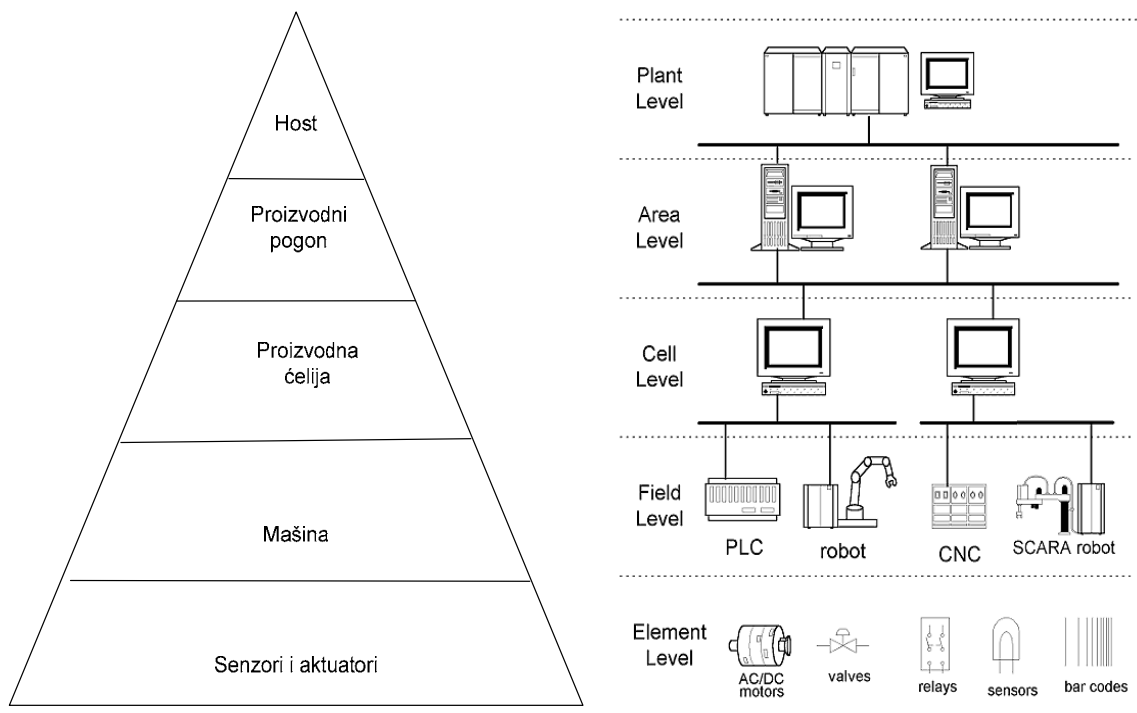


# 1 Komunikacija u industrijskim sistemima

## 1.1 Hijerarhijska organizacija industrijskog sistema

Programabilni kontroleri doveli su do svojevrsne revolucije u načinu proizvodnje. Oni su omogućili fleksibilnu i profitabilnu automatizaciju proizvodnih procesa. Osim direktnog upravljanja proizvodnjom, programabilni kontroleri i računarom upravljani uređaji, generišu i mnoštvo podataka. U mnogim slučajevima, podaci mogu biti vredniji od samog proizvoda. Mnogi proizvodni procesi su izrazito neefikasni. Prikupljanjem podataka o procesu i njihovom analizom može se učiniti da proces postane efikasniji, da kvalitet proizvoda bude viši, da se skрати vreme zastoja. Prvi korak u ovom pravcu jeste prikupljanje i prenos podataka. Za prenos podataka koriste se komunikacione mreže. Postoji velik broj tipova komunikacionih mreža prilagođenih specifičnim zahtevima industrijskih sistema.

Industrijski sistemi mogu biti veoma složeni i obično su organizovani na hijerarhijski način. Svakom nivou hijerarhije odgovara nivo komunikacije koji postavlja neke specifične zahteve u pogledu komunikacione mreže. Na Sl. 1 prikazan je primer hijerarhije kod industrijskih sistema. Najniži nivo je vezan za sam proizvodni proces i obuhvata senzore, aktuatorne i mašine. Kako se krećemo naviše, uz piramidu, primarni zadatak postaje menadžment i upravljanje proizvodnjom na visokom nivou.



Sl. 1 Pet nivoa industrijskog sistema.

### 1.1.1 Nivo senzora i aktuatora

Nivo senzora i aktuatora predstavlja najniži nivo hijerarhije upravljanja i komunikacije u jednom industrijskom preduzeću. Na ovom nivou nalaze se uređaji kao što su ventili, senzori, starteri motora, regulatori temperature i drugi, najrazličitijih U/I uređaji. Velika većina senzora i aktuatora su jednostavni uređaji, koji osim mogućnosti razmene on/off signala sa PLC kontrolerom ili računarom, ne poseduju druge načine za komunikaciju.

Međutim, u novije vreme, čak i u senzore i aktuatore počinju da se ugađuju specijalizovani komunikacioni interfejsi koji omogućavaju serijsku ili paralelnu komunikaciju. Serijski komunikacioni standardi RS232, RS422 i RS455, zajedno sa paralelnim komunikacionim standardom IEEE488 predstavljaju najčešće korišćene komunikacione protokole na ovom nivou. Vremenom, ovi komunikacioni standardi, koji su svi tipa od-tačke-do-tačke, evoluirali su u tzv. industrijske magistrale. Industrijske magistrale omogućavaju povezivanje većeg broja uređaja na zajednički komunikacioni medijum. Kod ovakog pristupa, cena povezivanja je niža. Umesto da se svaki uređaj direktno, posebnim provodnicima povezuje sa PLC kontrolerom ili računarom, za spregu svih uređaja koristi se zajednička komunikaciona linija (magistrala). Jedna od glavnih karakteristika komunikacije na nivou senzora i aktuatora jeste zahtev da vremena prenosa podataka bude predvidljivo. Da bi se u što većoj meri skratilo vreme prenosa podataka, reprezentacija podataka mora biti kompaktna. Komunikacione mreže na ovom nivou su tipa *senzorskih magistrala*. To su obično mreže manjeg obima, niske cene, specijalizovane za brzi prenos diskretnih informacija. Primeri komunikacionih mreža koje se koriste na nivou uređaja su: DeviceNet, Profibus DP, ASI, Interbus/S, Seriplex, SDS, i Compobus/S.

### 1.1.2 Nivo mašina

Pod mašinom se podrazumeva deo opreme za proizvodnju ili rukovanje proizvodima ili procesima. Primeri mašina su roboti, pokretne trake, računarski upravljani alati, tj. CNC mašine, i drugo. Mašine su u direktnom kontaktu sa proizvodom. Neke prenose proizvod, a druge učestvuju u njegovoj izradi. Mašina se sastoji od većeg broja senzora i aktuatora i mehaničkih delova. Mašinom upravlja kontroler mašine, putem priključenih aktuatora i senzora. Kontroler mašine radi po fiksnom programu koji je zapamćen u memoriji kontrolera. U toku rada, svaki od ovih uređaja generiše i podatke koji su od značaja za upravljanje proizvodnjom: brojanje proizvoda, vreme proizvodnje pojedinačnih proizvoda, vreme zastoja i td.

### 1.1.3 Nivo proizvodnih ćelija

Proizvodna ćelija predstavlja grupu uređaja i mašina od kojih svaka ima neku specifičnu ulogu u procesu izrade jednog ili više proizvoda. Ćelije su obično tako koncipirane da se mogu koristiti za izradu različitih tipova proizvoda iz iste familije proizvoda. Svaka mašina, tipično, poseduje programabilnu upravljačku jedinicu, a za komunikaciju sa nadređenim nivoom upravljanja koristi neki specifični komunikacioni protokol. Mašine ne komuniciraju između sebe, već postoji kontroler ćelije koji direktno komunicira sa svakom mašinom. U suštini, kontroler ćelije integriše mašine u kooperativnu proizvodnu ćeliju. Osnovni zadaci kontrolera ćelije su da:

- Puni unapred pripremljene programe u memoriju upravljačkih jedinica mašina i uređaja (*program download*). Izmenom programa rada pojedinačnih mašina, ćelija se može konfigurisati za izradu novog tipa proizvoda.

- Razmenjuje upravljačke i statusne informacije sa mašinama, podešava parametre rada, prikuplja podatke o tekućem stanju proizvodnog procesa i sl.
- Koordinira rad mašina: startuje/zaustavlja mašine, postavlja mašine u određene režime rada.
- Prati performanse mašina.

U jednom proizvodnom pogonu obično postoji više proizvodnih ćelija, od kojih svaka ima svoj kontroler. Osim sa podređenim mašinama, kontroleri proizvodnih ćelija komuniciraju jedni sa drugima i sa nadređenim nivoom upravljanja.

Postoje dva osnovna tipa komunikacije između kontrolera ćelije i mašina: primitivna i serijska.

### **Primitivna komunikacija**

Primitivna komunikacija se koristi u slučajevima kada mašina ne poseduje mogućnost razmene poruka, tj. ne poseduje specijalizovane komunikacione interfejsne, a ni dovoljno veliku moć obrade podataka. Primitivna komunikacija se ostvaruje putem jednostavnih *handshake* procedura koje uključuju dva ili više ulazno/izlazna digitalna signala. Na primer, zamislimo robot koji je tako programiran da čeka da ulaz broj 1 postane aktivan pre nego što pređe na izvršenje programa broj 13. Nakon završetka programa, robot aktivira izlaz broj 2. Pretpostavimo, dalje da su robot i PLC kontroler povezani sa dva digitalna signala: izlaz 2 robota povezan sa ulazom 1 PLC kontrolera, a izlaz 1 robota sa izlazom 2 PLC kontrolera. Na ovaj način oformljena je proizvodna ćelija koja ima jednu mašinu – robot. Aktiviranje svog izlaza 2, PLC izdaje komandu robotu da započne rad; aktiviranjem svog izlaza 2, robot obaveštava PLC da je završio naloženu operaciju.

### **Serijska komunikacija**

Mnogi uređaji i mašine poseduje značajno veće mogućnosti za komunikaciju od proste primitivne komunikacije. Na primer, ako je potrebno napuniti program rada u memoriju upravljačke jedinice mašine, to se ne može postići primitivnom komunikacijom. Za ovu i slične namene, kod mnogih uređaja dostupan je RS-232 komunikacioni port preko koga se ostvaruje asinhrona serijska komunikacija. Komunikacija se ostvaruje uz pomoć posebnog kabla koji povezuje dva uređaja, a koji sadrži veći broj žica gde svaka ima neku specifičnu funkciju. Raspored pinova na priključnom konektoru, namena svakog pina, električne karakteristike interfejsnih kola, talasni oblici signala koji se prenose, i osnovni format poruke su standardizovani, tako da se na ovaj način mogu povezivati uređaji različitih proizvođača.

#### **1.1.3.1 PLC kao kontroler ćelije**

S obzirom da su namenski projektovani za industrijske primene, PLC kontroleri se lako ugrađuju u industrijske sisteme, rukovanje i programiranje PLC kontrolera je jednostavno i ukoliko ne postoje posebni zahtevi, PLC predstavlja prvi izbor za kontroler ćelije. PLC kontroleri su naročito pogodni ako je neophodno obezbediti primitivnu komunikaciju. Takođe, ako u ćeliji postoje i drugi PLC kontroleri istog tipa, za komunikaciju između glavnog i podređenih PLC kontrolera može se koristiti komunikaciona magistrala namenjen toj konkretnoj PLC familiji. Sa druge strane, u odnosu na računar, sa PLC kontrolerom je teže realizovati efikasan interfejs prema operateru. Mada, u novije vreme, sa pojavom namenskih grafičkih terminala i displeja i ova funkcija postaje dostupna i na nivou PLC kontrolera.

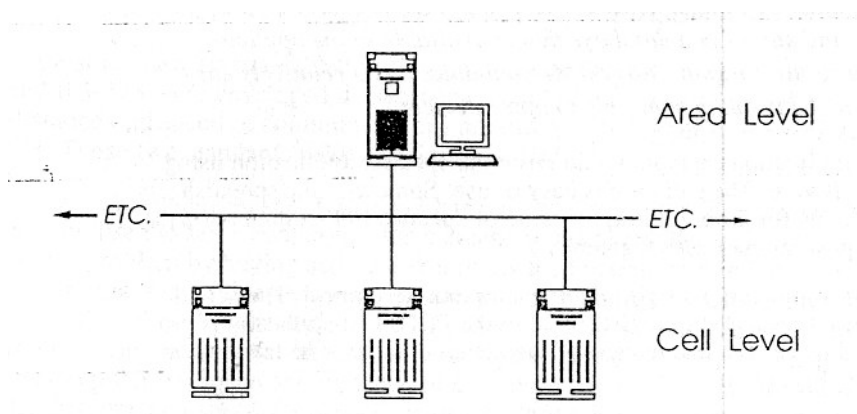
### 1.1.3.2 Računar kao kontroler ćelije

Računari se sve češće koriste kao kontroleri proizvodnih ćelija. U poređenju sa PLC kontrolerima, računari se odlikuju većom fiksibilnošću i većim mogućnostima u pogledu obrade podataka i realizacije složenih algoritama upravljanja. Takođe računari poseduju daleko veće mogućnosti za komunikaciju. Po pravilu, gotovo svi industrijski uređaji i mašine poseduju mogućnost komunikacije sa PC računarom, dok su samo pojedini prilagođeni za komunikaciju sa PLC kontrolerima. Problem je u tome što na nivou PLC kontrolera ne postoji jedinstveni, opšti standard za komunikaciju, već svaki proizvođač PLC kontrolera definiše svoj komunikacioni protokol koji omogućava laku spregu PLC kontrolera i modula tog proizvođača. Sa druge strane svaki PC računar poseduje barem mogućnost RS-232 komunikacije, a ugradnjom specijalizovanih kartica i pratećeg softvera lako se može prilagoditi bilo kom drugom načinu komunikacije.

Za komunikaciju između računara, kao kontrolera ćelije, i mašina tipično se koristi softvera koji se zove SCADA (*supervisory control and data acquisition*). SCADA se izvršava na centralizovanom računaru, kontroleru ćelije, i omogućava komunikaciju sa najrazličitijim tipovima uređaja i mašina. Softver je koncipiran u vidu gradivnih blokova. Programer razvija program upravljanja korišćenjem menija i ikona (vizuelno programiranje), a zatim učitava drajvere za uređaje sa kojima aplikacija treba da komunicira. Drajver je softver, namenski pisan da omogući komunikaciju sa nekim specifičnim tipom uređaja. Komunikacioni drajveri su dostupni za veliki broj uređaja.

### 1.1.4 Nivo proizvodnog pogona

Proizvodni pogon je oblast koja obuhvata jednu ili više proizvodnih ćelija. Kontroler proizvodnog pogona je računar koji prima instrukcije od nivoa hosta i raspoređuje zadatke proizvodnim ćelijama (Sl. 2). Ovaj računar, takođe, komunicira sa kontrolerima drugih proizvodnih pogona u cilju sinhronizacije proizvodnje. Komunikaciona mreža na ovom nivou se karakteriše velikom brzinom prenosa podataka, i determinističkim kašnjenjem. Najčešće korišćene komunikacione mreže na ovom nivou su: ControlNet, Profibus FMS, Genius I/O, ARCNet, ModBus, LON, Sysmac Link, i Controller Link.



Sl. 2 Sprega kontrolera proizvodnog pogona i kontrolera ćelija.

### 1.1.5 Nivo hosta

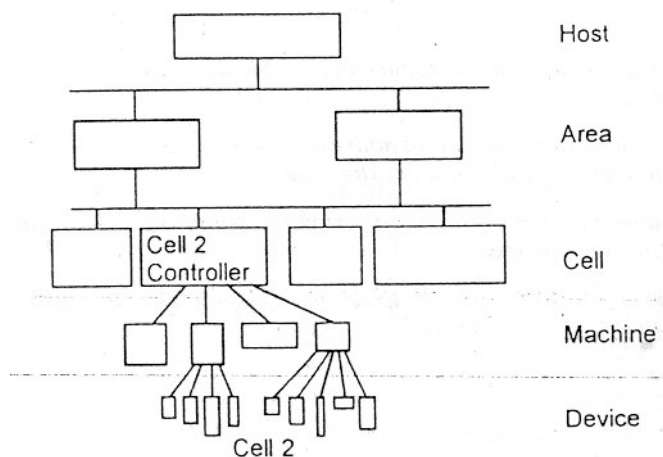
Nivo hosta (ili informacioni nivo) predstavlja najviši nivo u hijerarhiji upravljanja i komunikacije u okviru jednog automatizovanog preduzeća. Ovaj nivo ostvaruje komunikaciju između menadžmenta i proizvodnog pogona, a čini ga jedan ili više računara (*mainframe*-ovi, radne stanice, PC mašine) umreženih u LAN (lokalna računarska mreža). LAN se karakteriše

visokom propusnom moći, složenom strukturom, umerenom brzinom prenosa podataka i nedeterminističkim kašnjenjem. U novije vreme razlike u pogledu performansi između *mainframe*-ova, radnih stanica i PC mašina se sve više smanjuju, a generalni trend je ka distribuiranoj obradi (kooperativnom radu većeg broja računara).

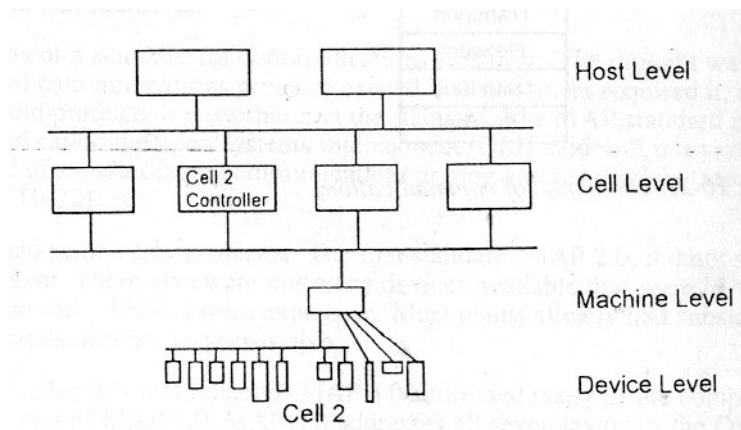
Nivo hosta je odgovaran za poslovni softver, inženjerski softver, softver za poslovnu komunikaciju i slično. Poslovni softver za primenu u industriji se naziva softver za planiranje proizvodnih resursa ili MRP (*Manufacturing Resource Planning*). Ovaj softver se koristi za unos narudžbi, evidenciju sirovina, evidenciju inventara i td. Na osnovu ovih informacija, softver se koristi za generisanje radnih naloga za proizvodnju, naručivanje sirovina i komponenti, organizovanje proizvodnje i sl. Softveri ovog tipa se sve više koriste i za planiranje i predviđanje proizvodnje.

U novije vreme, nivo hosta počinje da se koristi za optimizovanje rada preduzeća. Podaci se automatski prikupljaju u proizvodnim pogonima, direktno od mašina, senzora i operatorskih terminala i prenose do nivoa hosta, gde se obrađuju, analiziraju i koriste kao bi se pronašli načini za povećanje produktivnosti na nivou celog preduzeća. Ključni problem se odnosi na prikupljanje podataka. Kvalitetnije poslovne odluke zahtevaju ažurne i aktuelne podatke o proizvodnji koji su dostupni u obliku razumljivom ljudima.

Na Sl. 3 prikazani su nivoi komunikacije u preduzeću. Organizaciona šema sa Sl. 3 predstavlja klasičan način organizacije, sa jasno razdvojenim hijerarhijskim nivoima, gde se na svakom nivou, kao što je prethodno opisano, koristi određeni tip računarske opreme i neki specifični komunikacioni mehanizmi i protokoli. Međutim, nagli napredak u oblastima računarstva, računarskih mreža i mrežne opreme uslovljava promene u komunikacionoj strukturi. Cena umrežavanja postaje sve niža što čini ekonomičnim da i jednostavni kontroleri mašina, pa čak i senzori i aktuatori budu direktno i ravnopravno povezani na LAN zajedno sa PLC kontrolerima i računarima (Sl. 4). Na taj način, pojednostavljuje se komunikacija na nivou proizvodnog pogona i olakšava funkcija prikupljanja podataka o toku proizvodnje. U budućnosti, očekuje se da će većina senzora i aktuatora biti povezana na LAN. Takođe, u organizacionoj strukturi sa Sl. 4 nivo proizvodnog pogona ne postoji, odnosno pridružen je nivou hosta. To je posledica činjenica da moć obrade miniračunara (PC mašina), koji se koriste kao kontroleri pogona, postaje sve veća, uporediva sa računarima na nivou hosta.



Sl. 3 Tipično hijerarhijsko upravljanje.



Sl. 4 Organizacija sa manjim brojem nivoa.

## 1.2 OSI referentni model

U ranim sedamdesetim, međunarodna organizacija za standardizaciju (*International Standard Organization – ISO*) uočila je potrebu za razvojem mrežnog modela koji bi pomogao u realizaciji kompatibilnih mrežnih proizvoda i rešenja. Kao rezultat definisan je *Open System Interconnection (OSI)* model. Iako postoje i drugi mrežni modeli, OSI model se obično koristi kao referenca za klasifikaciju i opis najrazličitijih komunikacionih protokola.

OSI referentni model definiše sedam nivoa ili slojeva komunikacije:

- sloj aplikacije
- sloj prezentacije
- sloj sesije
- transportni sloj
- mrežni sloj
- sloj veze
- fizički sloj

OSI model opisuje način na koji informacija putuje od aplikacije sa jednog računara ili nekog uređaja do aplikacije koja se izvršava na nekom drugom umreženom računaru ili uređaju. (Za računar ili bilo koji drugi uređaj koji je u stanju da razmenjuje informacije sa drugim umreženim računarima ili uređajima koristimo zajednički termin *mrežni čvor* ili samo *čvor*.) Informacija se prenosi od aplikacije u izvorišnom čvoru do aplikacije u odredišnom čvoru. Na svom putu informacija najpre prolazi naniže kroz sve slojeve izvorišnog čvora, zatim se fizički, putem komunikacione linije, prenosi do odredišnog čvora gde prelazi ponovo, ali sada naviše, u obrnutom redosledu kroz sve slojeve da bi konačno stigla do odredišne aplikacije. Svaki sloj obavlja određenu transformaciju podataka ili sprovodi skup aktivnosti koje se odnose na jedan specifični aspekt komunikacije. Tako, fizički sloj je usredsređen na prenos binarnih signala preko fizičkog prenosnog medijuma, sloj veze problem puzdanog prenosa, mrežni sloj reguliše rutiranje poruka u mreži složene topologije i tako dalje, sve do sloja aplikacije koji je u direktnoj interakciji sa krajnjim korisnikom (ili korisničkom aplikacijom). Sloj višeg nivoa koristi usluge sloja nižeg nivoa. Na primer, sloj veze priprema pakete koji će biti poslani na mrežu, ali ne učestvuje u neposrednom prenosu paketa već je to zadatak fizičkog nivoa koji jedini zna kako treba pobuditi prenosnu liniju da bi se na njoj postavila logička 0 ili 1.

### 1.2.1 Fizički sloj

Fizički sloj je odgovoran za prenos bitova preko deljivog fizičkog medijuma mreže, odnosno mrežnog kabla ili bežične veze. Fizički sloj definiše električne karakteristike signala koji egzistiraju u prenosu podataka, kao što su naponski nivoi logičke 0 i logičke 1 i bitska brzina prenosa. Ovaj sloj takođe definiše način sprege čvora na prenosni medijum sve do nivoa tipova utičnica i rasporeda pinova na priključnim konektorima. Ukoliko se radi o bežičnom prenosu, tada se na fizičkom nivou specificira frekvencija nosećeg signala i tip modulacije.

### 1.2.2 Sloj veze (Data Link Layer)

Zadatak ovog sloja je da kreira komunikacioni kanal za pouzdani prenos podataka između dva čvora povezana fizičkom komunikacionom linijom. Sloj veze može se podeliti na dva podsloja:

- *kontrola logičke veze*: zadužen za pakovanje informacija u okvire i kontrolu ispravnosti prenosa.
- *kontrola pristupa medijumu* (Media Access Control – MAC) reguliše pristup zajedničkom fizičkom medijumu od strane više predajnika.

*Kontrola logičke veze* rešava sledeće probleme:

- (1) *Pakovanje informacije* u pakete ili frejmove (okvire) sa predvidljivim početkom, i krajem. Da bi prenos podataka između predajnika i prijemnika bio sinhronizovan neophodno je da početak i kraj okvira budu jednoznačno određeni. To se obično postiže tako što svaki okvir počinje nekom tačno određenom sekvencom bitova, a završava se nekom drugom. Kod nekih protokola svi okviri moraju imati istu, tačno određenu dužinu. Kod drugih protokola, dužina okvira može biti promenljiva, ali manja od maksimalno dozvoljene dužine.
- (2) *Adresiranje čvorova*. Tipično, svaki čvor u mreži ima jedinstvenu fizičku adresu. Okvir se uvek šalje sa neke *izvorišne* na neku *odredišnu* fizičku adresu, ili skup odredišnih fizičkih adresa. Adresa je deo okvira. Ako se koristi deljivi fizički medijum tada svi čvorovi u mreži primaju okvir, a onda poređenjem odredišne adrese iz okvira i svoje adrese odlučuju da li će prihvatiti okvir ili ne. Ako okvir ima samo jedno odredište, radi se prenosu tipa od-tačke-do-tačke (*unicast*). Ako pošalje okvir treba da prihvate svi čvorovi u mreži, radi se o tipu prenosa emisija svima (*broadcast*). Ako je okvir namenjen jednom određenom podskupu čvorova, prenos je tipa delimična emisija (*multicast*). Protokol sloja veze definiše pravila za dodelu fizičkih adresa čvorova. Treba napomenuti da neki protokoli ne koriste adresiranje čvorova, već se na ovom nivou komunikacije oslanjaju na broadcast (svi primaju sve), dok filtriranje okvira prepuštaju nekom višem mrežnom sloju koji je upućen u značenje podataka koji se prenose.
- (3) *Kontrola ispravnosti prenosa*. Zbog uticaja električnih smetnji i šumova može doći do pojave grešaka u prenosu. Bit koji je poslat kao 1 može biti protumačen od strane prijemnika kao 0 i obrnuto. Kontrola ispravnosti prenosa obezbeđuje mehanizam za detekciju greške u prenosu i mehanizam za korekciju grešaka. Za detekciju grešaka koristi se princip zaštitnog kodiranja. Nad bitovim informacije koja se šalje jednim okvirom primenjuje se određena matematička formula ili algoritam koji kao rezultat daje tzv. *čeksumu*. Čeksuma se pridodaje bitovima informacije i u okviru istog okvira prenosi se do prijemnika. Prijemnik izdvaja iz primljenog okvira bitove informacije i bitove čeksum; nad bitovima informacije primenjuje isto

izračunavanje i dobijeni rezultat poredi sa primljenom čeksumom. Ako ove dve vrednosti nisu jednake, prijemnik zaključuje da je okvir primljen sa greškom i zbog toga ga odbacuje. Način izračunavanja čeksume definisan je konkretnim protokolom. Da bi se prevazišla situacija koja nastaje kada prijemnik dobije pogrešan okvir, različiti protokoli ovog nivoa postupaju na različite načine. Neki protokoli potvrđuju svaki ispravno primljeni okvir, tako što pošiljaocu paketa šalju posebnu poruku potvrde ispravnog prijema, tzv. ACK okvir (odgovara principu pozitivnog potvrđivanja). Neki drugi protokoli, koriste princip negativnog potvrđivanja, tj. obaveštavaju pošiljoca da su primili pogrešan okvir slanjem tzv NACK okvira. Kod prvog pristupa, pošiljalac nakon slanja paketa čeka na potvrdu prijema i ako za neko određeno vreme ne primi potvrdu, ponavlja slanje istog okvira. U drugom slučaju, pošiljalac ponavlja slanje paketa samo ako to zahteva primalac.

- (4) *Eliminacija dupliranih okvira i kontrola toka.* U izvesnim slučajevima može doći do dupliranja okvira (npr., ako se izgubi ACK okvir). Takođe, može se desiti da prijemnik nije u stanju da prihvati sve okvire koje mu predajnik šalje, zato što je opterećen nekim drugim zadacima. Da do toga ne bi došlo, protokl sloja podataka obezbeđuje mehanizme za kontrolu toka podataka.

*Kontrola pristupa deljivom medijumu* treba da obezbedi koordinisano korišćenje zajedničkog, deljivog fizičkog prenosnog medijuma od strane više primo/predajnika. Naime većina fizičkih komunikacionih linija su tipa magistrale. U jednom vremenu preko magistrale se može prenositi samo jedna poruka. Problem nastaje kada više od jednog predajnika želi da šalje podatke. Postoji više različitih načina kako se ovom problemu pristupa:

- (1) *Multiplexiranje na vremenskoj osnovi (Time Division Multiplex).* Svakom čvoru se dodeljuje ograničeni vremenski interval u toku koga može koristiti deljivu komunikacionu liniju za slanje svojih podatak. Ovi vremenski intervali mogu biti dodeljivani čvorovima kružno, a čvor može, ali i ne mora da iskoristi dodeljeno vreme. Kod rešenja ovog tipa, tipično se koristi princip tokena, koji se proleđuje od čvora do čvora, a samo onaj čvor koji poseduje token ima pravo da šalje podatke.
- (2) *Centralizovana arbitraža:* Svaki čvor koji želi da šalje podatke mora najpre da dobije dozvolu od arbitra. Čvorovi mogu imati različite prioritete, a dozvolu dobija čvor najvišeg prioriteta od svih čvorova koji su uputili zahtev.
- (3) *Detekcija kolizija.* Kod ovog pristupa, bilo koji čvor može da započne slanje svojih podataka ako je linija slobodna. Međutim, ako dva ili više čvora počnu sa prenosom u isto vreme dolazi do *kolizije* (ili sudara na magistrali). Svaki čvor koji vrši predaju u obavezi je da nadgleda magistralu i kada ustanovi da je nastupila kolizija obustavlja predaju, čeka neko slučajno vreme i ponavlja slanje istog okvira.

Osim navedenih postoje i druge tehnike za kontrolu pristupa deljivom prenosnom medijumu.

Protokol sloja veze stvara iluziju idealnog komunikacionog kanala. Drugim rečima, protokoli višeg nivoa nisu opterećeni greškama u prenosu podatak, i drugim neregularnim situacijam koje se mogu javiti u toku prenosa podataka preko fizičke komunikacione linije.

### 1.2.3 Mrežni sloj

Osnovni zadatak mrežnog sloja je da obezbedi prenos podataka između udaljenih čvorova, koji mogu, ali ne moraju da dele zajednički fizički prenosni medijum. Savremene komunikacione mreže, kao što je npr. Internet, sastoje se iz mnoštva izolovanih podmreža,



tzv. lokalnih mreža. Lokalne mreže se sprežu u globalnu mrežu pomoću specijalizovanih mrežnih uređaja koji se zovu *ruteri*. Svaki čvor globalne mreže ima jedinstvenu logičku adresu, a prenos podataka između dva ne-lokalna čvora ostvaruje se preko *putanje* koju čine dva ili više rutera. Zadatak protokola mrežnog nivoa je da na osnovu logičke adrese izvorišnog i logičke adrese odredišnog čvora odredi putanju. Ovaj zadatak se zove *rutiranje*. Na primer, za Internet je karakteristično *decentralizovano rutiranje*. To znači da putanju ne određuje pošiljalac, niti se putanja određuje na nekom centralizovanom mestu, već svaki ruter, lokalno, samo na osnovu odredišne logičke adrese određuju da li će i kom susednom ruteru prosediti primljenu poruku ili će poruku uputiti nekom lokalnom čvoru. Protokol mrežnog nivoa definiše format i šemu dodele logičkih adresa, kao i pravila rutiranja. Protokol mrežnog nivoa koji se koristi na Internetu zove se IP (*Internet Protokol*).

Drugi tipičan zadatak protokola mrežnog nivoa je *segmentacija podataka*. Na strani pošiljaoca (izvorišta podataka), protokol mrežnog nivoa deli podatke dobijene od protokola višeg nivoa (transportni nivo) na segmente koje, zatim, zajedeno sa logičkim adresama izvorišta i odredišta pakuje u tzv. *datagram*. Veličina datagrama mora biti usklađena sa dužinom okvira koja je propisana protokolom sloja podataka. Datagrami se nezavisno prenose kroz mrežu. Svaki ruter na putanji između izvorišta i odredišta, može dodatno da podeli datagram na dva ili više manjih datagrama, ako veličina datagrama nije usklađena sa dužinom okvira koja je propisana za fizički medijum kojim datagram treba da nastavi dalje. Datagrami stižu do odredišta u proizvoljnom redosledu, a zadatak mrežnog protokla na stani primaoca je da prikupi sve datagrame i rekonstruiše polaznu poruku. Da bi na prijemnoj strani, datagrami mogli da se slože u pravilnom redosledu, svakom datagramu, još na strani izvorišta, dodeljuje se jedinstveni redni broj.

Pri prenosu kroz mrežu, datagram može biti izgubljen. Ruter može poništiti primljeni datagram zbog uočene greške u prenosu ili zato što je propterćen intenzivnim saobraćajem. Mrežni sloj ne garantuje isporuku datagrama, već brigu o tome prepušta transportnom sloju.

Dakle, mrežni sloj formira datagrame i brine o rutiranju datagrama do zadate odredišne logičke adrese. Mrežni sloj nije opterećen načinom prenosa datagrama između dva susedna čvora na putanji između izvorišta i odredišta, već je to zadatak sloja podataka i fizičkog sloja.

## 1.2.4 Transportni sloj

Transportni sloj omogućava komunikaciju između krajnjih računara koji razmenjuju podatke. Glavne funkcije ovog sloja su da prihvati podatke od sloja iznad (sloj sesije), ako je potrebno podeli podatke na manje delove, tzv. *pakete*, prosledi pakete mrežnom sloju i stara se o tome da poslati podaci budu uspešno primljeni na strani odredišta. Na taj način, transportni sloj u potpunosti sakriva logičku i fizičku strukturu mreže od sloja sesije.

Dva osnovna zadatka transportnog sloja su:

- uspostavljanje veze (konekcije)
- kontrola toka podataka

- (1) *Uspostavljanje veze*. Jedan računar može u isto vreme da razmenjuje podatke sa više drugih računara. Na primer, na jednom računaru može se istovremeno izvršavati više aplikacija, od kojih svaka komunicira sa odgovarajućom aplikacijom na nekom drugom računaru. Međutim, računar ima jedinstveni priključak na mrežu, tako da svi podaci bez obzira od kog računara stižu prolaze kroz isti mrežni priključak. Upravo je zadatak transportnog sloja da razgraniči podatke namenjene različitim aplikacijama u okviru istog računara. Ovo se postiže uvođenjem koncepta *konekcije*. Konekcija predstavlja logički komunikacioni kanal između udaljenih aplikacija.

Krajnje tačke konekcije su *portovi*. Svaki port se identifikuje parom: broj porta, logička adresa čvora (tj. računara)). Svaki podatak (bajt) poslat kroz jedan port pojavice se na odgovarajućem udaljenom portu. Dakle, aplikacije razmenjuju podatke tako što ih prosto, redom upisuju/čitaju u/iz portova konekcije koja ih povezuje. Ovu iluziju direktnog komunikacionog kanala između dve aplikacije stvara transportni sloj. Pre nego što je spremna za korišćenje, konekcija mora biti otvorena. To se postiže razmenom posebnih paketa između transportnih slojeva izvorišnog i odredišnog računara.

- (2) *Kontrola toka podataka*. Nakon što je konekcija otvorena, između dve aplikacije podaci mogu da se prenose u oba smera. Međutim, da bi se ostvarila iluzija toka podataka, transportni sloj treba da obavi niz zadataka. Najpre, neprekidni tok podataka, kako ga vide aplikacije, deli se na pakete i svaki paket se nezavisno šalje mrežnom sloju. Na odredišnoj strani, zadatak transportnog sloja je da prihvata dolazeće pakete i rekonstruiše tok podataka. Međutim, paketi mogu biti primljeni izvan redosleda, sa zakašnjenjem, ili može se desiti da pojedini paketi budu izgubljeni, a neki drugi duplirani. To zahteva da svaki paket mora biti numerisan, da postoji mogućnost potvrđivanja uspešno primljenih paketa i mogućnost ponovnog slanja paketa koji nisu stigli do svog odredišta. Takođe, može se desiti da odredište nije u stanju da prihvata podatke onom brzinom kojom se oni šalju. Zato treba da postoji mogućnost da odredišni računar obavesti izvorišni kada treba da uspri slanje paketa, a kada može da ubrza.

### 1.2.5 Sloj sesije

Sloj sesije se stara o uspostavljanju, nadgledanju i završetku dijaloga između dve aplikacije koje se izvršavaju na udaljenim računarim. Tipične funkcije ovog sloja su: prijavljivanje (login), autorizacija, sinhronizacija i odjavljivanje (logout).

U mnogim slučajevima, interakcija dve udaljene aplikacije ne uključuje samo prostu razmenu podataka. Obično, jedna stana u komunikaciji ima ulogu *klijenta* (onaj ko traži neku uslugu), a druga *servera* (onaj ko pruža usluge). Da bi bi server opslužio klijenta, klijent najpre mora da se predstavi i na neki način dokaže svoj identitet (npr. putem korisničkog imena i lozinke), kako bi server bio siguran da klijent ima pravo korišćenja tražene usluge ili resursa koji su pod kontrolom servera. Po završetku interakcije, klijent se odjavljuje. Sve ove aktivnosti čine jednu *sesiju*. Takođe, sloj sesije treba da obezbedi bezbedno korišćenje svojih resursa. Na primer, ako je resurs mrežni štampač, jasno je da u jednom vremenenu servis štamanja može da opslužuje samo jednog klijenta. Drugim rečima, uvek može da bude otvorena najviše jedna sesija. Na taj način servis se brine o sinhronizaciji.

Dakle, slično konekciji sa transportnog nivoa, sesija se otvara, traje i zatvara. Međutim, jedna sesija može da uključi veći broj konekcija. Na primer, svaka faza sesija može zahtevati posebnu konekciju. Konekcija može nepredviđeno da se prekine, a zadatak sloja sesije je da konekciju ponovo otvori. Šta više, server ne mora biti jedan računar, već jedan računar može biti zadužen za prijavljivanje i autorizaciju, dok drugi može sadržati bazu podataka. Sloj sesije sve ove detalje sakriva od klijenta, koji ima utisak da komunicira sa jedinstvenim serverom preko jedinstvene sesije.

### 1.2.6 Sloj prezentacije podataka

Sloj prezentacije određuje način formiranja podataka pri njihovoj razmeni između računara na mreži. Podaci dobijeni od sloja aplikacije se prevode u određeni standardni format. Sloj prezentacije je odgovoran za konverziju sintakse između dva računara. Na

primer, ako jedan računar koristi ASCII standard za predstavljanje tekstualnih podataka, a drugi EBCDIC.

Drugi tipični zadaci sloja prezentacije podataka su komprimovanje/dekomprimovanje i kriptovanje/dekriptovanje podataka. Između računara često se prenose obimne datoteke koje sadrže tekst, slike ili neki druge tipove podatka. Veličina ovih datoteka može biti više desetina ili čak stotina MB. Prenos tako velikih datoteka traje dugo i zauzima značajan deo kapaciteta mreže. Da bi se skratilo vreme prenosa, na predajnoj strani se obavlja komprimovanje datoteke, čime se njena veličina smanjuje. Komprimovana datoteka se prenosi preko mreže do odredišnog računara, gde se obavlja dekomprimovanje i datoteka, u originalnom obliku, isporučuje odredišnoj aplikaciji.

Kriptovanje (ili šifritanje) se odnosi na zaštitu podataka od novlašćenog korišćenja. Mnoge računarske mreže (kao što je Internet) su javne mreže, što znači da su dostupne svim zainteresovanim korisnicima. To takođe znači da komunikaciona infrastruktura koja povezuje dva udaljena korisnika nije pod kontrolom ni jednog od njih, pa informacije koje se razmenjuju mogu biti dostupne i nekoj trećoj strani i eventualno zloupotrebene. Da bi se to sprečilo, poverljive informacije koje se šalju, na predajnoj strani se kriptuju i prenose u kriptovanom obliku. Na prijemnoj strani, obavlja se dekriptovanje i informacija u originalnom obliku isporučuje odredišnom korisniku. Za bilo koje treće lice koje dođe u posed kriptovane informacije, ona nema značaj jer niko osim pošiljaoca i primaoca nije u stanju da protumači njeno značenje.

Dakle, u sloju prezentacije obavljaju se transformacije podataka, koje su neophodne kako bi se uskladili formati podataka, omogućilo racionalno korišćenje komunikacionog kapaciteta mreže i obezbedila sigurnost podataka.

### **1.2.7 Sloj aplikacije**

Sloj aplikacije je vršni sloj OSI modela koji je u direktnoj interakciji sa krajnjim korisnikom. U okviru ovog sloja su programi (ili aplikacije) koji međusobno razmenjuju podatke, kao što je na primer program za slanje/primanje elektronske pošte ili program za prenos datoteka između udaljenih računara. Ovi programi sakrivaju od korisnika sve aktivnosti koje se dešavaju na nižim slojevima, tako da je za krajnjeg korisnika mreža transparentna.

Da bi dva programa mogla da komuniciraju neophodno je da postoje pravila koja definišu skup dozvoljenih poruka i aktivnosti koje program preduzima po prijemu poruke. Na primer, program za slanje elektronske pošte omogućava korisniku da putom jednostavnog grafičkog interfejsa napiše e-mail, navede odredišnu e-mail adresu i prostim klikom na dugme pošalje e-mail. Zadatak programa je da sadržaj pisma, adresu pošiljoca, adresu primaoca zajedno drugim pratećim informacijama upakuje u poruku koja će biti razumljiva za program koga koristi primalac pisama, a da zatim uspostavi vezu sa Mail serverom i isporuči mu poruku. Dakle, ono što za krajnjeg korisnika predstavlja jednostavnu aktivnost, program razlaže na čitav niz akcija koje uključuju interakciju i dijalog sa nekim udaljenim programom. Da bi dva programa mogla da se razumeju neophodno je da oba poštuju neka zajednička standardizovana pravila. Upravo ova pravila interakcije između udaljenih aplikacija predstavljaju protokole koji spadaju u sloj aplikacije.

Napomenimo da korisnik ne mora biti čovek. Korisnik može biti neka druga aplikacija koja se izvršava na istom računaru. U tom slučaju, interfejs programa prema korisniku nije tastatura miš i ekran već skup funkcija koje su na raspolaganju korisničkom programu.

Slojevi OSI referentnog modela realizuju različite zadatke koje je neophodno obaviti kao bi dva udaljena sistema bila u stanju da komuniciraju. Treba napomenuti da postoje dva aspekta mrežne interakcije: prenos podataka i koordinacija. Prenos podataka sam za sebe nije dovoljan da bi sistem kao celina mogao da radi na zajedničkom zadatku. Neophodno je da postoje načini za koordinaciju aktivnosti različitih delova distribuiranog sistema. Upravo je kooperacija zadatak tri vršna sloja. Ovi slojevi omogućavaju razmenu informacija tako što uspostavljaju, održavaju i završavaju konekciju, staraju se o usklađenosti kodiranja i formatiranja informacije i propisuju pravila dijaloga između dve udaljene aplikacije. Sa druge strane, četiri niža sloja OSI modela, usredsređena su na prenos podatka. Oni ne ulaze u smisao podataka koji se prenose, već ih tretiraju kao niz bajtova (ili bitova) koje treba pouzdano preneti od izvorišta do odredišta.

Treba napomenuti da ne postoji jedinstveni komunikacioni protokol koji pokriva sve slojeve OSI referentnog modela. Obično, protokol se vezuju za određene slojeve, a kompletna komunikaciona infrastruktura se formira slaganjem protkola u strukturu koja se zove *protokol stek*. Na primer, Internet koristi TCP/IP protokol stek, kod koga su dva glavna protokola TCP (odgovara transportnom sloju) i IP (odgovara mrežnom sloju). Fizički i sloj podataka pokriva Ethernet protokol ili PPP protokol. Iznad TCP protokola, u oblasti tri najviša sloja, postoji čitav niz standardizovanih servisa, kao što su Telnet, FTP, e-mail, HTTP.

Podela na slojeve nije tako striktna. Protokol ne mora biti ograničen samo na jedan sloj, već može da obuhvata i više slojeva. Takođe, da bi komunikacija bila omogućena nije neophodno da postoje protokoli za sve slojeve. Na primer, korisnička aplikacija može da se oslanja direktno na fizički sloj, ali to znači da programer te aplikacije mora da reši sve one probleme koji nastaju u komunikaciji, a koji se inače rešavaju protokolima višeg nivoa. U tom smislu, postojanje i dostupnost komunikacionih protokola i standarda značajno olakšava razvoj distribuiranih aplikacija. Uz to, distribuirani sistemi zasnovani na standardnim komunikacionim protokolima su *otvoreni*, što znači da nisu ograničeni samo na umrežavanje proizvoda jednog proizvođača, već su otvoreni za sve proizvode koji su usklađeni sa konkretnim standardima.

Potpuni OSI referentni model, kako je naveden, karakterističan je za računarske mreže. Kod industrijskih mreža, tipično, ne koriste se svi slojevi. Industrijske mreže su obično realizovane kao lokalne mreže i s toga definišu niže slojeve: fizički i sloj podataka. Mrežni i transportni slojevi su potrebni kog globalnih mreža koje objedinjavaju veći broj lokalnih mreža. Slojevi sesije i prezentacije su značajni kod računarskih mreža i heterogenih distribuiranih sistema kao bi se unificirala predstava podataka i omogućili standardizovani oblici interakcije u sistemu. Kod industrijskih mreža, obično, ne postoji potreba za ovim slojevima jer su njeni čvorovi uređaji male složenosti kao što su PLC kontroleri, senzori i aktuatori.

Međutim, možda suprotno onome što bi se moglo očekivati, sloj aplikacije je zastupljen kod savremenih industrijskih mreža. Ovi standardi ne definišu samo fizičke karakteristike prenosnog medijuma i mehanizme za prenos poruka, već idu i korak dalje propisujući funkcionalne modele pojedinih uređaja koji se tipično koriste u industrijskim sistemima. Drugim rečima, ovi standardi teže unifikaciji funkcionalnosti uređaja srodnog tipa. Uređaj koji je povezan u mrežu, dostupan je za komunikaciju sa drugim uređajima. Poruke se razmenjuju sa ciljem da se prikupe podaci od uređaja kao što su senzori ili postave izlazi kod uređaja kao što su aktuatori. Da bi se očitala vrednost senzora, senzoru je potrebno poslati tačno određenu poruku, koju će on prepoznati i na koju će odgovoriti slanjem poruke koja sadrži tražene informacije. Pojednostavljeno rečeno, postojanje aplikacionog sloja u okviru

standarda industrijske mreže znači da je poruka za očitavanje vrednosti identična za sve senzore koji su deklarirani da podržavaju taj konkretni standard. Na ovaj način, jedan senzor se može zameniti senzorom istog tipa, ali nekog drugog proizvođača, a da to ne zahteva bilo kakve izmene u softveru - način na koji se senzori vide preko mrežu je uvek isti bez obzira na način kako su realizovane njegove funkcije.

### 1.3 Lokalne računarske mreže

Lokalna računarska mreža (*LAN - Local Area Network*) funkcioniše u okviru jedne lokacije (na jednom spratu zgrade ili u okviru jedne zgrade). Postoji više tipova LAN mreža koje se razlikuju po topologiji mreže, tipu kabla za povezivanje i metodu pristupa.

Topologija mreže se odnosi na fizički raspored računara i način kablovske instalacije mreže. Danas su upotrebi sledeće tri mrežne topologije:

- Magistrala (bus)
- Star (zvezda)
- Ring (prsten)

#### 1.3.1 Bus mreže

*Bus (magistrala)* predstavlja najjednostavniji metod za umrežavanje. Bus mreža je povezana jednim kablom koji povezuje sve računare, servere i ostale periferijske uređaje (Sl. 5). Računari se priključuju na mrežu posredstvom *mrežnog adaptera* koji je u vidu kartice ugrađen u kućište računara. Svaki mrežni adapter poseduje jedinstvenu *fizičku adresu*. Fizička adresa se naziva *MAC - Media Access Control*. Računari na mreži komuniciraju jedan sa drugim tako što šalju podatke preko kabla, direktno na fizičku adresu odredišnog računara.

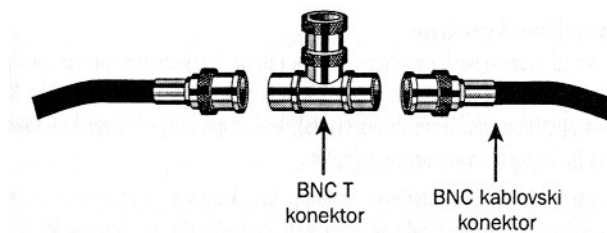


Sl. 5 Bus mreža.

Kada se podaci nađu na mreži oni se zapravo šalju svim računarima u mreži. Mrežni adapteri svih računara analiziraju te podatke u cilju provere da li se odredišna adresa podataka poklapa sa njegovom MAC adresom. Ako se adrese poklapaju, mrežni adapter prosleđuje podatke računaru, u suprotnom ih odbacuje.

#### *Ethernet*

Ethernet (*ethernet*) je najzastupljeniji komunikacioni standard za realizaciju magistralnih računarskih mreža. Kod ovog tipa prenosa podataka, koristi se metod pristupa CSMA/CD - *Carrier Sense Multiple Access with Collision Detection*. Ovaj metod obezbeđuje da samo jedan računar u datom momentu može da šalje podatke na mrežu. Ukoliko u istom momentu neki drugi računar pokuša da preda podatke i otkrije da na mreži postoji saobraćaj, on će sačekati da se mreža oslobodi pre nego što ponovo pokuša da preda podatke. Ukoliko dva računara istovremeno pokušaju da pristupe mreži nastaje kolizija (*collision*). Računari dobijaju informaciju o koliziji i u tom slučaju odlažu predaju podataka za neko vreme, određeno na slučajnoj bazi.



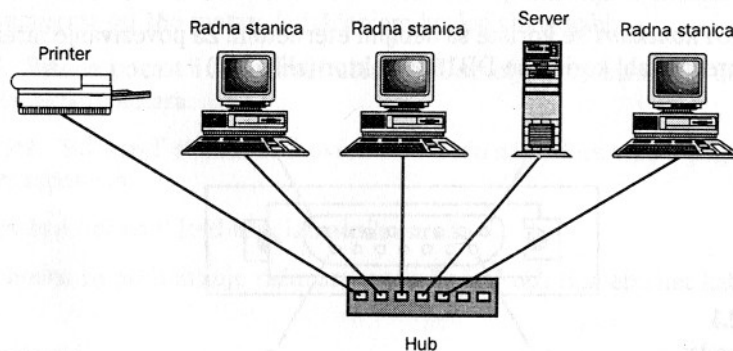
Sl. 6 T konektor.

Računari se povezuju korišćenjem mrežnih kablova. Za kabliranje magistralnih mreža koriste se koaksijalni kablovi. Krajevi koaksijalnog kabla su završeni tzv. terminatorom (obezbeđuje električno prilagođenje), dok se računari priključuju na kabl posredstvom T-konektora (Sl. 6).

Bus mreže su jednostavne za instalaciju, ali poseduju izvesne nedostatke. Otkaz mrežnog kabla (prekid veze između bilo koja dva računara) znači otkaz celokupne mreže. Takođe, zbog mogućnosti pojave kolizija, vreme potrebno za prenos poruke se ne može prethodno odrediti - *nedeterministički prenos*. Naime, zbog kolizija poruka može da zakasni na određeno vreme što u nekim slučajevima može da poremeti rad sistema.

### 1.3.2 Star mreže

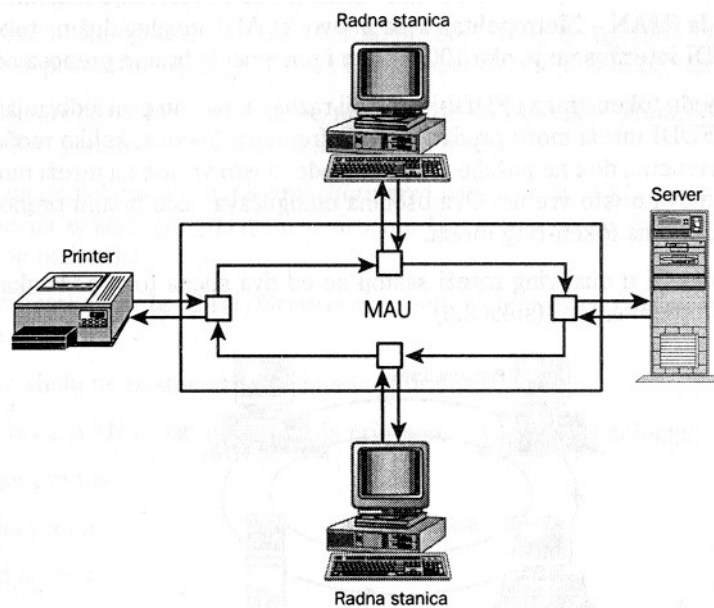
Kod LAN mreža star topologije, svi umreženi računari se povezuju na centralnu tačku mreže, tzv. *hub (hub)*.



Sl. 7 Star mreža.

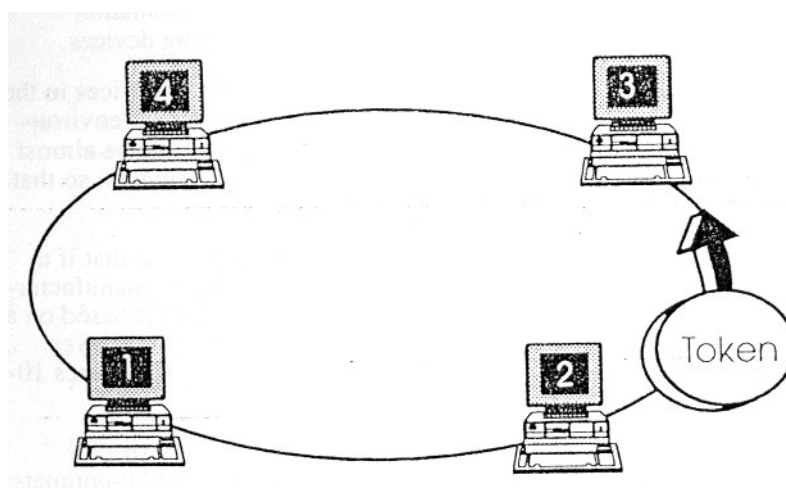
### 1.3.3 Ring mreže

Ring mreža povezuje računare u jedan logički krug. Izlazna (predajna) linija jednog računara se povezuje kao ulazna (prijemna) linija sledećeg računara, tako da podaci putuju u krug i prolaze kroz svaki računar. Kada računar želi da preda poruku on je šalje na predajnu liniju preko koje se poruka prenosi do prvog sledećeg računara. Računar koji je primio poruku ispituje da li je poruka upućena njemu. Ako jeste, računar preuzima poruku; ako nije, računar prosleđuje poruku sledećem računaru.



Sl. 8 Ring mreža.

Fizički, ring mreža ima izgled star mreže. Ključna razlika je u mestu konekcije, koja se ovde naziva *Multi-Station Access Unit* (MAU). U okviru MAU jedinice, podaci se prosleđuju između računara u mreži (Sl. 8).



Sl. 9 Token passing.

Za prenos podataka kroz ring mrežu koristi se metod pristupa *token passing*. Token je oblik kontrolne poruke koja se neprekidno, sukcesivno prenosi od jednog do drugog računara sve dok ne stigne do računara koji želi da koristi mrežu (Sl. 9). Na mreži uvek postoji samo jedan token. Ukoliko neki računar želi da preda poruku, a token je već u upotrebi, on mora sačekati da dobije token. Samo računar koji je primio token može da šalje podatke na mrežu. Konkretnije, algoritam rada ring mreže je sledećeg oblika:

- Računar koji želi da koristi mrežu čeka na slobodan token.
- Računar koji je primio slobodan token, a želi da šalje podatke, modifikuje token tako da on za preostale računare postaje zauzet, i pridodaje tokenu podatke koje šalje zajedno sa MAC adresom odredišnog računara.
- Podaci prolaze pored računara u mreži, sve dok ne stignu na odredište.

- Odredišni računar preuzima podatke, a zatim modifikuje token, tako da označi uspešan prijem podataka, i šalje token ponovo na mrežu.
- Token nastavlja put duž ringa sve dok se ne vrati do računara koji je poslao poruku. Po prijemu tokena, a nakon provere uspešnosti prenosa podatka, predajni računar modifikuje token tako da on postaje slobodan i šalje ga na mrežu.

Treba uočiti da je za prenos jedne poruke između bilo koja dva računara neophodno da token načini jedan pun krug krećući se duž ringa. Iako to izgleda neefikasno, treba imati na umu da brzina prenosa tokena može biti veoma velika (npr. u ring mrež obima 400m, token napravi 5000 krugova u sekundi). Takođe, za razliku od bus mreža, vreme prenosa poruke kod ring mreže se može precizno odrediti - *deterministički prenos*, što je od velikog značaja kod industrijskih sistema za rad u realnom-vremenu.

## 1.4 Industrijske komunikacione mreže

Industrijski sistemi postaju sve složeniji, sa sve većim brojem senzora, aktuatora, automatizovanih mašina i uređaja. Svi ovi raznorodni elementi moraju biti objedinjeni u jedinstvenu kooperativnu celinu. Klasična rešenja automatizacije nekog procesa podrazumevala su postojanje jednog elektro-ormara, od koga su ka aktuatorima i senzorima vodili spletovi kablova sa mnoštvom žica. Takvi sistemi su bili komplikovani za projektovanje i montažu, prilično skupi (troškovi ožičavanja i vremena provedenog na montaži), a posebni problemi su nastajali pri detekciji i otklanjanju nekog kvara upravo zbog nepreglednosti takvih sistema.

Težnja da se navedeni problemi prevaziđu uslovlja je razvoj i izgradnju distribuiranih (decentralizovanih) sistema upravljanja. Osnovne postavke ovakvog pristupa su sledeće:

1. Kompletan sistem se “razbija” na module koji su fizički biti bliže procesu, smanjujući na taj način troškove ožičenja, uz povećanje pouzdanosti rada i pojednostavljenje održavanja.
2. Moduli poseduju ugrađenu “inteligenciju” (otuda termin distribuirana inteligencija), date u vidu CPU-a i softvera. Inteligentni moduli preuzimaju na sebe neke od zadataka automatizacije (prikupljanje ulaznih signala, njihova obrada i ažuriranje izlaza)
3. Moduli komuniciraju međusobno i sa “višim nivoom” u hijerarhiji automatizacije procesa, i primaju od njega komandne i upravljačke signale, pomoću jeftinog prenosnog medijuma (tipično, dvožični provodnik).
4. Sistem je otvorene arhitekture, što omogućava jednostavno dodavanje modula i njihovo priključenje na prenosni medijum (tj. magistralu).
5. Izgradnja distribuiranih sistema je zasnovana na standardizovanim rešenjima, čime se obezbeđuje kompatibilnost sa sistemima drugih proizvođača.

Okosnica svakog distribuiranog sistema su komunikacione mreže, koje se još zovu i *industrijske magistrale* ili *field* (čita se *feld*) *magistrale*.

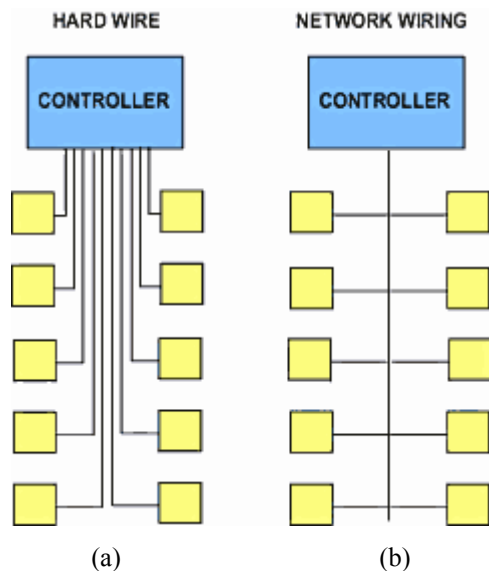
Industrijske mreže imaju mnoge zajedničke osobine sa računarskim mrežama. Računarske mreže omogućavaju komunikaciju između velikog broja računara, a da pri tome računari ne moraju biti direktno, fizički povezani, svaki sa svakim. Svaki računar ima samo jednu vezu sa mrežom. Računarske mreže, takođe, omogućavaju da pojedini uređaji, kao što su štampači, budu dostupni svim računarima na mreži. Slični ciljevi se postavljaju i pred industrijske mreže, s tom razlikom da sada mrežni čvorovi nisu računari već industrijski U/I uređaji i kontroleri.



Glavne prednosti korišćenje industrijskih mreža su sledeće:

#### *Niža cena povezivanja*

Prvi i najočigledniji razlog za izgradnju industrijske mreže je pojednostavljenje povezivanja velikog broja uređaja. Zamislimo jednu automatizovanu, složenu mašinu koja sadrži više stotina U/I elemenata. Ako bi se koristio klasičan pristup i svaki U/I element direktno povezao sa kontrolerom mašine (Sl. 10(a)), troškovi povezivanja svih tih elemenata bili bi ogromni, kao i utrošeno vreme. Takođe, kod ovakvog pristupa, često dolazi do grešaka u povezivanju, a pronalaženje i otklanjanje takvih grešaka je teško i dugotrajno. Sa druge strane, industrijske magistrale omogućavaju da se veliki broj uređaja (čak i do nekoliko stotina) poveže preko jedinstvene komunikacione linije (Sl. 10(b)). Šta više, u nekim slučajevima više U/I uređaja mogu posredstvom namenskih U/I blokova da dele zajednički priključak na mrežu. Treba istaći da je cena U/I uređaja koji poseduju komunikacione interfejse neophodne za umrežavanje viša u odnosu na odgovarajuće U/I uređaje koji nemaju mogućnost komunikacije. Međutim, industrijske mreže donose uštede u ceni instalacije, ceni uloženog rada za povezivanje i kasnije održavanje i otklanjanje kvarova. Na današnjem nivou tehnologije, industrijska mreža postaje isplativo rešenje ako je broj umreženih U/I uređaja veći od 100.



Sl. 10 Dva načina organizacije automatizovanog industrijskog sistema; (a) *Hard wire* - direktna sprega; (b) *Network wiring* - sprega putem industrijske mreže.

#### *Modularnost*

Umreženi industrijski sistemi su modularni. To znači da se novi elementi mogu lako ugraditi u sistem prostim priključenjem na zajedničku magistralu. Celokupan sistem se može rekonfigurisati, a to ne zahteva dodatno ožičavanje. Kontroleri upravljaju U/I uređajima putem poruka, a ne posredstvom U/I pinova. To omogućava da se način na koji je sistem logički povezan reguliše softverom, a ne fizičkim prevezivanjem postojećih veza.

#### *Dijagnosticiranje otkaza*

Umreženi U/I uređaji poseduju veći broj funkcija u odnosu na odgovarajuće klasične U/I uređaje. Zbog potrebe komuniciranja, u takve U/I uređaje ugrađuje se mikroporcesor, koji se osim za komunikaciju može koristiti i za niz drugih naprednih funkcija. Jedna od takvih funkcija je i samo-testiranje. U/I uređaj može, po uključenju napajanja ili po zahtevu nadređenog kontrolera da obavi proveru sopstvene ispravnosti i da onda rezultat testiranja

prosledi nadređenom kontroleru ili računaru. Ovakve informacije su od velikog značaja kada treba locirati kvar u sistemu.

### *Samo-konfiguracija*

Umreženi U/I uređaji rade po programu koji se može daljinski, preko magistrale, napuniti u programsku memoriju uređaja. Izmenom programa, ili nekih fiksnih parametara, uređaj se može konfigurirati za obavljanje nekog specifičnog zadatka. Takođe, umreženi U/I uređaji imaju mogućnost sopstvene identifikacije. Svaki uređaj, u okviru svoje fiksne memorije sadrži sopstveni opis (tip uređaja, spisak podržanih funkcija, i td.). To omogućava da centralizovani kontroler identifikuje sve priključene uređaje, a zatim obavi konfigurisanje celokupnog sistema, upisom odgovarajućih programa i konfiguracionih parametara u svaki uređaj pojedinačno. Na taj način, drastično se skraćuje vreme potrebno za rekonfiguraciju sistema, u slučajevima kada, na primer, sistem treba prilagoditi za proizvodnju nekog novog proizvoda.

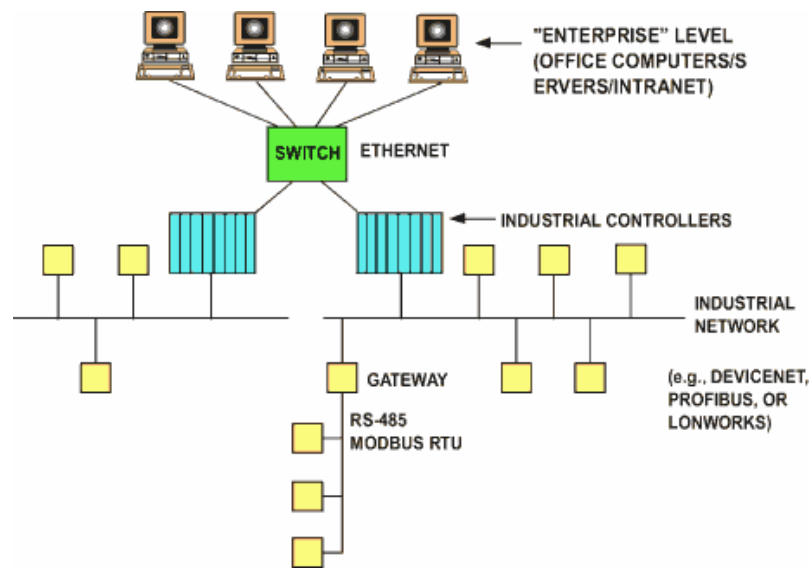
**Primer:** Na slici je prikazana fotografija kontrolera protoka gasa firme Brooks Instrument Division. Ovaj uređaj se isporučuje u dve varijante. Prva (sa oznakom 6950) je klasična, sa ukupno 7 U/I pina putem koji se ostvaruje priključene uređaja na PLC kontroler. Druga varijanta (sa oznakom 6960) ima mogućnost sprege na industrijsku magistralu preko jednog mrežnog konektora. Pored svih funkcija klasične varijante, umrežena varijanta dodatno omogućava pristup više od 100 promenljivih i funkcija svrstanih u 39 kategorija koje pokrivaju sve aspekte regulacije i nadgledanja protoka gasa. Sve te informacije omogućavaju da se u realnom vremenu ostvari precizan uvid u odvijanje procesa i identifikuju izvora eventualnih grešaka.



**Sl. 11 Kontroler protoka gasa 6950/6960 firme Brooks Instrument Division.**

Industrijska preduzeća su složeni informacioni sistemi. Efikasno upravljanje zahteva prikupljanje i obradu velike količine raznorodnih informacija, od finansija i knjigovodstva, evidencije radnog vremena do podataka o ostvarenoj proizvodnji. Izgradnom industrijske mreže, otvara se mogućnost da proizvodni pogon postane deo jednog sveobuhvatnog informacionog sistema preduzeća. Informacije koje potiču čak i od najjenostavnih senzora i aktuatora mogu biti od velike važnosti za praćenje procesa proizvodnje, analizu efikasnosti proizvodnje i kao takve mogu biti osnova za povećanje produktivnosti i profita. Na Sl. 12 je prikazana struktura jednog takvog celovitog industrijskom informacionog sistema. Okosnica ovog sistema je LAN mreža (Ethernet) koja povezuje računare za poslovne primene i servere baze podataka sa industrijskim PC računarima i kontrolerima, a koji se sa druge strane putem

industrijske mreže u vezi sa U/I uređajima, mašinama i automatima. *Gateway* je komunikacioni modul koji povezuje uređaje starije generacije na industrijsku magistralu.



Sl. 12 Struktura mreže industrijskog preduzeća.

#### 1.4.1 Tipovi industrijskih magistrala

Danas se u industriji primenjuje veliki broj standardizovanih komunikacionih mreža, tzv. industrijskih magistrala. Izbor pravog standarda zavisi od zahteva konkretno primene, ali i od faktora kao što su dostupnost uređaja koji podržavaju konkretni standard i njihove cene.

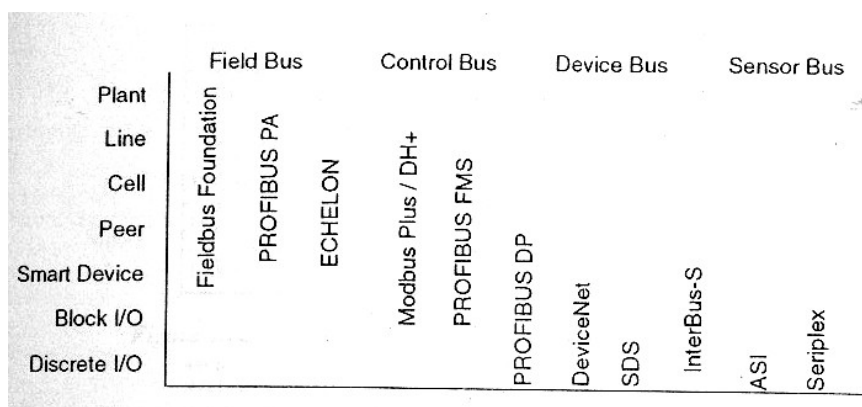
Performanse i pouzdanost automatizovanih industrijskih sistema u velikoj meri su uslovljeni karakteristikama komunikacione mreže. Komunikaciona mreža za industrijske namene mora da garantuje performanse, kao što su iskorišćenost mreže, propusni opseg i maksimalno kašnjenje u prenosu. U mnogim industrijskim postojenjima prisutni su brojni izvori električnih šumova i smentnji (npr. motori, aparati za zavarivanje i sl.). Pod uticajem ovih smetnji, performanse komunikacione mreže mogu biti značajno degradirane (zbog pojave grešaka u prenosu informacija). Može se slobodno reći da je komunikaciona mreža najosetljivija komponenta automatizovanog industrijskog sistema.

U proteklih desetak godina razvijen je i standardizovan veliki broj magistrala i odgovarajućih komunikacionih protokola za primenu u industriji. Ovi standardi su razvijani u različita vremena, od strane različitih proizvođača, za različite namene. Ne postoji standard koji se može smatrati najboljim rešenjem, već svaka od mnoštva industrijskih magistrala, poseduje neke specifične karakteristike koje određuju njenu oblast primene. Izbor magistrale zavisi od zahteva konkretno primene, od kojih su najbitniji:

- maksimalno dozvoljeno vreme odziva
- značaj informacija za bezbedan rad opreme
- količina informacija koju treba prenositi
- maksimalno rastojanje između umreženih uređaja
- namena uređaja koji se povezuju
- mogućnost proširenja
- sigurnost

Dve osnovne kategorije industrijskih komunikacionih mreža su: magistrale uređaja (*device bus*) i magistrale procesa (*process bus*). *Magistrale uređaja* se koriste za prenos kratkih poruka, dužine od nekoliko do nekoliko desetina bajtova. Većina uređaja koji se povezuju ovim tipom magistrale su diskretni uređaji, kao što su diskretni senzori, tasteri i granični prekidači. Na magistrale uređaja mogu se sprežati i analogni uređaji, koji ne zahtevaju obimnu razmenu podataka, kao što su regulatori temperature, neki tipovi drajvera motora, termoparovi i slično. Magistrale uređaja se dalje klasifikuju na bajt- i bit-orijentisane magistrale. Bajt orijentisane magistrale prenose poruke dužine 50 i više bajtova. Bit orijentisane magistrale se koriste za razmenu poruka od 1 do 8 bita sa jednostavnim diskretnim uređajima kao što su senzori i aktuatori. Bit orijentisane magistrale se nazivaju *senzorske magistrale*.

*Magistrale procesa* su nemenjene prenosu dužih poruka (paketa), dužine do nekoliko stotina bajtova. Magistrale procesa su sporije, zato što su paketi duži. Ove magistrale se tipično koriste za prenos parametara rada kontrolerima procesa koji su u većini slučajevima analogni uređaji (mere ili upravljaju analognim veličinama). Od većine analognih uređaja ne zahteva se brzi odziv, zato što se koriste za upravljane procesima koji su po svojoj prirodi spori: protok fluida, koncentracija gasa, temperatura.



Sl. 13 Pregled industrijskih mreža.

Postoji veliki broj tipova industrijskih mreža, od veoma jednostavnih, za spregu sa diskretnim U/I uređajima, do veoma složenih, koje se koriste na nivou celog proizvodnog pogona. Na grafikonu sa Sl. 13 naznačena je oblast primene nekoliko najčešće korišćenih industrijskih mreža. U/I blokovi su komunikacioni moduli koji se mogu koristiti za upravljanje većim brojem U/I uređaja. Sa jedne strane U/I blok je povezan na magistralu, a sa druge direktno upravlja U/I uređajima. Pametni (*smart*) uređaji su oni koji mogu samostalno da obavljaju naprednije funkcije upravljanja. Primer pametnog uređaja je senzorski modul koji ima mogućnost samo-kalibracije ili drajver motora koji može samostalno da realizuje složena kretanja. *Peer* nivo se odnosi na mogućnost komunikacije između kontrolera procesa, kao što su PLC kontroleri. Zatim slede nivoi ćelije, proizvodnog pogona i nivo celokupne fabrike. Kao što se može videti sa slike, ne postoji magistrala koja pokriva sve nivoe. Razlog za to leži u činjenici da različiti nivoi postavljaju različite zahteve. Pri tome treba imati na umu na magistrala nije samo komunikaciona linija, već i skup pravila, tj. protokola, i kojih uređaji spregnuti magistralom moraju da se drže.

### 1.4.2 Standardne magistrale uređaja

Magistrale uređaja, ili fieldbus, su posebna forma lokalne mreže namenjene primenama u oblasti akvizicije podataka i upravljanja sensorima i aktuatorima koji su sastavni deo mašina ili proizvodnih postrojenja. Za razliku od tradicionalnih računarskih mreža, kao što je

Ethernet, gde se performanse mere propusnom moći (količina prenetih podataka u jedinici vremena) pri prenosu velikih blokova podataka, magistrale uređaja su optimizovane za razmenu kratkih informacija o statusu i upravljačkih poruka.

Početni motiv za razvoj magistrala uređaja bila je zamena postojećeg tzv. 4-20mA analognog standarda. Magistrale uređaja su digitalne, bi-direkzione, multidrop, serijske komunikacione mreže koje se koriste za povezivanje izolovanih *field* uređaja, kao što su kontroleri, transduktori, aktuatori i senzori. Svaki *field* uređaj tipično poseduje ugrađen mikrokontroler, što ove uređaje čini *pametnim uređajima*. Pametni uređaji su u mogućnosti da samostalno obavljaju jednostavne funkcije, kao što je dijagnostika, upravljanje i bi-direkciona komunikacija. Takođe, u stanju su da automatski izveštavaju o nastalim kvarovima ili npr. o potrebi kalibracije. Ovi uređaji ne samo da omogućavaju pristup sa daljine, već su često sposobni da komuniciraju jedni sa drugima. U suštini, industrijske magistrale zamenjuju koncept centralizovanog upravljanja, konceptom *distribuiranog upravljanja*. Iz tog razloga, industrijske magistrale su mnogo više od proste zamene 4-20mA analognog standarda. Industrijske magistrala doprinose povećanju kvaliteta, smanjenju troškova i povećanju efikasnosti proizvodnje. Sve ove prednosti, u velikoj meri su posledica činjenice da se prenos informacija obavlja u digitalnom obliku. Digitalni prenos je daleko precizniji od analognog prenosa. Takođe, uređaji sa ugrađenim mikrokontrolerom su fleksibilni.

#### 1.4.2.1 ASI magistrala

ASI (*Actuator Sensor Interface*) magistrala spada u grupu senzorskih magistrala i koristi se na niskim nivoima industrijskih postrojenja. Projektovana je tako da bude laka za instalaciju, održavanje i rekonfiguraciju. Cena instalacije po jednom mrežnom čvoru je veoma niska.

Prenosni medijum je dvo-žični kabl sa upredenim provodnicima. Brzina prenosa podataka iznosi 167Kb/s. Topologija ASI magistrale je oblika stabla. U čvorovima stabla su repiterima koji povezuju magistralne segmente, pri čemu dužina svakog segmenta nesme biti veća od 100m.

ASI je *master-slave* magistrala sa jednim *masterom*. Kod ove magistrale, napajanje i podaci se prenose preko istog kabla, koji može da poveže jedan kontroler (*master*) sa do 31 uređaja (*slave*-ova), pri čemu svaki *slave* može imati do 4 ulaza i 4 izlaza. Svaki *slave* ima jedinstvenu adresu iz opsega 1-31 koja je zapamćena u permanentnoj memoriji *slave*-a.

Poruke koje se prenose ASI magistralom imaju fiksni format. Poruka se sastoji iz ukupno 10 bita: 5 bita za adresu *slave*-a, 4 bita za podatak i 1 bit parnosti, za kontrolu ispravnosti prenosa. Master kružno proziva *slave*-ove. Komunikacija je striktno sinhrona, master ne pravi pauze između dve prozivke, a vremena kada će pojedini *slave*-ovi biti prozvani se mogu unapred predvideti. Vreme jednog punog ciklusa traje 5ms, za slučaj kada je na mreži prisutno maksimalnih 31 *slave* uređaja.

Prvi ciklus prozivke po uključanju mastera predstavlja *start-up* fazu. U okviru prvog ciklusa, master proziva sve dozvoljene adrese (od 1 do 31) i čeka odgovore. Ako se *slave* odazove, master beleži da je na datoj adresi prisutan *slave*. Master u svojoj memoriji poseduje tabelu koja sadrži informacije o konfiguraciji mreže (koje adrese su zauzete, a koje nisu). Poređenjem ovih informacija, sa informacijama prikupljenim u toku *start-up* faze, master može da detektuje greške u mreži, kao što je neispravna adresa ili nepostojeći čvor. U narednim ciklusima, koji odgovaraju radnom režimu mreže, master se obraća samo onim adresama na kojima postoje *slave*-ovi.

Vreme u toku koga se master obraća jednom slave-u naziva se *ćelija*. Postoje tri vrste ćelija: *slave* ćelija, parametarska ćelija i dijagnostička ćelija. U radnom režimu, jedan ciklus se sastoji od: onoliko slave ćelija koliko ima slave-ova na mreži, plus jedna parametarska i jedna dijagnostička ćelija. U okviru slave ćelije, master razmenjuje 4-bitne podatke sa adresiranim slave-om. *Master* proziva redom *slave*-ovi, tako što svakom *slave*-u šalje jedan 4-bitni podatak (obično definiše izlazne signale *slave*-a). Odmah po prijemu poruke, *slave* odgovara, takođe 4-bitnim podatkom (obično se odnosi na stanje ulaznih signala *slave*-a). Ako odziv od *slave*-a izostane, *master* još jednom ponavlja poruku, a ako i pri drugom pokušaju nema odziva, *master* prelazi na prozivku sledećeg *slave*-a. Master će pokušati da dobije odziv od *slave*-a koji se nije javio i u sledećem ciklusu, a ako ni tada ne dobije odgovor, *master* zaključuje da je taj *slave* u kvaru. Ako postoji potreba prenosa podataka dužih od 4 bita (što je karakteristično za analogne veličine), onda se prenos takvih podataka obavlja u više uzastopnih ciklusa.

U okviru parametarske ćelije *master*, ako za tim postoji potreba, šalje specifične konfiguracione parametara izabranom *slave*-u. Na primer, to može biti instrukcija *slave*-u da promeni opseg merenja.

U okviru dijagnostičke ćelije, *master* proziva jednu od adresa na kojoj ne postoji *slave*. Ovo omogućava da *slave*-ovi koji su naknadno priključeni budu brzo prepoznati i uvršteni u listu aktivnih *slave*-ova.

#### 1.4.2.2 CAN magistrala

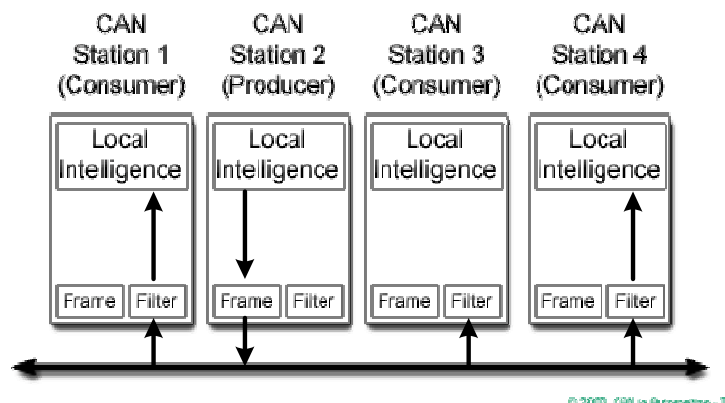
CAN (*Control Arrea Network*) je magistrala namenski razvijena za promenu u motornim vozilima. To je pouzdana magistrala velike brzine prenosa podataka. Zbog svojih dobrih karakteristika i jednostavne ugradnje, CAN magistrala nalazi primenu u mnogim komercijalnim proizvodima, kao i u automatizovanim industrijskim sistemima. U suštini, CAN je serijska komunikaciona magistrala projektovana za povezivanje inteligentnih uređaja i izgradnju inteligentnih sistema i podsistema.

Osnovne karakteristike CAN magistrale su sledeće:

- CAN je multi-master magistrala, što omogućava izgradnju složenih, redundantnih distribuiranih sistema na bazi uređaja različitog nivoa složenosti, a bez potrebe ugradnje centralizovanog, master kontrolera.
- Za distribuciju poruka CAN koristi *broadcast* prenos i pri tome garantuje integritet podataka u celokupnoj mreži. (garantuje da će svi umreženi uređaji primiti poslatu poruku u istom obliku)
- CAN obezbeđuje više efikasnih mehanizama za detekciju grešaka u prenosu podataka, što doprinosi povećanju pouzdanosti celokupnog sistema.
- CAN je prihvaćen kao standard od strane Evropske automobilske industrije.

##### *Mehanizam razmene poruke*

CAN je zasnovan na *broadcast* komunikacionom mehanizmu. Za realizaciju *broadcast* komunikacije koristi se tzv. *porukama-orijentisan transmisioni protokol*. Ovaj protokol ne definiše mrežne čvorove i adrese čvorova, već jedino definiše tipove poruka. Svaka poruka se identifikuje uz pomoć *identifikatora poruke*. Identifikator poruke je jedinstven (ne postoje dve poruke koje imaju isti identifikator) i definiše: (1) sadržaj (tj. značenje) poruke i (2) prioritet poruke. Prioritet poruke se koristi u slučajevima kada u isto vreme više čvorova želi da pošalje poruku. Prednost se daje poruci višeg prioriteta.



Sl. 14 Broadcast poruka u CAN mreži.

Na Sl. 14 ilustrovan je način prenosa poruka preko CAN magistrale. Sistem se sastoji od više mrežnih čvorova (ili, u CAN terminologiji, stanica). Stanica može biti uređaj tipa senzor ili aktuator, kontroler tipa PLC, ili računar. Što se tiče komunikacije, sve stanice su ravnopravne (ne postoje master i slave čvorovi). Svaka stanica poseduje lokalnu inteligenciju (softver koji realizuje konkretne funkcije tog uređaja) i komunikacioni interfejs koji upravlja prenosom poruka. Kada stanica želi da pošalje neke svoje podatke, podaci se najpre pakuju u poruku (*frame* - frejm) kojoj se pridodaje identifikator poruke, koji ukazuje na smisao podataka, a zatim se poruka prosleđuje na magistralu. Stanica ne šalje poruku na neko konkretno određište, već emituje poruku svima (broadcast). Poruku primaju sve stanice, ali odmah nakon prijema poruke, stanice obavljaju filtriranje poruke na bazi identifikatora poruke i ka lokalnoj aplikaciji propuštaju samo one poruke za koje je konkretna stanica zainteresovana, a odbacuju sve ostale.

Na primer, stanica može biti merač nivoa tečnosti u rezervoaru. U regularnim vremenskim intervalima, merač emituje informaciju o nivou tečnosti. Ova informacija se prenosi u obliku poruke sa identifikatorom koji preostalim stanicama ukazuje da ta poruka sadrži podatak koji znači "nivo tečnosti u rezervoaru R1". Stanica merača protoka ne šalje poruku nekoj drugoj konkretnoj stanici, već emituje poruku svim stanicama na mreži i pri tome je ne interesuje ko će i da li će uopšte neko biti zainteresovan za tu informaciju. Neka je druga stanica ventil koji reguliše dotok tečnosti u rezervoar. Ventil je, po prirodi stvari, zainteresovan za nivo tečnosti u rezervoaru; jer ako nivo padne ispod minimalno dozvoljenog, ventil treba da se otvori. Znači, poruka o nivou tečnosti prolazi kroz filter poruka stanice ventila. Za ventil nije bitno ko je poslao poruku, već je bitano značenje poruke. U mreži može postojati i računar koji ne učestvuje direktno u upravljanju, ali vodi arhivu u koju beleži kretanje nivoa tečnosti u rezervoaru. I računari mogu biti stanice u CAN mreži. Poruka o nivou tečnosti proći će i kroz filter poruka računara. Neka je treća stanica na mreži LCD displej koji ima zadatak da prikazuje temperaturu u rezervoaru. LCD displej nije zainteresovan za nivo tečnosti i poruka koju je poslao merča nivoa ne prolazi kroz filter poruka LCD displeja. Displej je zainteresovan za informaciju o temperaturi. Ako ne postoji stanica koja meri temperaturu, na displeju neće biti prikaza. Međutim, onog trenutka kada se na mrežu priključi merač temperature i počene da emituje informaciju o tekućoj temperaturi, na displeju će se pojaviti prikaz.

Koncept identifikatora poruka obezbeđuje veću fleksibilnost prilikom konfigurisanja sistema. Veoma je lako na postojeću mrežu priključiti novu stanicu. Sve dok stanica ima ulogu samo prijemnika, nikakve izmene u hardveru i softveru postojećih stanica nisu potrebne. Na ovaj način omogućeno je modularno projektovanje distribuiranog sistema: sistem se izgrađuje modul po modul. Servisiranje sistema je olakšano: s obzirom da je rad

sistema zasnovan na značenju poruka, a ne na specifičnim tipovima stanica, neispravna stanica može biti zamenjena stanicom iste funkcije nekog drugog proizvođača.

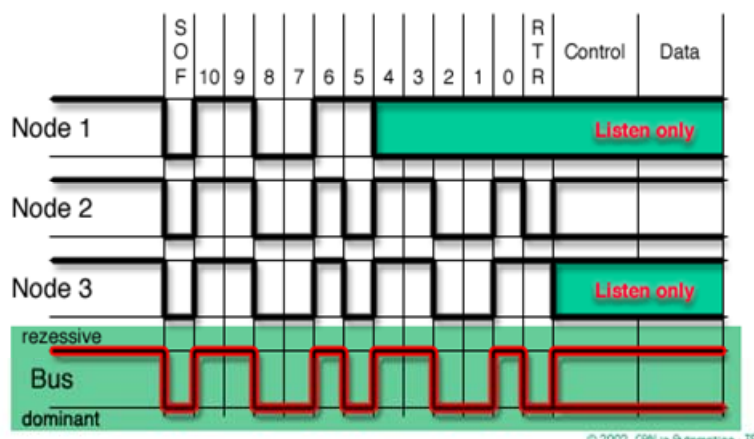
### *Mehanizam pristupa magistrali*

Urgentnosti poruka koje se razmenjuju u distribiranom sistemu za rad u realnom vremenu mogu značajno da se razlikuju. Brzo promenljive veličine (kao što je npr. opterećenje motora) moraju biti češće prenošene i zbog toga kašnjenje u prenosu takvih poruka treba biti kraće u odnosu na neke druge veličine (npr. temperatura). Urgentnost poruka se reguliše prioritetom poruke. Kada dve ili više stanica u isto vreme žele da pošalju poruke, prednost dobija poruka sa najvišim prioritetom. Kod CAN magistrale, prioritet poruke određen je identifikatorom poruke. Prioriteti se dodeljuju porukama u fazi projektovanja sistema i ne mogu se dinamički (u toku rada sistema) menjati. Identifikator sa najmanjom binarnom vrednošću ima najviši prioritet. Treba uočiti da se prioritet vezuje za poruku, a ne za stanicu. Jedna stanica može da šalje više različitih poruka (sa različitim identifikatorima). Bez obzira što potiču iz iste stanice, u kontekstu celokupnog sistema neke poruke imaju niži, a druge viši prioritet.

Konflikti prilikom pristupa magistrali razrešavaju se arbitražom koja uključuje poređenje identifikatora poruka koje različite stanice pokušavaju da pošalju u isto vreme. Razrešavanje konflikta se obavlja na nivou magistrale i interfejsnih kontrolera stanica i ne zahteva postojanje nekog centralizovanog arbitra. Stanica koja ima poruku za slanje najpre čeka da se magistrala oslobodi, a onda na magistralu šalje svoju poruku, bit-po-bit. Poruka počinje identifikatorom poruke, a najpre se šalje bit najveće težine identifikatora. Ako je više od jedne stanice imalo poruku za slanje, onda će nakon oslobađanja magistrale sve one početi sa slanjem svojih poruka, odnosno identifikatora poruka. Magistrali je jedna žica i na njoj može biti prisutan samo jedan bit informacije. U slučajevima kada dve ili više stanica istovremeno šalju svoje poruke, na samoj magistrali obavlja se logička operacija "žičano I" nad bitovima svih poruka koje se šalju. U prvom bitskom intervalu magistrala se pobuđuje bitovima najveće težine identifikatora poruka koje se šalju; u drugom bitskom intervalu sve predajne stanice postavljaju sledeći bit svojih poruka i td. Rezultujuća vrednost na magistrali biće logičko 1 ako su svi postavljeni bitovi 1. Ako je barem jedna stanica na magistralu postavila logičku nulu, rezultujuća vrednost na magistrali biće 0. U kontekstu CAN protokola, logička 0 se zove *dominantni bit*, a logička 1 *recesivni bit*. Kada se na magistrali nađu barem jedan dominantni bit i proizvoljan broj recesivnih bitova, rezultujuće stanje magistrale je dominantno. Samo ako su svi postavljeni bitovi recesivni, stanje magistrale biće recesivno. Nakon postavljanja bita na magistralu, svaka stanica poredi rezultujuće vrednosti na magistrali sa bitom koga je upravo postavila. Ako se ove dve bitske vrednosti razlikuju, stanica se samostalno isključuje, tj. prestaje sa slanjem poruke i prelazi u režim prijema poruke i ne pokušava sa slanjem svoje poruke sve dok magistrala ponovo ne postane slobodna. Kako vreme odmiče, tako se sa magistrale isključuju predajne stanice, sve dok ne preostane samo jedna, čija poruka ima najmanju binarnu vrednost identifikatora (najviši prioritet), koja nesmetano nastavlja dalje sa slanjem svoje poruke.

Opisani mehanizam razrešavanja konflikta obezbeđuje da će poruke biti prenošene u redosledu važnosti poruka za sistem u celini. Prednost ovakvog pristupa naročito dolaze do izražaja u situacijama kada je magistrala preopterećena. Pošto je pristup magistrali regulisan prioritetima, može se garantovati da će urgentne poruke uvek imati malo kašnjenje. Na primer, poruka čiji je identifikator 0, uvek će biti prenesena bez dodatnog kašnjenja, bez obzira na to koliko drugih poruka čeka na prenos. Prenos poruke sa identifikatorom 1 biće odložen samo ako u isto vreme postoji zahtev za prenosom poruke čiji je identifikator 0.





Sl. 15 Arbitraža na CAN magistrali

Mehanizam razrešavanja konflikta kod CAN magistrale ilustriran je na Sl. 15. Tri stanice, Node 1, Node 2 i Node 3 istovremeno otpočinju slanje poruka. Poruke imaju sledeće identifikatore:

Node 1: 110011000000  
 Node 2: 110010110010  
 Node 3: 110010110011

Decimalni ekvivalenti identifikatora su 6300 (Node 1), 3250 (Node 2) i 3251 (Node 3). Stanice sinhronizovano počinju da šalju svoje poruke. Prvih 5 bita svih identifikatora su identični. Za vreme dok se šalju ovih 5 bita, rezultujuće vrednosti na magistrali iste su kao i bitovi koje stanice šalju. Ni jedna stanica ne detektuje konflikt i zato smatra da je sama na mreži. Šesti bit identifikatora poruke koju šalje stanica Node 1 ima vrednost 1, dok su odgovarajući bitovi koje šalju druge dve stanice jednaki 0. Rezultujuća vrednost na magistrali je 0. Node 1 primećuje da vrednost na magistrali ne odgovara onome što je on postavio na magistralu i zbog toga prekida dalje slanje svoje poruke i prelazi u režim slušanja (prijema poruke). Dalje nastavljaju stanice Node 2 i Node 3. Ove dve stanice rade u sinhronizmu sve do poslednjeg bita identifikatora. U tom poslednjem bitskom intervalu konačno pobeđuje Node 2, zato što je bit najmanje težine identifikatora njegove poruke 0, za razliku od odgovarajućeg bita koga šalje stanica Node 3. Zbog takve situacije, stanica Node 3 prekida predaju i prelazi u režim prijema, dok Node 1 nastavlja sa slanjem tela poruke. Ovu poruku primiće sve stanice u mreži, uključujući i stanice Node 1 i Node 3, koje su izgubile od stanice Node 1 u nadmetanju za pristup magistrali.

### Formati poruka

CAN protokol predviđa dva različita formata poruka (ili frejmova – *frames*). Jedina suštinska razlika između dva formata je u dužini identifikatora poruke. Dužina identifikatora kod standardnog frejma iznosi 11 bita (varijanta CAN standarda sa oznakom CAN 2.0A), dok je kod proširenog frejma dužine 29 bita (varijanta CAN standarda sa oznakom CAN 2.0B). Struktura CAN poruke prikazana je na Sl. 16.



Sl. 16 Standardni frejm

Poruka je uokvirena Start bitom (SOF - *Start Of Frame*) i IFS poljem. SOF bit označava početak poruke. SOF je uvek dominantni bit. Vodeća ivica ovog bita se koristi za

sinhronizaciju prijemnih stanica. Sa druge strane poruka se završava poljem IFS (*Intermission Frame Space*). Polje IFS uvek sadrži tri recesivna bita i služi da obezbedi minimalno vremensko rastojanje između dve uzastopne poruke. Vremenski period između dve poruke (vreme nakon završetka polja IFS prethodne do bita SOF naredne prouke) može biti proizvoljan. Kaže se da je u tom vremenu, magistrala pasivna (*idle*). U pasivnom stanju, vrednost na magistrali odgovara recesivnom bitu.

Unutrašnjost poruke se sastoji iz sledećih segmenata:

- **Polje za arbitražu**, koga čine
  - a) Identifikator. Dužina identifikatora je 11 bita, kod frejmova standardnog formata, odnosno 29 bita kod frejmova proširenog formata.
  - b) bit RTR (*Remote Transmission Request*). Vrednost bita RTR ukazuje na tip poruke: frejm podataka (*data frame*) ili frejm zahteva za podacima (*data request frame*). Frejm zahteva za podacima se naziva *udaljeni frejm*. Stanica šalje poruku ovog tipa kada od drugih stanica u mreži traži određene podatke.
- **Kontrolno polje**, koga čine:
  - a) bit IDE (*Identifier extension*) - ukazuje na format frejma: standardni ili prošireni.
  - b) 3-bitno polje DLC (*Data Length Code*) - ukazuje na broj bajtova u polju podataka (Data) koje sledi. Ako je poruka tipa zahtev za podacima, DLC sadrži broj bajtova koji se zahtevaju.
- **Polje podataka** – sadrži podatke koji se prenose i može imati najviše 8 bajtova.
- **Polje CRC** - koristi se za kontrolu ispravnosti prenosa.
- **Polje za potvrđivanje**, koje se sastoji iz:
  - a) bita ACK slot. Predajna stanica na ovu bitsku poziciju recesivni bit, dok prijemne stanice koje su ispravno primile poruku (korektno CRC) postavljaju dominantni bit.
  - b) bita ACK graničnik - uvek sadrži recesivni bit.
- **Polje EOF** (*End of Frame*) – označava kraj poruke.

#### *Tipovi poruka*

Dva osnovna tipa poruka kod CAN protokola su: fremovi podataka i frejmovi zahteva za podacima. Ako je RTR bit dominantan, poruka je frem podataka; u suprotnom, a ako bit RTR recesivan poruka predstavlja zahtev za podacima. Kao što ime kaže, frejm podatka se koristi za prenos podataka (npr. vrednosti veličina koje stanica meri). Stanica šalje frejm podataka inicirana nekim internim događajem ili protokom vremena ili kao odgovor na primljeni zahtev za podacima. Na primer, neka je stanica senzor temperature. Stanica može biti programirana da informaciju o tekućoj temperaturi, u vidu frejma podataka sa odgovarajućim identifikatorom, šalje u regularnim vremenskim intervalima (npr. 5 min.). Alternativno, stanica može biti programirana da poruku o temperaturi šalje samo onda kada je prekoračena maksimalna granična temperatura (npr. 100°C). Takođe, stanica će poslati poruku o temperaturi uvek kada primi poruku tipa zahtev za podacima. Da bi stanica odgovorila na zahtev, poruka zahteva za podacima mora imati isti identifikator kao poruka o temperaturi, koja se traži. Poruke tipa zahtev za podacima omogućavaju realizaciju prozivke.

Uočimo da bit RTR sledi odmah nakon identifikatora i da učestvuje u arbitraži. S obzirom da je kod frejmova podataka ovaj bit dominantan, a kod fremova zahteva za podacima

recesivan, frejmovi podatak imaće prioritet nad odgovarajućim frejmovima zahteva za podacima.

### *Detekcija i signalizacija grešaka*

CAN protokol garantuje da će sve stanice primiti poslatu poruku u istom obliku ili ni jedna stanica neće primiti poruku. U toku prenosa poruke mogu se javiti greške. Zbog postojanja električnih šumova i smetnji, ili neispravnosti instalacije, može se desiti da neka prijemna stanica primi dominantni bit iako je predajna stanica postavila na magistralu recesivni bit, ili obrnuto. U ispitivanju ispravnosti prenosa učestvuju sve stanice u mreži. Međutim, za razliku od drugih, srodnih protokola, CAN protokol ne koristi tehniku potvrđivanja ispravnog prijema poruke, već predviđa da svaka stanica koja uoči grešku u prenosu ima obavezu da signalizira ostalim stanicama da je došlo do greške.

Za detekciju grešaka na nivou prouka predviđena su tri mehanizma:

- **CRC** (*Cyclic Redundancy Check – Ciklično redundantni kod*). Odnosi se na zaštitno kodiranje sadržaja. Na predajnoj strani, na osnovu bitova poruke počev od start bita do kraja polja podataka, primenom šeme kodiranja koja se zove ciklično redundantno kodiranje, izračunava se 16-bitna vrednost i smešta u polje CRC. Na prijemnoj strani, postupak izračunavanja se ponavlja i rezultat poredi sa primljenom CRC vrednošću. Ako dve vrednosti nisu jednake, prijemna stanica zaključuje da je u prenosu poruke došlo do greške.
- **Provera formata poruke**. Ovaj mehanizam verifikuje strukturu frejma tako što ispituje vrednosti fiksnih bitskih pozicija u primljenoj poruci. Ako se primljena poruka ne uklapa u dozvoljeni oblik frejma, detektuje se greška formatiranja.
- **ACK greška**. Kao što je već napomenuto, sve prijemne stanice potvrđuju prijem poruke postavljanjem dominantnog bita na magistralu u vremenu bitskog intervala ACK. Ako u vremenu biskog intervala ACK, na magistrali ostane recesivna vrednost, predajna stanica detektuje ACK grešku.

CAN protokol, takođe, predviđa i dva mehanizma za detekciju grešaka na nivou pojedinačnih bitova:

- **Monitoring**. Svaka stranica koja šalje poruku, u isto vreme nadgleda vrednost na magistrali, tako da je u mogućnosti da utvrdi neslaganje između poslatog i primljenog bita. Ovaj mehanizam omogućava kako detekciju globalnih grešaka (zbog kvara magistrale ili prisustva električnih smetnji), tako detekciju lokalnih grešaka (otkaz interfejsne elektronike stanice).
- **Umetanje bitova**. Ovaj mehanizam testira kodiranje pojedinačnih bitova. Za reprezentaciju bitova na CAN magistrali koristi se kod NRZ (*Non Return to Zero – NZR promena*). Poruka se šalje na magistralu u vidu sekvence bitova, i to bit-po-bit. U skladu sa kodom NRZ, naponski nivo na magistrali odgovara logičkoj vrednosti bita u toku celokupnog bitskog intervala. To znači da u slučaju prenosa više uzastopnih nula ili više uzastopnih jedinica, signal na magistrali ostaje nepromenjen, tj. konstantan, što može da oteža sinhronizaciju predajne i prijemnih stanica. Da bi se eliminisali dugački intervali konstantnog naponskog nivoa na magistrali, neposredno pre slanja na magistralu, poruka se prekodira, tako da se nakon svake pete uzastopne iste bitske vrednosti ubacuje bit komplementarne vrednosti. Na prijemnoj strani, ubačeni bitovi se detektuju i izbacuju. To znači da pojava konstantnog naponskog nivoa na magistrali trajanja dužeg od trajanja 5 bitskih intervala, ukazuje na grešku u prenosu.

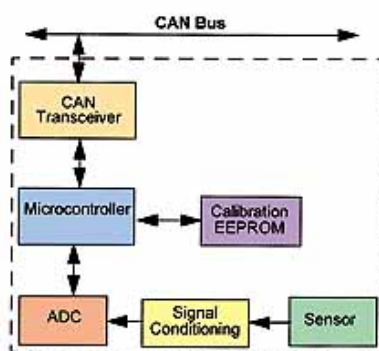
Ako jedna ili više stanica, korišćenjem opisanih mehanizama, detektuju jednu ili više grešaka tekući prenos se obustavlja slanjem tzv. flega greške (*error flag*). Na taj način, preostale stanice, koje nisu uočile grešku, sprečavaju se da prime poruku. Znači, dovoljno je da jedna stanica otkrije neki tip greške u poruci pa da poruka bude odbačena od strane svih stanica. Tako se obezbeđuje konzistentnost podatak u celokupnom sistemu – ili su sve stanice primile istu poruku ili poruku nije primila ni jedna stanica. Nakon što je zbog detektovane greške prenos poruke prekinut, predajna stanica ponavlja slanje iste poruke.

#### *Lokalizacija otkaza*

I pored svih opisanih mehanizama za detekciju grešaka, prenos može biti poremećen ako neka stanica otkáže na način koji uslovljava da neke ili sve poruke koje se prenose preko magistrale budu prekinute. Da se ne bi desilo da otkaz jedne stanice, blokira ceo sistema, CAN protokol predviđa mehanizam za razlikovanje sporadičnih (prolaznih) grešaka i permanentnih (trajnih) grešaka. Ovo se postiže tako što svaka stanica vodi evidenciju o tipovima detektovanih grešaka i učestanosti njihovog pojavljivanja. Na osnovu jedne takve analize, sama stanica može da zaključi da sa njom nešto nije u redu i da se samostalno isključi sa mreže, kako ne bi ometala ostatak sistema.

#### *Struktura CAN stanice*

Na Sl. 17 je prikazana unutrašnja struktura jedne tipične CAN stanice. Stanica predstavlja pametni senzor i sadrži senzorski element, analogne i digitalne komponente. Analogni signal koga generiše senzorski element se najpre uobličava, a zatim digitalizuje pomoću AD konvertora. Dobijene digitalne vrednosti obično zahtevaju dodatnu obradu (linearizacija) koju obavlja mikrokontroler na osnovu kalibracionih podataka smeštenih u postojanoj memoriji tipa EEPROM. Sprega sa CAN magistralom se ostvaruje pomoću CAN primo-predajnika (transivera). CAN primo-predajnik realizuje CAN protokol i zadužen je za formatiranje poruka, kontrolu grešaka, predaju i prijem frejmova, filtriranje poruka. Mikrokontroler realizuje i druge, aplikaciono-specifične funkcije koje se odnose na razmenu prouka sa drugim stanicama u sistemu. Stanica može biti realizovana u obliku jedinstvenog integrisanog kola, ili kao uređaj koji se sastoji od više integrisanih kola. Pojedini mikrokontroleri novije generacije poseduju ugrađen CAN kontroler, i mogu se direktno povezivati na CAN magistralu.



**Sl. 17 CAN stanica.**