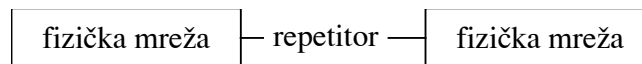


8. Mrežne komponente

Gradivni blokovi savremenih računarskih mreža su različiti tipovi elektronskih uređaja čiji je osnovni zadatak da primljenu informaciju proslede na korektnu destinaciju. U zavisnosti od funkcije koje obavljaju njihova složenost može da bude različita. Najprostiji su repetitori, a najsloženiji *gateway*, a između ova dva krajnja tipa uređaja postoji širok dijapazon drugih. U tekstu koji sledi ukazaćemo na osnovne karakteristike i funkcije koje obavljaju neki od najčešće zastupljenih komunikacionih uređaja.

8.1. Repetitori

Duž prenosnog medijuma (bilo da je veza ostvarena kablovskim putem, preko optičkog vlakna ili kroz etar) amplituda signala na prijemnom kraju opada sa porastom rastojanja. Da bi se premostilo ovo ograničenje, koriste se specijalni tipovi pojačavača poznati kao repetitori. Repetitor ima zadatak da podigne nivo signala, a time i direktno obezbedi uslove za povećanje rastojanja (dometa prostiranja signala) na fizičkom medijumu. Kod OSI referentnog modela, repetitor radi samo na nivou 1, tj. fizičkom nivou. Ne postoji potreba da se njemu dodeli hardverska adresa, tj. repetitor ne zahteva adresu. Signal se jednostavno pojačava, a za slučaj da je signal digitalni isti se samo regeneriše. Repetitor se koristi za spajanje dve mreže istih fizičkih karakteristika, kakve su recimo 10base 2CSMA/CD mreže (Slika 1.)

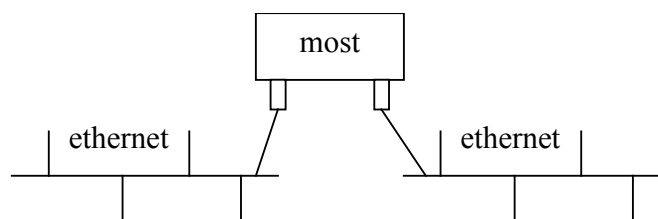


Slika 8.1 Mreža sa repetitorom

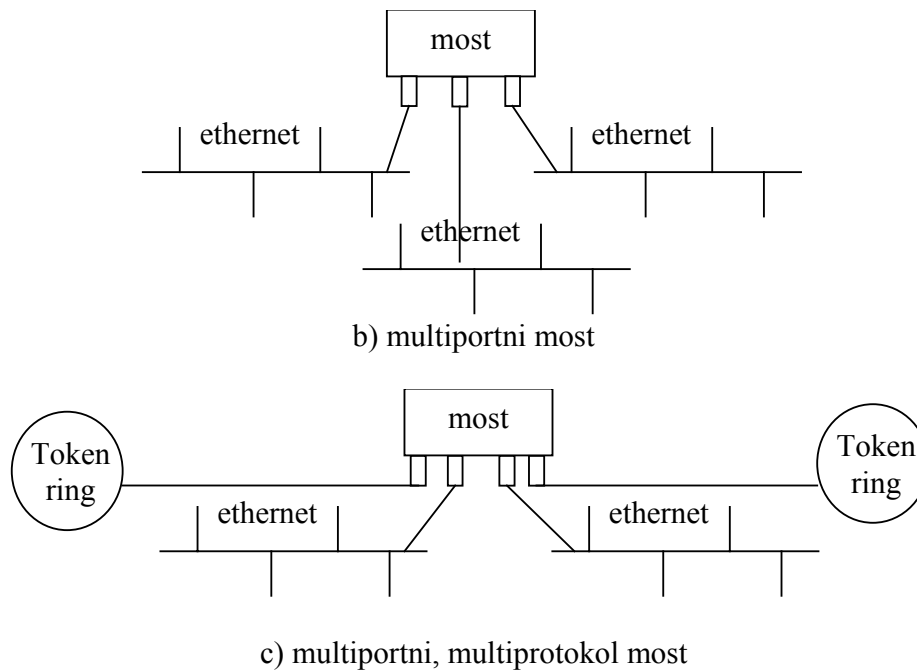
8.2. Mostovi

Veoma često kod LAN mreža se javlja potreba za prenosom paketa na većim rastojanjima od dozvoljenih, ali pri tome treba naglasiti da dodatne uređaje nije moguće priključiti s obzirom da je specificirani (dozvoljeni) broj uređaja po segmentu na samoj ivici da bude premašen. Kod ovakvih situacija poželjno je koristiti mostove (*bridge*). Sa tačke gledišta OSI modela most je operativan na nivou-veze (*link level*). Most prima okvire, proverava njihovu adresu, i na osnovu sadržaja svoje interne "tabele-prosledjivanja" odlučuje da li će primljeni okvir proslediti prema sledećoj mreži ili će ga izbaciti. Obično se mostovi koriste za spajanje dve mreže istog tipa, mada se oni mogu koristiti i za spajanje *Ethernet*-a sa *Token-Ring*-om na nivou veze. Ovakvi mostovi se nazivaju *transitional bridge* jer su u stanju da konvertuju bajt uređenu *Token-Ring* hardversku adresu na *Ethernet* adresu, i obratno. Obično se za potrebe upravljanja mostovima može dodeliti IP adresa. Na primer, SNMP (*Simple Network Management Protocol*) zahteva da uređaj ima IP adresu kako bi se isti pozvao i istestirao njegov status.

Tri standardna načina povezivanja lokalnih LAN-ova pomoću mostova prikazana je na Slici 2.



a) jednostavni most



Slika 8.2 Ostvarivanje lokalnih konfiguracija sa mostovima

Na Slici 2a) prikazana je jednostavna konfiguracija sa mostom koja povezuje dva *Ethernet* LAN-a. Ovakva konfiguracija se može ostvariti i pomoću repetitora sa izuzetkom da sav saobraćaj između DTE-ova u jednom LAN-u ne prolazi preko drugog LAN-a. Jedini saobraćaj koji se razmenjuje preko mosta između dva LAN-a je onaj koji zahteva inter-LAN komunikaciju.

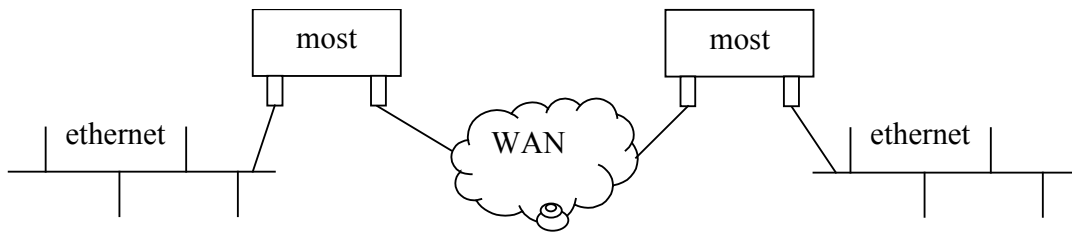
Na Slici 2b) prikazana su tri LAN-a povezana preko multiport mosta. U ovom slučaju most otvara paket, čita MAC adresu i komutira paket ka odgovarajućem portu na putu prema određenoj adresi. Obično su ovakvi mostovi samoučeći. Da bi poslao paket, most obično analizira sve pakete koje prima, a u tabeli izvornih adresa pamti portove sa kojih je primio te poruke. Za slučaj da se određena adresa ne nalazi u tabeli izvornih adresa tada most koristeći tehniku plavljenja (*flooding*) šalje paket ka svim portovima i otkriva tačan port ka kome je trebalo poslati poruku, pa nakon toga zapisuje broj porta u tabeli. Tabela se periodično, na svakih nekoliko minuta, čisti (briše) od neaktivnih adresa.

Standardno, mostovi su namenjeni za komutaciju paketa između LAN-ova. Da bi ostvarili ovaj cilj mostovi imaju ugrađenu mogućnost memorisanja-i-prosledjivanja (*store-and-forward*) paketa. Obično, mostovi koji povezuju lokalne mreže realizuju se samo za jedan protokol, i imaju osnovni zadatak da komutiraju i filtriraju intra-LAN-ov saobraćaj. No treba istaći da karakteristika memorisanja-i-prosledjivanja obezbeđuje mostovima mogućnost da mogu ostvariti i konverziju protokola. Na Slici 2c) prikazana je jedna takva multiport multiprotokol konfiguracija koja povezuje LAN-ove tipa *Ethernet* i *Token Ring*. Konverzija protokola se vrši na nivou 2 OSI modela (*data link level*).

8.2.1. Udaljeni mostovi

Kao što je prikazano na Slici 2 mostovi se uglavnom koriste kao mesta centralizovanog povezivanja kod lokalnih LAN konfiguracija, tako da se mostovi sa Slike 2 nazivaju lokalni mostovi (*local bridges*). No mostovi se mogu koristiti i za međusobno povezivanje dve mreže preko WAN linka (vidi Sliku 3). Ovakvi

mostovi se nazivaju udaljeni mostovi (*remote bridges*) i obavezno operišu u parovima (jedan na jednom kraju, a drugi na drugom kraju WAN-a).



Slika 8.3 Konfiguracija sa udaljenim mostom

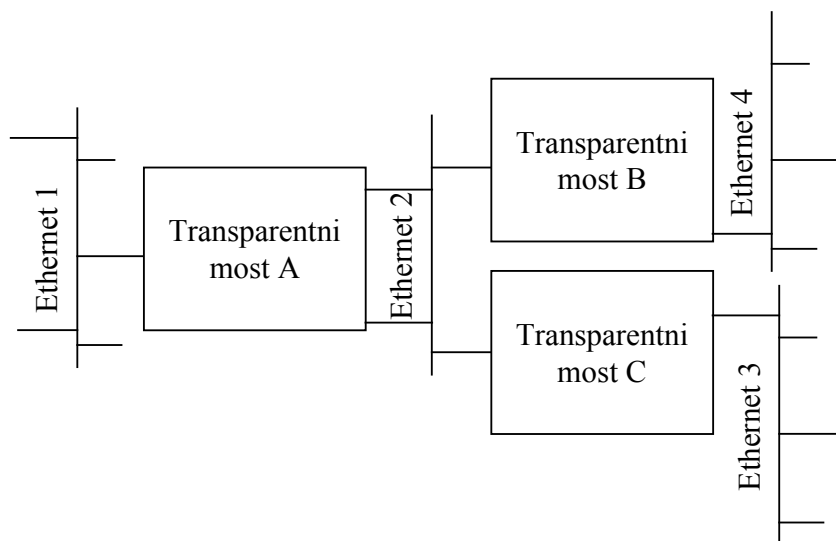
LAN-ovi se mogu povezati preko mostova kako u slučajevima kada se koristi topologija stabla tako i u slučajevima kada se koristi topologija rešetke. Mreže povezane preko mostova koriste sledeća dva algoritma za rutiranje: (a) *spanning tree algorithm* kod povezivanja dva *Ethernet* LAN-a; i (b) *source routing algorithm* kod opvezivanja *Token-Ring LAN*-ova.

8.2.2. Transparentni mostovi

Na Slici 4 prikazano je kako su četiri LAN-a povezana u topologiji tipa stablo pomoću tri mosta. Svaki most zna samo svog suseda i transparentan je drugim mostovima i LAN-ovima, pa se zbog toga ovakav tip mosta naziva transparentni most (*transparent bridge*).

Transparentni most, za potrebe rutiranja, koristi *spanning tree algorithm* koji određuje adrese memoriše u tabeli portova. Kada paket pristigne do mosta, most ga šalje ka portu koji vodi ka odredištu. Most ne zna gde se nalazi odredišni LAN, on samo zna gde je povezan susedni čvor za tu odredišnu adresu.

Transparentni most saznaje podatke o rutiranju putem povratnog saznanja (učenja). Naime, kada paket pristigne na port, on pamti izvornu adresu porta i pridružuje adresu tog porta svojoj tabeli rutiranja. Zatim prosledjuje paket ka odredištu. Ako se odredišna adresa ne nalazi u tabeli rutiranja, transparentni most predaje *broadcast* poruku da bi pribavio (saznao) gde se nalazi (misli se na kom portu) ta adresa.



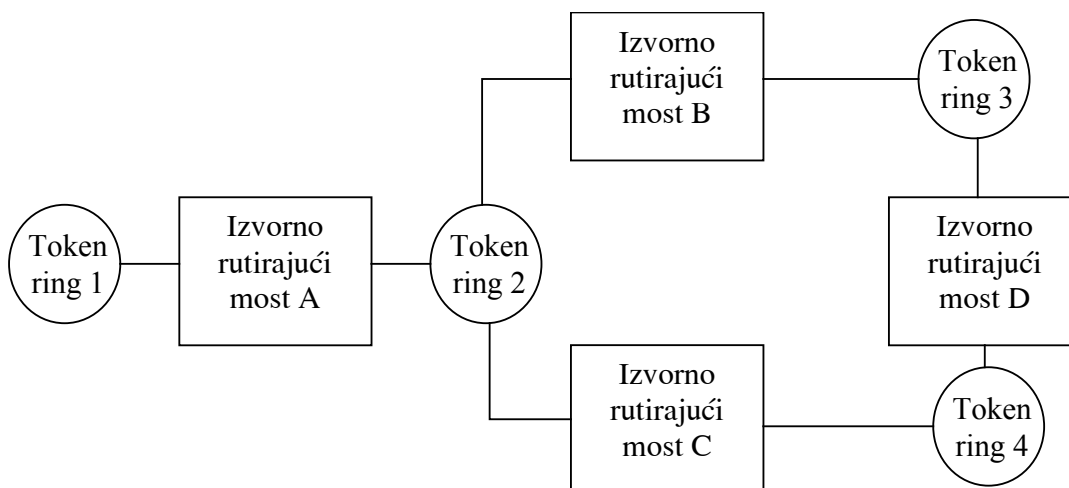
Slika 8.4 Mreža zasnovana na transparentnim mostovima

Kao što se vidi sa Slike 4 transparentni mostovi se koriste kod topologije tipa stablo, pri čemu zatvaranje petlji nije dozvoljeno. Kod ovakve topologije jedan od čvorova se ponaša kao glavni, a konkretno je to transparentni most A. U slučaju kada između dva LAN-a postoji više fizičkih puteva, algoritam za rutiranje bira samo jedan da bi se eliminisale zatvorene petlje. Tako na primer, ako transparentni most B, sa Slike 4, ima vezu na *Ethernet 3* i *Ethernet 4* tada će se zatvoriti (formirati) petlja forme *Ethernet 3* - transparentni most B - *Ethernet 2* - transparentni most C - *Ethernet 3*. *Spanning-tree-algorithm* neće dozvoliti da transparentni most B predaje i prima pakete preko veze na *Ethernet 3*.

Sagledajmo sada put poruke koju host povezan na LAN3 treba da preda *host-u* koji je povezan na LAN4. Poruka po stablu putuje naviše do mosta A, a zatim naniže do LAN4. Na osnovu ovoga je evidentno da most koji se nalazi na vrhu stabla podržava najveći saobraćaj.

8.2.3. Izvorno rutirajući mostovi

Izvorno rutirajući mostovi (*source routing bridge*) se koriste, kako je prikazano na Slici 5, da povežu *token-ring* LAN-ove. Ovi mostovi koriste algoritam rutiranja koji se bazira na tome da izvorište poznaje kompletnu putanju do odredišta. Paket koji prolazi kroz mrežu pored informacije sadrži i odredišnu adresu, tako da čvorovi na putu samo prosledjuju paket. Tabela rutiranja poruka kroz mrežu može biti zapamćena na nekom centralnom serveru ili u svakom izvorno-rutirajućem-mostu. Putanja se određuje korišćenjem tehnike *broadcast* paketa putem plavljenja paketa kroz mrežu.



Slika 8.5 Mreža bazirana na izvorno-rutirajućim mostovima

8.2.4. Nedostaci mostova

I pored jednostavne realizacije koja indirektno diktira nisku cenu, mostovi imaju veći broj nedostataka koji bi se mogli iskazati kroz sledeća zapažanja:

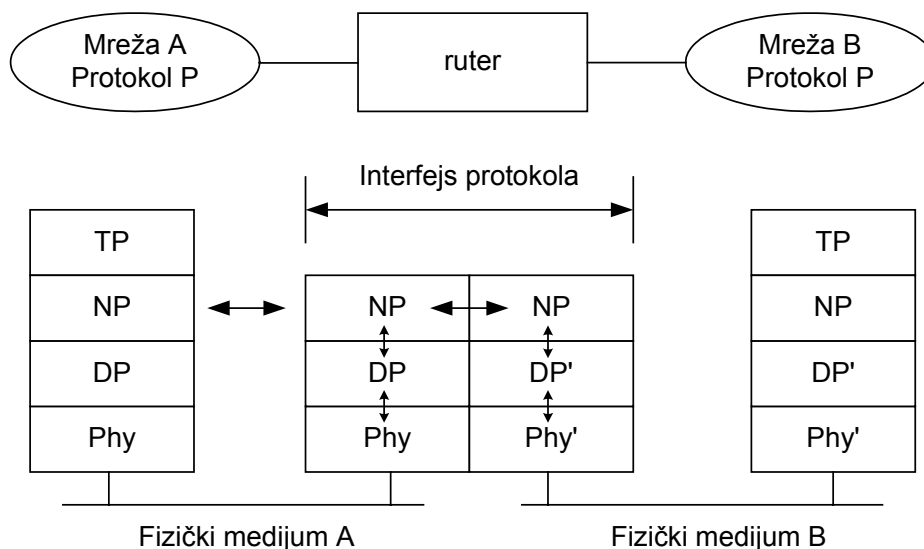
1. Mostovi se koriste kod mreža specijalne namene. U ostalim slučajevima, zbog razlika koje postoje u algoritmima za rutiranje, komunikacija između medija (mreža) koji koriste različite protokole je otežana. Tipično je to za komunikaciju između LAN-ova tipa *Ethernet*, *Token Ring* ili *FDDI*.

2. Odluka o prosledjivanju poruka (posebno se to odnosi na *spanning tree*) nije dovoljno sofisticirana. Naime, u slučaju kada u tabeli rutiranja ne postoji zapis koji ukazuje na to gde treba usmeriti paket, most će proslediti paket svim portovima sa izuzetkom onog sa koga je paket primljen.
3. *Spanning-tree* protokol ne dozvoljava formiranje petlji.
4. Mostovi se uglavnom koriste za povezivanje mreža istog tipa. Izuzetak su mostovi koji vrše transliranje.

Imajući u vidu nabrojane nedostatke (posebno se to odnosi na povezivanje različitih tipova mreža, kakve su *Ethernet* i *Token Ring*) logički se nameće zaključak da je efikasnije u čvornim tačkama mreže ugradjivati uređjaje koji će raditi na mrežnom nivou. Takvi uređjaji su ruter i *gateway* koji mogu rutirati pakete između različitih medijuma (*FDDI* i *Ethernet*) i različitih mreža (koriste različite mrežne protokole) na jedan transparentan način.

8.3. Ruteri

Kao što i samo ime ukazuje ruter se, kao uređjaj, koristi za usmeravanje paketa kroz mrežu. Svaki ruter u računarskoj mreži poseduje znanje koje se odnosi na moguće puteve koje paketi podataka treba da predju na putu od izvorišta do odredišta. “Znanje” o usmeravanju paketa (rutiranju) ugradjuje se u ruting tabelu koja se periodično ažurira i memoriše u bazi podataka rutera. Da bi odredio portove kojima se pridružuju odredišne adrese ruter koristi *broadcast* šemu koja se bazira na ARP-u (*Address Resolution Protocol*). Ono što je takodje važno i treba istaći je to da ruter može čitati sadržaje paketa podataka koji pristižu na dati port na osnovu čega može da odredi izvorišnu i odredišnu adresu, zatim tip podataka, kao i vreme prijema paketa. Nakon toga, koristeći ruting tabelu, ruter usmerava paket podataka ka jednom ili većem broju izlaznih portova, tj. prema odredišnoj adresi. Naime, paket podataka se usmerava ka portu ako se poruka prenosi od izvorišta ka odredištu, ili ka većem broju portova ako je paket tipa *broadcast* ili *multicast*. Na Slici 6 prikazana je konfiguracija rutera sa pratećom protokol arhitekturom. Uočimo da je protokol na nivou mreže, NP, identičan za obe mreže, dok protokoli na nivou veza, DP, i fizičkom nivou, Phy, mogu biti različiti.



Slika 8.6 Konfiguracija rutera

U odnosu na mostove ruteri su univerzalniji jer podržavaju rad i topologiju mreža tipa petlje. U cilju optimiziranja balansa opterećenja i smanjenja cene komuniciranja u rutere se ugrađuju različiti algoritmi za rutiranje. Od brojnih algoritama, sigurno najčešće korišćeni je otvori-prvo-najkraći-put (OSPF - *open-shortest-path-first*). Kod ovog algoritma svaki ruter *broadcast*-uje pakete-zahteva na svim vezama sa kojima je on povezan. Ostali ruteri u mreži potvrđuju zahtev i ponavljaju proces. Na ovaj način kreira se distribuirana baza podataka koristeći algoritam najkraćeg puta i vrši se ažuriranje baze uvek kada se izvrši promena u mrežnoj konfiguraciji.

Mrežni menadžer kreira ruting tabele sa ciljem da optimizira mrežne performanse u odnosu na nekoliko parametara, kakvi su cena ruta, kašnjenje, propusni opseg, i dr.

Performanse koje se pre svega odnose na propusnost i kašnjenje kod mreža zasnovanih na mostovima su bolje u odnosu na mreže bazirane na ruterima jer ruteri imaju ugrađenu obradu na još jednom nivou (mrežnom nivou). Ipak mostovi se koriste samo kod onih mreža kod kojih se zahteva veća brzina rada. Sa druge strane koncepcija rada rutera se zasniva na obradi informacije na mrežnom nivou OSI modela, čiji je glavni cilj uredjivanje paketa (tj. prenos paketa od izvorišnog do odredišnog čvora). To znači da se degradacija performansi rutera u odnosu na most, sa aspekta više cene i manje brzine, može “progutati” u odnosu na beneficije koje se dobijaju. Na primer, glavni zadatak IP protokola je da obezbedi da će se paketi isporučiti do korektnog odredišta, pri čemu se ne treba brinuti o integritetu podataka. Ova aktivnost je prepuštena drugim, višim nivoima protokola.

8.4. Prednosti rutera u odnosu na mostove

Kao prvo, ako se protokolom specifikira adresa na mrežnom nivou (*network layer*) tada se taj paket može rutirati (tipično je to za TCP/IP, DECnet i dr.).

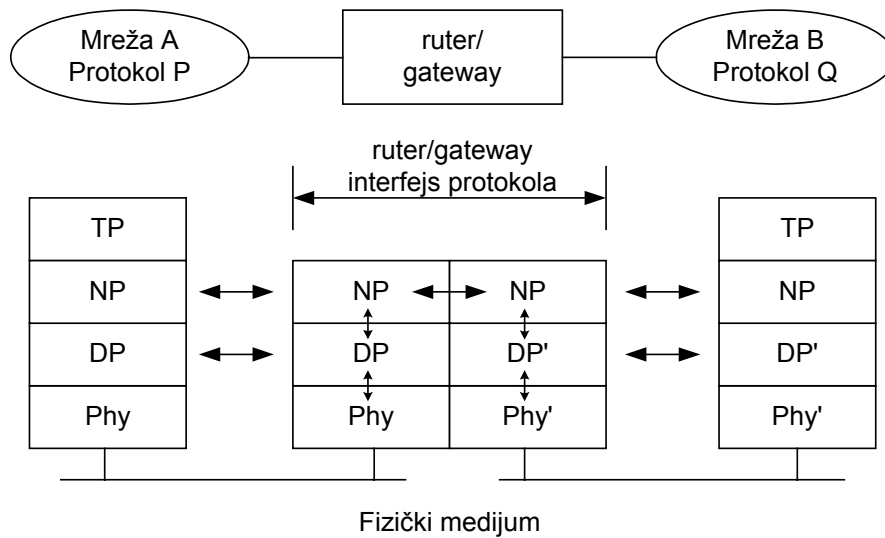
I pored toga što ruteri imaju svoje nedostatke koji se pre svega odnose na smanjenje propusnosti i povećanje latentnosti kod prenosa poruka, ipak oni u odnosu na mostove imaju sledeće prednosti:

1. Odluke o prosledjivanju paketa koje se donose od strane rutera variraju od jednostavnih do jako sofisticiranih. Odluku o tome na koji način se vrši rutiranje donosi projektant mreže i implementator.
2. Kod mreža kod kojih se vrši rutiranje poruka topološke strukture mreže tipa petlje su poželjne. Nasuprot tome, kod mreža zasnovanih na mostovima, petlje treba razbiti (raskinuti) tako da saobraćaj ne prolazi kroz redundantne veze. Protokol rutiranja kod mreža zasnovanih na ruterima je taj koji određuje kako će se saobraćaj usmeravati. Rutirajuće petlje kod ovih mreža se mogu javiti i evidentno je da su one “škodljive po zdravlje mreže”. Ipak određene mere zaštite kao što je brojanje broja preskoka poruke i vreme-života smanjuju beskonačni kružni obilazak poruke kroz petlju.
3. *Broadcast* saobraćaj karakterističan za nivo veze (*link layer*) ne prolazi kroz ruter.
4. Pomoću rutera je moguće spojiti različite tipove mreža.

8.5. Gateway i konvertori protokola

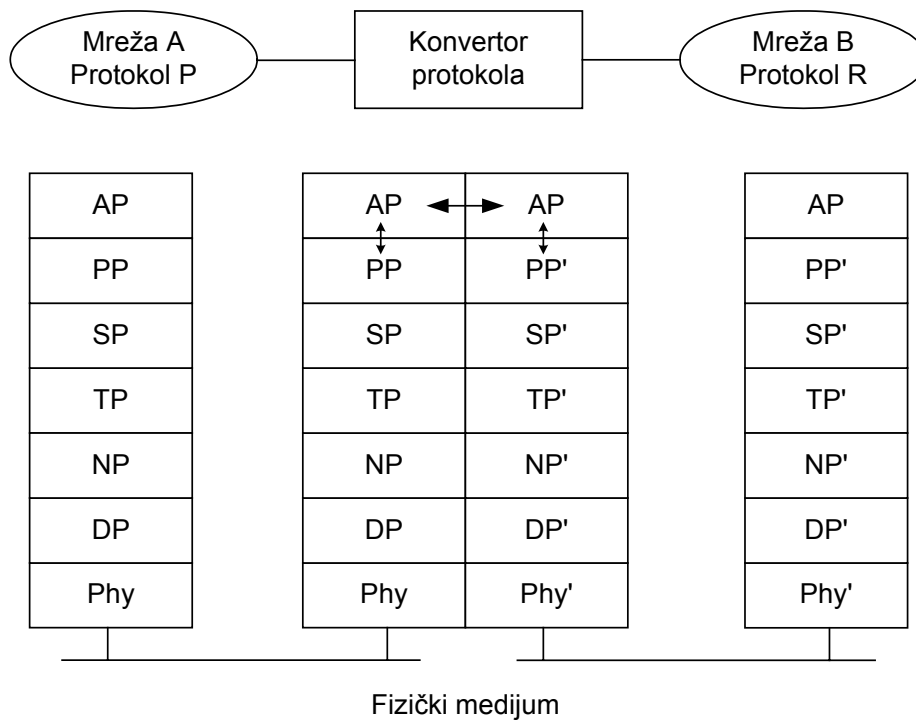
Danas u svetu postoji veliki broj autonomnih mreža, svaka sa svojim različitim hardverom i softverom. Autonomne mreže međusobno se razlikuju sa stanovišta većeg broja aspekata - algoritmima za rutiranje, implementiranim protokolima, DNS (*Domain Name Servers*), procedurama za administriranje i vođenje politike mreže i dr. No nezavisno od nabrojanih razlika, korisnici jedne mreže često žele da komuniciraju sa korisnicima povezanim na drugu mrežu. Kada su dve ili veći broj mreža uključene (učestvuju) u aplikaciji, režim rada između ove dve mreže nazivamo umrežavanje (*internetworking*). Često se pojam *internet* ili *internetwork* koristi za složene mreže tipa LAN/WAN/LAN. Svaki građivni blok internet mreže (LAN ili WAN) naziva se submreža ili subnet.

Shodno ISO terminologiji, uređaj koji međusobno povezuje dve mreže naziva se IS (*Intermediate System*) ili IWU (*Interworking Unit*). Alternativno, imajući u vidu da je jedna od glavnih funkcija koja se obavlja od strane IS-a vezana za rutiranje, ovakav jedan uređaj se često naziva ruter. S obzirom da ruter obezbeđuje vezu između dve autonomne mreže on se još naziva i *gateway*. Sa druge strane, protokol-konvertor je uređaj tipa IS koji povezuje dve mreže koje rade sa potpuno različitim stek-protokolima, kao na primer ISO-stek i protokol-stek nekog specifičnog proizvođača. Razlika između rutera i protokol-konvertora prokazana je na Slici 7.



Napomena: TP - transparentni protokol
 NP, NP' - mrežni protokoli
 DP, DP' - protokoli na nivou veze
 Phy, Phy' - protokoli na fizičkom nivou

a)



Napomena: *AP* - aplikacioni protokol
PP, PP' - prezentacioni protokoli
SP, SP' - protokoli sesija
TP, TP' - transportni protokoli
NP, NP' - mrežni protokoli
DP, DP' - protokoli na nivou veza
Phy, Phy' - protokoli fizičkog nivoa

b)

Slika 8.7 Konfiguracija *gateway*-a (a); konfiguracija protokol-konvertora (b)

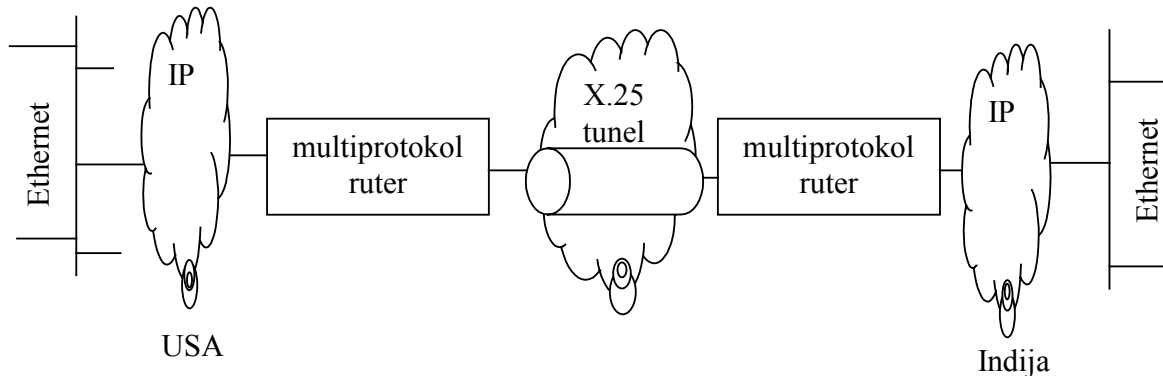
Na Slici 7a) usvajamo da je svaka mreža različitog tipa i da ruter/*gateway* ima ugrađen različit skup mrežnih protokola, po jedan na svakom svom portu. Veza rutera/*gateway*-a naziva se port. Konverzija protokola sa *gateway*-a se vrši na mrežnom nivou tako da se u *gateway*-u pored funkcije konverzije nadgrađuje i funkcija rutiranja. Gornji nivoi - transportni do aplikacioni - su identični za sve hostove i DTE-ove priključene na sistem.

Protokol-konvertor, kako je to prikazano na Slici 7b), obavlja konverziju protokola na nivou aplikacije. Naime pored razlike u mrežnim protokolima kod oba tipa mreže, postoje i razlike u višim nivoima stek protokola. Veoma često, u stručnoj literaturi, pojam protokol-konvertor se poistovećuje sa pojmom *gateway*. U suštini, pojam ruter se koristi za uređaj koji usmerava poruke u okviru istog autonomnog sistema (mreže) koristeći pri tome iste algoritme za rutiranje, dok uređaj tipa ruter koji usmerava pakete do odredišta koje se nalazi van granica autonomnog sistema (mreže) se naziva *gateway*-ruter.

8.6. Multiprotokol ruteri i tuneliranje

Kao što smo već napomenuli, komunikacija između dve autonomne mreže se ostvaruje pomoću *gateway*-a. Alternativno rešenje se bazira na korišćenju multiprotokol-rutera. U opštem slučaju, tuneliranje kao tehnika se koristi kada su odredišna i izvorišna stanica locirane u sličnim mrežama, ali podaci prolaze

preko mrežnih sistema koji mogu imati različite protokole. U ovom slučaju okviri podataka u medju-mrežama ne prolaze kroz fazu konverzija-protokola nego se samo kapsuliraju i tuneliraju za prenos kroz medju-mreže. Na Slici 8 prikazano je kako se ostvaruje komunikacija izmedju dva *Ethernet* LAN-a na IP mrežama. Ilustracije radi jedan LAN može biti lociran u USA, a drugi u Indiji. Ono što je važno uočiti je to da se podaci prenose preko Evrope koja za prenos podataka koristi X.25 paketski-komutiranu mrežu. Multiprotokol-ruter na bližem kraju kapsulira IP pakete u X.25 okvire i prenosi ih ka multiprotokol-ruteru na daljem kraju. Multiprotokol-ruter na daljem kraju dekapulira okvire i ponovo ih usmerava kroz svoju mrežu kao IP pakete.



Slika 8.8 Tuneliranje koristeći multiprotokol-rutere

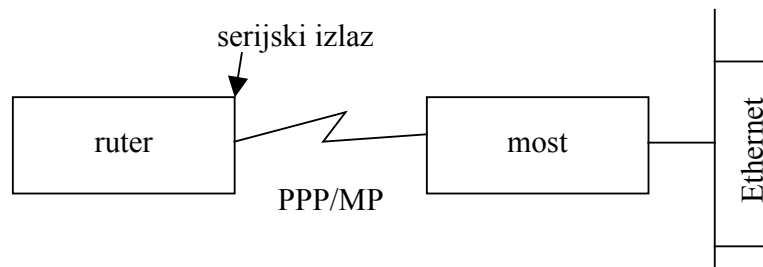
8.7. Polu-mostovska konfiguracija rutera

Često je poželjno ostvariti komunikaciju tipa tačka-ka-tački. Na primer, kada stanica u nekom većem poslovnom objektu ili zgradi komunicira sa ISP-om (*Internet Service Provider*) neophodno je koristiti PPP (*Point-to-Point Protokol*). Ovo se ostvaruje standardnom metodom za multi-protokol datagrame preko veze tipa tačka-ka-tački. Ovaj metod komunikacija je proširen na PPP MP (*Multilink Protokol - MP*). Koristeći MP protokol datagrami se mogu deliti, sekvencirati, prenositi preko većeg broja paralelnih veza (linija), i ponovo kombinovati kako bi se dobila izvorna poruka. MP protokol povećava propusnost i efikasnost komunikacione veze tipa tačka-ka-tački.

Povećanjem Interneta mala preduzeća kao i mali IPS-ovi povezuju se na Internet preko serijskih veza koristeći metod poziva putem birajna (kao kod telefona). Oni zahtevaju vezu Internet - Lokalni LAN samo kada za to postoji potreba, tj. ne postoji potreba za permanentnom vezom koja se ostvaruje preko zakupljenih linija. Da bi se ostvarile ovakve veze koristi se veći broj različitih PPP protokola. Najpoznatiji je SLIP (*Serial Link Internet Protocol*). Polu-mostovi (*half-bridges*) su uređaji koji obezbeđuju metod za povezivanje LAN-a preko mosta na ruter. Na Slici 9 prikazana je jedna polu-mostna konfiguracija. Port rutera koji je povezan na most je konfigurisan kao serijski interfejs ka PPP polu-mostu. Interfejs funkcioniše kao virtuelni čvor na *Ethernet*-ovoj podmreži mosta. Serijskom interfejsu u okviru *Ethernet* podmreže je pridružena IP adresa. Na ovaj način je *Ethernet* podmrežna adresa 155.55.123.1, serijskom interfejsu na ruteru bi trebalo dodeliti adresu 155.55.123.5.

Kada paket namenjen *Ethernetu* stigne do rutera, on se konvertuje do Ethernet pakete, kapsulira u PPP okvire i šalje ka liniji na kojoj je povezan *Ethernet* most. U suprotnom smeru, *Ethernet* paketi se

kapsuliraju u PPP okvire, izdvajaju od strane rutera, koji ih konvertuje u IP pakete, i usmerava ih (rutira) kroz Internet.



Slika 8.9 Konfiguracija polu-mosta

8.8. Komutatori

Komutatori (*switches*) operišu na fizičkom nivou OSI referentnog modela. Već smo napomenuli da, kao komponenta, komutator ostvaruje fizičku vezu između ulaznih i izlaznih portova i da bitovi i bajtovi ulaze i izlaze u istom redosledu. Mostovi i ruteri koriste funkciju komutacije u toku rutiranja paketa. Najveći broj tehnologija komutiranja, koje se danas koriste, bazira se na poluprovodničkoj tehnologiji (svi komutatori se prave kao VLSI IC), a što je interesantnije, brzina komutiranja postaje sve veća i veća. Današnje brzine prenosa podataka po mrežama reda su 10 Gbps-a. To znači da su performanse mreže najvećim delom određene brzinom komutiranja i multipleksiranja podataka kroz uređjaje.

U zavisnosti od načina korišćenja, funkcije komutiranja se mogu podeliti na komutaciju kola i komutaciju paketa. Telefonske komunikacije koriste tehniku komutiranja kola. Karakteristično za ovaj tip prenosa je to što se veza od jednog korisnika do drugog uspostavlja pre početka prenosa (ovaj postupak se naziva *call-setup*). U toku telefonskog razgovora veza se ne raskida nezavisno od toga da li se u datom trenutku kroz tu vezu prenosi govorni signal ili ne prenosi.

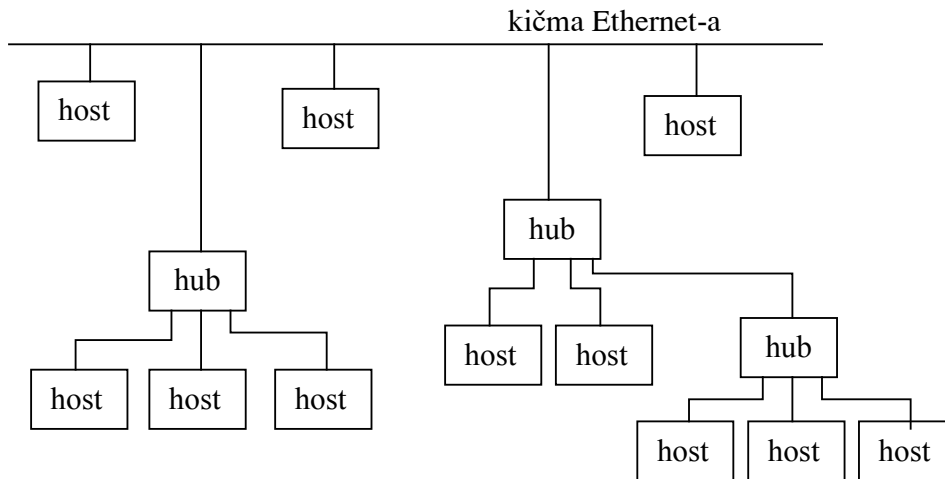
Sa druge strane računarski saobraćaj je po svojoj prirodi paketni (*bursty*) i zbog toga koristi tehniku paketne komutacije. Osnovna prednost ovog načina prenosa je ta što se propusni opseg kanala daleko efikasnije koristi. Podaci se dele na pakete i svaki paket se komutira nezavisno. Podaci se multipleksiraju od strane većeg broja izvora tako da dostupan propusni opseg kanala postaje deljiv za veći broj korisnika.

Kod rutera se obično koristi paketna komutacija. Obim paketa se bira (zavisi od slučaja do slučaja) tako da ruter radi najefikasnije. Cela poruka koja potiče od izvorišta se deli na veći broj paketa i prenosi kroz mrežu. Put svakog paketa kroz mrežu ne mora da je isti tako da paketi mogu pristizati na odredište van redosleda. Na odredištu paketi se uređjuju po redosledu. Ovaj tip servisa se naziva *datagram* servis. Drugi tip prenosa koji se koristi kod paketne komutacije bazira se na rad sa virtuelnim kolima. O ovoj problematici više detalja biće dato kasnije.

8.9. Habovi

Ethernet je prvobitno zamišljen sa topologijom tipa magistrala. Veoma brzo su projektanti mreže uočili da ovakav pristup nije najpogodniji kod povezivanja hostova u velikim zgradama. Šta više, prekid na kابلu uzrokuje da sistem postane neoperativan.

Daleko povoljnija strategija povezivanja je ona koja se bazira na topologiji tipa zvezda, posebno kod velikih zgrada, tj. onih instalacija koje koriste ožičenje slično telefonskom. Da bi se na određeni način uspešno rešili problemi, koristi se varijanta *Etherneta* tipa zvezda koja se zasniva na razvodu sa upredenim kablovima. Centar zvezde u ovom slučaju se naziva *hub* (vidi Sliku 10). *Hub* povezuje veći broj hostova, a deluje kao repetitor.

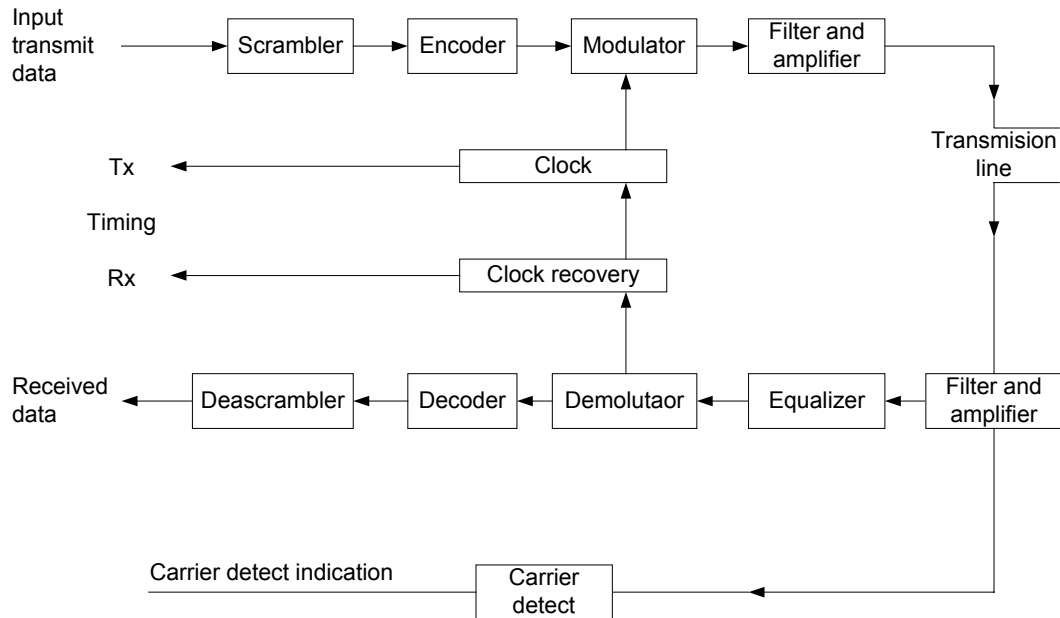


Slika 8.10 Ethernet koji koristi *hub*-ove

U suštini *hub* se ponaša kao multiport repetitor (broj portova može biti 4, 8, 16). To znači da kada *host* predaje tada *hub* prema ostalim vezama deluje (se ponaša) kao repetitor, prosledjujući podatke na principu bit po bit. Ako dva hosta pokušaju sa predajom u istom trenutku dolazi do kolizije.

8.10. Dodatak C: Struktura tipičnog modema

Savremeni modemi su veoma kompleksni uređaji jer pored modulatora i demodulatora sadrže veliki broj drugih specifičnih kola. Blok-dijagram jednog pojednostavljenog tipičnog modema prikazan je na slici C1.



Slika C1 Tipični modem

Funkcija gradivnih blokova je sledeća:

1) Predajna strana:

- a) *Scrambler* - prekôdira (*randomizes*) ulazni niz podataka sa ciljem da ne dozvoli pojavu dugog niza binarnih nula i jedinica
- b) *Encoder* - kôdira podatke sa ciljem da obezbedi zaštitu od grešaka u prenosu
- c) *Modulator* - konvertuje ulazni digitalni signal u analogni signal
- d) *Filter and amplifier* - obezbedjuje da signal koji se prenosi telefonskom linijom ima odgovarajući propusni opseg i zadovoljavajuću amplitudu.

2) Prijemna strana:

- a) *Filter and amplifier* - ograničava propusni opseg na odgovarajući nivo sa ciljem da redukuje efekte šuma i pojačava ulazni signal na nivo koji je adekvatan za pobudu demodulatora.
- b) *Equalizer* - kompenzira prenosne karakteristike telefonske linije koje se odnose na grupno kašnjenje i amplitudna izobličenja
- c) *Demoluator* - vrši konverziju analognog signala u digitalnu formu
- d) *Decoder* - proverava da li se javila greška u prenosu i izbacuje redundantne bitove koji su insertovani na predajnoj strani
- e) *Descrambler* - vraća podacima početnu formu.

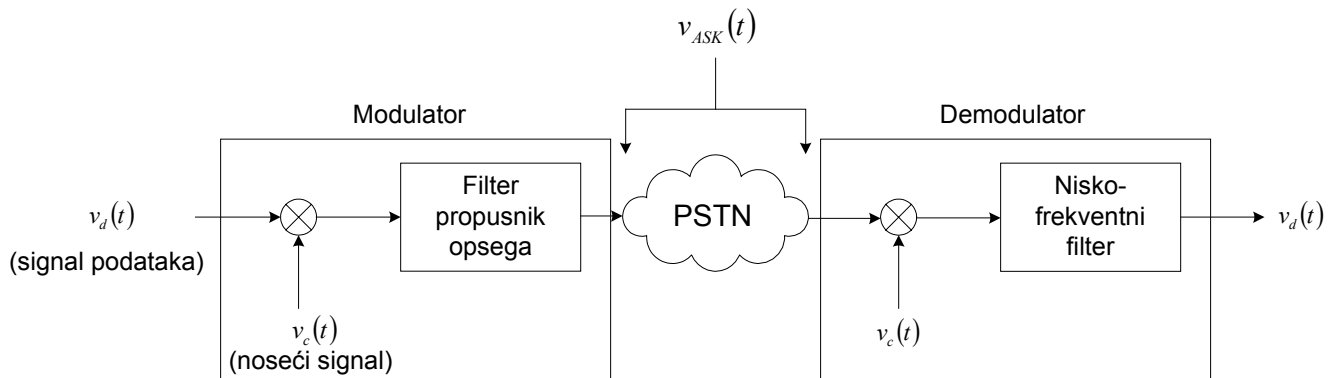
- f) *Clock/clock recovery* - obezbeđuje da tajming (vremenska sinhronizacija) predajnih i prijemnih podataka bude konzistentna.

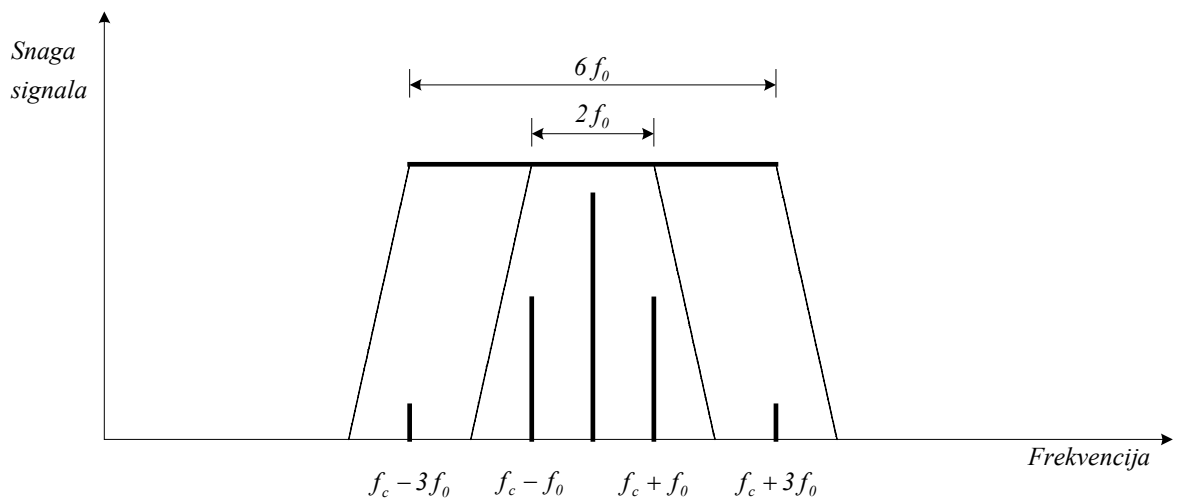
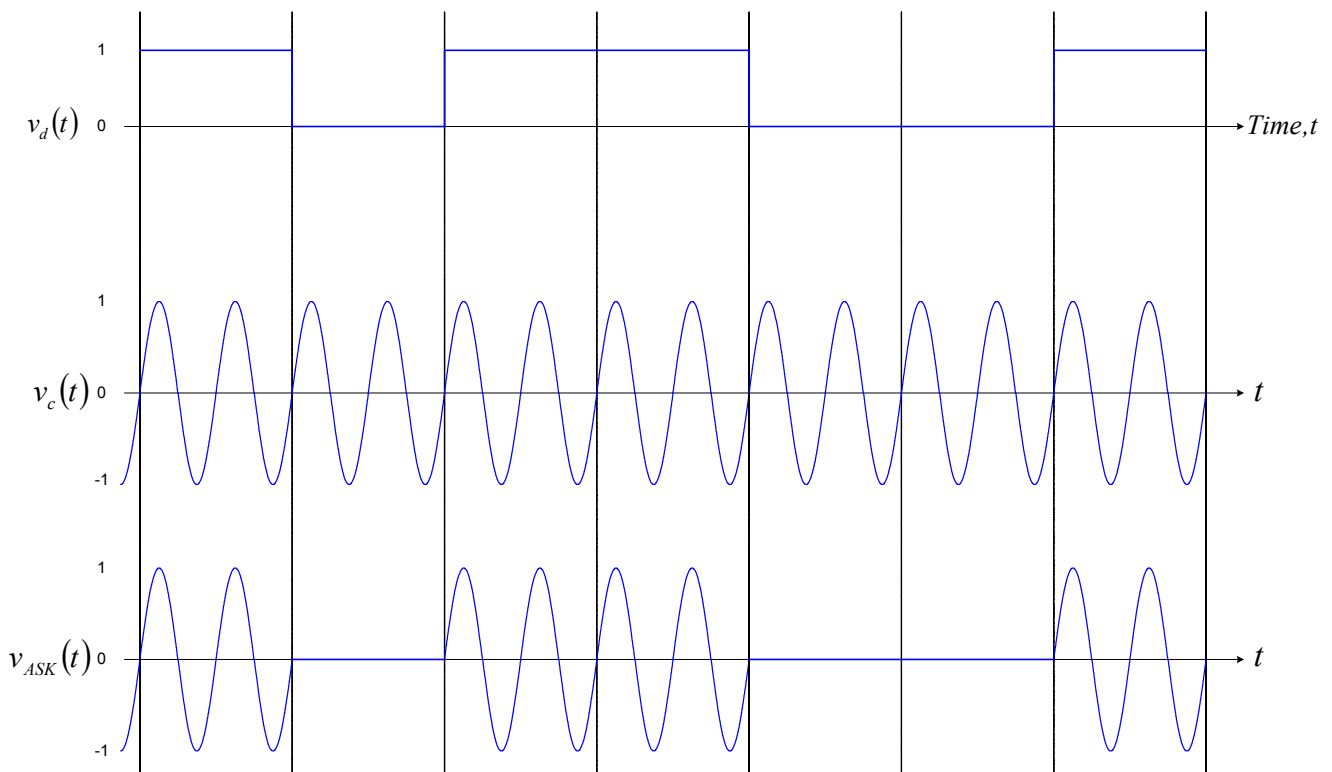
8.11. C.1 Modulacija

Kod modema se koriste tri osnovna tipa modulacije radi konverzije binarnog signala preko javne telefonske mreže (PSTN), a to su: amplitudna (ASK-*amplitude-shift keying*), frekventna (FSK-*frequency-shift keying*) i fazna (PSK-*phase-shift keying*).

A. ASK

Princip rada ASK-a prikazan je na slici C2 a), a odgovarajući talasni oblici na slici C2 b). U suštini amplituda čistog tona (*audio ton*) komutira između dva nivoa brzinom određenom od strane bitske brzine digitalnih podataka kojih treba preneti. Čisti-ton naziva se noseća frekvencija (*carrier frequency*).





$f_0 = \text{Osnovna frekventna komponenta} = 1/2 \text{ brzine prenosa (Hz)}$

Slika C2 ASK; (a) blok šema; (b) talasni oblici; (c) propusni opseg

Ako je amplituda nosećeg signala jedinična tada se on može zapisati u formi:

$$v_c(t) = \cos \omega_c t$$

gde je ω_c noseća frekvencija u radijanima po sekundi.

Unipolarni periodični signal podataka, $v_d(t)$, jedinične amplitude i osnovne frekvencije ω_0 može se predstaviti pomoću *Fourier*-ovog niza kao:

$$v_d(t) = \frac{1}{2} + \frac{2}{\pi} \left[\cos \omega_0 t - \frac{1}{3} \cos 3\omega_0 t + \frac{1}{5} \cos 5\omega_0 t - \dots \right],$$

dok se ASK može matematički predstaviti sledećim izrazom:

$$v_{ASK}(t) = v_c(t) \cdot v_d(t)$$

Uvedeći odgovarajuće zamene, dobijamo:

$$v_{ASK}(t) = \frac{1}{2} \cos \omega_c t + \frac{2}{\pi} \left[\cos \omega_c t \cdot \cos \omega_0 t - \frac{1}{3} \cos \omega_c t \cos 3\omega_0 t + \dots \right]$$

Kako je:

$$2 \cos A \cos B = \cos(A - B) + \cos(A + B)$$

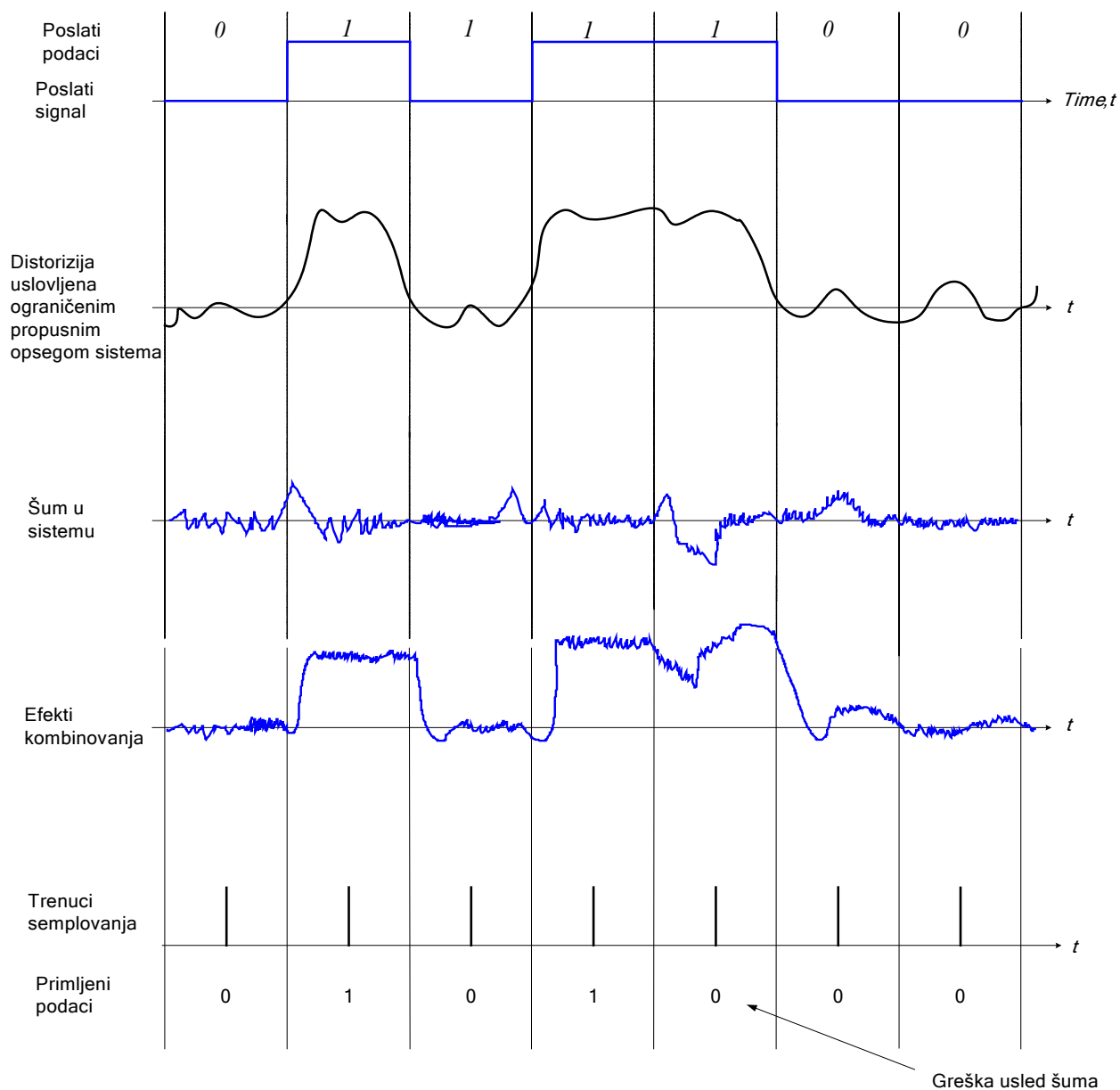
to zamenom imaćemo

$$v_{ASK}(t) = \frac{1}{2} \cos \omega_c t + \frac{1}{\pi} \left[\cos(\omega_c - \omega_0)t + \cos(\omega_c + \omega_0)t - \frac{1}{3} \cos(\omega_c - 3\omega_0)t - \frac{1}{3} \cos(\omega_c + 3\omega_0)t + \dots \right]$$

To znači da spektar ASK signala čini glavni nosioc i dva simetrična bočna opsega.

Osnovna prednost ASK predstavlja jednostavnost implementacije. Ali, na žalost, šum ima dramatičan efekat na amplitudu prijemnog signala tako da se greške mogu lako javiti (vidi sliku C3).

Prenos podataka – Mrežne komponente



Slika C3. Uticaj šuma na digitalni signal

Primer

Usvajimo da se koristi amplitudna modulacija. Proceniti propusni opseg potreban kanalu da se izvrši prenos sledećih bitskih brzina: 300 bps, 1200 bps i 4800 bps, u sledećim slučajevima:

- (a) za povorku impulsa 101010... prima se samo osnovna frekventna komponenta
- (b) za povorku impulsa 101010 prima se osnovna frekventna komponenta plus treći harmonik

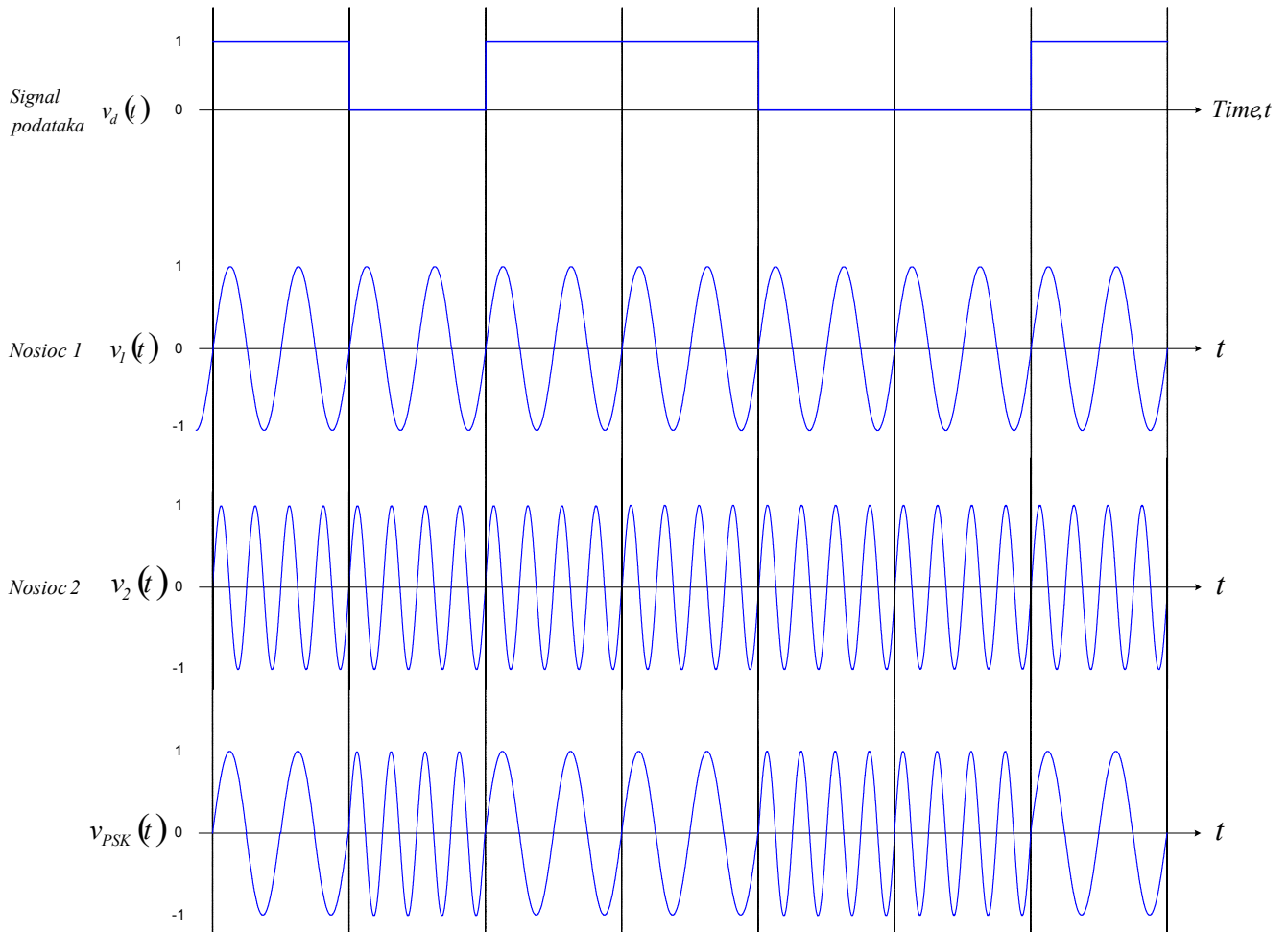
Odgovor

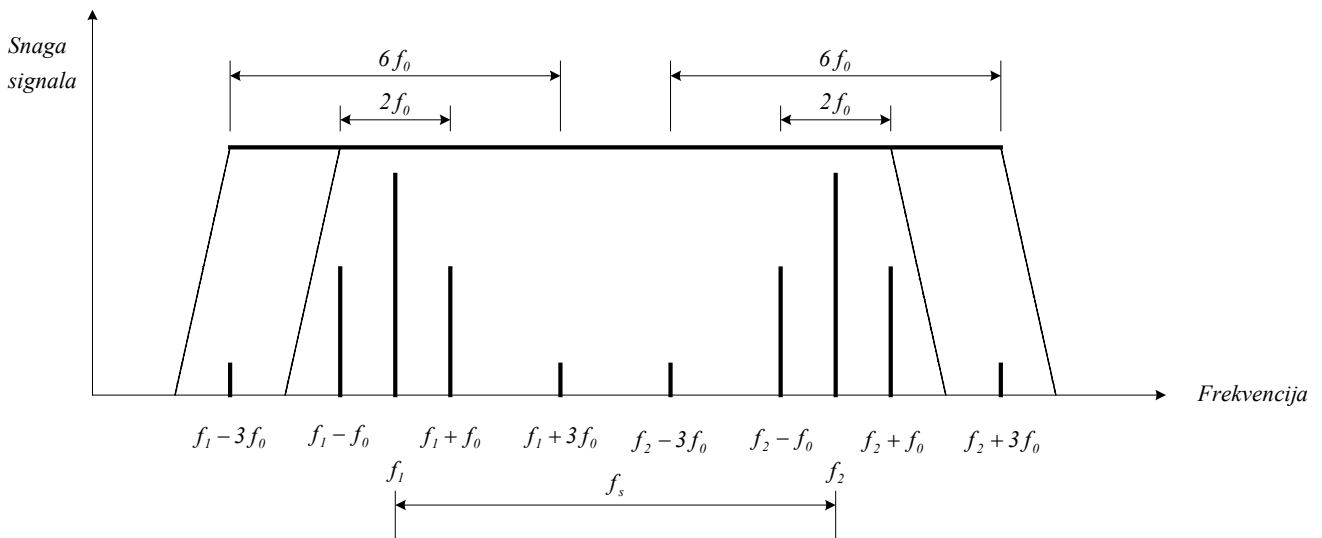
<i>Bitska brzina</i>	<i>300 bps</i>	<i>1200 bps</i>	<i>4800 bps</i>
Osnovna frekventna komponenta	150 Hz	600 Hz	2400 Hz
treći harmonik	450 Hz	1800 Hz	7200 Hz
Propusni opseg samo osnovne komponente	300 Hz	1200 Hz	4800 Hz
Propusni opseg osnovne komponente i trećeg harmonika	900 Hz	3600 Hz	14400 Hz

Naglasimo da je propusni opseg kanala javne telefonske mreže 3000 Hz. Shodno tome samo bitsku brzinu od 300 bps je moguće ostvariti ako se prenosi i treći harmonik. Bitska brzina od 1200 bps je moguće postići samo ako se prenosi osnovna komponenta. Bitsku brzinu od 4800 bps kroz kanal javne telefonske mreže ne može se ostvariti korišćenjem ASK.

B. FSK

S obzirom da modemi prenose samo binarne jedinice i nule, podatke možemo predstaviti kao promene dva tona različitih frekvencija. Ova tehnika se naziva FSK. Razlika između dva nosioca se naziva *frequency shift* i prikazana je na slici C4. Operacija modulacije kod ove metode ekvivalentna je zbiru izlaza obadva izdvojena ASK modulatora; jedan koji radi na jedan nosioc i moduliše se originalnim signalom podataka, a drugi radi sa drugim nosiocem i moduliše se komplementom signala podataka.





f_0 = Osnovna frekventna komponenta = 1/2 brzine prenosa (Hz)

f_s = Frekvencija pomeranja

Slika C4 FSK; (a) princip rada; (b) propusni opseg

Propusni opseg FSK se može odrediti na osnovu sledećeg izraza:

$$v_{FSK}(t) = \cos \omega_1 t \cdot v_d(t) + \cos \omega_2 t \cdot v'_d(t)$$

gde su ω_1 i ω_2 frekvencije signala oba nosioca, a $v'_d(t)$ je komplement originalnog signala, $v_d(t)$. Matematički posmatrano $v'_d(t) = 1 - v_d(t)$ i ako usvojimo da je signal podataka periodični kao i da je njegova osnovna frekvencija ω_0 , tada imamo:

$$v_{FSK}(t) = \cos \omega_1 t \left[\frac{1}{2} + \frac{2}{\pi} (\cos \omega_0 t - \frac{1}{3} \cos 3\omega_0 t + \dots) \right] + \cos \omega_2 t \left[\frac{1}{2} - \frac{2}{\pi} (\cos \omega_0 t - \frac{1}{3} \cos 3\omega_0 t + \dots) \right].$$

Odgovarajućom zamenom dobija se:

$$v_{FSK}(t) = \frac{1}{2} \cos \omega_1 t + \frac{1}{\pi} \left[\cos(\omega_1 - \omega_0)t + \cos(\omega_1 + \omega_0)t - \frac{1}{3} \cos(\omega_1 - 3\omega_0)t - \frac{1}{3} \cos(\omega_1 + 3\omega_0)t + \dots \right] + \frac{1}{2} \cos \omega_2 t + \frac{1}{\pi} \left[\cos(\omega_2 - \omega_0)t + \cos(\omega_2 + \omega_0)t - \frac{1}{3} \cos(\omega_2 - 3\omega_0)t - \frac{1}{3} \cos(\omega_2 + 3\omega_0)t + \dots \right]$$

Oдавде se zaključuje da je propusni opseg FSK signala jednak zbiru dva ASK modulisana nosioca frekvencije ω_1 i ω_2 .

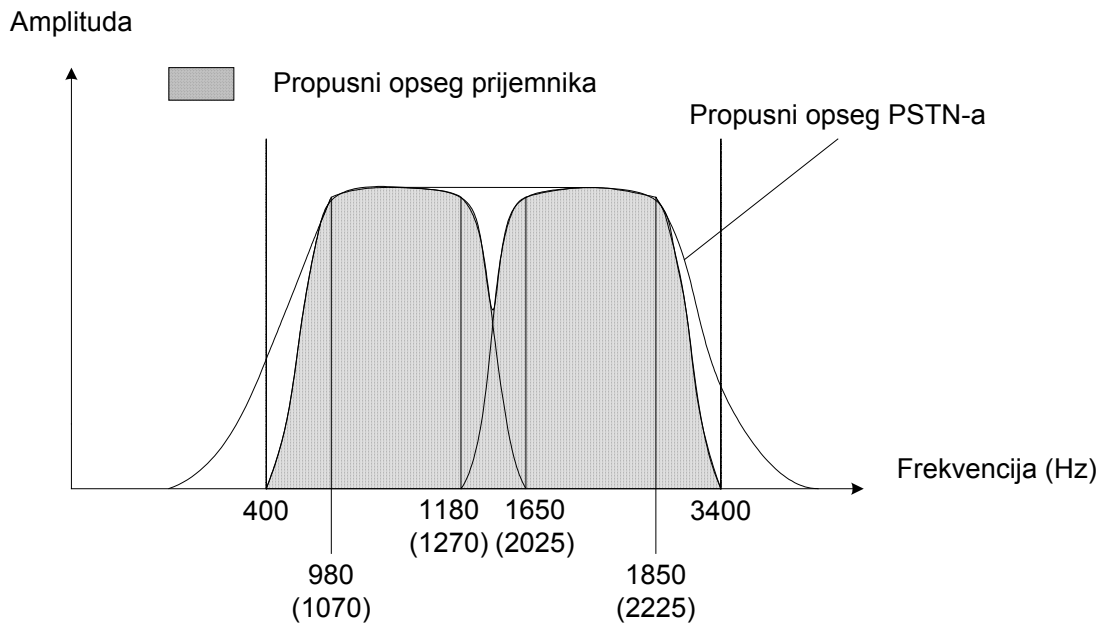
Kao što smo već naglasili, najviše frekvencije signala se generišu za binarnu sekvencu 101010... Kod FSK, svaki od signala binarne 1 i 0 moduliše poseban nosioc, a to znači da je minimalni propusni opseg potreban svakom nosiocu jednak polovini bitske brzine, što znači da je bitska brzina jednaka najvišoj osnovnoj frekventnoj komponenti svakog nosioca, f_0 , i jednaka je polovini one koja odgovara za ASK. Zbog toga ako usvojimo da se prima samo (najviša) osnovna komponenta signala tada ukupni propusni opseg

potreban kod FSK iznosi $4f_0$ plus frekventni pomeraj (*frequency shift*) f_s . Ipak pošto je f_0 polovina od one za ASK ukupan potreban propusni opseg je isti kao za ASK plus frekventni pomeraj. Na sličan način ako treba primiti treći par harmonika, propusni opseg biće $6f_0$ plus potrebni frekventni pomeraj.

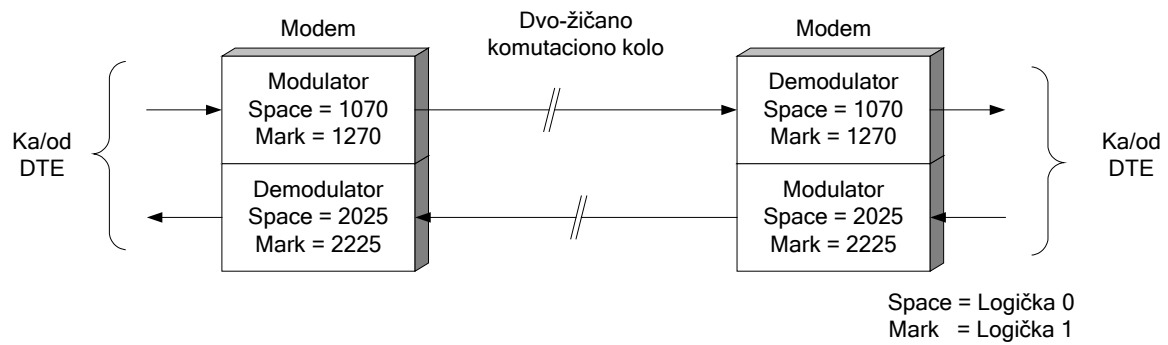
Na primer ako je maksimalna bitska brzina 600 bps, tada je maksimalna bitska brzina prenosa po nosiocu 300 bps. To znači da nosioc ima maksimalnu osnovnu frekventnu komponentu od 150 Hz. Frekventni spektar, će zbog toga, sadržati primarne bočne komponente koje su razmaknute na po 150 Hz na svakoj strani i to kod oba nosioca. Za slučaj da izaberemo da frekventni pomeraj između dva nosioca iznosi 400 Hz, tada će ovim pristupom da se obezbedi razmak od 100 Hz između primarnih bočnih komponentata za oba nosioca. Ovo znači da je ukupno potrebno propusni opseg od 900 Hz.

Propusni opseg kanala kroz javnu telefonsku mrežu je 3000 Hz, što ukazuje da je moguće prenositi dva kanala od po 900 Hz, po jedan za svaki smer prenosa.

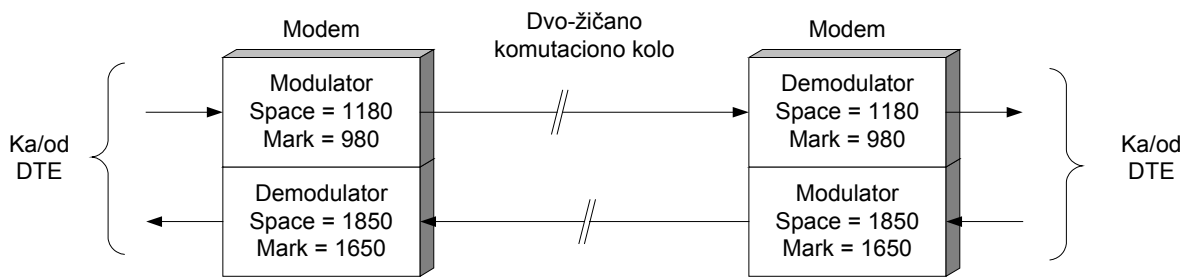
Na slici C5 prikazana je dodela frekvencija kod dva tipa FSK modema koji rade u potpunom-dupleksu pri brzini prenosa podataka između dva DTE-a od 300 bps. Jedna frekventna dodela definisana je od strane EIA za modem *Bell 103*, a druga od strane ITU za modem V.21. Uočimo da kod ovih modema osnovna frekventna komponenta koja prati svaki nosioc iznosi 75 Hz. Saglasno tome frekventni pomeraj od 200 Hz obezbedjuje 50 Hz razdvajanje između oba primerna bočna opsega.



a)



b)



c)

Slika C5 300 bps FSK modem za potpuni dupleks; (a) EIA-ova dodela frekvencija; (b) dodela frekvencija po ITU-T (V.21)

FSK modem koji je projektovan shodno standardu ITU-T V.21 (brzina prenosa podataka od 300 bps) koristi dodelu frekvencije kao na slici C6. Po dogovoru, modem koji inicira poziv (*call*) koristi niži nosioc frekvencije radi predaje, a više radi prijema.

	Mark	Space	Carrier
<i>Originating modem</i>	980Hz	1180Hz	1080Hz
<i>Answering modem</i>	1650Hz	1850Hz	1750Hz

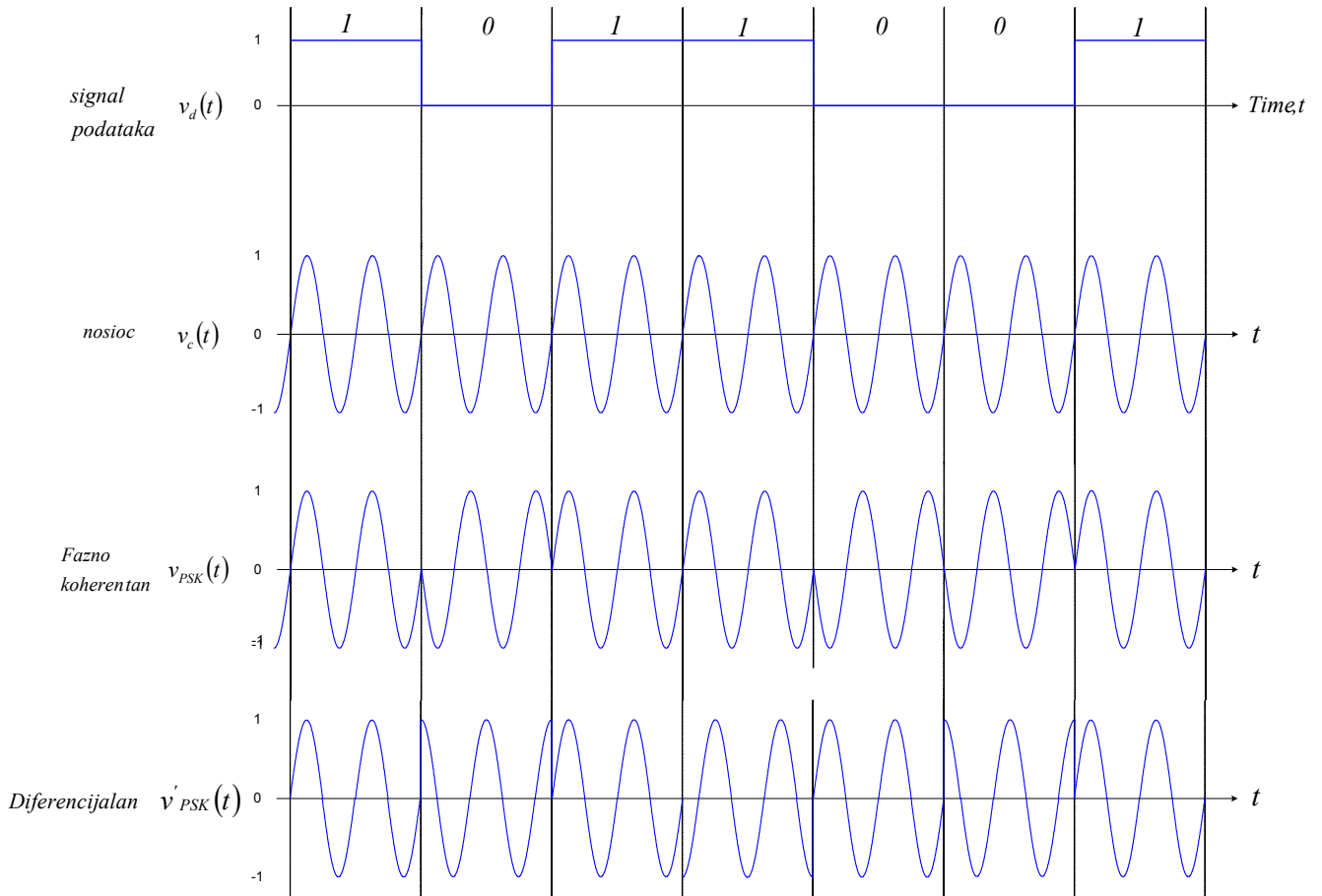
Slika C6 Dodela frekvencija od strane ITU-T za modem V.21

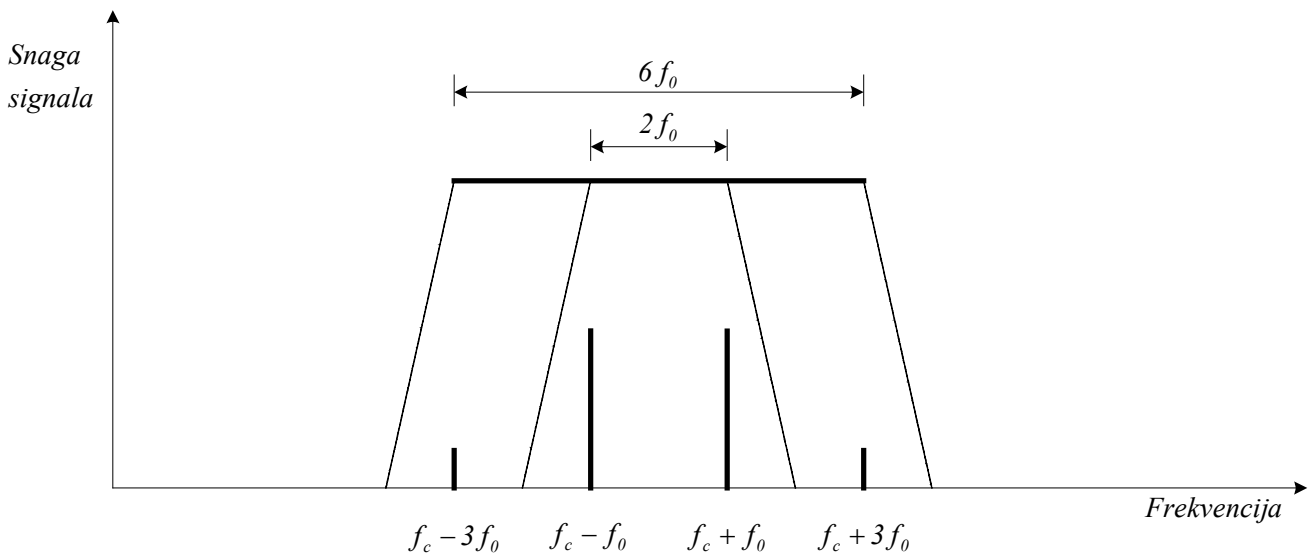
Frekventni spektar sadrži dva bočna opsega kod kojih su komponente razmaknute na po 150Hz na svakoj strani nosioca. Modem V.21 ima 200Hz razmak izmedju nosioca što je dovoljno da “priljubi” jedan uz drugi bočne opsege svakog nosioca. FSK modemi sa maksimalnom brzinom od 1200 bps imaju maksimalnu osnovnu komponentu od 600Hz i razdvajanje frekvencija od 1000Hz.

Za brzine podataka od 9600 bps maksimalna osnovna frekventna komponenta je 4800 Hz a to premašuje veličinu propusnog opsega dodeljenog telefonskom kanalu i zbog toga ovaj tip modulacije nije pogodan za ove brzine.

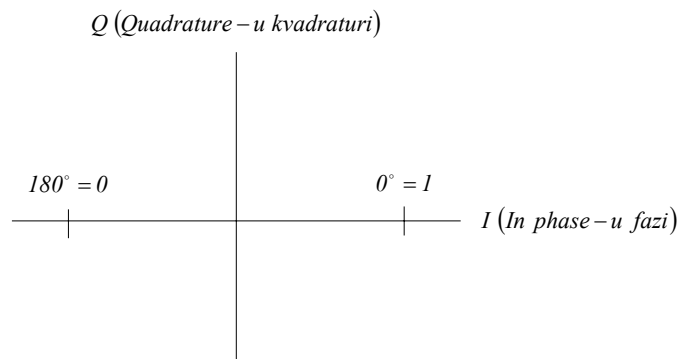
C. PSK

Kod PSK modulacije frekvencija i amplituda signala nosioca se održavaju konstantnim, ali se faza nosioca menja u ritmu predaje svakog bita podataka. Princip rada ove šeme prikazan je na slici C7.





$f_0 =$ Osnovna frekventna komponenta = $1/2$ brzine prenosa (Hz)



Slika C7 PSK; (a) princip rada; (b) propusni opseg; (c) fazni dijagram

Kao što se vidi sa slike C7, koriste se dva tipa PSK. Prvi koristi dva fiksna nosioca signala da bi predstavio binarnu 0 i 1 između kojih postoji fazna razlika od 180° . Imajući u vidu da je jedan signal samo invertovan drugi, ovaj postupak se naziva fazno-koherentni PSK (*phase-coherent PSK*). Nedostatak ove šeme je ta što je na prijemnom kraju potrebno raspolagati sa referentnim signalom kako bi se njegova faza upoređivala sa prijemnim signalom. U praksi, ova tehnika zahteva znatno kompleksija demodulatorska kola u odnosu na alternativno rešenje koje se naziva diferencijalni PSK (*differential PSK*). Kod ove šeme, fazni pomeraj se javlja na prelazu svake bit pozicije nezavisno od toga da li se predaje niz binarnih 0 ili 1. Fazni stav od 90° relativno u odnosu na tekući signal ukazuje da je binarna 0 naredni bit, dok fazni stav od 270° ukazuje da je to binarna 1. Kao rezultat, kolo demodulatora treba da odredi samo iznos svake fazne promene, a ne i apsolutnu vrednost.

Matematički, da bi odredili propusni opseg PSK predstavimo binarni signal u bipolarnoj formi tako da negativni nivo signala uzrokuje promenu faze nosioca od 180° . Bipolarni signal podataka jedinične amplitude i osnovne frekvencije ω_c može se predstaviti pomoću *Fourier*-ovog reda kao:

$$v_d(t) = \frac{4}{\pi} \left[\cos \omega_0 t - \frac{1}{3} \cos 3\omega_0 t + \frac{1}{5} \cos 5\omega_0 t - \dots \right]$$

dok za fazno modulirani signal imaćemo:

$$v_{PSK} = \frac{4}{\pi} \left[\cos \omega_c t \cdot \cos \omega_0 t - \frac{1}{3} \cos \omega_c t \cos 3\omega_0 t + \dots \right]$$

$$= \frac{2}{\pi} \left[\cos(\omega_c - \omega_0)t + \cos(\omega_c + \omega_0)t - \frac{1}{3} \cos(\omega_c - 3\omega_0)t - \frac{1}{3} \cos(\omega_c + 3\omega_0)t + \dots \right]$$

Analizom izraza zaključujemo: Ovaj signal ima isti frekventni spektar kao i ASK sa izuzetkom da ne postoji komponenta nosioca. Propusni opseg PSK signala je prikazan na slici C7 b). Ako usvojimo da se prima samo osnovna frekventna komponenta binarne sekvence 101010..., tada je minimalni propusni opseg potreban da se izvrši prenos jednak $2f_0$ i jednak je bitskoj brzini u Hz. Odsustvo nosioca, ipak ukazuje da se veća snaga prenosi bočnim opsezima, koji sadrže informaciju o podacima, a to čini da PSK bude mnogo otporniji na šum u odnosu na ASK ili FSK.

S obzirom da se kod PSK koristi jedinstveni nosioci, ali sa različitim faznim pomerajem koji odgovara svakoj binarnoj cifri, često se PSK, kako je prikazano na slici C7 c), predstavlja u formi faznog dijagrama. Fazni dijagram predstavlja sinusni nosioci kao jednu liniju - nazvanu *vektor* - čija dužina odgovara amplitudi signala. Linija rotira oko osa u smeru kazaljke na satu konstantnom brzinom koja je jednaka ugaonoj brzini ω_c . Binarna vrednost 1 se predstavlja kao vektor u fazi sa nosiocem dok binarna 0 kao vektor za 180° pomeren u odnosu na fazu nosioca. Ose se nazivaju *I (in-phase)* i *Q (quadrature)*.

Primer

Kod DPSK sistema faza nakon zadnjeg primljenog bita je $+90^\circ$. Proceniti fazni pomeraj na kraju sekvence 101101.

Odgovor

<i>binarni kod</i>	<i>fazni pomeraj</i>	<i>ukupni fazni pomeraj</i>
1	-90°	0°
0	$+90^\circ$	$+90^\circ$
1	-90°	0°
1	-90°	-90°
0	$+90^\circ$	$+0^\circ$
1	$+90^\circ$	-90°

Napomena: Faza se menja relativno u odnosu na fazu prethodno emitovanog bita. Za binarnu 0 faza se advansira 90° u odnosu na fazu prethodnog bita, a za binarnu 1 ona kasni 90° u odnosu na fazu prethodnog bita. To znači da komparator treba da upoređuje samo dva susedna bita kako bi odredio binarni kod.

8.12. C.2 Više-nivovske modulacione metode

DPSK metod se često proširuje sa ciljem da kôdira veći broj stanja. Naime, bolje se koristi propusni opseg ako se svaki signalni elemenat predstavi više od jednim bitom. Kao što je prikazano na slici C8, tehnika kodiranja nazvana QPSK (*Quadrature Phase Shift Keying*) uobičajeno koristi fazne pomeraje kao multiple od 90^0 . QPSK alternativno se naziva 4 PSK.

<i>fazni pomeraj</i>	<i>binarni kod</i>
+45°	00
+135°	01
+225°	11
+315°	10

Slika C8 QPSK

Kao što se vidi sa slike C8, svako signalno stanje se predstavlja sa dva umesto sa jednim bitom. Ovaj tip modulacione šeme implementiran je kod V.26 modema i radi sa bitskom brzinom od 2400 bps u direktnom smeru, a 1200 bps u suprotnom smeru. 8-fazna (3-bitna) verzija je takodje implementirana kod V.27 i radi sa bitskom brzinom od 4800 bps.

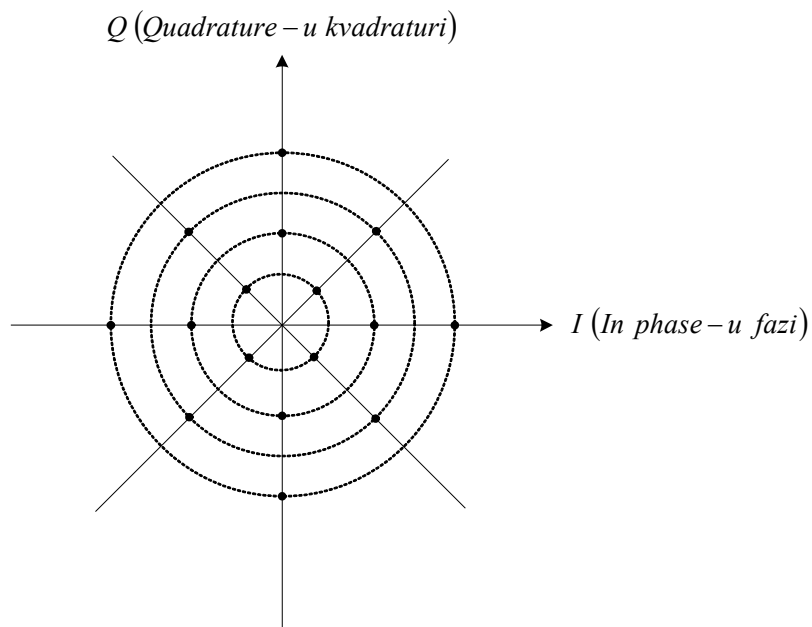
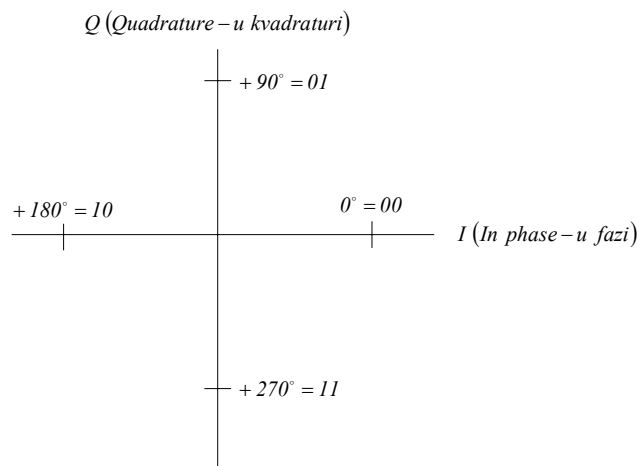
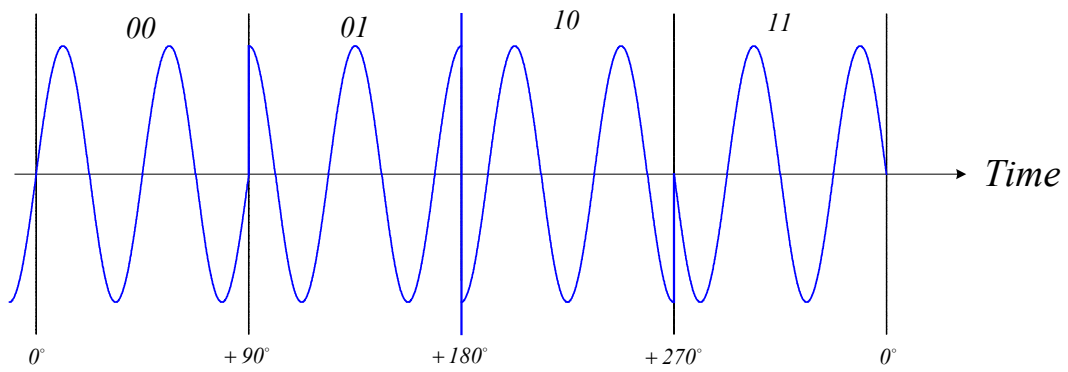
Primer

DQPSK (*Differential Quadrature Phase Shift Keying*) sistem se koristi za kodiranje niza podataka 00 11 01 10. Proceniti fazni pomeraj na kraju ako je fazni pomeraj na početku 0^0 .

Odgovor

<i>binarni kod</i>	<i>fazni pomeraj</i>	<i>ukupni fazni pomeraj</i>
00	+ 45 ⁰	+45 ⁰
11	+225 ⁰	+270 ⁰
01	+135 ⁰	+45 ⁰
10	+315 ⁰	0 ⁰

Kao što smo već uočili, veće bitske brzine kod QPSK se mogu postići korišćenjem 8- ili čak 16-faznim promenama. U praksi ipak postoji ograničenje koje se odnosi na broj korišćenih faza. Naime, smanjenje fazne razlike čini ovu tehniku u toku prenosa u većoj meri podložnoj šumu i faznim promenama (varijacijama) signala. Zbog toga da bi povećali bitsku brzinu prenosa još više, prirodno je uvesti pored varijacije (promene) faze svakog vektora još i promenu amplitude. Ovaj tip modulacije naziva se QAM (*quadrature amplitude modulation*). Primer je prikazan na faznom dijagramu sa slike C9 c). Ovaj tip modulacije ima 16 nivoa po signalnom elementu - koristi 4 simbola - i naziva se 16 QAM. Fazni dijagram se uobičajeno naziva *16-point constellation*.

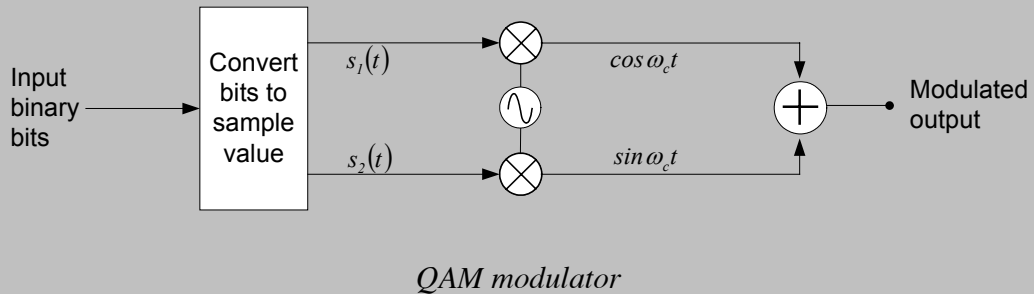


Slika C9 Alternativne modulacione tehnike; (a) talasni oblik kod 4-PSK; (b) fazni dijagram kod 4-PSK; (c) fazni dijagram kod 16-QAM

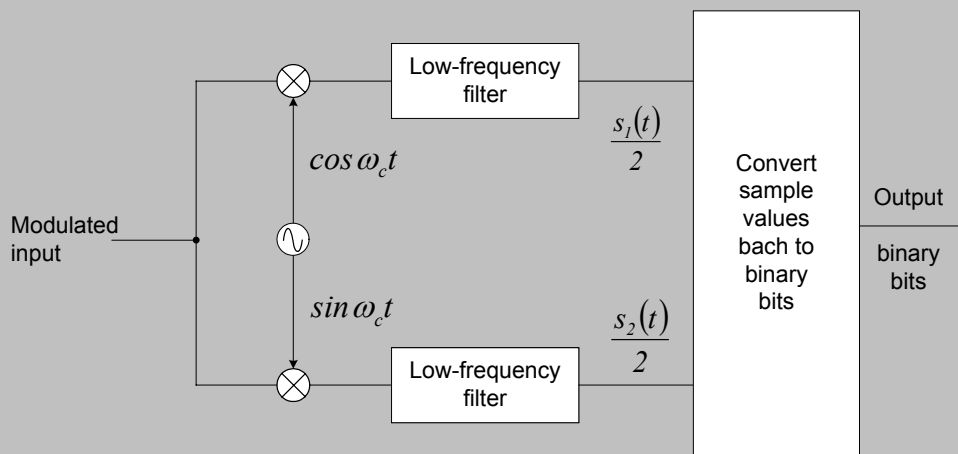
Primer

Nacrtati blok šeme: (a) QAM modulatora; (b) QAM demodulatora; (c) fazni dijagram *four-point QAM constellation*.

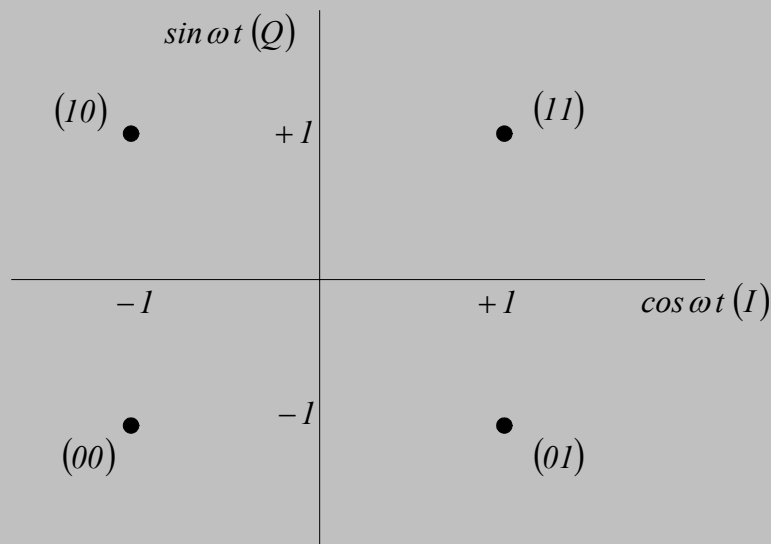
Odgovor



QAM modulator



QAM demodulator



Napomena: 0- kvdrturni
1- u fazi

Four-point QAM constellation

Kao što smo uočili, robustnost modulacionih šema na greške određena je od blizine (*proximity*) susednih tačaka u konstelaciji. Zbog toga kod slike C8, gde se koristi 8 faza, amplitudski nivoi signala koji se pridružuju susednim fazama čine se da budu različiti. Na ovaj način operacija izdvajanja (razlikovanja) na prijemnoj strani je manje podložna greškama, ali se ostvaruje po ceni uvođenja redundanse s obzirom da sva četiri amplitudska nivoa nisu izvodljiva za svaku fazu. Kao i kod svih modulacionih šema, pre modulacije, niz bitova prolazi kroz *scrambler* kolo koje konvertuje niz u pseudoslučajnu sekvencu. Ovakav način izvođenja ima za efekat smanjenja verovatnoće da će se uzastopni bitovi u povorci nalaziti na susednim bit pozicijama. Na prijemnom kraju nakon procesa demodulacije, povorka impulsa prolazi kroz *descrambler* kolo koje dovodi povorku bitova u prvobitni red (restaurira). Ovaj tip modulacije se koristi kod V.29 modema - za faksimil prenos - a postiže se bitska brzina od 9600 bps.

8.13. C.3 Trellis kôdna modulacija

Kada brzine kod prenosa podataka premaše vrednost od 9600 bps teško je kod QAM sistema smestiti sve *constellation* tačke dovoljno udaljene jedna od druge, a da se pri tome u potpunosti zaštitimo od uticaja grešaka. Da bi premostili ovaj problem razvijena je nova generacija modema koja se zasniva na TCM (*Trellis Code Modulation*). Kod QAM-a nesavršenost uzrokuje da se tačke prijemnog signala pomere u odnosu na svoju korektnu lokaciju u konstelaciji. Prijemnik zatim bira signalnu tačku u konstelaciji koja je najbliža u odnosu na prijemnu tačku. Ali kada su smetnje dovoljno velike i dovode da prijemna tačka nije više blizu signalnoj tački na predajnoj strani javiće se greška. TCM ima ugrađeno koder koji dodaje redundantne bitove svakom simbolu kako bi se minimizirale ove greške.

Kod V.32 bis modema predajnik deli dolazeće bitove u 6-bitne simbole. Dva od šest bitova kodiraju se binarnim konvolucionim koderom. Konvolucionni koder dodaje još jedan kôdni bit na svaka dva ulazna bita i na taj način formira po tri kôdirana bita za svaki simbol. Kao rezultat ovog načina kodiranja 6 ulaznih bitova konvertuje se u 7 kodiranih bitova i nakon toga preslikavaju u *constellation* signal sa 128 tačaka.

Redundansa koja je uvedena načinom kodiranja znači da su samo određene sekvence signalnih tačaka važeće. Ako sada nesavršenost uzrokuje da se signalna tačka pomeri, prijemnik će uporedjivati samo primljenu (observiranu) tačku u odnosu na važeću tačku i odabrati tačku koja je najbliža observiranom signalu. TCM modem je manje osetljiv na šum u odnosu na QAM modem, a procenat greške na izlazu (u odnosu na QAM) smanjuje za tri reda veličine.

TCM se standardno koristi kod V.32 modema za bitsku brzinu prenosa od 9600 bps, i kod V.32 bis modema za bitske brzine prenosa do 14400 bps. Bitske brzine prenosa od 19200, 24000 i 28800 bps su moguće kod V.34/V-Fast modema.

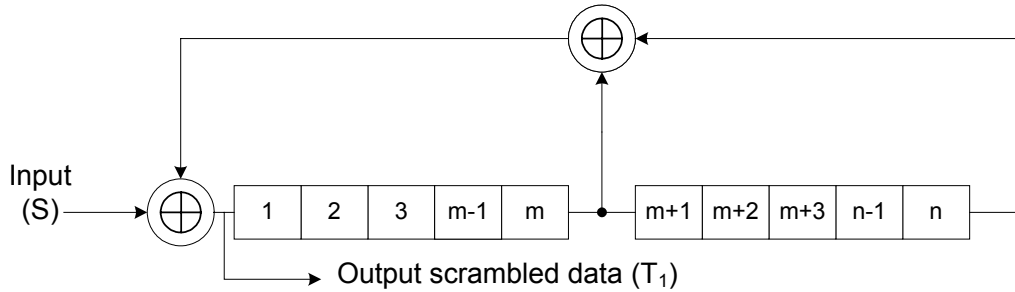
8.14. C.4 Skrembleri

Veliki broj komunikacionih podsistema, kakvi su modemi, rade bolje ako u toku prenosa koriste slučajne sekvence bitova. Nizovi binarnih jedinica ili nula ili periodične sekvence se mogu pojaviti na izlazu bilo kog izvora informacije. Da bi se povećala otpornost ovih sekvenci na šumove veoma često izvorne sekvence impulsa se propuštaju kroz jedno kolo koje se naziva *scrambler* čija je osnovna uloga da na svom slučaju generiše pseudoslučajnu povorku bitova. Sa ciljem da se spreči pojava neželjenih sekvenci kodiranih podataka *scrambler*-i se obično projektuju samo za linijske kôdove. Svaki *scrambler* na predajnoj strani prati *descrambler* kolo na prijemnoj strani. *Descrambler* se koristi da rekonstruiše ulazne podatke.

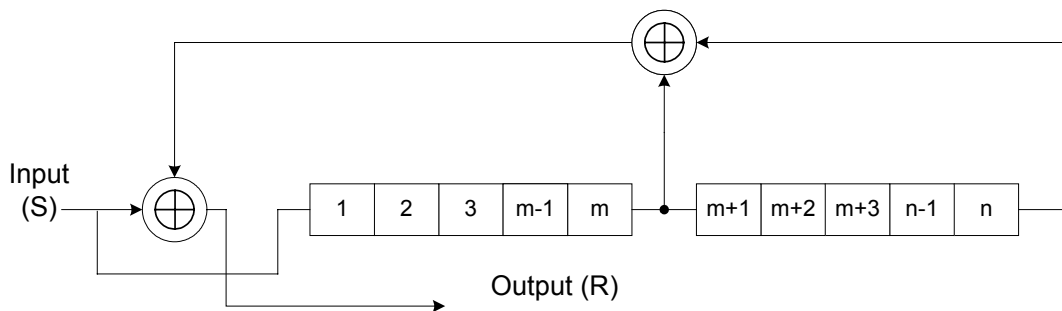
Tipično, strukturu *scrambler*-a čini *feedback shift register*, a odgovarajući *descrambler feed forward shift register*. Kod oba bloka (*scrambler* i *descrambler*), izlazi iz nekoliko stepeni pomeračkog registra sabiraju se svi zajedno po-modulu-2 (pomoću EXOR kola) a zatim se dodaju (koristeći ponovo aritmetiku po-

modulu-2) povorci podataka. Sadržaj pomeračkog registra pomera se bitskom brzinom određenom od strane sistema.

Blok šema tipičnog *scrambler*-a prikazana je na slici C10 a *descrambler*-a na slici C11.



Slika C10 Struktura *scrambler*-a



Slika C11 Struktura *descrambler*-a

Rad *scrambler*-a se može analizirati ako usvojimo da operator D predstavlja (odgovara) jedno-bitno kašnjenje. $D^1 S$ predstavlja sekvencu S zakašnjenja za jedan-bit, a $D^k S$ odgovara sekvenci S zakašnjenju za k -bitova.

Shodno slici C10, rad *scrambler*-a se opisuje kao:

$$\begin{aligned} T_1 &= S \oplus (T_1 D^m \oplus T_1 D^n) \\ &= S \oplus F T_1 \quad ; \text{ gde je } F = D^m + D^n \\ &= \frac{S}{I \oplus F} \end{aligned}$$

Izlaz *descrambler*-a se opisuje kao

$$\begin{aligned} R &= T_2 \oplus (T_2 D^m \oplus T_2 D^n) \\ &= T_2 \oplus F T_2 \\ &= T_2 (I \oplus F) \end{aligned}$$

U slučaju kada ne postoji greška $T_2 = T_1$ pa zbog toga *unscrambled* (neskremblovani) signal dat kao

$$R = \frac{S}{I \oplus F} \times (I \oplus F)$$

Izlaz je duplikat ulaza.

ITU-T specificira *scrambler* pomoću *tap* polinoma. Tako na primer *scrambler* sa slike C.10 ima povratnu vezu sa stepena m i n tako da se on opisuje pomoću *feedback tap*-ova kao (m,n) , ili je njegov *tap* polinom oblika:

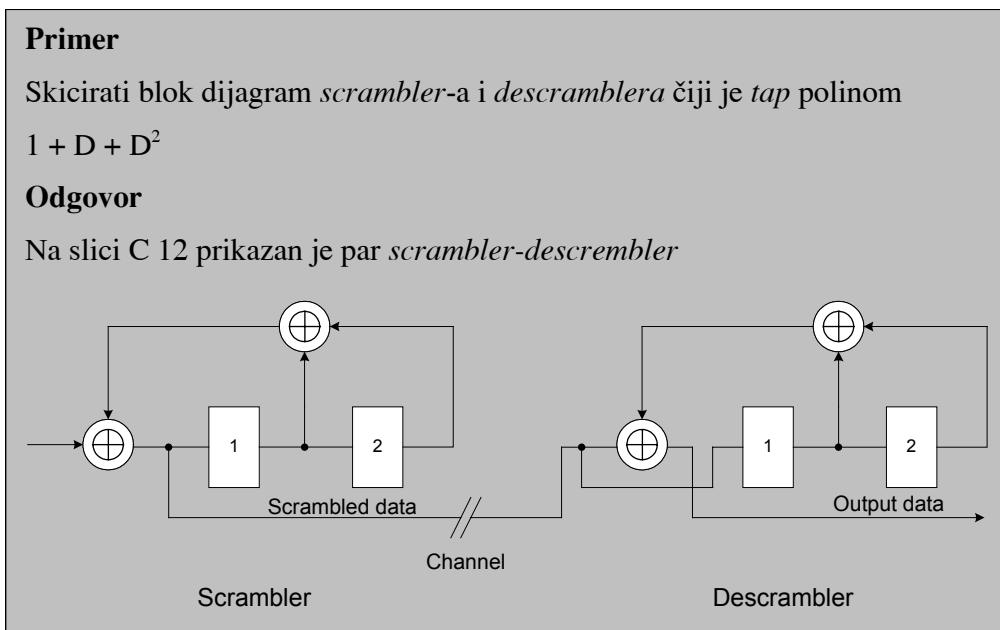
$$1 + D^m + D^n$$

Na osnovu prethodne analize se može ustanoviti da je

$$T_1 = S \quad (\text{podeljen tap polinomom})$$

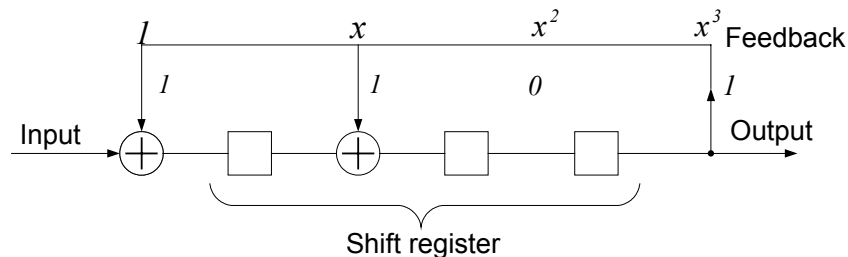
dok

$$R = T_2 \quad (\text{pomnožen tap polinomom})$$



8.15. C.5 Kola za deljenje po modulu – 2

Deljenje po modulu-2 se realizuje elektronskim kolom koje repetitivno pomera i oduzima. Ova ideja se lako implementira korišćenjem pomeračkih registara i *ExOR* kola (oduzimanje po modulu-2 identično je kao i sabiranje). Kolo koje deli sa $1 + x + x^3$ prikazano je na slici C.13.



Slika C13 Kolo za deljenje po modulu-2

Najbolji način da se razume rad ovog kola je kada se analizira njegov rad korak-po-korak (vidi sliku C.14). Kao prvo, odabraćemo ulazni bit oblik koji će dati poznati ostatak. U konkretnom slučaju koristićemo

ulaz 1100010. Ovaj broj je deljiv sa 1011 ($x^3 + x + 1$) pa će generisati ostatak 000. Količnik će se pojaviti na izlazu kola. Ipak, kod koderskih kola samo je ostatak od značaja.

Step J	Input on J th shift	Feedback on J th shift	Shift register after J th shift	Output after J th shift
0	-	-	000	0
1	1	0	100	0
2	1	0	110	0
3	0	0	011	1
4	0	1	111	1
5	0	1	101	1
6	1	1	000	0
7	0	0	000	

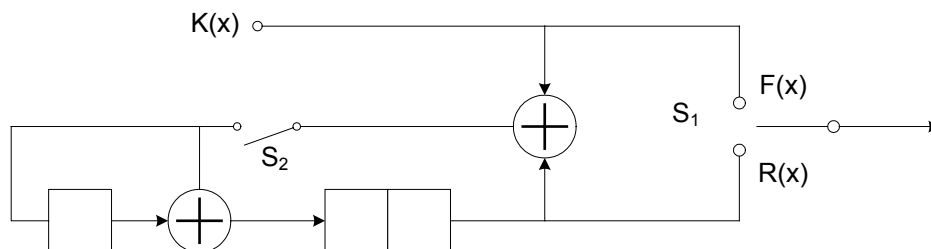
Slika C14 Deljenje po modulu-2

8.16. C.6 Kola za ciklično kodiranje i dekodiranje

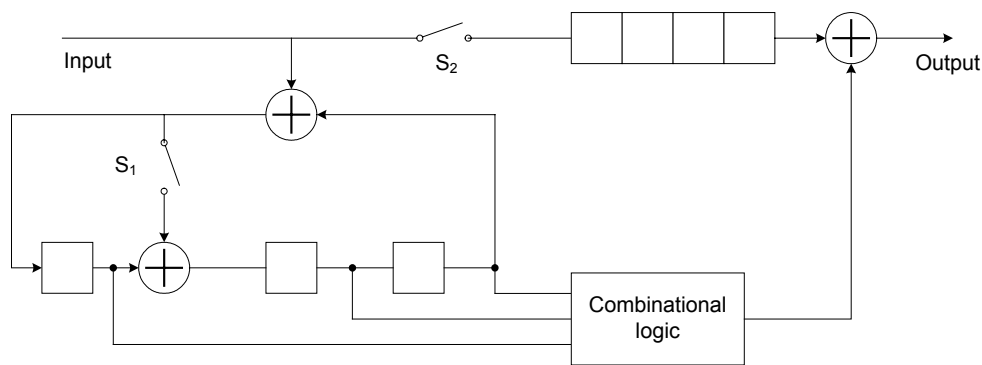
Kodiranje kod cikličnog kôda čini proces deljenja polinoma $F(x)$ (dolazeću informaciju sa nulama na *check bit* pozicijama) sa generator polinomom $G(x)$ sa ciljem da se odredi ostatak koji se zatim dodaje (pridružuje) $F(x)$ -u kako bi se dobila kodna reč $F(x) + R(x)$. Da bi dobili nulte *check* bitove, informacijski bitovi $K(x)$ pomeraju se c mesta pre početka procesa deljenja. Kodersko kolo kod kojeg je generator polinom $G(x) = 1 + x + x^3$ prikazano je na slici C.15.

Dovodjenjem informacijskih bitova $K(x)$ na ulaz kola u tački kako je to prikazano na slici C.15 a ne kao na slici C.13, ostvaruje se ekvivalentno pomeranje za c mesta (u konkretnom slučaju $c=3$). Inicijalno S_1 je u gornjem položaju, a S_2 je zatvoren, tako da se $F(x)$ istovremeno predaje i procesira u kolu za deljenje. Nakon k pomeranja, pomerački registar sadrži ostatak $R(x)$. S_1 se tada postavlja u donju poziciju, a S_2 otvara pa se na taj način predaje ostatak.

Sličnu strukturu ima dekodersko kolo prikazano na slici C.16. Inicijalno, informacijski bitovi se primaju u k -to stepeni pomerački registar sa S_1 i S_2 zatvorenim. Istovremeno, *check* bitovi ponovo se izračunavaju uz pomoć kola za deljenje, a zatim vraćaju nazad u kombinaciono logičko kolo. *Check* bitovi se primaju kada su S_1 i S_2 otvoreni i direktno se vode u kombinaciono logičko kolo gde se upoređuju sa iznova izračunatim *check* bitovima sa ciljem da se odredi sindrom. S_1 i S_2 se tada ponovo zatvaraju, a sindrom se koristi od strane kombinacionog kola da bi korigovao podatke na izlazu dekoderskog kola



Slika C 15 Kolo za ciklično kodiranje



Slika C16 Kolo za ciklično dekodiranje

8.17. C.7 Ekvilajzer

Da bi se kompenzovale degradacije telefonske linije (kompenzovala amplitudno frekventna i fazna karakteristika) u modem se umeće kolo koje se naziva *equalizer*. Funkcija ekvilajzera je da koriguje amplitudno frekventnu i faznu karakteristiku kanala. Idealan ekvilajzer je onaj na čijem se izlazu dobija signal (koji je prošao kroz kanal i ekvilajzer) čija je amplitudno frekventna karakteristika ravna (nema slabljenja), a faza se linearno menja sa frekvencijom (u opsegu od interesa za prenos), tj. da je grupno kašnjenje konstantno. Da bi se ostvario ovaj cilj modem treba unapred da zna kakav je frekventni odziv kanala. Na nesreću, uvek kada se modem odazove na poziv kanala, velika je verovatnoća da on prima signal sa različitog puta (fizički put) tako da idealno rešenje nije moguće postići.

Kod malih bitskih brzina, gde su impulsi širi koriste se fiksni ekvilajzeri i obično se oni lociraju na strani predajnika.

Kod prenosa većih bitskih brzina koriste se adaptivni ekvilajzeri. Ovaj tip ekvilajzera prilagođava svoje karakteristike u toku rada kako bi učinio optimalnim prijem signala. Obično se adaptivan ekvilajzer realizuje kao mikroprocesorsko kontrolisani digitalni filter.

8.18. C.8 V standardi

Najveći broj tekućih standarda za komuniciranje računarskim podacima razvijen je od strane organizacija kakve su ITU, koja donosi komunikacione preporuke važeće za ceo svet. Njene specifikacije se karakterišu slovom "V" za PSTN (*Public Switched Telephone Network*) i "X" za PDN (*Public Data Network*). Slovo "V" se odnosi na standard, a broj koji sledi nakon slova ukazuje na pojedini standard. Termin "bis" se koristi da ukaže na drugu verziju pojedinog standarda, dok termin "terbo" na treću verziju. Na slici 1. dat je pregled V standarda koji važe za modeme

<i>V-serija</i>	<i>brzina prenosa bps</i>	<i>Modulacija</i>	<i>ekvilizacija</i>
V.21	300	FSK (2 frekvencije)	--
V.23	1200/600	FSK (2 frekvencije)	--
V.26 bis	2400/1200	4-faze DPSK	sadrži fiksnu
V.27	4800	8-faze DPSK	ručno podešavanje
V.27 ter	488/2400	8-faze DPSK	automatsko adaptivna
V.29	do 9600	16 tačaka <i>constelation</i> QAM	automatsko adaptivna
V.32	do 9600	16 tačaka QAM ili 32 tačke (redundantni) TCM	automatsko adaptivna
V.32 bis	do 14400	do 128 tačaka TCM	adaptivna sa poništavanjem ehoa
V.34	n*2400 do 28800	prekodiranje sa <i>multipoint</i> 4-dimenzionim TCM-om	uvodi se adaptivna ekvilizacija u propusnom opsegu

Slika 1 V-standardi kod modema