

4. PODATKOVNA RAZINA

4.1 UVOD

Podatkovna razina obavlja poslove potrebne za uspješni prijenos okvira iz jednog čvora u drugi na način da oblikuje okvir i preda ga fizičkoj razini za prijenos. Jedinicu informacije (PDU) nazivamo okvirom. Oblik okvira ovisi o izabranom protokolu podatkovnog sloja, koji provodi sinkronizaciju na razini okteta. Protokol može biti znakovno orijentirani (character oriented) ili bitovno orijentirani (bit oriented). Podatkovnu razinu dijelimo u dvije podrazine: 2.1 i 2.2, slika 4.1.



Slika 4.1. Podjela podatkovne na dvije razine: 2.1 i 2.2

Funkcije podrazine 2.1:

- * sastavljanje i rastavljanje jedinica informacije PDU (Protocol Data Unit);
- * sinkronizacija PDU po oktetu;
- * detekcija pogriješke i odbacivanje neispravnog okvira.

Funkcije podrazine 2.2:

- * kontrola toka zbog usklađivanja brzine prijenosa između prijemnika i predajnika;
- * kontrola pogriješki u smislu retransmisije, ovisno da li je protokol spojevni ili bespojni
 - Kod spojevnih protokola na osnovu numeracije PDU (po modulu) otkrijemo gubitak i tražimo retransmisiju.
 - Kod bespojnih protokola gubitak PDU ne izaziva nikakvu reakciju, za konzistentnost podataka brine se neka od nadređenih razina.
- * multipleksiranje se provodi ako je po mrežnom priključku potrebno prenositi podatke više protokola mrežne razine. Unutar jedne mreže možemo koristiti više različitih protokola, pa je potrebno definirati pristupne točke (SAP, Service Access Point) da bismo ih mogli identificirati.

4.2 KODOVI ZA DETEKCIJU POGRIJEŠKI

Zaštita od pogriješki je potrebna da bi informacija stigla na odredište u izvornom obliku. Pogriješku možemo detektirati, a nakon toga korigirati na prijemnoj strani ili tražiti retransmisiju izgubljene informacije, ovisno o primjeni. Npr. kod govornih i video komunikacija zbog vremenskih ograničenja je nemoguće retransmitirati izgubljeni PDU, ali se eventualno može obaviti korekcija na prijemnoj strani. U digitalnom prijenosu podataka najčešće nakon detekcije obavljamo retransmisiju izgubljenih PDU, pa je važno koristiti što sigurnije kodove za detekciju pogriješke.

Pogriješke mogu biti izazvane EM spregom između vodova, razlikom potencijala između masa, smetnjama zbog blizine energetskih uređaja, smetnjama izazvanim starenjem komunikacijskih uređaja, promjenama napona napajanja i slično. Te su pogriješke slučajnog karaktera, pa ih često identificiramo sa šumom. Takve pogriješke otkrivamo kodovima za detekciju pogriješki.

U modernim mrežama, dominantan mehanizam gubitaka PDU je zagušenje, kada zbog potpunosti konačnih memorija čvorišta PDU bude izgubljen. Bez obzira na uzrok gubitka, nadoknadimo ga retransmisijom izgubljenog PDU.

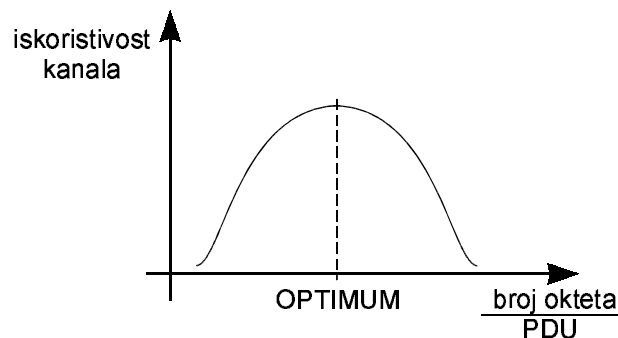
Kao mjerilo kvalitete prijenosa koristi se *prosječna vjerojatnost grješke BER* (Bit Error Rate), pogriješki po bitu. Možemo smatrati da je vjerojatnost pogriješke od 10^{-6} prihvatljiva, 10^{-7} dobra, a 10^{-5} i sve ispod toga je loše.

Tehnike za otkrivanje pogreški zasnovane su na unošenju redundancije (zalihosti) u kod. Koder ugrađujemo na predajnoj strani kanala sa zadatkom da izvorni (koncentrirani) kod pretvori u redundantni. Dekoder ugrađujemo na prijemnoj strani komunikacijskog kanala, a zadatak mu je da provjeri ispravnost primljene kodne riječi. Neispravnu kodnu riječ odbacuje, a ispravnu prevodi natrag na izvorni kod. U sustavima za prijenos podataka zaštita od pogreški provodi se nad PDU kao cjelinom. Kodna riječ u smislu kodiranja i dekodiranja je čitav PDU, ovisno o razini komunikacijskog sustav. Na fizičkoj to može biti znak, na podatkovnoj okvir, na mrežnoj paket itd.

Korištenjem kodnih riječi s povećanim brojem bita dobivamo veliki broj neiskorištenih kodnih riječi. Prijem neiskorištene kodne riječi znak je da je nastupila pogreška, dok prijem ispravne kodne riječi samo s nekom vjerojatnošću garantira da je primljena originalna kodna riječ. Naime, one pogreške koje ispravnu kodnu riječ pretvore u neku drugu ispravnu riječ, nije moguće otkriti. Treba pronaći kod u kojem unesena redundancija ima visok stupanj učinkovitosti. Kriteriji učinkovitosti su:

- efikasnost otkrivanja pogreški, ovisi o minimalnoj distanci među kodnim rječima
- efikasnost algoritma, ovisi o potrebnim operacijama odnosno sklopovlju za kodiranje i dekodiranje
- efikasnost protokola ovisi o odnosu korisne informacije prema veličini zaglavlja.

Kod dugačkih poruka je odnos SDU prema PDU približno 1, ali je vjerojatnost pogreške veća. S druge strane, poruka ne smije biti ni prekratka, jer će odnos PDU i SDU biti malen. Dakle, postoji optimalna veličina PDU za koju je iskoristivost kanala maksimalna, slika 4.2.

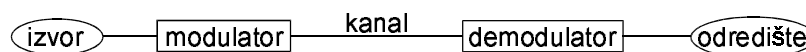


Slika 4.2. Iskoristivost kanala u ovisnosti o veličini jedinice informacije

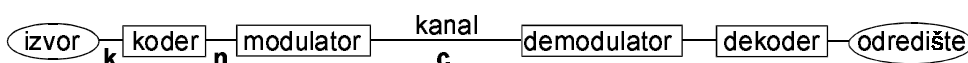
Pogreške se obično pojavljuju u snopovima. Kako smetnja traje određeno vrijeme, usnopljavanje je veće što je veća brzina prijenosa. Dobra strana usnopljavanja je što sa istom kvalitetom kanala i istim brojem pogrešnih bitova dobijemo manji broj pogrešnih blokova (okvira). Loše je to što možemo očekivati učestale pogreške unutar jednog PDU, a to nam otežava otkrivanje pogreški.

4.2.1 VRSTE KODOVA ZA DETEKCIJU POGREŠKI

Poruku možemo prenositi bez (slika 4.3) ili sa zaštitom od pogreški (slika 4.4).



Slika 4.3. Prijenos poruke bez zaštite od pogreški



Slika 4.4. Prijenos poruke sa zaštitom od pogreški

4.2.1.1 Sistematski i nesistematski kodovi

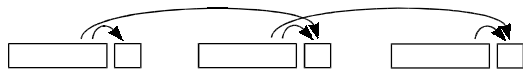
Kod može biti sistematski (originalnoj poruci se dodaju redundantni bitovi) i nesistematski (neki općeniti kod). Za primjenu su interesantniji sistematski kodovi, kod kojih se prenosi originalna informacija od k bita, kojoj se dodaje c redundantnih bita. Kodna riječ redundantnog koda koja se prenosi preko kanala ima ukupno $n=k+c$ bita. Iskoristivost takvog koda je:

$$\eta = k / n.$$

4.2.1.2 Konvolucijski i blok kodovi

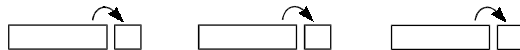
Kodove, osim na sistematske i nesistematske, dijelimo na konvolucijske i blok kodove.

Konvolucijski kodovi se najčešće koriste kod malih kodnih riječi. Nastaju tako da se svakom bloku dodaju redundantni bitovi na osnovu tog i prethodnog bloka, slika 4.5. Prednost im je to što lakše detektiraju pogrešku, a mana što se pogreška multiplicira i propagira na slijedeće blokove.



Slika 4.5. Konvolucijski kodovi

Redundantni bitovi se kod **blok kodova** dodaju na osnovi vlastite informacije, slika 4.6.



Slika 4.6. Blok kodovi

4.2.1.3 Kodovi s paritetnim ispitivanjem

Kod kodova s paritetnim ispitivanjem, bitovima originalne kompleksije se dodaju paritetni kontrolni bitovi, tako da za definirano paritetno ispitivanje broj jedinica bude paran ili neparan. Za paritetno ispitivanje koristimo operaciju sume po modulu $m=2$. Zbrajanje svih znamenki neke kodne riječi po modulu 2 daje jedinicu, ako je broj jedinica u riječi bio neparan, a nulu ako je bio paran. Suma po modulu se koristi zbog svojih svojstava asocijativnosti, komutativnosti i zatvorenosti, odakle slijedi mogućnost rješavanja sustava jednačbi slično linearnim jednačbama i primjene matričnog računa.

4.2.2 SISTEMATSKI BLOK KODOVI S PARITETNIM ISPITIVANJEM

Sistematske blok kodove s paritetnim ispitivanjem možemo opisati tablično ili matrično. Zbog veličine tablica češće se koristi opis matricom (n,k) , gdje je n ukupan broj bita, a k broj bita originalne informacije, s tim da vrijedi da je $k \leq n$. Slučaj $k = n$ opisuje prijenos informacije bez zaštite.

4.2.2.1 Kodiranje koda s paritetnim ispitivanjem

Kodiranjem 2^k k-torki originalne informacije dobije se 2^n n-torki, od čega je 2^k ispravnih, a $(2^n - 2^k)$ neispravnih (neiskorištenih).

Kodna kompleksija \underline{c} nastaje tako da na originalnu kompleksiju \underline{d} djelujemo generator matricom \underline{G} :

$$\underline{c} = \underline{d} * \underline{G}$$

\underline{c} je n-torka iz skupa 2^k ispravnih n-torki, \underline{d} k-torka iz originalnog skupa, a \underline{G} matrica veličine $k \times n$.

Matrica \underline{G} se sastoji od jedinične matrice veličine $k \times k$ i permutacijske matrice \underline{P} veličine $k \times (n-k)$, koja definira c kontrolnih bita.

$$\underline{G} = \left| \begin{array}{c|c} I & P \end{array} \right|$$

Primjer za blok kod (7,4):

$$\underline{G} = \left| I_k \quad P_{k \times (n-k)} \right| = \left| I_4 \quad P_{4 \times 3} \right| = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{vmatrix}$$

Ako definiramo vektor \underline{c} :

$$\underline{c}_7 = \underline{d} * \underline{G} = \left| 1 \quad 0 \quad 1 \quad 0 \right| * \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{vmatrix} = \left| 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \right|$$

$$\begin{aligned} c_1 &= d_1 & c_5 &= d_2 \oplus d_3 \\ c_2 &= d_2 & c_6 &= d_1 \oplus d_3 \oplus d_4 \\ c_3 &= d_3 & c_7 &= d_2 \oplus d_4 \\ c_4 &= d_4 \end{aligned}$$

Raspored bita u matrici \underline{P} nam određuje koji će dio kompleksije \underline{d} biti zaštićen. Zato \underline{P} ne smije biti proizvodljna matrica, već mora zadovoljiti neke uvjete:

1. Ne smije imati dva ista stupca, jer se u tom slučaju rezultirajućoj matrici \underline{c} dodaju dva ista kontrolna bita koja ne doprinose sigurnosti.
2. Ni jedan redak ne smije sadržavati same nule, jer tada određeni bit originalne kompleksije neće ući u paritetno ispitivanje.
3. Ni jedan stupac ne smije sadržavati same nule, jer će tada određeni kontrolni bit biti jednak nuli.
4. Ne smije imati dva jednaka retka, jer bi se pojavom pogreški na odgovarajućim mjestima doprinos tih bita poništio, pa bismo mogli detektirati samo jednostruku pogrešku.

Svojstvo je kodne riječi \underline{c} da nastaje sumom redaka matrice \underline{G} .

4.2.2.2 Dekodiranje koda s paritetnim ispitivanjem

Na prijemnoj je strani potrebno provjeriti ispravnost primljenog koda. U tu svrhu definiramo matricu kontrole pariteta \underline{H} i njenu transponiranu matricu \underline{H}^T :

$$\underline{H} = \left| P^T \quad I \right| \quad ; \quad \underline{H}^T = \begin{vmatrix} P \\ I \end{vmatrix}$$

Vrijedi:

$$\underline{G} * \underline{H}^T = 0$$

Primjer za blok kod (7,4):

$$\underline{H} = \begin{vmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{vmatrix} \quad ; \quad \underline{H}^T = \begin{vmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$\underline{G} * \underline{H}^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} * \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = 0$$

Ova jednadžba znači da redak matrice \underline{G} pomnožen sa bilo kojim stupcem matrice \underline{H}^T daje nulu. Kako je stupac \underline{H}^T jednak retku \underline{H} , vrijedi:

$$\text{redak}(\underline{G}) \times \text{redak}(\underline{H}) = 0$$

Matrica \underline{H}^T primijenjena na \underline{c} mora dati nulu ako je \underline{c} iz skupa ispravnih kompleksija:

$$\underline{d} * \underline{G} = \underline{c}$$

$$\underline{d} * \underline{G} * \underline{H}^T = \underline{c} * \underline{H}^T = 0$$

Ako \underline{c} nije ispravna kompleksija, umnožak će biti $\neq 0$, čime smo detektirali pogrješku.

$$\forall \underline{c} * \underline{H}^T = 0 \Rightarrow \underline{c} \in \mathbf{C}$$

$$\forall \underline{c}' * \underline{H}^T \neq 0 \Rightarrow \underline{c}' \notin \mathbf{C}$$

4.2.2.3 Sindrom koda s paritetnim ispitivanjem

Za sindrom s vrijedi:

$$\underline{c}' * \underline{H}^T = \underline{s}$$

Primljena kodna riječ \underline{c}' može biti pogrješna:

$$\underline{c}' = \underline{c} \oplus \underline{e}$$

gdje je \underline{c} poslana kodna riječ, a \underline{e} vektor pogrješke. Jedinica na mjestu određene težine u vektoru \underline{e} znači da je bit iste težine u primljenoj kodnoj riječi pogrješan, a nula znači da je identičan bitu te težine u poslanoj kodnoj riječi.

Vrijedi:

$$\underline{c}' * \underline{H}^T = (\underline{c} \oplus \underline{e}) * \underline{H}^T = \underline{c} * \underline{H}^T \oplus \underline{e} * \underline{H}^T = 0 \oplus \underline{s} = \underline{s}$$

Iz toga vidimo da sindrom ovisi o karakteru pogrješke:

$$\underline{s} = \underline{e} * \underline{H}^T$$

Ograničenja sindroma:

1. Sistematski blok kod sa paritetnim ispitivanjem ne može otkriti onu pogrješku koja je sama za sebe ispravna kodna riječ.

$$\underline{e} * \underline{H}^T = \underline{s} = 0, \text{ ako je } \underline{e} \in \mathbf{C}$$

Zato se ponekad koriste konvolucijski kodovi.

2. Postoji određeni postotak pogrješki koje ne možemo detektirati:

$$\frac{\text{ukupan_broj_ispravnih_kompleksija}}{\text{ukupan_broj_kompleksija}} = \frac{2^k}{2^n} = \frac{1}{2^{n-k}} \cdot 100[\%]$$

Dakle, $\frac{1}{2^{n-k}} \cdot 100[\%]$ pogrješki ne možemo detektirati.

3. Ako je sindrom duljine $n-k$, onda u skupu sindroma \mathbf{S} imamo 2^{n-k} različitih sindroma, među kojima je i $s_0=0$:

$$s_0, s_1, \dots, s_{2^{n-k}-1} \in \mathbf{S}.$$

Po jednom sindromu će biti raspoređeno $\frac{2^n}{2^{n-k}} = 2^k$ različitih pogriješki, što znači da nakon detekcije pogriješke ne možemo sa sigurnošću izvršiti korekciju. Odavde slijedi da je za prijenos podataka daleko pogodnije koristiti samo detekciju pogriješki, te nakon toga retransmisiju (ponovno slanje) izgubljenih podataka.

4.2.2.4 Statistička svojstva pogriješki

Pokazalo se da sve pogriješke nisu jednako vjerojatne. Višestruke pogriješke su relativno rijetke, a i snopovi relativno velike dužine su malo vjerojatni.

Ukoliko smo ispravno dizajnirali matricu \underline{G} , u 2^k pogriješki će se naći jedna koja je vjerojatnija od svih drugih i koju nazivamo *dominantna kodna riječ pogriješke*. Nastojimo tako konstruirati kod da pogriješke koje su najvjerojatnije i koje imaju relativno malu distancu svakako budu grješke koje možemo detektirati. Ako one pripadaju raznim sindromima, možemo ih i korigirati.

4.2.3 SISTEMATSKI CIKLIČKI KODOVI

Mana je općenitih kodova s paritetnim ispitivanjem što za praktične duljine poruka od nekoliko stotina ili tisuća bita prilikom kodiranja i dekodiranja moramo rješavati vrlo velike sustave jednadžbi. Stoga je interesantno naći takve kodove, kod kojih se kodiranje i dekodiranje može obaviti jednostavnim algoritmom. Takvi su **ciklički kodovi**.

Ciklički kodovi su nastali iz blok kodova operacijom cikličke permutacije nad matricom \underline{G} . Ciklička permutacija se radi tako da svaki bit pomaknemo u lijevo, a najznačajniji bit dolazi na najmanje značajno mjesto. Kodiranje i dekodiranje ovih kodova mogu biti elegantnije riješeni, jer se generator matrica i matrica pariteta ne trebaju predstavljati tabelarno, nego mogu biti generirane preko prikladnog algoritma.

4.2.3.1 Kodiranje cikličkih kodova

Vektor \underline{c} možemo zapisati kao polinom:

$$c(x) = c_{n-1}x^{n-1} \oplus c_{n-2}x^{n-2} \oplus \dots \oplus c_1x^1 \oplus c_0x^0$$

x nam pokazuje mjesto na kojem se unutar kodne riječi nalazi neki od koeficijenata c_{n-k} .

Ako ovakav polinom pomnožimo sa x , dobit ćemo pomak koeficijenata u lijevo. Dobiveni polinom tada podijelimo sa $x^n \oplus 1$, čime smo napravili ciklički pomak.

Primjer:

$$1001 \rightarrow 0011 \rightarrow 0110 \rightarrow 1100 \rightarrow 1001$$

$$1001 \rightarrow 0011: (x^3 \oplus 1) * x = x^4 \oplus x$$

$$(x^4 \oplus x) : (x^4 \oplus 1) = 1$$

$$\underline{x^4 \oplus 1}$$

$$x \oplus 1 \rightarrow 0011$$

$$0011 \rightarrow 0110: (x \oplus 1) * x = x^2 \oplus x$$

$$(x^2 \oplus x) : (x^4 \oplus 1) = 0$$

$$\underline{0}$$

$$x^2 \oplus x \rightarrow 0110$$

itd.

$c(x)$ izračunavamo jednostavno, dijeljenjem pomaknutog $d(x)$ sa $g(x)$:

$$\frac{x^{n-k} * d(x)}{g(x)} = q(x) \oplus \frac{r(x)}{g(x)} \Big| * g(x)$$

$$x^{n-k} * d(x) = q(x) * g(x) \oplus r(x) \Big| \oplus r(x)$$

$$x^{n-k} * d(x) \oplus r(x) = q(x) * g(x) = c(x) \Rightarrow k(x) = r(x)$$

Dakle, $c(x)$ dobijemo pomakom $d(x)$ ulijevo i dodavanjem ostatka dijeljenja sa $g(x)$, slika 4.8.



Slika 4.8. Dodavanje kontrolnih bitova originalnoj kompleksiji.

$$c(x) = x^{n-k} d(x) \oplus r(x)$$

$$r(x) = \text{rem} \frac{x^{n-k} d(x)}{g(x)}$$

Na prijemnoj strani dobijemo kodnu riječ $c'(x)$ i provjeravamo njenu djeljivost sa $g(x)$:

$$c'(x) = e(x) \oplus c(x)$$

$$\frac{c'(x)}{g(x)} = \frac{e(x)}{g(x)} \oplus \frac{c(x)}{g(x)}$$

Za $c(x)$ znamo da je djeljivo sa $g(x)$, pa rezultat dijeljenja ovisi o $e(x)$. Može se pokazati da je ostatak dijeljenja $\frac{e(x)}{g(x)}$ jednak sindromu napisanom u obliku polinoma.

$$\text{rem} \frac{e(x)}{g(x)} = s(x)$$

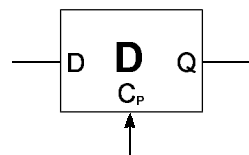
Ako je ostatak dijeljenja $s(x)=0$, vektor \underline{e} je ili iz skupa ispravnih kompleksija ili je $=0$.

Iz svega ovoga možemo vidjeti svojstva matrice \underline{P} s obzirom na detekciju i korekciju pogriješke:

1. Ako želimo detektirati jednostruku pogriješku, svi retci moraju biti $\neq 0$, jer ako je i jedan redak $=0$, pogriješka se neće moći detektirati.
Ako želimo korigirati jednostruku pogriješku, svi retci moraju biti različiti, kako bismo za svaku pogriješku dobili jedinstven rezultat, odnosno jedinstven sindrom.
2. Ako želimo detektirati dvostruku pogriješku, sume dvaju redaka ne smiju biti $=0$.
Ako želimo korigirati dvostruku pogriješku, sume dvaju redaka ne smiju biti iste.
3. Za trostruku i ostale višestruke pogriješke vrijedi slično kao i za dvostruku.

4.2.3.3 Sklopovlje za kodiranje i dekodiranje cikličkog koda.

Kašnjenje u digitalnoj tehnici možemo realizirati korištenjem D-bistabila, slika 4.9.



Slika 4.9. D-bistabil kao element za kašnjenje

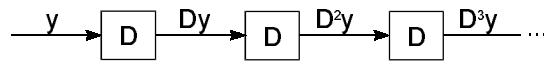
Za D-bistabil vrijedi:

$$Q^{n+1} = D^n$$

Ako je Q izlazna varijabla z , a D ulazna varijabla y , imamo da je:

$$z=D*y$$

s tim da D u ovoj formuli znači operator kašnjenja.



Slika 4.10. Serijski spoj D-bistabila

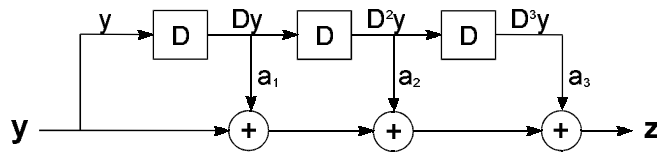
Ako imamo više elemenata u seriji (slika 4.10), dobit ćemo sekvencu:

$$y=y_0+y_1D+y_2D^2+y_3D^3+\dots$$

Ova sekvencu kao niz bita odgovara polinomu:

$$y(x)=x^{n-k-1}y_0+x^{n-k-2}y_1$$

Dalje možemo izvršiti kombinaciju trenutnih vrijednosti na izlazu sa prethodnim vrijednostima, slika 4.11.



Slika 4.11. Linearna kombinacija različitih kašnjenja ulaza

Za slučaj kada je $a_1=0$, $a_2=1$ i $a_3=1$ imamo:

$$z = y \oplus D^3 y \oplus D^4 y = y(1 \oplus D^3 \oplus D^4)$$

$$\frac{z}{y} = G^*(D) = 1 \oplus D^3 \oplus D^4$$

Ovaj sklop se naziva binarni filter. Može se opisati funkcijom $G^*(D)$, koja ima karakter prijenosne funkcije. U slučaju da postoji povratna veza (slika 4.12), imamo:

$$z = P'(D)y \oplus G'(D)z$$

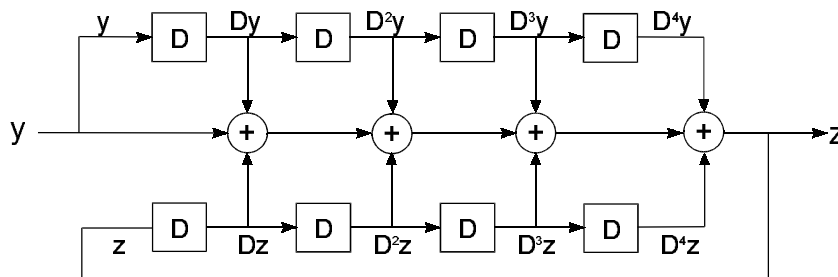
$$z(1 \oplus G'(D)) = P'(D)y$$

$$\frac{z}{y} = \frac{P(D)}{1 \oplus G(D)}$$

Prijenosnu funkciju za ovaj slučaj možemo napisati i ovako:

$$\frac{z}{y} = \frac{P(D)}{G(D)} = G^*(D)$$

Za $G'(D)=0$ imamo slučaj bez povratne veze, odnosno $G^*(D)=P(D)$.



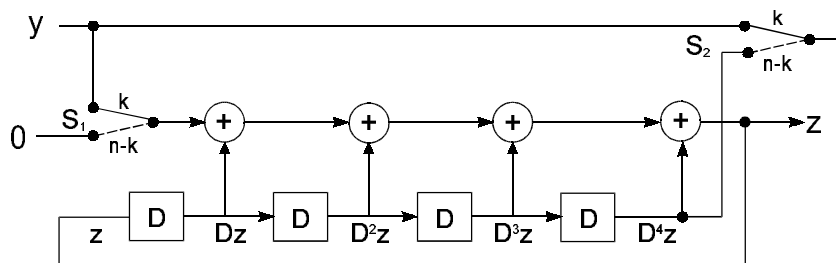
Slika 4.12. Linearna kombinacija zakašnjenog ulaza i izlaza

Poseban slučaj je za $P=1$:

$$G^*(D) = \frac{1}{G(D)} ; z = \frac{y}{G(D)}$$

odnosno, izlazna sekvenca nastaje dijeljenjem ulaza sa polinomom. Nas zapravo zanima ostatak tog dijeljenja.

Sklop za kodiranje i dekodiranje cikličkog koda prikazan je na slici 4.13.



Slika 4.13. Koder i dekoder cikličkog koda

Na predajnoj strani bistabili se postave u početnu vrijednost 0 i originalna poruka $d(x)$ se propušta kroz sklop nepromijenjena. Istovremeno se vrši dijeljenje te poruke sa generator polinomom $g(x)$, a u bistabilima sklopa se akumulira ostatak dijeljenja. Nakon emitiranja poruke $d(x)$, preklopnici se prebace u drugi položaj ($n-k$) i na kanal se pošalje ostatak dijeljenja $r(x)$.

Na prijemnoj strani se kroz sklop propušta cijela poruka $c(x)$. Ukoliko nije bilo pogriješke, ostatak dijeljenja u bistabilima sklopa mora biti jednak nuli.

Ovakvo korištenje cikličkih kodova ne osigurava detekciju pogriješki nastalih umetanjem ili ispuštanjem nule na početku poruke (zbog gubitka sinkronizacije). Zato se po CCITT V.42 bistabili sklopa početno postavljaju u 1. Rezultat dijeljenja na prijemnoj strani jednak je konstanti (ostatku dijeljenja početnih jedinica sa generator polinomom).

4.2.4 PRAKTIČNA PRIMJENA KODOVA ZA DETEKCIJU POGRIJEŠKI

U praksi se najčešće koriste tri vrste kodova za detekciju pogriješki:

1. vertikalna zaštita VRC;
2. dužinska zaštita LRC;
3. ciklička zaštita CRC.

4.2.4.1 Zaštitno kodiranje pomoću vertikalne i dužinske metode.

Vertikalna zaštita VRC (Vertical Redundancy Check) se provodi dodavanjem paritetnog bita svakom znaku. Paritet može biti paran ili neparan, odnosno znak može imati paran ili neparan broj jedinica.

Primjena VRC omogućuje otkrivanje jedne pogriješke. Da bi se povećala mogućnost detekcije i omogućila korekcija pogriješke, primjenjuje se dužinska zaštitna metoda LRC (Longitudinal Redundancy Check), u okviru koje se određuje paran ili neparan broj jedinica u nizu bitova istog brojnog mjesta.

Istodobna zaštita sa VRC i LRC omogućuje otkrivanje dviju pogriješki ili otkrivanje i ispravljanje jedne. Ako postoji višestruka pogriješka, na ovaj je način nije moguće ispraviti.

4.2.4.2 Polinomska ili ciklička zaštita kodova.

Kod suvremenih protokola primjenjujemo cikličku zaštitu CRC (Cyclic Redundancy Check).

Imamo više različitih cikličkih kodova ovisno o korištenom generirajućem polinomu:

- * CRC-12 je kod za IBM-ova stara računala. Polinom za ovaj kod je:

$$x^{12} \oplus x^3 \oplus x^2 \oplus x \oplus 1 = (x \oplus 1)(x^{11} \oplus x^2 \oplus 1).$$

* CRC-16 za IBM: $x^{16} \oplus x^{15} \oplus x^2 \oplus 1 = (x \oplus 1)(x^{15} \oplus x \oplus 1).$

* CRC-CCITT se koristi kod svih bitovno-orijentiranih protokola koji imaju 16-bitnu zaštitu:

$$x^{16} \oplus x^{12} \oplus x^5 \oplus 1 = (x \oplus 1)(x^{15} \oplus x^{14} \oplus x^{13} \oplus x^{12} \oplus x^4 \oplus x^3 \oplus x^2 \oplus x \oplus 1).$$

Za dužinu poruke od 1000 bita CRC-CCITT otkriva 99.999% grješaka.

* CRC-32 se danas često upotrebljava kod lokalnih mreža. Polinom mu je:

$$x^{32} \oplus x^{26} \oplus x^{23} \oplus x^{22} \oplus x^{16} \oplus x^{12} \oplus x^{11} \oplus x^{10} \oplus x^8 \oplus x^7 \oplus x^5 \oplus x^4 \oplus x^2 \oplus x \oplus 1.$$

4.2.4.3 Korištenje CRC-CCITT prema V.42, X.25 i ISO 3309.

Ako dođe do gubitka sinkronizacije, može biti preskočen ili dodan bit 0 na početku poruke. CRC kodovi ne mogu detektirati tu grješku, te koristimo 16-bitni zaštitni kod kao komplement sume po modulu članova

a) $rem \frac{x^k (x^{15} \oplus x^{14} \oplus \dots \oplus x \oplus 1)}{g(x)}$

b) $rem \frac{x^{n-k} d(x)}{g(x)}$

gdje je $g(x)$ polinom prema CRC-CCITT $g(x) = x^{16} \oplus x^{12} \oplus x^5 \oplus 1$, a k je broj bita okvira između zadnjeg bita okvirnog znaka i prvog bita zaštitnog koda, ne uključujući granične bitove.

Kod predajnika se početni sadržaj registra uređaja koji izračunava ostatak dijeljenja sastoji od samih jedinica, a zatim se modificira dijeljenjem na gore navedeni način; ostatak tog dijeljenja se prenosi kao 16-bitni zaštitni kod. U prijemniku se na početku sadržaj registra također postavlja u same jedinice, a nakon množenja sa x^{16} i dijeljenja sa generator-polinomom, zaštitni kod je "0001 1101 0000 1111" (ako nema pogreški u prijenosu).

32-bitni zaštitni kod je komplement sume po modulu članova:

a) $rem \frac{x^k (x^{31} \oplus x^{30} \oplus \dots \oplus x \oplus 1)}{g(x)}$

b) $rem \frac{x^{n-k} d(x)}{g(x)}$

$g(x) = x^{32} \oplus x^{26} \oplus x^{23} \oplus x^{22} \oplus x^{16} \oplus x^{12} \oplus x^{11} \oplus x^{10} \oplus x^8 \oplus x^7 \oplus x^5 \oplus x^4 \oplus x^2 \oplus x \oplus 1$, a za k vrijedi isto što i gore. Također vrijedi sve što je rečeno za prijemnik i predajnik, jedino što je rezultirajući 32-bitni zaštitni kod "1100 0111 0000 0100 1101 1101 0111 1011".

4.3 PROTOKOLI PODATKOVNOG SLOJA

Tokom razvoja pojavile su se tri grupe protokola podatkovnog sloja:

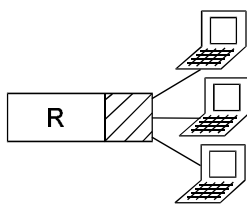
1. znakovno-orijentirani protokoli;
2. bitovno-orijentirani protokoli;
3. protokoli na lokalnim mrežama (također bitovno orjentirani).

4.3.1 ZNAKOVNO ORIJENTIRANI PROTOKOLI

Znakovno orijentirani protokoli polaze od pretpostavke da je na fizičkoj razini obavljena sinkronizacija po znaku. Osnovna jedinica informacije kod znakovno-orijentiranih protokola je blok sastavljen od pojedinog znakova. Karakteristika znakovno-orijentiranih protokola je da se prijenos odvija znak po znak i da svaki znak mora imati određeno značenje. Mogu funkcionirati i na sinkronim i na

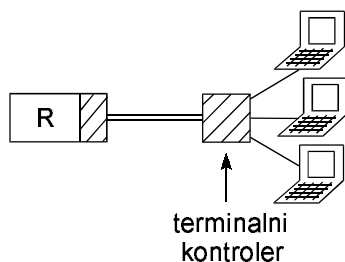
asinkronim kanalima. Loše je to što nisu transparentni, jer su neki znakovi rezervirani za upravljanje prijenosom, pa se ne smiju pojaviti u korisnikovoj poruci.

Znakovno-orijentirani protokoli su se prvi put pojavili na mrežama terminala. Imali smo dvije izvedbe takvih mreža. U prvoj izvedbi je inteligentan uređaj, na kojeg su se priključivali znakovni terminali, bio na računalu, slika 4.14.



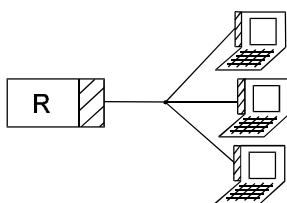
Slika 4.14. Znakovni terminali, priključeni direktno na računalo

U drugoj izvedbi je inteligentni uređaj, kontroler terminala, odvojen od računala serijskim kanalom, na kome se koristi znakovno-orijentirani protokol, slika 4.15.



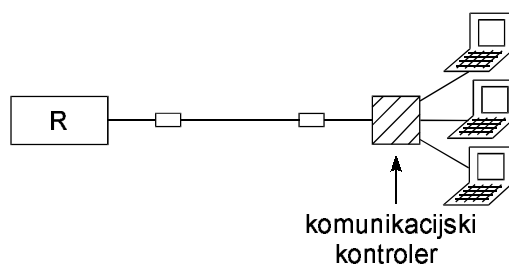
Slika 4.15. Znakovni terminali, priključeni na kontroler

U trećoj izvedbi su na kanal, na kome se koristi znakovno-orijentirani protokol sa funkcijom prozivanja i selektiranja, priključili blok-orijentirane terminale, slika 4.16.



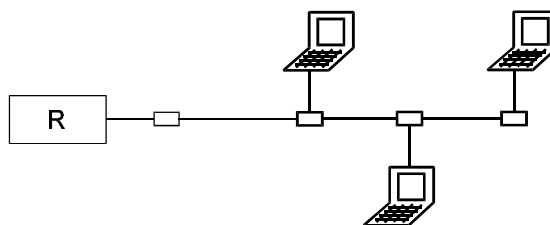
Slika 4.16. Način priključivanja blok-orijentiranih terminala

Za priključak terminala na udaljene lokacije koristimo modeme. Ovdje također imamo dvije mogućnosti. Prva koristi vezu od točke do točke (jednospojno povezivanje), slika 4.17.



Slika 4.17. Priključak terminala na udaljene lokacije korištenjem jednospojnog povezivanja

Druga mogućnost je višespojno povezivanje korištenjem specijalnih (multidrop) modema, slika 4.18.



Slika 4.18. Priključak terminala na udaljene lokacije korištenjem višespojnog povezivanja

Pri kodiranju znakova primjenjuju se dva koda:

- a) ASCII / CCITT N₀5 / ISO 646, koji je danas u upotrebi.
- b) IBM-ov EBCDIC, koji se danas sve manje koristi;

4.3.1.1 Faze komuniciranja prema ISO 1745

ISO 1745 je protokol napravljen dijelom na temelju IBM-ovog BSC znakovno-orijentiranog protokola, definiranog za sinkroni prijenos. Interesantan nam je zbog definiranja faza u kojima se odvija komunikacija. Od 5 navedenih, faze 1. i 5. su izvan ovog standarda.

- | | |
|--|--|
| <ul style="list-style-type: none"> 1. uspostava fizičkog kanala: <ul style="list-style-type: none"> * biranje (telefonskog) broja; * prospajanje; * sinkronizacija modema. 2. uspostava logičkog kanala: <ul style="list-style-type: none"> * prozivanje; * selektiranje. | <ul style="list-style-type: none"> 3. prijenos podataka. 4. raskid logičkog kanala. 5. raskid fizičkog kanala: <ul style="list-style-type: none"> * isključenje vala-nosioca; * raskid telefonskog kanala. |
|--|--|

Ove faze ne moraju slijediti jedna iza druge. Npr. ako je prozivanje i selektiranje u fazi uspostave logičkog kanala bilo neuspješno, ponavljamo ga sve dok ne uspijemo, nakon čega možemo prijeći u fazu prijena podataka. Nakon raskida logičkog kanala ne moramo nužno raskinuti i fizički, već možemo ponovo obaviti prozivanje i selektiranje.

Prozivanje i selektiranje obavlja primarna stanica ili *master*. Stanica koja se odaziva kada je primarna stanica prozove ili selektira naziva se sekundarna stanica ili *slave*. Obje stanice se tokom svog rada mogu naći u nekom određenom stanju. Primarna će stanica u trenutku kada vrši prozivanje ili selektiranje biti u kontrolnom stanju, a sekundarna, ako nije prozvana ni selektirana, u neutralnom stanju. Kad sekundarna stanica prepozna poruku prozivanja, prelazi u stanje prijema, a primarna stanica u stanje predaje. Ako primarna stanica pošalje poruku selektiranja, a sekundarna ima spremne podatke, sekundarna će prijeći u stanje predaje, a primarna u stanje prijema.

4.3.1.2 Vrste okvira

Prema ISO 646 definirano je više različitih vrsta znakova za prijenos znakovno-orijentiranih poruka. Oni se ne smiju pojaviti u korisnikovoj poruci, zbog čega kod ove vrste protokola imamo problem transparentnosti.

Nama su posebno interesantni kontrolni znakovi:

- | | |
|---|--|
| SOH (01) - početak zaglavlja (Start of Header); | ACK (06) - potvrda (Acknowledgement); |
| STX (02) - početak teksta (Start of Text); | DLE (10) - iznimka (Data Link Escape); |
| ETX (03) - kraj teksta (End of Text); | NAK (15) - negativna potvrda (Negative Ack.); |
| EOT (04) - kraj prijena (End of Transmission); | SYN (16) - sinkronizacija (Synchronous idle); |
| ENQ (05) - upit (Enquiry); | ETB (17) - kraj bloka (End of Transmission Block). |

Korištenjem ovih znakova definiraju se određene vrste blokova (okvira), koje stanice međusobno izmjenjuju radi upravljanja kanalom i prijena podataka. Kod sinkronog prijena, okviri započinju s jednim ili dva sinkronizacijska znaka SYN.

Informacijski blokovi

Informacijski blokovi (okviri) koriste se prvenstveno u fazi prijena podataka i mogu biti:

- a) Blokovi bez zaglavlja, koriste se kod najjednostavnijih protokola na jednospojnom mediju:

$$\begin{array}{ccc} S & E(B) & S & E(B) \\ T \text{ tekst} & T(C) & T \text{ tekst} & T(C) \\ X & X(C) & X & X(C) \end{array}$$

Ako je sadržaj poruke dulji nego što protokol dopušta, poruka se rastavlja na potreban broj blokova. Svaki se blok prenosi zasebno, ali predviđenim redosljedom. Iza znaka ETX ili ETB dolazi polje zaštitnih bitova BCC. Polje BCC može biti duljine jednog znaka ili više njih. Način pripreme zaštitnih bitova i njihove provjere ovisi o protokolu.

- b) Blokovi sa zaglavljem. Zaglavlje je dio poruke koji sadrži adresu odredišta, pozitivnu ili negativnu potvrdu prijena, prijenosni put, prednost, redni broj i druge podatke potrebne za odvijanje prijena. Njegov sastav određen je protokolom. Korisnik nema uvid u zaglavlje.

$$\begin{array}{ccc} S & S & E(B) & S & S & E(B) \\ O \text{ zaglavlje} & T \text{ tekst} & T(C) & O \text{ zaglavlje} & T \dots T & T(C) \\ H & X & X(C) & H & X & X(C) \end{array}$$

Svaki blok započinje znakom SOH i ima potrebno zaglavlje, u kojem se može nalaziti podatak o rednom broju bloka i ukupnom broju blokova koji sačinjavaju poruku. Ako je samo zaglavlje toliko veliko da ga trebamo ubaciti u blok, oblik okvira je:

$$\begin{array}{ccc} S & E(B) & S & E(B) \\ O \text{ zaglavlje} & T(C) & T \dots T & T(C) \\ H & B(C) & X & X(C) \end{array}$$

Kontrolni blokovi

- a) Primarna stanica šalje okvire prozivanja i selektiranja, koji imaju općeniti oblik.

$$\begin{array}{c} z \\ n \\ \text{prefix} \\ a \\ k \end{array}$$

Za prozivanje prefiks zamijenimo adresom prozivanja, a za selektiranje adresom selektiranja. Primarna stanica može poslati poruku selektiranja u prvom informacijskom bloku, kako bi izbjegla čekanje na odziv.

$$\begin{array}{ccc} E & E & S & S & E \\ \text{adresa_prozivanja} & N & \text{adresa_selektiranja} & N & \text{adresa_selektiranja} & O \dots T \dots T \\ Q & Q & H & X & X \end{array}$$

- b) Nakon što je primarna stanica izvršila prozivanje, sekundarna stanica, ukoliko nema spremnih podataka, ostaje u neutralnom stanju i šalje negativni odgovor EOT. Ako ima spremne podatke, prelazi u predajno stanje i odgovara sa jednim ili nizom informacijskih okvira:

$$\begin{array}{ccc} E & S & S & E \\ \text{prefix} & O & O \dots T \dots T & \\ T & H & X & X \end{array}$$

Nakon primljene poruke selektiranja, sekundarna stanica šalje NAK ukoliko se nije spremna odazvati, ili prelazi u prijemno stanje i šalje ACK:

<i>N</i>	<i>A</i>
<i>prefix A</i>	<i>prefix C</i>
<i>K</i>	<i>K</i>

- c) Ako je u fazi prijenosa podataka uspješno prihvaćen blok, prijemna stanica šalje ACK, a ako blok nije prihvaćen, šalje NAK:

<i>A</i>	<i>N</i>
<i>prefix C</i>	<i>prefix A</i>
<i>K</i>	<i>K</i>

Kad predajna stanica pošalje blok podataka, očekuje od prijemne pozitivnu ili negativnu potvrdu. Kada prijemna stanica pošalje pozitivnu potvrdu ACK, predajna nastavlja sa slanjem podataka. Ako prijemna stanica primi pogriješan blok, šalje negativnu potvrdu NAK, te predajna ponavlja prijenos prethodno odaslane poruke. Za slučaj da odgovor prijemne stanice izostane, predajna nakon nekog vremena čekanja šalje upit i tek nakon primitka odgovora reagira na odgovarajući način.

U praksi se može dogoditi da kompletna poruka bude izgubljena. Tada prijemna stanica ne odgovara na poslani blok, jer ga nije ni primila. Kada predajna stanica pošalje upit, prijemna joj odgovara potvrdno, jer smatra da se upit odnosi na prethodno primljeni blok podataka (predzadnji koji je poslan), te predajna stanica šalje slijedeći blok. Rezultat toga je gubitak jednog bloka. Da bi se to izbjeglo, uvodi se mehanizam alternativne potvrde:

<i>S S</i>	<i>A S S</i>	<i>A</i>
<i>O O T ... (parna poruka) ;</i>	<i>0 C O 1 T ... (neparna poruka) ;</i>	<i>1 C</i>
<i>H X</i>	<i>K H X</i>	<i>K</i>

To je bio prvi pokušaj numeracije okvira, i to s prozorom 1.

Da bi se riješio problem transparentnosti, koristi se posebni znak DLE, koji mijenja značenje znaka iza njega. Npr. sa DLE STX se postiže da STX više ne znači početak teksta, već je to sada korisnikov podatak. Slično, sa DLE DLE se postiže da DLE bude prenesen kao korisnikov podatak:

<i>D S</i>	<i>D D</i>
<i>L T</i>	<i>L L</i>
<i>E X</i>	<i>E E</i>

Alternativno, koriste se znakovi SO (Shift Out), pređi u transparentni mod, te Si (Shift In), vrati se u kontrolni mod.

- d) Nakon obavljenog prijenosa podataka, predajna stanica može raskinuti vezu šaljući EOT, odnosno vezu i fizički kanal šaljući DLE EOT:

<i>E</i>	<i>D E</i>
<i>prefix O</i>	<i>prefix L O</i>
<i>T</i>	<i>E T</i>

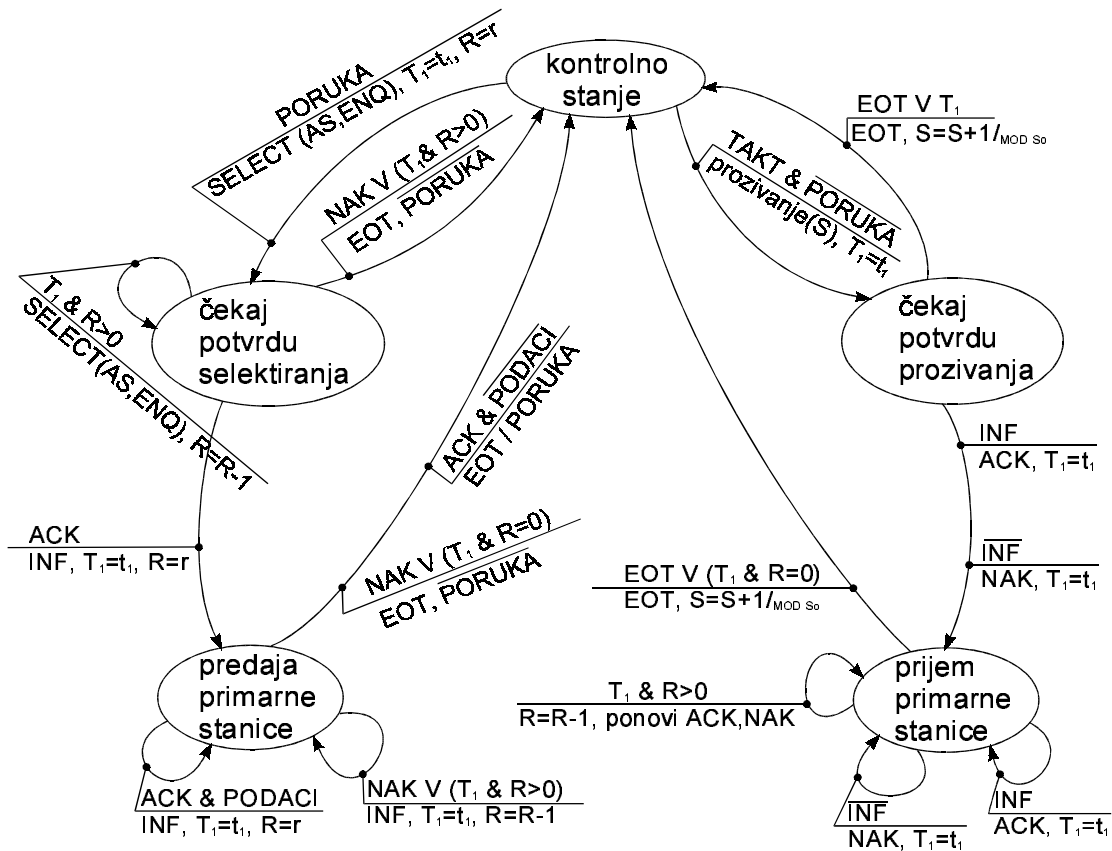
4.3.1.3 Dijagrami stanja za primarnu i sekundarnu stanicu.

U dijagramima stanja znakovno-orijentiranog protokola za primarnu (slika 4.19) i sekundarnu stanicu (slika 4.20) korišteni su ovi znakovi:

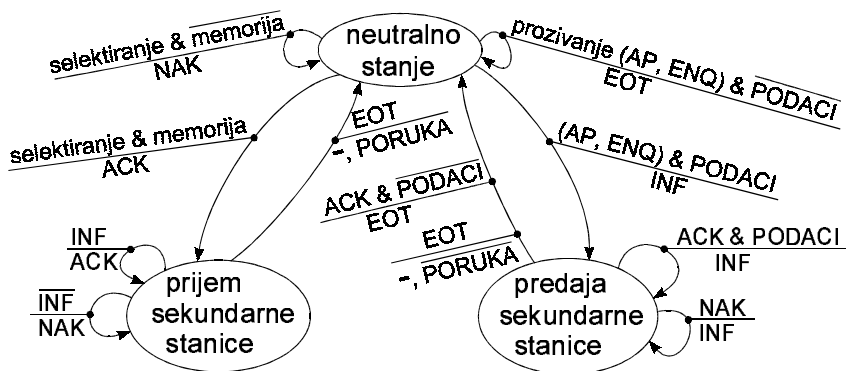
- * T_1 - vremenski sklop, postavlja vrijeme t_1 ($T_1=t_1$);
- * R - brojač retransmisija;
- * r - max. broj retransmisija;
- * INF - informacijski blok;

- * SELECT (AS; ENQ) - selektiranje;
- * TAKT - takt prozivanja;
- * PORUKA - poruka poslana.

Ako neki od navedenih znakova ima crtu iznad, značenje mu je suprotno (negirano).



Slika 4.19. Dijagram stanja primarne stanice



Slika 4.20. Dijagram stanja sekundarne stanice

4.3.2 ZNAKOVNO ORIJENTIRANI PROTOKOLI U PRAKSI

Unatoč znatnim nedostacima, znakovno orijentirani protokoli se masovno koriste zbog sposobnosti korištenja asinkronih kanala. Sama činjenica da osobna računala standardno raspolažu asinkronim veznim sklopom, ograničava prijenos podataka telefonskim kanalom na znakovno orijentirane protokole. Zbog kompatibilnosti, čak i interni modemi prividno s terminalom komuniciraju asinkrono. Pri tome nema značenja što se komunikacija između para modema odvija sinkrono, bitovno orijentiranim protokolom, jer je ta aktivnost nevidljiva (transparentna) za korisnika.

4.3.2.1 Samoodredni protokoli

Samoodredni protokoli su također znakovno-orijentirani protokoli. Kao i znakovno-orijentirani, i samoodredni protokoli se koriste za asinkroni i za sinkroni prijenos (sinkronizacijski znak SYN). Jedan od njih je DDCMP (Digital Data Communications Message Protocol) protokol u DNA arhitekturi. Oblik okvira samoodrednog protokola je:

$$\begin{pmatrix} S & S \\ Y & Y \\ N & N \end{pmatrix} \begin{matrix} S & L & B & B \\ O \text{ zaglavlje} & E & C & \text{tekst} & C \\ H & N & C & C \end{matrix}$$

Zaglavlje samoodrednog protokola razlikuje se od zaglavlja običnog znakovno-orijentiranog po tome što sadrži i podatak o duljini poruke LEN, zbog čega je nepotreban znak za kraj teksta. Time je riješen problem transparentnosti. Međutim, protokol je postao osjetljiviji na pogreške zaglavlja, pa je uvedene posebno kodiranje za detekciju pogreški zaglavlja.

Razvoj samoodrednih protokola bio je korak naprijed ka razvoju bitovno orijentiranih protokola.

4.3.2.2 SLIP protokol Interneta

SLIP (Serial Line Internet Protocol) je de-fakto standard za modemske prijenos IP (Internet Protocol) paketa preko telefonskih kanala. On nikada nije prihvaćen kao službeni standard. Specifikacija definira samo oblik okvira u kojem je sadržan IP paket, s mogućnošću kompresije IP zaglavlja. SLIP ne sadrži mehanizme adresiranja, identifikacije vrste okvira, te detekcije i korekcije pogreški.

SLIP koristi 4 posebna znaka, <END> (hex C0, decimalno 196), <ESC> (hex DB, decimalno 219), različito od ASCII "ESC" znaka, <hexDC> (dekadski 220) i <hexDD> (dekadski 221).

Originalna SLIP specifikacija predviđa jednostavno slanje niza znakova koji završava <END> znakom. Taj znak ima značenje okvirnog znaka. Ukoliko postoji šum, moguće je da će neki znakovi biti primljeni prije početka emitiranja paketa, pa je naknadno specificirano slanje <END> znaka na početku paketa. Time se znakovi nagomilani zbog šuma odbacuju (na mrežnoj razini, IP protokol), a slijedi projenos znakova IP paketa.

Ukoliko se <END> znak nalazi unutar korisnikove informacije (IP paket), šalje se dvoznačna sekvenca <ESC> <hex DC>. Slično, ukoliko korisnikovi podaci sadrže <ESC> znak, šalje se sekvenca <ESC><hex DD>. Time je osigurana transparentnost protokola. Oblik okvira prikazan je na slici 4.21.



Slika 4.21. Format SLIP okvira

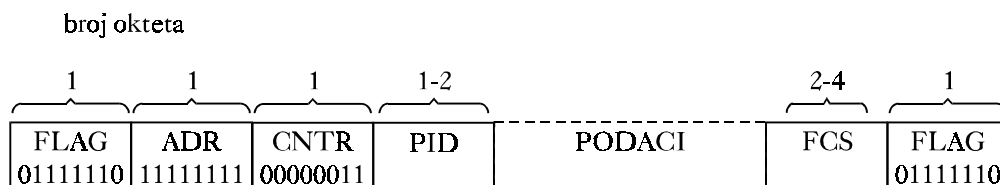
Maksimalna dužina SLIP paketa nije definirana i standardno se koristi dužina od najmanje 1006 okteta. Prednost SLIP-a je u jednostavnosti, a mane u nedostatku mehanizama adresiranja (jednospojno povezivanje), nedostatku identifikacije mrežnog protokola (samo jedan protokol mrežne razine po fizičkom kanalu, najčešće IP) i nedostatku kompresije podataka (moguće korištenje modema sa MNP5 ili V42bis kompresijom). Naknadno je uvedena kompresija SLIP i TCP/IP zaglavlja, naročito važna kod prijenosa kratkih paketa (Telnet).

4.3.2.3 PPP protokol Interneta

PPP (Point to Point Protocol) je službeni protokol Interneta za modemske i druge serijske kanale. Specificira okvir kojim je moguće prenositi pakete raznih mrežnih protokola po istom kanalu i

mehanizme upravljanja protokolima podatkovnog (LCP, Link Control Protocol) i mrežnog (NCP, Network Control Protocol) nivoa. Omogućava prijenos asinkronim i sinkronim kanalima.

Specifikacijom je predviđena uporaba okvira sličnog HDLC bitovno orijentiranom protokolu. Za asinkrone kanale, transparentnost se postiže korištenjem <ESC> znaka (hex 7D). Okvirni znak šalje se na početku i na kraju okvira, nakon izračunavanja zaštitnog polja po polinomu CRC-CCITT ili CRC-32. Adresno i kontrolno polje su fiksni, ali postoje kako bi se mogli koristiti sinkroni vezni sklopovi. PI polje identificira protokol mrežne razine, čiji je paket smješten u polju "podaci". Oblik okvira prikazan je na slici 4.22.



Slika 4.22. Format PPP okvira

PPP okvir se standardno komprimira izostavljanjem adresnog i kontrolnog polja, te svodenjem PI polja na 1 oktet. Maksimalna duljina podatkovnog polja se dogovara i standardno iznosi 1500 okteta.

U fazi prijenosa podataka očito se radi o bespojnom protokolu. LCP se odnosi na dogovaranje opcija (duljina okvira, kompresija zaglavlja), funkciju kontrole kvalitete i funkciju provjere identiteta korisnika (lozinke). NCP se definira za pripadni mrežni protokol i koristi se za određivanje dinamički dodjeljivanih mrežnih adresa (npr. kod pristupa korisnika komutiranim kanalom), te za uključenje kompresije TCP/IP zaglavlja.

4.3.2.4 Protokoli za prijenos datoteka modemom

Prije pojave Interneta modemi su se najčešće koristili za povezivanje dvaju računala radi razmjene datoteka i poruka preko sustava za prijenos poruka BBS (Bulletin Board System). U tu su se svrhu koristili različiti komunikacijski programi koji su omogućavali terminalski pristup drugom računalu koristeći neki od terminalskih protokola (ANSI, TTY, VT52, VT102), te razmjenu datoteka koristeći neki od znakovno orijentiranih protokola (XMODEM, YMODEM, ZMODEM). Komunikacijski programi uključivali su i poseban komandni (script) jezik, kojim se automatizirao pristup udaljenom sustavu, upravljanje komunikacijskim veznim sklopovima i upravljanje slanjem i primanjem datoteka.

Unatoč velikoj popularnosti Interneta BBS sustavi još uvijek predstavljaju dobar izvor informacija. Prednost im je što je njihovo korištenje potpuno besplatno. BBS je kombinacija oglasne ploče, konferencijske dvorane i poštanskog sandučića. Na njima možemo čitati oglase, sudjelovati u raspravama o raznim temama, razmjenjivati poštu te primati i slati datoteke. Najvažnija usluga je elektronička pošta. Na BBS sustavima se obično mogu naći velike kolekcije programa i tekstova, raspodijeljenih u odgovarajuće sekcije. BBS program prihvaća pozive, vrši identifikaciju korisnika, omogućuje čitanje i pisanje poruka, pretraživanje datoteka, pokreće procese za primanje i slanje datoteka. Većina BBS-ova je povezana u svjetsku mrežu FidoNet. U FidoNet-u veza među BBS-ovima nije stalna već se računala obično povezuju jednom dnevno kako bi razmijenila poštu, korisničke podatke i datoteke.

XMODEM protokol

XMODEM je vrlo jednostavan i učinkovit protokol koji je ugrađen u svim komunikacijskim programima. Predviđen je za asinkroni prijenos datoteka uz 8 podatkovnih bita, jednim stop bitom i bez paritetnog bita. Za upravljanje vezom koriste se ASCII kontrolni znakovi:

- SOH - 01hex (Start of Header)
- EOT - 04 hex (End of Transmission)
- ACK - 06 hex (Acknowledge)
- NAK - 15 hex (Negative Acknowledge)
- CAN - 18 hex (Cancel)

Podaci se prenose u blokovima veličine 128 okteta koji su numerirani i zaštićeni zaštitnim znakom, slika 4.23:

<SOH><broj bloka><255-broj bloka><128 okteta podataka><checksum>

Slika 4.23. Format XMODEM okvira

- <SOH> - kontrolni znak
- <broj bloka> - oktet koji se koristi za numeraciju blokova (numeracija ide od 01 do FF hex)
- <255-broj bloka> - komplement broja bloka
- <checksum> - zaštita podataka, 1 oktet. Za checksum vrijedi da u slučaju da nije bilo greške u prijenosu suma zaštićenih okteta i checksum okteta mora biti nula.

Prijenos je obosmjernan, a započinje tako da prijemna stanica šalje NAK kontrolni znak. Nakon toga predajna stanica šalje blok podataka i ovisno o ispravnosti primljenog bloka prijemna stanica šalje ACK kontrolni znak, što znači da može primiti sljedeći blok, ili NAK kontrolni znak, što znači da je potrebna retransmisija bloka. U slučaju nastanka pogreške na ACK kontrolnom znaku pa ga predajna stanica ne primi ispravno, predaja se prekida, a nakon 10 sekundi prijemna stanica šalje NAK i prethodni blok se šalje ponovo. Predajna i prijemna stanica moraju uvijek biti sinkronizirane, što znači da prijemna stanica može primiti samo očekivani blok (u slučaju da nije bilo pogrešaka u prijenosu), ili opet isti blok (u slučaju pogreške na primljenom bloku ili na ACK kontrolnom znaku). Primanje bilo kojeg drugog bloka znači da došlo do gubitka sinkronizacije.

Kada predajna stanica nema više podataka, ona šalje EOT, a kada primi ACK veza se prekida.

Kao mane XMODEM protokola pokazale su se:

- nedovoljna veličina bloka koja onemogućuje veće brzine prijenosa
- slaba zaštita od pogreški koju pruža zaštitna suma
- nezaštićeni ACK i NAK kontrolni znakovi koji mogu uzrokovati nepotrebne retransmisije i gubitak sinkronizacije
- mogućnost prenošenja samo jedne datoteke u jednom prijenosu
- ne prenosi se ime datoteke, pa ga je potrebno utipkati i na predajnoj i na prijemnoj strani

Neki su od navedenih problema riješeni proširenjima XMODEM protokola.

XMODEM/CRC za zaštitu od pogreški umjesto zaštitne sume koristi 16 bitnu CRC zaštitu.

XMODEM -1k je XMODEM/CRC s blokovima veličine 1024 okteta.

YMODEM protokol

YMODEM je proširenje XMODEM/CRC protokola koje koristi poseban blok rednog broja nula za slanje imena, veličine i datuma posljednje izmjene datoteke. YMODEM omogućuje i slanje većeg broja datoteka u jednom prijenosu (batch prijenos), te omogućuje korištenje blokova veličine 128 i 1024 okteta u istom prijenosu.

Povećanje veličine bloka na 1024 okteta je povećalo brzinu prijenosa, ali je ona još uvijek kod obosmjernog prijenosa ograničena time što predajna stanica mora čekati na ACK za prethodni blok prije slanja sljedećeg bloka (širina prozora 1). Ovo je riješeno YMODEM-G proširenjem protokola kod kojeg se primljeni blokovi uopće ne potvrđuju ACK znakovima. Ovo je značilo odricanje od retransmisije u slučaju grešaka. Kada prijemna stanica primi pogrešan okvir prijenos se prekida.

ZMODEM protokol

Sva spomenuta proširenja XMODEM i YMODEM protokola ugrađena su u ZMODEM protokol. Za razliku od XMODEM blokova koji se numeriraju od 1 do 255 i koji su točno određene duljine, ZMODEM prenosi podatke kao okvire sa zaglavljem i proizvoljnim brojem okteta podataka, a umjesto numeracije koristi pomak od početka datoteke. Zaglavlje je duljine 5 okteta:

tip okvira F3 F2 F1 F0

ili:

tip okvira P0 P1 P2 P3

a nakon zaglavlja slijede paketi podataka od kojih svaki može biti duljine od 0 do 1024 okteta, tako da se u jednom okviru može poslati cijela datoteka. Zaglavlje i paketi podataka se štite 16 ili 32 bitnom CRC zaštitom. Neki od tipova okvira dani su u tablici 4.1:

<i>tip okvira</i>	<i>funkcija okvira</i>
ZRQINIT, ZRINIT	uspostava veze
ZFILE	podaci o datoteci
ZDATA	prijenos datoteke
ZRPOS	indikacija pozicije u datoteci od koje će se vršiti prijenos
ZFIN	prekid veze

Tablica 4.1. Tipovi okvira ZMODEM protokola

F0, F1, F2 i F3, te P0, P1, P2 i P3 su po 4 okteta koji predstavljaju zastavice (flags), odnosno pomak od početka datoteke. Zastavice se koriste samo kod nekih tipova okvira, a služe za određivanje nekih dodatnih opcija, npr. kod ZFILE okvira zastavice određuju pod kojim će se uvjetima obaviti prijenos datoteke ako na prijemnom računaru već postoji datoteka s istim imenom, da li će se primijeniti sažimanje ili šifriranje datoteke, da li će se obaviti konverzija znakova za kraj linije.

Okteti P0, P1, P2 i P3 određuju koji dio datoteke je primljen odnosno koji se dio treba retransmitirati, a ovo omogućuje slijedno (streaming) način prijenosa, tj. slanje niza okvira bez čekanja na potvrdu nakon svakog okvira da bi se poslao sljedeći okvir. Korištenje pozicije u datoteci umjesto numeracije blokova omogućuje da se nakon prekida i ponovne uspostave veze slanjem ZRPOS okvira prijenos nastavi od točke u kojoj je stao, bez potrebe za ponovnim prijenosom već primljenog dijela datoteke.

4.3.3 BITOVNO ORIJENTIRANI PROTOKOLI

Kod znakovno-orijentiranih protokola svaka se poruka sastojala od znakova, od kojih su neki imali posebno značenje i nisu se smjeli pojavljivati unutar korisnikove poruke. Upotrijebljeni protokoli zahtijevali su neposrednu potvrdu poruke nakon prijenosa, a nova se poruka nije odašiljala prije primitka potvrde. To je bilo vrlo sporo i neprikladno, jer se svaki put trebalo okretati smjer komuniciranja, te čekati vrijeme kašnjenja na kanalu.

Zbog problema sporosti i transparentnosti razvili su se novi protokoli i oblici okvira, kod kojih se unutar teksta smiju prenositi bilo kakve kombinacije korisnikovih bita. Prvi takav protokol bio je IBM-ov SDLC (Synchronous Data link Control), koji je standardiziran kao ANSI protokol ADCCP (ANSI Data Communications Control Protocol), ISO 3309/4335 protokol HDLC (High-speed Data link Control), te protokol prema ITU-T preporuci X.25 LAP-B (Link Access Protocol Balanced). Nadalje, bitovno orijentirani protokoli se masovno koriste kod lokalnih mreža kao MAC (Media Access Protocol) i LLC (Logical Link Control), te kod sinkronih modemske komunikacije među inteligentnim modemima (transparentno za korisnika) pod nazivom LAP-M (Link Access Protocol for Modem).

Jedina je mana bitovno-orijentiranih protokola da se isključivo koriste na sinkronim kanalima.

Podatkovni sloj se dijeli u dva dijela: 2.1 i 2.2. Oba imaju definirane osnovne oblike okvira prema ISO standardu, i to 2.1 prema ISO 3309, a 2.2 prema ISO 4335.

4.3.3.1 Razina 2.1 prema ISO 3309

Na slici 4.24. prikazan je osnovni oblik okvira:



Slika 4.24. Okvir bitovno-orijentiranog protokola na razini 2.1

- * **F - okvirni znak**, flag. Ima oblik 01111110 i označava kraj jednog i početak drugog okvira. Između dva okvira može biti jedan ili više okvirnih znakova. Ako kanal nije opterećen, stanice su dužne slati više uzastopnih okvirnih znakova, kako bi se održala aktivnost kanala i sinkronizacija po znaku i okviru. Transparentnost korisnikove poruke se postiže ubacivanjem nule nakon svakih pet jedinica. Prijemnik poslije pet primljenih jedinica bezuslovno odbacuje nulu.
- * **A - adresno polje**. Sadrži adresu na razini podatkovne razine. Njegova duljina može iznositi jedan ili više okteta i određena je pravilima upotrijebljenog protokola. Poruku s adresnim poljem popunjenim jedinicama primaju sve stanice, odnosno to je univerzalna adresa. Poruku s adresnim poljem popunjenim nulama neće preuzeti nijedna stanica. Ovdje nedostaje adresa pošiljaoca (što je naknadno uvedeno za LLC). Preporuka ITU-T X.25 ne predviđa višeočetno adresno polje, pa je kod višespojnog povezivanja broj stanica na istom mediju ograničen.
- * **C - upravljačko (kontrolno) polje**. Ima osnovnu duljinu od 8 bita. Sadrži parametre koji određuju vrstu okvira, te parametre vezane za numeraciju blokova i retransmisiju. ITU-T X.25 preporuka i kasniji standardi predviđaju i upravljačko polje duljine 16 bita.
- * **I - informacijsko polje**. Sadrži blokove korisnikove informacije. Kod nekih je protokola maksimalna duljina propisana brojem bita, a kod drugih brojem okteta.
- * **FCS - zaštitno polje**. Koristi se CRC-CCITT zaštita (16-bitna ciklička zaštita), koja djeluje na cijeli okvir (A, C, I). Kod lokalnih mreža koristi se CRC-32.

Prijenos okvira se obavlja se isključivo sinkronim kanalima. Na prijemnoj strani je potrebno sastaviti okvir, za što postoje dva načina. Jedan je način da prijemnik nakon detekcije okvirnog znaka odmah provjerava adresu odredišta. Ako je adresa prepoznata kao vlastita, nastavlja se sa kompletiranjem okvira, a ako nije, kompletiranje okvira se zaustavlja i čeka se sljedeći kontrolni znak. Drugi način je da prijemnik kompletira okvir, a tek onda provjerava adresu. Ovakav način je neprikladan i nepotrebno opterećuje uređaj, jer nema potrebe kompletirati okvire koji nisu namijenjeni tom uređaju. Nakon prepoznavanja adrese, prijemnik nastavlja sa kompletiranjem okvira sve dok ne nađe na kontrolni znak. Tada provjerava je li primljeni okvir ispravan. Ako jest, cijeli se okvir prosljeđuje razini 2.2. Ako je detektirana pogreška, okvir se odbacuje. Višoj razini se ne dojavljuje ništa, jer nije poznato da li je adresa ispravna i da li je kompletiran blok namijenjen tom uređaju.

4.3.3.2 Razina 2.2 prema ISO 4335

Razina 2.2 očekuje da joj stižu ispravni okviri sa razine 2.1, te očekuje da će razina 2.1 poslati njene okvire. Originalna specifikacija po ISO 4335 definira postojanje primarne i sekundarne stanice. Primarna stanica šalje komandne okvire (command) prema sekundarnim stanicama i od njih prima odgovore u obliku odzivnih okvira (response). Sekundarna stanica može održavati komunikaciju samo s jednom primarnom stanicom.

Opisane stanice mogu raditi u načina prijenosa podataka:

1. normalni odzivni mod NRM (Normal Response Mode), namijenjen obosmjernim (half duplex) vezama;
2. asinkroni odzivni mod ARM (Asynchronous Response Mode), namijenjen dvosmjernim (full duplex) vezama.

U NRM načinu rada sekundarna stanica može pokrenuti prijenos samo na temelju izričite dozvole primarne stanice, a kod ARM se prijenos može pokrenuti i bez te dozvole. NRM je pogodan za prozivne sustave, gdje se sve akcije odvijaju prema točnom redosljedu, a ARM za sustav jedne primarne i jedne sekundarne stanice, koje žele razmjenjivati informacije bez čekanja i postupka prozivanja.

Oblik upravljačkog polja

C-polje je oblikovano tako da ima tri različite vrste (grupe) okvira:

- * I - informacijski okvir (information) - služi za prijenos informacija. Prvi bit C-polja ovog okvira je 0;

- * S - nadzorni okvir (supervision) - služi za prijenos nadzornih informacija podatkovnog sloja (potvrda I okvira, zahtjev za ponovni prijenos I okvira, privremeni prekid prijena I okvira). Početni bitovi C-polja su 10;
- * U - nenumerirani okvir (unnumbered) - služi za dodatne upravljačke naloge podatkovnog sloja. Početni bitovi C-polja su 11.

Upravljačko polje može biti 8-bitno, slika 4.25 ili prošireno (16-bitno), slika 4.27. 8-bitno C-polje sadrži slijedeće podatke:

- * N(S) (Send Sequence Number) - redni broj odaslanog okvira;
- * N(R) (Receive Sequence Number) - redni broj okvira kojeg predajna stanica očekuje primiti;
- * S (Supervisory function bit) - bitovi nadzorne funkcije. Kako postoje 2 S-bita, imamo $2^2=4$ različitih S okvira;
- * M (Modifier function bit) - bitovi modifikatorske funkcije. Ima ih 5, pa postoji $2^5=32$ različitih U okvira.
- * P/F (Poll/Final) - prozivni bit.

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

Slika 4.25. 8-bitni oblik upravljačkog C polja

Kod S i I okvira ista vrsta okvira može biti komanda ili odziv, ovisno da li ga šalje primarna ili sekundarna stanica (C/R - Command/Response). U okvir može biti ili samo komanda ili samo odziv, tj. jedne U okvire šalje samo primarna, a druge samo sekundarna stanica.

Svaka stanica na vezi održava dvije varijable: V(S) (Send State Variable) i V(R) (Receive State Variable). V(S) označava redni broj okvira koji će slijedeći biti poslan, a V(R) broj informacijskog okvira za kojeg očekujemo da će slijedeći biti primljen.

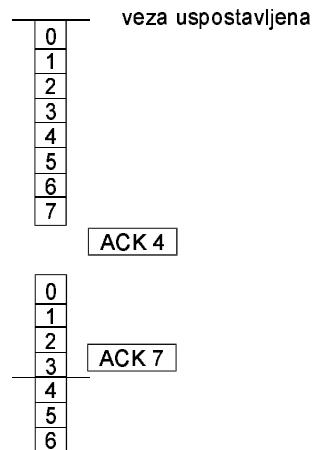
Kada neka stanica šalje informacijski okvir, tada u polje N(S) kopira sadržaj svoje varijable V(S), te nakon toga poveća za 1 (increment) V(S) po modulu 2^n . Na taj način je u zaglavlju (C-polju) određen redni broj pod kojim se taj okvir šalje. Ista stanica kopira sadržaj V(R) u N(R) u trenutku kada šalje informacijski ili nadzorni okvir. Time javlja koji okvir očekuje iz suprotnog smjera i potvrđuje prijem svih prethodno primljenih okvira.

Kada stanica prima nadzorni ili informacijski okvir od korespondentne stanice, analizira njeno polje N(R), čime provjerava koji je od do tada poslanih okvira potvrđen, odnosno da li je taj broj okvira ispravan. Ukoliko se radi o informacijskim okvirima, provjerava N(S) i uspoređuje sa V(R). Ako je $V(R)=N(S)$, primili smo očekivani okvir, pa V(R) povećamo za 1. Ako je $V(R) \neq N(S)$, došlo je do dupliciranja nekog okvira, ili do njegova gubitka. Duplicirani se okvir odbacuje. Prijemna stanica može odbaciti prekoredni paket i tražiti neselektivnu retransmisiju svih okvira od izgubljenog dalje (Go Back N), ili prekoredni zadržati i tražiti selektivna retransmisiju onog okvira koji nedostaje (Selective Retransmission).

Blokove korisnikovih informacija numeriramo po modulu $m=2^n$, gdje je n broj bitova podatka u C-polju, a m širina prozora. Numeraciju blokova po modulu koristimo istovremeno za kontrolu toka i kontrolu pogreški.

Kontrolom pogreški na razini 2.2 otkriva se gubitak blokova na razini 2.1 na osnovu njihovih rednih brojeva, te se traži njihova selektivna ili neselektivna retransmisija. Detekcija gubitka na osnovu izostanka okvira (odnosno prijemom prekorednog okvira) moguća je na podatkovnoj razini zbog toga, jer na fizičkom mediju, jednospojnom ili višespojnom, ne može doći do promjene redosljeda okvira. Pretpostavlja se, naravno, da predajnik šalje okvire redom.

Numeracija okvira po modulu može se koristiti i za kontrolu toka. Predajnik ne smije poslati više okvira nego što je širina prozora (kod 8-bitnog C-polja širina prozora je 8, a kod 16-bitnog 128). Mehanizam prozora prikazan je na slici 4.26.



Slika 4.26. Mehanizam prozora

Predajnik smije poslati onoliko okvira kolika je širina prozora, a prijemnik šalje potvrde prijema. Zbog kašnjenja se može dogoditi da predajnik pošalje npr. 8 okvira, a da do prijemnika stignu samo 4 (numerirana od 0 do 3). Prijemnik tada potvrđuje ta četiri okvira (ACK4), te predajnik šalje slijedeće okvire numerirane od 0 do 3. Ako u međuvremenu stigne potvrda za 7 okvira (ACK7), predajnik nastavlja sa slanjem slijedećih okvira, numeriranih od 4 do 6. Postupak se slično nastavlja dalje.

P/F je prozivni mehanizam, koji se koristi ovisno o vrsti rada. U C-polju komande nalazi se P (Poll) bit, a u C-polju odziva nalazi se F (Final) bit. Mehanizam P/F bita usko je povezan s primarnim vremenskim sklopom.

Kod NRM načina rada, primarna stanica šalje niz okvira i u zadnjem okviru pošalje P bit u jedinici (P=1), čime želi dozvoliti sekundarnoj stanici da šalje svoje okvire. Sekundarna stanica se na to dužna odazvati nizom okvira i u posljednjem okviru poslati F=1, čime daje znak primarnoj stanici da je spremna ponovo primiti njene okvire. na taj način se efikasno kontrolira promjena smjera komuniciranja na obosmjernom mediju.

Kod ARM načina rada, primarna stanica u svakom trenutku može sa P=1 zahtijevati odziv sekundarne stanice. Sekundarna stanica se na to mora odazvati sa F=1 što je prije moguće. Ako sekundarna stanica trenutno nema podataka, poslat će nadzorni S okvir. Okvir s F=1 smatra se direktnim odgovorom na okvir s P=1, te je lako ostvariti sinkronizaciju numeracije okvira.

Prošireni oblik upravljačkog polja je prikazan na slici 4.27.

	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	X	X	X	X	P/F	N(R)							
U	1	1	M	M	U	M	M	M	P/F	X	X	X	X	X	X	X	

Slika 4.27. Prošireni oblik upravljačkog polja

Bitovi označeni sa X i U nemaju za sada određeno značenje i postavljaju se u nulu. Kod nekih bitovno orijentiranih protokola, U okviri imaju uvijek 8-bitno C-polje.

Nadzorni S-okvir ima dva S-bita, što znači da imamo četiri različita nadzorna okvira:

- SS=00 ⇒ RR (Receiver Ready) - prijemnik spreman;
- SS=01 ⇒ RNR (Receiver Not Ready) - prijemnik nije spreman;
- SS=10 ⇒ REJ (Reject) - odbacivanje okvira (zahtjev za ponovnim slanjem);
- SS=11 ⇒ SREJ (Selective Reject) - selektivno odbacivanje okvira.

Komandom/odzivom RR naznačuje se spremnost stanice za prijem okvira, a istovremeno se potvrđuju svi dotada preneseni okviri (do N(R)-1). Komanda RR uz P=1 je upit o stanju druge stanice.

Komandom/odzivom RNR naznačuje se privremena nesposobnost stanice da primi okvir I, a istodobno se potvrđuju svi okviri sa rednim brojevima do N(R)-1. RNR se uz P=1 može upotrijebiti i za traženje podatka o stanju druge stanice.

Komandom/odzivom REJ traži se ponovni prijenos okvira s rednim brojem N(R) i svih okvira poslanih nakon njega, a istovremeno se potvrđuju okviri do N(R)-1. U jednom trenutku za dani smjer smije postojati samo jedan okvir S sa nalogom REJ.

Komandom/odzivom SREJ zahtijeva se ponovni prijenos samo jednog okvira, i to onog sa rednim brojem N(S)=N(R). Prijemom tog okvira prijenos se nastavlja dalje.

Kontrolni U-okvir ima 5 M-bitova, pa imamo 32 različita U okvira. Time se naznačuje niz mogućih komandi i odziva, od kojih ćemo navesti samo one koji su definirani sa ISO 4335:

- Komande koje šalje primarna stanica su slijedeće:
 - * 11 000 ⇒ SARM (Set ARM) - postavi asinkroni odzivni mod;
 - * 00 001 ⇒ SNRM (Set NRM) - postavi normalni odzivni mod;
 - * 00 010 ⇒ DISC (Disconnect) - komanda za raskid veze;
 - * 11 010 ⇒ SARME (Set ARM Extended) - postavi asinkroni odzivni mod s proširenim zaglavljem;
 - * 11 011 ⇒ SNRME (Set NRM Extended) - postavi normalni odzivni mod s proširenim zaglavljem.
- Sekundarna stanica šalje sljedeće odzive:
 - * 00 110 ⇒ UA (Unnumbered Acknowledgment) - nenumerirana potvrda;
 - * 10 001 ⇒ CMDR (Command Reject) - odbacivanje komande.

Informacijsko polje CMDR okvira je veličine 20 bita, slika 4.28.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
kopija C-polja odbačene komande									0	N(S)	0	N(R)	W	X	Y	Z			

Slika 4.28. Oblik informacijskog polja za CMDR

- * W=1 ako je primljeno C-polje nerazumljivo;
- * X=1 ako je primljeno neispravno C-polje, koje sadrži informacijsko polje kakvo nije dozvoljeno ili okvir S ili U nedozvoljene duljine (tada je i W=1);
- * Y=1 ako je informacijsko polje predugo;
- * Z=1 ako je N(R) neispravan.

Za prošireni oblik zaglavljaja informacijsko polje je veličine 36 bita.

Vremenski sklop

Vremenski sklop se koristi da bi se detektirao izostanak aktivnosti pojedine stanice. Kod NRM samo primarna stanica ima vremenski sklop, a kod ARM i sekundarna. Vremenski sklop primarne stanice označavamo s TP (Primary Timer), a vremenski sklop sekundarne stanice s TS (Secondary Timer).

Pravila za korištenje TP (vremenskog sklopa primarne stanice) kod NRM su:

- * START P=1 Vremenski sklop pokrećemo svaki put kad je P=1.
- * RESTART F=0 Sekundarna stanica se može odazvati sa nizom od više okvira, a TP nanovo pokrećemo svaki put kada primimo F=0.
- * STOP F=1 TP zaustavljamo kad je F=1.
- * istek pogriješka Ukoliko prođe vrijeme predviđeno vremenskim sklopom, detektirali smo pogriješku.

Pravila za korištenje TP kod ARM su:

- * START P=1 Pokretanje TP.

- * RESTART ne postoji
- * STOP F=1 Veza je dvosmjerna, sekundarna stanica šalje F=1, čime zaustavlja TP.
- * istek pogrješka Ako sekundarna stanica u određenom vremenu ne pošalje F=1, došlo je do pogrješke u prijenosu.

Pravila za korištenje TS (vremenskog sklopa sekundarne stanice) kod ARM su:

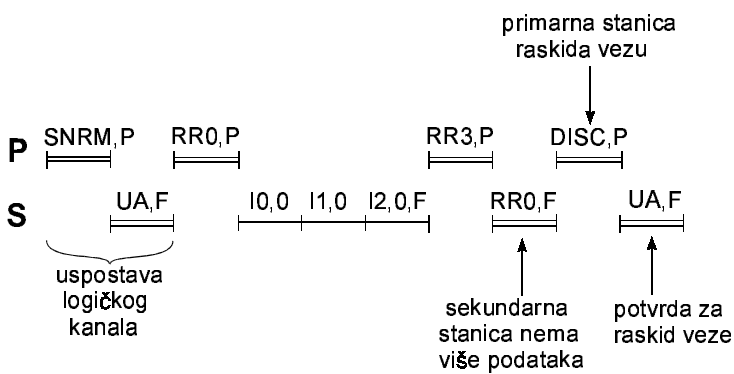
- * START I(N(S)) Pokreće se kada pošaljemo neki I okvir (sa rednim brojem N(S)).
- * RESTART ne postoji
- * istek pogrješka Očekujemo da nam stigne okvir primarne stanice koji sadrži N(R) i time potvrdi N(S), odnosno I(N(S)). Ako predviđeno vrijeme istekne, nastupila je pogrješka.

Rad primarne i sekundarne stanice u NRM modu

== predstavlja U ili S okvir, P i F znači P=1 i F=1, P je primarna stanica, a S sekundarna.

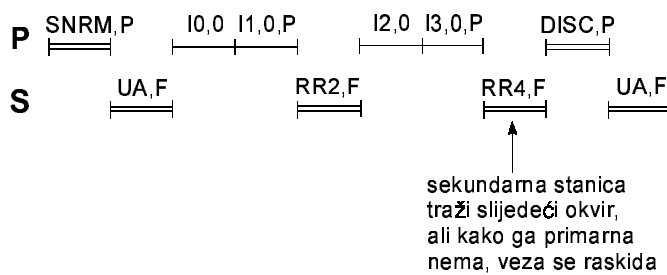
I0,0 - prvi broj je redni broj okvira, a drugi je broj okvira kojeg očekujemo iz suprotnog smjera.

- Prozivanje, slika 4.29.



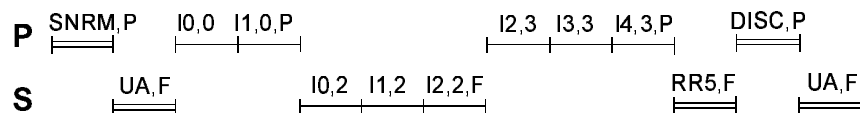
Slika 4.29. Prozivanje u NRM modu

- Selektiranje, slika 4.30.



Slika 4.30. Selektiranje u NRM modu

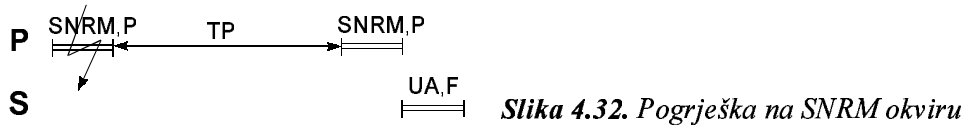
- Dvosmjerni prijenos, slika 4.31.



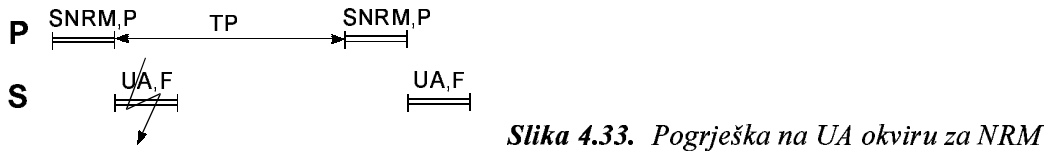
Slika 4.31. Dvosmjerni prijenos u NRM modu

- Pojava pogriješki:

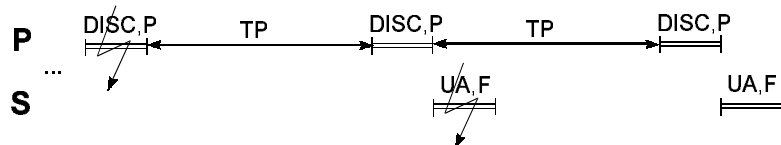
- * Pogriješke u fazi uspostave kanala, slika 4.32. Ako se smetnja pojavi na SNRM,P okviru, sekundarna stanica se ne odaziva. Nakon što prođe određeno vrijeme, primarna stanica shvati da se nešto dogodilo i da mora ponoviti poziv.



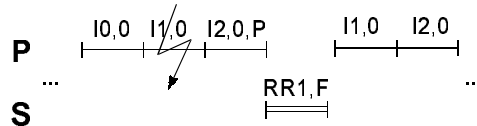
Ako je pogriješka na UA,F okviru, postupak je isti, slika 4.33.



- * Pogriješka u fazi raskida kanala, slika 4.34. Primarna stanica mora slati DISC sve dok se veza ne raskine.

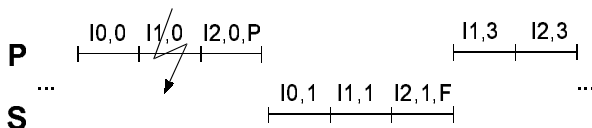


- * Pogriješka na informacijskom okviru, slika 4.35.

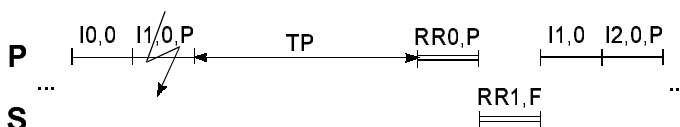


Sekundarna stanica je potvrdila prijem samo prvog okvira i obavijestila primarnu da očekuje prijem ostalih. Tako je primarna stanica shvatila da mora retransmitirati drugi okvir, a i sve ostale. Ovdje smo mogli upotrijebiti i REJ umjesto RR.

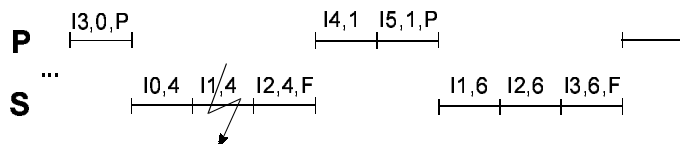
Traženje retransmisije sekundarna stanica može obaviti i bez RR ili REJ, slika 4.36.



Pogriješka se može pojaviti i na posljednjem informacijskom okviru, slika 4.37. Tada koristimo vremenski sklop primarne stanice. Ni ovdje nije nužan REJ okvir.

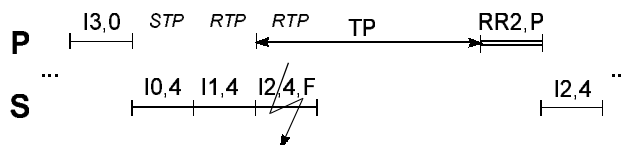


Pogrješka može nastupiti i na informacijskom okviru sekundarne stanice, slika 4.38.



Slika 4.38. Pogreška na informacijskom okviru sekundarne stanice u NRM modu i traženje retransmisije bez upotrebe RR ili REJ okvira

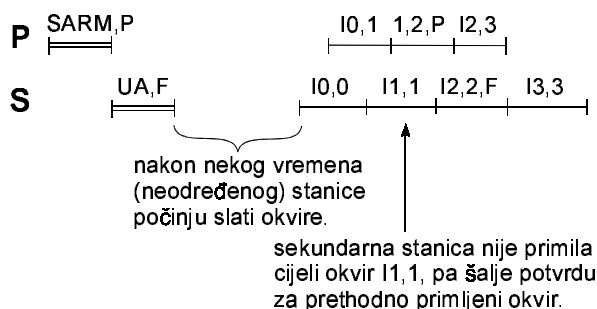
Retransmisija se može tražiti i na način da se TP pokreće svaki put kada primimo informacijski okvir (STP - Start TP; RTP - Restart TP), slika 4.39.



Slika 4.39. Pogreška na informacijskom okviru sekundarne stanice u NRM modu i traženje retransmisije upotrebom vremenskog sklopa primarne stanice

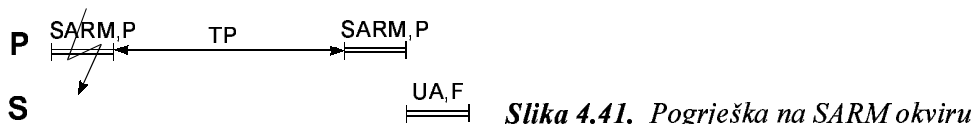
Rad primarne i sekundarne stanice u ARM modu

- Prijenos okvira, slika 4.40. Stanice u bilo kojem trenutku smiju poslati P=1, odnosno F=1. Npr. u ovom primjeru primarna stanica šalje I1,2,P, a sekundarna I2,2,F da bi potvrdila da je sve u redu s prijemom okvira I1,2.



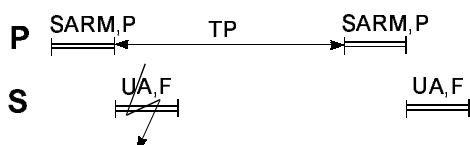
Slika 4.40. Prijenos okvira u ARM modu

- Pojava pogreški:
 - * Pogreške u fazi uspostave kanala, slika 4.41. Ako se na SARM,P okviru pojavi smetnja, sekundarna stanica se ne odaziva. Nakon što prođe određeno vrijeme, primarna stanica shvati da se nešto dogodilo i da mora ponoviti poziv.



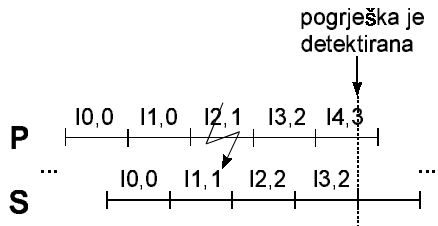
Slika 4.41. Pogreška na SARM okviru

Ako je pogreška na UA,F okviru, postupak je isti, slika 3.42.



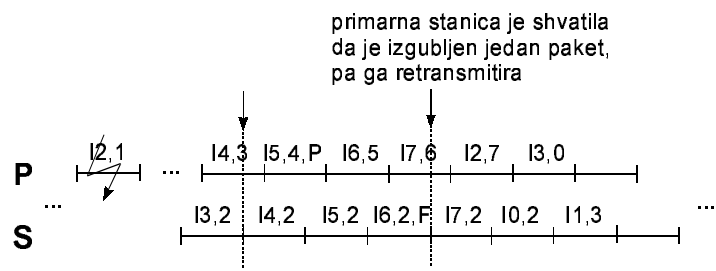
Slika 4.42. Pogreška na UA okviru u ARM

* Pogreška na informacijskom okviru, slika 4.43.



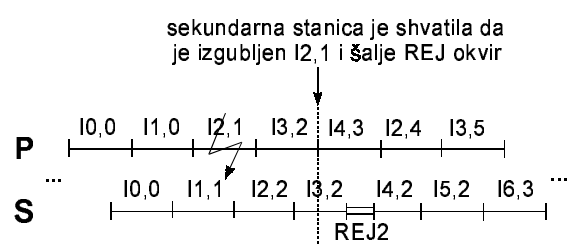
Slika 4.43. Pogreška na I okviru u ARM modu

Nakon detekcije pogreške možemo postupiti na nekoliko načina. Možemo koristiti P/F mehanizam, slika 4.44. Na taj način kasnimo za 6 okvira (od I₂ do I₇), koje moramo retransmitirati.



Slika 4.44. Retransmisija P/F mehanizmom u ARM modu

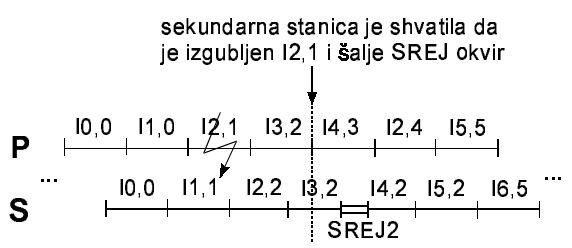
Drugi je način da sekundarna stanica pošalje REJ okvir, slika 4.45.



Slika 4.45. Retransmisija REJ okvirom u ARM modu

Na osnovu REJ2 okvira primarna stanica egzaktno zna da je došlo do gubitka okvira I₂. Ovdje smo nepotrebno poslali samo 2 okvira.

Treći je način korištenje selektivne retransmisije, slika 4.46.



Slika 4.46. Retransmisija SREJ okvirom u ARM modu

Ovdje imamo malo kašnjenje zbog retransmisije samo jednog okvira. Bez obzira na to, ovaj sustav primjenjujemo samo kada nam je potrebna veća brzina odziva, jer zahtjeva veću memoriju i kompleksniju obradu.

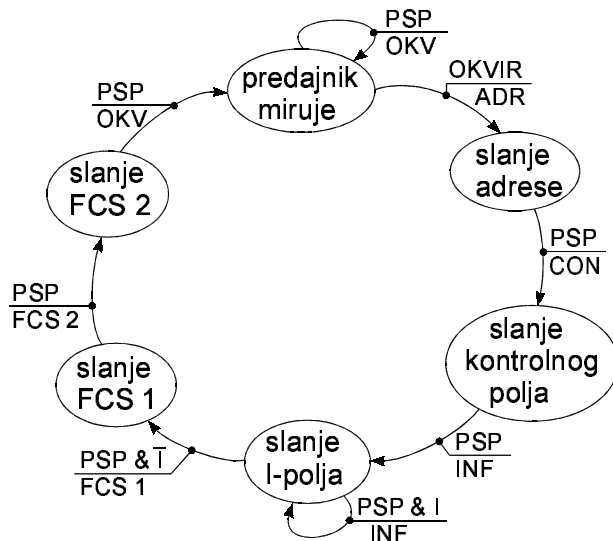
Da bi se pomoglo održavanje sinkronizacije, primarna stanica u ARM modu povremeno šalje P=1, pri čemu se drži pravila da ne smije poslati slijedeći P bit dok ne dobije odgovor na prethodni. Ako ne dobije odziv do isteka nekog određenog vremena, onda ponovo šalje okvir koji je imao P=1.

4.3.3.3 Dijagrami stanja predajnog i prijemnog automata.

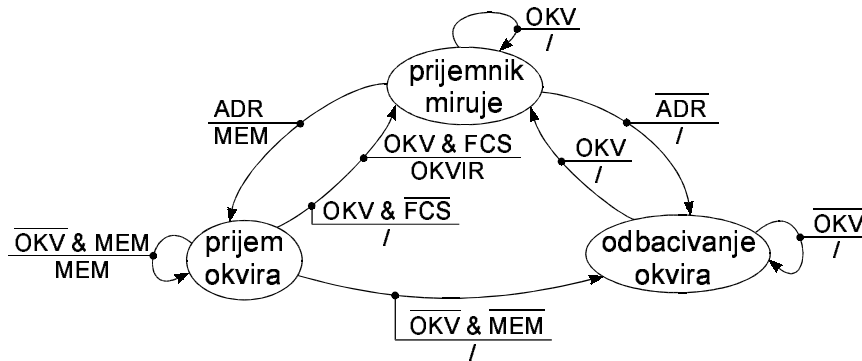
U dijagramima predajnog (slika 4.47) i prijemnog (slika 4.48) automata korištene su slijedeće oznake:

- * PSP - predajni sklop prazan;
- * OKV - slanje okvirnog znaka;
- * OKVIR - nadređena razina zahtjeva predaju;
- * CON - upravljačko polje (C-polje);
- * I - informacijsko polje;
- * INF - informacijski blok;
- * FCS - zaštitni blok;
- * MEM - memorija nije popunjena.

Ako neki od navedenih znakova ima crtu iznad, značenje mu je suprotno.



Slika 4.47. Dijagram predajnog automata



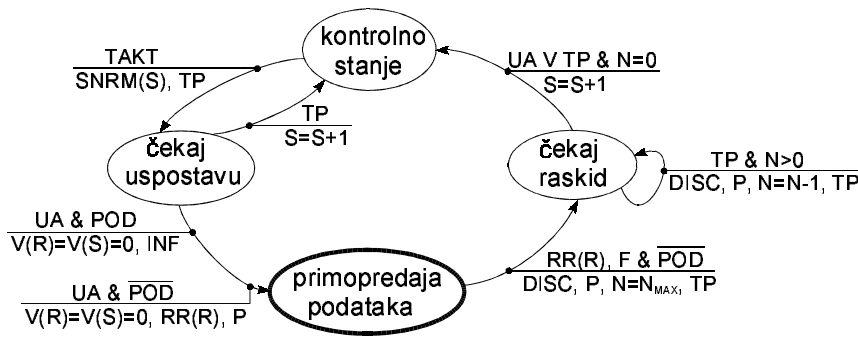
Slika 4.48. Dijagram prijemnog automata

4.3.3.4 Dijagrami stanja za NRM

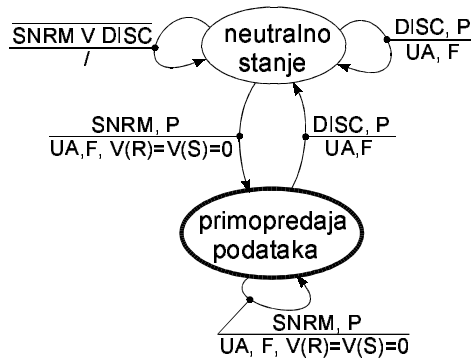
U dijagramima primarne (slika 4.49) i sekundarne stanice (slika 4.50) u fazi uspostave i raskida veze korištene su slijedeće oznake:

- * TAKT - takt prozivanja;
- * TP - vremenski sklop primarne stanice;
- * N - brojač retransmisija;

Ako neki od navedenih znakova ima crtu iznad, značenje mu je suprotno.



Slika 4.49. Primarna stanica u fazi uspostave i raskida veze



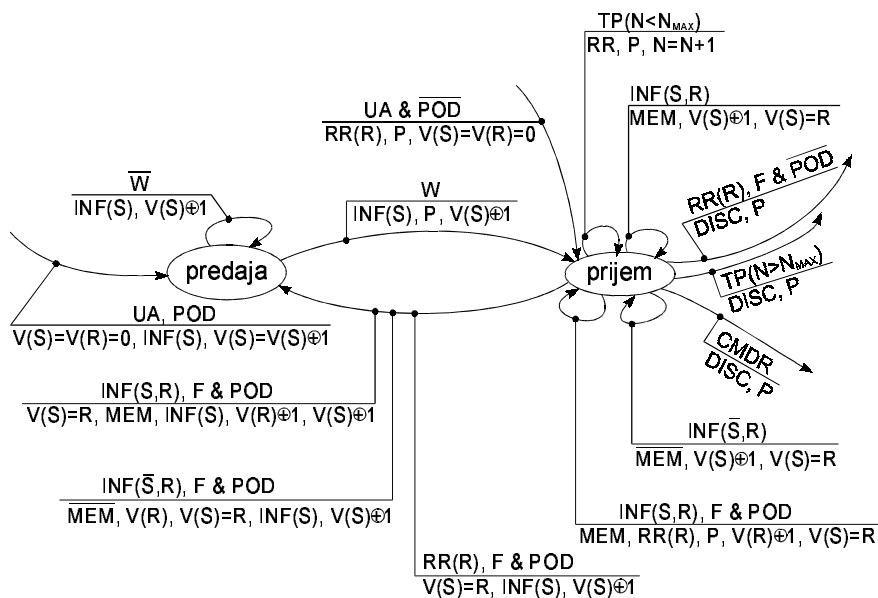
Slika 4.50. Sekundarna stanica u fazi uspostave i raskida veze

4.3.3.5 Dijagrami primopredaje podataka

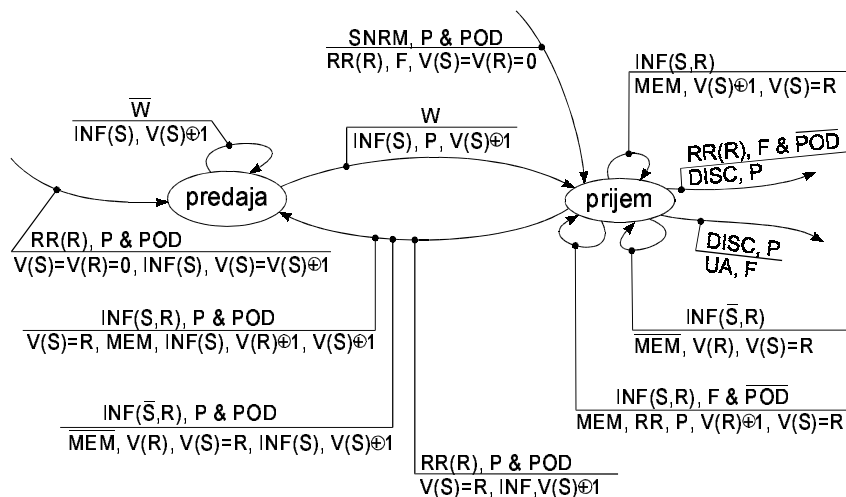
U dijagramima primopredaje podataka za primarnu (slika 4.51) i sekundarnu stanicu (slika 4.52) korištene su slijedeće oznake:

- * W - popunjen prozor;
- * INF(S,R) - točan redni broj informacijskog okvira;
- * TP - vremenski sklop primarne stanice;
- * N - broj retransmisija;
- * POD - podaci poslani.

Ako neki od navedenih znakova ima crtu iznad, značenje mu je suprotno.



Slika 4.51. Primarna stanica u fazi primopredaje podataka



Slika 4.52. Sekundarna stanica u fazi primopredaje podataka

4.3.4 PODATKOVNA RAZINA CCITT X.25 PROTOKOLA

Podatkovna razina X.25 protokola definira novi mod rada - asinkroni balansni mod, koji se pokazao boljim od asinkronog odzivnog moda. Također ima definirane nove komande i odzive kod U okvira. Ovim protokolom nije predviđeno višespojno povezivanje korisnika.

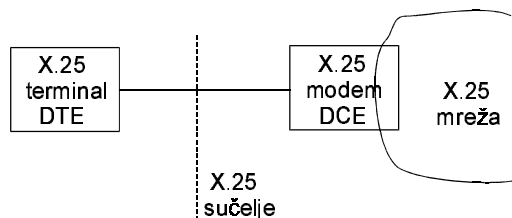
Protokolom X.25 podatkovne razine predviđena su 3 nadzorna okvira:

- RR (Receive Ready) - prijem spreman;
- RNR (Receive Not Ready) - prijem nije spreman;
- REJ (Reject) - odbačen.

Kod U okvira predviđene su određene komande i odzivi:

- komande:
 - * SARM (Set Asynchronous Response Mode) - postavi asinkroni odzivni mod;
 - * SABM (Set Asynchronous Balanced Mode) - postavi asinkroni balansni mod;
 - * DISC (Disconnected) - raskid veze.
- odzivi:
 - * UA (Unnumbered Acknowledgment) - nenumerirana potvrda;
 - * CMDR (Command Reject) - odbacivanje komande;
 - * FRMR (Frame Reject) - odbacivanje okvira;
 - * DM (Disconnected Mode) - nepriključeno stanje.

Za razliku od ISO 4335 standarda, X.25 definira i sučelje, slika 4.53.



Slika 4.53. Sučelje po CCITT X.25

Prema X.25 definirane su dvije 8-bitne adrese:

- A - 1100 0000;
- B - 1000 0000.

Upotrebljavamo ih u 4 različita slučaja (tablica 4.2):

- * adresa B - kada komanda putuje od DTE prema DCE;
- * adresa B - kada odziv putuje od DCE prema DTE;
- * adresa A - kada komanda putuje od DCE prema DTE;
- * adresa A - kada odziv putuje od DTE prema DCE.

	DTE	DCE
K	B	A
O	A	B

Tablica 4.2. Primjena adresa A i B

U ovom standardu se usvaja novi termin - LAP (Link Access Protocol), kojim se definira procedura uspostave i raskida logičkog kanala. Uvođenjem asinkronog balansnog moda LAP-B (Link Access Protocol Balanced) napuštena je striktna podjela na primarne i sekundarne stanice i uveden je pojam kombinirane stanice. Kombinirana stanica izdaje i prima naloge od druge kombinirane stanice, odnosno prema potrebi izvršava funkcije ili primarne ili sekundarne stanice. Zbog razlike u adresama moguća je istovremena uspostava dva logička kanala, tako da je na jednom kanalu primarna stanica DTE, a na drugom DCE.

4.3.5 PRIJENOS PODATAKA MEĐU INTELIGENTNIM MODEMIMA

U poglavlju 3. je spomenuto da inteligentni modemi međusobno komuniciraju protokolom podatkovne razine. Ova komunikacija odvija se automatski, bez znanja korisnika, pa je za njega transparentna.

Protokoli koje koriste inteligentni modemi razvili su se najprije kao industrijski standardi. Poznata je serija standarda firme Micronic pod nazivom MNP (Modem Networking Protocol). MNP-5 određuje mehanizme kontrole pogreški, a MNP-10 kompresiju korisnikovih podataka.

Potreba za službenim standardom rezultirala je u donošenju ITU-T preporuka:

V.42 specificira upotrebu LAP-M (Link Access Protocol for Modems) bitovno orijentiranog protokola, uključivo kontrolu pogreški.

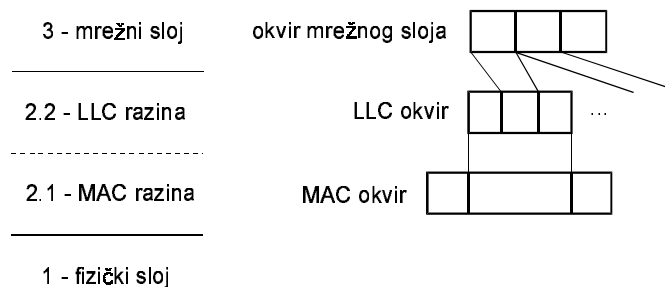
V.42bis specificira pravila kompresije korisnikovih podataka

LAP-M koristi asinkroni balansni mod kao kod X.25, kojemu je vrlo sličan. Specificira uporabu SREJ okvira, te dvaju novih U okvira: XID i TEST. XID se koristi za razmjenu identifikacijskih podataka.

4.3.6 LOKALNE MREŽE

Lokalne mreže su mreže velike brzine, malog kašnjenja i kratkog doseg. Povezuju računala unutar jedne prostorije ili zgrade. Od mnogih tehnologija lokalnih mreža, najčešće se koristi ETHERNET, a poznate su TOKEN-RING, TOKEN-BUS, bežične i druge.

Danas se za lokalne mreže najviše koristi standard IEEE 802.x, kojim su definirane razine LLC (2.2) i MAC (2.1), slika 4.54. LLC specificira bitovno-orijentirani protokol, a MAC oblik okvira koji se pojavljuje na mediju i metode pristupa mediju.



Slika 4.54. Podjela podatkovne razine na LLC (2.2) i MAC (2.1) razinu prema IEEE 802.x

4.3.6.1 LLC (Logical Link Control) protokol

LLC je bitovno-orijentirani protokol, zajednički za sve 802 lokalne mreže, i to u dvije klase:

I. klasa omogućuje bespojni prijenos (connectionless oriented) korištenjem UI okvira. Klasa I. ima definirane određene komande i odzive:

- komande:
 - * UI (Unnumbered Information) - samostalna informacija;
 - * XID (Exchange Identification) - razmijeni podatke za prepoznavanje;
 - * TEST.
- odzivi:
 - * XID;
 - * TEST.

II. klasa omogućuje spojevni prijenos (connection oriented). Sadrži klasične okvire protokola. Stanica koja radi tako da koristi okvire klase I, mora biti u stanju koristiti i okvire klase II.

- S okviri:
 - * RR, prijemnik spreman
 - * RNR, prijemnik nije spreman,,
 - * REJ, odbacivanje.
- U okviri:
 - * komande:
 - ◇ SABME - (Set Asynchronous Balanced Mode Extended) - postavi asinkroni prošireni balansni mod;
 - ◇ DISC.
 - * odzivi:
 - ◇ UA;
 - ◇ DM;
 - ◇ FRMR.

U praksi se od svega ovoga najčešće koristi samo komanda UI.

Nedostatak okvira ranijih bitovno-orijentiranih protokola je bio u nedostatku adrese pošiljaoca. Prema IEEE 802.2, oblik LLC okvira je definiran kao na slici 4.55.

DSAP 8 bita	SSAP 8 bita	C 8 ili 16 bita	I Mx8 bita
----------------	----------------	--------------------	---------------

Slika 4.55. LLC okvir prema IEEE 802.2

- DSAP je adresa odredišne, a SSAP izvorišne priključne točke, slika 4.56. Pomoću DSAP i SSAP identificiraju se korisnici logičkog kanala. Predviđeno je da se ove adrese koriste dinamički u trenutku uspostave veze. Kako na mrežnoj razini korisnik može biti samo protokol mrežne razine, i to isti za oba učesnika, nedostatak koncepta priključnih točaka je u nepostojanju standardne identifikacije mrežnih protokola.

DSAP I/G DDDDDDD	SSAP C/R SSSSSS
---------------------	--------------------

Slika 4.56. Oblik adresa DSAP i SSAP

- * I/G=0 - individualna adresa;
- * I/G=1 - grupna adresa;
- * D - bitovi DSAP adrese;
- * C/R=0 - komanda;
- * C/R=1 - odziv;
- * S - bitovi SSAP adrese.
- Kod originalnog ISO standarda nedostatak je što se za C polje unaprijed ne zna da li će biti 8-bitno ili 16-bitno. Kod LLC je to riješeno na način da se za S i I okvire koristi 16-bitno C polje, a za U okvire 8-bitno, slika 4.57.
- Informacijsko polje se, za razliku od HDLC, sastoji od cjelovitih okteta, odnosno duljine je Mx8b.

	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	X	X	X	X	P/F	N(R)						
U	1	1	M	M	U	M	M	M	P/F							

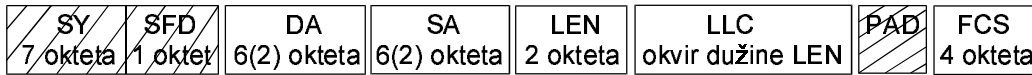
Slika 4.57. C-polje na LLC razini

4.3.6.2 MAC (Media Access Control) razina

Definirana je prema IEEE standardima 802.x, za svaku vrstu lokalnih mreža zasebno. Odgovara razini 2.1. MAC je prilagođen vrsti medija, pa je LLC zajednički za sve vrste lokalnih mreža.

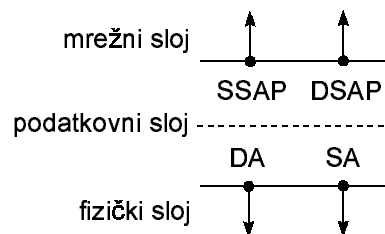
IEEE 802.3 – ETHERNET.

MAC okvir (slika 4.58) se sastoji od slijedećih polja:



Slika 4.58. Okvir za ETHERNET prema IEEE 802.3

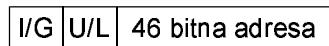
- SY (Synchronization) je znak dužine 7 okteta, oblika 10101010. Služi za uspostavu sinkronizacije.
- SFD (Start Frame Delimiter) je okvirni znak, oblika 10101011.
- DA (Destination Address) je odredišna adresa, a SA (Source Address) izvorišna adresa. DA i SA su adrese fizičkog uređaja prema fizičkom sloju, za razliku od SSAP i DSAP, koje su adrese korisnika na mrežnoj razini, slika 4.59.



Slika 4.59. Adrese fizičkog uređaja DA i SA i adrese korisnika DSAP i SSAP

- DA i SA mogu biti 48-bitne (slika 4.60) ili 16-bitne (slika 4.61):

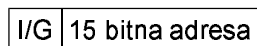
1) 48-bitno adresno polje ima slijedeća polja:



Slika 4.60. 48-bitno adresno polje

- ◇ I/G=0 - individualna adresa;
- ◇ I/G=1 - grupna adresa. Ako su svi bitovi grupne adrese u jedinici, ona postaje univerzalna adresa i podatak adresiran na taj način mogu primiti sve stanice.
- ◇ U/L=0 - globalno administrirana adresa, upisuje proizvođač veznog sklopa;
- ◇ U/L=1 - lokalno administrirana adresa, određuje samostalno administrator mreže.

2) 16-bitno adresno polje.



Slika 4.61. 16-bitno adresno polje

- ◇ Ovdje nije potreban bit U/L, jer je 16-bitna adresa uvijek lokalno administrirana.

Iako globalno administrirane, ove su adrese neupotrebjive kao mrežne adrese. Zato se koriste adrese na mrežnoj razini. Postavlja se problem povezivanja adrese na MAC razini sa adresom na mrežnoj razini. Kod Interneta je taj problem riješen ARP (Address Resolution Protocol) protokolom.

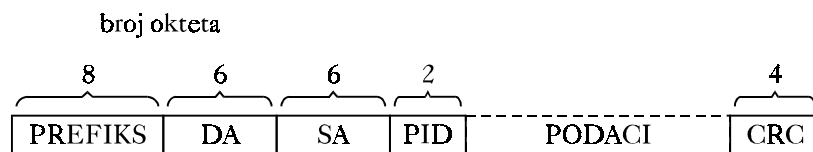
- LEN (Length) je polje koje nam kaže koliko ima okteta korisnikove informacije.
- Informacijsko polje sadrži korisnikove podatke, npr. kompletan LLC okvir dužine LEN.
- FCS je polje zaštite. koristi se ciklička zaštita polinomom CRC-32 koja uključuje sve bitove okvira.

- PAD okvir - zbog kašnjenja informacije, uzrokovano dužinom kabela između stanica, 802.3 predviđa da polje mora imati neku minimalnu dužinu da bi se moglo prenijeti. Ako je dužina manja od minimalne, između LLC i FCS se mogu ubaciti dodatni okteti kao nadogradnja. Npr. kod ETHERNET-a, brzine prijenosa 10 Mb/s, definirana je minimalna dužina okvira od 512 bita, odnosno 64 okteta. Vrijeme predaje takvog okvira iznosi 51.2 μ s. Kako je na kابلu dužine 2.5 km predvidljivo kašnjenje od 12.5 μ s, sigurno je da će okvir stići do najudaljenije stanice prije isteka njegove predaje, te na taj način osigurati otkrivanje kolizije od strane svih stanica na vezi.

Problem standarda 802.3 je u tome što nije adekvatno riješena identifikacija protokola na mrežnoj razini, što se nastojalo riješiti tokom razvoja.

Specifičnosti u standardizaciji ETHERNET-a.

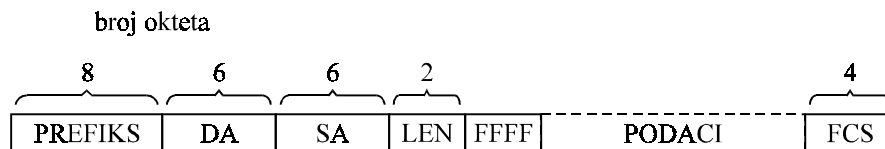
Razvoj ETHERNET-a je počeo u firmi XEROX PARC. Zatim su firme DEC, INTEL i XEROX napravile DIX ETHERNET. Nakon toga je objavljena ETHERNET II specifikacija (slika 4.62), koja se i danas masovno koristi, iako nije formalno standardizirana.



Slika 4.62. ETHERNET II okvir

Osnovna razlika u odnosu na 802.3 okvir je u PID (Protocol Identifier), 16-bitnom polju za identifikaciju protokola. U polju podaci prenosi se kao SDU paket mrežne razine, dakle LLC protokol se ne koristi.

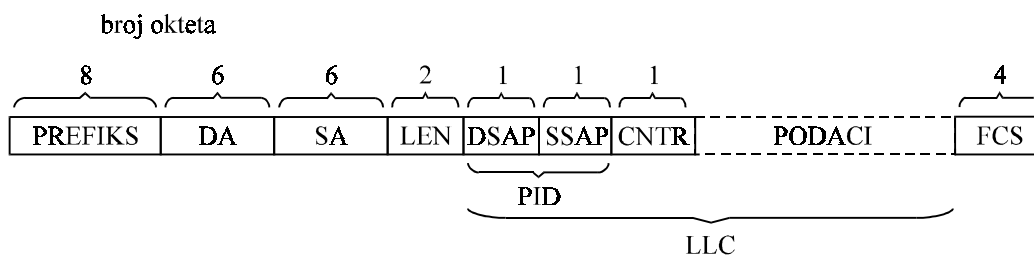
Poslije je usvojen standard 802.3 (vidi gore), kao napor da se standardiziraju sve vrste lokalnih mreža. Kod njega je polje PI zamijenjeno poljem LEN. Firma NOVELL je prva počela koristiti 802.3 okvir, ali bez LLC okvira. Danas je on poznat kao "802.3" okvir, slika 4.63.



Slika 4.63. Okvir prema standardu 802.3 za IPX paket

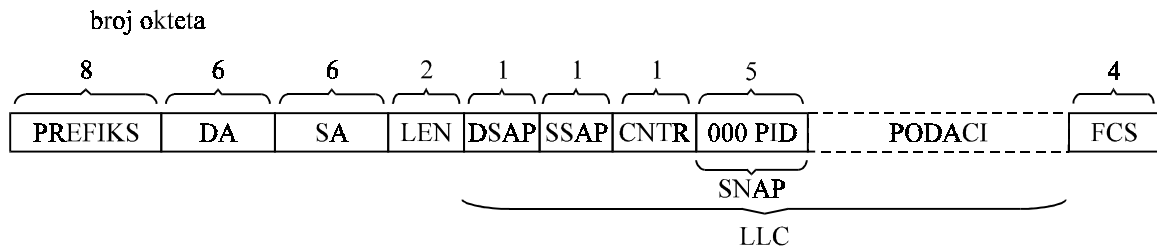
U polju iza LEN prvi oktet FF je ekvivalentan adresi odredišne priključne točke DSAP, a drugi adresi polazišne priključne točke SSAP. To su univerzalne adrese, što je prihvatljivo za DSAP, ali je neprihvatljivo za SSAP. Oktetima FFFF definiran je IPX mrežni protokol u informacijskom polju. Okviri 802.3 i ETHERNET II se na istoj mreži međusobno ne prepoznaju, a katkad i smetaju jedan drugome. ETHERNET II se od ostalih 802.3 okvira razlikuje po vrijednosti PI polja, koja je veća od 2048 okteta, dok je najveća dužina okvira 1500 okteta.

Zatim se pokušalo iskoristiti originalnu specifikaciju 802.3 sa uključenim LLC okvirom, ali na način da polja SSAP i DSAP zajedno posluže kao identifikator protokola mrežne razine, slika 4.64.



Slika 4.64. Okvir prema standardu 802.3, sa poljima DSAP i SSAP kao PI

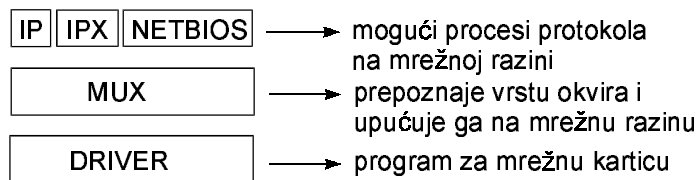
DSAP i SSAP su imale fiksno dodijeljene vrijednosti, što je bilo neprihvatljivo zbog malog broja adresa. Osim toga, ova koncepcija se nije slagala sa originalnim standardom i nije bila šire prihvaćena, pa je definiran SNAP (Sub Network Access Protocol) okvir, slika 4.65.



Slika 4.65. LLC sa SNAP okvirom

Prednost je SNAP okvira što je kompatibilan i standardan za sve vrste lokalnih mreža.

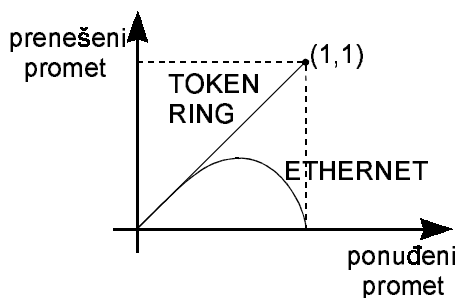
Danas se na lokalnim ETHERNET mrežama mogu naći 4 različita okvira. Zbog toga programska podrška podatkovne razine mora imati sustav multipleksiranja (slika 4.66), koji će prepoznati vrstu okvira i uputiti je na ispravne procese na mrežnoj razini.



Slika 4.66. ETHERNET sa više različitih okvira i sustavom multipleksiranja

IEEE 802.5 – TOKEN-RING.

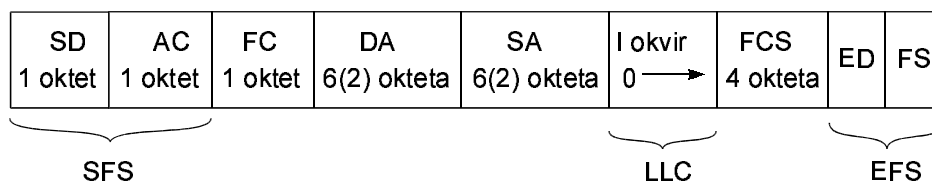
Kod IEEE 802.5 lokalnih mreža stanice su povezane u prsten. Pristupna riječ (Token) kruži prstenom sve dok ne naiđe na stanicu koja ima spremne podatke. Ta stanica umjesto pristupne riječi šalje okvir, koji putuje kroz cijeli prsten i uništi ga kada ponovo dođe do nje. Iskorištenje kanala je maksimalno, čak i kod velikog opterećenja, slika 4.67., jer su za razliku od ETHERNET mreže, izbjegnute kolizije.



Slika 4.67. Usporedba TOKEN-RING i ETHERNET LAN

Toliko poboljšanje ipak nije dovoljno da bi se dala prednost prstenastim mrežama, zbog nekompatibilnosti službenog standarda i najpoznatijih proizvođača opreme.

Okvir TOKEN-RING mreže (slika 4.68) sastoji se od slijedećih polja:



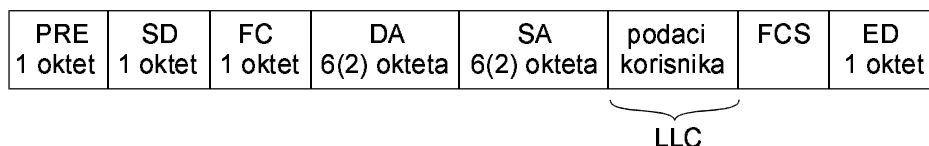
Slika 4.68. Okvir prema IEEE 802.5

- SFS (Start of Frame Sequence) - sekvenca koja služi za sinkronizaciju. Sastoji se od:
 - * SD - okvirni znak $J K 0 J K 0 0 0$, gdje su J i K non-data bitovi, koji remete normalni način prijenosa podataka i služe za identifikaciju početka i kraja okvira. Iskorišteni su da bi protokol bio transparentan.
 - * AC (Access Control) - oktet oblika $P P P T M R R R$.
 - ◇ PPP su tzv. bitovi prioriteta (Priority bit), koji omogućuju da se odredi prioritet poruke. Oni postoje u svakoj pristupnoj riječi i njih mogu iskoristiti stanice sa tim ili većim prioritetom;
 - ◇ T je token bit. Ako je T=0, onda imamo pristupnu riječ. Ta riječ putuje kroz mrežu i, kada naiđe na stanicu koja ima spremne podatke, mijenja token bit u T=1, što je znak za informacijski okvir;
 - ◇ M - monitor bit. Služi da bi se spriječilo beskonačno kruženje iste poruke kroz prsten. Jedna stanica u prstenu ima ulogu nadzorne stanice. Ona svakom okviru postavi M=1 (bio je M=0). Ako ponovo primi okvir sa M=1, skida ga sa prstena, jer je to okvir koji je već prošao cijeli prsten;
 - ◇ RRR - rezervni bitovi.
 - FC - kontrolno polje. Omogućuje prijenos različitih vrsta okvira, tj. okvira za upravljanje mrežom i okvira s korisnikovim podacima. Oblik mu je $F F Z Z Z Z Z Z$. FF bitovi služe za identifikaciju okvira. Ako je FF=00, radi se o MAC okviru za upravljanje mrežom, a ako je FF=01, onda je to LLC okvir, koji nosi korisnikovu informaciju. Ako se radi o MAC okviru, Z-bitovi određuju koju akciju treba poduzeti;
 - DA i SA su adrese polazišta i odredišta.
 - I - informacijsko polje. Nema definiranu dužinu, odnosno može biti dugo više okteta. Zato na kraju imamo sekvencu EFS da se naznači kraj okvira.
 - EFS (End of Frame Sequence). Sastoji se od :
 - * ED - omogućuje detekciju kraja okvira. Oblik mu je $J K 1 J K 1 I E$.
 - ◇ J i K, non-data bitovi;
 - ◇ I (Intermediate frame bit) - omogućuje slanje okvira u lancu, odnosno stanica može poslati više okvira za jednu kontrolnu riječ;
 - ◇ E (Error detected) - ako primljena stanica detektira pogrešan okvir, postavlja E=1, a ako je okvir ispravno primljen, ostavlja E=0. Na taj način predajna stanica kada skida okvir iz prstena može utvrditi da li je okvir ispravno primljen. Ako nije, retransmitira ga.
 - * FS (Frame Status) - ima oblik $A C r r A C r r$.
 - ◇ A (Adress recognize) - adresa prepoznata;
 - ◇ C (frame copied) - okvir je primljen.
- Imamo više slučajeva:
- a) A=0, C=0 - nema te stanice;
 - b) A=1, C=0 - stanica postoji, ali okvir nije primljen;
 - c) A=1, C=1 - stanica postoji i okvir je primljen.

IEEE 802.4 – TOKEN-BUS.

Stanice nisu fizički spojene u prsten, nego uspostavljaju logički prsten na način da generiraju slučajan broj, na osnovu kojega same određuju svoj redoslijed u prstenu. Ako se dogodi da dvije stanice izaberu isti broj, potrebno je ponoviti postupak generiranja slučajnog broja. Nakon uspostave logičkog prstena šalje se pristupna riječ i daljnji rad stanica u prstenu je isti kao da se radi o fizičkom prstenu. Ovakav spoj zadržava prednosti prstenastog spoja, a ima i prednosti sabirničkog, jer se pri isključenju stanice iz prstena ili uključenju nove stanice u prsten treba samo uspostaviti novi logički prsten.

Okvir je nešto drugačiji od okvira u prstenastim mrežama, slika 4.69.



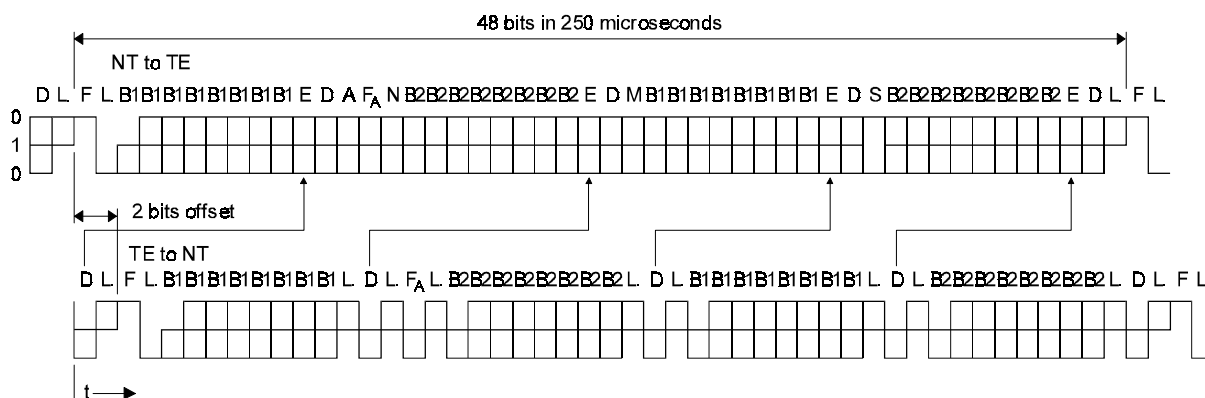
Slika 4.69. Okvir prema IEEE 802.4

- PRE - uvodna riječ. Služi za sinkronizaciju po bitu;
- SD - okvirni znak, oblika $N N 0 N N 0 0 0$. N je non-data simbol i služi za sinkronizaciju sekvence.
- FC - polje kontrole okvira, oblika $F F C C C C C C$.
 - ◊ FF - određuje vrstu okvira (MAC ili LLC);
 - ◊ C bitovi određuju dodatne parametre.
- ED - omogućuje detekciju kraja okvira. Oblik mu je $N N 1 N N 1 I E$.
 - ◊ I (Intermediate bit) označava da li slijedi još okvira u lancu;
 - ◊ E - detektira pogrešku na okviru.

4.3.7 ISDN PRETPLATNIČKE MREŽE

Podatkovna razina ISDN S/T sučelja specificirana je u dvije podrazine: specifična MAC podrazina dio je specifikacije fizičke razine, preporuka ITU-T I.430, dok je podrazina 2.2 definirana nizom preporuka ITU-T I.440 = Q920, nazvana DSS1 (Digital Subscriber Signaling 1). DSS1 se odnosi samo na signalizacijski protokol za D kanal sučelja, dok se korisnikovi podaci prenose B kanalima bit po bit bez primjene nekog protokola (ovisi o korisniku).

I.430 omogućava prijenos bitova 2B+D kanala u strogo sinkronom režimu, neophodnom za prijenos govora i podataka PCM kanalima 64 kb/s. Oblik okvira prikazan je na slici 4.70:



Slika 4.70. Okvir I.430

- | | |
|---|--|
| F = okvirni (framing) bit | N = -FA, negacija FA bita |
| L = bit za eliminaciju istosmjerne komponente | B1, B2 = bit kanala B1 ili B2 |
| D = bit D kanala | A = bit aktiviranja |
| E = bit jeke (echo) D kanala | S = bit S kanala |
| FA = dodatni okvirni bit | M = bit za povezivanje okvira (multiframe) |

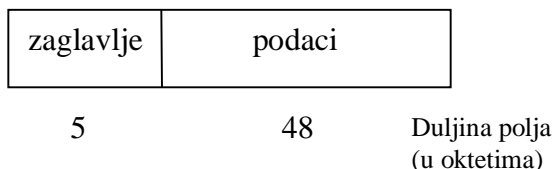
NT kao primarna stanica šalje okvir, a TE1 se odaziva sinkrono s 2 bita zakašnjenja.

LAP-D je gotovo identičan LAP-B protokolu preporuke X.25.

4.4 ATM MREŽE

4.4.1 ATM TEHNOLOGIJA

ATM (Asynchronous Transfer Mode) je prijenosno-komutacijska tehnologija, razvijena za potrebe izgradnje širokopoljnih digitalnih mreža integriranih usluga (B-ISDN), za prijenos glasa, video zapisa i podataka. To je mreža s prospajanjem kratkih paketa fiksne duljine nazvanih ćelijama ili stanicama (cell), slika 4.71. Svaka ćelija sastoji se od zaglavlja duljine 5 okteta i tijela s podacima korisnika ili za upravljanje mrežom duljine 48 okteta, svega 53 okteta. Izabrana duljina ćelije je kompromis između zahtjeva za malim početnim kašnjenjem kod govornih komunikacija i za sklopovskim prospajanjem s jedne, te iskorištenja prijenosnog medija s druge strane.

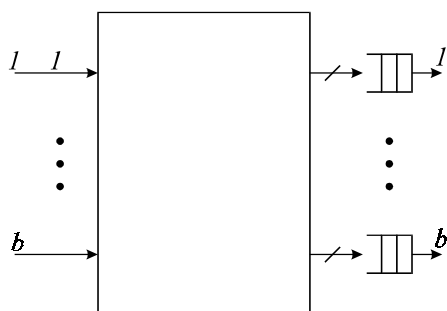


Slika 4.71. ATM ćelija

Ćelije se u ATM mrežama prosljeđuju virtualnim kanalima (Virtual Channel) i virtualnim stazama (Virtual Path). **Virtualni kanal** je put kroz mrežu kojim prolaze sve ćelije određenog toka, određen na početku prijenosa. Redoslijed isporuke ćelija je zagarantiran. **Virtualna staza** je skup virtualnih kanala koji na nekom segmentu mreže dijele zajednički put. Zaglavlje ćelije nosi identifikatore virtualnog kanala i staze, koji zajedno jednoznačno određuju tok podataka.

4.4.1.1 Prospajanje ćelija

Karakteristična topologija ATM mreža je stablasta, kod koje u čvorištima nalazimo ATM prospojnike. ATM prospojnici svaku pojedinu ćeliju sa ulaznog komunikacijskog kanala prebacuju na traženi izlazni komunikacijski kanala. Pri tome ćelije čekaju na prijenos u izlaznim redovima čekanja, slika 4.72. Prospajanje se odvija potpuno sklopovski, čime se postižu velike brzine prijenosa.

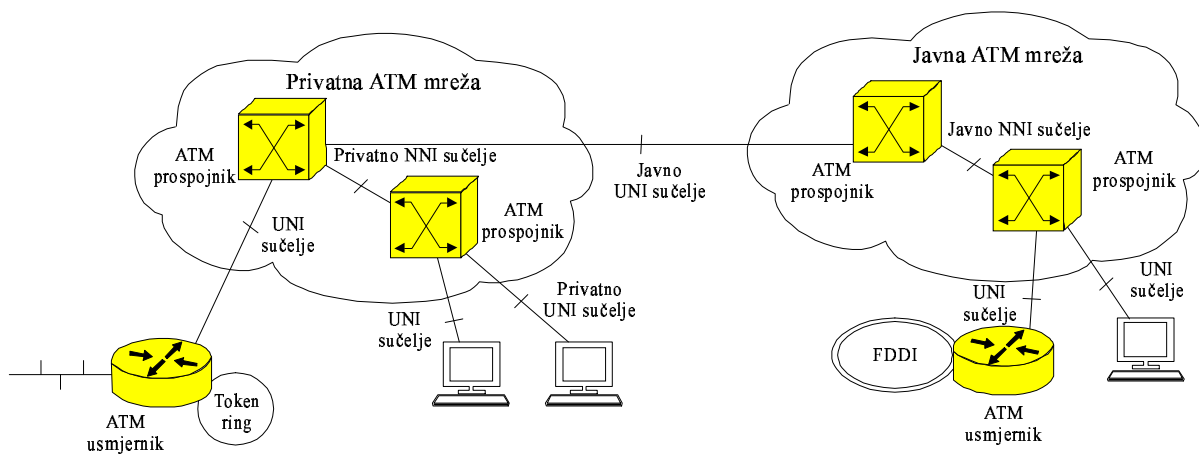


Slika 4.72. Struktura ATM prospojnika

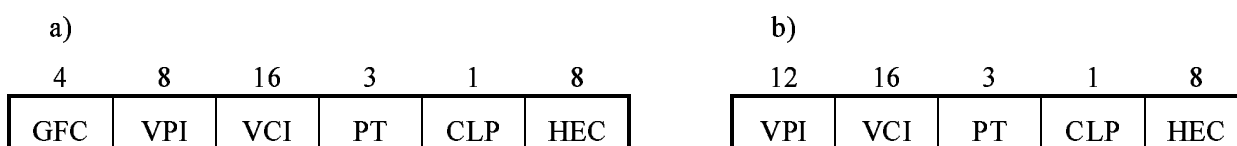
Podjela na virtualni kanal i stazu omogućava korištenje dviju vrsta ATM prospojnika. **Prospojnik kanala** identificira svaki tok na osnovu identifikatora virtualnog kanala i staze, te prospaja ćelije tako da one pri tome mogu mijenjati stazu. **Prospojnike staze** identificira samo grupu tokova koji pripadaju stazi na osnovu identifikatora staze, te prospaja ćelije tako da ostanu u istoj stazi. Prospojnik staze je jednostavniji i koristi se za ostvarivanje virtualnih mreža.

Kod ATM mreža definirana su dva sučelja, korisničko sučelje UNI (User Network Interface) i mrežno sučelje NNI (Network Node Interface). Kod privatnih mreža ta sučelja su PUNI, privatni UNI i PNNI, privatni NNI, slika 4.73.

UNI i NNI sučelja su vrlo slična, ali se razlikuju u formatu zaglavlja ćelije. Kod korisničkog (UNI) sučelja u zaglavlje ćelije uključeno je polje GFC (Generic Flow Control). Kod mrežnog (NNI) sučelja tog polja nema, ali je zato prošireno polje identifikatora virtualne staze, slika 4.74. Na taj je način omogućena kontrola ponašanja korisnika (brzine odašiljanja ćelija) na UNI sučelju, a povećan broj mogućih virtualnih staza na NNI sučelju.



Slika 4.73. UNI i NNI sučelja kod ATM mreža



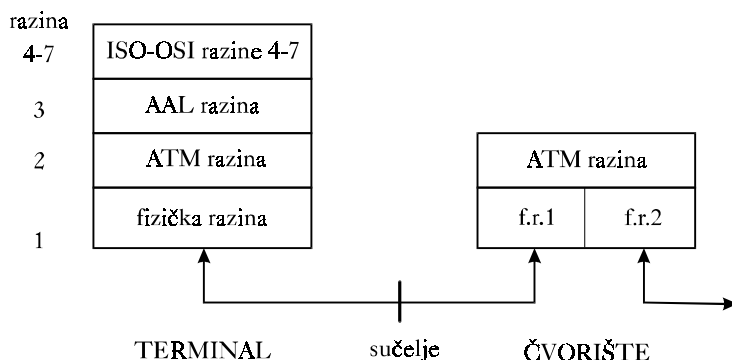
Slika 4.74. Format zaglavlja za a) UNI i b) NNI ćelije

Pri tome korištene kratice imaju sljedeća značenja:

- GFC (Generic Flow Control) koristi se za lokalnu kontrolu toka, u praksi nema značenja.
- VPI (Virtual Path Identifier) identificira virtualnu stazu kojoj ćelija pripada.
- VCI (Virtual Channel Identifier) identificira virtualni kanal kojem ćelija pripada.
- PT (Payload Type) identificira vrstu ćelije. Prvi bit određuje da li ćelija prenosi korisničke podatke. U tom slučaju drugi se bit koristi za indicaciju zagušenja, a treći za označavanje kraja niza ćelija koji pripadaju jednom AAL5 okviru.
- CLP (Congestion Loss Priority) određuje prioritet ćelije. Ćelije nižeg prioriteta biti će odbačene ako dođe do zagušenja. Ćelija može biti označena nižim prioritetom ako je to propisao za tip usluge, ili ako korisnik prekorači svoja prava brzine emitiranja ćelija.
- HEC (Header Error Control) je polje zaštitnih bitova zaglavlja, generirano korištenjem cikličkog koda.

4.4.1.2 Referentna ATM arhitektura

ATM mreža izvorno je zamišljena kao homogena mreža. ATM arhitektura sastoji se od fizičke, ATM i AAL razine, slika 4.75:



Slika 4.75. ATM arhitektura

Fizička razina ATM mreža omogućava prijenos ćelija različitim medijima i opisana je ranije.

ATM razina odgovorna je za uspostavu i raskid virtualnih kanala i staza, te za prosljeđivanje ćelija tim putovima. Za to se koriste podaci u zaglavlju ćelije. Obavlja funkcije kontrole pristupa i toka.

Kod uspostave virtualnog kanala, šalje se ćelija koja sadrži globalnu adresu odredišta, te zahtjevane parametre kvalitete usluge. Koristi se više formata adrese, tri za privatne i jedan (E.164) za javne mreže. Ako ćelija pronade put kroz mrežu koji garantira traženu kakvoću usluge, formira se virtualni kanal kojim prolaze sve ćelije toga toka. Taj kanal definiraju tablice usmjeravanja i identifikatora.

Na raznim segmentima mreže isti virtualni kanal može imati različite identifikatore (jer je početni broj kanala na nekom srednjem segmentu bio zauzet), pa se kod svakog usmjeravanja ćelije mijenja identifikator virtualnog kanala i staze, te preračunava HEC.

U fazi prijenosa podataka preuzima podatkovna polja ćelija od AAL razine, te formira zaglavlja. Na prijemnoj strani odvađa zaglavlja od podatkovnog polja, te podatkovna polja isporučuje AAL razini.

AAL prilagodna razina (ATM Adaptation Layer) ostvaruje vezu između većih jedinica podataka (SDU) i ATM ćelija. AAL razina prima SDU od nadređenih razina, npr. pakete mrežne razine Interneta, te obavlja segmentaciju na dijelove duljine 48 okteta. Sastoji se od dvije podrazine, SAR (Segmentation and Reassembly) i CS (Convergence Sublayer). SAR obavlja samu segmentaciju, dok funkcija CS ovisi o vrsti usluge.

Interesantno je primijetiti da arhitektura ATM mreža ne odgovara potpuno ISO-OSI modelu. Kad se radi o homogenoj ATM mreži, prospajanje i usmjeravanje s kraja na kraj mreže obavlja se na ATM (drugoj) razini, iako bi to trebalo obavljati na mrežnoj (trećoj) razini. AAL razina je samo prilagodba prema višim razinama i nema funkcije mrežne razine.

Kada je ATM mreža samo dio kompleksne heterogene mreže, i ATM i AAL razine nalaze se ispod mrežne razine te mreže. Kako danas još ne postoji globalna ATM mreža, a ATM tehnologija se najčešće koristi za izgradnju lokalnih mreža ili dijelova Interneta, čitava ATM mreža zapravo pripada podatkovnoj razini ISO-OSI modela.

4.4.1.3 Usluge ATM mreže

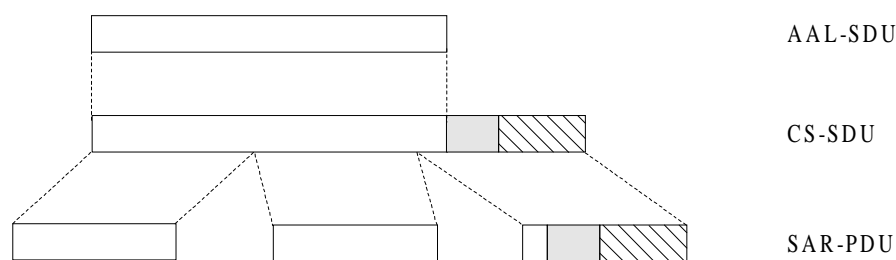
B-ISDN mreže, a time i ATM tehnologija, namijenjeni su integraciji svih vrsta prometa. To se ostvaruje rezerviranjem kapaciteta mreže za tokove koji zahtijevaju određenu kvalitetu usluge, te kontrolom pristupa i kontrolom toka. Predviđene su četiri vrste usluge:

- **CBR** (Constant Bit Rate), usluga stalne brzine prijenosa za nekomprimirane govorne i video signale. Odgovara joj AAL1.
- **VBR** (Variable Bit Rate), usluga promjenljive brzine prijenosa za komprimirane govorne i video signale. Odgovara joj AAL2. Koristi se u dvije varijante ovisno o realnom vremenu.
- **ABR** (Available Bit Rate), usluga raspoložive brzine prijenosa za prijenos podataka. Odgovara joj AAL3/4 i AAL5. Mreža obavještava izvorište o zagušenju ili o optimalnoj brzini, tako da ono može reducirati brzinu slanja ćelija.
- **UBR** (Unspecified Bit Rate), usluga nespecificirane brzine prijenosa za prijenos podataka. Odgovara joj AAL5. Mreža ne obavještava izvorište o zagušenju, već jednostavno u slučaju zagušenja odbacuje ćelije. Predajnik na osnovu gubitaka treba reducirati brzinu slanja ćelija.

Za ove usluge definirane su različite AAL razine:

- **AAL0** je prazna AAL razine. Funkcije AAL razine prepuštene su korisniku.
- **AAL1** osigurava prijenos nekomprimiranih govornih signala. Koristi se kod CBR usluge.
- **AAL2** osigurava prijenos komprimiranih govornih signala. Koristi se kod VBR usluge.
- **AAL3/4** osigurava prijenos podataka spojevnim protokolom. Najprije definirane kao različite usluge, AAL3 i AAL4 su ujedinjeni u zajedničku specifikaciju. Kompleksnost protokola prijeći masovno korištenje, pa je naknadno razvijen AAL5.

- **AAL5** osigurava prijenos podataka bespojnim protokolom. SDU korisnika (AAL SDU) nadopunjuje se AAL5 CS-SDU dodatkom, te se tako formirani CS-SDU segmentira na ćelije (SAR-PDU), slika 4.76.



Slika 4.76. AAL5

4.4.1.4 Kvaliteta usluge ATM mreže

Kod uspostave virtualnog kanala, korisnik i mreža ugovaraju parametre kvalitete usluge, tablica 4.3

ugovorena veličina	CBR	VBR(rt)	VBR(nrt)	UBR	ABR
PCR, CDVT	DA			DA	DA
SCR, MBS, CDVT	NE	DA		NE	
MCR	NE			NE	DA
CDV _{pp}	DA	DA	NE	NE	NE
CTD _{sr}	NE	NE	DA	NE	
CTD _{max}	DA	DA	NE	NE	
CLR	DA			NE	DA
povratna veza	NE			NE	DA

Tablica 4.3. Parametri kvalitete usluge za ATM kanale

gdje je:

- PCR (Peak Cell Rate) vršna brzina odašiljanja ćelija
- SCR (Sustained Cell Rate) srednja brzina odašiljanja ćelija
- MCR (Minimal Cell Rate) minimalna brzina odašiljanja ćelija
- CDVT (Cell Delay Variation Tolerance) tolerancija perioda među odašiljanim ćelijama
- MBS (Maximum Burst Size) najveća duljina praska
- CDV (Cell Delay Variation) varijacija kašnjenja na mreži
- CTD (Cell Transfer Delay) prijenosno kašnjenje ćelija
- CLR (Cell Loss Ratio) omjer izgubljenih ćelija

Ukoliko je mreža u stanju pružiti traženu kvalitetu usluge, virtualni kanal su uspostavlja. U suprotnom, zahtjev se odbija. Time je ostvarena kontrola pristupa (CAC, Customer Admission Control).

Nakon uspostave virtualnog kanala, očekuje se da će korisnik poštovati ugovorene parametre maksimalne i srednje brzine odašiljanja. Mreža na UNI sučelju provjerava sukladnost korisnikovog toka s ugovorenim parametrima. Ukoliko korisnik emitira brže nego što je ugovoreno, ćelije se odbacuju ili označavaju nižim prioritetom. Na taj način, mreža se štiti od zlonamjernog korisnika, te štiti druge korisnike ostvarujući ugovorenu kvalitetu usluge.

Kod ABR kategorije usluge, mreža dojavljuje zagušenje korisniku. Predajnik periodično šalje RM (Resource Management) ćelije, koje sadrže polja za dojavu zagušenja.

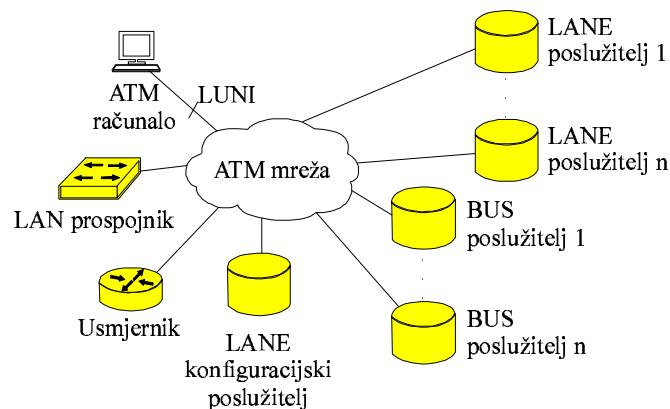
4.4.2 KORIŠTENJE ATM MREŽA

Kako ne postoji homogena globalna ATM mreža, ATM tehnologija se koristi za izgradnju lokalnih, kao i segmenata mreža širokog doseg. Za prijenos podataka razvijeno je nekoliko arhitektura.

4.4.2.1 Lokalna mreža s ATM tehnologijom (LANE)

LANE (LAN Emulation) je skup protokola kojim se omogućava korištenje ATM tehnologije za izgradnju bespojnih lokalnih mreža, čija je funkcionalnost usporediva s Ethernet mrežom. ATM terminali (računala) posredstvom ATM mreže komuniciraju kao da su spojeni na lokalnu mrežu. Takva lokalna mreža predstavlja domenu prostiranja (broadcast) i može se povezati kao podmreža na Internet.

LANE je zapravo simulacija (emulacija) lokalne mreže ATM mrežom. Prednost takvog pristupa je u tome, što je moguće koristiti protokole mrežne razine (npr. IP Interneta) bez promjene, kao da se radi o stvarnoj klasičnoj lokalnoj mreži. Elementi LANE mreže prikazani su na slici 4.77:



Slika 4.77. LANE

LANE korisnik (ATM računalo) prijavljuje se LANE poslužitelju, te po potrebi traži pretvorbu mrežne adrese odredišta na ATM adresu.

LANE poslužitelj održava listu pretvorbe mrežnih adresa na ATM adrese i obavlja pretvorbu za potrebe korisnika.

BUS poslužitelj obavlja razdiobu PDU s univerzalnom adresom (broadcast) i rukuje s PDU čije je odredište nepoznato.

LANE konfiguracijski poslužitelj omogućava rad više LANE mreža na istoj ATM mreži.

LAN prospojnik povezuje ATM LANE s nekom drugom, konvencionalnom lokalnom mrežom.

Usmjernik povezuje LANE mreže s Internetom.

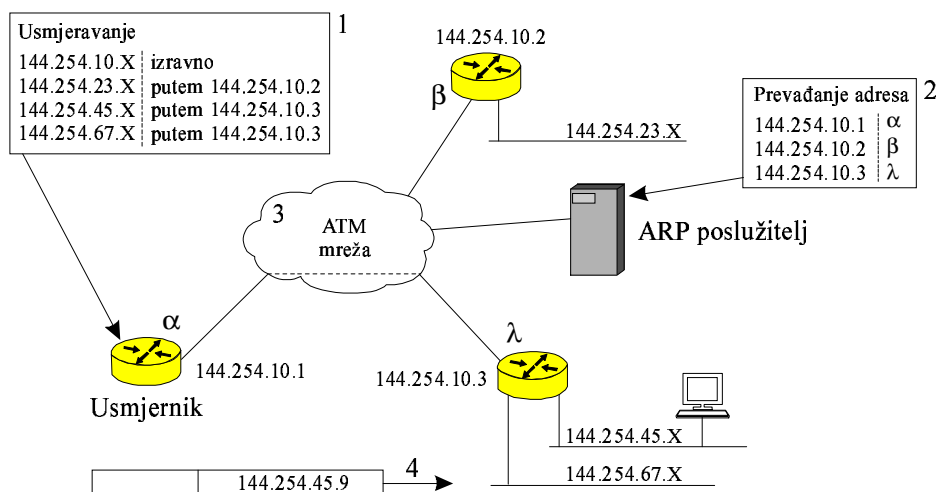
4.4.2.2 Klasični IP preko ATM mreže (CIPOA)

Klasični IP/ATM (CIPOA, Classical IP over ATM) je tehnologija omogućava prijenos IP paketa preko ATM mreže, uz korištenje specifičnog ATMARP (ATM Address Resolution Protocol) sustava. Na ovaj način je moguće koristiti ATM mrežu kao izravnu zamjenu za lokalne mreže ili kanale telekomunikacijskih mreža, slika 4.78.

CIPOA protokolom formira se logička IP podmreža (LIS, Logical IP Subnetwork). Na LIS, korisnik korištenjem ATMARP protokola od ARP poslužitelja traži prevođenje IP adrese na ATM adresu. Uloga ARP poslužitelja je slična ulozi LANE poslužitelja, samo što kod CIPOA mreže ne postoji mogućnost slanja PDU s univerzalnim adresama na podatkovnoj razini.

CIPOA omogućava prijenos raznih PDU mrežne razine korištenjem LLC/SNAP zaglavlja i AAL5 specifikacije. Time je ujedno ostvarena kompatibilnost s ostalim tehnologijama lokalnih i drugih mreža. Značajna je mogućnost multipleksiranja više tokova paketa jednim virtualnim kanalom.

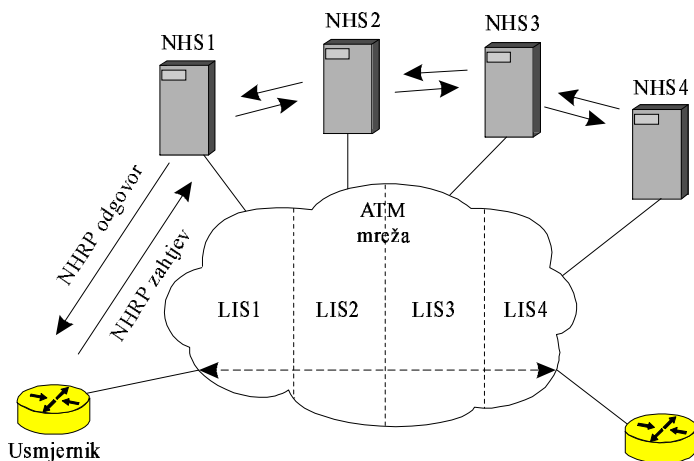
Mana CIPOA je u potrebi da se prijenos podataka između LIS obavlja posredstvom usmjernika, bez obzira da li su one možda dio iste ATM mreže. Prednosti direktne ATM veze se ne koriste.



Slika 4.78. CIPOA

4.4.2.3 Direktni IP preko ATM mreže (NHRP)

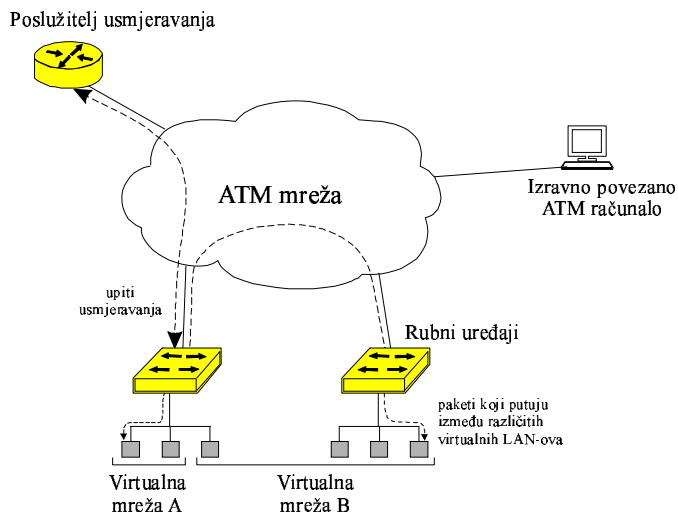
NHRP (Next Hop Resolution Protocol) je modifikacija IPOA na način da se veza najprije ostvari među usmjernicim (NHS Next Hop Server). Na taj se način dozna ATM adresa određiškog računala, te se ostvaruje direktni virtualni kanal između izvoriškog i određiškog ATM uređaja (računala ili usmjernika), slika 4.79.



Slika 4.79. NHRP

4.4.2.4 Više protokola preko ATM mreže (MPOA)

MPOA (Multi Protocol Over ATM) objedinjuje dobre strane ranijih tehnologija. MPOA povezuje virtualne podmreže različitih tehnologija ATM virtualnom kanalima, slika 4.80.



Slika 4.80. MPOA

4.5 MREŽE ZA PRIJENOS OKVIRA (FRAME-RELAY)

4.5.1 OPĆA SVOJSTVA MREŽA ZA PRIJENOS OKVIRA

Mreže za prijenos okvira (Frame Relay) razvile su se s namjenom povezivanja udaljenih lokalnih mreža radnih organizacija. Nakon inicijative zainteresiranih kompanija, standardizaciju je obavio ITU-T. Osnovne usluge definirane su preporukom I-233, a protokol preporukama Q-922 i Q-923.

Prije pojave mreža za prijenos okvira (oko 1985), povezivanje udaljenih mreža neke radne organizacije obavljalo se je modemskim prijenosom podataka između prenosnika preko iznajmljenih telefonskih kanala. Takve veze su bile relativno spore (do 9600 b/s) i skupe, jer su telekomunikacijske kompanije naplaćivale najam kanala bez obzira na količinu prenesenih podataka. Donošenje X.25 preporuke za paketne mreže nije bitno poboljšalo situaciju. Zbog kompleksnosti X.25 protokola i tadašnjeg stanja tehnike, brzine prijenosa su bile niske, a cijena visoka. Istovremeno, razvijao se uskopojasni ISDN, čija je funkcionalnost bila ograničena na komutirane kanale, dakle opet bez prednosti visokog iskorištenja i niske cijene koje donosi statističko multipleksiranje.

Potrebe korisnika za jeftinom i brzom mrežom koja će omogućiti povezivanje njihovih lokalnih mreža riješene su povezivanjem dobrih osobina ISDN pretplatničkog sučelja, prospajanja paketa i prosljeđivanja virtualnim kanalima. Nastale su mreže za prijenos okvira sa sojjevnim ili bespojnim prijenosom, koje zbog niske razine usluge možemo smatrati mrežama podatkovne razine, bez obzira što (kao i ATM mreže) obavljaju prijenos podataka s kraja na kraj mreže.

Na **fizičkoj razini**, koristi se osnovni ili primarni ISDN priključak, tako da korisniku na raspolaganju stoji D, B ili H (nxB) digitalni kanal. Prijenos podataka ovim kanalom (na sučelju korisnika i mreže) stalno je aktivan (za permanentne) ili se aktivira na zahtjev (za prosposjene virtualne kanale).

Na **podatkovnoj razini**, koristi se bitovno orijentirani protokol gotovo identičan HDLC protokolu, nazvan LAP-F (Link Access Protocol for Frame Relay). Predviđene su dvije vrste usluge, bespojni prijenos (prijenos okvira, Frame Relay) i spojivni prijenos (prospajanje okvira, Frame Switching). U oba slučaja koriste se prosposjeni ili permanentni virtualni kanali. Kod prosposjenih virtualnih kanala, korisnik zahtjeva uspostavu i raskid kanala. Kod permanentnih, korisnik se pretplaćuje na virtualni kanal, što znači da je kanal nakon aktiviranja stalno raspoloživ. Uspostavljanje prosposjenog ili permanentnog virtualnog kanala obavlja se unošenjem podataka u tablice usmjeravanja (automatizirano), tako da okviri LAP-F protokola putuju po uvijek istom, unaprijed poznatom putu.

4.5.2 FORMAT OKVIRA LAP-F PROTOKOLA

Specifikaciju LAP-F protokola možemo podijeliti na podrazine 2-1 i 2-2. Usluga prijenos okvira koristi samo podrazinu 2.1, dok usluga prospajanja okvira koristi obje podrazine (cjeloviti protokol).

Na **podrazini 2-1** nailazimo na standardni okvir bitovno orijentiranog protokola, koji se sastoji od okvirnog znaka (01111110), modificiranog adresnog polja, polja podrazine 2-2, zaštitnog znaka (standardna ciklička zaštita po polinomu CRC-CCITT) i ponovno okvirnog znaka, slika 4.81.

0 1 1 1 1 1 1 0	okvirni znak
2-4 okteta	adresno polje
varijabilan broj okteta	podrazina 2-2
CRC-CCITT	zaštitni znak
0 1 1 1 1 1 1 0	okvirni znak

Slika 4.81. LAP-F okvir prema Q-922

Jedina je razlika u adresnom polju, koje sadrži niz podataka potrebnih za prijenos okvira, slika 4.82.

8	8	7	6	5	4	3	2	1
osnovni format adrese (2 okteta)	Upper DLCI						C/R	EA=0
	Lower DLCI			FECN	BECN	DE	EA=1	
8	8	7	6	5	4	3	2	1
format adrese s 3 okteta	Upper DLCI						C/R	EA=0
	DLCI			FECN	BECN	DE	EA=0	
	Lower DLCI or DL-CORE control						D/C	EA=1
8	8	7	6	5	4	3	2	1
format adrese s 4 okteta	Upper DLCI						C/R	EA=0
	DLCI			FECN	BECN	DE	EA=0	
	DLCI							EA=0
	Lower DLCI or DL-CORE control						D/C	EA=1

Slika 4.82. Format adresnog polja LAP-F okvira

Pojedina polja adresnog polja imaju sljedeće značenje:

EA	(Address field extension bit), indicira kraj zaglavlja.
C/R	(Command response bit), određuje da li se radi o komandi ili odzivu
FECN	(Forward explicit congestion notification), indicira zagušenje preko prijemnika
BECN	(Backward explicit congestion notification) indicira zagušenje povratno
DLCI	(Data link connection identifier), identifikator virtulanog kanala
DE	(Discard eligibility indicator), indikator niskog prioriteta, okvir za odbacivanje
D/C	(DLCI or DL-CORE control indicator), indicira DLCI ili DL-CORE format
DL-CORE	(Data Link Core), kontrolne poruke za održavanje virtualnog kanala

Identifikator virtualnog kanala DLCI može biti duljine 10, 16 ili 23 bita. Neki identifikatori su rezervirani za potrebe signalizacije.

Korisnikova brzina slanja okvira određena je parametrom CIR (Committed Information Rate), koji je glavni parametar ugovora između korisnika i mreže. Mreža garantira da će, pod normalnim uvjetima, omogućiti prijenos podataka CIR brzinom. U slučaju podopterećenja mreže, dozvoljen je brži prijenos, ali je u slučaju zagušenja (smatra se nenormalnim uvjetima rada) korisnik dužan smanjiti brzinu predaje. Zagušenje se indicira FECN ili BECN bitom. Ukoliko korisnik ne poštuje naredbu o smanjenju brzine, njegovi se okviri označuju DE bitom, a takvi će se okviri u slučaju popunjenosti memorije usmjernika prvi odbaciti.

Podrazina 2-2 sadrži kontrolno C polje bitovno orijentiranog protokola (I, RR, RNR, REJ, SABME, DISC, XID, UA, DM, FRMR), te korisnikove podatke (samo I i UI okvir). Uočimo postojanje UI (Unnumbered Information) komande. Korisnikovi podaci najčešće sadrže cjeloviti okvir podatkovne razine lokalne mreže.

Usluga prijenosa okvira koristi samo UI okvir, te tako osigurava bespojni prijenos podataka korisnika. **Usluga prospajanja okvira** koristi asinkroni balansni mod s numeracijom PDU i retransmisijom, te osigurava spojevni prijenos podataka korisnika.

Mreže za prijenos okvira su naročito interesantne za banke i slične korisnike zbog relativne sigurnosti podataka koju pružaju korisnicima. Naime, kako svi okviri putuju permanentnim virtualnim kanalima, nije moguće neovlašteno prisluškivanje ili podmetanje podataka, kao ni pristup umreženim računalima (osim, naravno, neovlaštenim pristupom opremi telekomunikacijske kompanije, pa i tada samo uz znatne poteškoće).