ASINHRONA SERIJSKA KOMUNIKACIJA IZMEĐU PC-A I PLC INDUSTRIJSKIH KONTROLERA

Zoran Milić (zrondjul@yahoo.com), Petar Nikolić (nikpetar@tigar.com), *Tigar MH*,
Miljana Sokolović (miljana@venus.elfak.ni.ac.yu), *Elektronski Fakultet, Niš*

Sadržaj – *U radu je opisana asinhrona serijska komunikacija između industrijskih kontrolera i PC računara. Pri projektovanju sopstvenog HMI intefejsa za upravljanje mašinom razvijena je Windows aplikacija za vizuelizaciju i akviziciju podataka. Predstavljeni su osnovni principi serijske asinhrone veze i dati primeri konkretne primene ovog protokola za čitanje i upis u registre PLC kontrolera. Pri razvoju posebnog SCADA okruženja ovaj protokol se koristi za komunikaciju PC-a sa interfejsom industrijske mreže koju čine PLC kontroleri.*

1. UVOD

Za vizuelizaciju interfejsa operatera prema mašini upravljanoj PLC-om moguće je koristiti različita hardverska rešenja, od industrijskih panela, namenjenih isključivo za ove svrhe i vezanih za određeni tip PLC-a, do industrijskih, ali i klasičnih PC računara na kojima je instaliran odgovarajući operativni sistem i na kojima radi određeni aplikativni program. Za realizaciju je pogodnije koristiti PC računare, jer to omogućava proširivost sistema i integraciju u širi npr. informacioni sistem.

Industrijski paneli specifične namene nemaju mogućnost multi-tasking rada. Oni zahtevaju dodatnu komunikaciju sa računarima ili drugim kontrolerima u upravljačkim sistemima u kojima postoji potreba za složenim proračunima. Na osnovu tih proračuna se vrši izmena sadržaja registara kontrolera i aktiviranje odgovarajućih izlaza. Ovi paneli nemaju dodatne portove, a ni mogućnost ugradnje dodatnih kartica (PCI, PCI express) koje isporučuju proizvođači senzora i aktuatora. Kada su industrijski paneli vezani u širu industrijsku mrežu, ukoliko jedan od uređaja blokira slanje poruka, tada može doći do prekida komunikacije na celoj mreži.

Za komunikaciju PC-a sa kontrolerom moguće je koristiti komunikacione aplikacije proizvođača kontrolera ili razviti sopstveni komunikacioni drajver. Komunikacioni softver omogućava laku integraciju u aplikacije za upravljanje mašinom ili SCADA softvere, čime se smanjuje vreme potrebno za realizaciju celokupnog sistema. Neki od tih komunikacionih softvera su istovremeno i OPC serveri tako da omogućavaju relativno laku integraciju izvora podataka različitih proizvođača u sisteme za akviziciju i vizuelizaciju.

Pisanje sopstvenog komunikacionog drajvera omogućva dobro upravljanje kontrolerom. Moguće je upravljati modom rada kontrolera, tj. vršiti prebacivanje između program, run i test modova što je korisno prilikom testiranja aplikacije. Upravljanje upload-om i download-om koda ka i od kontrolera je takođe moguće direktno preko serijskog interfejsa ili kroz mrežu preko odgovarajućeg mrežnog modula. Na ovaj način ostvaruje se i promena statusnih registara, upravljanje restartom ili gašenjem kontrolera, i inicijalizacija memorije. Promenu parametara samog komunikacionog linka, (npr. baud rate, parity bit, stop bit, hardware/software control, BCC/CRC) je moguće ostvariti direktno, i to predstavlja prednost pri testiranju sistema. HMI (Human Machine Interface) aplikacija koja komunicira sa kontrolerom direktno preko drajvera komunicira u realnom vremenu.

Ovakvo rešenje ima bolju dijagnostiku, omogućuje adekvatan real-time odziv mašine, pruža veću nezavisnost ali i povećava vreme projektovanja celog sistema. Pored vremena projektovanja sistema kao parametar prilikom izbora rešenja, za komunikaciju, bitan faktor predstavlja i cena. U slučaju projektovanja SCADA aplikacije cene oba sistema su uporedive obzirom da je broj mesta u sistemu sa kojih se vrši sakupljanje podataka – a samim tim i mesta na kojima je neophodan komunikacioni softver, relativno mali u odnosu na broj PLC kontrolera u sistemu. Kod HMI aplikacija situacija je potpuno drugačija. Broj mesta na kojima je neophodan komunikacioni softver jednak je broju PLC kontrolera – tj. svaka mašina mora imati PLC i odgovarajući interfejs prema operateru, a koji mora komunicirati sa tim PLC-om. U ovom slučaju cena izrade sopstvenog drajvera neuporedivo je niža od kupovine gotovog rešenja.

2. MREŽNI SLOJEVI KOMUNIKACIONOG MODELA

U zavisnosti od izabrane mrežne arhitekture industrijska mreža ima određen broj slojeva i na nivou svakog od njih određeni protokol. U serijskoj komunikaciji najčešće postoje četiri sloja. To su fizički sloj, sloj veze podataka, sloj mreže i aplikacijski sloj [1].

Glavna uloga fizičkog sloja jeste da dobijeni niz bitova prenese duž komunikacionog kanala. Fizički sloj sačinjava skup medijuma i interfejsa između njih, koji služe da formiraju kanal za prenos podataka između čvorova na mreži [2]. U slučaju serijske RS232/RS422 komunikacije fizički sloj čine RS232/RS422 kabal i RS232/RS422 portovi PC-a i PLC kontrolera koji su međusobno povezani tim kablom, naponski signali (nule i jedinice), broj bitova podataka, broj stop bitova, bit parnosti, bitska brzina, način uspostavljanja i način prekida veze nakon što PC i PLC završe prenos podataka [3].

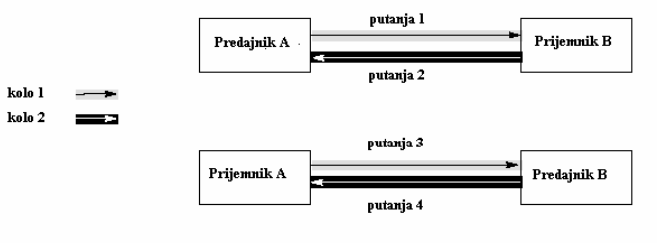
Uloga sloja veze podataka je da uobliči niz bitova koje treba poslati u odgovarajuće okvire kako bi se mogla vršiti kontrola isparavnosti slanja podataka – tj. kontrola greške. Ovaj sloj je zadužen za kontrolu ispravnosti slanja podataka. To znači da. protokol na nivou ovog sloja treba da obezbedi mehanizam za potvrdu ispravnog slanja i ispravnog prijema podataka [2].

Sloj mreže je zadužen za upravljanje rutiranjem paketa na mreži. On je odgovoran za upravljanje paketa do njegovog odredišta, odnosno za uspostavljanje konekcije između čvorova na mreži [4].

Aplikacijski sloj sadrži protokole potrebne korisnicima – aplikacijama za izvršenje određenih radnji [2]. U slučaju PLC i PC komunikacije on sadrži komande koje jedna drugoj zadaju PC i PLC aplikacije.

3. DF1 SLOJ VEZE PODATAKA

DF1 predstavlja Allen Bradley protokol sloja veze podataka, koji se zasniva na ANSI x3.28 specifikaciji. Najosnovniji princip DF1 protokola biće objašnjen na primeru Full Duplex razmene podataka [1].

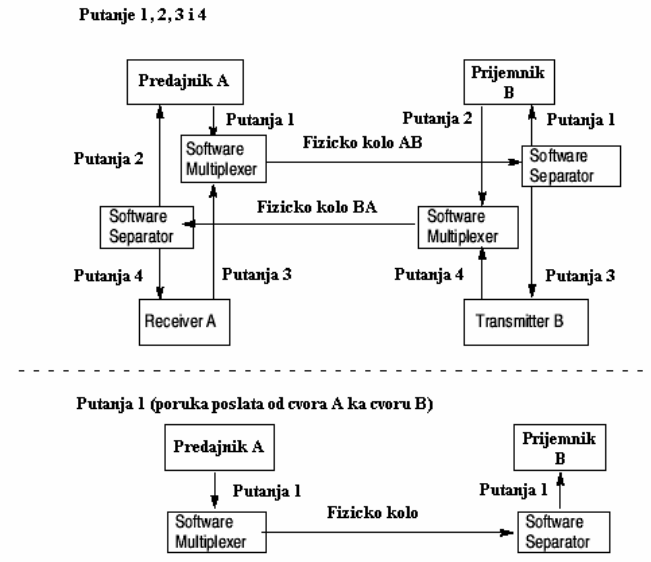


Slika 1. *Putanje podataka za dvosmernu simultanu komunikaciju (Full duplex)*

Kod Full Duplex protokola (slika 1), link koristi dva odvojena fizička kola za istovremenu razmenu poruka. Ova dva fizički odvojena kola obezbeđuju komunikaciju na četiri komunikaciona kanala.

- U prvom kolu predajnik A šalje poruke prijemniku B (putanja 1) i prijemnik A šalje povratne kontrolne poruke predajniku B (putanja 3)
- U drugom kolu predajnik B šalje poruke prijemniku A (putanja 4) i prijemnik B šalje povratne kontrolne poruke predajniku A (putanja 2)

Sve poruke i simboli u svakom od kola prenose se u jednom smeru; u prvom od A do B, a u drugom od B do A.

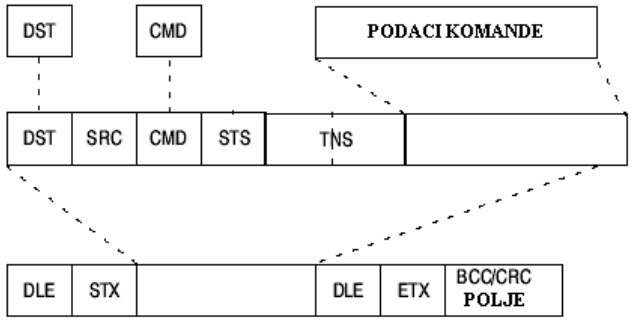


Slika 2. *Softverska implementacija putanja podataka*

- Da bi se implementirale četiri logičke putanje u dva fizički nezavisna kola potreban je softverski multiplekser – on služi da kombinuje komandne poruke (koje šalje predajnik) sa povratnim porukama tj. odgovorima (koje šalje prijemnik) koji se šalju u istom smeru.
- Na drugom kraju linka softver separator razdvaja komandne poruke od povratnih poruka. Softver separator treba da šalje komandne poruke odgovarajućem prijemniku, a povratne poruke odgovarajućem predajniku.
- Iako komandne poruke i povratne poruke u istom kolu funkcionišu nezavisno jedne od drugih, ipak između njih postoji izvesna povezanost. Npr. komandna poruka u kolu AB biće zakašnjena ukoliko je povratna poruka prijemnika A ubačena u sekvencu komandnih poruka predajnika A. Takođe će i svaki hardverski problem koji utiče na komandne simbole u jednom kolu uticati i na povratne simbole u tom istom kolu [1].

4. GENERISANJE I SEPARACIJA OKVIRA PODATAKA U FULL DUPLEX PROTOKOLU

Okvir poruke u Full Duplex protokolu ima različite forme u zavisnosti od toga na kom mrežnom sloju se posmatra, a s obzirom da se na različitim mrežnim slojevima generišu različiti delovi poruke. Slika 3. ilustruje na koji način određeni mrežni sloj utiče na generisanje okvira poruke [5]. Na slici nije prikazan uticaj fizičkog sloja.



Slika 3. *Prikaz okvira za podatke, odozgo na dole: sloj aplikacije, sloj mreže, sloj veze podataka*

Na aplikacionom sloju generiše se komanda (npr. upis ili čitanje), odredište (adresa kontrolera na mreži) i podaci vezani za komandu (npr. ukoliko čitamo to bi bila specifikacija početne adrese u memoriji kao i veličina bloka za čitanje).

Sloj mreže je odgovoran za uspostavljanje konekcije između čvorova koji komuniciraju, tako da se ovde dodaju adresa sa koje se vrši slanje poruke (kako bi povratna poruka imala ispravnu odredišnu adresu), polje o statusu prenosa (koje sadrži kod greške nastale pri prenosu, ukoliko je naravno došlo do greške) kao i jednoznačni identifikator – jedinstven za svaku poruku (kako bi se znalo koja je povratna poruka odgovor datoj izdatoj komandi).

Sloj veze podataka odgovoran je za kontrolu ispravnosti prenetih podataka – ovde se dodaje polje za početak i kraj poruke kao i polje za kontrolu greške.

Da bi se projektovalo odgovarajuće HMI rešenje potrebno je napisati drajver koji pokriva širok spektar komunikacionih poruka kako bi se omogućilo čitanje i upis u registre kontrolera, upravljanje modom rada kontrolera, promena statusnih registara, promena parametara samog komunikacionog linka, i da se istovremeno pokriva i relativno širok spektar uređaja za komuniciranje.

Drajver je pisan namenski za asinhronu Full Duplex serijsku komunikaciju sa Allen Bradley PLC 5 i SLC 5 familijama kontrolera kao i za Allen Bradley KF2 komunikacioni modul [6].

Drajver podržava je sledeći set komandi koje:

- Za PLC 5 familiju kontrolera i KF2 modul:
    - "Word Range Read" – čitanje bloka reči iz memorije kontrolera
    - "Word Range Write" – upis bloka reči u memoriju kontrolera
    - "Typed Read" – čitanje bloka podataka iz memorije kontrolera (ova komanda je istovremeno podržana i od strane SLC 5/03 i SLC 5/04 procesora iz familje SLC500)
    - "Typed Write" – upis bloka podataka u memoriju kontrolera (ova komanda je istovremeno podržana i od strane SLC 5/03 i SLC 5/04 procesora iz familje SLC 500)
    - "Read – Modify - Write" – bit upis komanda
    - "Set Variables" – podešava parametre serijskog linka – broj ENQ paketa, broj NAK paketa I timeout vreme
    - "Set CPU Mode" – promena moda rada kontrolera: Test, Program i Run
    - "Diagnostic Status" – čitanje sadržaja statusnih registara kontrolera
- Za SLC 500 i MicroLogix 1000 familiju konrolera:
    - "Protected Typed Logical Read With Three Address Fields" – čitanje bloka podataka iz memorije kontrolera počev od zadate adrese
    - "Protected Typed Logical Write With Three Address Fields" – upis bloka podataka u memoriju kontrolera počev od zadate adrese
    - "Change mode" – promena moda rada kontrolera: Test, Program i Run
    - "Diagnostic Status" – čitanje sadržaja statusnih registara kontrolera

5. PRIMER

Između PC-ja i PLC kontrolera u sistemima za upravljanje mašinama u gumarskoj industriji koristi se asinhrona serijska komunikacija. Ulogu interfejsa između kontrolera i operatera obavlja aplikacija na industrijskom PC računaru. Razlog za to je što je u sistemu neophodna dodatna procesorska snaga za obradu parametara dobijenih od pametnih senzora, a što ili većina PLC kontrolera ne može da zadovolji, ili je cena implementacije previsoka.

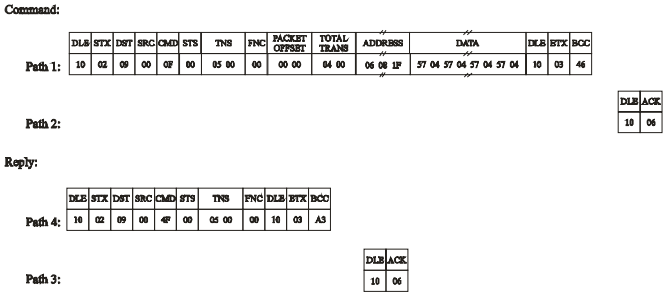
Komunikacija u toku jednog radnog ciklusa mašina različitih namena se obavlja na sličan način. Kontroler šalje zahtev za podacima recepture od korisničke aplikacije preko asinhrone serijske komunikacije. Podaci se čitaju sa odgovarajućih senzora i na osnovu njih aplikacija čita odgovarajuće parametare iz baza podataka i proračunava parametre recepture. Na osnovu ovih podataka vrši se podešavanje mašine nakon čega sledi početak odgovarajućeg ciklusa rada mašine. U toku ciklusa može postojati više komunikacionih sesija između kontrolera i PC-a. U toku jedne sesije se vrši određen broj sukcesivnih upisa podataka, potrebnih za upravljanje mašinom, u registre kontrolera, kao i čitanja registara radi dobijanja podataka potrebnih za dodatnu analizu u PC-ju.

Posle završetka ciklusa i analize koju vrši korisnička aplikacija, vrši se upis rezultata obrade u bazu i šalje signal kontroleru za inicijalizaciju narednog ciklusa.

Opisana HMI aplikacija implementirana je korišćenjem Microsoft Visual Basic. NET 2005 razvojnog okruženja [7].

U primeru je prikazano upisivanje četiri celobrojne reči 1111 (decimalno) u Allen Bradley PLC 5/20 procesor preko serijskog linka, počev od memorijske lokacije N8:31.Za ovo je korišćena "Word Range Write" komanda. Na sličan način obavlja se i celokupna komunikacija preko serijskog linka, pri čemu se u pakete ubacuju odgovarajuće komande, adrese i podaci. Tom prilikom vrši se sledeća razmena podataka (slika 4.):

- Aplikacija šalje podatak za upis drajveru, tj. specificira komandu upisa, sve četiri reči podataka i odredišnu adresu kontrolera
- Drajver generiše celokupni okvir podataka za slanje.
- Predajnik drajvera šalje paket prijemniku PLC kontrolera
- Prijemnik PLC kontrolera prima paket, prosleđuje ga procesoru na obradu, a na link šalje kontrolnu poruku (DLE ACK) o uspešnom prijemu paketa.
- Pošto je izvršena komanda (tj. obavljen upis) PLC procesor prosleđuje predajniku poruku o uspešnom izvršenju.
- Predajnik kontrolera šalje poruku na link.
- Prijemnik drajvera prima poruku o uspešnom upisu, aplikaciji prosleđuje poruku o uspešnom izvršenju, a na link šalje kontrolnu poruku (DLE ACK) o uspešnom prijemu paketa.



Slika 4. *Prikaz razmene paketa preko serijskog linka*

Paket se sastoji od sledećih komandi:

- DLE STX – početak paketa sloja veze podataka
- DST – odredišna adresa kontrolera
- SRC – adresa stanice koja šalje paket
- CMD – tip komande (tj. grupu komande)
- STS – status poruke (komandan aporuka uvek treba da sadrži 00 (hexa) u ovom polju. Povratna poruka ukoliko dođe do greške pri prenosu (ili je nešto loše specificirano u samoj poruci) sadrži kod greške u ovom polju
- TNS – šesnaestobitni brojač koji se inkerementira prilikom slanja svake poruke. Ovo polje jednoznačno identifikuje poruku, tj. ukoliko se jave dve poruke sa istim DST, SRC i TNS poljima druga poruka se smatra duplikatom prve poruke
- FNC – koristi se u kombinaciji sa CMD poljem. Služi da specificira komandu iz grupe komandi specificirane od strane CMD polja
- Packet Offset – sadrši ofset u odnosu na aadresu specificiranu u polju Address
- Total Trans – sadrži ukupan broj reči podataka za upis u memoriju kontrolera u celokunoj transakciji
- Address – specificira početnu adresu u memoriji kontrolera od koje se vrši upis. Kod PLC kontrolera postoje: Logičko adresiranje, ASCII Logičko adresiranje, Fizičko adresiranje i Floating Point adresiranje. Ovde je primenjeno Logičko adresiranje.
- SIZE – specificira broj reči za upis ovom komandom
- DLE ETX - kraj paketa veze podataka
- BCC – Block Check, moguće je izabrati između CRC i BCC kontrole greške. BCC bajt predstavlja komplement dvojke osmobitne sume svih bajtova između DLE STX i DLE ETX polja. Kod računanja CRC – a koristi se $X^{16} + X^{15} + X^2 + X^0$

6. ZAKLJUČAK

Razvijanje sopstvenog komunikacionog drajvera za komunikaciju PC-a sa PLC kontroleroma pri projektovanju HMI intefejsa za upravljanje mašinom omogućuje bolju dijagnostiku, adekvatan real time odziv mašine i pruža veću nezavisnost u odnosu na upotrebu komunikacione aplikacije proizvođača kontrolera. Izrada sopstvenog rešenja i njegova integracija u HMI aplikacije povećava vreme projektovanja celog sistema, ali je u isto vreme cena izrade sopstvenog drajvera dosta niža od kupovine gotovog rešenja i nisu potrebne dodatne licence za svaku narednu instaliranu aplikaciju. U radu je dat prikaz komunikacionog modela koji koristi asinhronu serijsku komunikaciju i opisana je jedna praktična realizacija upisa podataka potrebnih za upravljanje mašinom, dobijenih u PC-ju, u registre kontrolera, i čitanja podataka potrebnih za dodatnu analizu u PC-ju, ili za osvežavanje podataka koji su prikazani na ekranu.

LITERATURA

[1] -, “DF 1 Protocol and command set – Reference Manual”, Publication 1770 – 6. 5. 16, Allen_Bradley, Milwaukee, USA, October 1996.

[2] Andrew S. Tanenbaum, “Computer Networks”, London, Prentice Hall PTR, 2002.

[3] -, “Data Highway or Data Highway Plus Asynchronous (RS-232-C or RS-422-A) Interface Module” User Manual, Allen_Bradley, Milwaukee, USA, March 1989

[4] Anthony Chiarella, “Umrežavanje pomoću Cisco i Microsoft tehnologija”, Čačak, Kompjuter biblioteka, 2005.

[5] -, “Data Highway Plus and DF1 Communication Protocols”, Allen_Bradley, Milwaukee, USA, 2004

[6] -, “Allen-Bradley DF1 Serial Communication Interface API”, DASTEC Corporation, 2003

[7] Michael Halvarson, “Visual Basic.NET Step by Step”, CET Computer Equimpent and Trade, 2002.

Abstract – The basic principles of the asinhronius serial communication between PLC controllers and PC based applications are presented in this paper. In order to create HMI interface between operator and controller at the machine, the Windows based application for vizualization and data acquisition was developed. This application is using self-oriented driver for serial communication which was developed and used for purposes of communication and data exchange between industrial controllers working at machines in the plant floor, and PCs. Basic principle of generating communication packets and examples of writing data into PLC’s memory registers are given. This driver is also used in self-oriented SCADA programming for data exchange between SCADA application and controllers on the industrial network.

ASINHRONIUS SERIAL COMMUNICATION BETWEEN INDUSTRIAL CONTROLLERS AND PC’S

Zoran Milić, Petar Nikolić, Miljana Sokolović