

Mikroprocesorska elektronika

Serijska komunikacija

11 SERIJSKA KOMUNIKACIJA

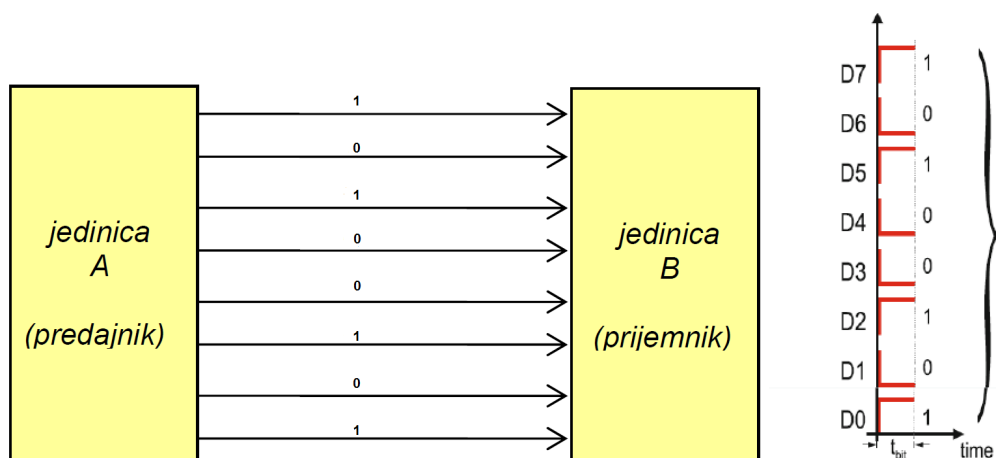
U dosadašnjim razmatranjima upoznali smo se sa paralelnim prenosom podataka kod koga se između dve jedinice istovremeno prenosi dva ili više bita. Međutim, serijski komunikacioni kanali su danas bez sumnje najzastupljeniji vid komunikacije koja se koristi unutar digitalnih sistema. Različiti oblici serijskih komunikacionih formata i protokola koriste se u aplikacijama počevši od kratkih komunikacionih linkova koji povezuju module unutar istog ili odvojenih integrisanih kola, pa sve do komunikacionih linkova koji obezbeđuju vezu sa svemirskim sondama koje putuju na udaljene planete.

Praktično skoro svi oblici komunikacije koji se danas koriste u potrošačkoj elektronici baziraju na serijskoj komunikaciji. Neki od najpoznatijih serijskih komunikacionih protokola korišćenih danas su:

- RS-232, USB (eng. *Universal Serial Bus*), Bluetooth, FireWire
- Ethernet, WiFi
- I2C, SPI, TWI
- PCIe
- SATA
- ...

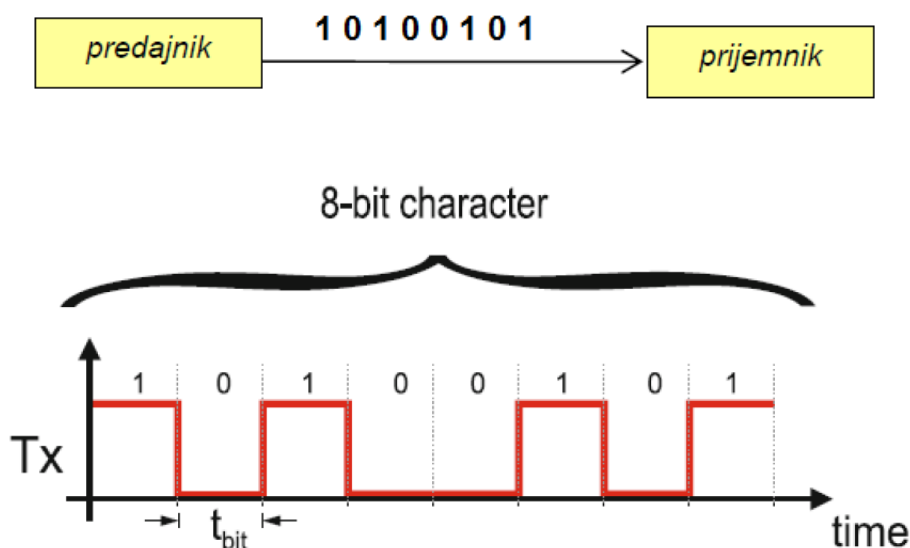
Mikrokontroler ATmega328 poseduje sledeće serijske interfejsse. To su: USART (sinhroni i asinhroni), SPI (eng. *Serial Peripheral Interface*) i TWI (eng. *Two Wire Interface*). U okviru ovog predmeta ćemo detaljno obraditi sinhronu i asinhronu UART komunikaciju, a SPI i TWI se obrađuje u okviru predmeta Računarska elektronika u VIII semestru.

Paralelni komunikacioni kanal, prilikom prenosa n-bitnog podatka, koristi n simultanih signalnih linija, jednu za svaki bit podatka koji se prenosi. Primer za 8-bitnu paralelnu komunikaciju je dat na slici 11.1.



Slika 11.1 8-bitna paralelna komunikacija

Sa druge strane, serijski komunikacioni kanal, prilikom prenosa n -bitnog podatka, koristi samo JEDNU signalnu liniju, pri čemu se n -bita podatka prenosi sekvencijalno, jedan za drugim. Svaki bit koji se prenosi serijski zahteva unapred definisanu količinu vremena, t_{bit} , tako da je za prenos n -bitnog podatka potrebno ukupno $n \times t_{\text{bit}}$ sekundi (slika 11.2).



Slika 11.2 8-bitna serijska komunikacija

Kako je u jednom trenutku moguće preneti samo jedan bit, očigledan nedostatak serijskog prenosa je u tome što je sporiji u odnosu na paralelni prenos. Intuitivno, paralelni prenos sa n provodnika brži je n puta u odnosu na serijski prenos, naravno pod pretpostavkom da su svi ostali parametri prenosa isti. Ovo je uvek tačno, međutim kako rastojanje na kojem se vrši komunikacija i brzina kojom se prenose podaci rastu, talasni efekti na komunikacionim vodovima, preslušavanje i uticaj šuma imaju sve veći negativan uticaj u slučaju paralelnih komunikacionih kanala, ograničavajući njihovu praktičnu primenu. Ovo je jedan od najvažnijih razloga za masovno korišćenje serijskih komunikacionih kanala u savremenim elektronskim uređajima i sistemima.

Ukoliko je ipak potrebna paralelna komunikacija, a da se uticaj šuma i drugih efekata smanji tada se koristi metodologija diferencijalnog povezivanja što je dosta skuplje rešenje i od običnog paralelnog, a pogotovo od serijskog povezivanja.

Brzina serijske komunikacije

Brzina prenosa podataka preko serijskog kanala određena je količinom vremena potrebnog za prenos jednog bita informacije (t_{bit}).

U praksi se koriste dve metrike za iskazivanje brzine prenosa podataka preko serijskih kanala:

- Bitska brzina (eng. *bit rate*)
- Simbolska brzina (eng. *baud rate*)

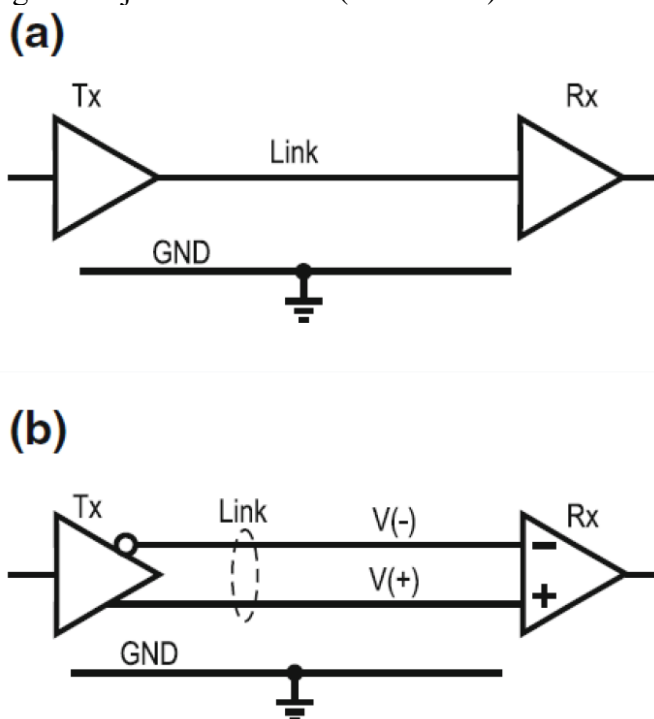
Bitska brzina (eng. *bits per second*, bps) pokazuje koliko je bita moguće preneti tokom jedne sekunde preko serijskog kanala. Ukoliko je poznato bitsko vreme, t_{bit} , bitska brzina računa se pomoću sledećeg izraza. Na primer, serijski kanal sa bitskim vremenom $t_{\text{bit}} = 10 \text{ ns}$ ima bitsku brzinu od 100 Mbit-a u sekundi (100 Mbps, 100 Mbit/s).

Simbolska brzina (eng. *baud rate*), odnosi se na broj karaktera (simbola) koji se mogu preneti preko serijskog kanala tokom jedne sekunde. Ukoliko je svaki simbol predstavljen pomoću jednog bita, simbolska i bitska brzina su jednake.

Ukoliko se koriste složenije modulacione tehnike, moguće je kodirati veći broj bita za svaku promenu signala na serijskom kanalu. Na primer, ukoliko se koristi fazna modulacija (eng. *Phase-Shift Keying*, PSK), moguće je kodirati dva ili više bita u svaku promenu faze signala koji ide preko serijskog komunikacionog linka. U ovom slučaju bitska brzina će biti veća od simbolske brzine.

Komunikacioni kanal

Komunikacioni kanali mogu biti realizovani na razne načine. Ožičeni kanali mogu da koriste *single-ended* ili *diferencijalne veze* (žice) za prenos podataka. U slučaju *single-ended* veze, kanal se sastoji iz jednog fizičkog voda (žice) i referentne linije (masa-ground). U slučaju n-bitnog paralelnog *single-ended* kanala bilo bi potrebno ukupno $n+1$ žica za ostvarivanje veze: po jedna žica za svaki bit plus referentna linija mase. U slučaju n-bitnog serijskog *single-ended* kanala dovoljne bi bile samo dve žice: jedna signalna i jedna referentna (slika 11.3a).



Slika 11.3 *Single-ended* i diferencijalni prenos

U slučaju diferencijalne veze, informacija koja se prenosi kodirana je kao naponska razlika između dve fizičke žice (slika 11.3b). Ova vrsta veze je otpornija na uticaj

smetnji, jer se uticaj aditivnih smetnji međusobno poništava prilikom prenosa podataka preko diferencijalne veze.

Paralelni diferencijalni n-bitni kanal zahtevao bi $2n+1$ žica za ostvarivanje veze: po dve žice za svaki bit, plus jedna žica za prenos referentnog napona (mase). U slučaju serijskog diferencijalnog kanala bile bi potrebne samo 3 žice: dve za prenos informacija i jedna za prenos referentnog napona.

Bežični kanali koriste bežične načine komunikacije za prenos podataka bazirane na različitim fizičkim fenomenima:

- Optičke, u slučaju infracrvenih ili laserskih primopredajnika
- Akustičke, u slučaju podvodnih komunikacija
- Elektromagnetne, u slučaju radio komunikacija kao što su WiFi, Bluetooth

i sl.

U bilo kom od ovih slučajeva nema potrebe za zajedničkom referencom, ali bi u slučaju paralelnog prenosa i dalje bilo potrebno n bežičnih linkova, dok bi u slučaju serijskog prenosa bio dovoljan samo jedan ovakav link.

Na osnovu prethodne analize može se naslutiti glavna prednost serijskih kanala u odnosu na paralelne: **njihova cena**. Mnogo je jeftinije i jednostavnije imati samo jedan serijski link za prenos podataka nego n paralelnih, sinhronizovanih linkova u slučaju paralelnog prenosa.

Svi serijski komunikacioni kanali zahtevaju postojanje stabilnog izvora referentnog sinhronizacionog signala (takta) na osnovu kojega se mogu odrediti brzine predaje i prijema podataka i izvršiti unutrašnja sinhronizacija prijemnika i predajnika. U zavisnosti od načina na koji se obezbeđuje signal takta, kanal može biti:

- Asinhroni
- Sinhroni

Asinhroni kanali koriste nezavisne generatore sinhronizacionog signala na prijemnoj i predajnoj strani. **Sinhroni kanali** distribuiraju jedan, zajednički sinhronizacioni signal, zajedno sa podacima koji se prenose. Ovaj zajednički sinhronizacioni signal generiše se na jednom mestu, najčešće na predajnoj strani.

Komunikacioni paketi

I asinhroni i sinhroni serijski komunikacioni protokoli dele poruke koje prenose preko kanala na osnovne komponente koji se nazivaju **paketi podataka ili datagrami**. Svaki paket sastoji se iz tri dela:

- zaglavlja (eng. *header*)
- tela (eng. *body*)
- repa (eng. *footer*)

Zaglavlje i repa započinju, odnosno završavaju, svaki paket i sadrže sinhronizaciona informaciona polja koja su dodata od strane komunikacionog protokola kako bi se obezbedio pouzdan prenos podataka kroz komunikacioni kanal.

Zaglavlje paketa sadrži:

- Polje koje označava početak paketa
- Opciona polja koja sadrže adresne informacije neophodne u slučaju postojanja većeg broja potencijalnih prijemnika (eng. *multi-point* topologija)
- Opciona polja koja sadrže informaciju o dužini i tipu paketa

Rep paketa sadrži:

- Polje koje označava kraj paketa
- Opciono polje koje sadrži informacije na osnovu kojih se može proveriti ispravnost primljenog paketa (eng. *frame/package check sequence*)

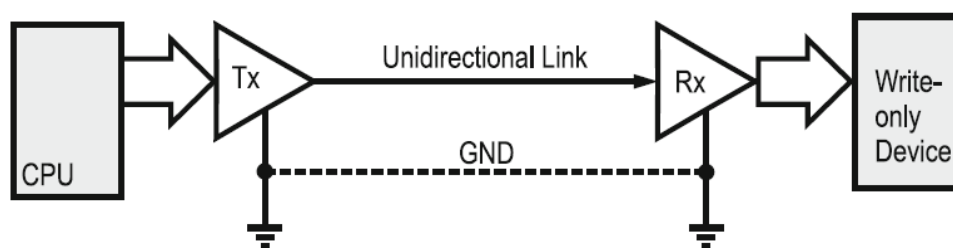
Zaglavlje i rep razlikuju se po dužini i formatu u slučaju asinhronih i sinhronih kanala kao i od komunikacionog protokola koji se koristi.

Vrste serijskih komunikacionih kanala

Na osnovu mogućnosti prenosa podataka između predajnika i prijemnika, serijski kanali dele se u sledeće grupe:

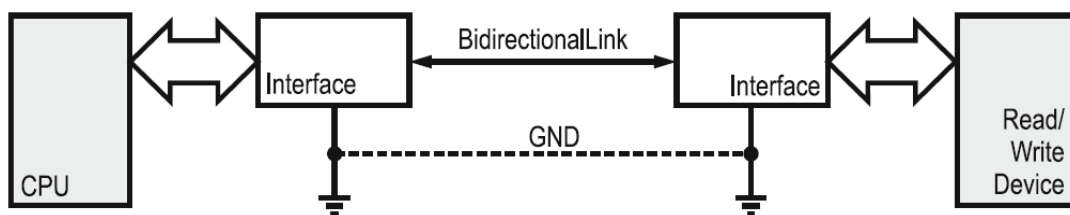
- Unidirekcioni (eng. *simplex*)
- Polu-bidirekcioni (eng. *Half-duplex*)
- Bidirekcioni (eng. *full-duplex*)

Unidirekcioni (eng. *simplex*) serijski kanal prenosi podatke samo u jednom smeru koristeći odgovarajući komunikacioni link. Na jednom kraju nalazi se predajnik, dok se na drugom kraju nalazi prijemnik (slika 11.4). Ovakvi kanali nemaju način za potvrdu i verifikaciju ispravnog prijema podataka od strane prijemnika. Primeri ovih kanala su radio i TV predajni sistemi, izlazne periferije poput LED ili displeja, ulazne periferije poput tastera i tastatura.

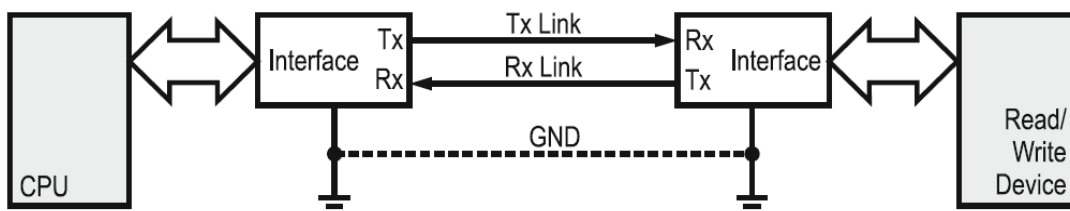


Slika 11.4 Unidirekcioni serijski kanal

Half-duplex serijski kanali koriste jedan komunikacioni link, ali se preko njega podaci mogu prenositi u oba smera pri čemu se u svakom trenutku podaci mogu prenositi samo u jednom smeru (slika 11.5). Kod ovih sistema na oba kraja nalaze se serijski primopredajnici (eng. *serial transceiver*), koji imaju mogućnost slanja i prijema podataka. Svaki put kada je potrebno promeniti smer prenosa, primopredajnici na oba kraja menjaju modove rada. Ova promena zahteva postojanje odgovarajućih pravila kako bi se izbegla situacija da na oba kraja primopredajnici pokušavaju da pošalju podatke. Ovaj skup pravila koji određuje ponašanje primopredajnika na oba kraja kanala i način na koji oni organizuju i interpretiraju podatke koji putuju kanalom naziva se **komunikacioni protokol**.

Slika 11.5 *Half-duplex* serijski kanal

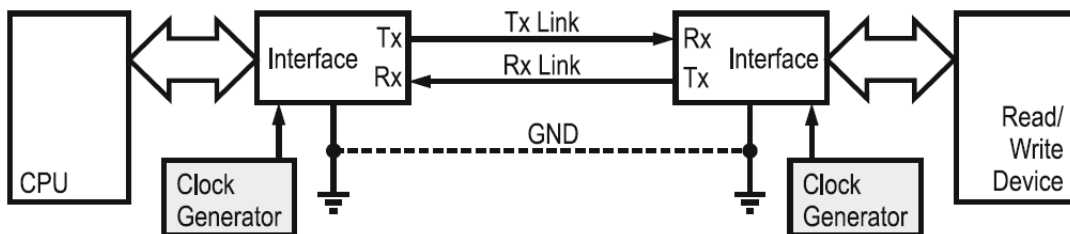
Bidirekcionni serijski kanali koriste dva posebna komunikaciona linka, jedan za prenos a drugi za prijem podataka, omogućavajući istovremeni prenos podataka u oba smera (slika 11.6). Uređaji na krajevima kanala imaju mogućnost istovremenog prijema i predaje podataka, ne zahtevajući nikakvu promenu u režimu rada, na taj način omogućavajući neprekidni bidirekcionni prenos podataka. Većina savremenih serijskih komunikacionih kanala koji se koriste u embeded sistemima koriste *full-duplex* kanale.

Slika 11.6 *Full-duplex* serijski kanal

Komunikacioni kanali prikazani na prethodnim slikama predstavljaju takozvane **point-to-point** topologije, kod kojih kroz jedan kanal komuniciraju tačno dva uređaja. Postoje i složenije topologije koje dozvoljavaju postojanje većeg broja uređaja koji međusobno komuniciraju koristeći jedan, zajednički kanal. Ovakve topologije poznate su pod imenom **multi-point** ili **multi-drop** topologije (npr. RS-422 ili RS-485).

Asinhrona serijska komunikacija

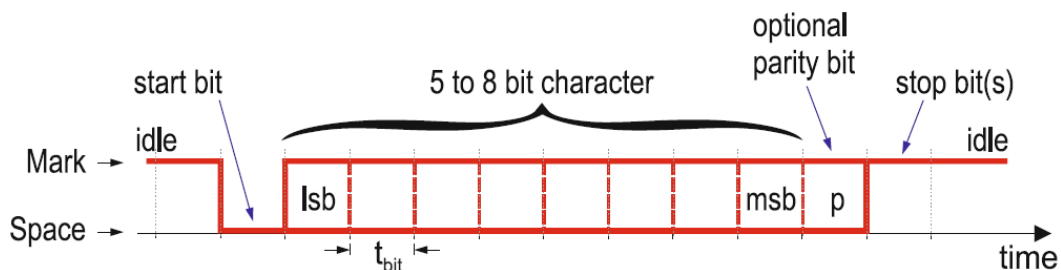
U asinhronim komunikacionim kanalima, na svakom kraju kanala postoji nezavisni izvor sinhronizacionog signala. Da bi se obezbedila pouzdana komunikacija, uređaji na oba kraja kanala konfigurisani su na takav način da generišu podatke **istom brzinom** i koriste **isti protokol** za razmenu paketa. Slika 11.7 prikazuje topologiju asinhronog, *full-duplex* serijskog kanala sa jasno vidljivim nezavisnim izvorima sinhronizacionog signala.



Slika 11.7 Sistem asinhronog serijskog prenosa

Asinhroni paketi podataka imaju vrlo jednostavnu strukturu. Njihovo zaglavlje sadrži samo jedan bit, koji se zove start bit. Vrednost start bita jednaka je logičkoj

nuli (“space” simbolu) i označava početak paketa. Telo paketa sadrži karakter, kodovan pomoću 5 do 8 bita organizovanih od bita najmanje značajnosti (LSB) do bita najveće značajnosti (MSB). Rep paketa sadrži opcioni bit parnosti, koji omogućava detekciju 1-bitnih grešaka koje su se javile prilikom prenosa kao i jedan ili više stop bita koji označavaju kraj paketa. Vrednost stop bita jednaka je logičkoj jedinici (“mark” simbolu) i označava kraj paketa. Ovo je ilustrovano na slici 11.8.



Slika 11.8 Struktura asinhronih paketa

Kada je asinhroni kanal u neaktivnom stanju (*idle state*, nema prenosa podataka) nalazi se u stanju logičke jedinice. Start bit ukazuje na tranziciju kanala iz neaktivnog u aktivno stanje. Validni start bit treba da ima trajanje od jednog bitskog vremena, t_{bit} . Kada prijemnik detektuje tranziciju sa logičke jedinice na logičku nulu, dok se nalazi u neaktivnom stanju, on je shvata kao početak novog paketa podataka. Da bi potvrdio činjenicu da je u toku prijem start bita, prijemnik obično proverava vrednost asinhronne komunikacione linije još jednom, nakon $t_{bit}/2$ vremena na sredini bitskog vremena, i ukoliko je vrednost linije i dalje na logičkoj nuli to je potvrda da je u toku prijem start bita. Nakon izvršene validacije start bita, prijemnik očitava (eng. *sample*) asinhronu liniju u regularnim vremenskim intervalima dužine t_{bit} sekundi, kako bi preuzeo preostale bitove tekućeg paketa.

Na predajnoj strani, brzina slanja individualnih bitova paketa određena je na osnovu lokalnog izvora sinhronizacionog signala. Na prijemnoj strani lokalni sinhronizacioni signal prijemnika određuje brzinu prijema podataka. Iako su predajni i prijemni izvori sinhronizacionog signala međusobno nezavisni, obe strane moraju da se “dogovore” o korišćenju iste brzine slanja/prijema podataka.

Uređaji na oba kraja linka obično koriste delitelje učestanosti kako bi generisali potrebno bitsko vreme na osnovu njihovih lokalnih izvora sinhronizacionog signala. Usled drifta koji se obično javlja kada se koriste nezavisni izvori sinhronizacionog signala, dužina asinhronog paketa se mora držati malom, a start bit je neophodan da bi se izvršila resinhronizacija predajnika i prijemnika. Takođe, da bi se obezbedila pouzdana detekcija vrednosti individualnih bitova, asinhrona linija se obično očitava (sempluje) sa učestanošću koja je k puta (16, 32, 64) veća od bitske brzine koja se koristi.

Detekcija greške

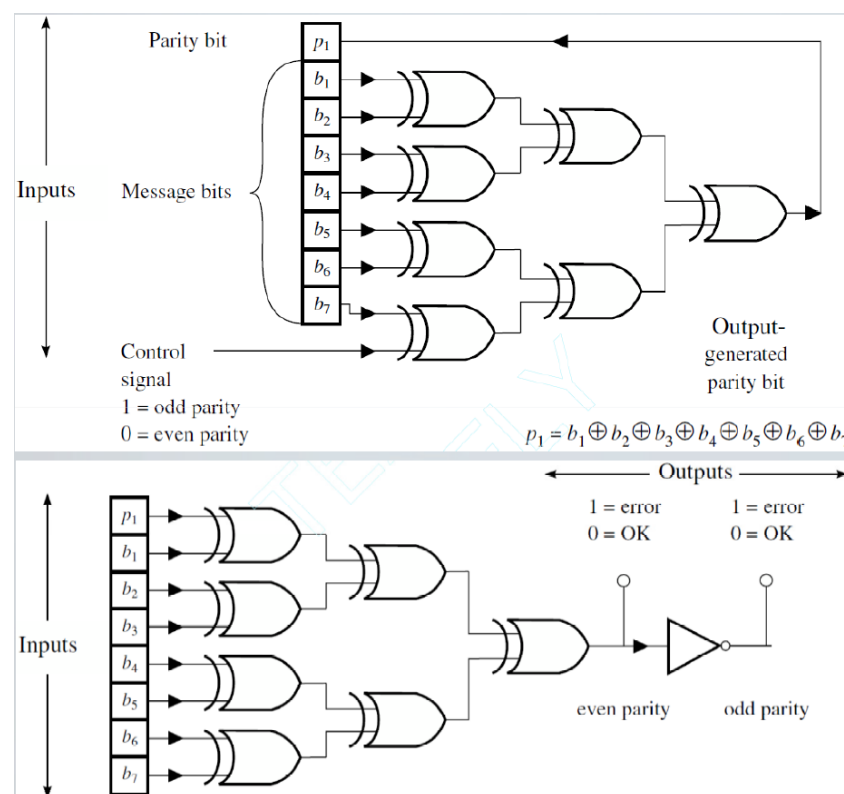
Detekcija grešaka prilikom asinhronog prenosa podataka ostvaruje se pomoću bita parnosti, koji je opcioni deo repnog dela paketa. Bit parnosti predstavlja elementarni mehanizam detekcije grešaka pomoću kojega je moguće detektovati **1-bitne greške** koje se mogu javiti prilikom prenosa podataka. Bit parnosti

predstavlja indikaciju parnog (eng. *even parity*) ili neparnog (eng. *odd parity*) broja logičkih jedinica u tekućem karakteru, plus bit parnosti, koji se prenosi. U slučaju *even parity* varijante, bit parnosti se postavlja na nulu ili jedinicu kako bi ukupan broj jedinica u karakteru koji se prenosi i bitu parnosti bio paran. U slučaju *odd parity* varijante, bit parnosti se postavlja na potrebnu vrednost koja će obezbediti da broj jedinica u karakteru zajedno sa bitom parnosti bude neparan.

Predajnik, prilikom slanja tekućeg karaktera računa i potrebnu vrednost bita parnosti, u zavisnosti od odabrane varijante (*even* ili *odd*) koju šalje u okviru repnog dela paketa.

Prijemnik ponavlja istu operaciju prilikom prijema karaktera. Ukoliko se izračunata vrednost bita parnosti na prijemnoj strani ne poklapa sa vrednošću koja je bila poslata, došlo je do pojave 1-bitne greške.

Logičke mreže za računanje bita parnosti su vrlo jednostavne i sastoje se iz odgovarajućeg broja XOR logičkih kapija, kao što je prikazano na slici 11.9.



Slika 11.9 Detekcija greške pomoću XOR kola

Formiranje paketa

Start bit, bit parnosti i stop bit automatski se dodaju od strane modula za serijsku komunikaciju na predajnoj strani. Na prijemnoj strani ovi bitovi se takođe automatski odstranjuju, pri čemu se vrši detekcija potencijalnih grešaka nastalih prilikom prenosa ukoliko se koristi bit parnosti. Korisnički softver ne mora da brine o ovim detaljima, jedino treba da obezbedi karakter koji je potrebno poslati preko

asinhronog serijskog linka ili obradi primljeni karakter. U slučaju detekcije greške, korisnički softver će odrediti eventualne neophodne korake za oporavak od greške.

Modul za asinhronu serijsku komunikaciju

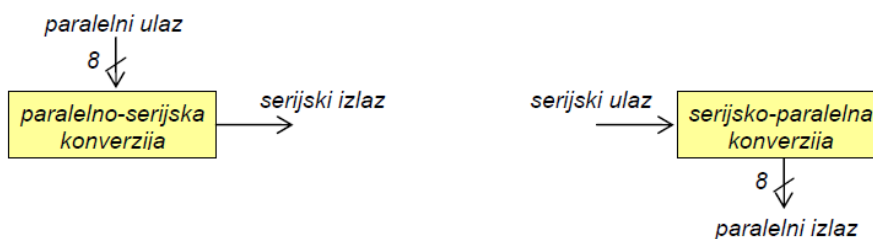
Modul za asinhronu serijsku komunikaciju poznat je pod nazivom UART (eng. *Universal Asynchronous Receiver and Transmitter*). Da bi se obezbedila komunikacija između dva uređaja pomoću asinhronog serijskog komunikacionog kanala oba uređaja moraju programirati njihove UART module na sledeći način:

- Da koriste istu bitsku brzinu
- Da koriste isti broj bita prilikom prenosa karaktera
- Da li će koristiti detekciju grešaka prilikom prenosa, i ako hoće, koji tip detekcije će biti korišćen (*even parity* ili *odd parity*)
- Koliki broj stop bita će biti korišćen za označavanje kraja tekućeg paketa

UART kombinuje funkcionalnost predajnika i prijemnika unutar jednog modula sa ciljem obezbeđivanja serijskog interfejsa sa *full-duplex* mogućnošću.

Kod serijskog prenosa podataka na predajnoj strani mora da se obavi paralelno–serijska, a na prijemnoj strani serijsko–paralelna konverzija. Na sledećoj slici levo prikazan je princip paralelno-serijske, a na slici desno serijsko-paralelne konverzije. Obe vrste konverzije zasnovane su na primeni pomeračkog registra sa serijskim i paralelnim ulazima i izlazima. Unutrašnje komponente UART-a nalaze se oko dva ovakva pomeračka registra (videti sliku 11.10):

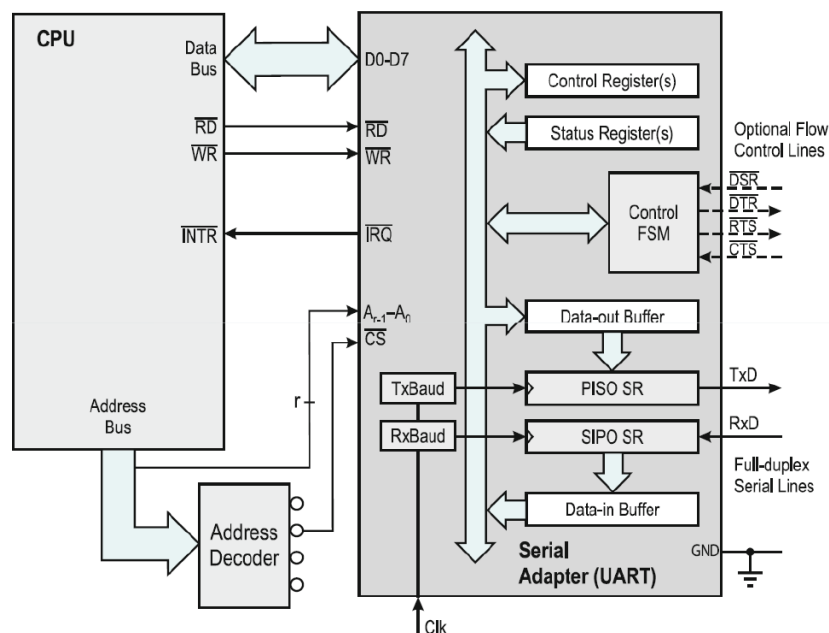
- PISO (eng. *Parallel-Input Serial-Output*) registra unutar predajnika
- SIPO (eng. *Serial-Input Parallel-Output*) registra unutar prijemnika



Slika 11.10 Konverzije podataka

Da bi inicirao prenos, CPU upisuje podatak (karakter) koji je potrebno preneti u *Data-out* bafer. Svaki karakter, jedan po jedan, se zatim okružuje sa odgovarajućim poljima zaglavlja i repa, što je funkcija koju realizuje upravljački automat (eng. *Control FSM*) i upisuje u PISO registar.

U ovom trenutku upravljački automat takođe računa i vrednost bita parnosti (ukoliko se on koristi). Sadržaj PISO registra se zatim šalje preko serijskog linka brzinom koju određuje *baud-rate* generator predajnika (TxBaud). Kada se svi karakteri iz *Data-out* bafera pošalju, UART podiže TxReady zastavicu, označavajući da je spreman za prijem novih karaktera koje je potrebno poslati preko serijskog linka. Upis u *Data-out* bafer automatski briše TxReady zastavicu. Jednostavni UART moduli imaju *Data-out* bafer u koji se može smestiti samo jedan karakter, dok je kod složenijih UART-a ovaj bafer veći i dostiže veličine i od nekoliko kilobajta. Struktura UART-a je prikazana na slici 11.11.



Slika 11.11 Struktura UART-a

Prilikom prijema, validni start bit resetuje SIPO registar i započinje upis bitova koji se primaju preko serijskog linka u SIPO registar. Kada se svi bitovi smešteni u SIPO registar, upravljački automat uklanja start i stop bitove, izvodi proveru parnosti (ukoliko je neophodna) i uklanja bit parnosti, postavlja primljeni karakter u *Data-in* bafer i podiže RxReady zastavicu. Kada detektuje da je RxReady zastavica aktivirana, CPU čita primljeni karakter iz *Data-in* bafera. Čitanje *Data-in* bafera obično briše RxReady zastavicu. U slučaju da su detektovane greške parnosti ili je paket nepravilno formiran, UART signalizira novonastalu situaciju podizanjem odgovarajućih zastavica unutar Status registra.

Ukoliko se prilikom prijema *Data-in* bafer puni brže nego što CPU iz njega čita karaktere, može doći do gubitka nekih od karaktera koji su bili poslani. Ovakva situacija poznata je pod nazivom “**overrun error**”. Da bi se izbegla ova situacija u praksi se koriste *Data-in* baferi koji imaju mogućnost prihvatanja većeg broja karaktera, na taj način oslobađajući CPU od potrebe za brзом reakcijom nakon indikacije prijema podataka PISO i SIPO registri rade na učestanosti predajnog i prijemnog takta koji određuju i brzine predaje i prijema podataka. Svi indikatori koji označavaju trenutni status predajnika, prijemnika i eventualno postojanje grešaka dostupni su preko odgovarajućih statusnih registara.

Konfigurisanje različitih parametara UART modula vrši se pomoću kontrolnih registara. Kontrolni registri omogućavaju konfigurisanje različitih opcija unutar UART modula kao što su:

- Dužina karaktera
- Uključivanje/isključivanje bita parnosti
- Tip parnosti koji se koristi (*even* ili *odd*)
- Dužinu stop bita
- Brzine predaje i prijema
- Konfigurisanje generisanja zahteva za prekidom
- Uključivanje/isključivanje predajnika

Statusni registri omogućavaju utvrđivanje statusa u kome se nalazi kanal i UART modul i obično sadrže sledeće indikatore:

- Indikatore grešaka (*buffer overrun, framing error, parity error*)
- Status predajnika i prijemnika (TxReady, RxReady)

Interfejsi UART-a

Interfejs UART modula sastoji se iz tri celine:

CPU interfejsa – koji omogućava povezivanje UART modula sa procesorom preko njegove adresne, kontrolne i magistrale podataka

Interfejs takta – koji obezbeđuje referentni izvor sinhronizacionog signala na osnovu kojega ce biti određene bitske brzine predajnika i prijemnika

Interfejsa serijskog kanala – preko kojeg se vrši povezivanje UART modula sa fizičkim serijskim linkom

UART se povezuje sa procesorom preko sledećih magistrala čineći CPU interfejs:

Magistrale podataka – skupa bidirekcionih linija podataka (obično je reč o 8-bitnoj magistrali podataka) koje služe za prenos podataka od i ka UART modulu

Read/write kontrolnih signala – koji određuju smer toka podataka

CS signala – preko kojega se aktivira UART modul

Adresne magistrale – skupa unidirekcionih selekcionih linija ($A_{r-1} - A_0$), preko kojih je moguće jednoznačno odrediti kojem unutrašnjem registru se želi pristupiti.

CPU interfejs obično uključuje i jednu ili više linija za generisanje zahteva za prekidom (IRQ) pomoću kojih UART modul može da generiše zahtev za opsluživanjem od strane procesora. U zavisnosti od modela UART-a postoji jedna ili više nezavisnih IRQ linija. U slučaju postojanja samo jedna IRQ linije, indikacija o uspešnoj predaji i prijemu podataka kao i indikacija o eventualnim greškama prilikom predaje i prijema signalizirala bi se preko iste IRQ linije. Prilikom obrade zahteva za prekidom CPU bi prvo morao da utvrdi razlog za prekidom (proveravajući sadržaj statusnih registara) kako bi utvrdio redosled akcija potreban za obradu prekida. Postojanje višestrukih IRQ linija omogućuje razdvajanje individualnih izvora prekida.

Svaki UART modul poseduje jednu ili više ulaznih linija takta koje su povezane na unutrašnje generatore bitskih brzina predajnika i prijemnika. Ovi takti ulazi obezbeđuju referentni sinhronizacioni signal učestanosti f_{clk} koji se zatim deli pomoću unutrašnjih delitelja takta (TxBaud i RxBaud) kako bi se generisale bitske brzine predajnika i prijemnika. Većina mikrokontrolera koji na sebi imaju integrisani UART modul, generiše potrebne bitske brzine na osnovu sistemskog takta, mada kod nekih postoji mogućnost korišćenja posebnih spoljašnjih izvora referentnog takt signala.

Glavne signalne linije interfejsa ka serijskom kanalu su TxD and RxD, koje zajedno sa linijom referentnog napona (GND) prenose dolazne i odlazne serijske tokove podataka ka i od UART modula. TxD (eng. *Transmitted Data*) predstavlja serijski izlazni port koji je direktno povezan na izlaz PISO registra. RxD (eng. *Received*

Data) predstavlja serijski ulazni port koji je direktno povezan na ulaz SIPO registra. Kod većine mikrokontrolera TxD i RxD signali su multipleksirani sa GPIO signalima ili nekim drugim signalima tako da je potrebno izvršiti odgovarajuća konfigurisanja mikrokontrolera ukoliko se žele koristiti.

Neki UART moduli mogu uključiti i dodatne signale unutar interfejsa serijskog kanala pomoću kojih je moguće obezbediti hardversku kontrolu toka podataka (eng. *hardware flow control*):

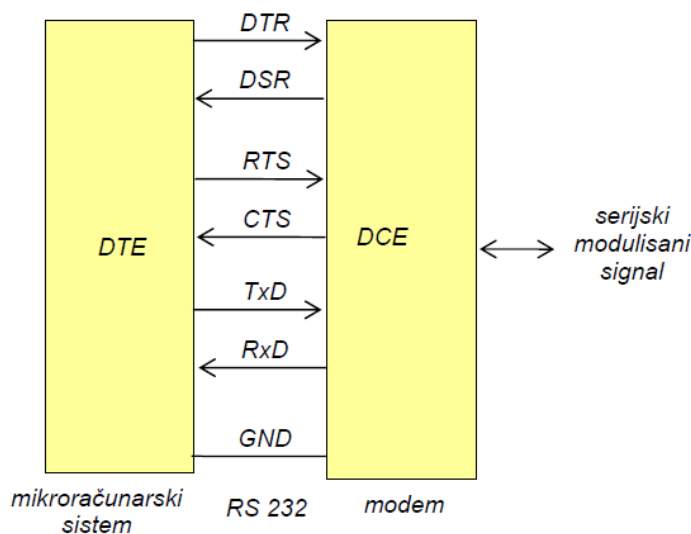
- DSR – (eng. *Data Set Ready*), ulazni signal UART modula koji predstavlja indikaciju da je spoljašnji komunikacioni modul spreman za rad

- DTR – (eng. *Data Terminal Ready*), izlazni signal UART modula koji predstavlja indikaciju da je UART modul spreman za rad

- RTS – (eng. *Ready to Send*), izlazni signal UART modula koji obaveštava spoljašnji uređaj da želi da izvrši slanje podataka

- CTS – (eng. *Clear to Send*), ulazni signal UART modula koji predstavlja indikaciju da je spoljašnji komunikacioni modul spreman da primi podatke

Ovi dodatni signali, zajedno sa RI (eng. *Ring Indicator*) i DCD (eng. *Data Carrier Detected*) čine skup signala koji definišu dobro poznati RS-232 komunikacioni interfejs (slika 11.12), međutim većina savremenih mikrokontrolera ih ne podržava direktno.



Slika 11.12 RS232 interfejs

Naponski nivoi

Svi signali koji čine interfejs serijskog kanala moraju da koriste logičke naponske nivoe koji se koriste unutar UART modula. Na primer, ukoliko UART modul radi na naponu napajanja od 5V, on će biti u stanju da radi i toleriše TTL kompatibilne signale. Ukoliko je UART deo nekog mikrokontrolera koji radi na 3.3V, on će biti u stanju da prihvata signale iz opsega 0-3.3V. Korišćenje logičkih naponskih nivoa za primopredaju serijskih podataka u mnogome će smanjiti maksimalno rastojanje na koje ovi podaci mogu da “putuju” na desetine cm, pre nego što oni postanu previše “zaprljani” šumom, pogotovo u slučaju velikih transmisionih brzina.

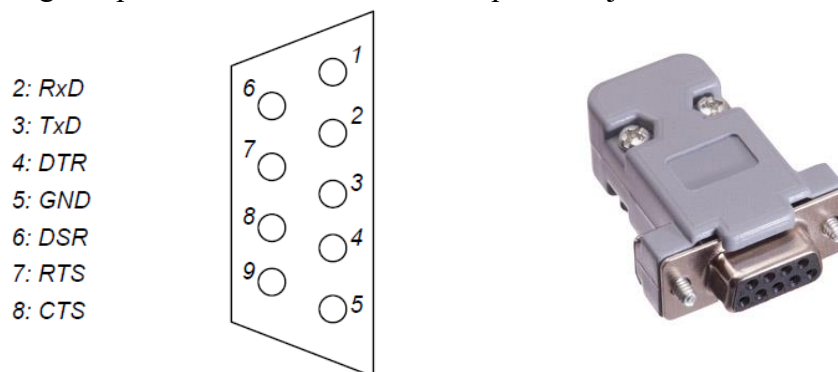
Da bi se obezbedila mogućnost prenosa serijskih podataka na veće udaljenosti moraju se koristiti drugačiji naponski nivoi i formati signala.

Fizički standardi za serijsku komunikaciju obično koriste veliki dinamički opseg naponskih nivoa i/ili odgovarajuće signalizacione mehanizme u cilju povećanja "otpornosti" na uticaje šuma prilikom prenosa podataka kroz kanal. Na primer, u slučaju RS-232 kanala, naponski nivoi koji se koriste za prenos "nule" i "jedinice" su -12 V i +12 V. Ovi naponski nivoi nisu direktno kompatibilni sa logičkim nivoima signala koje očekuje ili generiše UART modul. Zbog toga je u većini ovakvih sistema potrebno korišćenje naponskih translatora (eng. *Level Shifter*) kako bi se ovo povezivanje ostvarilo na način koji će obezbediti pouzdan rad i neće dovesti do oštećenja komunikacione opreme.

Mehanička specifikacija

Originalni standard RS 232 predvideo je korišćenje konektora sa 25 nožica (naziva se DB25), koji podržava dva dupleks serijska kanala. Međutim, pošto je u praksi obično korišćen samo jedan kanal, vremenom je konektor od 25 nožica zamenjen konektorom sa 9 nožica (koji se naziva DB9).

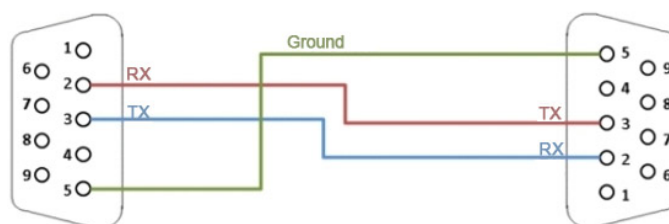
Raspored signala po nožicama konektora DB9 prikazan je na slici 11.13.



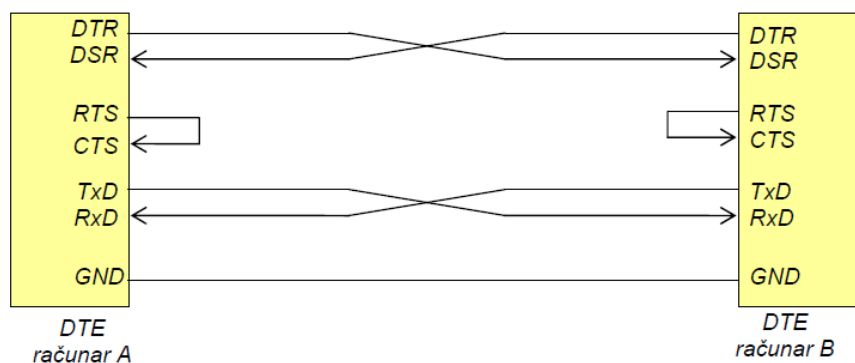
Slika 11.13 Konektor za povezivanje putem serijske komunikacije

Povezivanje dva računara bez modema

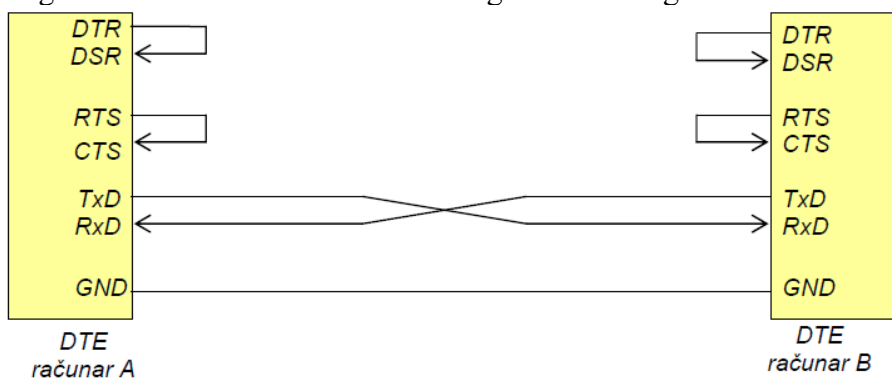
Ako su dva mikroračunara sa međusobnom serijskom vezom dovoljno blizu, moguće je prenositi digitalne signale u originalnom obliku, bez korišćenja modema za modulaciju serijskog signala. U ovom slučaju RS 232 signali dva mikroračunara mogu se neposredno spojiti provodnicima. Pošto u takvoj vezi nema modema, ona se obično naziva **null modem** veza a kabel sa provodnicima naziva se null modem kabel.



Na sledećoj slici prikazana je veza bez modema između mikroračunara A i mikroračunara B sa rukovanjem signalima DTR i DSR, pri čemu je izlazni signal RTS kratko vezan na ulazni signal CTS istog mikroračunara.



Na sledećoj slici prikazana je jednostavna veza bez modema između mikroračunara A i mikroračunara B bez rukovanja sa signalima DTR i DSR, pri čemu je takođe izlazni signal RTS kratko vezan na ulazni signal CTS istog mikroračunara.



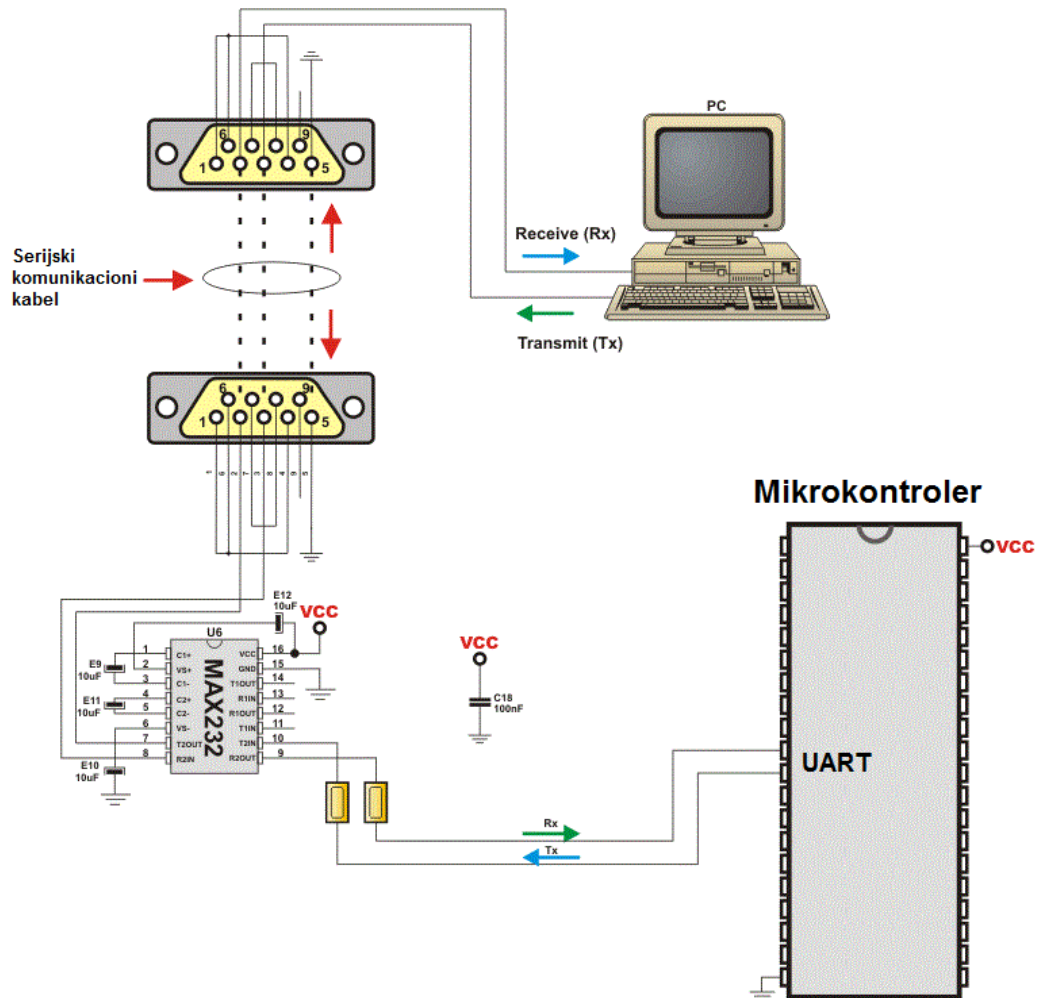
Povezivanje računara i mikrokontrolera

Da bismo povezali računar i mikrokontroler potrebne su sledeće komponente (pogledati narednu sliku):

- računar
- pretvarač naponskog nivoa
- serijski kabel
- mikrokontroler

Konvertor napona

Kao što je već rečeno električne karakteristike signala u mikroračunarskom sistemu drugačije su od električnih signala definisanih standardom RS232. Signali mikroračunara, na primer signali mikrokontrolera x51, kompatibilni su sa TTL standardom, pa je između mikroračunara i uređaja sa signalima koji su u električnom pogledu kompatibilni sa RS 232 standardom potrebno staviti konvertor za električno prilagođenje signala ova dva standarda.



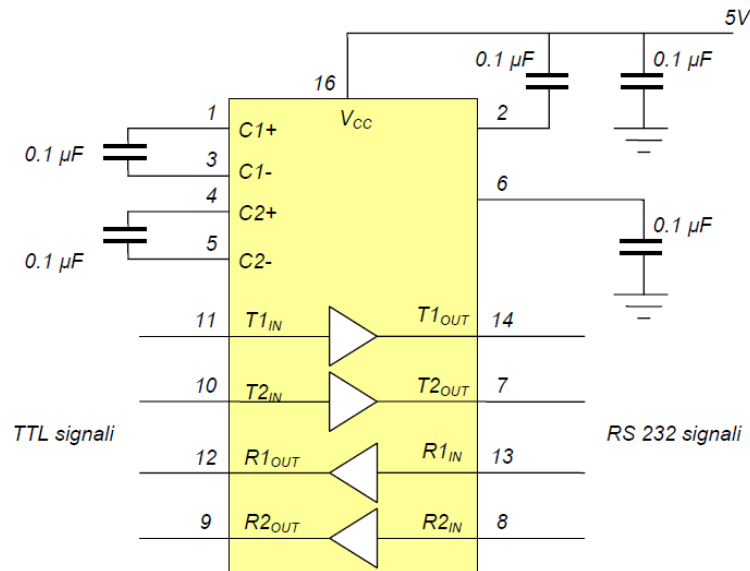
Konverzija napona sa TTL ulaznih signal na RS 232 kompatibilne izlazne signale treba da se obavi u skladu sa sledećom specifikacijom:

logički nivo	TTL napon [V]		RS 232 napon [V]	
	min	max	min	max
logička 0	0	0.8	3	25
logička 1	2.4	5.0	-3	-25

Konverzija napona sa RS 232 ulaznih signal na TTL kompatibilne izlazne signale treba da se obavi u skladu sa sledećom specifikacijom:

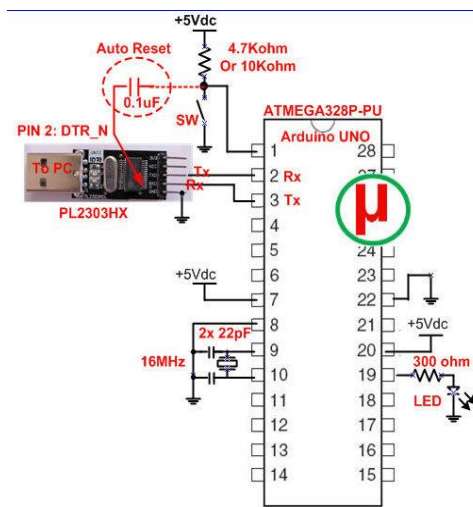
logički nivo	RS 232 napon [V]		TTL napon [V]	
	min	max	min	max
logička 0	3	25	0.0	0.4
logička 1	-3	-25	2.8	5.0

Kao primer konvertora napona uzećemo kolo MAXIM MAX232E, čiji blok dijagram je prikazan na sledećoj slici. Ovo kolo ima dva konvertora T1 i T2 koji digitalne signale ulazne signale T1IN i T2IN kompatibilne sa TTL standardom, prevode u digitalne izlazne signale T1OUT i T2OUT koji su kompatibilni sa RS 232 signalima. Druga dva konvertora R1 i R2, digitalne ulazne signale R1IN i R2IN kompatibilne sa RS232 standardom, prevode u digitalne izlazne signale R1OUT i R2OUT koji su kompatibilni sa TTL signalima.



Zanimljivo je da kolo MAX232E ima samo jedno napajanje, +5 V. Interne naponske pumpe za čiji rad su potrebni kondenzatori prikazani na slici, napon od +5V konvertuju u napone +10 V i -10 V koji su potrebni za izlazne signale koji su kompatibilni sa RS 232 standardom. Na slici su prikazani brojevi spoljnih priključaka kola MAX232E i nazivi spoljnih signala.

Današnji personalni računari i laptopovi najčešće nemaju serijski port na raspolaganju, a imaju USB port. Tada se može koristiti konvertor USB-RS232 koji nakon instalacije formira virtuelni serijski port na personalnom računaru. Primer je dat na narednoj slici.



Primer povezivanja preko Max232 na Arduino je dat na narednoj slici.

