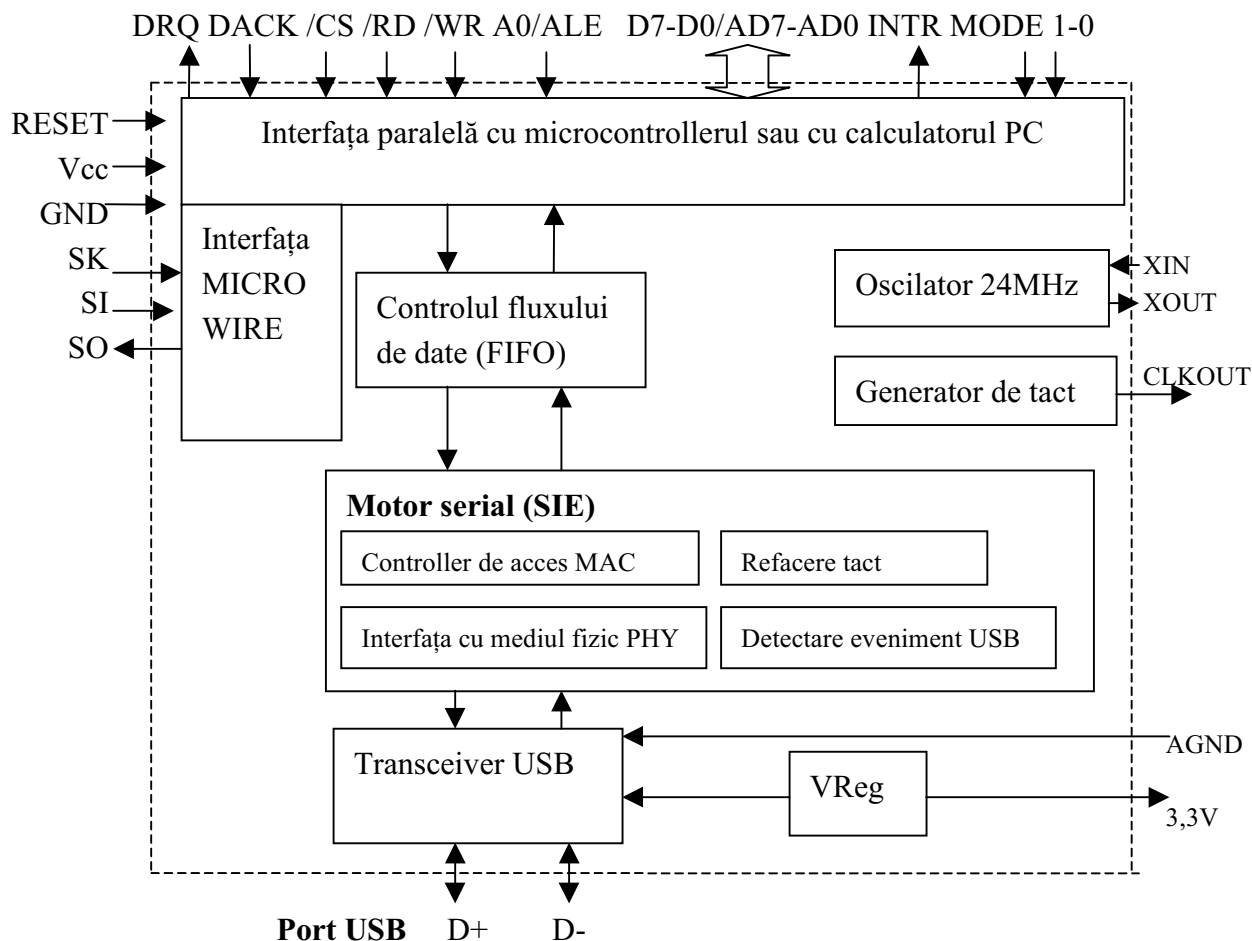


Fig. 7.1. Schema bloc a circuitului USBN 9603

Transceiverul USB conține un receptor diferențial, un receptor asimetric cu referința de tensiune integrată și un transmițător cu sursă de curent integrată. Transmițătorul are panta de creștere controlată pentru a minimiza interferențele electromagnetice. Transceiverul are posibilitatea de trecere în înaltă impedanță pentru a se putea efectua transferuri bidirecționale half duplex. Receptorul diferențial este format din 2 linii de recepție asimetrice, cu referința de tensiune de la pinul corespunzător (3,3V) față de masa AGND de transmisie. La linia D+ trebuie conectat un rezistor de 1,5K Ω .

Blocul VREG creează tensiunea de referință de 3,3V din tensiunea de alimentare de 5V generată de sursa sistemului sau transmisă prin cablul USB de la gazda USB. Pinul de referință de tensiune trebuie decuplat cu un condensator de 1 μ (tantal) la masă.

Motorul serial este compus din interfața cu mediul fizic (Physical Layer Interface, PHY) și controllerul de acces (Media Access Controller, MAC). PHY include circuitul de refacere a tactului din datele citite, circuitul de detecție a sfârșitului pachetului de date EOP, logica de adăugare de biți (la emisie) și extragere de biți (la recepție) (bit stuffing) pentru a se obține

transmisii autosincronizabile. Modulul MAC realizează formarea pachetelor de date, generarea și verificarea CRC și detecția de adresă. Motorul serial este responsabil de detectarea evenimentelor specifice USB cum ar fi NodeReset, NodeSuspend sau NodeResume. Semnalele de date de ieșire spre transceiver sunt sincronizate cu o eroare de 1ns. Logica de adăugare de biți inserează un 0 după fiecare 6 de 1 consecutivi în șirul de date pentru a asigura numărul de tranziții necesare refacerii tactului din datele citite. Datele citite (codate NRZI) sunt folosite pentru extragerea tactului de transmisie (12MHz) cu ajutorul unui tact fix de 48MHz obținut de la oscilatorul de tact prin dublarea frecvenței cu o buclă PLL.

Un punct terminal USB (endpoint) este sursa sau destinația datelor. Fluxul USB reprezintă mutarea datelor între USB și memoria calculatorului gazdă. Sunt acceptate la un moment dat maximum 31 de puncte terminale. Se alocă 16 puncte terminale de recepție, 16 puncte terminale de transmisie iar punctul terminal 0 (de control) este bidirecțional. Fiecare punct terminal este o entitate adresabilă și trebuie să răspundă la instrucțiuni de intrare sau ieșire de la gazda USB. Un IN înseamnă că gazda primește date de la un punct terminal iar un OUT înseamnă că gazda trimite date. La detecția unui IN de pe bus, un punct terminal adresat trebuie să răspundă cu un pachet de date. Dacă punctul terminal este oprit se trimite un pachet STALL iar dacă punctul terminal este validat dar nu are date se trimite un pachet NAK (Negative ACK). La fel răspunde un punct terminal și la detecția unui OUT.

În orice moment circuitul poate fi în una dintre stările:

- NodeOperational- operare normală;
- NodeSuspend- operare suspendată datorită inactivității USB;
- NodeResume- dispozitivul revine la operarea normală din starea de operare suspendată;
- NodeReset- dispozitivul se inițializează.

Din starea de operare normală, dacă 3ms nu sunt date pe magistrală, circuitul intră în NodeSuspend, prin poziționarea unui bit în registrul ALTEV și printr-o cerere de întrerupere dacă este validată. Dispozitivul poate intra în starea de NodeResume ca răspuns la o cerere generată de gazda USB. De asemenea dispozitivul poate iniția o cerere de WAKE-UP către gazda USB, dacă aceste cereri sunt validate. Când circuitul USBN detectează o activitate pe magistrală inițiază starea de NodeResume folosind registrul NFSR. Semnalul de NodeResume este trimis pe bus cel puțin 20ms, urmat de secvența EOP (End of Packet) după care circuitul este operațional.

Pachetele de date sunt trimise de la gazdă la toate nodurile rețelei USB. Detectarea adresei proprii este implementată hardware în circuit pentru selectarea mesajelor destinate circuitului respectiv. Dacă se recunoaște adresa proprie pachetul este recepționat în FIFO iar dacă nu este abandonat. Adresa proprie este stocată în FADR (Function Address Register). Dacă adresa este recunoscută următorul câmp din datele seriale, câmpul de punct terminal este comparat în paralel cu toate registrele de control al punctelor terminale (EPCx). O egalitate determină ca șirul de date să fie transmis / recepționat în bufferul FIFO corespunzător, figura 7.2.:

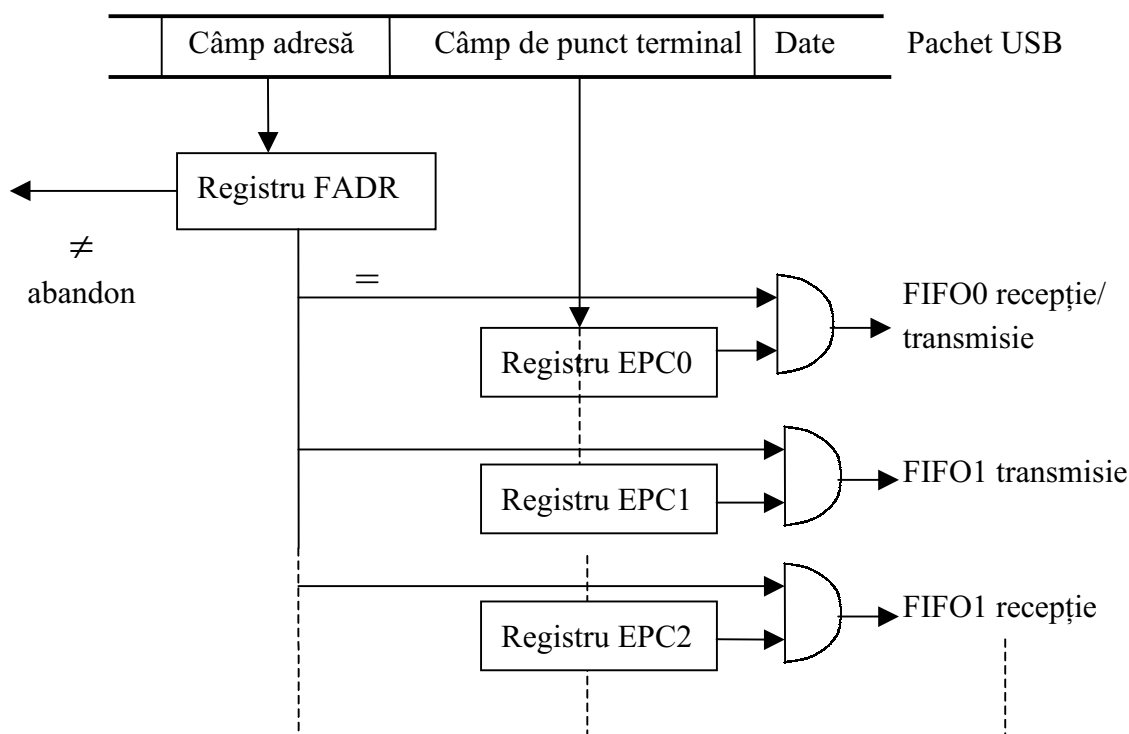


Fig. 7.2.: Schema bloc a fluxului de date la operarea cu puncte terminale

Un dispozitiv USB folosește 7 buffere FIFO: unul bidirecțional de control, 3 de transmisie și 3 de recepție, corespunzătoare celor 7 puncte terminale. FIFO de control are dimensiunea de 8 octeți iar celelalte sunt de 64 de octeți fiecare. Fiecare FIFO poate fi programat să corespundă unui punct terminal și corespund toate unei adrese (unui dispozitiv).

Fluxul de date între USB și microcontroller sau calculatorul gazdă este reprezentat în figura 7.3.:

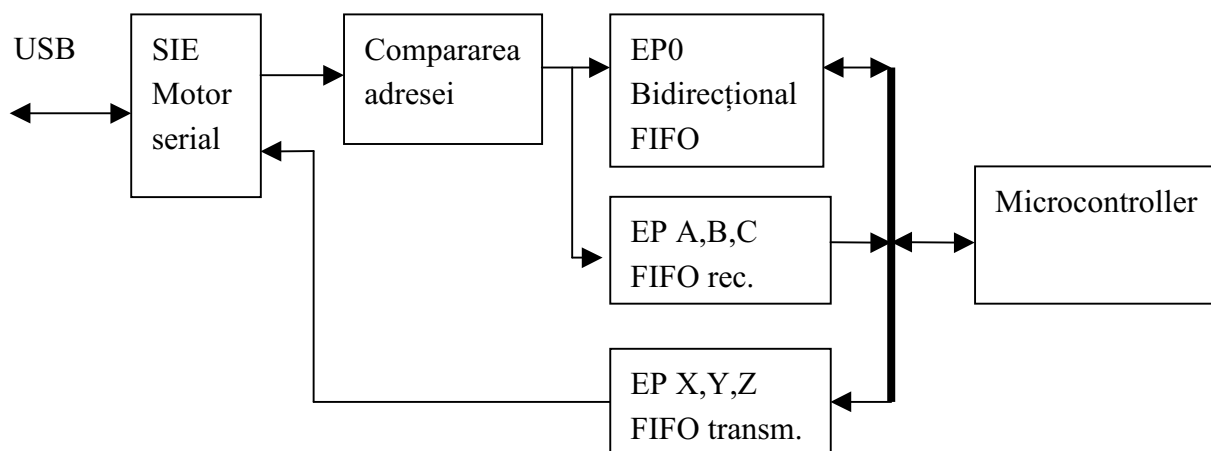


Fig. 7.3.: Fluxul de date între USB și calculatorul gazdă

Bufferele FIFO de recepție sunt EPA, EPB și EPC iar cele de transmisie EPX, EPY și EPZ, iar bufferul bidirecțional este EP0.

USBN poate fi conectat la un sistem (microcontroller) prin interfața paralelă pe 8 biți sau prin interfața serială MICROWIRE. Tipul interfeței este selectat cu semnalele MODE 1 și MODE 0. Prin interfața paralelă conexiunea se poate realiza în mod nemultiplexat sau multiplexat.

Modul nemultiplexat folosește liniile D0-D7 pentru date și adrese și semnalele /CS, A0, /RD, /WR. Liniile D0-D7 intră în registrul de adrese sau în cel de date funcție de linia A0. La transferul în mod salvă registrul de adrese se înscrie o dată și se execută citiri /scrieri la adrese consecutive. USBN conține un registru de date citite și un registru de date scrise.

Modul multiplexat folosește liniile D0-D7 pentru date și adrese și semnalele /CS, /RD, /WR și ALE. Adresele sunt selectate (memorate într-un registru intern) cu semnalul ALE. Datele sunt active la următorul impuls /RD (la citire) sau /WR (la scriere).

USBN admite mod de transfer DMA numai prin interfața paralelă, controlul fiind asigurat de un controller DMA extern. Pentru dialogul DMA se folosesc semnalele DRQ și /DACK, iar adresele (de pe liniile D0-D7 validate cu A0 sau ALE) sunt ignorate. Semnalul /CS trebuie să fie inactiv pe timpul ciclului DMA, altfel se execută un ciclu de citire /scriere normal. Un singur punct terminal USB poate fi selectat pentru un transfer DMA la un moment dat.

Prin interfața serială MICROWIRE se pot transmite de asemenea comenzi și date. Legătura se face prin semnalele /CS (selecție), ceas serial SK, date seriale de intrare SI și date seriale de ieșire SO.

Circuitul admite mod de lucru cu economie de energie. Modul HALT este un mod static în care circuitul de tact este oprit, toate registrele fiind memorate. Dacă este validată starea de HALT circuitul poate intra în această stare după o comandă de NodeSuspend sau dacă modulul nu este atașat la USB.

Programarea circuitului se face prin registre. Adresa și numele registrelor sunt date în continuare:

Adresa	Nume registru	Semnificație
00	MNCTRL	Controlul circuitului
01	CCONF	Configurare ceas
03	RID	Identificare variantă
04	FADR	Adresa proprie
05	NFSR	Starea nodului
06	MAEV	Evenimente
07	MAMSK	Mască pentru evenimente
08	ALTEV	Evenimente (alternativă)
09	ALTMSK	Mască pentru evenimente (alternativă)
0A	TXEV	Evenimente la transmisie
0B	TXMSK	Mască la transmisie

0C	RXEV	Evenimente la recepție
0D	RXMSK	Mască la recepție
0E	NAKEV	Eveniment NAK
0F	NAKMSK	Mască NAK
10	FWEV	Eveniment FIFO
11	FWMSK	Mască FIFO
12	FNH	Numărul cadrului (High)
13	FNL	Numărul cadrului (Low)
14	DMACNTRL	Control DMA
15	DMAEV	Eveniment DMA
16	DMAMSK	Mască DMA
18	DMACNT	Contor DMA
19	DMAERR	Contor erori DMA
1B	WKUP	Trezire
20	EPC0	Control punct terminal 0
21	TXD0	Date transmise punct terminal 0
22	TXS0	Stare transmisie punct terminal 0
23	TXC0	Comenzi transmisie punct terminal 0
25	RXD0	Date recepționate punct terminal 0
26	RXS0	Stare recepție punct terminal 0
27	RXC0	comenzi recepție punct terminal 0

Urmează registrele pentru punctele terminale 1,2,3,4,5 și 6 (3 de recepție și 3 de transmisie).

Protocolul USB

Toate pachetele USB încep cu o secvență de sincronizare (SYNC) care permite receptorului să-și sincronizeze viteza de recepție pentru șirul de date transmis. Secvența SYNC are 8 biți pentru viteza mică de transfer și 32 de biți pentru viteza mare.

Fiecare pachet începe cu un delimitator numit SOP (Start of Packet) și se termină cu un delimitator EOP (End of Packet)¹. SOP face parte din câmpul SYNC.

După SOP urmează un câmp de identificare a pachetului (Packet Identifier PID) care constă în 4 biți care indică tipul pachetului (formatul pachetului și modul de corecție al erorilor) și 4 biți de verificare a corectitudinii recepției primilor 4 biți (complementul primilor 4 biți).

Pachetele pot fi: de date (Data), de protocol (Handshake), speciale (Special) și de semnalizare (Token). Toate dispozitivele decodează câmpurile PID, verifică corectitudinea lor și răspund la acest pachet dacă operația specificată este posibilă. Dacă nu (de exemplu la o cerere de IN adresată unui punct terminal de ieșire) dispozitivul USB ignoră solicitarea.

O clasificare a pachetelor este dată în tabelul următor:

¹ Ordinea transmiterii biților în cuvânt este începând cu LSB.

Tip PID	Nume PID	Descriere
Semnalizare	OUT	adresă + număr de punct terminal într-o tranzacție de la gazdă la punct terminal
	IN	la fel într-o tranzacție inversă
	SOF	începutul unui cadru (Start of Frame)
	SETUP	adresă + număr de punct terminal într-o tranzacție de programare a unui punct terminal
Date	DATA	pachet de date
Protocol	ACK	receptorul acceptă pachetul
	NACK	receptorul nu acceptă pachetul sau transmițătorul nu îl poate transmite
	STALL	punctul terminal este oprit
	NYET	nu există încă răspuns de la receptor
Special	PRE	gazda semnalizează că transmite un preambul
	ERR	eroare
	SPLIT	începe un transfer cu viteză mare
	PING	verificare a unui punct terminal (la viteză mare)

După PID se transmite câmpul de adresă care conține 2 părți, adresa dispozitivului și adresa punctului terminal. Câmpul de adresă al dispozitivului (7 biți) poate adresa 128 de dispozitive. Adresa punctului terminal este formată din 4 biți. Adresa 0 este rezervată. Adresa specifică sursa sau destinația datelor funcție de câmpul PID de semnalizare. Adresele sunt alocate de calculatorul gazdă în procesul de enumerare. Semnalarea conectării/deconectării unui dispozitiv USB se face de către calculatorul gazdă prin detectarea creșterii/ scăderii tensiunii dintre D+ și D- într-un timp determinat.

După adresă urmează numărul cadrului, pe 11 biți, care este transmis înainte de fiecare cadru fiind decrementat la fiecare cadru de către transmițător.

Câmpul de date poate conține un număr variabil de octeți, între 0 și 1024 de octeți.

Câmpul CRC (Cyclic Redundancy Check) asigură verificarea integrității datelor. 5 biți CRC asigură verificarea câmpului de adresă iar 16 biți CRC asigură verificarea câmpului de date.

Structura unui pachet de semnalizare este dată în figura 7.4.a, a unui pachet de date în figura 7.4.b, iar a unui pachet de protocol care constă doar din câmpul PID în figura 7.4.c.:

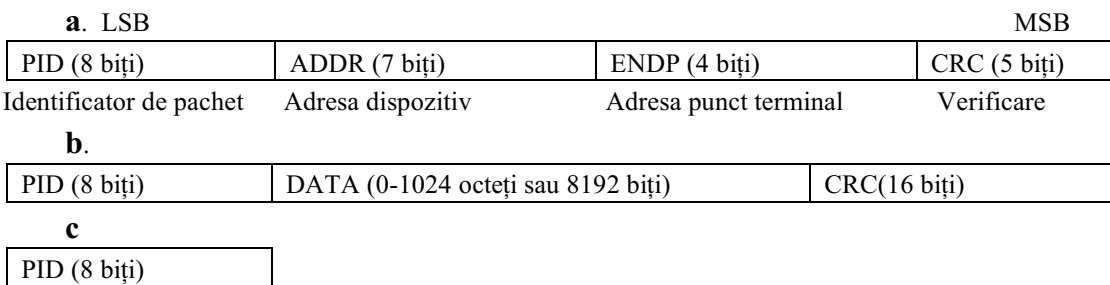


Fig. 7.4.:Structuri de pachete

Prin pachetul de protocol se comunică una dintre stările ACK, NACK, STALL, NYET.

Tranzacțiile USB (și punctele terminale aferente) sunt de mai multe feluri:

- “bulk”, înseamnă transferuri în care se verifică corectitudinea datelor recepționate și se execută reîncercări dacă au apărut erori. Acest tip de transfer se efectuează în 3 faze- pachet de semnalizare, de date și de protocol;

- izocrone, transferurile se efectuează în 2 faze- pachet de semnalizare urmat de pachet de date. Nu se execută reîncercări (de exemplu la transmisii de voce, unde o eroare nu este gravă);
- cu întrerupere, datele sunt prezentate pentru transfer în orice moment dar sunt transferate cu o rată specificată de dispozitiv (de exemplu folosit pentru transmisii de coordonate de către un mouse);
- de control, transferuri de configurare pentru dispozitivele nou conectate.

Datele sunt codate NRZI, adică un bit 0 este reprezentat de o schimbare de nivel iar un 1 de păstrarea nivelului, figura 7.5.:

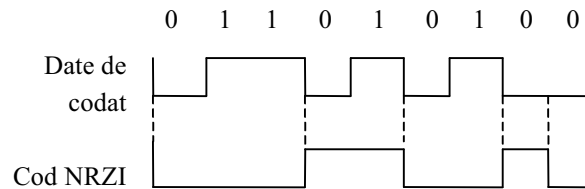


Fig. 7.5.Codarea NRZI

Un șir de 1 înseamnă că semnalul de date nu are tranziții și devine neautosincronizabil iar un șir de 0 înseamnă tranziții dese. Pentru ca semnalul să fie autosincronizabil se inserează automat un 0 la 6 de 1 consecutivi (bit stuffing). Este sarcina receptorului de a recunoaște biții de sincronizare (de umplură) și de a-i elimina. Recepționarea a 7 de 1 consecutivi înseamnă o eroare și cadrul este ignorat.

7.2.Microcontrollere USB

Microcontroller USB INTEL 9x930AX

8x930AX este un microcontroller compatibil MCS 51 (și MCS 251) care conține:

- un motor serial pentru USB care formează /extrage cadrul USB, generează și verifică CRC, codează / decodează NRZI șirul de biți adăugând biții de sincronizare conform standardului USB 1.0 pentru legături bidirecționale half duplex. 8x930 conține și transceiverul USB. Datele de transmis / recepționat sunt memorate în 4 buffere FIFO de transmisie și 4 de recepție. Câte 3 buffere sunt de 16 octeți iar câte unul (la transmisie și recepție) sunt configurabile până la 1Koctet.
- 1 Koctet memorie RAM pentru date. În exterior se pot conecta maximum 256Kocteți memorie. În circuit poate fi integrată și memorie ROM, există variante cu 0, 8 sau 16 Kocteți ROM.
- 3 circuite timer de 16 biți și un modul PCA (Programmable Counter Array) care conține 5 blocuri de captură și comparare.
- Port serial asincron (UART).
- Ceas de gardă (Watchdog).
- Operare la frecvența de 6 sau 12 Mhz.
- Capsulă PLCC cu 68 de pini.

Microcontroller USB Infineon C161U

Infineon a anunțat lansarea pe piață a microcontrollerului USB pe 16 biți C161U care realizează o rată de transfer mai mare de 10 ori decât celelalte circuite de pe piață, astfel încât se evită situația ca datele USB să nu poată fi preluate. Circuitul respectă standardul USB 1.1. și realizează o performanță de 18MIPS (3 până la 6 ori mai mult decât celelate circuite), ceea ce înseamnă o rată de transfer reală de 12 Mbps pe linia USB. Ca și aplicații posibile sunt menționate modemurile xDSL, achiziții de date, dispozitive de securitate, camere digitale, scannere, echipamente de stocare, imprimante etc. La un preț de numai 6USD, apropiat de cel al microcontrollerelor pe 8 biți, circuitul este mult mai flexibil și acoperă o gamă de aplicații care nu puteau fi acoperite de controllerele pe 8 biți. Circuitul este construit în jurul nucleului C166, oferind la partea de USB 8 puncte terminale configurabile software care suportă pachete USB de maximum 1024 de octeți. În plus, circuitul dispune de:

- 8 canale DMA;
- magistrală internă de date de 16 biți care lucrează la 36MHz;
- 3 Kocteți memorie RAM pe chip;
- 56 linii de I/O de uz general;
- canal serial sincron de 18Mbd;
- canal serial sincron / asincron USART de 4,5Mbd cu detecția automată a vitezei de transmisie;
- canal IrDA pentru transmisia în infraroșu la 1Mbps;
- 5 timere de 16 biți;
- ceas de gardă;
- 16 nuvele de întrerupere programabile;
- capsula este TQFP sau MQFP de 100 de pini.

Microcontroller USB AMD Am 186CC

Microcontrollerele AMD Am 186 se adresează pieței de electronică pentru comunicații și fac parte din familia Comm86. Aplicațiile tipice sunt terminale ISDN, aplicații xDSL, telefoane digitale, control industrial folosind HDLC, periferice USB etc.

Am 186CC conține 8 interfețe seriale:

- 4 canale HDLC, fiecare canal poate lucra prin DMA, viteza fiind de 10Mbps;
- interfața USB 1.0, 1.1 și 2.0 (12Mbps), cu 6 puncte terminale și suport DMA;
- 2 canale UART cu stabilirea automată a vitezei;
- interfață serială sincronă (SSI, Synchronous Serial Interface).

Cu canalele seriale pot fi implementate o serie întreagă de protocoale de comunicație: HDLC, SDLC, PPP, V.110, V.120 etc., cu aplicabilitate în modulele ISDN și modemurile DSL.

În plus circuitul dispune de:

- 3 timere de 16 biți;
- controller DMA cu 4 canale;

- 14 semnale de selecție în zona de adrese și I/O pentru conectarea de memorii externe sau dispozitive de I/O;
- controller de memorie RAM dinamică, admite memorii EDO și FPM, maximum 2 bank-uri de 256Kx 16 biți;
- 48 de linii de I/O de uz general;
- 17 linii de întrerupere gestionate de un controller de întreruperi;
- ceas de gardă (Watchdog).

Compatibilitatea soft a acestor microcontrollere cu familia x86 (8086, 80286 ...) are ca și consecință ușurarea programării și micșorarea timpului de punere la punct a unei aplicații.

Nucleul acestui microcontroller este unitatea centrală 80C186 la 50MHz, realizând 6,6MIPS. Circuitul poate fi alimentat cu 3,3V dar porturile de I/O pot lucra la nivele de 5V. Capsula este PQFP de 160 de pini.

Microcontrollerul USB Motorola 68HC908JB8

Acest microcontroller se bazează pe nucleul familiei 68HC08 (microcontroller pe 8 biți). Softul este compatibil cu cel al familiei 68HC05.

Microcontrollerul conține:

- 8Kocteți memorie FLASH care permite 10000 cicluri de reprogramare și 2 ms timp de programare a unui bloc de 64 octeți;
- 256 octeți RAM;
- 2 timere de 16 biți;
- interfața USB 1.1 cu 3 puncte terminale la viteza de 1,5MBPS;
- maximum 37 de linii de I/O (depinde de capsulă);
- alimentare la 3,3V;
- sistem de protecție la scăderea tensiunii de alimentare (prin RESET);
- capsula poate fi de 20 de pini PDIP, 28 pini SOIC sau 44 de pini QFP.

Ca aplicații Motorola sugerează tastaturi, mouse USB, comunicații radio de date (wireless), convertoare USB-paralel, dispozitive de securitate prin USB (inclusiv chei de siguranță), joystick-uri sau game pad-uri.

Microcontroller USB ATMEL AT 90SC6464C

AT 90SC6464C este un microcontroller RISC bazat pe arhitectura RISC AVR3. Nucleul AVR3 poate adresa 8Mocteți memorie program și 16Mocteți memorie de date, are 133 de instrucțiuni dintre care majoritatea se execută într-un singur tact. Aceasta înseamnă că se atinge o viteză de procesare de 1MIPS/MHz. Arhitectura nucleului este o arhitectură Harvard cu 32 de registre de uz general conectate direct la ALU, asigurând ca o instrucțiune de un ciclu să aibă acces la 2 registre diferite.

Circuitul include 128Kocteți de memorie FLASH care poate fi programată în sistem.

Interfața USB (specificațiile USB versiunea 1.1.) conține un motor serial (SIE) și o interfață cu mediul de transmisie. Interfața USB conține 4 puncte terminale EP0, EP1, EP2 și

EP3. Datele sunt dublu bufferate pentru a permite softului să citească un buffer de în timp ce partea hard umple un alt buffer (mod de lucru “ping-pong”).

Circuitul mai conține:

- 64Kocteți EEPROM cu o arie de 64 de octeți OTP;
- 64Kocteți FLASH;
- 3Kocteți RAM;
- 2 timere de 16 biți;
- generator de numere aleatoare;
- procesor cripto pentru protecția la atacuri software;
- monitor de tensiune și frecvență;
- interfață serială cu un calculator gazdă conform ISO 7816-2;
- număr redus de pini ai capsulei (10) datorită interfețelor seriale de transfer.

Microcontroller USB Mitsubishi M37641

Avantajul acestui microcontroller USB este că include pe același chip interfața USB conform standardului USB 1.1 la viteză de 12Mbps și 32Kocteți memorie FLASH. Circuitul mai conține 3,5Kocteți RAM și suportă alimentare de 3,3V ceea ce îl face potrivit aplicațiilor portabile, cum ar fi telefoane celulare, camere de luat vederi și instrumente muzicale. M37641 este al treilea microcontroller pe 8 biți USB realizat de Mitsubishi.

Microcontrollerul este construit în jurul nucleului 7600, care operează la 12MHz, proiectat pentru consum mic, interferențe electromagnetice (EMI) reduse și suport eficient de programare în C. Memoria FLASH dă posibilitatea ca microcontrollerul să poată fi programat în circuit, chiar și la distanță prin USB.

Pentru că pe cablu USB se transmite și o tensiune de +5V, un convertor intern DC-DC face conversia de la 3,3V la 5V.

Controllerul USB conține punctul terminal de control EP0, 4 puncte terminale pentru intrare și 4 puncte terminale pentru ieșire. Dimensiunea bufferelor FIFO poate fi programată. Un sintetizator intern de frecvență asigură ceasul de 48MHz.

Interfața cu calculatorul este o interfață paralelă, compatibilă INTEL 8042 care face ca microcontrollerul să poată lucra în mod MASTER sau SLAVE. Circuitul admite transfer DMA prin 2 canale independente, asigurând o viteză de transfer de 6Mocteți/s.

Microcontrollerul mai conține:

- 2 timere de 16 biți;
- 3 timere de 8 biți;
- 2 circuite UART full duplex;
- 66 de linii de I/O de uz general;
- mecanismul de adresare al memoriei externe asigură adresarea a 64 Kocteți;
- capsula este QFP sau LQFP de 80 de pini.

Bibliografie

1. C. Gerigan, P. Ogruțan, “Tehnici de interfațare”, Ed. Universității Transilvania Brașov, 2000

Link-uri utile

2. <http://usb.org/> Specificațiile USB
3. www.intel.com Microcontroller USB INTEL 8x930AX
4. www.winbond.com Microcontroller USB WINBOND W81E381D
5. www.microcontroller.com Microcontroller USB Motorola 68HC908JB8
6. www.atmel.com Microcontroller USB Atmel AT90SC6464C-USB
7. www.infineon.com Microcontroller USB Infineon (fost SIEMENS) C161U
8. www.mitsubishichips.com Microcontroller USB Mitsubishi M37641
9. www.amis.com Kit USB realizat cu microcontrollerul R8051 (variantă RISC a 8051)