

CAPITOLUL 3

TRANSPORTUL DATELOR PE O LEGĂTURĂ DE DATE

3.1 Legătura de date. Definiție și funcțiuni

Legătura de date este un ansamblu compus din elementele a două echipamente terminale de date, care sunt controlate de un protocol și care, prin intermediul circuitului de date ce le interconectează, permit, împreună, transferul datelor (fig. 3.1). Nivelul legătură de date este realizat pe conexiunea fizică asigurată de un circuit (fie el și dintr-o rețea), pentru a furniza un serviciu de transfer de date fiabil nivelului rețea sau, direct, nivelului aplicație.

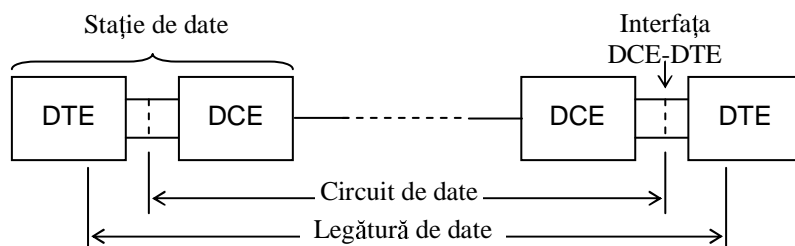


Fig. 3.1 Legătură de date

Circuitul de date este ansamblul format din două canale de transmisiune asociate pentru a asigura transmiterea datelor în ambele sensuri.

Stația de date este o unitate funcțională care furnizează date pentru transmisiune, primește datele transmise și realizează toate funcțiunile necesare pentru comunicația cu o altă unitate funcțională.

Protocolul legăturii de date este constituit dintr-un set de reguli care determină comportarea unităților funcționale în cursul comunicației, urmărind ca informația transferată să fie recepționată și interpretată corect.

Legătura de date poate fi considerată sub două aspecte: (1) *fizic*, cu referire la circuitul de date și transmisiunea datelor și (2) *logic*, legat de comanda legăturii și

coordonarea transferului de date pentru a-l face sigur și eficient. Din punct de vedere al configurației fizice legătura de date poate fi de mai multe tipuri: punct-la-punct, multipunct, în buclă (fig. 3.2).

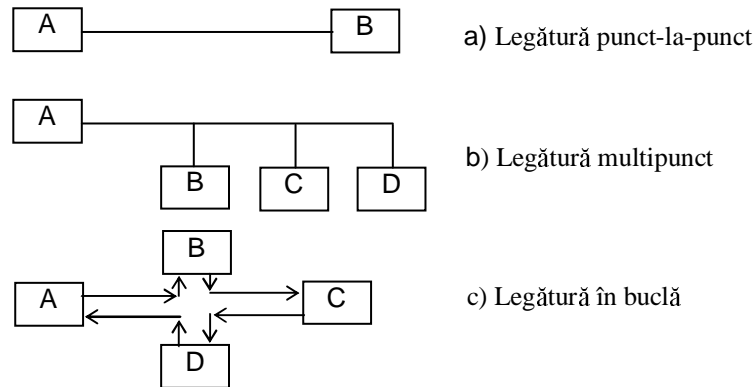


Fig. 3.2 Configurații ale legăturii de date

Pentru a asigura transferul sigur și eficient al datelor protocolul legăturii de date trebuie să realizeze următoarele funcțiuni principale: controlul erorii, controlul fluxului, formatarea datelor în cadre (blocuri), identificarea sursei și destinației datelor (în legăturile multipunct și în buclă). Desigur, protocolul legăturii de date trebuie să aibă în vedere și situațiile anormale de funcționare care pot surveni în cursul transferului datelor: întreruperi ale legăturii, saturarea unei stații, erori frecvente, etc.

De asemenea, protocolul trebuie să fie adecvat tipului aplicației. Aplicațiile pot fi de tipul conversațional, prelucrare pe loturi, prelucrare în timp real, transfer de fișiere, etc. Corespunzător tipului aplicației funcționarea legăturii de date poate diferi de la o aplicație la alta.

Astfel, în *aplicațiile conversaționale* un mesaj este emis într-un sens, în celălalt sens este transmis mesajul de răspuns, alternativ, ca într-o conversație și legătura de date trebuie să funcționeze în modul bidirecțional alternant.

În *prelucrările pe loturi* datele sunt transmise într-un volum mare într-un sens, către centrul de prelucrare. Rezultatele sunt transmise în celălalt sens, după terminarea prelucrării, care poate dura un timp îndelungat. Legătura funcționează în modul bidirecțional alternant, dar cu schimbări rare ale sensului de transmisiune.

Pentru *prelucrările în timp real* datele recepționate trebuie prelucrate rapid și rezultatele (sau deciziile) trebuie transmise imediat. Legătura trebuie să funcționeze în modul bidirecțional simultan. În aplicațiile spațiale, spre exemplu, timpul devine un factor primordial și, pentru controlul erorii, se pot folosi codurile corectoare de erori.

Protocolul legăturii de date trebuie să țină seama și de caracteristicile traficului de date în cele două sensuri de transmisiune. De lungimea cadrelor și volumul mesajelor ce trebuie transmise vor depinde dimensiunile memoriilor de emisie și de recepție, procedurile de reluare în cazurile de funcționare anormală, timpii de așteptare și de răspuns.

Protocolul legăturii de date trebuie să țină seama, de asemenea, și de caracteristicile circuitului de date: timpul de propagare, debitul datelor, procentul de erori.

3.2 Interfața modem-terminal

Subsistemul de comunicații al unui calculator este conectat la circuitul de date printr-o interfață compusă din mai multe circuite, nu numai cele utilizate pentru transmiterea și recepția datelor. Această interfață prezintă trei tipuri de caracteristici: mecanice, funcționale și electrice. În cele ce urmează vor fi prezentate câteva interfețe standardizate, utilizate atunci când este implicat și un modem în transmiterea datelor. Acesta este cazul legăturilor punct la punct stabilite pe rețeaua telefonică sau pe rețelele publice de date și al unor legături multipunct, dar nu și al rețelelor locale de calculatoare.

Din punct de vedere funcțional (al caracteristicilor funcționale) există două recomandări ITU-T privind interfața modem-terminal: V.24 și X.24. Ca o regulă generală, recomandările ITU-T privind transmisiunile de date pe rețeaua telefonică sunt grupate în seria de avize V, iar cele privind transmisiunile de date pe rețelele publice de date sunt grupate în seria de avize X. O interfață care corespunde unei anumite recomandări ITU-T este în mod obișnuit identificată prin numele recomandării respective.

Interfața V.24 (echivalentă cu RS-232C, un standard EIA – Electrical Industries Association) a fost definită inițial ca interfață standard între un echipament terminal de

date (DTE) și un modem (DCE) pentru transmisiuni de date pe rețeaua telefonică. Recomandarea V.24 specifică numele și funcțiunile pentru 39 circuite de interfață, grupate în seria 100 (după notația lor), și pentru alte 12 circuite, grupate în seria 200.

Circuitele din seria 200 au fost definite pentru apel și răspuns automat, dar ele nu sunt folosite în prezent, în locul lor fiind utilizate tot cele din seria 100. Din seria 100 sunt utilizate numai o parte, aproximativ 15 circuite, specificate pentru fiecare tip de modem în avizul corespunzător lui. În figura 3.3 sunt prezentate circuitele mai frecvent utilizate. Ele pot fi grupate astfel:

- circuite pentru comenzi, răspunsuri și semnalizări (105, 106, 107, 108, 109, 125);
- circuite pentru transmisie și recepție date (103,104);
- circuite pentru baza de timp (113,114,115).

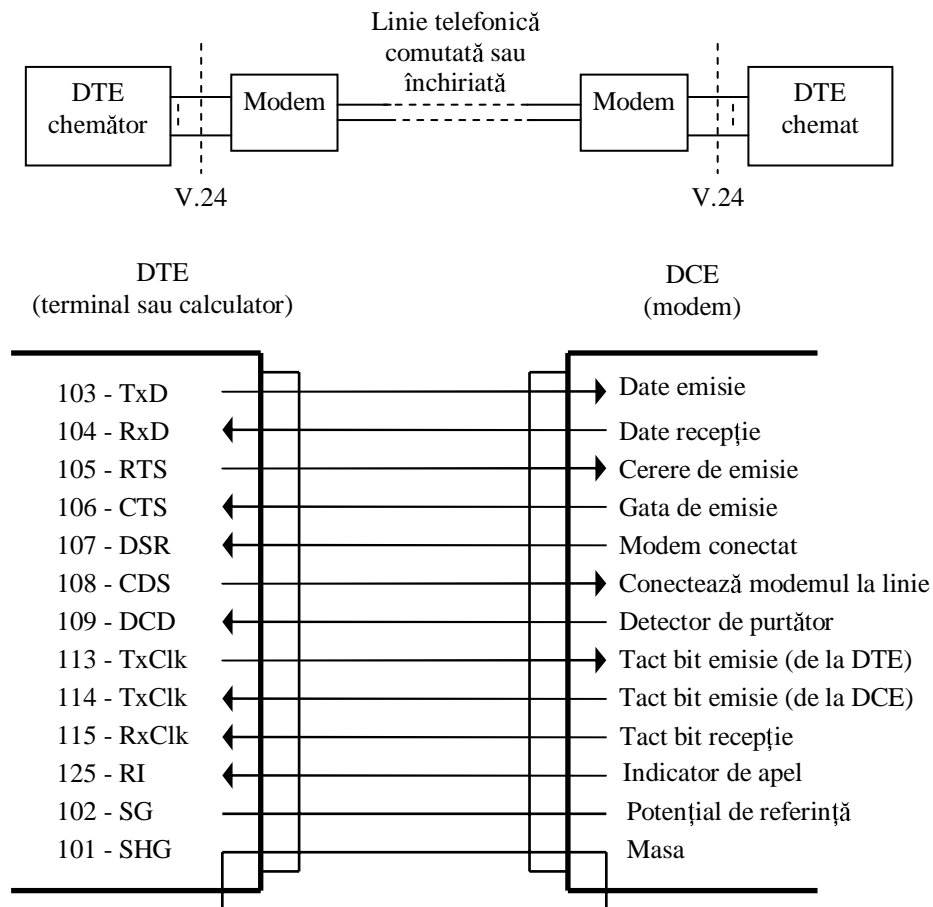


Fig. 3.3 Interfața V.24

Într-un apel manual pentru transmisiunea datelor circuitul de date se stabilește prin realizarea apelului telefonic, ca în modul normal pentru o convorbire telefonică și apoi urmează comutarea liniei telefonice la modemuri (printr-o comandă de la terminal pe circuitul 108 și un răspuns al modemului pe 107). Terminalul face apoi cererea de emisie (105) și, după ce primește răspunsul gata de emisie (106), poate începe transmisia datelor. Comenzile și răspunsurile sunt reprezentate electric prin starea ON (fig. 3.4).

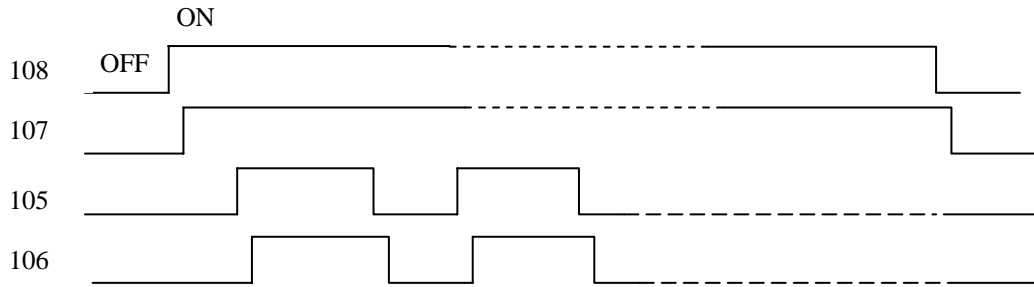


Fig. 3.4 Semnale pe circuite de interfață

Starea ON pe circuitul 109 arată că nivelul semnalului recepționat de modem din linie este peste nivelul minim admisibil, la care modemul mai poate reconstitui datele. Întârzierea cu care trece circuitul 106 în starea ON după ce 105 a trecut în ON este necesară pentru a permite celor două modemuri în corespondență să facă inițializarea (reglajele necesare pentru sincronizări, egalizare, compensarea ecoului, etc.).

X.24 este interfața definită pentru utilizarea în rețelele publice de date. Ea se compune din numai 11 circuite. Modul în care sunt folosite aceste circuite este specificat, pentru terminalele sincrone, în recomandarea X.21, iar pentru terminalele asincrone, în recomandarea X.20.

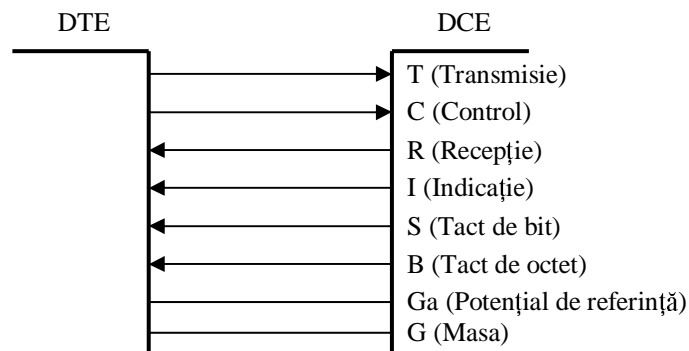


Fig. 3.5 Interfața X.21

Deoarece multe terminale sunt echipate cu interfață de tip V.24, recomandările X.21 bis (pentru terminale sincrone) și X.20 bis (pentru terminale asincrone) specifică modul în care pot fi utilizate circuitele acestei interfețe (V.24) la conectarea unor astfel de terminale în rețelele publice de date. Figura 3.5 prezintă circuitele utilizate de interfața X.21. Circuitele T și C sunt utilizate pentru a transmite date și informație de control de la DTE către DCE, iar circuitele R și I sunt utilizate pentru date și semnalizări de la DCE spre DTE.

După caracteristicile electrice ale circuitelor din care sunt alcătuite sunt mai multe tipuri de interfețe. Acestea sunt specificate de recomandările ITU-T V.28, V.35, V.10 (X.26), V.11 (X.27). În tabelul următor sunt specificate tipul circuitelor (neechilibrate - cu un singur fir și întoarcerea prin firul pentru potențialul de referință - sau echilibrate, cu două fire), tensiunile utilizate pentru reprezentarea celor două stări (ON="0", OFF="1") și viteza de modulație permisă.

Tipul interfeței	V.28	V.35(pentru 103, 104, 113, 114, 115)	V.10 (X.26)	V.11 (X.27)
Tipul circuitelor	neechilibrate	echilibrate	neechilibrate	echilibrate
Tensiuni: ON	3 ÷ 25 v	+0.55 v	> 0.3 v	> 0.3 v
OFF	-3 ÷ -25 v	-0.55 v	< -0.3 v	< -0.3 v
Viteza	≤ 20 kBd	> 20 kBd	1 MBd	100 MBd

3.3 Sincronizarea de cadru și de caracter

Pentru a decoda și pentru a interpreta corect șirul biților recepționați, receptorul trebuie să realizeze sincronizarea de bit, sincronizarea de caracter sau de octet și sincronizarea de cadru. Sincronizarea de bit, în cazul transmisiunii sincrone, este o funcție a circuitului de date. Celelalte două funcțiuni de sincronizare revin legăturii de

date și modul în care sunt realizate depinde de tipul tranmsiunii (sincronă sau asincronă), tipul datelor (text sau date pur binare) și tipul protocolului utilizat.

Protocoalele de comunicație utilizate de legătura de date se pot clasifica în două categorii: *protocoale orientate pe caracter* și *protocoale orientate pe bit*. La protocoalele orientate pe caracter diferitele funcțiuni ale legăturii de date sunt realizate prin intermediul unor caractere de control, în timp ce la protocoalele orientate pe bit acestea sunt realizate prin intermediul unor secvențe specifice de biți.

Protocoale orientate pe caracter

În transmisiunile asincrone se utilizează numai protocoale orientate pe caracter. Sincronizarea de caracter (sau de octet) este asigurată implicit prin intermediul elementelor de start și stop care încadrează fiecare caracter (octet). Pentru sincronizarea de cadru, adică delimitarea fiecărui cadru (determinarea începutului și sfârșitului cadrului), se folosesc (fig. 3.6) caracterele de control STX (start-of-text, începutul textului) și ETX (end-of-text, sfârșitul textului). În cazul transmiterii unor texte (șir de caractere grafice) nu există riscul ca în cadrul textului, deci în conținutul cadrului, să apară caracterele STX și ETX. Dacă însă se transmit date pur binare, cum este cazul fișierelor ce conțin programe compilate, este posibil ca un octet din conținutul cadrului să fie asemănător caracterului ETX, ceea ce, în mod eronat, ar fi interpretat de receptor ca sfârșit de cadru.

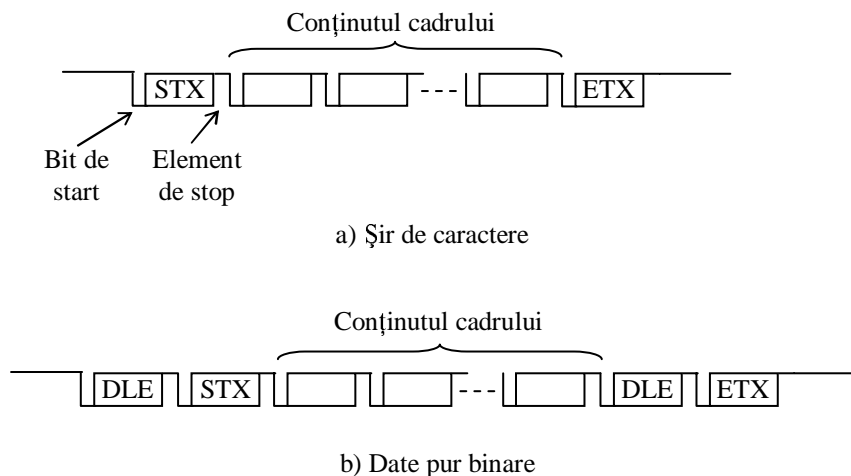


Fig. 3.6 Sincronizarea de cadru

Pentru a preîntâmpina astfel de situații, în cazul transmiterii datelor pur binare, cele două caractere de control folosite pentru delimitarea cadrelor sunt precedate, fiecare, de un alt caracter de control, numit *data link escape* (DLE). În plus, transmițătorul introduce suplimentar, în conținutul cadrului, un caracter DLE după fiecare octet ce are aceeași configurație ca și DLE, indiferent de ce urmează după acest octet. Receptorul va examina, după ce recepționează combinația DLE-STX, semnificând începutul unui cadru, fiecare octet pentru a constata dacă acesta este un caracter DLE. Dacă întâlnește un astfel de octet va determina dacă următorul este tot un DLE sau este ETX. Dacă este tot un DLE îl va elimina, iar dacă este un ETX va considera cadrul terminat.

În transmisiunea sincronă se utilizează ambele tipuri de protocoale, orientate pe caracter sau orientate pe bit. La protocoalele orientate pe caracter sincronizarea de cadru se realizează, ca și în transmisiunea asincronă, prin caracterele de control STX și ETX. Deoarece, în transmisiunea sincronă, nu se folosesc elementele de start și de stop pentru delimitarea fiecărui caracter, pentru sincronizarea de caracter fiecare cadru este precedat de cel puțin două caractere de control SYN (Fig. 3.7), prin intermediul cărora receptorul va putea să delimiteze corect caracterele în fluxul serial al biților recepționați.

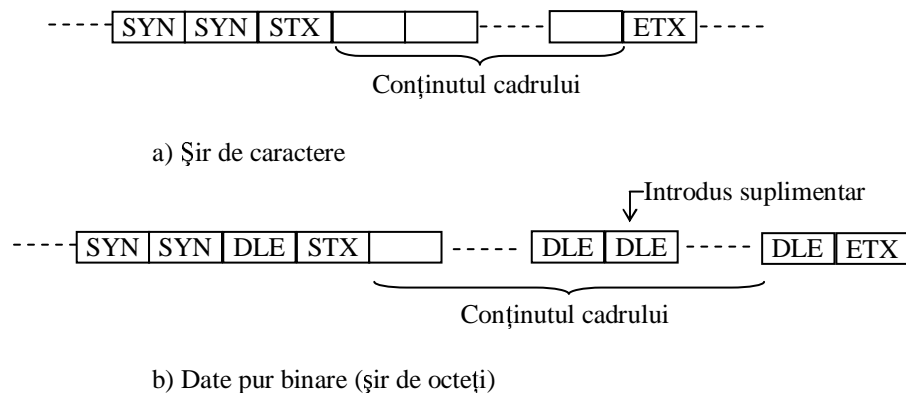


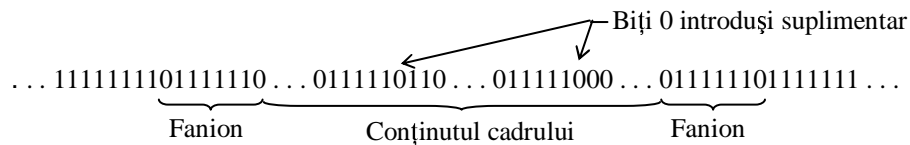
Fig. 3.7 Transmisiune sincronă orientată pe caracter

De asemenea, ca și în transmisiunea asincronă, în cazul datelor pur binare cadrele sunt delimitate de grupurile de caractere de control DLE STX și DLE ETX, iar în conținutul cadrului se introduce un caracter suplimentar DLE după fiecare octet care are structura DLE.

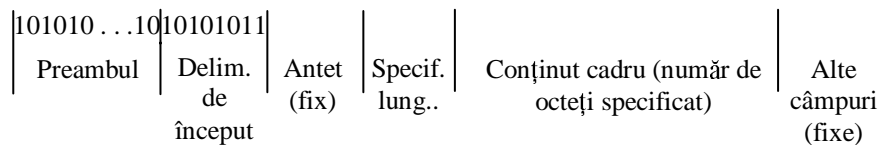
Protocoloale orientate pe bit

Din cauza grupurilor de caractere utilizate pentru delimitarea cadrelor și a caracterelor suplimentare DLE, introduse în cadru pentru a asigura transparența datelor, protocoalele orientate pe caracter sunt relativ neeficiente atunci când se transmit date pur binare. În plus, formatul caracterelor de control depinde de codul din care fac parte, chiar dacă în cadru sunt date pur binare. Astfel de probleme sunt evitate prin folosirea protocoalelor orientate pe bit.

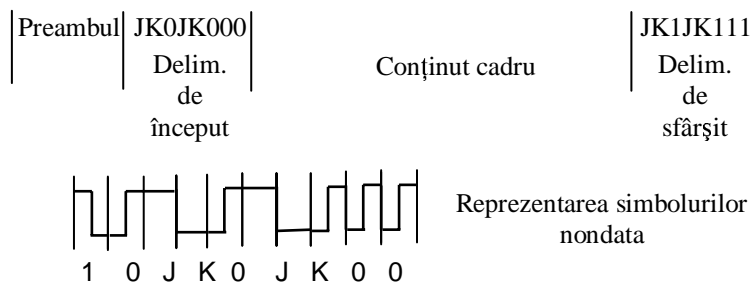
Delimitarea cadrelor la protocoalele orientate pe bit se face, de obicei, într-unul din următoarele moduri (Fig. 3.8):



a) Prin fanioane



b) Prin delimitator de început și specificarea lungimii



c) Prin delimitatoare conținând simboluri nondata

Fig. 3.8 Delimitarea cadrelor în protocoalele orientate pe bit

- cu fanioane (secvență particulară de biți, numită "flag" în limba engleză) la începutul și la sfârșitul cadrului;

- cu fanion la începutul cadrului și specificarea lungimii cadrului;
- cu delimitatoare de început și de sfârșit, conținând simboluri (nondata) care sunt reprezentate altfel decât biții de date.

În prima variantă, utilizată în special pe legăturile punct-la-punct, începutul și sfârșitul cadrului sunt semnalate printr-o aceeași secvență de opt biți 01111110, numită fanion. Pentru a asigura transparența datelor, adică să nu se introducă restricții privind secvența datelor ce urmează a fi incluse în cadru, transmițătorul va introduce automat un 0 după fiecare grup de cinci simboluri de date 1 consecutive. În felul acesta se evită prezența în conținutul cadrului a secvenței fanion, secvență ce ar semnala receptorului, în mod eronat, sfârșitul cadrului. La recepție, evident, simbolul 0 care apare după cinci simboluri 1 consecutive va fi eliminat. Între cadre, pentru a permite sincronizarea de bit, se pot transmite simboluri 1.

În cea de a doua metodă, utilizată în unele rețele locale, cadrul este precedat de un preambul constituit dintr-o secvență de biți 1 și 0, adecvată sincronizării de bit a stațiilor din rețea. Începutul cadrului este marcat prin octetul 10101011, după care urmează un antet de lungime fixă, conținând adresele stației destinație și stației sursă. Un alt câmp, tot de lungime fixă, specifică lungimea câmpului de date (conținutul cadrului) și în felul acesta nu mai este nevoie de un delimitator care să marcheze sfârșitul cadrului.

Cea de a treia variantă, utilizată de asemenea în unele rețele locale, folosește delimitatori de început și de sfârșit constituiți din biți de date și simboluri reprezentate altfel decât cele de date. În fig. 3.5 simbolurile de date sunt reprezentate în cod bifazic (Manchester), cu tranziție la mijlocul intervalului de bit, iar simbolurile nondata, notate J și K, sunt reprezentate prin nivel constant pe tot intervalul de simbol.

3.4 Circuite și echipamente pentru controlul comunicației

Multe din funcțiunile menționate în paragrafele precedente sunt realizate prin intermediul unor circuite integrate specializate sau programabile. Astfel, există circuite integrate specializate care asigură tactul de bit, sincronizarea de caracter, generarea și verificarea bitului de paritate pentru transmisiunile asincrone sau sincrone, orientate pe caracter. De asemenea, există circuite integrate care realizează o parte din funcțiunile

protocoalelor orientate pe bit: introducerea fanioanelor ce delimitează cadrele, introducerea și eliminarea bitului 0 pentru a asigura transparența datelor, generarea și verificarea CRC (codarea în cod ciclic). Aceste circuite poartă numele de: UART (Universal asynchronous receiver transmitter - Transmițător receptor asincron universal), USRT (Universal synchronous receiver transmitter), USART (Universal synchronous/asynchronous receiver transmitter - programabil pentru a realiza funcțiunile UART sau USRT), BOP (Bit-oriented protocol circuits). Există circuite pentru controlul comunicației universale, programabile să funcționeze ca UART, USRT sau BOP.

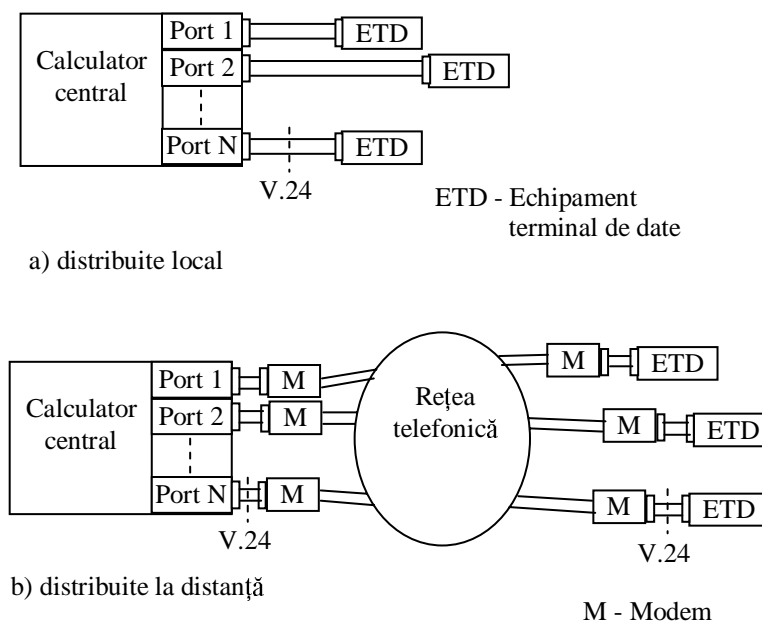


Fig. 3.9 Terminale izolate conectate la același calculator

În multe aplicații este necesar să se asigure accesul la un calculator central pentru un grup de terminale, distribuite într-o aceeași zonă sau în mai multe zone, aflate în apropierea calculatorului sau la distanță. Soluțiile adoptate sunt în funcție de dispunerea terminalelor unele față de altele și în raport cu calculatorul central. Spre exemplu, în cazul în care terminalele sunt distribuite în locuri diferite, se va folosi câte o legătură pentru a conecta fiecare terminal la un port al calculatorului, direct, printr-o interfață V.24, dacă ele sunt în apropierea calculatorului, sau prin intermediul rețelei telefonice, cu modemuri, dacă ele se află la distanță (fig. 3.9 a și 3.9 b).

În aplicațiile în care terminalele sunt grupate se pot folosi multiplexoare pentru a utiliza eficient liniile de transmisiune (fig. 3.10). Cu un astfel de echipament pe o linie de transmisiune se transmit, multiplexate, fluxurile de date de la mai multe terminale. Sunt utilizate multiplexoarele cu diviziune în timp, sincronă sau asincronă.

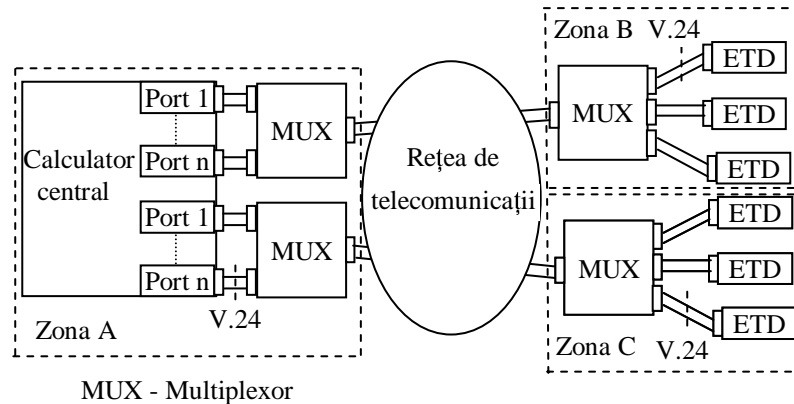


Fig. 3.10 Terminale conectate la calculator prin multiplexor

În cazul multiplexării cu diviziune în timp sincronă fiecărui terminal i se alocă o parte din capacitatea de transmisiune a multiplexorului, indiferent de starea sa, activă sau pasivă. Multiplexoarele cu diviziune în timp asincronă, numite și multiplexoare statistice, alocă capacitatea lor de transmisiune terminalelor la cerere sau pe baze statistice, prezentând astfel o eficiență mai bună în utilizarea suportului de transmisiune decât cele cu diviziune sincronă.

O altă variantă de utilizare eficientă a liniilor de transmisiune constă în a folosi configurația multipunct, cu mai multe terminale conectate la calculator, prin modemuri, pe o aceeași linie (fig. 3.11). Comunicația cu fiecare terminal este controlată de calculator prin procedura "poll-select" (interogare-selectare). Calculatorul central transmite periodic către fiecare terminal, identificat printr-o adresă, un mesaj de interogare (poll) prin care-l invită să transmită dacă are un mesaj pregătit pentru a fi transmis. Dacă calculatorul are de transmis un mesaj către un anumit terminal îi va transmite un mesaj de selecție, iar dacă terminalul selectat poate recepționa mesaje de date va răspunde printr-un mesaj adecvat (numit gata de recepție - "receive ready").

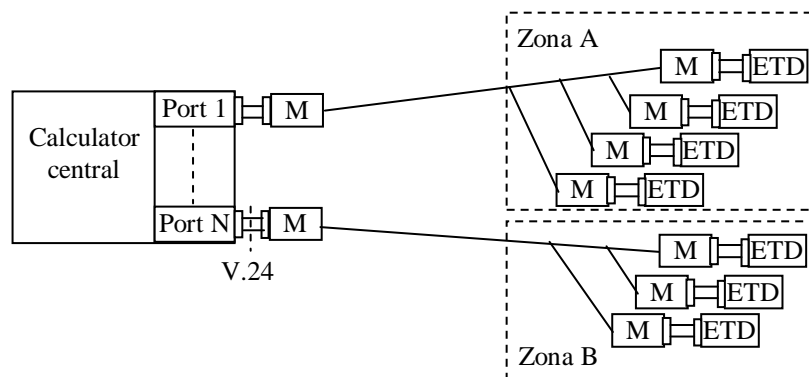


Fig. 3.11 Conectarea terminalelor prin legături multipunct

Pentru a scuti calculatorul central de problemele de comunicații, lăsându-i timp numai pentru prelucrarea datelor, se utilizează un echipament special, numit procesor frontal ("front-end processor"). De asemenea, pentru a reduce timpul de interogare-răspuns pentru grupul de terminale conectate la linia multipunct, se utilizează un alt echipament, numit controler de grup ("cluster controller"), care acționează ca intermediar între calculator și terminale, calculatorul sau procesorul frontal având de interogat sau de selectat numai controlerele de grup (fig. 3.12).

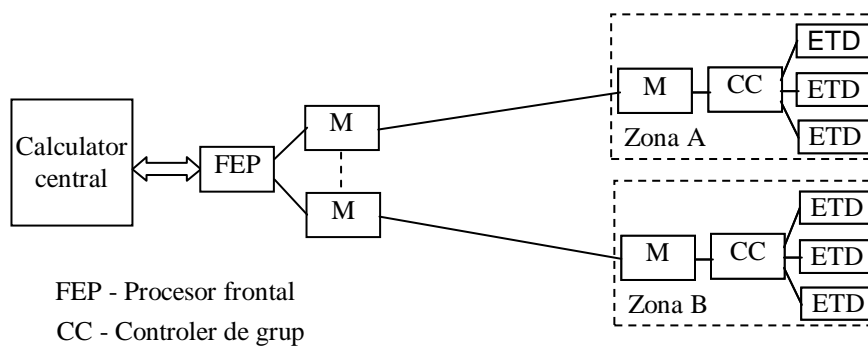


Fig. 3.12 Legături multipunct cu controlere de grup și procesor frontal

3.5 Controlul erorii și controlul fluxului - componente de bază ale protocoalelor

3.5.1 Controlul erorii

Funcția de control al erorii presupune detectarea la recepție a blocurilor de date recepționate cu erori și corectarea erorilor. Există un mod de operare al legăturii de date și, în general, și al altor nivele ale unei arhitecturi stratificate a interconectării sistemelor, numit mod de transmisiune *fără conexiune* (connectionless, best-try transmission), în care nu se realizează un control al erorii. Cel mult, dacă se utilizează procedee de codare și detecție a erorilor, blocurile detectate cu erori sunt eliminate la recepție. Alternativa la acest mod de funcționare o constituie modul de transmisiune *cu conexiune* (connection-oriented, reliable transmission), în care, prin regulile ce fac parte din protocolul de comunicație, se asigură, cu o anumită probabilitate, un transfer fiabil al mesajelor, fără erori și fără duplicate, în succesiune corectă.

Un mod simplu de control al erorii, manual, folosit pentru terminalele ce funcționează în modul caracter, mod în care datele sunt introduse în calculator direct de la tastatură (după fiecare tastă apăsată se transmite către calculator, serial, combinația de cod corespunzătoare caracterului asociat tastei) este cel al *verificării prin ecou* (echo checking). Dacă terminalul, aflat în apropierea calculatorului, este conectat direct la calculator, calculatorul va controla procesul de introducere a fiecărui caracter, va citi și va memora caracterul recepționat și-l va afișa pe ecranul monitorului. Dacă caracterul afișat diferă de cel ce s-a dorit a fi introdus, utilizatorul va introduce un caracter de control adecvat (*delete* sau *back space*), la recepția căruia calculatorul va elimina caracterul introdus anterior și-l va șterge de pe ecran. Dacă terminalul este conectat la calculator de la distanță, spre exemplu cu modemuri prin intermediul rețelei telefonice, pe ecranul terminalului nu se va afișa direct caracterul tastat. Acesta este transmis către calculator, calculatorul îl citește, îl memorează și-l va retransmite către terminal, pe ecranul terminalului afișându-se ceea ce acesta a recepționat. Desigur, dacă utilizatorul vrea să modifice acest caracter, poate iniția aceeași procedură de ștergere.

În cazul în care se transferă blocuri de caractere (cadre) pe o legătură de date serială, programul care controlează procesul de recepție trebuie să realizeze automat, fără

intervenția utilizatorului, procedura de control al erorii. În mod uzual, așa cum s-a arătat în capitolul 3, la recepția unui cadru se verifică dacă acesta este eronat și se transmite un mesaj de control pentru a confirma recepția corectă sau pentru a cere retransmiterea lui. Această procedură de control al erorii, numită ARQ (Automatic repeat request), se realizează în modurile *cu oprire și așteptare* (stop and wait) sau *retransmitere continuă*, cel de al doilea mod având variantele *retransmiterii cu întoarcere la N* (go-back-N) și a *repetării selective* (selective repeat).

- *Procedura retransmiterii cu oprire și așteptare* -

Această procedură este utilizată numai în protocoalele orientate pe caracter și funcționează în modul semiduplex deoarece stația care emite un cadru de informație, numită stație primară, trebuie să aștepte, după ce a emis cadrul, până când primește o confirmare de la cealaltă stație, stația secundară, privind modul în care a fost recepționat cadrul, corect sau nu. Sunt două variante de realizare a acestei proceduri: *retransmiterea implicită*, în care se confirmă numai cadrele recepționate corect (se transmit numai confirmări pozitive) și cu *cerere explicită* (se transmit confirmări pozitive și negative). Timpul de așteptare a confirmării de recepție este limitat și este marcat de un contor de timp declanșat la transmiterea fiecărui cadru de informație. La expirarea timpului de așteptare a confirmării de recepție stația primară retransmite cadrul de informație. Cadrele recepționate eronat, de informație sau de confirmare, sunt eliminate. Sunt posibile duplicate, adică recepția corectă, repetată, a aceluiași cadru de informație, atunci când un cadru de control de confirmare pozitivă (ACK) este recepționat eronat și, în consecință, este eliminat de către stația primară. Pentru a permite stației secundare să detecteze duplicatele, fiecare cadru de informație trebuie să aibă un identificator, numit număr de secvență. De asemenea, stația secundară trebuie să memoreze identificatorul ultimului cadru de informație recepționat corect. Dacă stația secundară recepționează o copie a acestui cadru, aceasta va fi eliminată. În același timp, pentru a permite stației primare resincronizarea (retransmiterea cadrului de informație precedent sau transmiterea următorului cadru de informație), în cadrele de confirmare, pozitivă sau negativă, stația secundară menționează identificatorul cadrului de informație la care se referă.

Numărul de secvență dintr-un cadru de informație se numește *număr de secvență la emisie*, și se notează cu $N(S)$, iar numărul de secvență dintr-un cadru de confirmare se

numește *număr de secvență la recepție* și se notează cu N(R). Figura 3.13 prezintă o structură simplificată a cadrelor de informație și de control, care pune în evidență modul de realizare a procedurii de control cu oprire și așteptare.

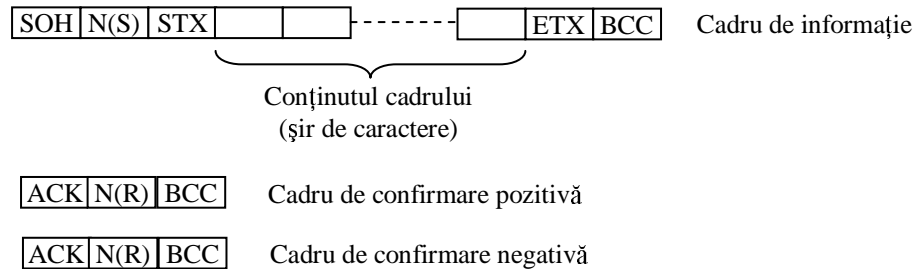


Fig. 3.13 Structura cadrelor de informație și de control

În afară de caracterele de control STX și ETX, prezentate anterior (par. 3.3), sunt utilizate și caracterele de control SOH (start-of-header, începutul antetului), ACK (confirmare pozitivă) și NAK (confirmare negativă). BCC reprezintă caracterul de verificare a blocului (block check character, par. 2.2).

Avantajul major al acestei proceduri de control constă în faptul că nu necesită o capacitate mare a memoriei. Stația primară trebuie să memoreze un singur cadru de informație, iar stația secundară memorează doar identificatorul ultimului cadru de informație recepționat corect.

- *Retransmiterea continuă* -

Așa cum s-a arătat în par. 2.4 strategia de control al erorii prin retransmiterea continuă asigură un randament mai bun al legăturii, dar necesită o capacitate mai mare a memoriei, atât în stația primară cât și în cea secundară.

Stația primară emite continuu cadre de informație, fără să aștepte confirmarea de recepție (totuși, așa cum s-a arătat, numărul cadrelor pe care le poate emite, fără a avea confirmarea de recepție pentru vreunul dintre ele, este limitat), dar reține o copie a fiecărui cadru transmis într-o *listă de retransmitere* care funcționează după principiul FIFO (primul intrat, primul ieșit - first-in, first-out).

Stația secundară transmite confirmări pentru fiecare cadru de informație recepționat corect și memorează, în ordine, într-o *listă de recepție*, identificatorii ultimelor n (se va arăta ulterior cum se stabilește valoarea lui n) cadre de informație

recepționate corect pentru a evita duplicatele. Cadrele de informație recepționate eronat sunt eliminate.

Pentru implementarea acestei tehnici stația primară utilizează o *variabilă de secvență la emisie* $V(S)$, care indică numărul de secvență $N(S)$ al următorului cadru de informație ce va fi transmis, iar stația secundară utilizează o *variabilă de secvență la recepție* $V(R)$, care indică numărul de secvență al următorului cadru de informație așteptat. Cum, de regulă, variabila $V(R)$ este incrementată imediat ce s-a recepționat cadrul de informație așteptat, înainte deci de a se genera cadrul de confirmare ACK, și valoarea ei este trecută în cadrul de confirmare ca $N(R)$, în cele mai multe protocoale care folosesc această procedură de control al erorii cadrul ACK(N) confirmă recepția corectă a cadrului $N-1$. La recepția unui cadru ACK stația primară va elimina din lista de retransmitere cadrul de informație pentru care s-a primit confirmarea de recepție corectă.

În varianta retransmiterii continue cu *repetare selectivă*, dacă stația primară primește un cadru NAK(N) ea va retransmite numai cadrul de informație N și va relua transmiterea cadrelor de informație de unde întrerupsese pentru retransmitere. De notat însă că la transmiterea unui cadru NAK stația secundară intră în *starea de retransmitere* în care nu mai transmite nici un cadru ACK până nu primește cadrul de informație semnalat. Altfel, dacă transmite o confirmare pozitivă pentru un cadru care-l succede pe cel pentru care a semnalat recepția eronată, iar cadrul NAK n-a fost recepționat de stația primară, aceasta va interpreta că toate cadrele de informație, până la cel menționat în cadrul ACK, au fost recepționate corect. Din starea de retransmitere iese numai după ce a recepționat corect cadrul semnalat. Repetarea selectivă nu este recomandabilă pentru aplicațiile în care se folosesc cadre de dimensiune mare și care trebuie livrate la recepție în ordine, deoarece necesită memorie de capacitate mare. Ea este recomandabilă pentru aplicații cu cadre independente sau cadre mici din care se recompune un mesaj.

Retransmiterea continuă cu *întoarcere la N* este mai puțin eficientă decât repetarea selectivă, stația primară retransmițând toate cadrele de informație, începând cu cel pentru care s-a primit confirmarea negativă.

În cele prezentate mai sus s-a presupus că informația se transmite într-un singur sens, iar în celălalt sens se transmit doar confirmări. Pentru a îmbunătăți eficiența utilizării legăturii confirmările se pot transmite în cadre de informație, dacă există astfel

de cadre de transmis în ambele sensuri. În acest caz fiecare cadru de informație conține un N(S), indicând numărul de secvență al respectivului cadru, și un N(R), reprezentând confirmarea pentru sensul invers de transmisiune.

3.5.2 Controlul fluxului

Controlul fluxului este o componentă importantă a protocolului de comunicație, prin care se controlează ritmul în care se transmit caracterele sau cadrele pe o legătură de date, astfel încât receptorul să aibă resurse de memorie suficiente pentru a le accepta înainte de a le prelucra. Două tehnici sunt frecvent utilizate în acest scop: X-ON/X-OFF și fereastra glisantă.

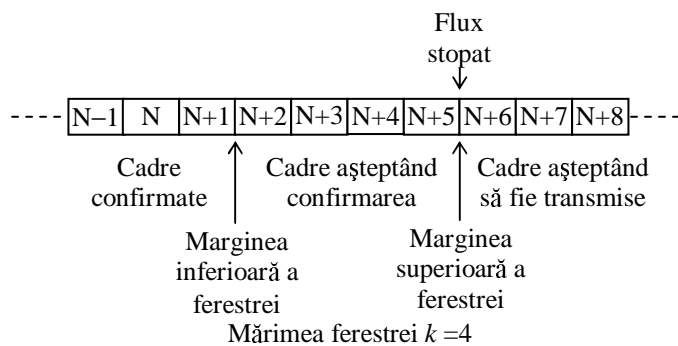
Tehnica X-ON/X-OFF

Este evident că în controlul manual al erorii prin ecou se realizează, implicit, și un control al fluxului. Dacă memoria tampon a calculatorului distant este plină, calculatorul nu va mai întoarce caracterul primit și utilizatorul, în mod normal, va înceta temporar să mai transmită. Totuși, frecvent se utilizează o tehnică automată de control al fluxului în astfel de aplicații. Aceasta constă în transmiterea de către calculator, în caz de suprasaturare, a unui caracter de control special, numit X-OFF, care va semnaliza dispozitivului de control din terminal să înceteze transmisia. La recepția acestui caracter terminalul fie ignoră caracterele tastate, fie le memorează, dacă dispune de memoria necesară, pentru a le transmite ulterior. Când situația de suprasaturare la calculatorul distant dispare, acesta va transmite un alt caracter de control, numit X-ON, prin care se semnalează terminalului că poate relua transmisia caracterelor. Această tehnică poate fi folosită și pe legătura dintre un calculator și o imprimantă sau un alt terminal, dacă acestea nu pot funcționa în același ritm cu viteza de ieșire a calculatorului, fluxul fiind controlat de terminal în aceste cazuri.

Fereastra glisantă

În strategia de control al erorii cu retransmitere continuă stația primară poate emite continuu cadre de informație, fără să aștepte cadrele de confirmare. Totuși, numărul cadrelor de informație ce pot fi emise, fără a avea confirmarea de recepție corectă pentru vreunul din ele, este limitat. Așa cum s-a arătat, cadrele de informație emise și neconfirmate (pozitiv) sunt memorate, pentru eventuale retransmiteri, în lista de

retransmitere. Prin urmare, limitând numărul de cadre ce pot fi emise fără a avea încă confirmarea de recepție, și lista de retransmitere are o limită maximă. Când lista se umple stația primară încetează transmiterea altor cadre de informație. Dacă stația secundară este saturată este suficient ca ea să nu mai transmită confirmări, ceea ce va avea drept consecință umplerea listei de retransmitere la stația primară, după ce această transmite un număr limitat de cadre, și încetarea transmiterii cadrelor de informație. Transmiterea lor va fi reluată după ce stația secundară începe din nou să transmită confirmări, care vor avea ca efect un proces de golire a listei de retransmitere. În felul acesta stația secundară controlează fluxul cadrelor de informație.



Funcționarea tehnicii de control al fluxului prin fereastră glisantă este prezentată în figura 3.14. După transmiterea unui cadru de informație marginea superioară a ferestrei avansează cu un pas și, la fel, după primirea unei confirmări pozitive, marginea inferioară avansează cu un pas. Dacă diferența între cele două margini devine egală cu dimensiunea maximă admisă pentru fereastră fluxul cadrelor de informație este stopat.

Așa cum s-a arătat, stația secundară trebuie să memoreze într-o listă de recepție, pentru a evita duplicatele, numerele de secvență ale ultimelor n cadre de informație recepționate corect. Setul acestor identificatori constituie *fereastra de recepție*.

În strategia de control prin retransmitere cu oprire și așteptare mărimea k a ferestrei de emisie este 1 și la fel este și mărimea n a ferestrei de recepție, egală cu 1. În strategia de retransmitere cu întoarcere la N fereastra de recepție are, de asemenea, $n=1$ deoarece stația secundară elimină orice cadru de informație recepționat care nu este următorul în secvență. În strategia de retransmitere cu repetare selectivă fereastra de recepție trebuie să fie egală cu cea de emisie, $n=k$, deoarece, spre exemplu, în cazul recepției corecte a unei secvențe de cadre al căror număr este k , și o eronare a confirmării pentru primul cadru, stația secundară trebuie să aibă posibilitatea să identifice copia lui, retransmisă de stația primară, ca un duplicat.

3.5.3 Numerele de secvență

Biții care reprezintă numerele de secvență fac parte din biții suplimentari (față de cei de informație, utili) ai cadrului. Eficiența unui protocol depinde și de numărul biților suplimentari în raport cu cei de informație și, din acest punct de vedere, este util să se determine domeniul minim de variație pentru numerele de secvență, adică numărul minim de identificatori necesari. Numărul identificatorilor necesari, pentru identificarea unică a fiecărui cadru de informație transmis, depinde de strategia de retransmitere utilizată și de mărimea ferestrei de emisie.

În strategia de retransmitere cu oprire și așteptare sunt necesari doi identificatori pentru a permite stației secundare să determine dacă un cadru recepționat este un cadru nou sau un duplicat. În mod tipic cei doi identificatori sunt 0 și 1 și variabila de secvență la emisie $V(S)$ va fi incrementată modulo 2.

În strategia de retransmitere cu întoarcere la N, cu fereastra de emisie k și fereastra de recepție 1, numărul identificatorilor trebuie să fie cel puțin $k + 1$. Dacă, spre exemplu, numărul identificatorilor ar fi k și toate cadrele de confirmare $ACK(0)$, $ACK(1)$, ..., $ACK(k-1)$ ar fi eronate, stația primară va retransmite cadrele de informație $I(0)$, $I(1)$, ..., $I(k-1)$ iar stația secundară nu va fi capabilă să determine dacă aceste cadre sunt duplicate sau reprezintă o nouă serie de cadre de informație. Dacă numerotarea ar fi modulo $k+1$ următorul cadru așteptat de stația secundară ar trebui să fie $I(k)$ și nu $I(0)$. Prin $I(j)$ s-a notat cadrul de informație cu numărul de secvență j , iar prin $ACK(j)$ cadrul de confirmare pozitivă corespunzător.

În strategia de retransmitere cu repetare selectivă, cu fereastră de emisie k , numărul identificatorilor trebuie să fie cel puțin $2k$. Presupunând, spre exemplu, că stația primară a transmis o secvență de cadre în număr de k , că toate au fost recepționate corect și toate cadrele ACK au fost eronate, stația secundară ar trebui să fie capabilă să determine dacă oricare din următoarele cadre, în număr de k , este un cadru nou sau un duplicat. Pentru aceasta este necesar să se aloce un nou set de k identificatori următoarei secvențe de cadre de informație, în număr de k , rezultând astfel că sunt necesari $2k$ identificatori. Tabelul următor prezintă sintetic dimensiunile ferestrelor de emisie și de recepție și numărul minim al identificatorilor pentru cele trei strategii de control prin retransmitere.

Strategie	Fereastra de emisie	Fereastra de recepție	Numărul identificatorilor
Oprire și așteptare	1	1	2
Întoarcere la N	k	1	$k+1$
Repetare selectivă	k	k	$2k$

3.5.4 Administrarea legăturii

Pentru ca mecanismele de control al erorii și al fluxului să funcționeze corect, așa cum au fost prezentate anterior, este necesar ca ambele părți în comunicație să fie inițializate pentru a fi gata să schimbe mesajele de informație. Spre exemplu, ambele părți trebuie să pornească cu aceleași variabile de secvență la emisie și la recepție și acest

În figura 3.15 se poate vedea cum, înainte de a transmite datele, utilizatorul serviciului furnizat de legătura de date (L), din terminalul sursă, utilizează primitiva de serviciu L-CONNECT.request pentru a cere stabilirea conexiunii logice cu terminalul destinație. Primind această primitivă de serviciu, entitatea de nivel legătură de date din terminalul sursă își inițializează variabilele de secvență și formează un cadru de inițializare (SETUP) pe care-l transmite entității de nivel legătură de date din terminalul de destinație. În terminalul de destinație entitatea de nivel legătură de date își inițializează variabilele de secvență, transmite utilizatorului serviciului L o primitivă de serviciu L-CONNECT.indication și un cadru de confirmare, nenumerotat (UA - unnumbered acknowledgement) către terminalul sursă. La primirea acestui cadru, entitatea de nivel legătură de date transmite utilizatorului L din terminalul sursă o primitivă de serviciu L-CONNECT.confirm. Urmează faza de transfer date și apoi faza de eliberare a legăturii.

3.6 Stările stațiilor conectate la legătura de date. Configurații posibile

Pentru formarea unei legături de date trebuie aleasă mai întâi configurația acesteia: punct-la-punct, multipunct, în buclă. Alegerea configurației depinde de mai mulți factori, precum: caracteristicile aplicației, costul legăturii, proprietățile suportului de transmisiune, performanțele dorite, repartizarea geografică a stațiilor.

Spre exemplu, o configurație punct-la-punct poate fi realizată pe o linie dedicată (închiriată) sau comutată și poate fi exploatată în modul unidirecțional, bidirecțional alternant sau bidirecțional simultan.

Configurația multipunct, utilizată în cazul în care repartizarea geografică și caracteristicile tehnice ale stațiilor o permit, asigură conectarea într-un mod economic a mai multor stații la o stație centrală. Această configurație necesită o linie dedicată. Legătura multipunct poate funcționa în modul centralizat, în care stația centrală inițiază totdeauna stabilirea comunicației cu oricare altă stație, sau în modul necentralizat, în care stația centrală comunică mai întâi cu una dintre celelalte stații și apoi, după încheierea comunicației, aceasta din urmă transmite controlul unei alte stații pentru comunicația cu stația centrală ș.a.m.d., până se ajunge la ultima stație, care va pasa controlul stației centrale, reluându-se apoi ciclul.

Legătura în buclă este un caz particular al legăturii multipunct necentralizate, în care fiecare stație funcționează ca un repetor.

După ce s-a ales configurația legăturii este necesar să se precizeze stările stațiilor, pentru a delimita responsabilitatea fiecăreia, atât în funcționarea normală cât și în situațiile de funcționare anormală a legăturii. Din acest punct de vedere se disting două feluri de stări: permanente și temporare.

Stări permanente

Într-o configurație multipunct sau în buclă controlul (supervizarea) legăturii se atribuie de regulă unei singure stații. Această stație este numită *stație de comandă* sau *primară* iar celelalte stații sunt *stații subordonate* sau *secundare* (fig. 3.16, a). Și într-o legătură punct-la-punct trebuie să existe o stație primară, care are responsabilitatea controlului legăturii, cealaltă stație fiind secundară.

Stația primară transmite comenzi către stația sau stațiile secundare, organizând astfel schimbul de date și asigurând controlul legăturii. De asemenea, în situațiile de funcționare anormală, are responsabilitatea restabilirii funcționării normale a legăturii.

Stația secundară execută comenzile primite de la stația primară și transmite acesteia răspunsurile.

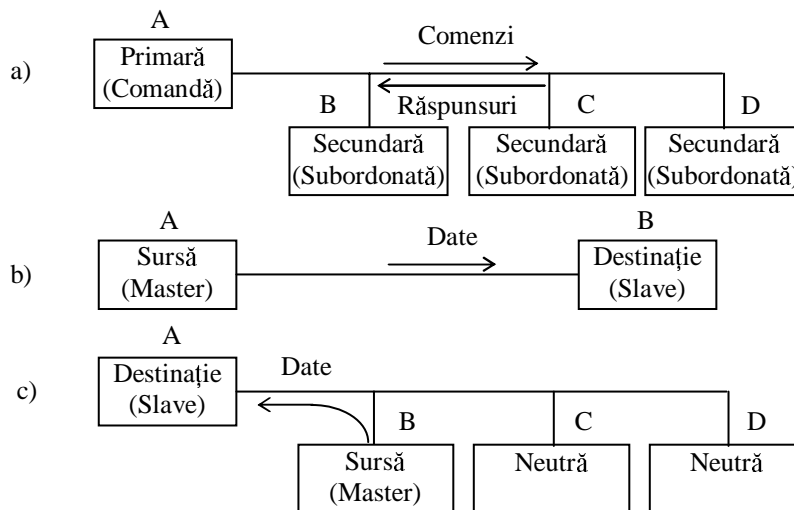


Fig. 3.16 Configurație multipunct, stări permanente (a) și temporare (b, c) ale stațiilor

Stări temporare

Stările temporare sunt atribuite stațiilor, așa cum arată și numele, în mod temporar, în funcție de rolul pe care-l au la un moment dat în cadrul schimburilor de date:

sursă de date (transmițătoare, master), *destinație* a datelor (receptoare, slave) sau nici una, nici alta, deci *neutră*. Aceste stări se schimbă în funcție de sensul de transmitere a datelor. Astfel, când datele sunt transmise de la stația primară A la stația secundară B, fig. 3.16 b, stația A este stație sursă, iar stația B este stație destinație. Când datele se transmit de la stația secundară B la stația primară A, fig. 3.16 c, stația B este stație sursă, iar stația A este stație destinație. În aceste situații stațiile secundare C și D sunt stații neutre, ele neparticipând la schimburile de date.

Atribuirea stărilor stațiilor într-o legătură de date

Totdeauna informația (datele) se transmite de la sursă la destinație și în sens invers se transmit confirmările de recepție, așa încât atribuirea stărilor de sursă și destinație pe o legătură de date nu necesită alte criterii. Există mai multe modalități de a realiza confirmarea de recepție. Spre exemplu, confirmările pot fi transmise ca răspunsuri independente sau pot fi incluse în alte mesaje de date. De asemenea, se poate transmite câte o confirmare pentru fiecare mesaj recepționat sau, cu o confirmare, se pot valida mai multe mesaje deodată.

Așa cum s-a arătat însă, pentru a asigura un transfer complet și corect al informației, este necesară o supervizare a legăturii de date și, în acest scop, au fost definite și atribuite stațiilor funcțiunile primară (de comandă) și secundară (subordonată).

Există mai multe moduri de asociere a perechilor de funcțiuni primară-secundară și sursă-destinație. Una din problemele ce trebuie avute în vedere la stabilirea unei asemenea asocieri este cea referitoare la modul de inițiere a transferului de informație. Din acest punct de vedere, în continuare, se vor prezenta câteva exemple.

Asocierea primară-sursă și secundară-destinație

Dacă se asociază funcția primară unei surse de informație și funcția secundară destinației (fig. 3.17 a), stația primară inițiază transferul de informație printr-un mesaj de control numit *invitație la recepție* ("selecting"). Stația secundară confirmă disponibilitatea pentru recepție și, în continuare, stația primară va transmite mesaje de date, iar cea secundară mesaje de confirmare.

Asocierea primară-destinație și secundară-sursă

Asocierea funcției primare cu stația de destinație și a funcției secundare cu stația sursă necesită, pentru inițierea transferului de informație, transmiterea de către stația

primară (fig. 3.17 b) a unui mesaj de invitație la emisie către stația secundară ("polling"). Stația secundară va transmite, dacă sunt pregătite, mesajele de informație și stația primară va transmite mesajele de confirmare a recepției.

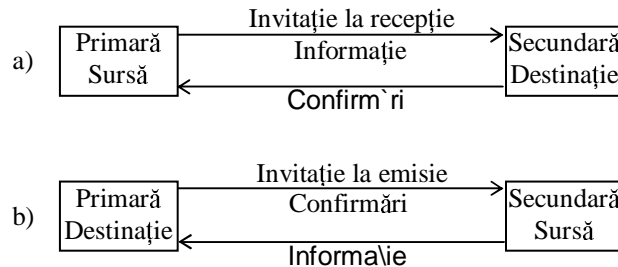


Fig. 3.17 Controlul transferului de informație prin:
a) invitație la recepție; b) invitație la emisie

Este evident că, în configurația multipunct, transmiterea invitației de a emite pentru o stație secundară care nu este disponibilă, adică nu are pregătit un mesaj de informație pentru stația primară, înseamnă pierdere de timp și utilizare inefficientă a legăturii de date. De asemenea, chiar transmiterea invitației la emisie înseamnă o pierdere de timp. Pentru a evita astfel de pierderi o soluție constă în a permite stațiilor secundare să încerce să emită atunci când au nevoie (la dorință). În această variantă pot apărea situații de contenție (coliziune), adică situații în care mai multe stații încearcă să transmită în același timp și protocolul legăturii de date trebuie să aibă în vedere rezolvarea acestora.

Și în legăturile punct-la-punct, cu exploatare bidirecțională alternantă, este posibil să apară coliziuni dacă stațiile primară și secundară pot emite la dorință. Astfel de probleme trebuie rezolvate prin procedura de comandă (control) a legăturii de date.

Dacă în configurațiile multipunct și în buclă atribuirea stărilor nu ridică probleme, fiecare stație putând fi sursă sau destinație dar numai o anumită stație, mereu aceeași, este primară, celelalte fiind stații secundare, în legăturile punct-la-punct sunt mai multe combinații posibile, fiecare cu avantajele și dezavantajele sale. O configurație frecvent utilizată este cea ce corespunde unei asemenea combinări a funcțiilor primară-secundară și sursă-destinație încât cele două stații joacă un rol simetric, atât pentru transferul informației cât și pentru controlul legăturii. În cele ce urmează sunt prezentate câteva configurații posibile.

Configurația punct-la-punct simetrică

Această configurație utilizează asocierea primară-sursă și secundară-destinație și invitația la recepție, cele două stații jucând un rol simetric pentru transferul mesajelor de informație și de control (fig. 3.18).

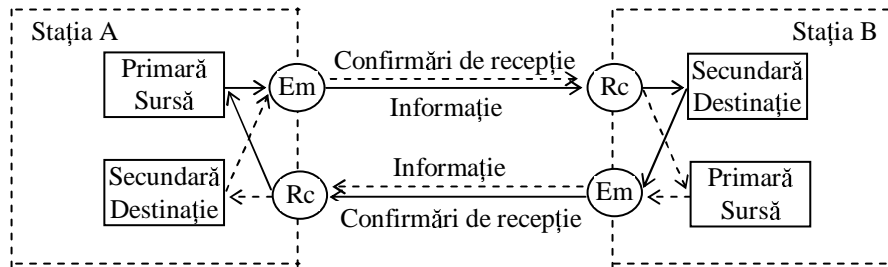


Fig. 3.18 Configurația punct-la-punct simetrică

Inconvenientul acestei asocieri constă în faptul că, neutilizând invitația la emisie, nu poate fi folosită și în legăturile multipunct. În multe aplicații se dorește o compatibilitate maximă între diferitele configurații ale legăturii de date, în special atunci când o stație mare este inclusă în mai multe configurații, punct-la-punct, multipunct sau în buclă.

Configurația punct-la-punct nesimetrică

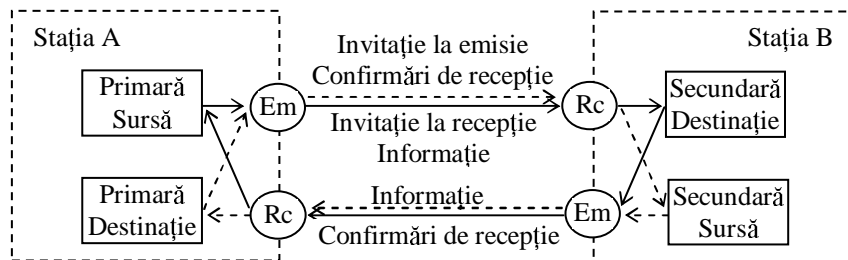


Fig. 3.19 Configurația punct-la-punct nesimetrică

Deoarece legătura-punct-la-punct este un subansamblu al legăturii multipunct, iar legătura multipunct este în esență nesimetrică, din cauza atribuirii funcției primare singurei stații de comandă, se poate concepe o repartitie nesimetrică a funcțiilor primară-secundară și sursă-destinație, rezultând configurația punct-la-punct nesimetrică (fig. 3.19). Această configurație folosește atât invitația la emisie, cât și invitația la recepție. Pentru a forma o configurație multipunct este suficient să se conecteze stații secundare în paralel pe legătura punct-la-punct.

Configurația punct la punct echilibrată

Configurațiile anterioare utilizează diferite moduri de asociere a funcțiilor primară-secundară și sursă-destinație. Cele două perechi de funcțiuni sunt însă independente și se poate elabora o configurație în care ele să fie separate, să nu mai existe o asociere permanentă a lor (fig. 3.20).

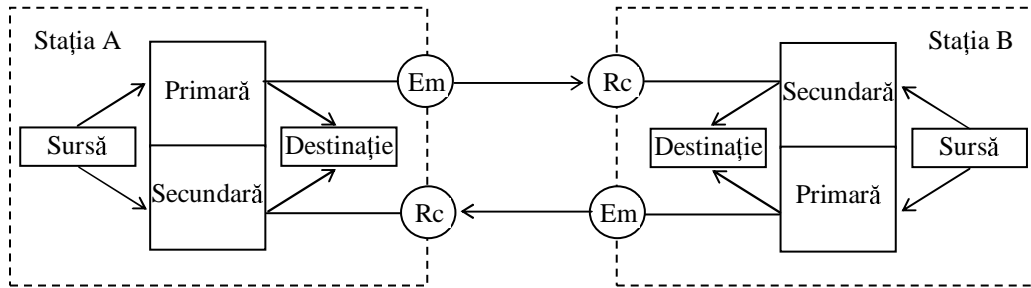


Fig. 3.20 Configurație punct-la-punct echilibrată

Prin separarea funcțiilor primară-secundară de funcțiile sursă-destinație informația poate fi transmisă fie ca o comandă (asociere temporară primară-sursă), fie ca un răspuns (asociere temporară secundară-sursă). Transmiterea informației ca un răspuns prezintă avantajul că mesajului de informație transferat într-un sens i se poate atașa și confirmarea pentru transferul de informație în celălalt sens, mărindu-se astfel randamentul legăturii. În plus, această configurație prezintă o compatibilitate mai bună cu configurația multipunct, deoarece funcția secundară poate fi atribuită și sursei de date.

3.7 Protocoale orientate pe caracter

Protocoalele orientate pe caracter sunt utilizate atât pe legăturile punct-la-punct, cât și pe legăturile multipunct. După cum se face transferul de informație în cele două sensuri protocoalele pot fi clasificate în:

- protocoale simplex, cu transfer al informației într-un singur sens;
- protocoale semiduplex, cu transfer al informației în ambele sensuri, dar nu simultan;
- protocoale duplex, cu transfer al informației în ambele sensuri, simultan.

3.7.1 Protocoale simplex

Protocoalele simplex sunt utilizate numai pe legăturile punct-la-punct, aplicațiile tipice constând în transferul de fișiere. Dintre cele mai cunoscute protocoale de acest tip fac parte protocolul Kermit și protocoalele X-modem și Y-modem.

Protocolul Kermit

Este folosit în transmisiunea sincronă, procedura de control al erorii fiind de tipul cu oprire și așteptare. Există mai multe versiuni ale acestui protocol, în funcție de tipul stațiilor între care se face transferul de fișiere (între calculatoare personale, între un calculator și un server de fișiere sau între un calculator personal și un calculator mare - mainframe), deosebirile constând în modul în care se inițiază procesul de transfer. În cele ce urmează va fi prezentată versiunea pentru transferul de fișiere între două calculatoare personale.

Utilizatorii dispun de un set de comenzi pe care le folosesc după ce programele Kermit au fost rulate în cele două calculatoare (fig. 3.21). Se presupune că între cele două calculatoare este disponibil un circuit fizic, spre exemplu prin intermediul rețelei telefonice, cu modemuri de bandă vocală.

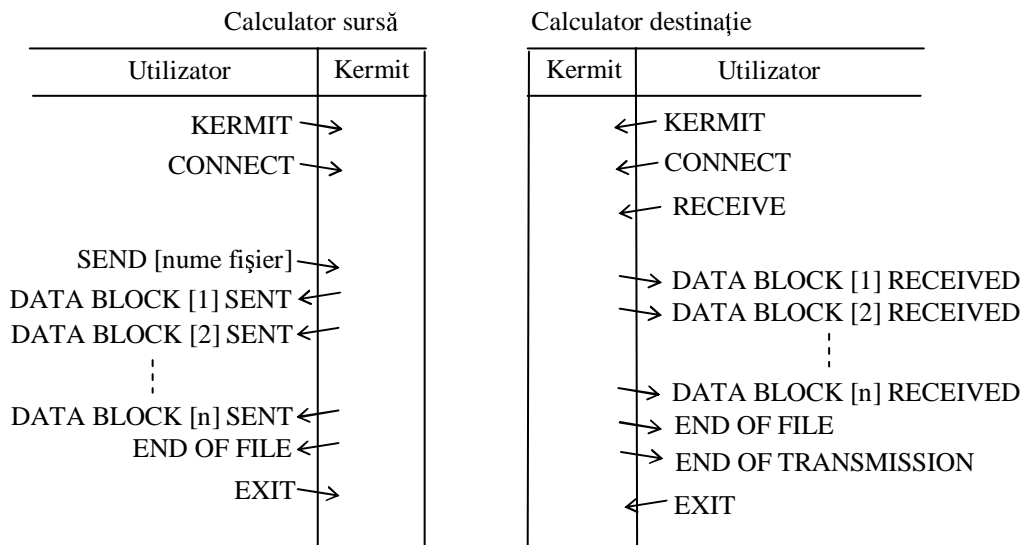


Fig. 3.21 Comenzi Kermit

După rularea programului Kermit în fiecare dintre cele două calculatoare, fiecare utilizator introduce comanda CONNECT, care va avea ca efect inițializarea circuitului de

date (fizic) între cele două calculatoare. Utilizatorul calculatorului în care se va transfera fișierul introduce apoi comanda RECEIVE, iar cel al calculatorului de la care se transferă fișierul introduce comanda SEND, urmată de numele fișierului. După aceste comenzi se realizează transferul fișierului, segmentat, pe ecranele monitoarelor celor două calculatoare fiind afișate mesaje după fiecare segment transferat. După ce transferul fișierului s-a terminat ambii utilizatori ies din Kermit prin comanda EXIT și revin în sistemul local de operare. Pentru transferul de fișiere în sens invers se execută, în mod corespunzător, aceleași comenzi. Se observă că Kermit este mai mult decât un protocol de legătură de date, deoarece realizează și funcțiuni adiționale, cum ar fi: citire/scriere, segmentarea fișierului și reasamblarea sa.

Formatul cadrelor este prezentat în figura 3.22.

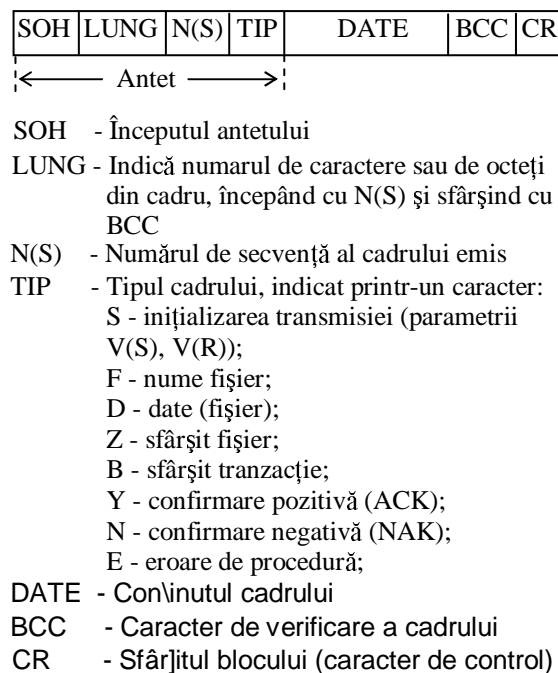


Fig. 3.22 Formatul cadrului Kermit

Începutul cadrului este marcat de caracterul de control SOH (start-of-header, începutul antetului). Urmează un alt octet care indică lungimea cadrului, începând cu N(S) și sfârșind cu BCC, în număr de caractere sau de octeți (pentru fișiere cu date pur binare). Numărul care indică lungimea este codat în exces față de 32, domeniul de valori

începând cu 35 (lungime minimă - 3 caractere: N(S), TIP și BCC, fără câmp de date) și sfârșind cu 126 (lungime maximă - 94 caractere, din care 91 caractere în câmpul de date). N(S), numărul de secvență al cadrului, este un octet care reprezintă numărul de ordine al cadrului, codat în exces față de 32. Numerotarea se face modulo 64, pentru cadrul 0 corespunzând numărul 32, iar pentru cadrul 63 corespunzând numărul 95. Câmpul TIP conține un caracter care indică tipul cadrului. Tipurile de cadre sunt menționate în figura 3.22.

Protocoalele X modem și Y modem

Protocoalele X modem și Y modem, elaborate în 1977 și, respectiv, în 1982, sunt utilizate tot pentru transferul de fișiere. Ambele folosesc procedura de control al erorii cu oprire și așteptare. Formatul cadrului pentru protocolul X modem este prezentat în figura 3.23. Câmpul CRC poate fi completat cu un octet de verificare a parității sau cu doi octeți rezultați în urma codării cu un cod ciclic al cărui polinom generator este de grad 16.

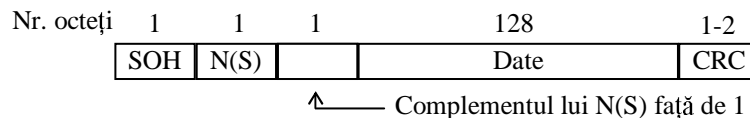


Fig. 3.23 Formatul cadrului pentru protocolul X modem

O variantă mai eficientă a acestui protocol, pentru că folosește și cadre cu lungimea câmpului de date de 1024 octeți, este protocolul X modem - 1k. În locul caracterului SOH se folosește caracterul STX pentru a marca începutul cadrului.

Protocolul Y modem prezintă ușoare modificări față de protocolul X modem -1k. Cu acest protocol se pot transmite mai multe fișiere într-o sesiune. Varianta Y modem - G, elaborată în 1985, prezintă modificări ce permit să nu se transmită confirmare după fiecare cadru, ci numai la sfârșitul transmiterii fișierului. De asemenea, în această variantă, receptorul are posibilitatea să întrerupă transferul, în particular dacă detectează un bloc eronat, transmițând mai multe caractere CAN (cancel - ignoră).

Din aceeași serie a protocoalelor X și Y modem face parte și protocolul Z modem, un protocol duplex, cu cadre de lungime variabilă și mecanism care asigură transparența datelor și care utilizează coduri ciclice cu polinom generator de grad 16 sau 32.

3.7.2 Protocoale semiduplex

Cele mai multe protocoale orientate pe caracter sunt protocoale semiduplex, care utilizează procedura de control al erorii cu oprire și așteptare. Marile companii producătoare de echipamente de calcul au elaborat propriile lor protocoale. Cel mai cunoscut protocol este cel elaborat în 1968 de compania IBM, numit "binary synchronous control" și notat BSC sau Bisync. El a constituit baza pentru protocolul orientat pe caracter elaborat de ISO și numit *control al legăturii în mod de bază* (basic mode link control). Alte două versiuni ale acestui protocol sunt numite *modul conversațional* și *modul transparent*. Conform terminologiei ISO (standard ISO 2382-8) stațiile conectate la o legătură de date pe care se folosește acest protocol sunt numite stație de comandă și stație (stații) subordonată (subordonate).

Modul de bază

Caracteristicile funcționale ale acestui protocol sunt:

- se folosește pe legăturile punct-la-punct și pe legăturile multipunct;
- codul utilizat este alfabetul internațional nr.5 (ITU-T); varianta IBM folosește codul EBCDIC;
- poate utiliza, în principiu, atât transmisiunea asincronă cât și transmisiunea sincronă, dar practic folosește transmisiunea sincronă;
- modul de exploatare este bilateral alternant (semiduplex).

Caracterele de control utilizate sunt:

SOH - start-of-header, indică începutul antetului unui cadru (când există antet; numai cadrele de informație au antet, cele de control nu au);

STX - start-of-text, încheie antetul și semnifică începutul textului (câmpul datelor);

ETX -end-of-text, specifică sfârșitul textului;

EOT - end-of-transmission, indică sfârșitul transmisiei, al unuia sau al mai multor cadre de informație și eliberează conexiunea logică;

ENQ - enquiry, utilizat ca cerere de răspuns de la o altă stație în procedurile poll-select (interogare-selectare); răspunsul poate include identificarea și/sau starea stației;

ACK - confirmare pozitivă; poate fi folosit cu 0 și 1 pentru confirmarea cadrelor cu număr de secvență par sau impar (ACK 0, ACK 1);

NAK - confirmare negativă, urmat de 0/1;

DLE - data link escape, utilizat pentru a schimba semnificația altor caractere de control;

SYN - utilizat pentru sincronizarea de caracter;

ETB -end-of-transmission block; utilizat pentru a indica sfârșitul unui cadru (bloc) de informație atunci când mesajul este divizat în mai multe cadre.

Formatul cadrelor

Sunt două tipuri de cadre: de informație și de control. Cadrele de informație au o lungime variabilă, limita maximă fiind stabilită ținând seama de caracteristicile circuitului de date, astfel încât să rezulte un randament cât mai bun al protocolului. O valoare tipică a lungimii este 256 de caractere. Toate cadrele sunt precedate de cel puțin două caractere SYN.

Caracterul de verificare a cadrului (BCC - block check character) urmează după caracterul ETX sau ETB. Sunt verificate câmpurile de la STX până la ETB sau ETX inclusiv. În cele mai multe sisteme care folosesc codul EBCDIC se utilizează un cod ciclic (CRC-16) în locul verificării parității prin BCC.

Dacă mesajul este divizat în mai multe cadre, fiecare cadru se termină cu ETB, mai puțin ultimul, care se termină cu ETX. Formatele diferitelor tipuri de cadre sunt prezentate în figura 3.24. Identificatorul unui cadru de informație este reprezentat de numărul de secvență la emisie, 0 sau 1. În câmpul de adresă se trece totdeauna adresa stației subordonate către care este transmis cadrul (de către stația de comandă) sau care transmite cadrul (către stația de comandă). Dacă stația invitată să emită (prin mesaj "poll") are de transmis mesaj de informație va folosi formatul cadrelor de informație, iar dacă nu are de transmis un astfel de mesaj va transmite un cadru EOT (fig. 3.24j).

Procedurile de restabilire a legăturii

Procedurile de restabilire sunt bazate pe utilizarea de temporizatoare și numărătoare. Dacă o stație de comandă sau o stație sursă nu obține un răspuns valid (conform protocolului) la o secvență de control sau la un cadru de informație, la expirarea unui timp dat ea poate proceda astfel:

- repetă secvența sau cadrul până la de n ori;
- trimite o cerere de răspuns (ENQ), de n ori eventual;

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{SOH}} \boxed{\text{Identificator}} \boxed{\text{Adresa stației}} \boxed{\text{STX}} \boxed{\text{Date}} \boxed{\text{ETX}} \boxed{\text{BCC}}$

a) Mesaj într-un singur cadru

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{SOH}} \boxed{\text{Identificator}} \boxed{\text{Adresa stației}} \boxed{\text{STX}} \boxed{\text{Date}} \boxed{\text{ETB}} \boxed{\text{BCC}}$

b) Primul cadru într-un mesaj divizat

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{SOH}} \boxed{\text{Identificator}} \boxed{\text{STX}} \boxed{\text{Date}} \boxed{\text{ETB}} \boxed{\text{BCC}}$

c) Cadru intermediar într-un mesaj divizat

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{SOH}} \boxed{\text{Identificator}} \boxed{\text{STX}} \boxed{\text{Date}} \boxed{\text{ETX}} \boxed{\text{BCC}}$

d) Ultimul cadru într-un mesaj divizat

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{EOT}} \boxed{\text{Adresa stației}} \boxed{\text{P/S}} \boxed{\text{ENQ}}$

e) Secvența poll/select (P=poll, S=select)

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{ACK}}$

f) Răspuns pozitiv la "select"

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{NAK}}$

g) Răspuns negativ la "select"

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{ACK}} \boxed{0/1}$

h) Confirmare pozitivă pentru cadre pare/impare

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{NAK}} \boxed{0/1}$

i) Confirmare negativă pentru cadre pare/impare

$\boxed{\text{SYN}} \boxed{\text{SYN}} \boxed{\text{EOT}}$

j) Sfârșitul transmisiunii

Fig. 3.24 Formatele cadrelor de informație și de control

- termină transmisia, transmițând EOT.

După n repetări sau cereri fără succes va fi informat nivelul superior sau operatorul, sau și unul și altul.

Modul transparent

Când se transmit date pur binare se folosește caracterul de control DLE pentru a asigura transparența datelor (par. 3.3). De asemenea, deoarece datele pur binare sunt

transmise sub forma unor secvențe de octeți, nu se poate folosi bitul de paritate (al optulea) și se folosește verificarea ciclică (CRC-16).

Modul conversațional

Este o extensie a modului de bază, care permite funcționarea în procedura de dialog. După intrarea în faza de transfer date conform modului de bază simplu, o stație poate răspunde la un cadru de informație printr-un alt cadru de informație, care ține loc și de confirmare pozitivă. Sunt introduse următoarele restricții:

- numai o confirmare pozitivă poate fi înlocuită de un cadru de informație;
- un cadru de informație nu poate fi transmis în locul unei confirmări pozitive decât atunci când cadrul de informație recepționat se termină prin ETX (cazul ETB nu este permis).

În figura 3.25 sunt prezentate exemple de funcționare a protocolului în modul de bază și în modul conversațional.

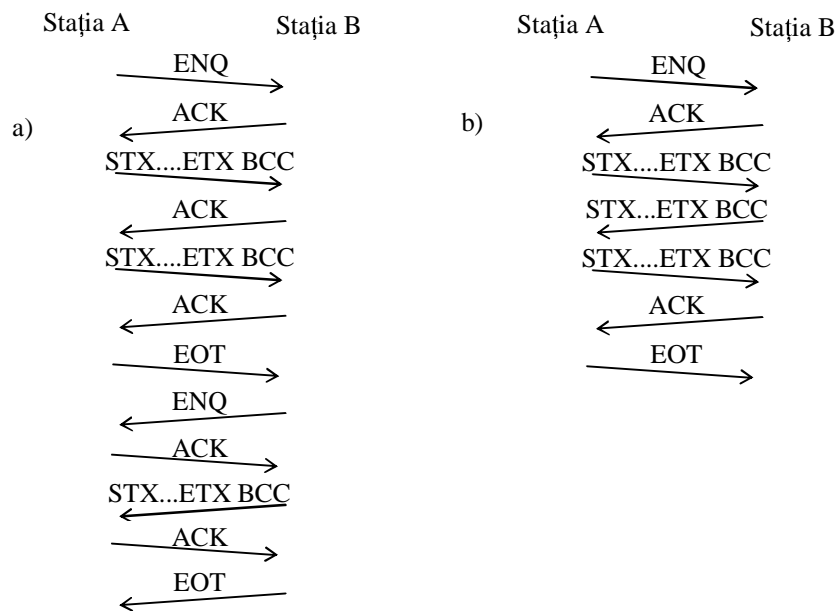


Fig. 3.25 Modul de bază (a) și modul conversațional (b)

3.8 Protocoale orientate pe bit

Protocoalele mai noi elaborate pentru legătura de date sunt protocoale orientate pe bit, acestea prezentând, așa cum s-a arătat în par. 3.3, o serie de avantaje în comparație cu

protocoalele orientate pe caracter. Practic, diferitele protocoale orientate pe bit sunt variante ale protocolului de *control al legăturii de date de nivel înalt* (HDLC - High-level data link control), elaborat de ISO.

3.8.1 Protocolul HDLC

Conform terminologiei ISO (standard ISO 2382-8), în cazul utilizării acestui protocol, cea parte a unei stații de date care asigură funcțiile de control primar al legăturii de date, generează comenzile ce trebuie transmise și interpretează răspunsurile primite se numește *stație primară*. Responsabilitățile specifice ale stației primare includ inițializarea schimbului de semnale de control, organizarea fluxului de date și operațiunile privind controlul erorii și funcțiunile de restabilire. Partea stației de date care execută funcțiile de control al legăturii de date corespunzător instrucțiunilor stației primare, interpretează comenzile recepționate și generează răspunsurile pentru a fi transmise se numește *stație secundară*. O stație de date capabilă să îndeplinească rolul unei stații primare sau al unei stații secundare se numește *stație combinată*.

Protocolul HDLC permite funcționarea în modul duplex, asigurând transmiterea datelor în ambele sensuri simultan și poate fi utilizat în diferite configurații: legături punct-la-punct, în configurații echilibrate sau neechilibrate, și legături multipunct. În oricare dintre aceste configurații o singură stație este stație primară, cealaltă sau celelalte sunt stații secundare. Cadrele transmise de stația primară sunt considerate *comenzi*, cele transmise de stația secundară sunt considerate *răspunsuri* (figura 3.26). Protocolul HDLC are trei moduri (proceduri) de operare:

- *Modul de răspuns normal* (NRM - Normal Response Mode), utilizat în configurațiile neechilibrate punct-la-punct și în cele multipunct, în care stația secundară poate transmite numai dacă a fost invitată de stația primară.

- *Modul de răspuns asincron* (ARM - Asynchronous Response Mode), utilizat de asemenea în configurațiile neechilibrate, de obicei punct-la-punct, în care o stație secundară poate iniția transmisiunea fără a primi o invitație de la stația primară, astfel că ea transmite cadre în mod asincron în raport cu cea primară.

- *Modul echilibrat asincron* (ABM - Asynchronous Balanced Mode), utilizat în configurațiile echilibrate, punct-la-punct, spre exemplu pentru comunicații calculator-

calculator sau între un calculator și o rețea publică de date. În acest mod de operare fiecare stație are același statut, realizând atât funcțiuni de stație primară, cât și funcțiuni de stație secundară. Această procedură, cunoscută și sub denumirea de procedură de acces al legăturii echilibrată (LAPB - Link access procedure balanced), este folosită în rețelele publice de date cu comutație de pachete, pe conexiunea utilizator-rețea, specificată de Recomandarea ITU-T X.25.

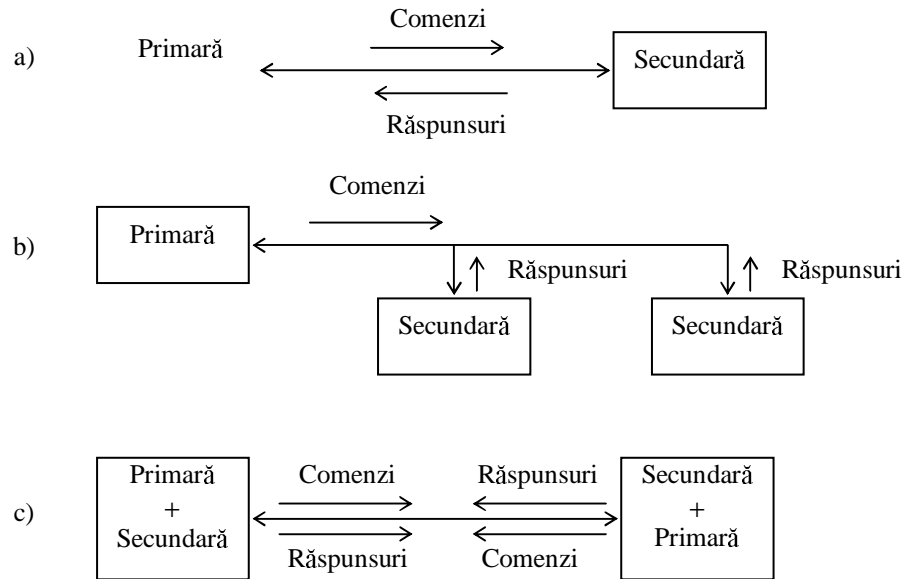


Fig. 3.26 Configurații HDLC:

- a) punct-la-punct neechilibrată (o stație, totdeauna aceeași, este primară, cealaltă este secundară);
- b) multipunct neechilibrată (o stație primară, celelalte secundare);
- c) punct-la-punct echilibrată (fiecare stație poate fi primară, cealaltă este secundară)

Structura cadrului HDLC

Formatul cadrului HDLC este prezentat în figura 3.27.

FANION	ADRESA	CONTROL	INFORMAȚIE	FCS	FANION
01111110	8/16 biți	8/16 biți	0 ÷ N biți	16/32 biți	01111110

Fig. 3.27 Formatul cadrului HDLC

Cadrul este delimitat de fanioane și, pentru a asigura transparența datelor, transmițătorul introduce în mod automat un simbol 0 după o secvență de cinci simboluri 1 consecutive care apare în interiorul cadrului, simbol ce va fi eliminat de receptor. Un același fanion poate fi delimitator de sfârșit de cadru, dar și delimitator de început de cadru pentru cadrul care urmează.

În câmpul de adresă se trece totdeauna adresa stației secundare, căreia îi este destinat cadrul, dacă acesta este emis de stația primară, sau care a emis cadrul respectiv, dacă acesta este un cadru de răspuns. Pe legăturile multipunct adresa identifică totdeauna stația secundară de destinație - pentru cadrul emis de stația primară, sau sursă - pentru cadrul emis de o stație secundară. Se pot folosi și adrese de grup (pentru destinație), în cazul în care cadrul este destinat mai multor stații secundare sau de difuziune (broadcast), când cadrul este destinat tuturor stațiilor secundare. Dacă numărul stațiilor secundare este mare se pot folosi adrese de 16 biți. Bitul din prima poziție a câmpului de adresă arată dacă adresa este de 8 biți (1) sau de 16 biți (0).

Pe legăturile punct-la-punct nu este nevoie de identificarea stației sursă sau a stației destinație. Prin câmpul de adresă se stabilește tipul cadrului: de comandă sau de răspuns. Într-un cadru de comandă se trece adresa stației care va primi cadrul, într-un cadru de răspuns se trece adresa stației care transmite cadrul.

Prin câmpul de control sunt definite mai multe tipuri de cadre (figura 3.24 a), împărțite în trei categorii, numite: de informație (I - information), de control sau de supervizare (S - supervisory) și nenumerotate (U - unnumbered). N(S) și N(R) sunt numerele de secvență la emisie și la recepție, formate din trei biți (numerotare modulo 8). Pe legăturile cu timp mare de propagare numerotarea cadrelor de informație se face modulo 128, numerele de secvență N(S) și N(R) sunt formate din 7 biți și câmpul de control are 16 biți (figura 3.28 b).

Numărul de secvență la emisie N(S) reprezintă numărul de ordine în secvență și, totodată, identificatorul cadrului de informație din care face parte câmpul de control respectiv. Numărul de secvență la recepție N(R) reprezintă numărul de ordine al cadrului de informație ce se așteaptă a fi recepționat (în secvența cadrelor transmise în sens invers), ceea ce înseamnă o confirmare pentru recepția corectă și în ordine a cadrelor de informație până la cadrul cu numărul $N(R)-1$, inclusiv (par. 3.5).

a)

(Bit)	1	2	3	4	5	6	7	8
I	0	N(S)			P/F	N(R)		
S	1	0	S	S	P/F	N(R)		
U	1	1	M	M	P/F	M	M	M

b)

(Bit)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
I	0	N(S)							P/F	N(R)						
S	1	0	S	S	—				P/F	N(R)						
U	1	1	M	M	P/F	M	M	M	P/F	—						

Fig. 3.28 Câmpul de control: a) 8 biți, b) 16 biți

Bitul notat P/F, de la "poll" și "final", se numește bit P în cadrele de comandă (orice cadru emis de stația primară) și bit F în cadrele de răspuns (orice cadru emis de o stație secundară). Bitul P=1 semnifică solicitarea de răspuns imediat (confirmare) de la stația secundară. În răspunsul la un astfel de cadru se va pune F=1.

Primul sau primii doi biți ai câmpului de control identifică una din cele trei categorii de cadre.

Cadre de supervizare

Prin cei doi biți S din cadrele de supervizare sunt definite patru tipuri de astfel de cadre, fiecare dintre ele putând fi o comandă sau un răspuns, utilizate de stația primară sau secundară astfel:

RR - Recepția gata (Receive ready), pentru a semnala disponibilitatea pentru recepția informației sau pentru a confirma recepția corectă a cadrelor numerotate până la $N(R)-1$;

RNR - Recepția nu este gata (Receive not ready), pentru a semnala lipsa, temporară, a disponibilității pentru recepția informației și pentru a confirma recepția corectă a cadrelor numerotate până la $N(R)-1$;

REJ - Rejectează (Reject), pentru a cere transmiterea sau retransmiterea cadrelor de informație începând cu cel numerotat cu $N(R)$;

SREJ - Rejectează selectiv (Selective reject), pentru a cere transmiterea sau retransmiterea unui singur cadru de informație, cel numerotat prin N(R).

Cadre nenumerate

Prin biții notați cu M sunt definite mai multe tipuri de cadre nenumerate, unele dintre ele sunt comenzi, altele sunt răspunsuri. Sunt nouă cadre de comandă. Prin șase dintre ele se stabilește modul de funcționare, trei pentru numerotare modulo 8 și trei pentru numerotare extinsă, modulo 128:

SNRM (Set normal response mode) - răspuns normal, stația secundară transmite numai la invitația stației primare, care controlează legătura;

SARM (Set asynchronous response mode) - răspuns asincron, stația secundară poate iniția transmisia fără a fi invitată de stația primară;

SABM (Set asynchronous balanced mode) - echilibrat asincron, fiecare stație poate fi primară și secundară;

SNRME (Set normal response mode extended), SARME, SABME - aceleași moduri de funcționare, numerotare extinsă.

Celelalte trei cadre de comandă sunt:

RSET (Reset) - pentru a iniția restabilirea legăturii;

FRMR (Frame reject) - pentru a cere resetarea legăturii (atunci când într-un cadru recepționat un număr de secvență este în afara domeniului normal de valori, ceea ce înseamnă o nesincronizare a numerelor de secvență);

DISC (Disconnect) - pentru a cere eliberarea legăturii.

Cadrele de răspuns nenumerate, transmise de o stație secundară, sunt:

UA (Unnumbered acknowledgement) - confirmare nenumerotată, pentru a indica stației primare recepția și acceptarea comenzilor nenumerate prezentate mai sus;

CMDR (Command reject) - comandă rejectată, pentru a indica rejectarea unei comenzi recepționate corect;

DM (Disconnect mode) - mod deconectat, pentru a refuza o comandă de stabilire a modului de funcționare, arătând că stația este deconectată logic;

FRMR (Frame reject) - pentru a indica o nesincronizare a numerelor de secvență, caz în care este necesară reinițializarea legăturii.

Funcționarea protocolului

Principalele funcțiuni ale protocolului sunt administrarea legăturii și transferul datelor.

Administrarea legăturii

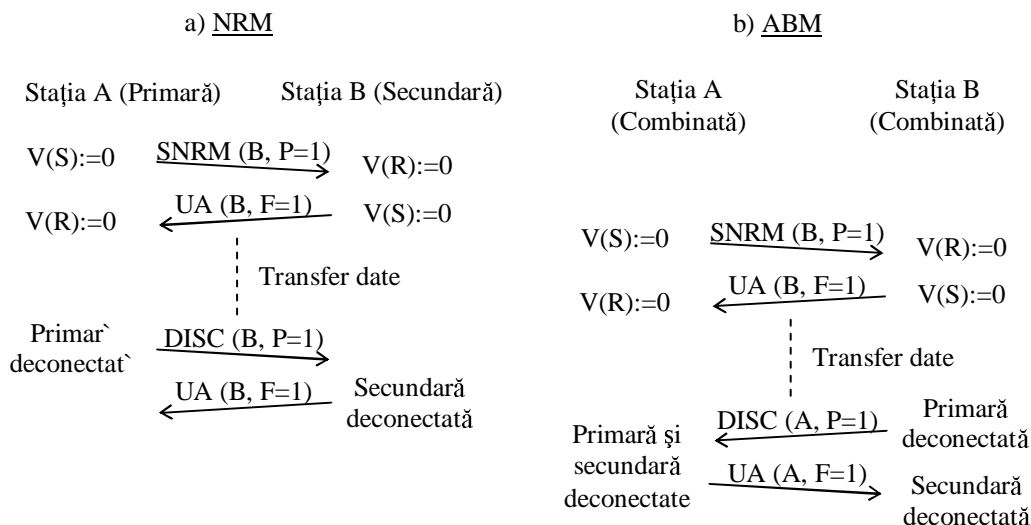


Fig. 3.29 Stabilirea și eliberarea conexiunii logice

Înainte de a transmite datele, pe linii punct-la-punct sau multipunct, între cele două stații în comunicație trebuie stabilită o conexiune logică, ceea ce se realizează prin schimbul de cadre nenumerotate. De asemenea, după transferul datelor, este necesară eliberarea conexiunii logice.

În modul NRM comenzile pentru stabilirea și eliberare conexiunii logice sunt transmise de stația primară (figura 3.29 a). În modul ABM oricare stație poate iniția stabilirea conexiunii logice. De asemenea, ambele stații pot iniția transferul cadrelor de informație în mod independent, astfel încât fiecare stație este o stație combinată deoarece trebuie să funcționeze și ca stație primară și ca stație secundară. În exemplul din figura 3.29 b stația A inițiază stabilirea conexiunii logice, ca stație primară, iar stația B, tot ca stație primară, o eliberează.

Transferul datelor

În NRM toate cadrele de informație sunt transferate sub controlul stației primare, una anumită, mereu aceeași. Stația primară transmite un cadru cu $P=1$, spre exemplu un cadru SNRM, pentru a invita stația secundară să transmită cadre de informație. Dacă

stația secundară are de transmis cadre de informație le va transmite, cu bitul $F=1$ numai în ultimul cadru pentru a semnala că acesta este ultimul în secvență. Dacă nu are cadre de informație pregătite pentru transmisie va răspunde printr-un cadru RNR cu bitul $F=1$.

Pentru legăturile punct-la-punct ABM este posibil un transfer de cadre I în ambele sensuri simultan. Confirmările se pot include chiar în cadrele de informație.

Așa cum s-a arătat în par. 3.5 numărul identificatorilor $N(S)$ distincți trebuie să fie cel puțin $k+1$ în strategia cu întoarcere la N (GBN, Go-back-N) și cel puțin $2k$ în strategia cu repetare selectivă, k fiind dimensiunea ferestrei de emisie. Prin urmare, cu o numerotare modulo 8 a cadrelor de informație, fereastra de emisie poate fi cel mult 7 ($k=7$) în GBN și cel mult 4 în repetarea selectivă.

Emiterea cadrelor de informație este oprită dacă $V(S) = \text{ultimul } N(R) \text{ recepționat} + k$. La recepția fiecărui cadru de informație stația secundară verifică dacă numărul de secvență $N(S)$ al acestuia și $N(R)$ sunt în domeniile de valori normale. Se poate deduce ușor că domeniile de valori normale pentru $N(S)$ și $N(R)$ dintr-un cadru recepționat sunt date de relațiile:

$$V(R) \leq N(S) < V(R) + k$$

$$V(S) \geq N(R) \geq V(S) - \text{numărul de cadre din lista de retransmitere}$$

În lista de retransmitere fiecare cadru I este plasat după ce a fost emis și este scos când s-a primit confirmarea pozitivă pentru el. Lista are lungimea maximă k și când se umple se oprește emisia cadrelor de informație.

Dacă $N(S)=V(R)$ totul este în ordine și cadrul este acceptat. Dacă $N(S) \neq V(R)$, dar este totuși în domeniul de valori normale, înseamnă că un cadru a fost eronat și va fi returnat un cadru REJ (în GBN) sau un cadru SREJ (repetare selectivă), indicând stației primare că a apărut o eroare în secvență și specificând de la ce cadru să înceapă retransmiterea, respectiv ce cadru să retransmită.

Dacă $N(S)$ și $N(R)$ sunt în afara domeniului de valori normale, înseamnă că numerele de secvență de la cele două capete sunt desincronizate și legătura trebuie reinițializată. Dacă această situație este detectată de către stația secundară, ea va transmite un cadru FRMR (în ABM) sau CMDR (în NRM) către stația primară care, la rândul ei, va elimina cadrele din lista de retransmitere și va proceda la reinițializarea legăturii prin comanda SABM/SNRM, așteptând apoi răspunsul UA.

Controlul fluxului este realizat, la stația primară, prin fereastra de emisie și, de către stația secundară, prin transmiterea de cadre RNR (prin cadre RR se reia transmisia).

3.8.2 Variante HDLC

a) LAPB (Link access procedure balanced - Procedură de acces al legăturii echilibrată). Este utilizată pe legăturile punct-la-punct duplex, între un calculator și o rețea cu comutație de pachete (publică sau privată, de tip X.25), de fapt între calculator și comutatorul de pachete la care acesta este conectat. Procedura funcționează în ABM.

Adresele celor două stații combinate de la capetele legăturii, numite DTE (Echipament terminal de date) și DCE (Echipament de terminație a circuitului de date - ceea ce "vede" terminalul spre rețea dar, de fapt, celălalt capăt al legăturii îl constituie comutatorul de pachete) sunt utilizate pentru a stabili tipul cadrului transmis - comandă sau răspuns. Adresele celor două stații sunt 03 Hex (DTE) și 01 Hex (DCE). Cadrul care conține adresa stației ce l-a emis este un cadru de răspuns, iar cel ce conține adresa stației ce-l va recepționa este un cadru de comandă. Astfel, dacă un cadru emis de DTE poartă adresa 01 este un cadru de comandă, iar dacă poartă adresa 03 este un cadru de răspuns. Toate cadrele de informație sunt cadre de comandă.

b) MLP (Multilink procedure - Procedura multilegătură). În unele aplicații, din motive de trafic sau de fiabilitate, se folosesc mai multe legături (conexiuni fizice) între două stații. Pentru controlul legăturilor de date stabilite pe fiecare conexiune fizică s-a definit o variantă de protocol, extensie a LAPB, numită procedură multilegătură. Fiecare legătură este controlată de un protocol LAPB. Deasupra lor operează MLP care le tratează ca un grup de legături disponibile pentru transmisiune (figura 3.30).

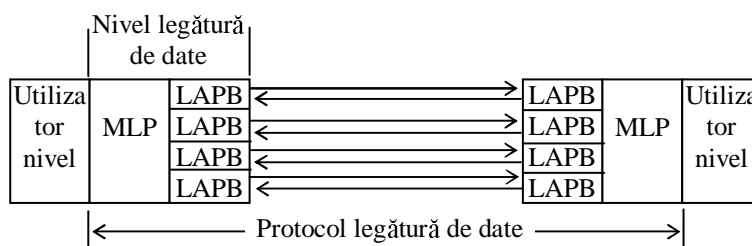


Fig. 3.30 Procedura multilegătură

Utilizatorul folosește aceeași interfață cu nivelul legătură indiferent dacă se utilizează numai LAPB sau și MLP. MLP adaugă, la fiecare bloc pe care-l primește pentru transmisie, un câmp de control (numit control multilegătură) prin care se realizează protocolul multilegătură. Blocul astfel format de MLP este trecut protocolului LAPB de pe o anumită legătură de date și acesta îl va trata ca și cum ar fi blocul de informație primit de la utilizator. Mecanismele de control al fluxului și de control al erorii utilizate în MLP sunt asemănătoare cu cele din LAPB.

c) LAPM (Link access procedure for modems - Procedură de acces al legăturii pentru modemuri). Este un protocol folosit opțional în modemuri pentru controlul erorii.

d) LAPD (Link access procedure D-channel). Este o variantă HDLC utilizată în rețelele digitale cu servicii integrate (ISDN) pentru a controla fluxul cadrelor de informație asociate canalului de semnalizare, numit canal D. De asemenea, cu modificări, este utilizată în rețelele releu de cadre (Frame relay) și în interfața radio dintre utilizator și rețeaua de comunicații mobile GSM (Global System for Mobile).

c) LLC (Logical link control - Controlul legăturii logice). Este o variantă a HDLC utilizată în rețelele locale.