Kazalo

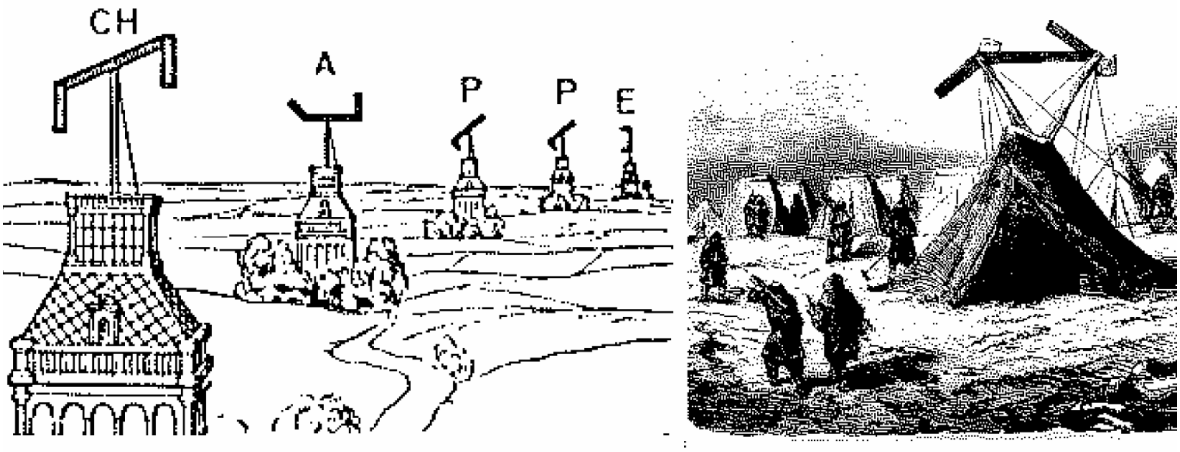
Kazalo **1**
5 Protokoli v sodobnih telekomunikacijah **2**
5.1 Splošno o protokolih 2
5.1.1 Začetki komunikacij na daljavo 2
5.1.2 Kaj je protokol 2
5.1.3 Storitve in protokoli 2
5.1.4 Protokolni sklad 5
5.2 Zanesljivost prenosa 9
5.2.1 Detekcija in popravljanje napak 9
5.3 Protokol IP 11
5.3.1 Splošno o IP protokolu 11
5.3.2 Pakiranje podatkov 11
5.3.3 Storitve, ki jih IP nudi 11
5.3.4 Zanesljivost IP protokola 12
3.1 Usmerjanje v IP 13
3.1.1 IGP usmerjevalni protokol 15
3.1.2 EGP usmerjevalni protokoli 21
5.4 ISDN 23
5.4.1 Uvod 23
3.2 Vrste dostopa 24
3.2.1 Osnovni dostop 24
3.2.2 Primarni dostop 25
3.3 ISDN referenčne oz. priključne točke 26
3.4 ISDN kanali 27
3.5 Signalizacijski D-kanal 28
3.5.1 Fizični sloj D kanala na S/T vodilu 28
3.5.2 Drugi nivo v protokolnem skladu – protokol LAPD 30
3.5.3 Mrežni nivo (Q.931) 34
3.5.4 Primer dejanskih okvirjev 35

5 Protokoli v sodobnih telekomunikacijah

5.1 Splošno o protokolih

5.1.1 Začetki komunikacij na daljavo

Že v zgodnjih časih razvoja, je človeštvo potrebovalo komunikacijo na daljavo. V ta namen so skupine, ki so želele komunicirati morale skleniti nek dogovor o načinu in pomenu sporazumevanja. Tako so se že v 5. st. pr.n.št. sporazumevali s kurjenjem kresov, bakel, ognja, ipd. Med 18. in 19. stoletjem našega štetja, se je pojavil prvi optični telegraf, kjer so s pomočjo položaja ročic prenašali sporočila.



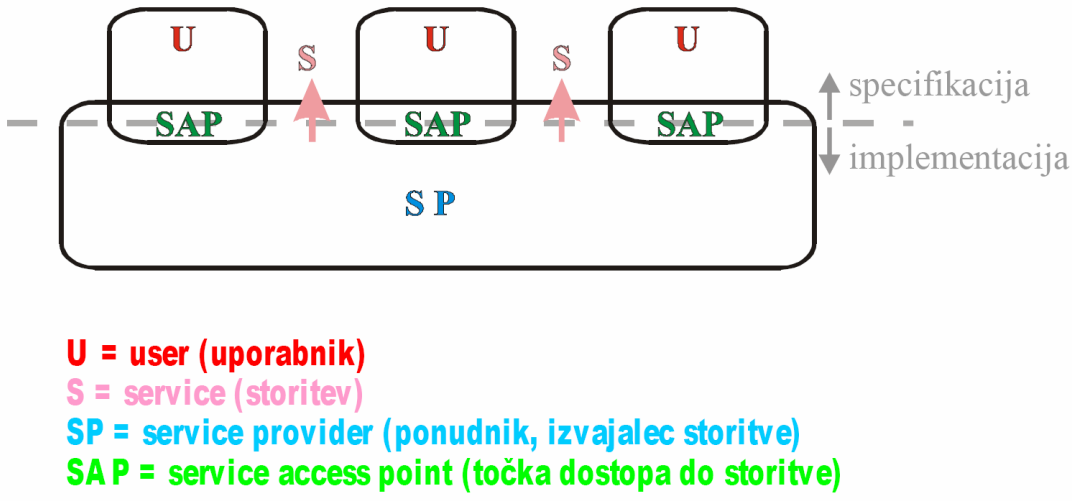
Slika 1. Omrežje telegrafskih naprav.

5.1.2 Kaj je protokol

Protokol izhaja iz grških besed protos (prvi) in kollan(lepiti). Protokol je v začetku pomenil podatke na prvi strani uradnih listin, danes pa je to širok pojem, ki v diplomatskem smislu predstavlja pravila za izvajanje diplomatskih postopkov ali uradni zapisnik o sklepih konference. V svetu telekomunikacij je to jezik za sporazumevanje med stroji.

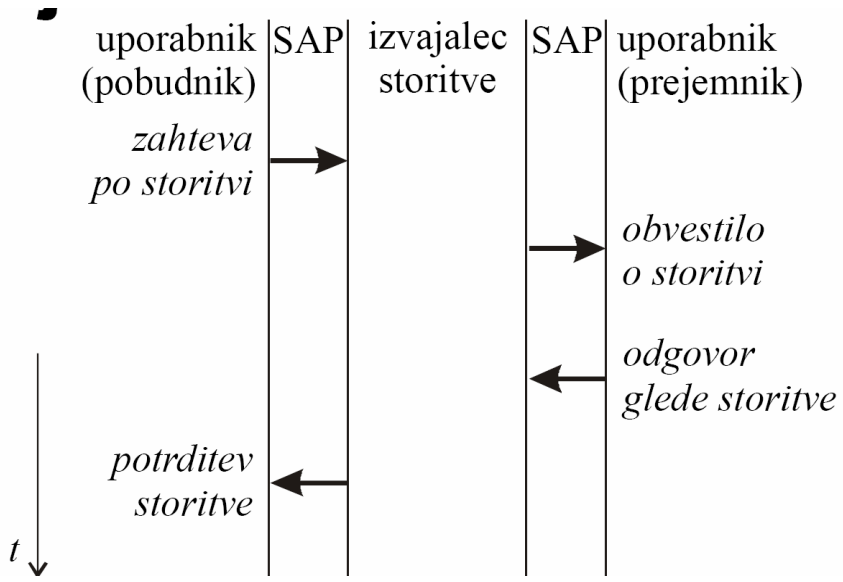
5.1.3 Storitve in protokoli

V telekomunikacijskih sistemih je glavni problem to, da sta sistem in informacija porazdeljena, noben komunikacijski osebek pa ne pozna stanja celotnega sistema. Protokoli skušajo v čim večji meri odpraviti te pomanjkljivosti.



Slika 2. Koncept storitve.

Osnovni namen komunikacije na daljavo je interakcija med dvema ali več uporabniki neke storitve, ki jo ponuja izvajalec storitve, oz. ponudnik. Uporabnik torej mora preko nekih mehanizmov dostopati do storitev, ki jih ponuja izvajalec. Mesto, kjer uporabnik komunicira s ponudnikom storitve imenujemo SAP (Service Access Point).
Vsak uporabnik mora izvajalcu storitve sporočiti, da želi koristiti storitev. Uporabnik (uporabnik 1) ponudniku torej pošlje zahtevo po storitvi (Request). Ponudnik storitve uporabniku (uporabnik 2), s katerim želi komunicirati uporabnik 1 pošlje obvestilo o storitvi (Indication). Uporabnik 2 poda odgovor glede storitve (Response), izvajalec storitve pa uporabniku 1 poda potrditev storitve (Confirmation).
Request, Indication, Response* in *Confirmation so temeljni tipi interakcije med uporabnikom in izvajalcem **(primitivi)**.



Slika 3. Osnovni tipi interakcije med uporabnikom in ponudnikom.

Storitev je zahteva uporabnikov in ponudba operaterja, protokol pa je širok pojem, ki definira naslednje:

- Implementacija storitve
- Nabor sporočil
- Format sporočil
- Pravila za pošiljanje sporočil

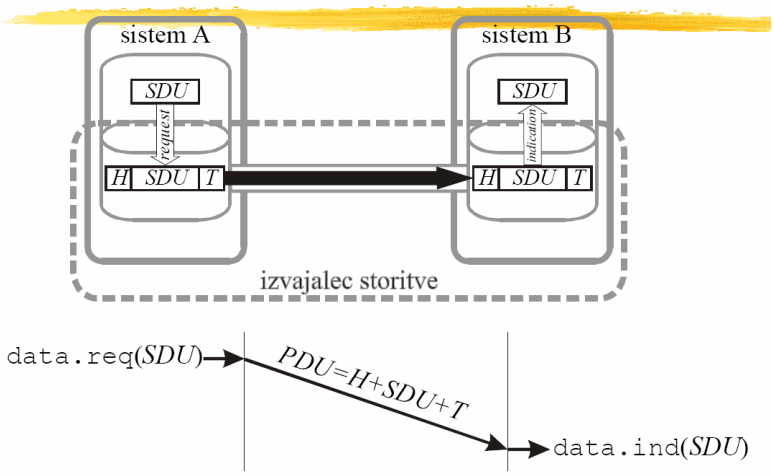
Vsako protokolno sporočilo je v splošnem sestavljeno iz *uporabniškega sporočila*, t.j. informacije, ki jo želi prenesti iz enega na drugo mesto in *nadzorne protokolne informacije*.

Celotno sporočilo imenujemo Protokolna podatkovna enota (Protocol Data Unit) in jo označimo s PDU. Uporabniško informacijo označimo kot Storitvena podatkovna enota (Service Data Unit) ali SDU. Nadzorno protokolno informacijo pa imenujemo PCI (Protocol Control information).

Tako lahko protokolno podatkovno enoto zapišemo kot

$$PDU=SDU+PCI$$

Pravimo, da uporabno sporočilo ovijemo v nadzorna sporočila. To imenujemo inkapsulacija.



Slika 4. Osnovni tipi interakcije med uporabnikom in ponudnikom.

Vsak napisan in potrjen protokol mora ustrezati naslednjim kriterijem:

- Logična pravilnost
- Popolnost, nedvoumnost, skladnost specifikacije
- Pravičnost
- Robustnost, konvergenca
- Učinkovitost
- Standardiziranost

Predvsem zaradi standardiziranosti so se v svetu pojavile različne organizacije, ki skušajo skrbeti, da so množično uporabljeni protokoli standardizirani in veljajo povsod po svetu. V nekaterih primerih jim to uspeva, spet v drugih pa je to lahko zelo težavno.

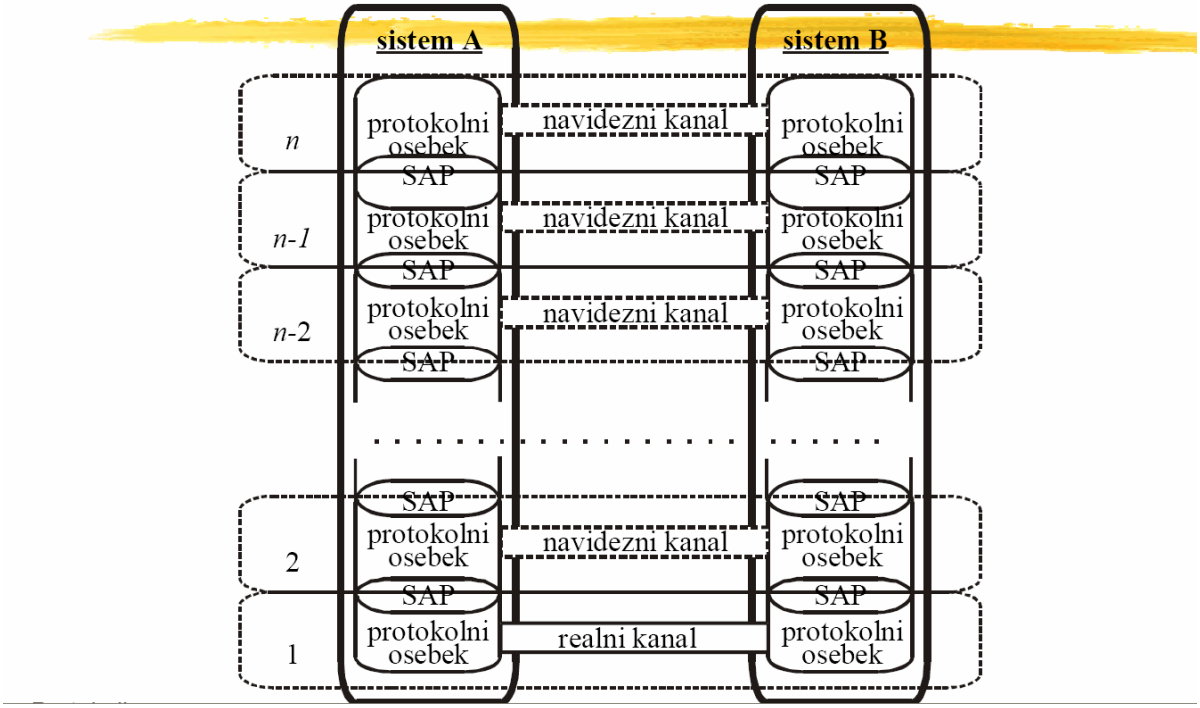
Organizacije lahko razdelimo na tri osnovne tipe:

- Mednarodne (ISO, ITU, ETSI)
- Strokovna združenja (IEEE, W3C, ATM Forum)
- Nacionalni (SIST, ANSI, DIN)

5.1.4 Protokolni sklad

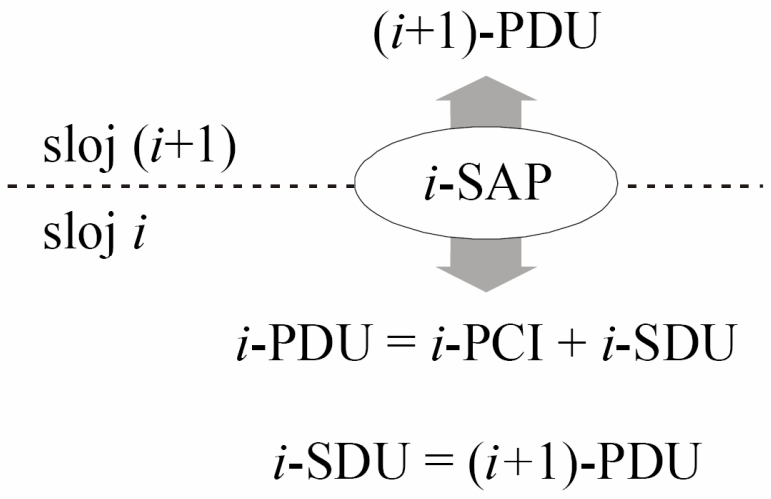
V moderni dobi komunikacij, je razvoj tako hiter, kompleksnost storitev pa narašča z bliskovito hitrostjo, zato je sistem nemogoče zasnovati tako, da bi bil definiran en sam protokol, ki bi skrbel za vse vrste problemov, ki se na komunikacijskih vodih dogajajo, hkrati pa še skrbel za pravilen prenos podatkov in hkrati komuniciral transparentno z uporabnikom.

Zato je bil definiran protokolni sklad, ki je razdeljen na sloje (Layers). Vsak sloj komunicira s svojim partnerjem istega sloja preko navideznih kanalov, posamezni nivoji pa med seboj komunicirajo preko SAP (Service Access point). Tako je vsak sloj svoj protokolni osebek in skrbi za specializirano področje, ki mu je dodeljeno.



Slika 5. Protokolni sklad.

Vsak sloj sporočilo prejšnjega sloja inkapsulira, torej mu doda nek svoj PCI. Razmere prikazuje slika 6:



Slika 6. Terminologija v slojih protokolnega sklada.

ISO je definiral referenčni model, ki bi naj bil osnova za vse sodobne protokole. Vsak realen protokol uporablja nekoliko prirejen ISO OSI model, kjer lahko posamezne nivoje bodisi še podrobneje razčleni ali pa nekatere združi. V spodnji tabeli so navedeni posamezni sloji

7	A	Application layer	Aplikacijski sloj
6	P	Presentation layer	Prikazni (prezentacijski) sloj
5	S	Session layer	Sejni (Konferenčni) sloj
4	T	Transport layer	Transportni sloj
3	N	Network layer	Omrežni sloj
2	DL	Data-Link layer	Kanalni (Povezavni) sloj
1	Phy	Physical layer	Fizični sloj

Fizični sloj definira vse fizične in električne lastnosti kanala, kar vključuje specifikacijo pinov, kablov, napetosti, kodiranja,…

Povezavni sloj Skrbi za prenos podatkov med omrežnimi entitetami. Popravlja napake, ki so se zgodili v fizičnem sloju.

Omrežni sloj zagotavlja prenos podatkov preko enega ali več omrežij z upoštevanjem kakovostnih meril, ki jih zahteva transportni sloj. Operira z logiko usmerjanja. Na tem nivoju delujejo routerji, switchi,…

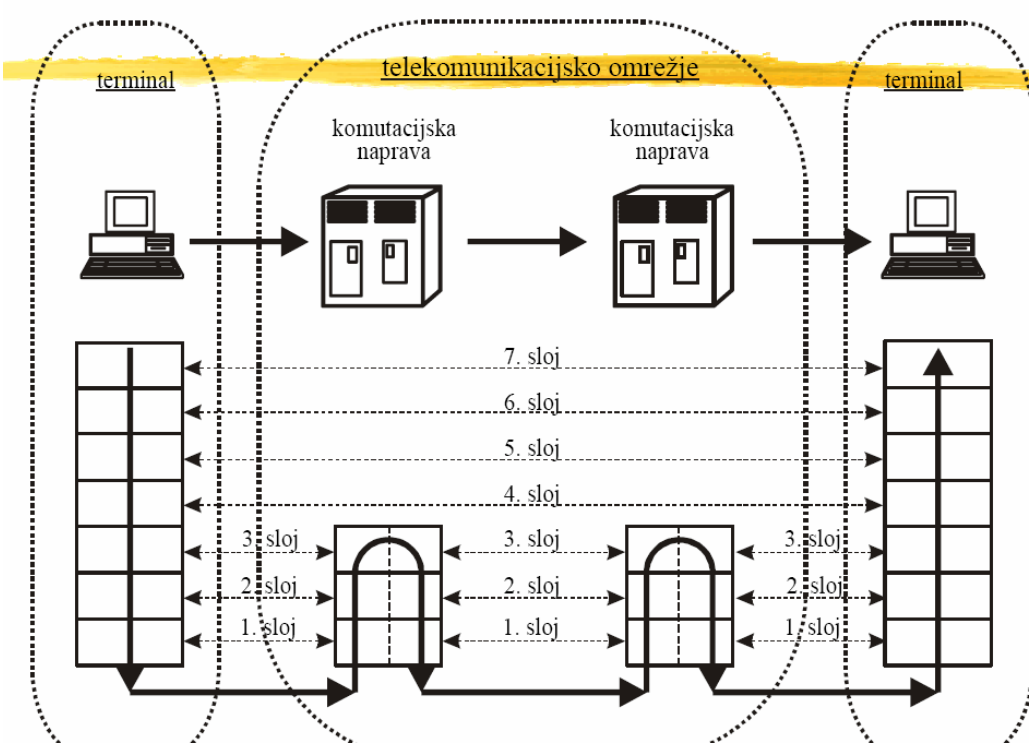
Transportni sloj zagotavlja transparentni prenos podatkov med končnimi uporabniki. Poskrbi, da višjim nivojem ni potrebno skrbeti za zanesljiv prenos. Najbolj znana sta TCP in UDP.

Sejni sloj nadzira dialoge med računalniki. Vzpostavlja, vzdržuje in

ruši povezave med lokalno in oddaljeno aplikacijo.

Prikazni sloj pretvori podatke tako, da ustrezajo aplikacijskemu sloju. Primer je kompresija, kriptiranje,…

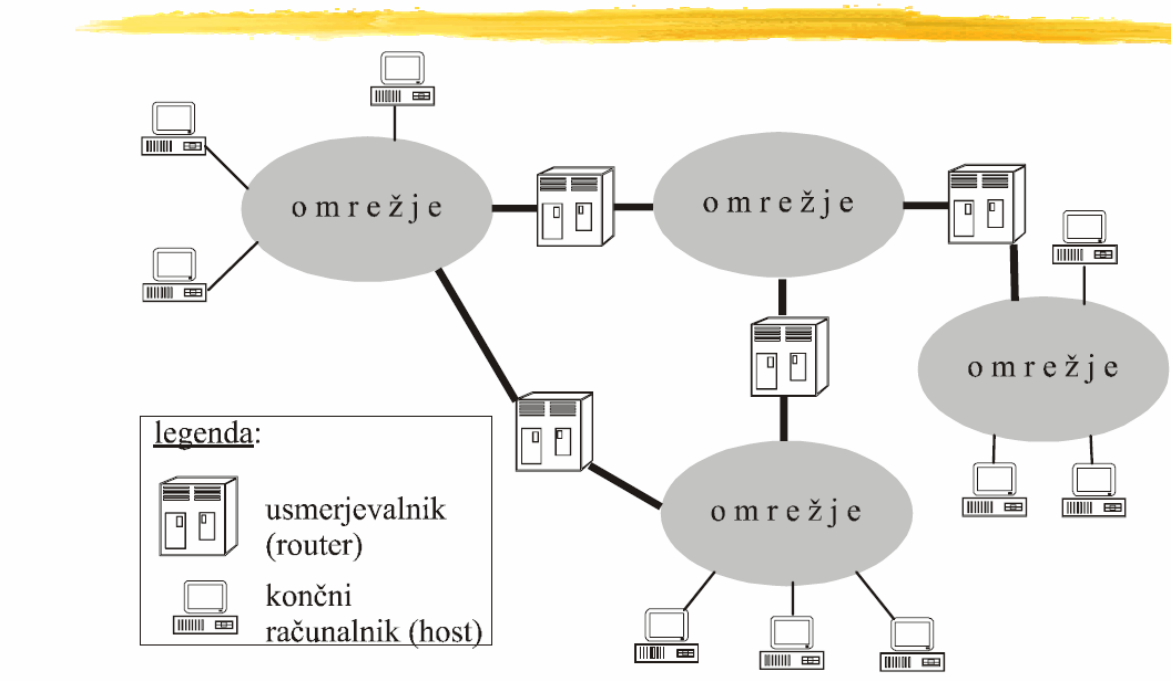
Aplikacijski sloj omogoča uporabniku dostop do informacije na omrežju. Primeri so Telnet, SMTP, FTP,…



Slika 7. Prikaz poteka komunikacije v modelu ISO OSI.

Slika 7 prikazuje komunikacijo preko nekega kanala. Ker se komutacija dogaja v 3 sloju, se višji nivoji ne ukvarjajo s tem, da gre komunikacija preko dveh komutacijskih naprav, saj zato transparentno poskrbijo nižji nivoji.

Internetno omrežje je kompleksno, sestavljeno iz mnogo manjših omrežij, usmerjevalnikov, računalnikov, ipd. V praksi se je tukaj uveljavil prirejen protokolni sklad, ki ima nekatere nivoje združene. Gre za Protokolni sklad TCP/IP.



Slika 8. Topologija interneta.

Protokolni sklad TCP/IP je sestavljen iz štirih slojev. Sloji TCP/IP so prikazani v spodnji tabeli

4	Application layer	Aplikacijski sloj
3	Transport layer	Transportni sloj
2	Internet layer	Internetni
1	Network access layer	Sloj dostopa do omrežja

5.2 Zanesljivost prenosa

Noben kanal ni idealen. To smo že spoznali v prejšnjih poglavjih. Med prenosom podatkov lahko prihaja do napačno sprejetih podatkov, izgube podatkov, dupliciranih podatkov, spremeni se lahko vrstni red sporočil in nenazadnje lahko v kanalu ostanejo neki paketi, ki jih ne potrebuje noben naslovnik več. Takšne pakete imenujemo »mrtve duše«.

Naloga protokolov je, da napake detektirajo, če je le mogoče popravljajo, nadomeščajo izgube, odkrivajo in odstranjujejo duplikate, zagotavljajo ohranitev vrstnega reda sporočil in odstranjujejo mrtve duše.

Če je le mogoče, skušamo napake tudi čimbolj minimizirati. To lahko med drugim dosežemo tudi s prepletanjem. Sporočila med sabo premešamo po določenem ključu. S tem sicer pokvarimo več sporočil, vendar pa so napake bistveno manj intenzivne.

Nekatere izmed postopkov si bomo ogledali v tem poglavju.

5.2.1 Detekcija in popravljanje napak

Ker uporabniki želijo transparenten prenos podatkov in jih zanima samo prenesena vsebina, mora protokol poskrbeti, da so podatki, ki jih je oddajnik sprejel, čimbolj verodostojni. Preden lahko protokol ukrepa, mora vedeti, kje je do napake sploh prišlo, oziroma katero sporočilo je sploh napačno, nato pa se odločiti, kako ukrepati. Protokol mora napako torej detektirati.
Napako najlažje detektiramo tako, da sporočilu dodamo nekaj redundantnih bitov, s katerimi jo lahko detektiramo. V splošnem so ti biti izračunani po:

- Paritetni metodi
- Metodi seštevanja
- Metodi krožnega redundančnega preizkusa

Ko katerakoli stran detektira napako, je le-to možno v nekaterih primerih popraviti, v nekaterih pa je potrebno sporočilo ponovno poslati. Obstajata dve osnovni metodi, ki ju seveda lahko kombiniramo. To sta

- FEC – Forward Error Correction (popravljanje na sprejemni strani)
- BEC – Backward Error Correction (popravljanje s ponovnim pošiljanjem)

Kakšen način uporabimo kje je seveda vprašanje same zahtevnosti in značilnosti protokola. V splošnem velja, da se BEC uporablja pri asinhronih prenosih in manjših pogostostih napak. BEC uporablja tudi manj režije. FEC se ponavadi uporablja pri sinhronih prenosih pri večjih

pogostostih napak. Zahteva več režije.
Obe metodi je možno kombinirati.
Kot zgled si bomo ogledali prenos govora v GSM z sledečimi parametri:

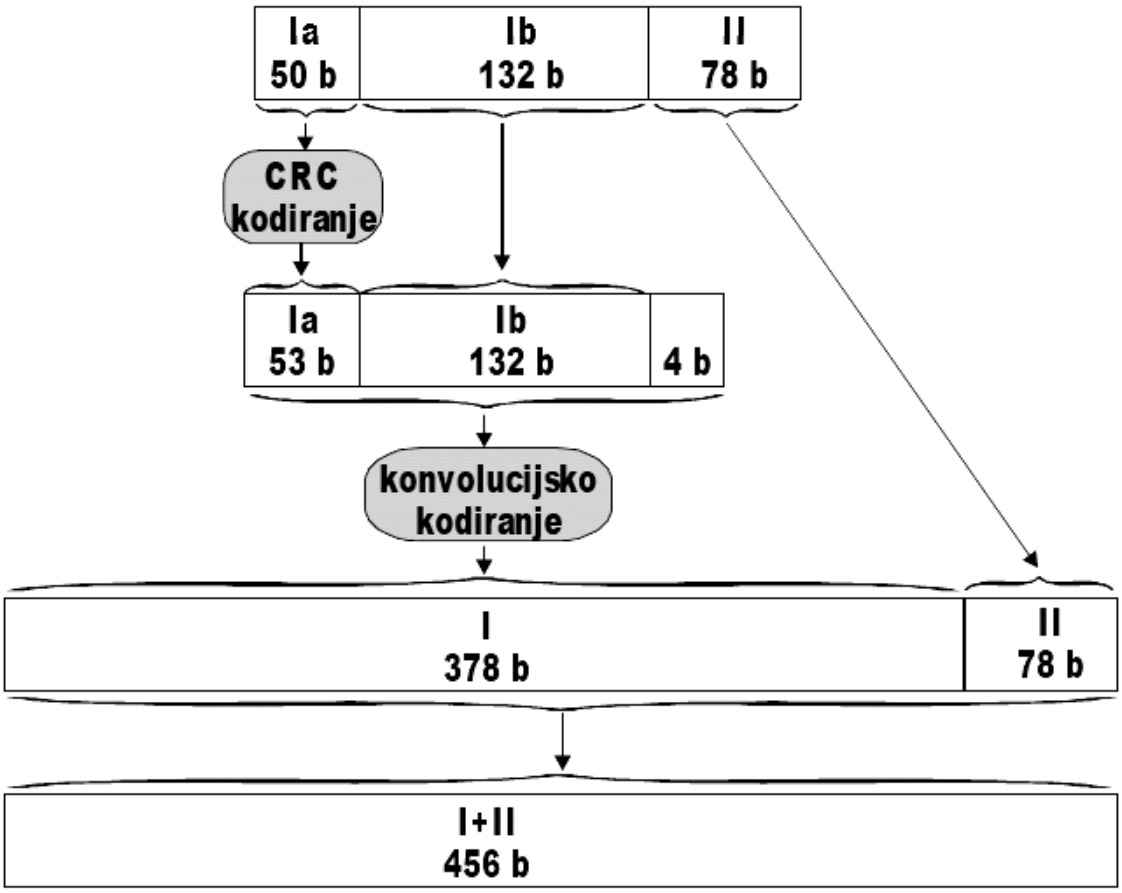
- Trajanje skupine vzorcev: 20ms
- Dolžina skupine vzorcev: 260 b
- Hitrost prenosa govora: 13 kb/s
- Število redundantnih bitov na blok: 456-260=196 b (43%)

Torej imam dolžino bloka 260 b. V njem imamo tri razrede pomembnosti bitov v bloku. Bolj kot je bit pomemben, bolj vpliva na kvaliteto prenesenega govora.

Ia (50 b): popravimo ali zavržemo

Ib (132 b): popravimo po najboljših močeh

II (78 b): ne zaščitimo



Slika 9. Kodiranje govora.

5.3 Protokol IP

5.3.1 Splošno o IP protokolu

IP – Internet Protokol je nepovezavno orientiran protokol, namenjen komunikaciji v omrežjih s paketno komutacijo. Je protokol tretjega, omrežnega nivoja in je inkapsuliran v protokolu povezovalnega sloja, npr. Ethernet.

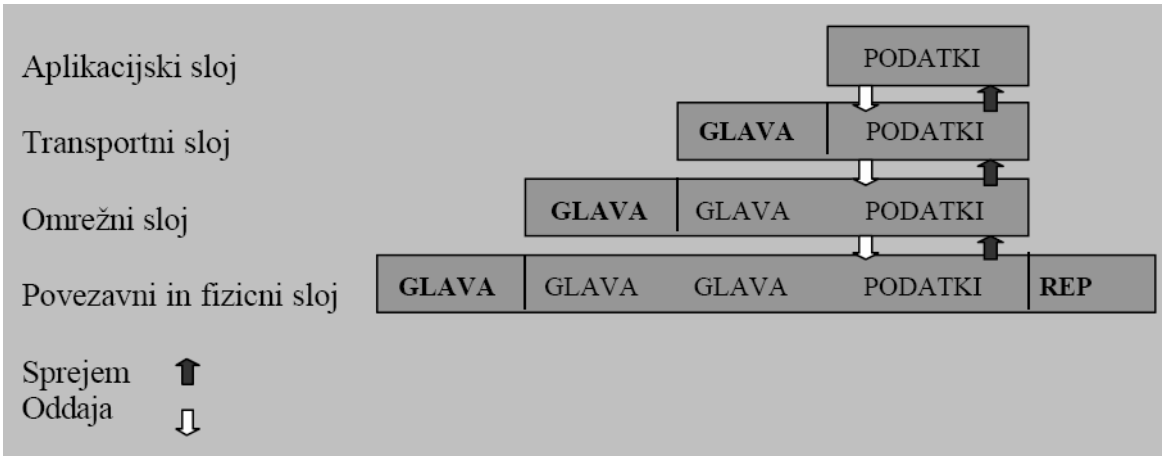
Kot protokol nižjega sloja omogoča storitev komunikacije znotraj sistema globalnega naslavljanja v omrežju interneta.

5.3.2 Pakiranje podatkov

Podatki protokola višjega sloja so inkapsulirani v enega ali več paketov oz. datagramov. Termin paket in datagram sta v okolju IP sinonima. Pred pošiljanjem paketa ni potrebno vzpostavljati nobene povezave, zato uvrščamo protokol IP med nepovezavno orientirane protokole. Še enkrat se spomnimo javnega telefonskega omrežja, kjer je potrebno pred prenosom podatkov vzpostaviti zvezo in tukaj govorimo o povezavno orientiranem protokolu.

5.3.3 Storitve, ki jih IP nudi

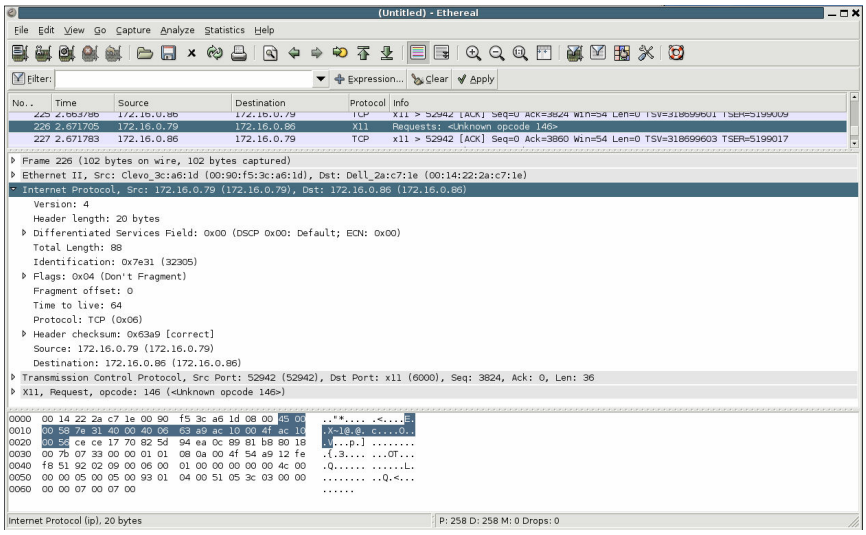
Zaradi visokega nivoja abstrakcije, ki jo nudi inkapculacija, lahko protokol IP uporabljamo tudi v precej heterogenih omrežjih, npr. kombinacijah Etherneta, ATM-a, FDDI, Wi-Fi, token ring, itd. Heterogenost spodaj ležečih omrežin ne vpliva na protokole zgornjih slojev.



Slika 10. Inkapsulacija uporabnikovih podatkov v UDP datagramu znotraj IP paketa.

Protokoli povezovalnega sloja imajo svoje načine naslavljanja in skrbijo

tudi za preslikavo IP naslovov v naslove povezovalnega sloja. Za to skrbi protokol ARP (Address Resolution Protocol).



Slika 11. Primer glave IP datagrama ujet s programom Ethereal.

5.3.4 Zanesljivost IP protokola

IP protokol ne zagotavlja zanesljive dostave paketa. Deluje po principu *best-effort*., a brez kakršnekoli garancije, da bo paket res dostavljen, da bodo podatki nepoškodovani, da bodo prišli v istem vrstnem redu kot so bili odposlani in da ne bo na cilj prišel kakšen paket podvojen.

Kar se tiče zanesljivosti, protokol IP zagotavlja le kontrolno vsoto za glavo paketa. Napaka v kontrolni vsoti pa povzroči opustitev paketa, nima pa mehanizma, da bi ta paket ponovno oddal.

Za reševanje navedenih problemov morajo torej poskrbeti protokoli višjih slojev.

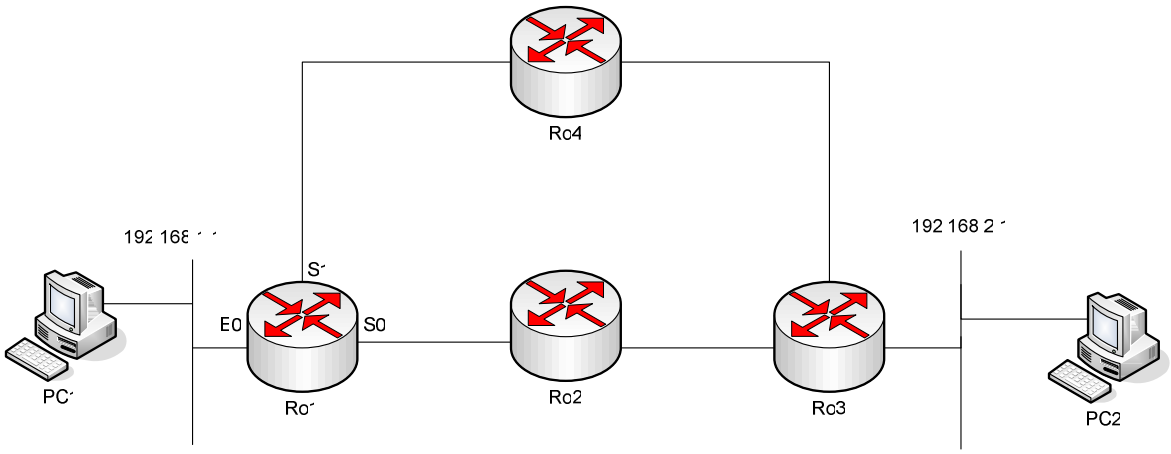
3.1 Usmerjanje v IP

Usmerjanje je postopek kako poslati podatek na željen naslov izven lokalnega omrežja. Usmerjevalnik zbira in vzdržuje informacije o topologiji omrežja in na podlagi nje usmerja poslane pakete na izbrane naslove. V konceptu ima vsak usmerjevalnik tabelo vseh možnih naslovov, ki jih doseže in postopek kako priti do tja. To tabelo lahko ročno (statično) napiše omrežni administrator, ali pa usmerjevalnik uporabi razne usmerjevalne protokole, ki na podlagi algoritma izračunajo in vzdržujejo (dinamično) to tabelo, glede na spremembe v topologiji omrežja.

Za uspešno usmerjanje mora usmerjevalnik storiti naslednje:

- prepoznati ciljni naslov
- prepoznati vire za pridobitev usmerjevalnih informacij
- Prepoznati poti omrežja
- Določiti pravilno pot

Če je usmerjevalnik na napravo vezan neposredno, brez ovir posreduje paket, saj pozna ciljni naslov. Če pa mora do ciljnega naslova prečkati več usmerjevalnikov, si mora usmerjevalnik izračunati najboljšo pot in zgraditi usmerjevalno tabelo.



Protokol	Ciljno omrežje	Izhodni priključek
Priključen RIP EIGRP	192.168.1.1 192.168.2.1 192.168.2.1	EO S1 S0

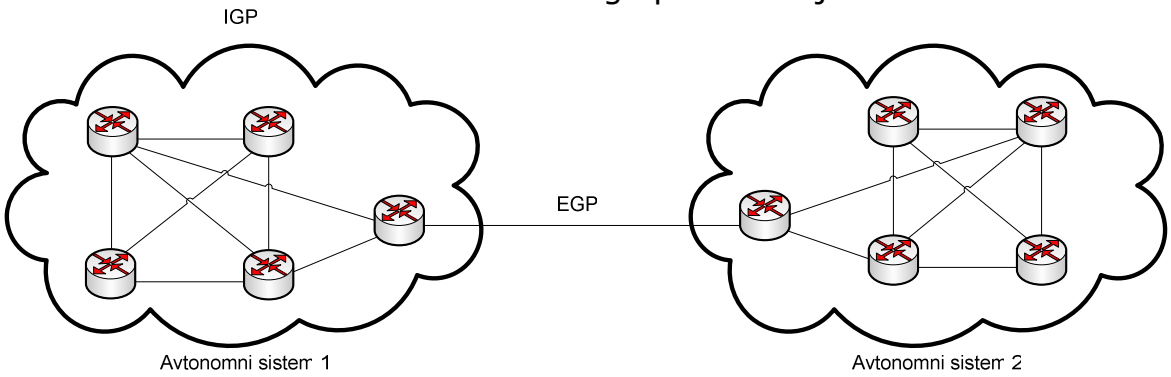
Slika 12. Omrežje.

Usmerjanje lahko poteka statično in dinamično. Pri statičnem usmerjanju administrator vpiše fiksen naslov v usmerjevalno tabelo in usmerjevalnik da prednost temu. Najpogostejša uporaba statičnega usmerjanja je pri omrežjih, kjer se navzven vežejo preko enega izhoda. Primer tega je povezava manjšega lokalnega omrežja v

svetovni splet preko ponudnika spletnih storitev (ISP). Dinamično usmerjanje pa je postopek, kjer usmerjevalnik uporablja protokol, s katerim sam izračuna najboljšo pot do cilja.

Obstajata dva tipa usmerjevalnih protokolov glede na avtonomnost omrežja:

- Protokol notranjih usmerjevalnikov (angl. IGPs - Interior Gateway Protocols): Ti se uporabljajo za izmenjavo usmerjevalnih informacij znotraj avtonomnega sistema. Primeri teh protokolov so: RIPv1, RIPv2, IGRP, EIGRP in OSPF
- Protokol zunanjih usmerjevalnikov (angl. EGPs - Exterior Gateway Protocols): Ti se uporabljajo za povezavo med avtonomnimi sistemi. Primer tega protokola je BGP.



Slika 13. Povezava avtonomnih sistemov.

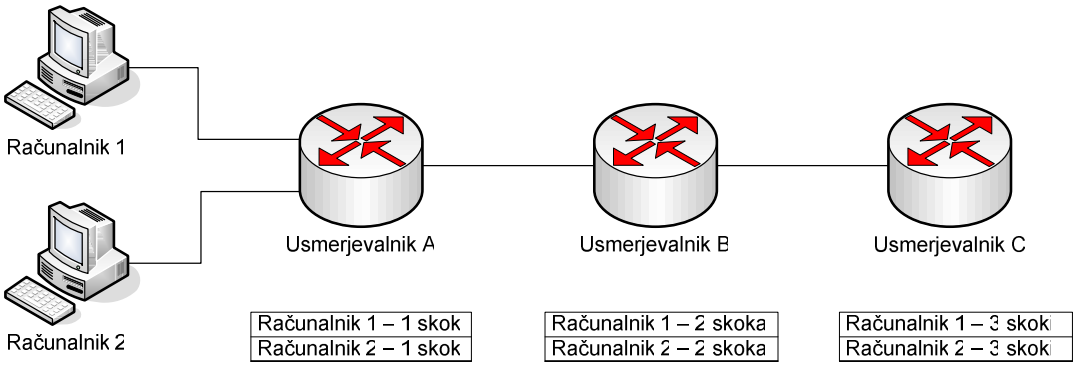
3.1.1 IGP usmerjevalni protokol

Znotraj posameznega avtonomnega sistema se IGP usmerjevalni protokoli delijo na tri tipe delovanja usmerjevalnega algoritma:

- **Distance vector:** Ta algoritem izračuna pot glede na smer (angl. vector) in dolžino poti (angl. distance), za katero se šteje število usmerjevalnikov, ki jih podatek prečka (angl. hops).
- **Link state**: Tudi znan kot »algoritem prednosti najkrajše poti« (angl. SPF - Shortest Path First), si ustvari natančno topologijo celotnega podomrežja v katerem se nahaja.
- **Balanced hybrid:** Ta algoritem pa združuje lastnosti link state in distance vector algoritmov.

Distance vector

Vsak distance vector usmerjevalnik pošlje svojo usmerjevalno tabelo, ali vsaj del nje, svojemu sosedu. Ta proces je znan tudi kot usmerjanje na podlagi govoric (angl. routing by rumor), saj usmerjevalnik sestavi sliko omrežja na podlagi sosedove perspektive nanjo. Za razliko od link state osveževanja, kjer pošiljajo manjše tabele po celem omrežju, ta pošlje celotno tabelo, ampak le svojemu sosedu. Predstavniki distance vector protokolov so RIPv1, RIPv2 in IGRP.



Slika 14. Primer usmerjanja distance vector protokola.

Usmerjevalnik A si zgradi usmerjevalno tabelo glede na naprave priključene nanj in jo pošlje svojemu sosedu, usmerjevalniku B. Ta jih doda v svojo tabelo, pri čemer preračuna skoke. Nato usmerjevalnik B pošlje svojo tabelo sosednjemu usmerjevalniku C, itd.

Kadar obstaja več možnih poti do cilja, je primarna naloga usmerjevalnega algoritma izbrati najboljšo pot iz tabele. Vsak distance vector usmerjevalni protokol uporablja drugačno matematiko za izračun najboljše poti. Algoritem ustvari številko, ceno poti (angl. cost), za vsako pot skozi omrežje. Manjša kot je številka, boljša je pot. Če usmerjevalnik uporablja več protokolov, se uporablja administrativna distanca. To je število med 0 in 255, ki je dodeljeno vsakemu protokolu. To je nekakšna ocena protokola, in boljši kot je, nižjo administrativno distanco ima. Uporablja pa se v primeru, ko se mora usmerjevalnik odločiti med dvema protokoloma in tu da prednost protokolu z nižjo administrativno distanco. Statično podana usmeritev ima prednost pred vsemi, zato ima administrativno distanco 1.

Vrsta protokola	Določena admin. distanca
Direktna povezava	0
Statičen usmerjeni naslov	1
EIGRP	90
IGRP	100
OSPF	110
RIPv1, RIPv2	120
Zunanji EIGRP	170
Neznan ali nemogoč	255, ne bo posredoval podatka

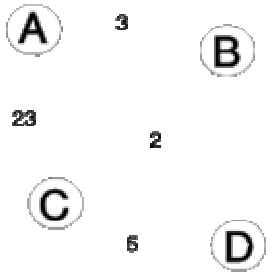
Kompleksnejši algoritmi za izračun najboljše poti upoštevajo več karakteristik poti, kot so:

- **Število skokov:** nam pove čez koliko usmerjevalnikov mora podatek potovati
- **Pasovna širina:** je podatkovna hitrost linije. Primer: 10-Mbps Ethernet linija je sprejemljivejša kot 64-kbps linija
- **Zakasnitev:** čas, ki ga vsak usmerjevalnik porabi za procesiranje
- **Obremenjenost omrežja:** količina aktivnosti na usmerjevalniku ali liniji.
- **Zanesljivost:** (angl. bit error rate), izražena za vsako linijo
- **Največja velikost paketa:** (angl. MTU – maximum transmission unit), izražena v oktetih, in je sprejemljiva za prenos po celotni liniji od izvora do cilja.

RIP in IGRP sta distance vector usmerjevalna protokola z razliko, da RIP uporablja število skokov za izračun, medtem ko IGRP uporablja naprednejša merila kot sta pasovna širina in zakasnitev že v osnovi. Dodatno lahko uporabi še ostale.

<u>Primer</u>:

V omrežju imamo 4 usmerjevalnike:



Slika 15. Primer topologije omrežja.

Začnemo s kreiranjem distančnih matrik vsakega usmerjevalnika.

Najkrajša pot je obarvana zeleno, nova najkrajša pot je obarvana rumeno.

T=0

from A	via A	via B	via C	via D
to A				
to B		3		
to C			23	
to D				

from B	via A	via B	via C	via D
to A	3			
to B				
to C			2	
to D				

from C	via A	via B	via C	via D
to A	23			
to B		2		
to C				
to D				5

from D	via A	via B	via C	via D
to A				
to B				
to C			5	
to D				

T=1

from A	via A	via B	via C	via D
to A				
to B		3	25	
to C		5	23	
to D			28	

from B	via A	via B	via C	via D
to A	3		25	
to B				
to C	26		2	
to D			7	

from C	via A	via B	via C	via D
to A	23	5		
to B	26	2		
to C				
to D				5

from D	via A	via B	via C	via D
to A			28	
to B			7	
to C			5	
to D				

T=2

from A	via A	via B	via C	via D
to A				
to B		3	25	
to C		5	23	
to D		10	28	

from B	via A	via B	via C	via D
to A	3		7	
to B				
to C	8		2	
to D	31		7	

from C	via A	via B	via C	via D
to A	23	5		33
to B	26	2		12
to C				
to D	51	9		5

from D	via A	via B	via C	via D
to A			10	
to B			7	
to C			5	
to D				

T=3

from A	via A	via B	via C	via D
to A				
to B		3	25	
to C		5	23	
to D		10	28	

from B	via A	via B	via C	via D
to A	3		7	
to B				
to C	8		2	
to D	13		7	

from C	via A	via B	via C	via D
to A	23	5		15
to B	26	2		12
to C				
to D	33	9		5

from D	via A	via B	via C	via D
to A			10	
to B			7	
to C			5	
to D				

Slika 16. Primer usmerjanja distance vector protokola.

Vzdrževanje usmerjevalnih informacij je bistveno za zaznavanje sprememb v topologiji omrežja in poteka korak za korakom od usmerjevalnika do usmerjevalnika. Osveževanje pri distance vector protokolih poteka tako, da vsak posamezen usmerjevalnik odda svojo usmerjevalno tabelo svojim sosedom. To oddaja periodično v določenih intervalih, lahko pa se pošlje takoj, ko usmerjevalnik zazna spremembo v topologiji omrežja. Za to uporablja prožilni časovnik (angl. trigger update). Ta ima nalogo, da ob zaznani spremembi topologije takoj pošlje usmerjevalno tabelo sosedom. Pri tem se sporočilo o napaki v omrežju hitreje razpošlje.
Ko usmerjevalnik sprejme sosedovo tabelo, jo primerja z lastno. Po algoritmu za izračun najboljše poti si preračuna vsako pot in jo primerja z potjo v svoji tabeli. Če usmerjevalnik izračuna boljšo pot,

kot ima zapisano v tabeli, posodobi svojo z novim podatkom. Vsaka pot v tabeli vsebuje informacije o kakovosti poti in logičnega naslova prvega usmerjevalnika na tej poti.

Napake distance vector usmerjevalnih protokolov

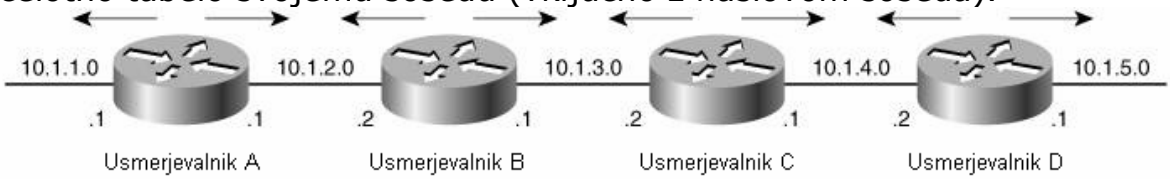
Distance vector usmerjevalni protokoli vsebujejo nekaj ovir in omejitev, ki so rešene na različne načine. Navedeni so trije primeri problematike z opisano rešitvijo.

Časovnik izpada (angl. Invalidation timer)

Če se prekine pot za usmerjevalnikom, ta prepozna problem, in v usmerjevalni tabeli označi to pot kot nedosegljivo, nato pa razpošlje informacijo naprej. Torej tu ni ovire. Problem nastane, če pride do izpada usmerjevalnika, saj v tem primeru sosednji usmerjevalnik tega ne zazna in pošilja pakete na njegov naslov, kjer se izgubijo. Za to oviro poskrbi interni časovnik, ki odšteva čas od zadnje osvežitve posamezne poti. Ko usmerjevalnik dobi osveženo usmerjevalno tabelo, obnovi vse časovnike na njihovo polno vrednost, ti odštevajo čas ki je definiran kot časovna dolžina treh do šestih posodobitev. Če v tem času usmerjevalnik ne dobi posodobitve, ne obnovi časovnika in avtomatsko izključi naslov iz usmerjevalne tabele. To se imenuje časovnik izpada (angl. invalidation timer).

Split horizon

Naslednja težava nastopi pri usmerjevalnih tabelah ki se razpošiljajo med sosednjimi usmerjevalniki. Kot je znano, usmerjevalnik pošlje celotno tabelo svojemu sosedu (vključno z naslovom soseda).



POT	SKOZI	SKOKOV
10.1.1.0	--	0
10.1.2.0	--	0
10.1.3.0	10.1.2.2	1
10.1.4.0	10.1.2.2	2
10.1.5.0	10.1.2.2	3

POT	SKOZI	SKOKOV
10.1.2.0	--	0
10.1.3.0	--	0
10.1.1.0	10.1.2.1	1
10.1.4.0	10.1.3.2	1
10.1.5.0	10.1.3.2	2

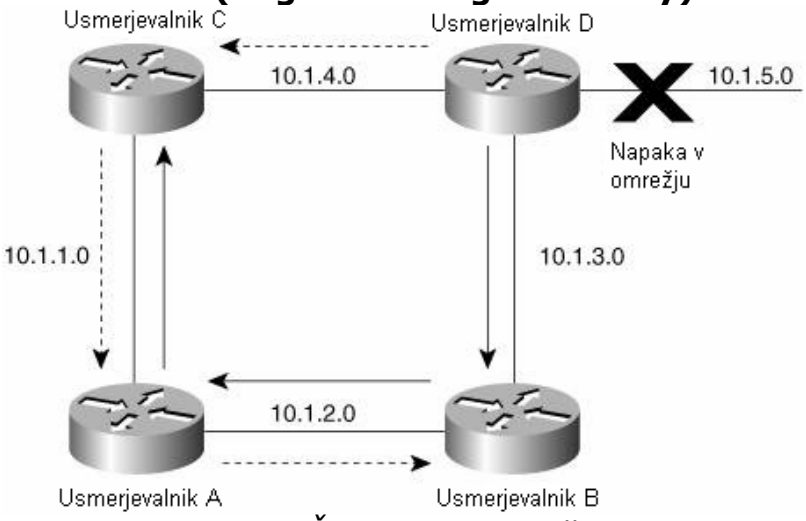
POT	SKOZI	SKOKOV
10.1.3.0	--	0
10.1.4.0	--	0
10.1.2.0	10.1.3.1	1
10.1.5.0	10.1.4.2	1
10.1.1.0	10.1.3.1	2

POT	SKOZI	SKOKOV
10.1.4.0	--	0
10.1.5.0	--	0
10.1.3.0	10.1.4.1	1
10.1.2.0	10.1.4.1	2
10.1.1.0	10.1.4.1	3

Slika 17. Primer osveževanja tabel.

V zgornjem primeru bi usmerjevalnik A brez potrebe poslal svojo tabelo usmerjevalniku B, saj vse poti, ki imajo več kot 0 skokov vodijo skozi njega. Usmerjevalnik bi v tem primeru celo podvojil tabelo z lažnimi potmi skozi usmerjevalnik A in dvema skokoma več. Temu se izognemo z metodo split-horizon, ki določa, da se naslovi na katerega je usmerjevalna tabela namenjena (povratni naslovi) izbrišejo iz nje. Prav tako pa se z to metodo ne razpošiljajo nepotrebni naslovi, kar pripomore h hitrejšemu procesiranju tabel.

Štetje v neskončnost (angl. Counting to infinity)



Slika 18. Štetje v neskončnost.

V tem primeru usmerjevalnik D zazna napako in pošlje pravilne tabele svojima sosedoma C (črtkana puščica) in B (polna puščica). Usmerjevalnik B označi pot do 10.1.5.0 kot nedostopno, vendar mu usmerjevalnik A sporoči, da lahko pride do 10.1.5.0 skozi C in D. B to doda v svojo tabelo in obvesti D, da ima pot, ki je tri skoke stran. D obvesti C, C obvesti A, in A obvesti B, da je pot do 10.1.5.0 sedaj šest skokov stran. Usmerjevalnik B sedaj popravi v svoji tabeli število skokov in cilj oglašuje naprej. Situacija se imenuje štetje v neskončnost, saj bodo vsi usmerjevalniki povečevali skoke v nedogled. To je preprečeno z definiranjem maksimalnega števila skokov. RIP protokol ima to definirano kot 16 skokov, IGRP kot 255, kar pomeni, da bodo sčasoma vsi usmerjevalniki dosegli to mejo in naslov 10.1.5.0 se bo smatral kot nedosegljiv. To se imenuje zastrupitev poti (angl. **route poisoning**). To je tudi način, kako usmerjevalniki sporočajo o prekinjeni poti. V poslani usmerjevalni tabeli za to pot nastavijo hop count na 15, in vsi usmerjevalniki, ki dobijo to tabelo opustijo naslov. Ta metoda se imenuje preprečitev zastrupitve (angl. poison reverse.)
To reši problem štetja v neskončnost, vendar si pri 30 minutnem osveževanju celotno omrežje počasi ustvari pravilno tabelo, medtem pa je izpostavljeno napakam. Temu pripomorejo sprožilne osvežitve (angl. triggered updates), ki pri vsakem zaznanju spremembe v topologiji omrežja takoj razpošljejo novo spremembo, kar pripomore h točnejšim tabelam.

Link state

Tudi znan kot SPF (angl. shortest path first) algoritem, vzdržuje kompleksno sliko topologije omrežja in na podlagi te izračuna najboljšo pot. »Link« predstavlja povezavo podobno kot pri usmerjevalniku, »state« pa opisuje povezljivost v omrežje in odnos do sosednjih usmerjevalnikov. Zato obratno kot distance vector algoritem, ki nima informacij o oddaljenih usmerjevalnikih, link state usmerjevalni algoritem vzdržuje celotno sliko o oddaljenih usmerjevalnikih in kako so povezani med sabo. Predstavnika link state protokolov sta IS-IS (angl. intermediate system – intermediate system) in OSPF (angl. open shortest path first).

Vzdrževanje usmerjevalnih informacij pri link state protokolu

Za vzdrževanje usmerjevalnih informacij, link state usmerjanje uporablja oglaševanje (angl. LSAs – link state advertisements), topološko tabelo, SPF algoritme, SPF drevesno strukturo in usmerjevalno tabelo poti in vrat do vsakega omrežja.
Link state usmerjevalni protokol zbira informacije o usmerjanju od vseh usmerjevalnikov v omrežju ali znotraj določenega območja (angl. area). Ko posamezen usmerjevalnik zbere vse informacije, si neodvisno od drugih izračuna najboljše poti do vseh ciljev v omrežju. Ker ima vsak usmerjevalnik svoj pogled na omrežje, je zelo majhna verjetnost obveščanja sosednjih usmerjevalnikov o napačnih poteh.
Link state usmerjevalni protokol je bil zasnovan z namenom izboljšanja omejitev, ki jih predstavlja distance vector usmerjevalni protokol. Hitro odzove na spremembe v omrežju, pošlje usmerjevalno tabelo le takrat, ko se zgodi sprememba v topologiji omrežja, periodične posodobitve (angl. link-state refreshes) pa pošilja občasno (30 minutni intervali). Za preverjanje dosegljivosti sosednjega usmerjevalnika se uporablja **Hello** mehanizem. To so manjši paketi, ki jih usmerjevalnik pošilja v hitrejših intervalih izključno z namenom, obvestiti sosede, da delujejo.
Ko se zgodi okvara v omrežju, kot je denimo nedosegljivost sosednjega usmerjevalnika, link state protokol poplavi omrežje z LSA paketi, pri čemer uporablja poseben večnamenski naslov (angl. multicast address). Vsak link state usmerjevalnik vzame kopijo LSA paketa, posodobi lastno link state (topološko) tabelo, in posreduje LSA paket vsem sosednjim usmerjevalnikom. LSA paket povzroči dodatno delo, saj mora vsak usmerjevalnik v omrežju preračunati nove poti. Ker se mora LSA paket razposlati po vsem omrežju, in mora vsak usmerjevalnik izračunati novo usmerjevalno tabelo, je pametno, da to omrežje vsebuje omejeno število link state usmerjevalnikov.
Osvežitev lahko vsebuje IP naslov priključka, masko omrežja, tip omrežja, na katerega je priključen, usmerjevalnike, ki so tudi priključeni na to omrežje, itd. Zbirka link-state predstavlja topološko tabelo povezav. Uporablja se za izračun najboljše poti skozi omrežje, pri tem pa uporabljajo Dijkstra SPF algoritem, ki iz topološke tabele najde najboljšo pot in jo vnese v usmerjevalno tabelo.

Link state protokoli uporabljajo dvo-slojno mrežno hierarhijo. Za to obstajata dva osnovna elementa:

- **Podomrežje** (angl. subnet) je skupina več omrežij, ki so manjše skupine logično razdeljenega avtonomnega sistema.
- **Avtonomni sistem** je sestava celotnega omrežja pod skupno administracijo. Delijo si tudi skupno usmerjevalno strategijo. Avtonomni sistem, včasih imenovan tudi domena je lahko logično razdeljena v več manjših podomrežij.

Znotraj vsakega avtonomnega sistema mora biti definirano neprekinjeno hrbtenično omržje (angl. backbone). Vsa ostala podomrežja so priključena nanjo in posredno komunicirajo z ostalimi omrežji preko nje. Pri OSPF se podomrežja lahko dodatno definirajo kot »stub« podomrežja, »totaly stubby« podomrežja ali »not-so-stubby« (NSSA) podomrežja z namenom, da zmanjšajo velikost link state usmerjevalne tabele.

Uravnovešeno hibridno usmerjanje

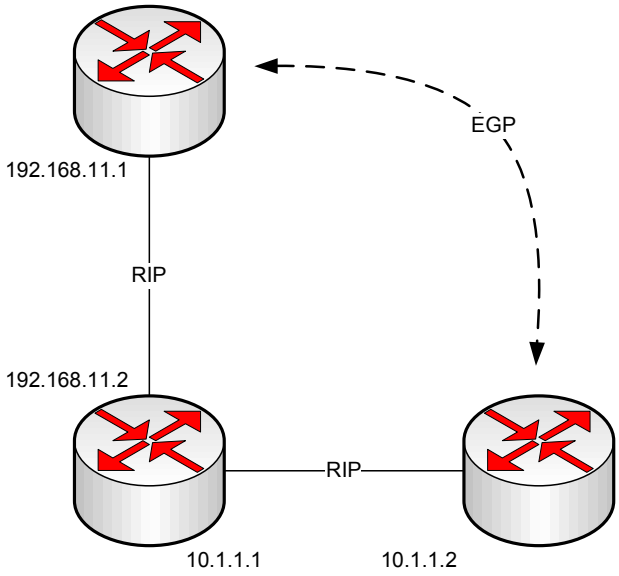
Hibridno usmerjanje je kombinacija distance vector in link state protokolov. Uporablja distance vectors vendar z bolj točno metriko za določitev najboljše poti do cilja. Toda, za razliko od večine distance vector protokolov, se pošiljanje paketov z usmerjevalnimi informacijami izvaja kadar nastopi sprememba v topologiji omrežja (jih ne pošilja periodično).

Odzivni časi hibridnega protokola so mnogo hitrejši od distance vector ali link state protokola. Za svoje delovanje pa rabi več spomina in procesorske moči.

Primer uravnovešeno hibridnega usmerjevalnega protokola je Cisco EIGRP.

3.1.2 EGP usmerjevalni protokoli

EGP protokoli po strogi definiciji praviloma ne spadajo med usmerjevalne protokole, saj v usmerjevalni tabeli shranjujejo le informacijo o dosegljivosti omrežij, ne pa dovolj podatkov, iz katerih bi lahko izračunali najkrajšo oziroma najboljšo pot ali preprečili zanke v omrežju. Njihova poglavitna naloga je povezovanje avtonomnih sistemov med sabo. Poleg komunikacije med mejnimi usmerjevalniki o EGP informacijah, lahko ti protokoli nadomestijo IGP protokole in z tem usmerjajo promet znotraj enega avtonomnega sistema. Komunikacija z EGP protokolom lahko poteka med dvema usmerjevalnikoma tudi, če običajno med njima usmerja promet kateri od IGP protokolov.



Slika 19. Komunikacija po EGP protokolu.

Slika prikazuje komunikacijo med usmerjevalnikoma, ki spadata v različna IP omrežja, med njima poteka komunikacija z RIP protokolom in lahko sta celo v različnih avtonomnih sistemih. Princip komunikacije z EGP protokolom se bistveno razlikuje od IGP načina komunikacije. Za začetek že EGP protokol ne uporablja IP naslovov, kar je pri IGP protokolih osnova za delujočo komunikacijo. EGP protokol tudi nima načina za izračun najboljših poti, saj uporablja sistem števila skokov za iskanje ciljev, kar v IGP protokolih uporablja RIP, eden prvih protokolov. Število skokov pa tudi ni optimalen faktor za hiter prenos podatkov po omrežju.

5.4 ISDN

5.4.1 Uvod

Digitalno omrežje z integriranimi storitvami oz. ISDN (Integrated Services Digital Network) je tip vodovno komutiranega telefonskega omrežja, ki je zasnovan tako, da omogoča digitalen prenos govora in podatkov po klasični telefonski bakreni parici. Omogoča višji nivo kvalitete in večje hitrosti kot klasičen analogen telefonski sistem.

Beseda ISDN je bila prvič uporabljena leta 1972, toda ideja o ISDN omrežju sega nazaj v pozna 60-ta leta.

Leta 1984 je takratno svetovno telekomunikacijsko telo za standarde (CCITT, sedaj ITU-T) definiralo ISDN na naslednji način:

ISDN je definiran kot omrežje, ki je razvito iz integriranega telefonskega omrežja in omogoča digitalno povezljivost med končnimi terminali na celotni prenosni poti, poleg tega pa podpira širok spekter govornih in negovornih storitev, do katerih imajo uporabniki dostop preko omejenega nabora standardnih, večnamenskih vmesnikov uporabnik-omrežje.

Bistvene prednosti, ki jih ponuja ISDN, so:

- Širši spekter storitev.
- Možnost več zvez na enem priključku.
- Možnost 7 kHz namesto 3.1 kHz pasovne širine za prenos zvoka.
- Tehnološko poenoteni uporabniški priključek za dostop do javnih komunikacijskih omrežij (PSTN, X.25) za prenos glasu, slike, besedila in podatkov.
- Večja udobnost komuniciranja.
- Digitalna komunikacija s hitrostmi do 2 Mbit/s.
- Krajši čas vzpostavljanja zveze.
- Izboljšana kakovost storitev.

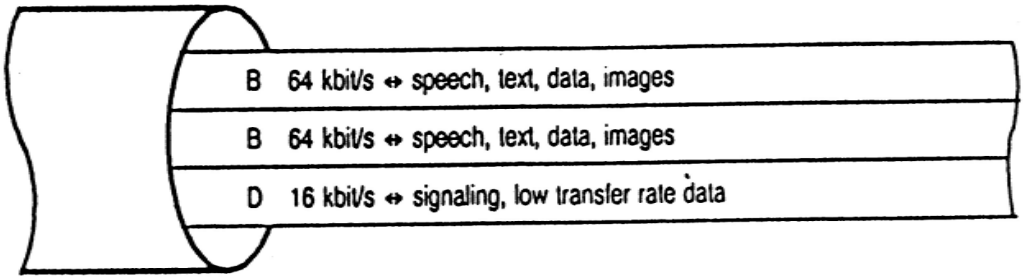
Telekomunikacijski operaterji precej vlagajo v položeno bakreno infrastrukturo, ki pa zaradi svojih fizikalnih zakonitosti omejuje pasovno širino komunikacije. Zamenjava obstoječega bakrenega omrežja z nečim manj omejujočim, kot je na primer optično vlakno, je v tem trenutku ekonomsko neupravičena, zato je bila v razvoju ISDN-ja ključna zahteva, da mora biti prepletena bakrena parica primerna za osnovni dostop do ISDN vstopne točke.

3.2 Vrste dostopa

Uporabnik se lahko priključi na omrežje prek priključne točke z osnovnim ali s primarnim dostopom.

3.2.1 Osnovni dostop

Osnovni dostop (BRA – Basic Rate Access) omogoča na svojem vodilu S_0 dva osnovna kanala (B-kanala) s standardnima hitrostma 64 kb/s in signalizacijsko-podatkovni kanal (kanal D) s hitrostjo 16 kb/s. Vsak od dveh B-kanalov omogoča prenos uporabniške informacije (govor, tekst, podatki, slika). Kanala sta lahko uporabljena in komutirana neodvisno eden od drugega. To pomeni, da obstajata istočasno dve neodvisni zvezi (govor, podatki ali faks). Lahko pa sta oba B-kanala uporabljena v eni sami zvezi. Primer je video zveza, kjer se v enem kanalu prenaša govor, v drugem pa slika.



Slika 20.Osnovni dostop – ISDN BRA.

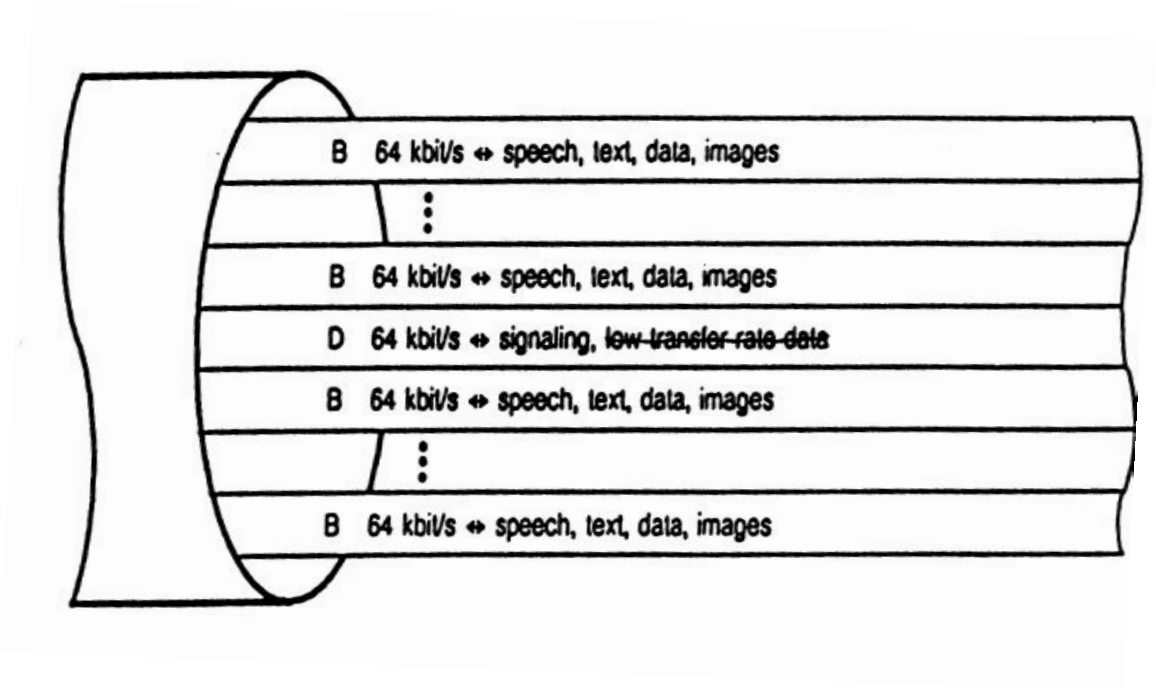
Po D-kanalu se prenaša signalizacija za vzpostavljanje in prekinjanje zveze, za prenos informacij o naročnikih in vrstah želenih storitev. Osnovni dostop fizično tvori par bakrenih žic, po katerih se prenašata oba B-kanala in signalni D-kanal. V vseh treh kanalih se prenaša 144kb/s koristne informacije (2 x 64 kb/s + 16 kb/s).

Istočasno lahko na en BRA priključimo do osem poljubnih terminalov, vsak od teh ima lahko svojo klicno številko in se lahko po želji tarifira posebej. Naenkrat pa sta lahko aktivna le dva.

Osnovni dostop je tako primeren za uporabnike z manjšimi zahtevami.

3.2.2 Primarni dostop

Primarni dostop (PRA – Primary Rate Access) omogoča na vmesniku S_2M 30 osnovnih kanalov B s hitrostjo 64 kb/s. PA torej omogoča 30 istočasnih zvez in je primeren za podjetja, ki so do sedaj uporabljala velike naročniške centrale in številne dodatne priključke za negovorne storitve. Podobno kot pri osnovnem dostopu, kjer se v D-kanalu prenaša signalna informacija, se tudi na PA uporablja D-kanal, ki je 16.kanal v PCM30. Časovna okna od 1 do 15 in od 17 do 31 so B-kanali in služijo kot uporabniški kanali za prenos govora, podatkov in slike.



Slika 21. Primarni dostop – ISDN PRA.

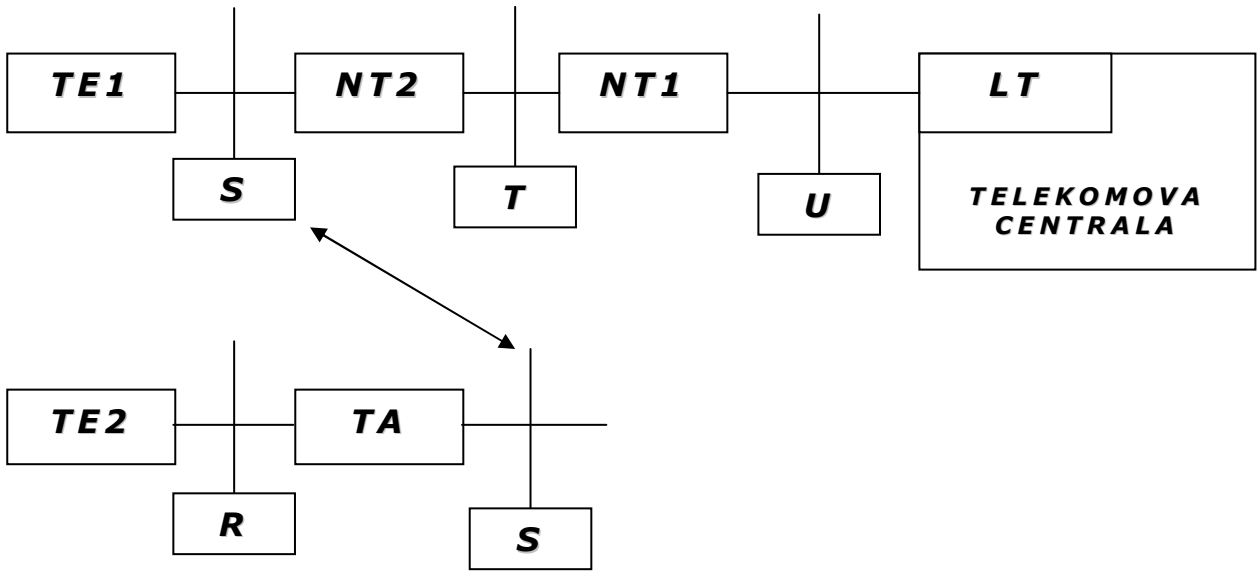
Tako kot pri BRA se B-kanali zasegajo in komutirajo individualno, odvisno od poziva, medtem ko se signalna sporočila za vse B-kanale prenašajo po D-kanalu. Osnovni in primarni dostop se razlikujeta tudi v funkciji D-kanala. Pri primarnem dostopu se v D-kanalu prenašajo izključno samo signalna sporočila. Paketno orientirani podatki, ki jih je možno pri osnovnem dostopu prenašati po D-kanalu, morajo biti pri primarnem dostopu ločeni od signalnih podatkov. Paketno orientirani podatki se pri primarnem dostopu prenašajo po B-kanalih. Hitrost prenosa je kar prenosna hitrost samega PCM, to je 2 Mb/s. Vsak od B-kanalov prenaša govor, podatke, tekst ali sliko s hitrostjo 64 kb/s in tudi v D-kanalu se signalna sporočila prenašajo s hitrostjo 64 kb/s.

3.3 ISDN referenčne oz. priključne točke

Poglejmo si nekatere osnovne kratice, ki jih srečujemo v sistemu *ISDN*:

- *LT* (Line Termination) je točka, kjer se zaključuje osnovni dostop. To je ponavadi centrala.
- *NT* (Network Termination) je oprema za zaključevanje linije pri uporabniku. Ta se deli naprej na *NT1* in *NT2*, vendar je *NT2* zelo redko uporabna. Privzeli bomo, da je naprava priključena na *NT1.*
- *TE* (Terminal Equipment) je oprema, ki se priključi direktno na *S/T* vmesnik. Ta vključuje *ISDN* telefone, modeme …
- *TA* (Terminal Adapter) je vmesnik, preko katerega se na omrežje lahko priključijo naprave, ki niso namenjene za delovanje v *ISDN* omrežju (npr. analogni telefon). Naprave, ki se priključijo na *TA* se imenujejo s skupno kratico *TE2*.
- *S/T* vmesnik se imenuje štiri-žično vodilo (pozitivna in negativna oddajna in sprejemna žica) med *NT1* in *ISDN* terminalno opremo.
- *U* vmesnik je dvožična povezava med *NT1* in centralo.

Na sliki so priključne točke videti takole:



Slika 22. Priključne točke in tipi ISDN opreme.

3.4 ISDN kanali

Pri izvedbi ISDN-a lahko krajevna zanka prenaša le digitalne signale, uporablja pa tri vrste kanalov, ki služijo signalizaciji in prenosu podatkov. To so:

- **B-kanal -** prenaša podatke za uporabnikove storitve, kot so govor, zvok, video in digitalni podatki. Hitrost prenosa je DS-0 (64 kb/s).

- **H-kanal –** ima enake naloge kot B kanal, le da prenaša pri višjih hitrostih kot DS-0.

- **D-kanal -** prenaša signalizacijske podatke med uporabnikom in omrežjem, lahko pa prenaša tudi uporabnikove podatkovne pakete.

B-kanal

Primarni namen B-kanala je, da zagotavlja nosilne storitve, te so prenos govora, zvoka, slik, podatkov in videa. Preko B-kanala se prenašajo storitvene zahteve. Vedno deluje s hitrostjo 64 kb/s, ki je hitrost, prilagojena digitalnim govornim aplikacijam.

B-kanal se lahko uporablja za povezavni ali paketni način prenašanja podatkov. Povezavni način zagotavlja transparentno povezavo od uporabnika do uporabnika, nad fizičnim slojem (64 kb/s) ni drugih definiranih protokolov. Vsak uporabnik je zadolžen za definicijo protokolov, ki se bodo uporabljali v povezavi in da zagotovi združljivost med napravama, ki ju povezuje B-kanal. Paketni način povezave podpira naprave, ki uporabljajo protokole kot so X.25 ali frame relay. ISDN lahko zagotovi notranjo storitev za paketni način ali pa nudi dostop do obstoječega PSPDN (javno digitalno omrežje s paketno komutacijo) omrežja.

B-kanali se lahko uporabljajo na klic ali so trajni. Če so B-kanali trajni, ni potrebna signalizacija D-kanala za delovanje B-kanala.

H-kanal

Za prenosne hitrosti, višje kot 64 kb/s, se uporabljajo H-kanali. Te hitrosti so ekvivalentne hitrosti določene skupini B-kanalov. Aplikacije, ki zahtevajo storitve H-kanala so: telekonference, video storitve, povezovanje lokalnih omrežij (LAN), visoko kvalitetni zvok …

Prvi je H0-kanal, ki deluje s hitrostjo 384 kb/s, kar je enakovredno logični združitvi šestih B-kanalov.

H1-kanal obsega vsa časovna okna, ki so na voljo na T1ali E1 nosilcu. H11- kanal deluje s hitrostjo 1.536 Mb/s in je enakovreden 24 časovnim oknom (24 B-kanalov, združljiv s T1 nosilcem), H12-kanal deluje s 1.920 Mb/s, kar znaša 30 časovnih oken (30 B-kanalov) in je združljiv z E1 nosilcem.

3.5 Signalizacijski D-kanal

Omrežje in uporabnikova naprava si vse zahteve po storitvah in ostala signalizacijska sporočila izmenjujejo preko D-kanala. Vendar pa se lahko preko D-kanala prenašajo tudi uporabnikovi podatki v paketnem načinu. Hitrost prenosa teh podatkov je lahko največ 9,6 kb/s. Seveda pa imajo signalizacijska sporočila prednost pred podatkovnimi paketi.

D-kanal deluje s hitrostjo 16 kb/s pri osnovnem dostopu (BRA) in s 64 kb/s pri primarnem dostopu (PRA).

Signalizacija zajema podatki, kot npr. podatki o kličočem, preusmeritvah, statusu na liniji, seveda pa tudi tiste osnovne, brez katerih ne moremo telefonirati, npr. vzpostavljanje zvez, podatki o zvonjenju, zasedenosti linije in drugi.

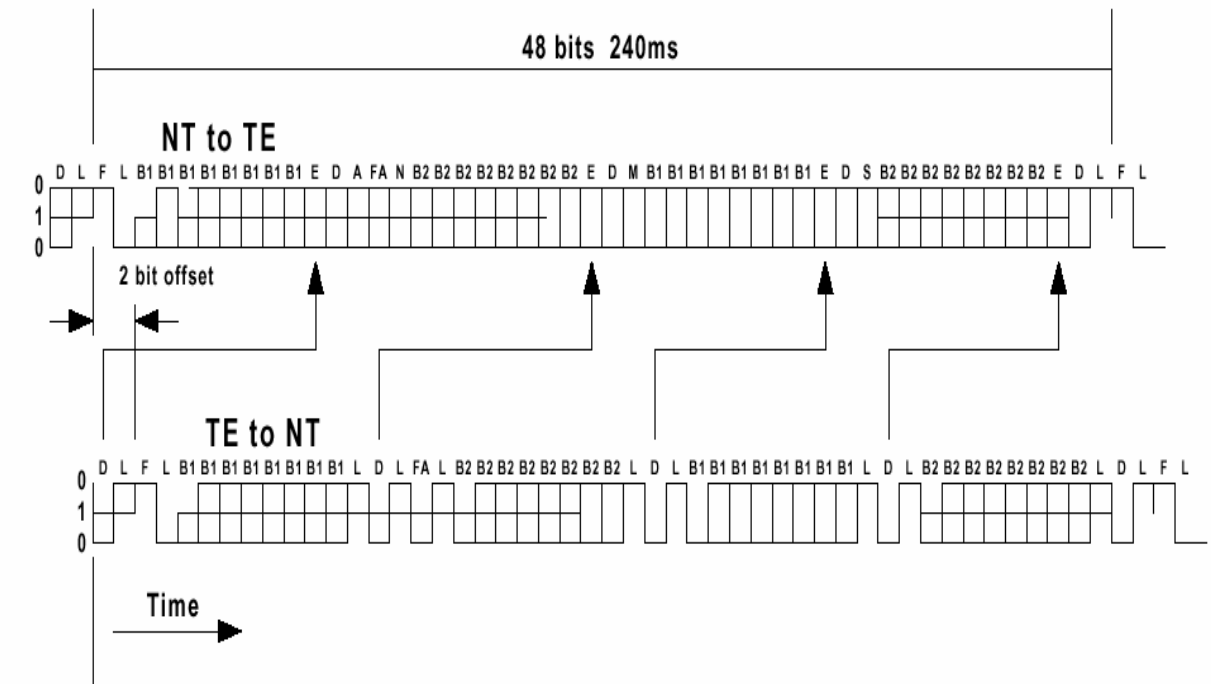
Če bi protokol na *D* kanalu umestili v *ISO-OSI* model, bi nivoji izgledali takole:

- Prvi nivo je fizični. Na tem nivoju se prenašajo kodirani podatki v t.i. psevdo-ternarni kodi.
- Drugi oz. povezavni nivo predstavlja *HDLC*, ki se v *ISDN* imenuje *LAPD* oz. Q.921.
- Na tretjem oz. mrežnem nivoju pa je protokol *Q.931*. Tukaj se prenašajo vsi podatki za vzpostavljanje zveze.
- Na višjih nivojih pa so vse aplikacije, ki jih *ISDN* omogoča.

3.5.1 Fizični sloj D kanala na S/T vodilu

V tem poglavju bo na kratko opisan le zelo splošen pregled dogajanja na tem nivoju. To podpoglavje smo uvrstili v poglavje o *D* kanalu zaradi celovitosti opisa dogajanja na slojih modela *ISO-OSI*, čeprav je ta fizični kanal seveda tudi podlaga za prenos govornih *B* kanalov.
Vodilo za prenos podatkov na *S/T* referenčni točki je štiri žično (dve sprejemni in dve oddajni žici). Prenos podatkov na tem nivoju poteka s

hitrostjo 192 Kbps. Linijsko kodiranje na tem nivoju je t. i. psevdo-ternarno. Pri takšnem kodiranju predstavlja logično enico nič voltov, logično ničlo pa impulzi $\pm 750mV$, ki se zaporedoma izmenjujejo.
Vsak psevdo-ternaren okvir je sestavljen iz 48 bitov, pri čemer je od tega 36 bitov *2B+D* kanala ostalih 12 bitov pa je pomožnih (overhead). Uporabni podatki v takšnem okvirju si sledijo takole: 8 bitov *B1*, 1bit *D*, 8 bitov *B2* in zopet 1 bit *D* kanala. Ta sekvenca se ponovi dvakrat na okvir. Pomožni biti se pojavljajo med skupinami *B* bitov in *D* bitov. Pomožne bite uporabljata *TE* in *NT* za aktivacijo, balansiranje… To je podrobneje vidno na spodnji sliki.
Na tem mestu naj tudi omenim, da je maksimalna dolžina voda med *TE* in *NT* 1 km, če je to povezava zgolj med enim *TE* in enim *NT*. Če pa je na en *NT* priključenih več *TE*, je maksimalna dolžina le 200 m. Te omejitve so zaradi faznega zamika, ki bi nastal ob prevelikih razdaljah.



- *F* (Framing bit) bit za sinhronizacijo okvirjev
- *L* (Balancing bit) bit za izničenje enosmerne komponente
- *D* (Data bit) bit signalizacijskega kanala
- *E* (Echo bit) odmev bita signalizacijskega kanala – t. j. za pogajanje za dostop do *D* kanala)
- *FA* (Auxiliary framing bit) ob vsakem veljavnem okvirju je postavljen na 0)
- *M* (multiframing bit) sprememba formata okvirja
- *B 1, 2* (Bearer 1, 2) uporabniški govorni podatki
- *A* (Activation bit) Za aktiviranje zanke (glejte poglavje 5.2)
- *S* Rezerviran za prihodnjo uporabo

Slika 23. Fizični sloj na S/T vodilu po ISO OSI modelu.

3.5.2 Drugi nivo v protokolnem skladu – protokol LAPD

Na drugem, oz. povezavnem sloju je v *ISDN* standardu definiran protokol *LAPD* (Link Access Procedures D Channel). Namen tega sloja je, da zagotovi prenos podatkov brez napak.
V osnovi je *LAPD* sestavljen iz petih sklopov, in sicer:

FLAG t.j. zaporedje osmih bitov 01111110, ki se pojavljajo na začetku in na koncu vsakega *LAPD* okvirja.

ADDRESS - 16 bitno polje, sestavljeno iz dveh delov, in sicer *SAPI* (Service Access Profile Identifier) in *TEI* (Terminal Endpoint Identifier), ki služita pravilnemu naslavljanju enot.

CONTROL – 8 oz. 16 bitno polje, ki pove, za kakšen tip okvirja gre.

INFO polje, kjer so podatki tretjega sloja (podatki o telefonski številki, zvonjenju, zasedenosti linije...).

FCS polje šestnajstih bitov pred tipično zaključno sekvenco

	oktet
Tipična sekvenca 0 1 1 1 1 1 1 0	1
Naslovno polje (prvi oktet)	2
Naslovno polje (drugi oktet)	
Kontrolno polje	3
Kontrolno polje	4
Informacije	5
FCS	. . .
FCS	N-2
Tipična sekvenca 0 1 1 1 1 1 1 0	N-1

Slika 24. Polja v splošnem LAPD okvirju.

Transparentnost LAPD protokola

Pojavi se vprašanje, kaj se zgodi, če se pojavi tipična sekvenca v samem prenosu podatkov tretjega sloja. V ta namen je v protokol vgrajen zaščitni mehanizem, ki takoj, ko opazi zaporedje petih enic vrine ničlo. Seveda mora sprejemnik narediti enako in vsako ničlo, ki sledi petim enicam

izbrisati, oziroma je ne sme upoštevati. Opisano se imenuje transparentnost *LAPD* protokola.

Naslovno polje

Naslovno polje, ki sledi tipičnemu zaporedju je namenjeno temu, da lahko enota *NT* pravilno naslavlja vsak *TE* posebej (vemo, da se na en *NT* lahko priključi več terminalov), saj je naslovno polje sestavljeno iz dveh delov *SAPI* in *TEI*.
Prvi del naslova je *SAPI*. Ta je namenjen temu, da usmeri podatek na pravilno logično enoto. Pri tem je logična enota kot nekakšna funkcija, ki je lahko vzpostavljanje zveze, odgovor ali pa rušenje zveze. Tako imajo vse funkcije, ki se nanašajo na kontrolo klica enak *SAPI*. Po priporočilih *ITU* obstajajo naslednji tipi vrednosti polja *SAPI.* Te vrednosti so prikazane v spodnji tabeli:

VREDNOST SAPI	LOGIČNA ENOTA
0	Procedure vzpostavljanja zveze
1-11	Rezervirano za prihodnjo standardizacijo
12	Teleaction
13-15	Rezervirano za prihodnjo standardizacijo
16	Paketna komunikacija prilagojena tretjemu sloju X.25
17-31	Rezervirano za prihodnjo standardizacijo
63	Upravljanje sloja 2
Vsi ostali	Niso na voljo LAPD proceduram

Tabela 1. Standardizirane vrednosti SAPI.

Seveda se vrednosti *SAPI* še vedno dopolnjujejo s strani *ITU-T.*
SAPI predstavlja šest bitov naslovnega polja. Dva bita, ki sledita šest bitni vrednosti sta bit *C/R* in bit *EA*. Slednji je tudi zadnji bit v drugem oktetu, kjer je vrednost *TEI*. Prvi pove ali je okvir ukazni, ali pa je odgovor na našo zahtevo. Ko pošlje uporabnik ukaz, se mora bit *C/R* postaviti na nič. Ko pa pošlje odgovor, se postavi *C/R* bit na ena. Na strani omrežja pa je ravno obratno. Kadar le-to hoče poslati ukaz, postavi *C/R* bit na ena, ko pa želi odgovoriti na ukaz pa na nič.

EA bit je indikator, ki pove, če je oktet v naslovnem polju prvi ali pa zadnji. Zato mora biti bit *EA* v oktetu, kjer je vrednost *SAPI* vedno postavljen na nič (ker je to prvi oktet naslovnega polja), če pa je *TEI* pa na ena. Če zahteva ni izpolnjena, je okvir zavrnjen kot neveljaven.
Ustavimo se sedaj še nekoliko pri drugem delu naslovnega polja (*TEI*). Vsak *TE* ima svojo *TEI*. Dodeljevanje številke *TEI* je lahko avtomatsko ali pa fiksno. V večini primerov je uporabljena avtomatska metoda. Po priporočilih *ITU* so na voljo naslednje *TEI*:

VREDNOST TEI	TIP TEI
0-63	Fiksni *TEI*
64-126	Avtomatski *TEI*
127	*TEI* za naslavljanje vseh *TE* na določenem *BRI*

Tabela 2. Standardizirane vrednosti TEI.

Pri avtomatski izbiri *NT* dodeli *TEI* katerokoli številko med 64 in 126. Pri našem vmesniku je zaradi specifičnosti izdelka (na vsak *NT* se priključi le en *TE*) uporabljen statičen *TEI* in je postavljen na vrednost 0.
SAPI in *TEI* tvorita skupaj naslovno polje, ki se imenuje *DLCI* (Data Link Connection Identifier).
Signalizacija v takšni obliki gre zgolj od *TE* do *LT*. V centralah operaterja se signalizacija preoblikuje v signalizacijo *SS7* (Signaling System Seven) za komunikacijo med centralami.

Kontrolno polje

Naslovnemu polju sledi kontrolno polje. Dolžina kontrolnega polja zavisi od tipa okvirja. Lahko je 8 ali pa 16 bitno. Obstajajo trije tipi okvirjev:

- Informacijski (*I frame*)
- Nadzorniški (*S frame*)
- Neoštevilčeni (*U frame*)

Ti pa se naprej delijo v podsklope, ki jih vidimo v spodnji tabeli:

Tip okvirja	Podsklop okvirja	Ime	Opis
I	**I**	**I**nformation	Namenjen prenosu podatkov tretjega sloja
S	**RR**	**R**eceiver **R**eady	Sprejemnik pripravljen
S	**RNR**	**R**eceiver **N**ot **R**eady	Sprejemnik ni pripravljen
S	**REJ**	**R**eject	Zavrni
U	**SABME**	**S**et **A**synchronus **B**alance **M**ode **E**xtended	Zahteva za inicializacijo
U	**DISC**	**D**isconnect	Prekini
U	**UI**	**U**nnumbered **I**nformation	Neoštevilčene informacije
U	**UA**	**U**nnumbered **A**cknowledgement	Potrditev neoštevilčenih okvirjev
U	**DM**	**D**isconnect **M**ode	Neaktivno stanje
U	**FRMR**	**Fr**a**m**e **R**eject	Zavrnitev okvirja
U	**XID**	**E**xchange **ID**	Izmenjava ID

Tabela 3. Tipi okvirjev v protokolu LAPD.

Vsak informacijski okvir je označen z zaporedno številko. Vrednosti te številke lahko zavzamejo vrednosti od 0 do *n-1*, pri čemer je *n* modul sekvenčnih števil in je enak 128. Zaporedne številke se zvrstijo do 127 in se nato resetirajo na nič.
V naslednjih vrsticah bomo opisali vrednosti, s pomočjo katerih določamo te zaporedne vrednosti. Vse vrednosti se resetirajo na nič, ko pošlje ena izmed strani okvir *UA* kot odgovor na okvir *SABME* (vzpostavitev logične povezave).

Vrednosti povezane s številčenjem okvirjev so naslednje:

- **V(S);** vsaka logična povezava mora imeti definirano vrednost *V(S)*. Le-ta označuje zaporedno vrednost naslednjega *I* okvirja. Vrednosti, ki jih lahko zavzame *V(S)* so v območju od 0 do *n-1*. Vrednost *V(S)* se poveča za ena vedno, ko pošljemo okvir *I* in ne sme preseči vrednosti *V(A)* za več kot je število nepotrjenih okvirjev *I, k*. Vrednost *k* je lahko med 1 in 127.
- **V(A);** vsaka logična povezava mora imeti definiran tudi *V(A)* pri uporabi informacijskih in nadzorniških okvirjev. *V(A)* označuje zadnji informacijski okvir, ki ga je nasprotna stran potrdila. *V(A)* lahko zavzame vrednosti med *0* in *n-1*. Vrednost *V(A)* se obnavlja z veljavnimi vrednostmi *N(R)* od nasprotne strani.
- **N(S);** To vrednost vsebujejo le informacijski okvirji. Preden je okvir namenjen za pošiljanje, se vrednosti *N(S)* priredi vrednost *V(S).*
- **V(R);** vsaka logična povezava mora imeti *V(R),* če uporablja informacijske ali nadzorniške *V(R)* okvirje. *V(R)* označuje zaporedno številko naslednjega *I* okvirja, ki ga pričakuje. Zavzame lahko vrednosti med *0* in *n-1.* Vrednost *N(R)* se poveča za ena za vsak sprejeti *I* okvir brez napak, katerega vrednost *N(S)* je enaka *V(R).*
- **N(R);** vsi informacijski in nadzorniški okvirji vsebujejo *N(R)*. To je pričakovana vrednost *N(S)* naslednjega sprejetega *I* okvirja. Ko je informacijski ali nadzorniški okvir pripravljen za pošiljanje, priredimo vrednosti *N(R)* vrednost *V(R). N(R)* nam pove, da so bili pravilno sprejeti vsi *I* okvirji do vključno *N(R) -1.*

Informacijsko polje

Informacijsko polje sledi kontrolnemu polju in se konča pred poljem *FCS* preverjanja napake. Vsebino informacijskega polja določa protokol *Q.931*, ki bo na kratko opisan v naslednjem poglavju. Glede protokola *LAPD* velja omeniti, da mora informacijsko polje biti sestavljeno iz naravnega števila oktetov.

FCS polje

FCS je 16 bitna sekvenca, ki jo dobimo kot rezultat izračuna 16 bitnega *CRC*, nad katerim izvedemo eniški komplement. *CRC* je ostanek pri deljenju dveh polinomov. Deljenec dobimo iz sporočila tako, da vzamemo kot koeficiente polinoma kar bite iz sporočila. Na mestu, kjer je v sporočilu enica, množimo trenutni člen polinoma z ena, kjer pa je ničla ga množimo z nič. V nekaterih izvedbah pomnožimo dobljeni polinom pred deljenjem še z X^{16}.
CRC lahko računamo na več načinov. Obstaja tudi mnogo različnih vrst *CRC*. Razlikujejo se lahko zaradi:

- različnih generacijskih polinomov,
- različnih začetnih vrednosti registrov,
- različnega števila bitov, ki sestavljajo *CRC.*

V protokolu *LAPD* je definiran *CRC* s strani organizacije *ITU-T*. *CRC* je 16 biten.

Format okvirjev

Nazadnje se nekoliko ustavimo še pri opisu in izgledu okvirjev v *LAPD*.
Biti so vedno poslani v takšnem vrstnem redu, da je prvi poslan bit 1 in nazadnje bit 8. Če je polje dolgo več kot en oktet, je najmanj pomemben bit tisti, ki je v oktetu z najvišjo vrednostjo in ima najnižjo zaporedno številko v danem oktetu.
Izjema pri tem opisu je polje *CRC*, ki je dolgo dva okteta. V tem primeru je prvi poslani bit najbolj pomemben bit v prvem oktetu. Zadnji poslani bit v drugem oktetu pa je najmanj pomemben bit.

3.5.3 Mrežni nivo (Q.931)

Tretji nivo oz. *Q.931* je dokaj obsežen, zato bomo navedli le nekaj najosnovnejših okvirjev.
V tretji nivo spadajo procedure vzpostavljanja, vzdrževanja in rušenja zvez.
Ko želi *TE* vzpostaviti zvezo, pošlje vmesniku sporočilo *SETUP* v informacijskem *LAPD* okvirju. V tem sporočilu so zajeti vsi podatki o klicu. Med temi naj omenim podatek o telefonski številki klicanega, telefonski številki kličočega, tipu izvorne številke, tipu ponorne številke, planu številčenja, referenčni številki klica, predlaganemu *B* kanalu … *NT* v splošnem potrdi klic z okvirjem *SETUP ACK*, v katerem seveda objavi referenčno številko klica, ki ga potrjuje in poleg ostalega pove, če se strinja z dodeljenim *B* kanalom, ali pa predlaga drugega. *CONNECT* (za vzpostavitev povezave), *DISCONNECT* (za rušenje povezave), *ALERTING* (za indikacijo, da klicanemu zvoni telefon), *RELEASE* (za sprostitev linije, t. j. sprostitev vrednosti referenčne številke), *RELEASE COMPLETE* (za potrditev druge strani o sprostitvi linije), ki sicer lahko vsebujejo nekatere dodatne podatke.
Ponovno želim poudariti, da je takšen opis sporočil le manjši del v primerjavi z vsemi možnostmi, ki jih ponuja protokol *Q. 931*. Mnogo obširnejšo razlago pa najdemo v [3].

3.5.4 Primer dejanskih okvirjev

Za konec tega poglavja pa si oglejmo, kako izgleda dejanski okvir, ki ga pošljemo skozi *D* kanal ter kaj v pomenijo posamezni biti. Kot primer bomo izbrali sporočilo *SETUP*, ki ga obdelamo in razkodiramo.

SETUP

	HEX	BIN	POMEN
2.nivo	7E	01111110	Startna sekvenca
	02	00000010	SAPI=0 C/R=1 (ukaz) EA=0 (prvo polje)
	01	00000001	TEI=0 EA=1 (zadnje polje)
	XX	XXXXXXX0	Vrednost N(S) Zadnji bit je vedno 0
	XX	XXXXXXXP	Vrednost N(R)
3.nivo	08	00001000	Protocol discriminator (vrednost je vedno enaka)
	01	00000001	Dolžina polja referenčne številke klica
	XX	XXXXXXXX	Referenčna številka klica
	05	00000101	Tip sporočila je SETUP
	A1	10100001	Sending complete
	04	00000100	Naslednja informacija je Bearer Capability
	02	00000010	Dolžina tega polja
	80	10000000	1 bit je vedno 1 kodiranje je ITU informacija je govor
	90	10000000	1 bit je vedno 1 prenos je tokokrogovni hitrost prenosa je 64kbps
	18	00011000	Naslednja informacija je channel identification
	01	00000010	Dolžina tega polja
	81	10000001	prvi bit je vedno 1 vmesnik je implicitno definiran dostop je osnoven (BRI) prazen bit

			predlagani B kanal je le priporočen
			identificirani kanal ni D kanal
			Izbira B kanala
	6C	01101100	Naslednja informacija je številka kličočega
	0A	00001010	Dolžina tega polja
	81	10000001	prvi bit je vedno 1
			tip št. je neznan
			plan številč. je ISDN
	30	00110000	1. številka
	34	00110100	2. številka
	31	00110001	3. številka
	33	00110011	4. številka
	36	00110110	5. številka
	38	00111000	6. številka
	32	00110010	7. številka
	35	00110101	8. številka
	34	00110100	9. številka
	70	01101100	Naslednja informacija je številka klicanega
	0A	00001010	Dolžina tega polja
	81	10000001	prvi bit je vedno 1
			tip št. je neznan
			plan številč. je ISDN
	30	00110000	1. številka
	33	00110011	2. številka
	38	00111000	3. številka
	39	00111001	4. številka
	39	00111001	5. številka
	36	00110110	6. številka
	31	00110001	7. številka
	31	00110001	8. številka
	30	00110000	9. številka
2.nivo	XX	XXXXXXXX	polje FCS
	XX	XXXXXXXX	polje FCS
7E	01111110	zaključna sekvenca	