

JEDNA REALIZACIJA INDUSTRIJSKE ETHERNET KOMUNIKACIJE SA TOLERANCIJOM PROLAZNIH OTKAZA

A REALIZATION OF INDUSTRIAL ETHERNET COMMUNICATION WITH TRANSIENT FAULTS TOLERANCE

Sandra Došić, Milun Jevtić, Milunka Damnjanović, *Elektronski fakultet Niš*

Sadržaj - *Ethernet mreže koje koriste switch-eve su sve zastupljenije u industrijskim upravljačko-nadzornim sistemima. Uz rad u realnom vremenu od njih se zahteva da sa visokom pouzdanošću ostvare komunikaciju. Zato je u radu predložen jedan način realizacije komunikacije koja ima i mogućnost tolerisanja otkaza. Pretpostavlja se da se otkazi mogu desiti na prenosnim linkovima i da se mreža oporavlja tj. toleriše otkaze ponovnim slanjem paketa uz zadovoljenje vremenskih ograničenja rada u realnom vremenu. Analizirana je mogućnost prevazilaženja otkaza korišćenjem vremenske redundanse. U radu je predstavljen i algoritam koji opisuje ceo proces komunikacije u realnom vremenu sa mogućnošću tolerisanja otkaza.*

Abstract - *Switched Ethernet network has been considered as a promising network technology for industrial control and monitoring systems. These systems work in real-time so they need high communication reliability. This paper presents one realization of fault tolerant communications over the switched Ethernet. We assumed that faults occur on transmission links and that network recovers from the faults retransmit frame again. In that situation all real-time transmission requirements must be met. We analyze possibility to overcome faults using time redundancy. We also propose algorithm for fault tolerant real-time communication on a switched Ethernet.*

1. UVOD

U današnje vreme ethernet (eng. *ethernet*), kao međunarodno standardizovana lokalna računarska mreža [1] za multimedijalne tehnologije, postaje veoma popularan za primenu i u industrijskim sistemima. Ukoliko se u okviru ethernet mreže koristi mrežni *switch*, ostvaruje se znatno poboljšanje performansi same mreže. Upotrebom *switch-a* moguće je i realizovati komunikaciju koja zadovoljava vremenska ograničenja rada u realnom vremenu. Danas se ethernet veoma lako može napraviti kao čisto ethernet *switch* mreža čime se veoma uspešno rešava i problem kolizije (eng. *collision*). Sve ovo znači da se ethernet *switch* mreže mogu smatrati kao veoma dobro rešenje za realizaciju komunikacije kod sistema koji rade u realnom vremenu, kao što su industrijski upravljačko nadzorni sistemi.

Za ethernet *switch* mreže koje se koriste u industrijskim sistemima neophodno je razviti odgovarajuće protokole za takve specijalizovane oblasti upotrebe. Poslednjih godina je predloženo nekoliko protokola koji podržavaju komunikaciju u realnom vremenu kod ethernet *switch* mreža. Više o ovim protokolima može se naći u [2], [3] i [4]. Ono što je zajedničko za predložene protokole jeste da uglavnom zahtevaju neke dodatne modifikacije *switch-a* i/ili umreženih uređaja kako bi se zadovoljila sva vremenska ograničenja rada u realnom vremenu.

Osim komunikacije koja može da zadovolji vremenska ograničenja rada u realnom vremenu, ovi industrijski sistemi često zahtevaju da komunikacija ima i mogućnost tolerisanja otkaza. Odnosno, neophodno im je da sa visokom

pouzdanošću ostvare komunikaciju. Tema ovog rada upravo je pouzdana komunikacija u realnom vremenu kod industrijskih ethernet *switch* mreža.

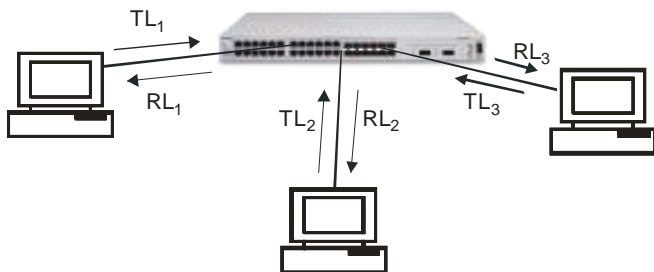
U ovom radu predložen je jedan model komunikacije u realnom vremenu kod ethernet *switch* mreža, a koja ima mogućnost da toleriše prolazne otkaze nastale na prenosnim linkovima. Predloženi model inspirisan je rešenjem koje je opasano u radu [5]. U njemu se ne predviđa, pa i ne razmatra pojava otkaza u prenosu informacije kroz mrežu. Ovde je to rešenje nadograđeno kako bi se ostvarila i mogućnost tolerisanja otkaza. U okviru ovde predloženog rešenja otkazi se tolerišu ponovnim slanjem paketa uz zadovoljenje vremenskih ograničenja rada u realnom vremenu. Zato se analizira mogućnost prevazilaženja otkaza korišćenjem određene vremenske redundanse. U radu će biti predstavljen i algoritam koji opisuje ceo proces komunikacije u realnom vremenu sa mogućnošću tolerisanja prolaznih otkaza u prenosu paketa.

2. STRUKTURA MREŽE I NAČIN KOMUNIKACIJE

U radu je razmatrana jedna ethernet *switch* mreža, prikazana na slici 1. Mreža je zvezdaste strukture i sastoji se od *switch-a* kao centralnog uređaja koji upravlja protokom podataka i krajnjih čvorova tj. umreženih računara.

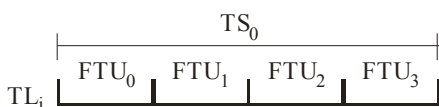
Pretpostavlja se da je *switch* standardni uređaj čiji način rada nije modifikovan, kao i da su veze između *switch-a* i umreženih računara dvosmerne tj. tipa *full-duplex* i da se sastoje od predajnog TL (eng. *transmission link*) i prijemnog linka RL (eng. *reception link*). I predajni i prijemni linkovi

rade nezavisno, dok *switch* koristi *store-and-forward* tj. sačuvaj-pa-prosledi mod za prosledivanje paketa sa ulaznih na izlazne portove. U okviru ovog moda rada *switch* određuje izlazni port i prosleđuje paket na taj port tek nakon prijema celog paketa. Takođe, polazi se i od pretpostavke da su svi umreženi uređaji sinhronizovani za prenos.



Slika 1. *Ethernet switch mreža*

Struktura linkova i predajnih i prijemnih zamišljena je na sledeći način. Neka su linkovi sastavljeni od vremenskih intervala označenih sa TS_i (eng. *time slots*). Svaki vremenski interval TS_i se dalje može podeliti na manje vremenske podintervale označene sa FTU_i (eng. *frame transmission units*). Na slici 2. prikazana je predložena struktura linka tj. raspodela vremena na predajnom linku TL_i koji spaja računar N_i i *switch*. Konkretno prikazan je vremenski interval TS_0 koji je podeljen na četiri podintervala označena sa FTU_0 do FTU_3 .



Slika 2. *Struktura linka*

U okviru podintervala FTU_i prenose se paketi. Kako je tema rada vezana za komunikaciju u realnom vremenu to znači da paketi koji se prenose kroz *ethernet switch* mrežu imaju stroge definisane rokove za prenos. U tom slučaju u okviru mreže treba obezbediti i dodeliti paketima dovoljno velike vremenske intervale kako bi se zadovoljila sva vremenska ograničenja. U radu se pretpostavlja da su ta vremena unapred poznata i da se dodeljuju paketima pre početka prenosa. Dužina podintervala FTU_i se određuje na osnovu inteziteta saobraćaja u mreži.

Sa $\{P_{ij}, D_{ij}, C_{ij}\}$ označene su vremenske karakteristike paketa koji se prenosi na linku pri čemu je sa P_{ij} označen period slanja paketa, sa D_{ij} rok koji ne sme biti prekoračen prilikom prenosa paketa i sa C_{ij} dužina paketa. Vremenske karakteristike svih paketa su unapred poznate. Indeks ij vremenskih parametara odnosi se na vremenske karakteristike paketa koji se šalje od čvora N_i ka čvoru N_j . Pretpostavlja se da je

$$P_{ij} = D_{ij} = k * FTL$$

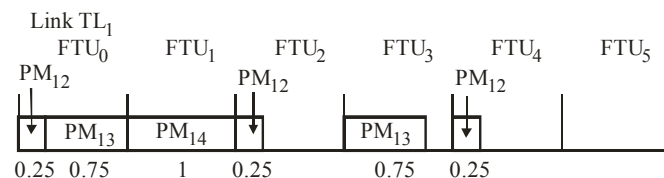
gde je FTL dužina podintervala FTU , dok je k neki ceo broj. Takođe pretpostavlja se da je broj podintervala FTU u intervalu TS jednak

$$n = NZS(P_{ij}/FTL)$$

gde je sa NZS označen najmanji zajednički sadržilac.

Mreža je zamišljena tako da svaki od krajnjih čvorova tj. računara zna strukturu svog predajnog i prijemnog linka. Takođe, računari imaju informaciju o tome kako su paketi raspoređeni na linku tj. algoritam po kome se odvija komunikacija. Može se jednostavno reći da računari imaju mogućnost da obezbede prenos paketa po tačno definisanom redosledu.

Na slici 3. prikazan je primer raspodele vremena na predajnom linku TL_1 koji spaja računar N_1 i *switch*. Konkretno prikazan je vremenski interval TS_0 koji se sastoji od šest FTU podintervala označenih sa FTU_0 do FTU_5 . Radi jednostavnijeg prikaza pretpostavlja se da je $FTL=1$. U prikazanom primeru računar N_1 šalje periodične pakete čvorovima tj. računarima N_2, N_3 i N_4 . Ovi paketi su označene sa PM_{12}, PM_{13} i PM_{14} respektivno.



Slika 3. *Redosled slanja paketa*

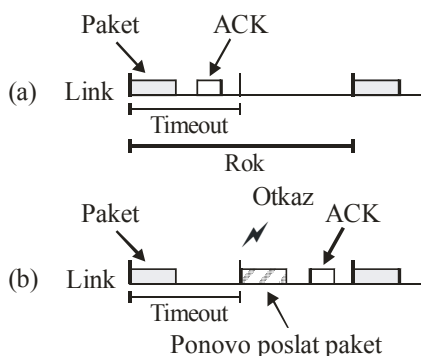
Vremenske karakteristike koje moraju da zadovolje ovi paketi su $PM_{12} \{2, 2, 0.25\}$, $PM_{13} \{3, 3, 0.75\}$ i $PM_{14} \{6, 6, 1\}$. Usvajeno je da su ovi paketi raspoređeni na linku po algoritmu predloženom u radu [5]. Sa slike 3. se jasno može videti redosled po kome se paketi šalju. Vremenski zahtevi, koji moraju biti ispunjeni tokom prenosa, svih paketa su zadovoljeni tako da je ovo primer uspešno realizovanog komunikacionog linka.

3. PREDLOŽENI ALGORITAM

Ponekad se može desiti da, usled različitih spoljašnjih uticaja na link (npr. elektromagnetnih), paketi uopšte ne stignu na odredište ili se "oštete" u toku prenosa. Kako je kod komunikacije u realnom vremenu imperativ ostvariti visoku pozdanost onda gubitak paketa ne bi smeo da se dogodi. Algoritam koji je predlažen u radu upravo predviđa mogućnost tolerancije prolaznih otkaza nastalih na prenosnim linkovima.

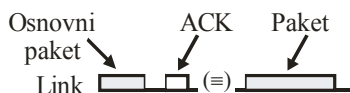
Za realizaciju predloženog algoritma iskombinona su dva osnovna mehanizma *acknowledgement* i *timeout* kako bi se dobila informaciju o pouzdanosti prenosa. Predviđeno je da ako računar u mreži pošalje paket i ako mu nakon toga stigne potvrda tj. *acknowledgement* (ACK) o prijemu paketa smatra se da je paket uspešno prenet. Ukoliko nema potvrde o prijemu paketa nakon isteka određenog vremena (*timeout-a*), znaci da prijemnik nije ni primio taj paket. U toj situaciji, po predloženom algoritmu, računar koji je poslao paket mora da ga ponovo pošalje ali da pri tome zadovolji sva vremenska ograničenja.

Ilustracija *acknowledgement* i *timeout* mehanizma prikazana je na slici 4. Slika 4.a) predstavlja slučaj kada je paket uspešno stigao do svog odredišta. Potvrda o prijemu paketa, ACK, je stigla predajniku pre isteka vremena *timeout*. Uz sve to ispoštovan je i rok koji ne sme biti prekoračen prilikom prenosa paketa. Slika 4.b) ilustruje slučaj kada paket nije uspešno stigao do svog odredišta i kada je neophodno ponovo ga poslati. Nakon prvog slanja paketa, do isteka vremena *timeout*, nije stigla potvrda o prijemu paketa, ACK, tako da je potrebno paket ponovo poslati. Ukoliko postoje uslovi tj. dovoljno slobodnog vremena na linku, odmah nakon isteka vremena *timeout* paket koji nije uspešno poslat šalje se ponovo. I u ovom slučaju nakon slanja paketa čeka se potvrda o prijemu, ACK. U primeru na slici 4.b) potvrda o prijemu paketa stiže pre isteka vremena *timeout*. Sve se završava pre roka koji ne sme biti prekoračen prilikom prenosa paketa, tako da je i ovo primer uspešno poslatog paketa.



Slika 4. Acknowledgement and timeout mehanizam

Radi jednostavnosti prikaza paketa na linku, usvojeno je da se vremenski interval potreban za slanje osnovnog paketa, čekanje na potvrdu o prijemu, ACK, kao i prijem te potvrde predstavlja kao jedan paket. Ovo je ilustrovano na slici 5.



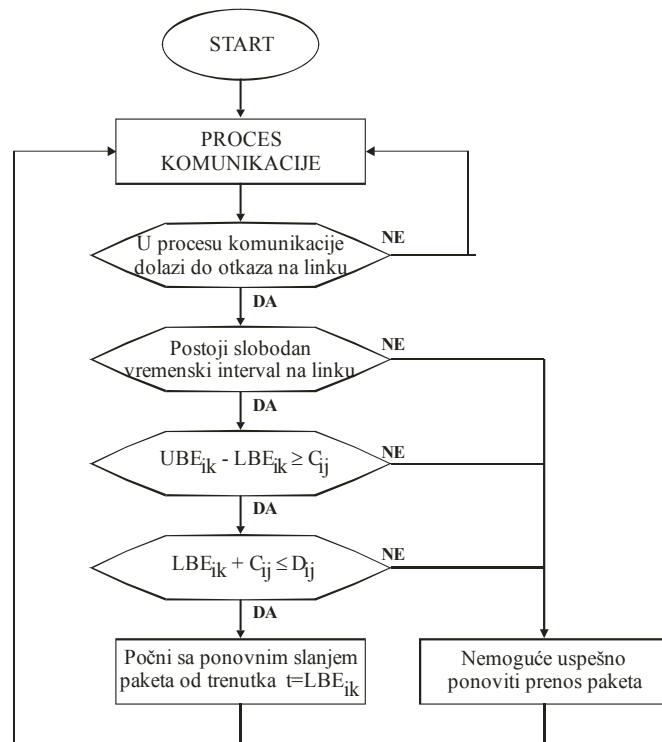
Slika 5. Osnovni paket i acknowledgment kao jedan paket

Kao što se iz jednostavnog primera sa slike 4. može videti osnovni problem prilikom ponovnog slanja paketa jeste da li ima dovoljno slobodnog vremena da paket bude uspešno ponovo poslat. Zbog toga, neophodno je čuvati informaciju o slobodnim tj. zauzetim vremenskim intervalima na linku. U predloženom algoritmu usvaja se da svaki od krajnjih čvorova tj. računara ima ovu informaciju za svoj predajni i prijemni link. Konkretno, svaki računar zna donje (LBE) i gornje (UBE) granice slobodnih intervala.

Činjenica da svi umreženi računari znaju kada su im linkovi zauzeti a kada slobodni iskorišćena je za rešavanje problema pojave prolaznih otkaza na linku. Osnovna ideja algoritma je sledeća. Ako se na određenom linku desio otkaz koji je uslovio da paket ne stigne uspešno do svog odredišta onda računar čiji je link u pitanju počinje proces ponovnog slanja paketa. U osnovi proces se sastoji od dva uslova. Prvo što treba računar da odradi to je procena da li na linku postoji dovoljno veliki slobodan vremenski interval koji je moguće

iskoristiti za ponovno slanje paketa. Sledeće što treba proveriti jeste da li će u procesu ponovnog slanja paketa biti zadovoljena sva vremenska ograničenja tj. sve vremenske karakteristike vezane za prenos u realnom vremenu. Znači ideja je da računar nađe prvi slobodan trenutak koji zadovoljava oba uslova i da ponovo pošalje paket. Osnova predloženog algoritma je činjenica da je vremenska redundansa iskorišćena za prevazilaženje otkaza.

Predloženi algoritam prikazan je na slici 6. Predstavljeni algoritam opisuju proces komunikacije u realnom vremenu kod ethernet switch mreža koja ima mogućnost tolerancije prolaznih otkaza.



Slika 6. Predloženi algoritam

Sa slike 6. se može videti da ako otkaza nema komunikacija se odvija na način objašnjen u poglavlju 2. Ako se u procesu komunikacije desi otkaz na linku počinje se sa prvom proverom, a to je da li na linku postoji slobodan vremenski interval ili ne. Ako postoji sledeće što treba proveriti jeste da li je slobodan prostor jednak ili veći od dužine paketa koji treba ponovo poslati. Odnosno, proverava se uslov

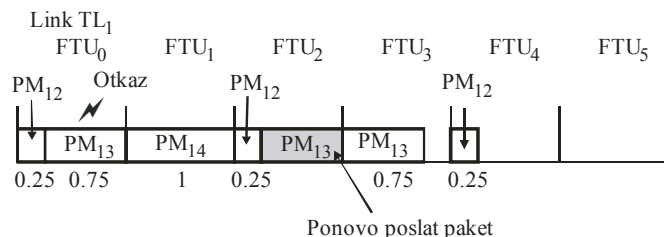
$$UBE_{ik} - LBE_{ik} \geq C_{ij}$$

Indeks *ik* kod oznake gornje UBE i donje LBE granice slobodnih intervala u opštem slučaju označava da je u pitanju link čvorova tj. računara N_i koji sadrži *k* slobodnih vremenskih intervala. Ukoliko je prethodni uslov ispunjen sledeće što treba proveriti vezano je za vremenske karakteristike paketa. Tačnije treba izračunati da li je donja granica tog slobodnog intervala plus dužina paketa koju treba ponovo preneti manja ili jednaka roku koji ne sme biti prekoračen prilikom prenosa paketa. U ovom slučaju proverava se uslov

$$LBE_{ik} + C_{ij} \leq D_{ij}$$

Ako je i ovaj uslov ispunjen moguće je ponovo poslati paket i pri tome zadovoljiti sva vremenska ograničenja. Trenutak kada će se započeti ponovno slanje paketa jednak je vremenu donje ivice slobodnog intervala. Nakon ponovnog slanja paketa nastavlja se proces komunikacije na način objašnjen u poglavlju 2.

Ako bilo koji od uslova nije ispunjen onda se proces ponovnog slanja paketa ne može izvršiti. U tom slučaju računar ima informacije o nemogućnosti tolerancije konkretnog otkaza tj. neuspešno prenetom paketu. Za komunikaciju koja radi u realnom vremenu ovaj slučaj je jako nepovoljan i praktično ne bi smeo da se dogodi.



Slika 7. Redosled po kome se paketi prenose u slučaju nastanka prolaznog otkaza

Na slici 7. ilustrovan je realizovani algoritam kroz jedan jednostavan primer prenosa paketa po linku. Konkretno iskorišćen je primer predajnog linka TL_1 prikazan na slici 3.

Kao što se sa slike 7. može videti pretpostavlja se da se u toku trajanja podintervala FTU_0 desio otkaz na linku. Odnosno, pretpostavka je da se otkaz desio u toku prenosa paketa PM_{13} i da potvrda o prijemu paketa ACK nije stigla pre isteka vremena *tajmaut*. U tom slučaju računar N_1 počinje proces ponovnog slanja paketa PM_{13} . Po predloženom algoritmu prvi slobodni odgovarajući vremenski interval je u okviru FTU_2 podintervala tako da će se on i iskoristiti za ponovljeni prenos paketa PM_{13} . Svi vremenski zahtevi, koji moraju biti ispunjeni tokom prenosa, paketa PM_{13} su zadovoljeni. Ovaj primer predstavlja slučaj uspešnog oporavka mreže nakon pojave otkaza.

4. ZAKLJUČAK

Ideja korišćenja ethernet switch mreža za realizaciju komunikacije u realnom vremenu predstavljena je u ovom

radu. Posebna pažnja posvećena je mogućnosti tolerisanja otkaza u procesu komunikacije.

U radu je predložen jedan način realizacije komunikacije koja ima i mogućnost tolerisanja prolaznih otkaza nastalih na prenosnim linkovima. Pokazano je da se tolerisanje otkaza može ostvariti korišćenjem vremenske redundanse. Vremenska redundansa iskorišćena je za ponovljeno slanje paketa koji je zbog pojave otkaza neuspešno prenet. Posebna pažnja posvećena je činjenici da u procesu ponovnog slanja paketa moraju biti zadovoljena sva vremenska ograničenja vezana za komunikaciju u realnom vremenu.

U radu je predstavljen i algoritam koji opisuje deo procesa komunikacije koji se odnosi na mogućnost tolerisanja otkaza. Realizovani algoritam daje mogućnost da se sa visokom pouzdanošću ostvari komunikacija u realnom vremenu kod ethernet switch mreža.

LITERATURA

- [1] L. R. Peterson, B. S. Davie, *COMPUTER NETWORKS A Systems approach*, third edition, Morgan Kaufmann Publishers, 2003
- [2] H. Hoang, M. Jonsson, U. Hagström, A. Kallerdahl, "Switched Real-Time Ethernet with Earliest Deadline First Scheduling - Protocols and Traffic Handling", *Proceedings of the International Parallel and Distributed Processing Symposium*, vol.2, pp0094b, IPDPS 2002 Workshops, 2002
- [3] J. Loeser, H. Haertig, "Low-latency Hard Real-Time Communication over Switched Ethernet", *Proceedings of the 16th Euromicro Conference on Real-Time Systems, ECRTS04*, Catania, Italy, June 2004
- [4] S. Varadarajan, T. Chiueh, "EtheReal: A Host-transparent Real-Time Fast Ethernet Switch", *Proceedings of the Sixth International Conference on Network Protocols*, October 1998
- [5] H. C. Lee, M. K. Kim, "A scheduling Algorithm of Periodic Messages for Hard Real-Time Communications on a Switched Ethernet", *International Journal of Computer Science and Network Security*, IJCSNS, vol.6, No.5B, May 2006