

2. CIRCUITE PENTRU REȚEA

În acest capitol vor fi prezentate circuite interfață de rețea Ethernet și Internet. În primul rând vor fi prezentate două circuite controller de rețea, RTL8019 și CS8900A, interfețe între o unitate centrală (care poate fi un PC și conexiunea se face pe ISA sau un microcontroller) și rețeaua Ethernet. Sunt prezentate exemple de conectare atât la ISA cât și la două microcontrollere diferite. Este prezentat apoi circuitul SEIKO S-7600 care gestionează hardware stiva TCP-IP și datorită căruia conectarea la Internet a unui sistem cu microcontroller poate fi realizată mult mai ușor. La sfârșit este prezentat un circuit W89C982 care gestionează transferul datelor prin canalele UTP de rețea Ethernet (are funcția de hub de rețea).

Obligatoriu la nivel de schema bloc si principiu de functionare

2.1.Circuitul interfață de rețea RTL 8019 (producător REALTEK)

RTL 8019 este un controller de rețea Ethernet care oferă o soluție simplă și performantă aplicațiilor cu transfer de date prin rețea. Circuitul permite transfer full duplex pe UTP (și dacă plăcile de rețea sunt cuplate între ele printr-un switch adecvat) mărind rata de transfer de la 10Mbps la 20Mbps. Circuitul suportă 3 nivele de economie de energie: mod adormit, mod oprit (Power Down) dar cu tactul în funcțiune și mod oprit cu tactul oprit.

La RTL 8019 poate fi conectată o memorie ROM numită BROM (Boot ROM) din care se poate încărca un set de date care vor aduce sistemul de operare de pe server, caz în care stația conectată nu are nevoie de hard disc (aplicație importantă la sistemele care comandă automatizări). Memoria ROM poate fi de 16K, 32K sau 64K și poate fi citită de sistem prin interfața ISA, începând de la o adresă configurabilă. După încărcarea sistemului de operare memoria BROM poate fi invalidată pentru a elibera zona de adrese ocupată.

Pentru a mări viteza de transfer, cadrele recepționate sau cele de emis pot fi stocate într-o memorie locală SRAM cuplată la RTL 8019. Se pot conecta până la 32Kocteți SRAM. Există variante de RTL 8019 cu memorie SRAM integrată.

Configurația și parametrii de inițializare pot fi stocați într-o memorie serială EEPROM cuplată la circuit. Această memorie (de tip 9346) poate fi programată în circuitul cu RTL 8019.

Schema bloc simplificată a circuitului RTL 8019 este dată în figura 2.1:

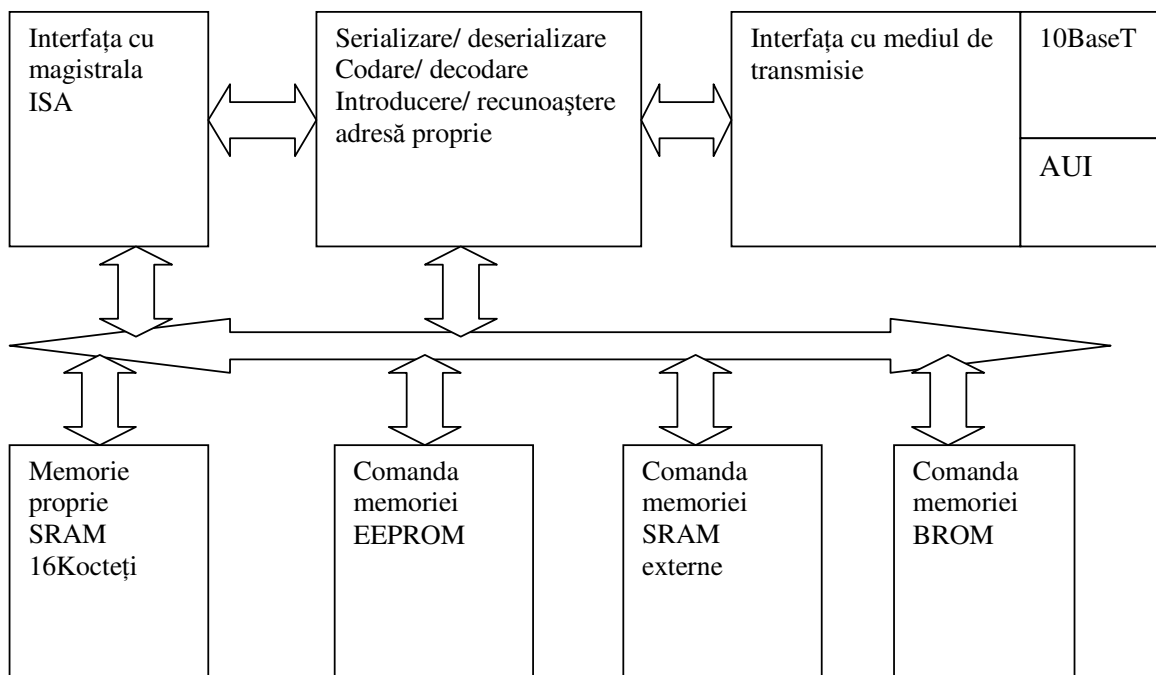


Fig. 2.1. Schema bloc a circuitului RTL 8019

Programarea circuitului se realizează cu un set de registre citite /scrise de procesor la adrese de I/O. Aceste adrese sunt relative la o adresă de bază I/O care poate fi selectată la inițializare (una din 16 adrese posibile).

Linia de întrerupere cu care lucrează circuitul poate fi programată ca una din 8 linii posibile.

Programarea adresei de bază de I/O, a dimensiunii memoriei BROM, linia de întrerupere se fac la inițializare prin jumperi. Desigur că unii parametri pot fi modificați prin scrierea registrelor de configurare. Circuitul selectează automat între transmisia 10BaseT sau AUI (Attachement Unit Interface). Mediul de transmisie (cablu coaxial sau cablu torsadat (UTP)) trebuie specificat cu jumperi.

Semnalele la pini pentru circuitul RTL 8019 se împart în mai multe categorii:

Semnale de interfață cu procesorul (compatibile ISA):

- AEN Address Enable;
- INT7-0 cereri de întrerupere, dintre care numai una este selectată la un moment dat. Circuitul citește aceste linii în registrul INTR pentru a monitoriza eventualele conflicte;
- IOCHRDY se introduc stări de WAIT care prelungesc ciclul de magistrală;
- IOCS16 selectează transferuri pe 8 sau 16 biți;
- IORB indică un ciclu de citire (din perspectiva procesorului);
- IOWB indică un ciclu de scriere (din perspectiva procesorului);
- RSTDRV - RESET (cu nivel 1 logic);
- SA19-SA0 magistrala de adrese;
- SD15-SD0 magistrala de date;
- SMEMRB indică un ciclu de citire din memorie (din perspectiva procesorului).

Interfața cu memoria:

- MA13-0 adrese pentru SRAM;
- MD7-0 date pentru SRAM;
- MRDB strob de citire pentru SRAM;
- MWRB strob de scriere pentru SRAM;

- MCSB selectare SRAM.
- BCSB selectare BROM;
- BA21-14 adrese pentru BROM;
- BD7-0 date pentru BROM;
- EECS selectare EEPROM;
- EESK ceas serial pentru EEPROM;
- EEDI date de intrare seriale pentru EEPROM;
- EEDO date de ieșire seriale pentru EEPROM;

Pini definiți pentru lucrul cu jumperi. Aceste semnale sunt citite la RESET:

- JP, se selectează modul de lucru cu jumperi. Dacă nu, se validează modul PnP (Plug and Play);
- PnP, în mod PnP se selectează dacă se ține cont de conținutul memorie EEPROM la configurare;
- BS4-0 selectează dimensiunea memoriei BROM;
- IOS3-0 selectează adresa de bază de I/O;
- IRQ2-0 selectează o linie de întrerupere din INT7-0.

Semnale de interfață cu mediul de transmisie:

- CD+, CD- semnalul diferențial de la AUI care indică coliziune;
- RX+, RX- semnalul diferențial de la AUI de date recepționate;
- TX+, TX- semnalul diferențial de la AUI de date emise;
- TPIN+, TPIN- date de intrare de la UTP (10Mbps, codate Manchester);
- TPOUT+, TPOUT- date de ieșire la UTP;
- X1 intrare de tact de la un generator extern sau cristal de 20Mhz;
- X2 la masă pentru generator extern sau la cristal.

Semnalizare:

- LEDBNC este 1 când se setează modul BNC;
- LED0 este 1 când se realizează conectarea;
- LED1, LED2 este 1 la recepție, respectiv la emisie;

Programarea circuitului

La adresa cu offset 00h se află registrul CR (Command Register) care conține 2 biți cu funcția de a selecta una din 4 pagini posibile cu registre de programare. Registrele de programare pot fi împărțite în registre de configurare compatibile NE2000 și registre PnP. În afară de registrele compatibile NE2000 sunt câteva registre pentru funcții speciale ale RTL 8019. Programarea prin registre compatibile NE2000 este importantă pentru că softul de programare este același pentru toate plăcile compatibile NE2000. Câteva din registrele mai importante sunt:

- CR (00h) selectează una din 4 pagini de registre și comandă citirea/ scrierea DMA a pachetului de date;
- ISR (07h, Interrupt Status Register) reflectă cererile de întrerupere datorate terminării transferului sau apariției unor erori;
- IMR (0Fh, Interrupt Mask Register) poate masca întreruperile din ISR;
- DCR (0Eh, Data Configuration Register) stabilește ordinea octeților în cuvânt și modul de operare în buclă de test sau normală;
- TCR (0Dh, Transmit Configuration Register) programează modul de transmisie normal sau în buclă de test și generarea CRC;
- TSR (04h, Transmit Status Register) indică starea pachetului de date transmis-transmisie terminată sau întreruptă din cauza unei coliziuni;

- RSR și RCR (0Ch, Receive Configuration (Status) Register) indică starea pachetului recepționat- adresa destinației corespunde sau nu, pachet acceptat sau nu, erori de transmisie (CRC), erori de ritm;
- CLDA0 și CLDA1 (01h, 02h, Current Local DMA Register) arată adresa DMA curentă;
- PSTART (Page START Register) stabilește adresa de START a paginii unde se află datele recepționate;
- PSTOP (Page STOP Register) stabilește adresa de STOP a paginii unde se află datele recepționate;
- TPSR (Transmit Page START Register) la fel ca PSTART dar pentru transmisie;
- TBCR0,1 (Transmit Byte Count Register) numărător de octeți pentru datele transmise;
- NCR (Number of Collisions Register) numărul de coliziuni;
- FIFO primul octet dintr-o stivă care conține datele citite;
- PAR0-5 (01h-06h în pagina 1, Physical Address Registers) conține adresa proprie Ethernet folosită pentru a compara adresa conținută în pachetul de date;
- CURR (07h în pagina 1, Current Page Register) adresa primei pagini de recepție;

Registre de configurare a RTL 8019:

- EEPROM CR (01h, pagina 3, EEPROM Command Register)
- BPAGE (02h, pagina 3, BROM Page Register)
- CONFIG0,1,2,3 (03h-06h, pagina 3, RTL Configuration Register) selectează mediul de transmisie, linia de cerere de întrerupere către procesor, adresa de bază de I/O (300h, 320h,... 3E0h sau 200h, 220h,... 2E0h), dimensiunea memoriei BROM și adresa de bază pentru BROM (C000h, 32K, etc.) indicația LED-urilor, comanda modurilor de economie de energie.

Pentru transmisia unui cadru, informația de transmis se înscrie în RTL 8019 în memoria proprie prin transfer de memorie la adresa la care RTL 8019 a fost configurat. Transmisia se face prin DMA sau prin adresare și înscrierea locației pe magistrala ISA sau prin înscrierea datelor în zona de I/O a RTL 8019 la un port I/O. Recepția datelor se face prin transfer programat de memorie pe magistrala ISA, prin DMA pe magistrala ISA sau la un port I/O sau prin citirea registrului FIFO care este vârful unei stive în care se află cadrul recepționat.

Conectarea memoriei externe la RTL 8019 și conectarea circuitului la magistrala ISA se realizează ca în figura 2.2:

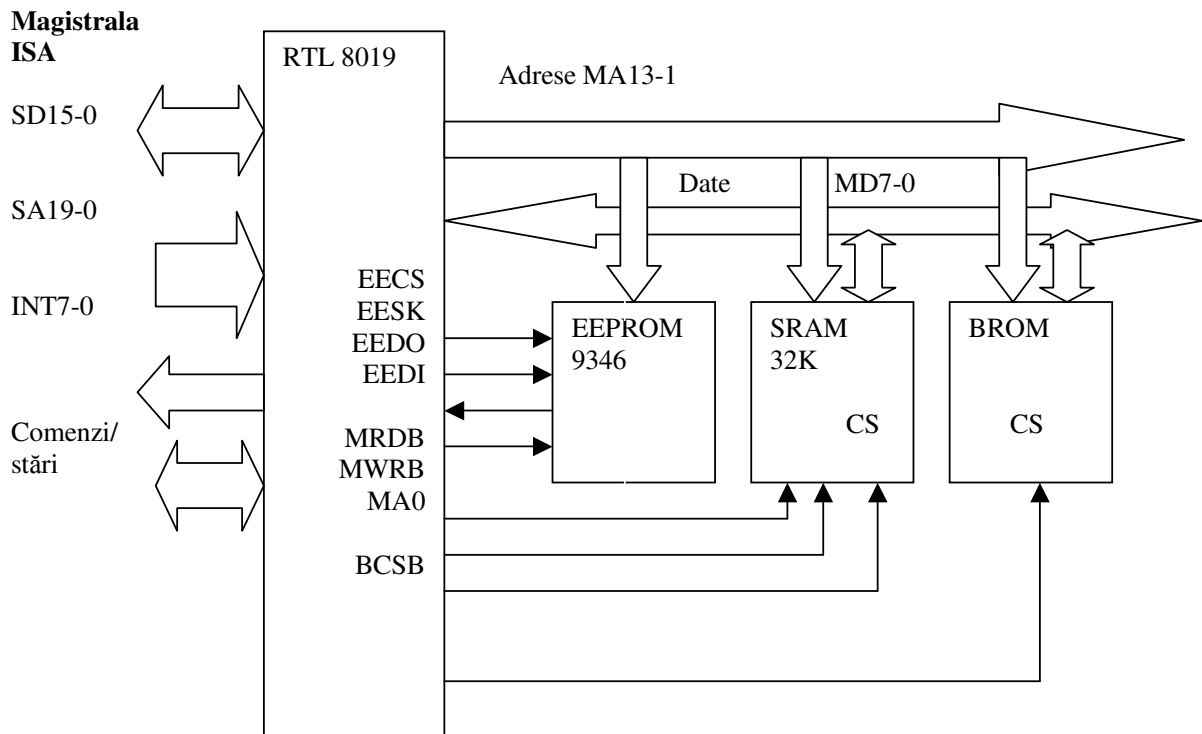


Fig. 2.2. Conectarea memoriei externe la RTL 8019

Selecția EEPROM se face cu semnalul EECS generat de RTL 8019, iar transferul de date are loc serial prin EEDO și EEDI. Memoria SRAM este selectată de linia de adresă MA0 iar sensul transferului este dat de semnalele de RD și WR (MRDB și MWRB). Memoria BROM din care se poate doar citi este selectată de semnalul BCSB.

O memorie SRAM ca buffer de date de 16Kocteți este inclusă în majoritatea modelelor de RTL 8019. În această memorie internă, controlabilă prin jumperi se pot stoca mai multe cadre. Memoria externă SRAM este opțională, la fel și memoria BROM. Dacă este suficientă configurarea prin jumperi și memoria EEPROM poate lipsi. Circuitul poate transfera date atât pe 8 cât și pe 16 biți. Configurația fără memorii externe și cu conectarea pe 8 biți este folositoare pentru aplicații cu microcontrollere.

Noțiunea de cadru Ethernet și adresă proprie plăcii sunt prezentate sumar în continuare.

Structura unui cadru de date la transmisia Ethernet

Protocolul MAC (Media Access Control) este folosit pentru implementarea nivelului Legătură de Date în tehnologia Ethernet. Protocolul MAC încapsulează datele adăugând un antet de 14 bytes în fața datelor și 4 bytes de informație de control (CRC) la sfârșitul datelor. Întregul cadru de date este precedat de scurtă perioadă de 9.6 microsecunde (la 10 Mbps) și un preambul de 8 bytes.

Formatul unui cadru de date Ethernet este format din următoarele câmpuri:

- Preambulul- Orice transmisie de cadru Ethernet începe cu o secvență de 8 bytes. Această secvență constă din 62 de biți alternați de 1 și 0 care sunt urmați de doi biți de 1. Scopul acestui câmp este de a permite receptorului sa-și sincronizeze ceasul de recepție a datelor cu ceasul de transmisie.
- Delimitatorul de început de cadru- Ultimul byte din preambul care se termină cu doi biți de 1 anunță începutul unui cadru de date. La primirea codului special "11" interfața Ethernet de recepție tratează următorii biți ca date.
- Adresa sursă și adresa destinație- Aceste adrese sunt de 6 bytes fiecare și reprezintă adresele fizice (adrese MAC) ale transmițătorului și receptorului.
- Tip- Prin acest câmp este specificat protocolul superior care va prelua datele după procesarea Ethernet (valoarea 0x0800 identifică protocolul IP, acesta fiind și cel mai folosit).
- Data- După procesarea Ethernet aceste date sunt trimise protocolului superior specificat în câmpul Tip. Lungimea minimă a acestui câmp este de 46 bytes și cea maximă este de 1500 bytes. Orice depășire a acestor limite este considerată eroare.

- CRC- acest câmp poartă și denumirea de FCS – Frame Check Sequence și este calculat de către sursă și recalculat la destinație pentru detectarea eventualelor erori survenite în timpul transmisiei. Dacă se detectează vreo eroare cadrul de date este abandonat. Protocolul MAC nu oferă nici un mecanism prin care să se indice sursei că un anumit cadru de date a fost abandonat din cauza apariției unei erori.

Trebuie menționat faptul că, fiind vorba de o topologie de tip multipunct (broadcast), cadrele Ethernet din rețea sunt preluate de către toate calculatoarele conectate la rețea. După efectuarea verificărilor privind corectitudinea cadrului, se verifică dacă acel cadru este destinat calculatorului respectiv. Verificarea se face pe baza adresei MAC de destinație. Această adresă poate fi chiar adresa MAC a calculatorului respectiv, și atunci cadrul este procesat, adresa MAC de broadcast, și în acest caz cadrul fiind procesat, sau o altă adresă MAC, și atunci cadrul este abandonat.

Deoarece tehnologia Ethernet implementează funcțiile celor două nivele OSI, Legătură de Date și Fizică, tot aici trebuie implementată și metoda de adresare fizică. Protocolul MAC conține și această schemă de adresare. Această adresă fizică, ca și întreaga tehnologie Ethernet, este implementată în interiorul interfeței de rețea.

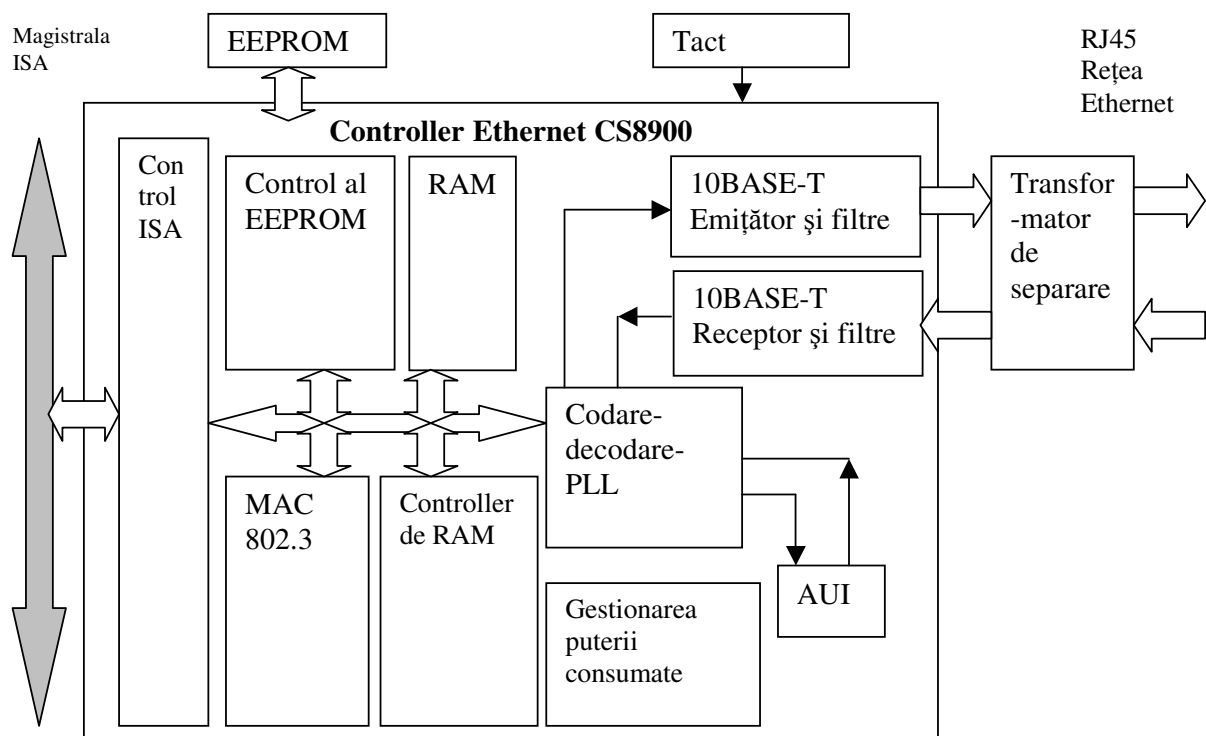
Adresa fizică constă dintr-un număr de 48 de biți scris în hexazecimal care identifică în mod unic în lume o interfață de rețea. Această unicitate este dată de primii 24 de biți care reprezintă un număr ce identifică producătorul interfeței. Acest număr este denumit OUI – Organizational Unique Identifier și este administrat și distribuit producătorilor de către IEEE. Ceilalți 24 de biți reprezintă numărul de serie al interfeței și este administrat de către producător.

Aceste adrese MAC mai poartă și denumirea de adrese BIA's – Burned-in Addresses deoarece ele sunt înscrise într-o memorie ROM de pe interfața de rețea.

2.2.Circuitul interfață de rețea CS8900A (producător CIRRUS LOGIC)

Circuitul CS8900 este o interfață pentru rețea Ethernet care încorporează toate circuitele digitale și analogice necesare conectării la rețea pe de o parte și la magistrala ISA pe de altă parte.

Schema bloc a circuitului CS8900 este dată în figura 2.3.:



Blocurile principale sunt:

1. Interfața cu magistrala ISA, toate liniile de interconectare având capacități de încărcare standard. Circuitul poate cere o întrerupere pe una din patru linii și poate cere un transfer DMA pe una din trei linii. Liniile se selectează la inițializarea circuitului.

2. Memoria internă (4K octeți) face ca o memorie externă să nu mai fie necesară. Circuitul introduce în această memorie un întreg cadru de trimis sau recepționat. Accesul la memoria proprie se poate face fără stări de WAIT.

3. Modulul MAC (Media Access Control) de acces la rețea asigură accesul conform standardului IEEE 802.3 în mod full duplex. MAC se ocupă de toate aspectele legate de transmisia cadrului cum ar fi detecția de coliziuni, generarea și detectarea preambului, generarea și verificarea CRC. Modulul MAC retransmite automat cadrul după detectarea unei coliziuni.

4. Interfața cu EEPROM este necesară pentru a citi un EEPROM serial opțional care conține datele de configurare ale circuitului. Se elimină astfel microîntrerupătoarele sau jumperii care sunt destul de scumpi.

5. Interfața analogică cu rețeaua conține codorul și decodorul Manchester, circuitul de refacere a tactului din semnalul recepționat (cu PLL), transceiver pentru 10BASE-T și o interfață AUI (Attachement Unit Interface). Selecția între AUI și 10BASE-T se face automat sau manual. Trei LED-uri arată starea circuitului: starea ON sau OFF a legăturii, activitatea Ethernet și starea magistralei. Transceiverul 10BASE-T conține emițătoare și receptoare de linie și filtre analogice, în exterior fiind necesar doar un transformator de separare. Sunt suportate cabluri cu impedanța caracteristică de 100, 120 și 150Ω, ecranate sau neecranate.

Semnalele la pini:

Semnale de interfață cu ISA: SA0-19 (adrese), SD0-15 (date), RESET, AEN, /MEMR, /MEMW, /REFRESH (indică un ciclu DMA de reîmprospătare pe ISA), /IOR, /IOCS16, /IOCHRDY, SBHE, INTRQ0-3, DMARQ0-2, DMACK0-2, ALE.

Semnale de interfață cu memoria EEPROM și BROM: EESK, EEDATAIN, EEDATAOUT, ELCS (CS pentru EEPROM), CSOUT (CS pentru BROM).

Semnale de interfață cu mediul de transmisie: TXD+, TXD-, RXD+, RXD- pentru 10BaseT, DO+, DO-, DI+, DI-, CI+, CI- pentru AUI.

Pini de uz general: XTAL conectare cristal, SLEEP comandă intrarea în modul economie de energie, LINK LED arată conectarea, LAN LED arată activitatea în rețea, TEST pentru testarea la producător.

Funcționarea circuitului are la bază 2 funcții: să trimită un cadru Ethernet și să recepționeze un cadru. Înainte de emisie sau recepție, circuitul trebuie configurat.

Spre deosebire de RTL 8019, CS 8900 nu gestionează liniile de date și adrese ale memoriei externe, fiind de aceea un circuit cu mai puține terminale și fiind astfel mai ușor de folosit.

O schemă bloc de cuplare pe ISA și gestionarea memoriei este dată în figura 2.4.:

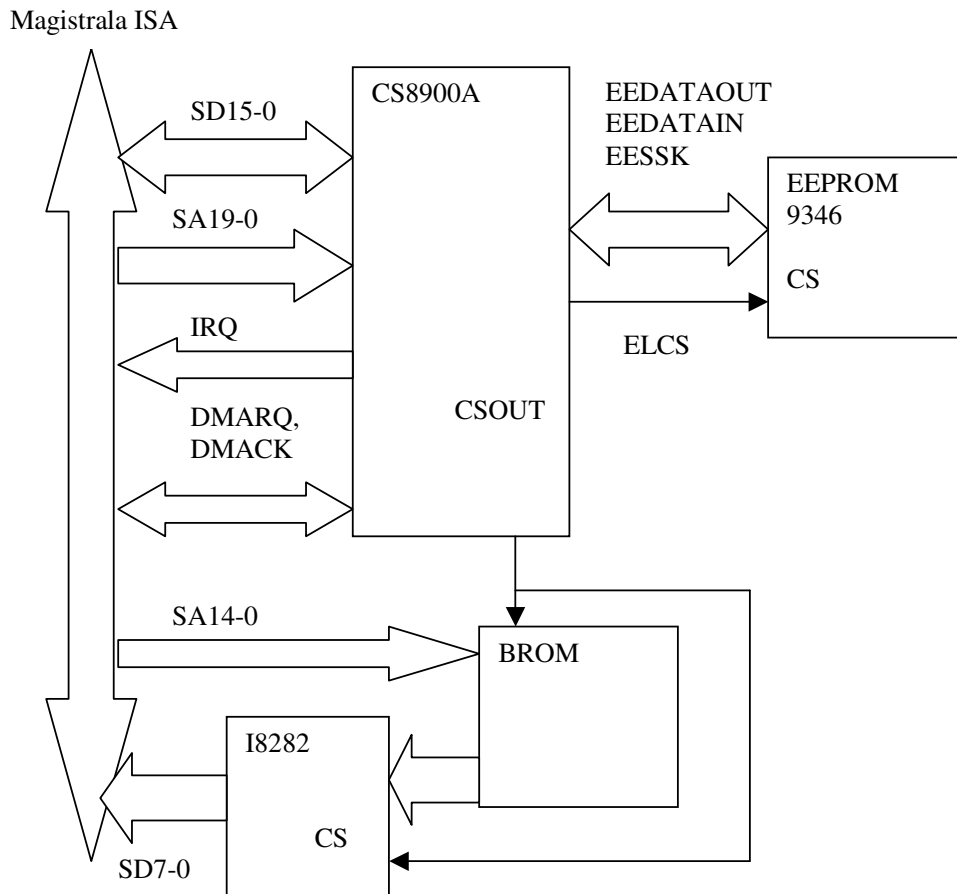


Fig. 2.4. Schema bloc de cuplare a CS8900 pe ISA și gestionarea memoriei

Configurarea circuitului se poate face prin scrierea /citirea registrelor interne de configurare sau datele de configurare pot fi încărcate dintr-un EEPROM serial.

Transmisia unui cadru începe cu o comandă (Transmit Command) prin care se precizează când să înceapă transmisia (după ce toți octeții au fost transmiși în CS8900 de exemplu), existența CRC etc. Se trimite apoi lungimea cadrului, (Transmit Length), după care se trimit datele în CS prin transfer de memorie în memoria proprie sau prin transfer I/O. Circuitul transmite cadrul în rețea începând cu un preambul urmat de adresa destinației, adresa sursei, date, octeți de CRC (se pot citi amănunte în paginile anterioare, la structura cadrului Ethernet).

Recepția unui cadru este realizată de CS și cadrul este stocat în memorie. Recepția se face prin decodare Manchester, apoi sunt eliminate preambulul, adresa destinatarului este verificată și dacă corespunde cu adresa programată în CS cadrul este memorat și se anunță procesorul printr-o întrerupere. În a doua fază datele din memorie se transferă în memoria sistemului pe magistrală, prin transfer de memorie, transfer I/O sau prin DMA.

Transferul de memorie se face prin accesul direct al procesorului la memoria internă a CS, adresată prin ISA de liniile de adresă, sensul transferului fiind dat de MEMR sau MEMW. Registrele de configurare pot fi accesate și ele în acest mod. La transferul I/O, CS este accesat prin 8 registre de 16 biți, văzute de procesor în spațiul de I/O. Sensul transferului este dat de IOR și IOW. Acest al doilea mod (I/O) este ales implicit la pornirea circuitului (sau la RESET).

Programarea circuitului CS8900 se face printr-un concept original de programare, prin intermediul memoriei interne numită PacketPage. Accesul la această memorie poate fi realizat atât prin transfer cu memoria cât și I/O. Transferul cu memoria este preferat deoarece pe ISA ciclul de memorie este cu un tact mai scurt decât cel de I/O. Conectarea lui CS8900 la un microcontroller face imposibilă tratarea PacketPage prin transfer de memorie. Memoria

PacketPage este utilizată pentru stocarea temporară a unui cadru. PacketPage are următoarea structură:

- Adresa 0000h-0045h, registrele de interfață cu magistrala;
- Adresa 0100h-013Fh, registrele de stare și control;
- Adresa 0140h-014Fh, registrele de inițiere a transmisiei;
- Adresa 0150h-015Dh, registrele de selecție a adresei circuitului;
- Adresa 0400h, vârful unei stive pentru cadrul recepționat;
- Adresa 0A00h, vârful unei stive pentru cadrul de transmis.

2.3. Conectarea circuitelor CS8900 și RTL 8019 cu microcontrollere

Circuitul CS8900A poate fi folosit cu multă ușurință prin cuplarea cu un microcontroller. În cazul folosirii unui microcontroller AT89S53 (pe 8 biți) transferul se face pe 8 biți. CS8900 poate lucra în acest mod cu unele restricții. Drivere pentru lucrul pe 16 și pe 8 biți sunt disponibile pe pagina de la Cirrus Logic.

Conectarea circuitului CS8900 la microcontrollerul AT89S53 este prezentată în Schema 1 și Schema 2. Legătura de date între circuite este realizată printr-o magistrală de date de 8 biți formată de portul P0 al microcontrollerului. Magistrala de date permite cuplarea și a altor circuite, care astfel pot comunica cu rețeaua, transferul fiind comandat de microcontroller. Se poate astfel realiza de exemplu un server de imprimantă care, cuplat la rețea și având un număr propriu de IP să permită cuplarea unei imprimante direct la rețea, fără un sistem de calcul.

Sensul transferului este comandat de semnalele /RD și /WR (P3.6 și P3.7). Cu un latch de adrese se memorează liniile de adresă, care sunt generate multiplexat cu cele de date la portul P0 al microcontrollerului. Latchul este comandat de semnalul ALE. Se obțin liniile de adrese A0, A1, A2, A3 necesare pentru selecția registrelor interne ale CS8900 și rămân libere liniile de adrese A4-A7 cu care se pot selecta diferite circuite cuplate pe magistrala de date. Circuitul CS8900 este selectat cu linia A7. Un semnal de RESET format în Schema 2 inițializează atât microcontrollerul cât și controllerul de rețea la punerea sub tensiune sau la apăsarea unui buton.

Microcontrollerul are prevăzute 2 interfețe pentru comunicația cu sistemul gazdă:

- Interfața serială RS232 pentru transferul de date. De la microcontroller spre RS232 este conectat un circuit pentru modificarea de nivel MAX232;
- Interfața serială sincronă cuplată la portul CENTRONICS al sistemului gazdă pe 4 biți (date de intrare, de ieșire, tact și validare).

Două generatoare de tact furnizează tactul pentru microcontroller (10MHz) și pentru controllerul de rețea (20MHz).

La controllerul de rețea se adaugă un minim de componente externe:

- Transformator de izolare pentru 10BASE-T (cablu UTP);
- 2 LED-uri de semnalizare.

În modul pe 8 biți se poate lucra cu CS8900A prin intermediul a 8 registre de 16 biți (citite sau scrise în 2 pași), selectate cu biții de adresă A0, A1, A2 și A3. Cele 8 registre sunt:

Adresa	Tip	Descriere
0000h	R-W	Date recepționate- Date transmise (Port 0)
0002h	R-W	Date recepționate- Date transmise (Port 1)
0004h	Numai W	TxCMD (Comanda transmisă)
0006h	Numai W	TxLenght (Lungime transmisă)
0008h	Numai R	Starea întreruperilor
000Ah	R-W	Pointer de PacketPage

000Ch	R-W	Date PacketPage (Port 0)
000Eh	R-W	Date PacketPage (Port 1)

* pentru o citire corectă a unui registru este nevoie de citirea ambilor octeți.

Restricțiunile care apar în mod 8 biți sunt:

- la accesul pe 8 biți memoria poate fi văzută doar în spațiul de I/O.
- nu se poate lucra în întreruperi, este nevoie de o testare software
- nu se poate lucra în mod DMA
- modul PacketPage are în mod 16 biți posibilitatea de autoincrementare, care în mod 8 biți nu poate fi folosită
- nu este posibilă conectarea unui EEPROM, este nevoie să se folosească anumite valori implicite la inițializare.

Secvențele mai importante de program pentru interconectarea cu AT89S53 sunt date în continuare:

1. Transmisia unui cadru

-se transmite o comandă de rezervare a spațiului din memoria internă pentru a încărca cadrul de transmis, memoria devenind astfel un buffer de transmisie. Comanda se transmite la TxCMD (04h și 05h).

MOVX 04h C0

MOVX 05h 00 (Se pornește transmisia automat după ce s-au încărcat toți octeții)

-se transmite lungimea cadrului (număr de octeți) la TxLenght (06h și 07h).

MOVX 06h 81d

MOVX 07h 00 (Se stabilește lungimea cadrului de 81 de octeți)

-se verifică dacă spațiul de memorie este disponibil. Pentru aceasta se scrie în pointerul de PacketPage adresa registrului BusST (stare Bus) (0138h):

MOVX 0Ah 38

MOVX 0Bh 01

Se citește registrul de stare BusST în registrele PacketPage (0C, 0D, 0E, 0F)

MOVX (se citește octetul de la Port 0 adresa 0Ch. Dacă bitul 8 Rdy4TxNow este 1 se poate începe transmisia datelor).

-se transmit datele:

MOVX 00h primul octet

MOVX 01h al doilea octet

MOVX 02h al treilea octet

.....

2. Recepția unui cadru

-se urmărește recepția unui cadru prin polling, testând registrul Rx Event prin citirea portului 0 de date, întâi octetul mai semnificativ (01h) apoi octetul mai puțin semnificativ (00h), în buclă.

MOVX de la 01h

MOVX de la 00h

Testare

-când se semnalizează primirea unui cadru se citește RxLenght prin citirea portului 0 de date, întâi octetul mai semnificativ (01h) apoi octetul mai puțin semnificativ (00h).

MOVX de la 01h

MOVX de la 00h

-se citesc datele din cadrul recepționat prin citirea portului 0 de date, întâi octetul mai puțin semnificativ (00h) apoi cel mai semnificativ (01h).

bucla:

MOVX de la 00h

MOVX de la 01h

Avantajele circuitului CS8900 sunt un număr mai mic de pini, documentație mai clară, modul de programare mai simplu datorită conceptului de PacketPage. Dezavantajul principal este că se procură greu.

O altă variantă de cuplare (arătată în Schema 3) este cu microcontroller RISC AT90S și controller de rețea RTL8019.¹ Diferența este că nu se creează o magistrală de date și adrese, deci nu se pot cupla și alte circuite. Altfel schema este asemănătoare. La microcontroller apar cele 2 interfețe de transfer de date RS232 cu adaptare de nivel (MAX232) și cea serială sincronă pentru programare cuplată la CENTRONICS. La controllerul de rețea apare transformatorul de izolare și LED-urilor de semnalizare. Transferul de date are loc prin portul PC al microcontrollerului iar selecția registrelor interne pentru RTL8019 precum și precizarea sensului transferului se realizează cu semnale din portul PA.

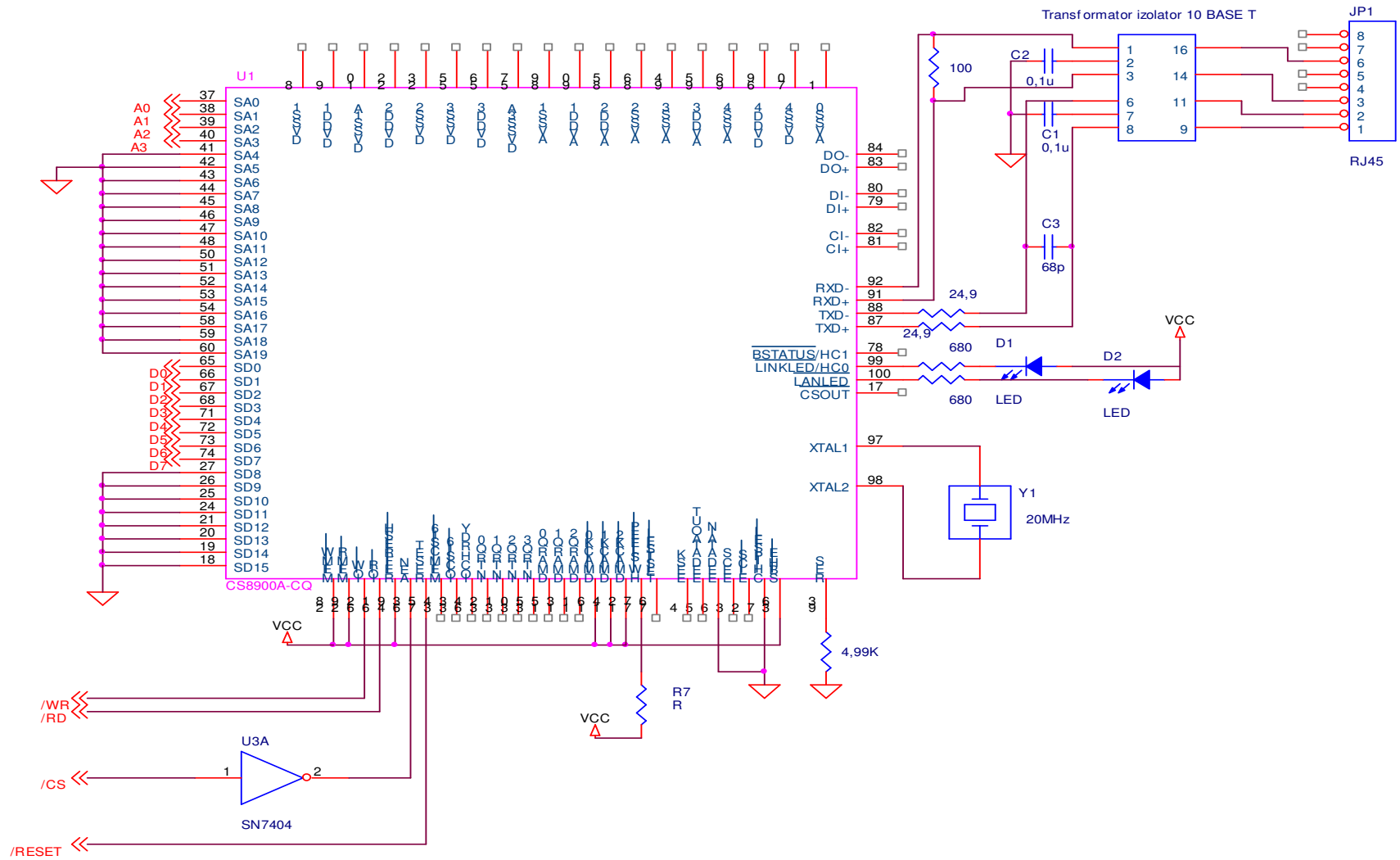
Un circuit RTL8019 se poate conecta cu un microcontroller din familia 8051, soluția fiind menționată la linkuri utile.

Ca avantaje ale folosirii circuitului RTL se menționează:

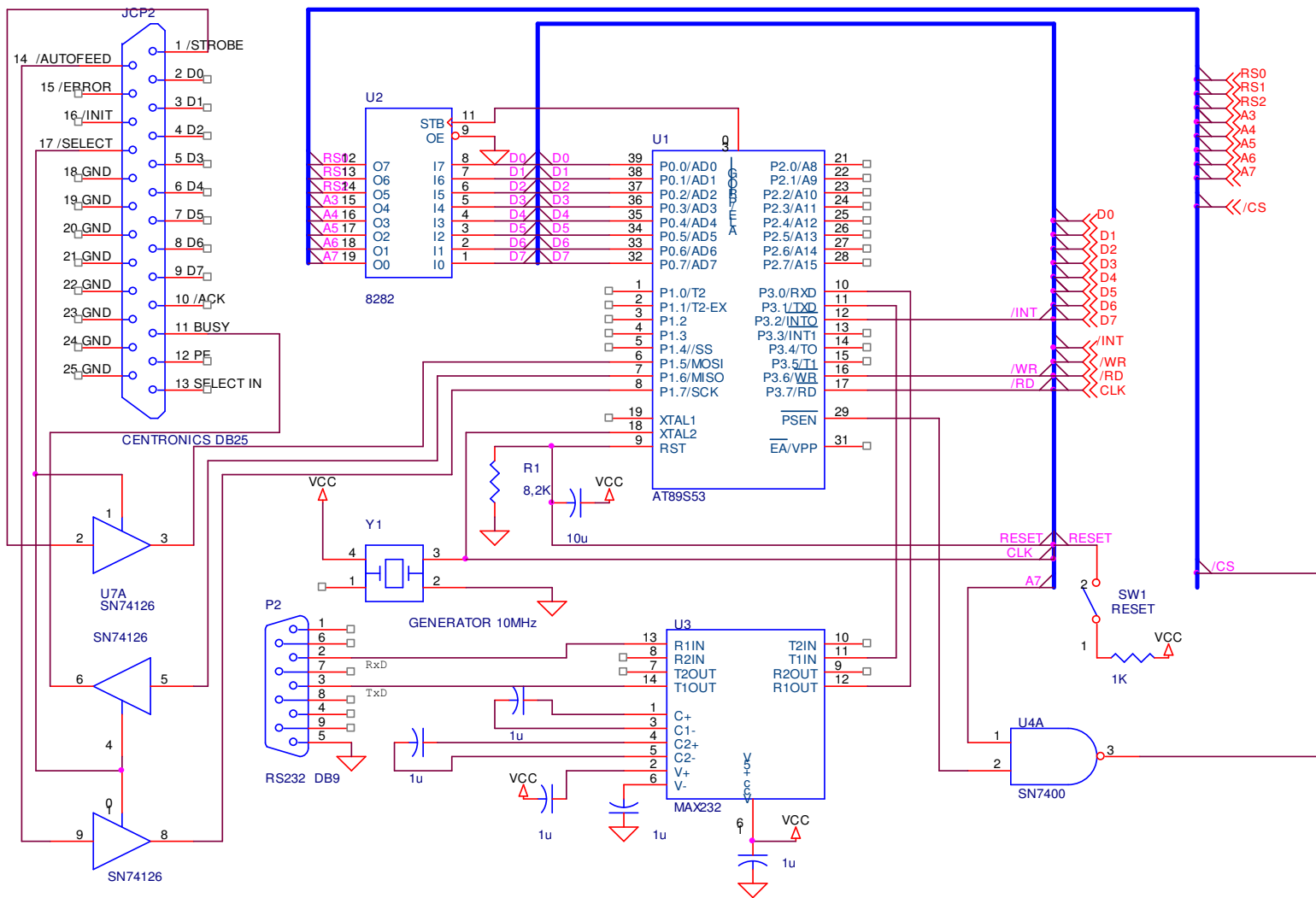
- Memoria RAM internă de 16 Kocteți asigură un număr mic de componente externe circuitului (ca și la CS8900);
- Pentru aplicații TCP-IP este necesară memoria EEPROM externă pentru a stoca codul IP al aplicației;
- Alegerea automată a mediului de transmisie;
- Circuitul este compatibil NE2000/NE1000, oferind posibilitatea accesului pe 8 biți.

Ca dezavantaj se menționează capsula circuitului (100 de pini) destul de dificil de lipit în condiții ne-industriale. În Internet se oferă link-uri către foile de catalog ale variantelor de circuite, software pentru 8051, sfaturi de conectare.

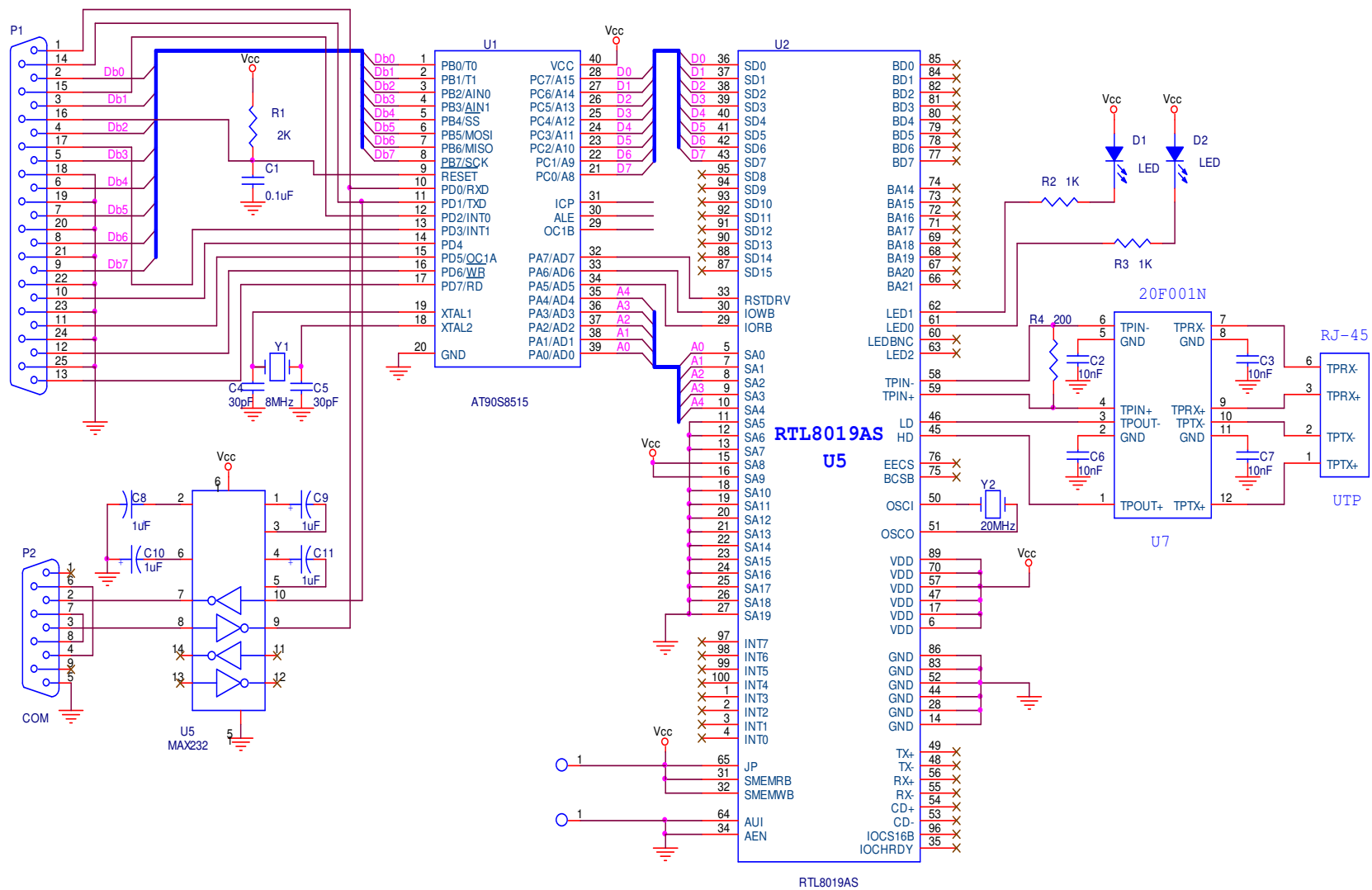
¹ Prin amabilitatea dl. Ing. Ciocea Daniel de la Siemens Automotive Timișoara



Schema 1 Circuitul CS8900 și componentele externe pentru cuplarea la AT89S53



Schema 2 Schema de conectare a circuitului AT89S53 pentru legarea cu CS8900



Schema 3 Conectarea unui circuit RTL8019 la un microcontroller RISC (AT90S8515)

2.4.Circuitul stivă TCP-IP SEIKO S-7600

Pentru a înțelege utilitatea acestui circuit este nevoie de o descriere sumară a protocolului TCP-IP și UDP.

Protocolul TCP/IP²

TCP/IP –ul este un set de protocole de comunicație prin rețea. Deși numele de TCP/IP implică un întreg șir de semnificații, el este o combinație a celor două protocole de bază: TCP – Transmission Control Protocol și IP – Internet Protocol

Modelul TCP/IP are patru nivele :

- Nivelul **Aplicație**. Proiectanții modelului TCP/IP au decis că protocolele de nivel înalt trebuie să includă și detaliile nivelului Prezentare și Sesiune. Ei au creat astfel un singur nivel superior care se ocupă de reprezentare, codare și controlul dialogului.
- Nivelul **Transport**. Acest nivel se ocupă de noțiunea de calitate a serviciilor de rețea (QoS – Quality of Services) și de siguranța acestora, de controlul fluxului și corecția de erori. Modelul TCP/IP are implementate două protocole de transport (TCP și UDP) fiecare având propriile caracteristici.
- Nivelul **Internet**. Scopul acestui nivel este de a trimite fiecare pachet de date din orice rețea către orice altă rețea interconectată cu aceasta independent de calea pe care o urmează. Protocolul care implementează aceste funcții este protocolul IP – Internet Protocol. La acest nivel se realizează alegerea căii precum și comutarea pachetelor.
- Nivelul **Acces Rețea**. Nivelul implementează toate funcțiile necesare pentru ca un pachet IP să poată fi transmis prin mediul fizic de transmisie. Aici sunt incluse detalii pentru tehnologii LAN și WAN precum și toate detaliile care apar în nivele OSI Legătură de Date și Fizic.

UDP (User Datagram Protocol) este un protocol de transport fără confirmare, ceea ce îl face să nu ofere siguranță , dar care este folosit în scopuri speciale care nu necesită în mod obligatoriu siguranța datelor.

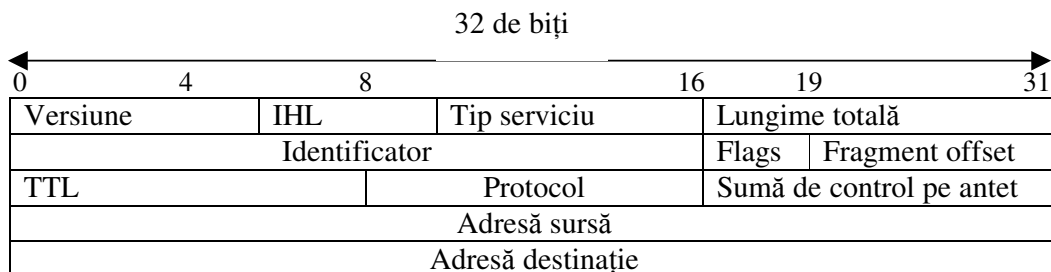
Protocolul IP (Internet Protocol) este responsabil pentru transferul prin rețea al pachetelor de date primite de la protocolele de transport TCP sau UDP. Pentru acest lucru folosește o adresă unică pentru fiecare calculator conectat în rețea.

La fel ca și un cadru de date, datagrama este divizată în două mari componente: antetul și datele. Tot similar cu cadrul da date, antetul datagramii conține adresele sursă și destinație și un câmp “TIP” care identifică conținutul datagramii.

În figura de mai jos este arătat formatul complet al unei datagramme IP:

Descrierea câmpurilor datagrammei (figura 2.5):

- Versiune, acest câmp de 4 biți indică versiunea de IP folosită.
- IHL – Internet Header Length, câmpul IHL de 4 biți indică lungimea antetului datagrammei specificat în cuvinte de 32 de biți. Cel mai mic antet poate fi de 20 bytes dar folosirea câmpurilor opționale pot mări lungimea antetului până la 24 bytes.
- Tip Serviciu (Type of Service), acest câmp de 8 biți informează IP –ul de modul cum trebuie procesată datagramă.
- Lungime totală, prin acest câmp se specifică lungimea totală a datagrammei dată în bytes, incluzând atât antetul cât și datele.
- Identificator, câmpul Identificator împreună cu următoarele două câmpuri este folosit pentru fragmentare / reasamblare. Acest câmp conține un număr care identifică în mod unic datagrama respectivă. Acest număr este necesar la reasamblarea mesajelor fragmentate pentru a se asigura de faptul că fragmentele unui mesaj nu vor fi încurcate cu ale altuia.



² Prin amabilitatea dl. Ing. Dan Ovidiu de la SIEMENS PSE Brașov

Opțiuni
Date

Fig.2.5. Câmpurile datagramei IP

- Flags, acest câmp de trei biți este folosit la fragmentare / reasamblare.
- Fragment Offset, dacă un bitul (MF) din câmpul precedent este setat indicând fragmentarea unei datagramme, câmpul va conține poziția pe care o ocupă acest fragment în datagrama originală. În acest fel se permite IP –ului să reasambleze fragmentele în ordinea corectă.
- TTL – Time To Live, în principiu, câmpul TTL specifică timpul în secunde care îi este permis unei datagramme să circule prin rețea. În momentul în care un calculator trimite o datagramă în rețea, automat se setează și timpul de viață al datagrammei respective. Toate stațiile intermediare care procesează datagrama trebuie să decrementeze valoarea acestui câmp. În momentul în care această valoare ajunge la 0, routerul trebuie să înlăture această datagramă din rețea și să genereze un mesaj de eroare către sursa ei. Folosirea acestui timp de viață a datagrammei este foarte importantă deoarece garantează faptul că nici o datagramă nu va circula la infinit prin rețea din cauza unei rutări deficitare (rutare ciclică).
- Protocol, acest câmp specifică un număr corespunzător protocolului de nivel superior căruia îi sunt adresate datele.
- Suma de control pe antet, prin acest câmp se asigură integritatea informației din antet (nu și a datelor).
- Adresa sursă și destinație, aceste câmpuri conțin adresele IP de 32 de biți ale stațiilor sursă și destinație. Aceste câmpuri sunt înscrise la sursă și nu suferă nici o modificare în timpul rutării.
- Opțiuni, acest câmp este opțional și este folosit pentru testarea și înlăturarea problemelor din rețea.

Adresarea logică. Adresa IP

Există o metodă de adresare fizică a fiecărui calculator din rețea prin adresa MAC. Însă, această modalitate de adresare nu oferă o modalitate fiabilă de rutare a pachetelor pentru dispozitivele de decizie (router). S-a impus astfel crearea unei scheme de adresare ierarhice, la nivelul IP, care să asigure o mai bună structurare în modalitățile de decizie.

Adresa logică, sau adresa de IP, este număr de 32 de biți care este scris în așa-numita notație cu punct (“dot notation”) unde fiecare byte este despărțit de următorul printr-un punct. În structura unei adrese de IP există două componente, o componentă care identifică rețeaua și o componentă care identifică stația căreia îi este atribuită respectiva adresă.

Protocolul UDP – User Datagram Protocol

În situa de protocoale TCP/IP, protocolul UDP oferă mecanismul primar folosit de aplicații pentru a trimite datagramme către alte aplicații.

Principalul termen pe care-l introduce protocolul UDP este termenul de “port” care reprezintă o aplicație. Fiecare port este identificat printr-un număr întreg pozitiv. Sistemul de operare local oferă un mecanism de interfață prin care un proces poate specifica un port sau îl poate accesa. Pentru a comunica cu un port de pe altă stație, emițătorul trebuie să cunoască atât adresa de IP a destinației cât și numărul portului de pe mașina de destinație.

Protocolul UDP oferă un număr de porturi pentru a distinge multiplele programe care se execută pe o singură mașină. De aceea, pe lângă datele care vor fi trimise, fiecare mesaj UDP va conține un număr de port de destinație și un număr de port sursă, făcând posibilă, pentru softul UDP de la destinație, livrarea mesajului către aplicația corectă.

UDP –ul nu folosește confirmări pentru a se asigura că mesajul a ajuns la destinație, nu reordonează mesajele primite și nu oferă nici un mecanism de control al fluxului. De aceea, mesajele UDP se pot pierde, pot fi duplicate sau pot ajunge la destinație în dezordine. Mai mult decât atât, pachetele pot veni mult mai repede decât poate procesa stația respectivă.

O aplicație care folosește UDP –ul trebuie să accepte întreaga responsabilitate pentru manipularea problemelor legate de siguranță, incluzând pierderea mesajelor, duplicarea lor, întârzierea, livrarea în dezordine și pierderea conectivității.

Fiecare mesaj UDP este numit datagramă. Din punct de vedere conceptual, o datagramă este compusă din două părți: un antet UDP și o zonă de date UDP. După cum se arată în figura 2.6. antetul este divizat în patru câmpuri de 16 biți care specifică porturile folosite (sursă și destinație), lungimea mesajului și o sumă de control.

0	16	31
PORT SURSĂ UDP	PORT DESTINAȚIE UDP	
LUNGIMEA MESAJULUI	SUMĂ DE CONTROL	
DATE		
.....		

Fig. 2.6. Datagrama UDP

Câmpurile care specifică porturile conțin numerele de porturi, pe 16 biți, folosite pentru a demultiplexa datagrama între procese care așteaptă să o primească. Portul sursă este opțional. Când este specificat, el semnifică portul unde trebuie trimis un eventual răspuns, iar dacă nu este specificat trebuie să fie zero.

Câmpul LUNGIME conține numărul de octeți din datagrama UDP, incluzând atât antetul cât și datele. Deci, cea mai mică lungime este de 8 octeți, adică exact lungimea antetului. Suma de control este un câmp opțional. Dacă acest câmp conține valoarea 0 înseamnă că suma de control nu a fost calculată.

Protocolul TCP – Transmission Control Protocol

Al doilea protocol de transport este protocolul TCP . Spre deosebire de UDP, protocolul TCP este un protocol care oferă un serviciu de transport sigur, orientat pe conexiune, cu proceduri de retransmisie și de control al fluxului.

Principalele caracteristici ale protocolului TCP sunt :

- *Stream Orientation* – transmisia este orientată pe flux de biți.
- *Virtual Circuit Connection*. Transferarea unui flux de biți este similară cu efectuarea unui apel telefonic. Înainte de începerea transmisiei, atât emițătorul cât și receptorul trebuie să interacționeze cu propriile sisteme de operare informându-le de dorința de a transfera fluxul de biți. Modulele software ale protocolului de pe cele două stații comunică între ele prin intermediul unor mesaje și verifică dacă transferul este autorizat. După au fost puse la punct aceste detalii, aplicațiile sunt anunțate că a fost stabilită o conexiune și că transferul poate începe. În timpul transferului, ambele module de protocol de pe cele două mașini continuă să comunice pentru a verifica dacă datele sunt transferate corect. Dacă apare vreo problemă, ambele mașini o vor detecta și vor raporta aplicațiilor acest lucru. O asemenea conexiune poartă numele de circuit virtual.
- *Full Duplex Connection*. Conexiunile realizate de protocolul TCP permit transferul concurrent în ambele direcții. O asemenea conexiune poartă numele de conexiune full duplex.
- *Acknowledgement and Retransmission*. Această tehnică de confirmări și retransmisie conferă protocolului TCP calitatea de serviciu de transport sigur. Prin acest mecanism se cere receptorului să comunice cu sursa, trimițând un mesaj de confirmare (Acknowledgement – ACK) imediat după primirea datelor. Astfel, la sursă se ține o evidență a pachetelor trimise, iar în cazul în care nu se primește confirmare pentru un anumit pachet acesta se va retransmite. De asemenea, pentru fiecare pachet trimis se pornește un timer care, dacă expiră înaintea venirii unei confirmări de primire pentru acel pachet, acesta va fi retransmis. Pentru a se evita și problema mesajelor duplicat, se folosește un mecanism cu confirmări pozitive cumulative. Formatul unui pachet de date la nivelul protocolului TCP este prezentat în figura 2.7.:

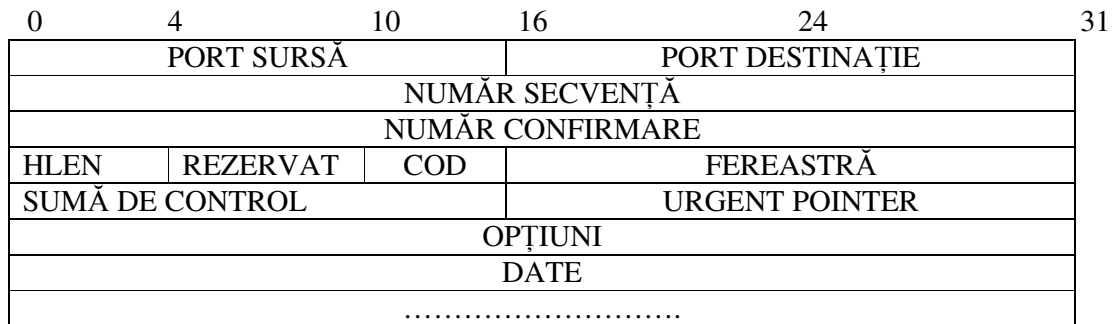


Fig. 2.7. Pachet de date TCP

Semnificația câmpurilor este următoarea :

- Portul sursă și portul destinație reprezintă, ca și la UDP, numerele de port asociate aplicațiilor de la sursă și de la destinație.
- Numărul de secvență este folosit pentru identificarea fiecărui pachet pentru o recepționare corectă, în ordine, a pachetelor dintr-un flux de biți.
- Numărul de confirmare identifică numărul octetului următor așteptat la recepție.
- Câmpul HLEN conține un întreg care specifică lungimea antetului specificată în multiplii de 32 de biți.
- Câmpul Rezervat nu este folosit și este setat pe 0.
- Biții de cod specifică niște funcții de control sau informații care determină scopul și conținutul respectivului pachet.
- Câmpul Fereastră (“WINDOW”) este câmpul prin care se furnizează mecanismul de control al fluxului. Prin acest câmp se specifică câte blocuri de date poate accepta mașina receptoare.
- Suma de control reprezintă mecanismul de detectare a datelor eronate.
- Câmpul Urgent Pointer indică, dacă bitul de cod URG a fost setat, porțiunea de date urgente.
- Un exemplu de opțiune TCP poate fi dimensiunea maximă a unui segment de date.

Circuitul S-7600 asigură o conectivitate superioară la Internet pentru orice gen de tehnică de calcul datorită faptului că realizează hardware stiva TCP-IP. Dacă un PC dispune de suficientă putere de calcul pentru a codifica / decodifica cadrul TCP-IP, pentru un microcontroller este vital ca legarea la Internet să fie realizată prin intermediul unei stive hardware, pentru că un cadru de date are structura complexă prezentată în paragraful anterior.

Schema bloc a circuitului este dată în figură 2.8.

Semnificația semnalelor la pini este:

- RS- selecție de registre (adrese sau date);
- C86- selecția modului de lucru al interfeței pentru procesoare 68K sau INTEL x80;
- READX- ciclu de citire;
- WRITEX- ciclu de scriere;
- PSX- selecție interfață paralelă sau serială;
- INT1, INT2- cereri de întrerupere;
- INTCTL- controlul cererii de întrerupere;
- SD0-7- 8 linii de date (în mod interfață paralelă);
- SD7- date de intrare seriale, SD6- ceas serial, SD5- date de ieșire seriale (în mod interfață de date serială);
- BUSYX- indică faptul că circuitul este ocupat.

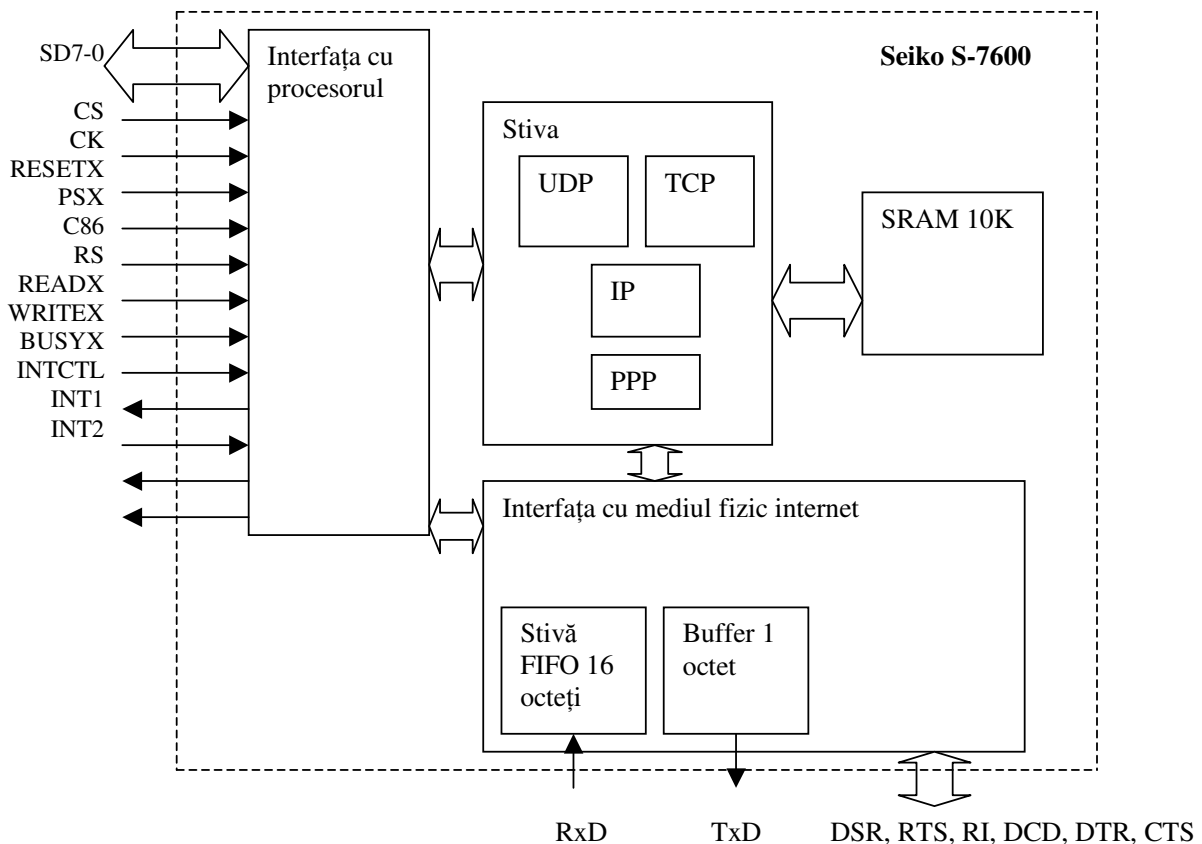


Fig. 2.8. Schema bloc a circuitului S-7600

În modul interfață paralelă pentru procesoare INTEL, pe liniile de date sunt prezente și adresele, validate de un RS activ, care pot fi încărcate în circuit la un impuls WRITEX. Datele de la locația adresată apar pe liniile SD0-SD7 în sensul dat de WRITEX și READX. Un ciclu nou poate începe după ce BUSYX revine la starea inactivă. Un ciclu este posibil când circuitul este selectat (CS activ).

În modul interfață serială pe cele 2 linii de date se transferă serial întâi adresa apoi datele, cu frecvența dată de linia de tact.

Circuitul dispune de o memorie SRAM internă de 10Kocteți în care este stocată stiva: bufferul cadrului recepționat, bufferul cadrului de transmis, numărul IP etc. Memoria poate stoca 2 cadre.

Circuitul poate lucra cu protocoalele TCP (Transmission Control Protocol), UDP (User Datagram Protocol), IP (Internet Protocol) și PPP (Point to Point Protocol, utilizat în legături punct la punct, așa cum este legătura între modemi pe linie închiriată).

Registreele circuitului sunt văzute în spațiul de memorie. O descriere sumară a registrelor și adresele la care se află:

- 00h- codul circuitului;
- 01h- registru de control general;
- 02h- localizarea cadrului 1 sau 2;
- 04h- registru de întreruperi;
- 08h- configurarea portului serial;
- 09h- întreruperi de la portul serial;
- 0Ah- măști pentru întreruperea de la portul serial;
- 0Bh- date pentru portul serial;
- 0Ch,0Dh- rata de divizare a tactului pentru viteza de transmisie serială (Baud Rate);
- 10h-13h- adresa IP proprie;
- 22h, 23h, 24h- Registru de configurare a cadrului;
- 26h- configurarea întreruperilor de la un cadru;
- 28h- date disponibile în cadru;
- 2Ah, 2Bh- Măști pentru întreruperi de la un cadru;
- 31h, 33h- lungimea bufferului de intrare / ieșire;
- 36h, 37h- adresa IP pentru destinatar;
- 60h, 64h- date pentru protocolul PPP.

O schemă bloc de cuplare a circuitului pe magistrala ISA este dată în figura 2.9.:

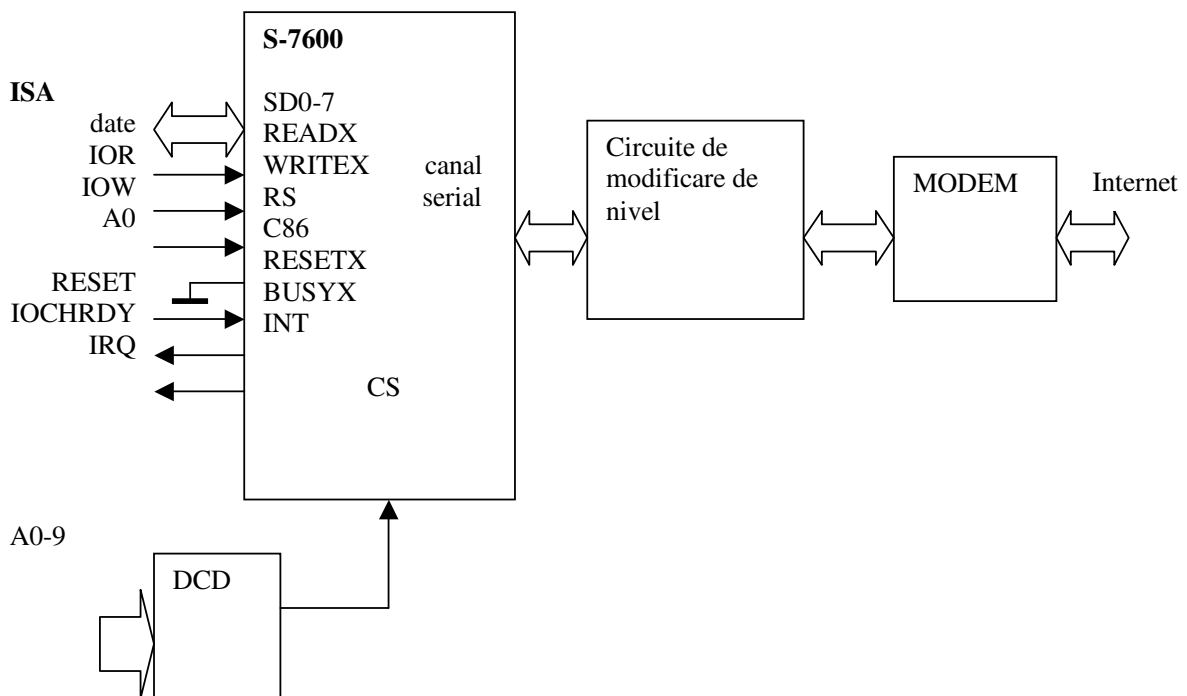


Fig. 2.9. Locul circuitului S-7600 între ISA și Internet

Circuitul este selectat prin selecție cu decodificator care activează semnalul CS. Transferul de date are loc pe magistrala de date de 8 biți SD0-7. Dacă circuitul este ocupat generează un BUSYX care prelungește ciclul de magistrală cu IOCHRDY. Cererea de întrerupere se leagă la o linie liberă a ISA. Linia C86 selectează modul de lucru pentru procesoare INTEL. Liniile din canalul serial se leagă la un modem după modificarea de nivel.

O schemă bloc de conectare a S-7600 la un microcontroller din familia 8051 este dată în figura 2.10:

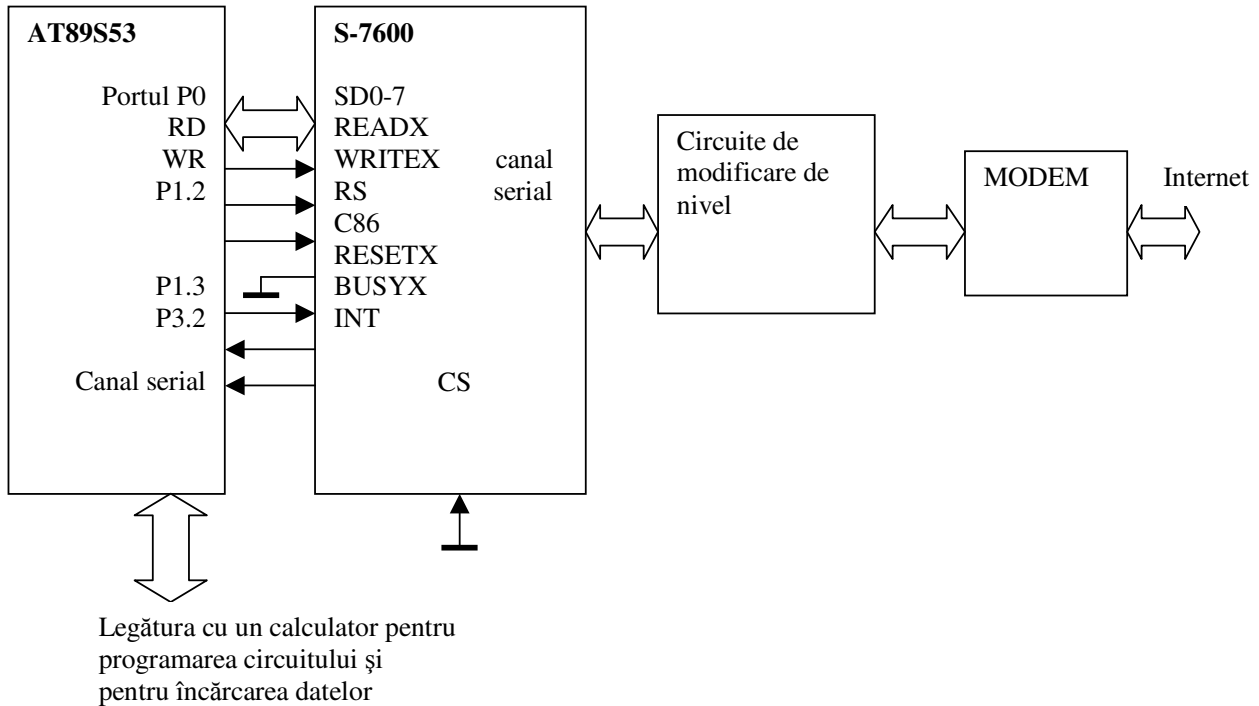


Fig.2.10. Schema bloc de conectare a S-7600 la un microcontroller din familia 8051

Semnalele de comandă pentru S-7600 sunt realizate de semnale din porturile microcontrollerului. Această schemă poate fi folosită pentru a stoca o pagină de WEB independent de un sistem de calcul. O dată încărcată pagina prin legătura serială cu sistemul gazdă, ea va fi accesibilă prin Internet.

2.5.Circuitul repeter W89C982 (producător WINBOND)

Circuitul repeter se numește IMPR II (Integrated Multiple Port Repeater) și este circuitul care realizează funcția de hub inteligent.

Echipamente de rețea Repetorul

Scopul unui repeter este de a regenera și retransmite semnalul din rețea pentru a permite acestuia să parcurgă segmente de cablu mai lungi. Unul din dezavantajele cablului UTP este lungimea maximă a unui segment de rețea care este de 100 metri. Prin introducerea unui repeter se extinde lungimea unui asemenea segment de rețea cu 100 de metri. Repetorul este considerat un dispozitiv de nivel 1 deoarece nu gestionează informația.

Hub –ul (concentratorul)

Scopul unui hub este acela de a regenera și retransmite semnalul din rețea. Acest lucru este realizat la nivel de bit pentru un număr mare de stații (4 , 8 sau chiar 24) folosind procesul numit concentrare (de aceea hub –ul se mai numește și concentrator). Se observă că această definiție este similară cu cea a repetorului și de aceea, un hub poate fi numit și un repetor multi-port. Diferența este că la hub se pot conecta mai multe stații iar la repetor doar două.

Hub-ul crează un punct central de conectare a cablului de rețea (deci o topologie fizică de tip stea) și crește siguranța comunicației în rețea. Ca și repetoarele, hub–urile sunt considerate dispozitive de nivel 1.

Există diverse clasificări ale hub –urilor. Prima clasificare împarte hub–urile în pasive și active. Majoritatea hub–urilor actuale sunt active. Ele regenerează semnalul având o sursă externă de tensiune. Cele pasive nu regenerează semnalul, ci doar îl distribuie către toate porturile. Aceste hub–uri pasive nu extind lungimea unui segment de cablu.

O altă clasificare a hub–urilor este: hub–uri inteligente și hub–uri neinteligente. Hub–urile inteligente au porturi de gestionare ceea ce înseamnă ca ele pot fi programate pentru a administra traficul din rețea. Cele neinteligente regenerează doar și retransmit semnalele la toate porturile fără a putea administra în vreun fel traficul.

Bridge –ul

Un bridge este un dispozitiv de nivel 2 proiectat pentru conectarea a două segmente de rețea LAN. Scopul unui bridge este de a filtra traficul din rețeaua LAN, pentru a menține local traficul local, dar permițând conectivitatea între segmente pentru traficul spre aceste segmente. Bridge–ul are memorată o listă de adrese MAC a tuturor calculatoarelor aflate pe cele 2 segmente de rețea și, la venirea unui pachet de date, pe baza adresei MAC, bridge–ul va ști exact de ce parte se află calculatorul de destinație.

Switch –ul

Switch–ul este, ca și bridge–ul , un dispozitiv de nivel 2. De fapt , switch–ul mai este denumit și bridge multi-port. Diferența dintre hub și switch este aceea că switch–ul ia decizii pe baza adresei MAC iar hub–ul nu ia nici un fel de decizii. Datorită acestor decizii pe care le iau, switch –urile aduc un plus de eficiență rețelelor LAN. Ele fac acest lucru comutând datele doar spre portul la care este conectat calculatorul destinație. Scopul switch–ului este concentrarea conectivității și eficientizarea transferului de date.

Router –ul

Routerul este dispozitivul de rețea care lucrează la nivelul 3, nivelul Rețea. Lucrând la acest nivel routerul ia decizii pe baza unor grupuri de adrese de rețea (clase de adrese) spre deosebire de adresele individuale de la nivelul 2, adresele MAC. Routerelor pot conecta tehnologii de nivel 2 diferite, cum ar fi Ethernet, Token Ring sau FDDI. Astfel, datorită capacității de a ruta pachetele pe baza informației de la nivelul 3, routerelor au devenit o componentă principală a Internetului.

Scopul unui router este de a examina pachetele care vin (date de nivel 3), de a alege cea mai bună cale spre destinație iar apoi de a trimite pachetul spre portul de ieșire corespunzător. Routerelor sunt cele mai importante dispozitive de regularizare a traficului din rețelele mari. Ele permit practic comunicația între orice calculator aflat oriunde în lume cu orice alt calculator aflat într-o cu totul altă locație.

Circuitul W89C982 realizează funcția de repetor conform standardului IEEE 802.3 și funcția de transceiver conform cu standardul 10BASE-T (pentru cablu UTP). Circuitul are 8 porturi UTP și un port AUI. La fiecare dintre porturile UTP poate fi conectat un tronson de cablu spre o stație de rețea. La portul AUI se poate conecta un cablu de maximum 50m. Circuitul are un port de stare care, folosind semnalele de stare ale fiecărui port indică coliziunea, conectarea, polaritatea și activitatea rețelei. Un port numit inter-IMPR conține semnale pentru a se putea conecta mai multe hub-uri. Printr-un port serial sincron circuitul poate fi gestionat și se poate citi starea portului de date. Blocurile de prelucrare internă a informației realizează:

- Gestionarea fragmentării;
- Gestionarea repetării pachetelor;
- Gestionarea coliziunii;
- Verificarea lungimii pachetelor pentru evitarea transmiterii pachetelor prea lungi;
- Fiecare port poate fi izolat când apare un număr prea mare de coliziuni și reconectat ulterior.

Schema bloc a circuitului este dată în figura 2.11.:

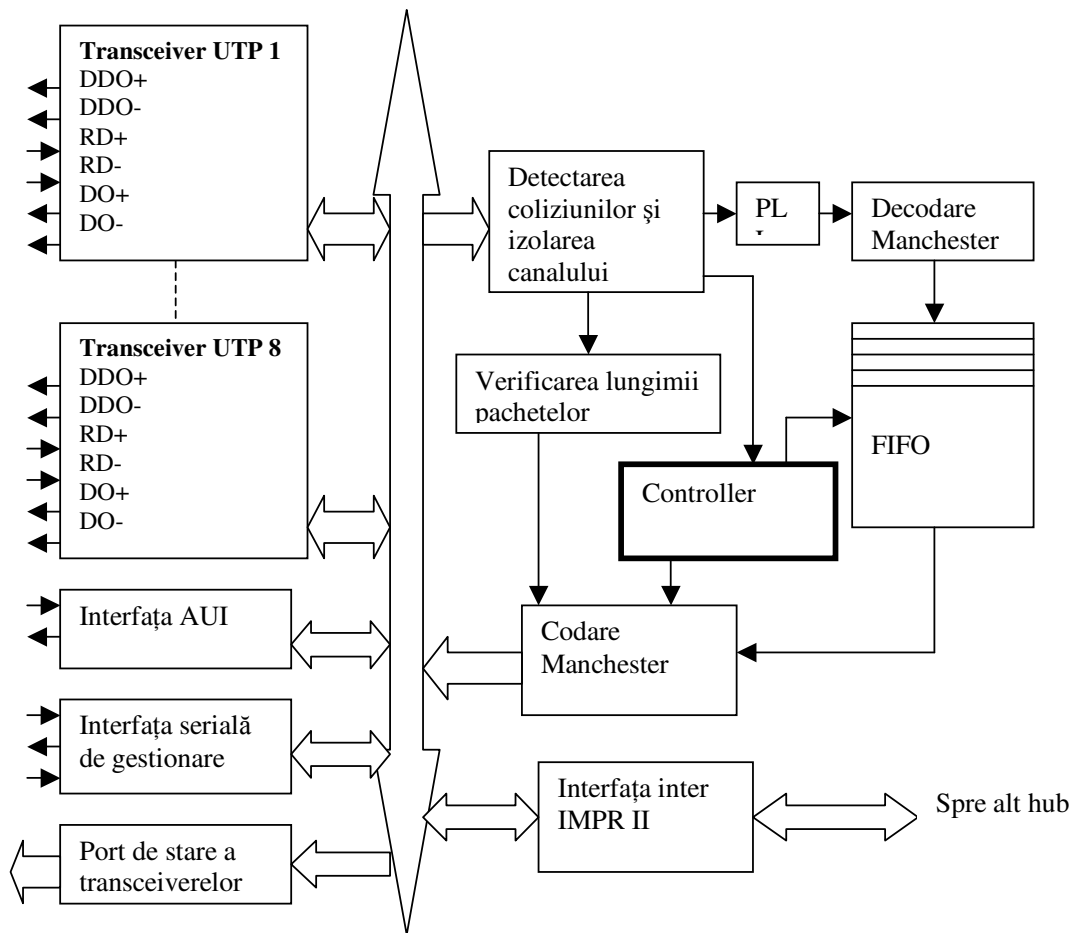
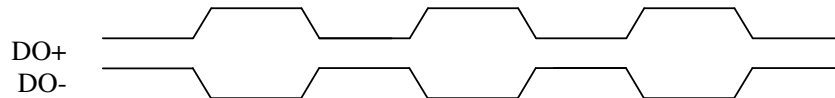


Fig.2.11. Schema bloc a circuitului repetor W89C982

La acest circuit se poate cupla o linie AUI, precum și o linie pe cablu coaxial 10BASE2 sau 10BASE5 printr-un transceiver de cablu coaxial. La cele 8 transceivere 10BASE-T se pot cupla 8 linii UTP. Semnalele de intrare în transceiver sunt cele de date recepționate, diferențiale (RD+ și RD-), iar cele de ieșire sunt datele emise (DO+ și DO-) precum și date emise întârziate (DDO+ și DDO-).

Transceiverele UTP au opțiunea de schimbare automată a polarității în funcție de polaritatea pachetelor de intrare. Blocul de verificare a polarității verifică polaritatea primului pachet de intrare după RESET sau polaritatea primele 3 impulsuri de conectare (link pulse) și dacă acestea sunt pozitive stabilește polaritatea portului ca pozitivă. Dacă polaritatea este negativă, datele vor fi retransmise după o inversare. Toate pachetele de date vor fi retransmise cu polaritatea stabilită în funcție de polaritatea pachetului de intrare. Transmisia fiind diferențială, datele arată ca în figura 2.12:

Date emise



Date inversate

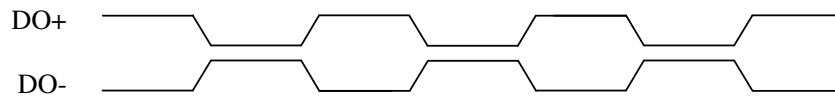


Fig.2.12. Diagrama de timp pentru date emise de ambele polarități

O dată polaritatea stabilită, se poate schimba doar după un RESET sau după intrarea în starea de așteptare a conectării. Verificarea polarității poate fi validată / invalidată prin interfața de gestionare a IMPR și automat este validată la pornirea circuitului.

Dacă un port UTP nu primește pachete sau impulsuri de conectare mai mult de 60ms intră în starea de așteptare. Din starea de așteptare portul nu poate ieși pentru primul pachet de retransmis, așa încât pachetul nu va fi retransmis. IMPR transmite impulsuri de conectare după 16 ms după transmiterea unui pachet, (dacă nu sosește un alt pachet) așa încât starea de așteptare se poate datora doar unei defecțiuni, aceste funcții fiind considerate funcții de testare a conexiunii.

O coliziune apare când 2 porturi UTP încearcă în același timp să transmită date sau se recepționează un semnal de coliziune de la AUI. Un port UTP este deconectat de IMPR dacă:

- Există o coliziune care durează 1024 perioade de bit;
- Apar 32 de coliziuni succesive.

Portul poate fi reconectat după 2 algoritmi:

- Algoritm standard, dacă un port deconectat poate transmite sau recepționa fără coliziuni un pachet de 512 perioade de bit;
- Algoritm alternativ, dacă un port deconectat poate transmite fără coliziuni un pachet de 512 perioade de bit.

Aceste funcții sunt necesare pentru a proteja rețeaua de segmente care funcționează prost.

Starea porturilor este raportată printr-un bit de coliziuni și un bit de stare pentru fiecare port. Cu 2 biți de comandă se selectează care aspect al stării se dorește a fi semnalizat. Se pot astfel semnaliza: activitate (stare de lucru sau de așteptare, iar impulsuri de 10Hz citite pe bitul de stare semnifică recepția unui pachet), polaritatea și procentul de utilizare al fiecărui port.

Interfața serială de gestiune constă într-o interfață serială sincronă care conține o intrare de date, o ieșire de date și un tact de comunicație. Comenzile către IMPR și răspunsul IMPR au aceeași structură ca la RS232: bit de start, 8 biți de date.

Interfața inter-IMPR permite cuplarea mai multor circuite IMPR pentru a asigura mai multe linii de rețea. 4 circuite IMPR asigură 32 de canale de rețea. Pentru a gestiona funcționarea circuitelor IMPR se folosește un circuit numit IMPR Integrator. Transmisia între circuite se face printr-o interfață serială.

Link-uri utile:

1. www.crystal.com CS8900A, foi de catalog, format pdf
2. www.crystal.com Ayres, J. Using the CRYSTAL CS8900A in 8 Bit Mode, AN181
3. www.winbond.com W89C982, Datasheet
4. cornelius@ethernet.isdn-development.de Werner C. Implementation notes for an Ethernet solution usable for embedded systems using the RTL8019, 2002
5. www.questlink.com Realtek Semi-Conductor CO. LTD, Taipei Taiwan, RTL8019, Advance Information