UNIVERZITET U BANJOJ LUCI
ELEKTROTEHNIČKI FAKULTET

Mladen Knežić

Optimizacija vremena odziva industrijskih komunikacionih mreža

DOKTORSKA DISERTACIJA

Banja Luka, 2017.

UNIVERSITY OF BANJA LUKA

FACULTY OF ELECTRICAL ENGINEERING

Mladen Knežić

Response time optimization of industrial communication networks

DOCTORAL DISSERTATION

Banja Luka, 2017.

Mentor:

Dr Branko Dokić, redovni profesor

Univerzitet u Banjoj Luci, Elektrotehnički fakultet

Naslov doktorske disertacije:

Optimizacija vremena odziva industrijskih komunikacionih mreža

Rezime:

Moderni sistemi industrijske automatizacije su orijentisani ka konvergenciji različitih tehnologija u cilju ostvarenja koncepta inteligentne fabrike u okviru četvrte industrijske revolucije, koja podrazumijeva povezivanje svih nivoa proizvodnje u jedinstvenu virtuelnu mrežu u sklopu paradigme Internet stvari. Industrijski komunikacioni sistemi zasnovani na Ethernet tehnologiji koji podržavaju rad u realnom vremenu (RTE), čine osnovnu kariku u lancu koji omogućava realizaciju prethodno pomenutog koncepta. Vrijeme odziva predstavlja objektivnu mjeru mogućnosti određenog RTE sistema. Predmet istraživanja ove disertacije obuhvata teorijsku i eksperimentalnu analizu RTE sistema iz kategorije koja omogućava najviše performanse i visok nivo sinhronizacije svih komponenata sistema. U radu su predloženi i eksperimentalno verifikovani novi analitički modeli, kao i algoritmi i metode za optimizaciju vremena odziva RTE sistema.

Ključne riječi:

Analiza performansi, EtherCAT, Ethernet u realnom vremenu, Ethernet Powerlink, indikatori performansi, Profinet, vrijeme odziva

Naučna oblast:

Elektronika i elektronski sistemi

Naučno polje:

Elektrotehnika

Klasifikaciona oznaka:

T 170

Tip odabrane licence Kreativne zajednice:

Autorstvo – dijeliti pod istim uslovima (CC BY-SA)

Supervisor:

Dr Branko Dokić, full professor

University of Banja Luka, Faculty of Electrical Engineering

Title of doctoral dissertation:

Response time optimization of industrial communication networks

Abstract:

Modern industrial automation systems are moving towards convergence of different technologies and aims at achieving the concept of Smart Factory as a part of Industrie 4.0 initiative, which means connecting all factory levels into a single virtual network conforming to the Inernet of Things paradigm. Ethernet-based industrial communication systems, commonly referred to as Real-Time Ethernet (RTE), represent the main link in the chain that allows realization of the aforementioned concept. Response time is an objective measure of the capabilities of an RTE system. The subject of this dissertation includes a theoretical and experimental analysis of the RTE systems from the highest-performance category with the high level of synchronization between all components of the system. New analytical models, as well as algorithms and methods for response time optimization of the RTE systems are proposed and verified experimentally in this study.

Keywords:

Performance analysis, EtherCAT, Real-Time Ethernet, Ethernet Powerlink, performance indicators, Profinet, response time

Scientific area:

Electronics and electronic systems

Scientific field:

Electrical engineering

Classification scheme:

T 170

Creative Commons license type:

Attribution-ShareAlike (CC BY-SA)

Sadržaj

1 Uvod **1**

1.1 Pregled istraživanja 2

1.2 Motivacija i predmet istraživanja 4

1.3 Kratak sadržaj i pregled rada 5

2 Industrijski komunikacioni sistemi **7**

2.1 Osnove EtherCAT protokola 8

2.1.1 Načini adresiranja 10

2.1.2 Komande 12

2.1.3 ECAT mašina stanja 14

2.1.4 Konfiguracija ECAT mreže 15

2.1.5 Struktura ENI fajla 16

2.2 Osnove Ethernet Powerlink protokola 22

2.2.1 Osnovne topologije EPL mreže 26

2.2.2 Konfiguracija EPL mreže 27

2.2.3 Multipleksirani način rada 28

2.2.4 Način rada sa ulančavanjem PRes okvira 29

2.3 Osnove Profinet IO protokola 30

2.3.1 Vrste uređaja u PNIO mreži 32

2.3.2 Model PNIO uređaja 34

2.3.3 Adresiranje PNIO uređaja 35

2.3.4 PNIO komunikacioni servisi 36

2.3.5 Organizacija PNIO komunikacionog ciklusa 38

2.4 Zaključak 42

3 Metrike u industrijskim komunikacionim sistemima **44**
3.1 Metrike definisane standardom IEC 61784-2 44
3.1.1 Vrijeme isporuke 45
3.1.2 Propusna moć 46
3.1.3 Propusni opseg rezervisan za NRT saobraćaj 47
3.1.4 Preciznost vremenske sinhronizacije 49
3.1.5 Preciznost sinhronizacije koja nije bazirana na globalnom vremenu . . . 50
3.1.6 Vrijeme oporavka od greške 50
3.1.7 Broj krajnjih stanica 51
3.1.8 Osnovna mrežna topologija 52
3.1.9 Broj mrežnih komutatora između krajnjih stanica 53
3.2 Specifično definisane metrike 53
3.2.1 Minimalno vrijeme komunikacionog ciklusa 54
3.2.2 Vrijeme odziva 55
3.3 Zaključak 59

4 Modelovanje industrijskih komunikacionih sistema **61**
4.1 Analitički model ECAT protokola 62
4.1.1 Linijska topologija 63
4.1.2 Topologija zvijezde 67
4.2 Karakterizacija vremenskih parametara ECAT analitičkog modela 69
4.3 Teorijska analiza ECAT protokola 78
4.4 Analitički model EPL protokola 83
4.4.1 Standardni način rada 85
4.4.2 PRC način rada 87
4.5 Karakterizacija vremenskih parametara EPL analitičkog modela 91
4.6 Teorijska analiza EPL protokola 98
4.7 Analitički model PNIO protokola 103
4.8 Karakterizacija vremenskih parametara PNIO analitičkog modela 107
4.9 Teorijska analiza PNIO protokola 110
4.10 Uporedna analiza industrijskih komunikacionih sistema 113
4.11 Zaključak 117

5 Mehanizmi optimizacije vremena odziva **119**

5.1 Optimizacija ECAT protokola 120

5.1.1 Optimizacija propagacionog kašnjenja ECAT mreže 121

5.1.2 Optimizacija dužine ECAT okvira 124

5.2 Optimizacija EPL protokola 137

5.2.1 Ulančavanje PRes okvira kontrolisano događajima 138

5.2.2 Eksperimentalna verifikacija 142

5.2.3 Simulaciona analiza 143

5.3 Optimizacija PNIO protokola 147

5.3.1 Dinamičko pakovanje okvira 148

5.4 Zaključak 156

6 Zaključak **157**

Slike

2.1 Struktura i način enkapsulacije ECAT okvira 9
2.2 Struktura ESC jedinice 10
2.3 Struktura zaglavlja ECAT telegrama 11
2.4 Ilustracija translacije logičke u fizičku adresu 12
2.5 ECAT mašina stanja 14
2.6 Konfiguracija ECAT mreže 16
2.7 Primjer mapiranja telegrama u procesnu sliku ECAT sistema 20
2.8 Struktura okvira EPL protokola 23
2.9 EPL komunikacioni ciklus 24
2.10 Primjer topologije EPL mreže 27
2.11 Konfiguracija EPL mreže 28
2.12 EPL komunikacioni ciklus sa multipleksiranim načinom rada 28
2.13 EPL komunikacioni ciklus sa ulačavanjem PRes okvira 29
2.14 Vrste PNIO uređaja i konfiguracija mreže 33
2.15 Struktura modularnog PNIO uređaja 34
2.16 Sekvenca DCP poruka za dodjelu simboličkog imena PNIO uređaju 35
2.17 Sekvenca DCP poruka za dodjelu IP adrese PNIO uređaju 36
2.18 Ilustracija koncepta virtuelnih PNIO kanala (AR i CR) 37
2.19 Sekvenca RPC poruka za uspostavljanje CR kanala i parametrizaciju uređaja . . 38
2.20 Ilustracija slanja periodičnih PNIO okvira 39
2.21 PNIO komunikacioni ciklus 39
2.22 Struktura PNIO RT okvira 40
2.23 Primjer mapiranja procesnih podataka u PNIO okvir 42

2.24 Struktura PNIO IRT okvira 42

3.1 Komponente komunikacionog sistema koje utiču na vrijeme isporuke 46
3.2 Osnovne mrežne topologije: (a) hijerarhijski organizovana topologija zvijezde, (b) topologija prstena i (c) linijska topologija 52
3.3 Primjer određivanja vremena odziva u industrijskoj mreži (bazirano na [64]) . . 56
3.4 Vrijeme odziva u potpuno sinhronizovanom (izohronom) RTE sistemu 58

4.1 Struktura standardnog ECAT uređaja sa dva MII porta 64
4.2 Prostorno-vremenski dijagram ECAT komunikacije 65
4.3 Struktura ECAT modula sa dva E-bus porta 65
4.4 Struktura ECAT sprežnika magistrale 66
4.5 Struktura modularnog ECAT komutatora sa 8 portova 67
4.6 Eksperimentalna postavka ECAT mreže 70
4.7 Izmjereni period komunikacionog ciklusa za (a) *TwinCAT* i (b) CIFX implementaciju nadređenog uređaja 71
4.8 Kašnjenje *Beckhoff* uređaja u (a) direktnom i (b) povratnom smjeru 73
4.9 Kašnjenje *Softing* uređaja u (a) direktnom i (b) povratnom smjeru 74
4.10 Kašnjenje *Hilscher* uređaja u (a) direktnom i (b) povratnom smjeru 74
4.11 Eksperimentalna postavka ECAT mreže sa modularnim uređajem 75
4.12 Kašnjenje modularne strukture EK1100-EK1122 u (a) direktnom i (b) povratnom smjeru 75
4.13 Kašnjenje modularne strukture EK1100-EL2002-EK1122 u (a) direktnom i (b) povratnom smjeru 77
4.14 Kašnjenje modularne strukture EK1100-EL2002-EL1012-EK1122 u (a) direktnom i (b) povratnom smjeru 77
4.15 $T_{C(min)}$ u funkciji broja uređaja za različite topologije: (a) *Beckhoff* i (b) *Hilscher* implementacija 79
4.16 Odnos T_{frames}/T_{net} u funkciji broja uređaja za (a) *Beckhoff* i (b) *Hilscher* implementaciju (ukupna veličina procesnih podataka 8 bajtova) 80
4.17 Odnos T_{frames}/T_{net} u funkciji broja uređaja za (a) *Beckhoff* i (b) *Hilscher* implementaciju (ukupna veličina procesnih podataka 64 bajta) 81
4.18 T_C i $P_{RTE(max)}$ u funkciji broja uređaja za (a) *Beckhoff* i (b) *Hilscher* implementaciju (linijska topologija) 82
4.19 Prostorno-vremenski dijagram izohrone faze EPL mreže u standardnom režimu 86

4.20 Prostorno-vremenski dijagram izohrone faze EPL mreže u PRC režimu 89
4.21 Ilustracija mjerenja propagacionog kašnjenja (*SyncDelay*) 90
4.22 Eksperimentalna postavka EPL mreže: (a) linijska i (b) topologija zvijezde . . . 92
4.23 Izmjereni period SoC okvira za (a) PC i (b) FPGA implementaciju MN čvora . 94
4.24 Izmjerene vrijednosti *PResTimeout* parametra (linijska topologija): (a) $T_{TO}^{CN\#1}$ i (b) $T_{TO}^{CN\#2}$ za PC, i (c) $T_{TO}^{CN\#1}$ za FPGA implementaciju MN čvora 95
4.25 Izmjereno kašnjenje za (a) OPLK, (b) *Softing*, (c) *Hilscher* i (d) *3Com* koncentrator 96
4.26 Izmjereno vrijeme odziva za (a) OPLK, (b) *Softing* i (c) *Hilscher* CN 98
4.27 Trajanje izohrone faze u funkciji broja uređaja za: (a) linijsku i (b) topologiju zvijezde (veličina procesnih podataka 4 bajta) 100
4.28 Trajanje izohrone faze u funkciji veličine procesnih podataka za: (a) linijsku i (b) topologiju zvijezde (broj čvorova 10) 101
4.29 Odnos $T_{ip}^{PRC}/T_{ip}^{Std}$ za dvije topologije u funkciji: (a) broja čvorova i (b) veličine procesnih podataka 102
4.30 T_C i $P_{RTE(max)}$ u funkciji broja čvorova za linijsku topologiju i (a) standardni i (b) PRC način rada 103
4.31 Ilustracija *downlink* slanja okvira kada se koristi *slipstreaming* mehanizam . . . 105
4.32 Ilustracija *uplink* slanja okvira kada se koristi *slipstreaming* mehanizam 105
4.33 Eksperimentalna postavka PNIO mreže 107
4.34 Izmjereni period komunikacionog ciklusa IOC uređaja 108
4.35 Izmjereni period komunikacionog ciklusa IOD uređaja: (a) IM151-3 PN i (b) *Softing* implementacija 109
4.36 Kašnjenje komutatora integrisanog u (a) IM151-3 PN i (b) *Softing* IOD uređaj . 110
4.37 Trajanje IRT faze u funkciji broja uređaja za dvije topologije 111
4.38 Trajanje IRT faze u funkciji veličine procesnih podataka za dvije topologije . . 112
4.39 T_C i $P_{RTE(max)}$ u funkciji broja uređaja za (a) linijsku i (b) topologiju zvijezde . 113
4.40 Tipična struktura mreže u sistemima automatizacije 114

5.1 Topologija ECAT mreže sa mrežnim komutatorom i linijskim ograncima 122
5.2 Zavisnost odnosa $T_{C(min)}$ linijske i poboljšane topologije od broja uređaja 123
5.3 Zavisnost $T_{C(min)}$ od broja portova i broja uređaja 123
5.4 Ilustracija određivanja lokalnog minimuma za slučaj sa 144 uređaja 124
5.5 Proces optimizacije veličine okvira prilikom konfiguracije 126
5.6 Primjer mrežne topologije 128

5.7 Mrežna topologija nakon optimizacije 129
5.8 Mrežna postavka korišćena pri testiranju 132
5.9 Mrežna topologija dobijena nakon optimizacije za prvi scenario 133
5.10 Mrežna topologija dobijena nakon optimizacije za drugi scenario 134
5.11 Prikaz komunikacije u okviru *Wireshark* softvera sa označenim ulaznim podacima za prvi scenario 135
5.12 Prikaz komunikacije u okviru *Wireshark* softvera sa označenim ulaznim podacima za drugi scenario 135
5.13 Veličina okvira (u bajtovima) dobijena različitim konfiguracionim alatima . . . 136
5.14 Ilustracija izbora sljedećeg uređaja za (a) $Diff < 0$ i (b) $Diff > 0$ 136
5.15 Primjer EPL komunikacije u (a) TT-PRC i (b) ET-PRC režimu 140
5.16 Primjer EPL komunikacije u multipleksiranom ET-PRC režimu 141
5.17 Izmjerena dužina trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci za linijsku topologiju sa dva upravljana čvora u (a) standardnom, (b) TT-PRC i (c) ET-PRC režimu 144
5.18 Izmjerena dužina trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci za topologiju zvijezde sa dva upravljana čvora u (a) standardnom, (b) TT-PRC i (c) ET-PRC režimu 145
5.19 Složenija EPL mrežna konfiguracija 147
5.20 Struktura DFP podokvira 148
5.21 Ilustracija koncepta dinamičkog pakovanja okvira pri *downlink* komunikaciji za (a) $T_F < T_{net}$ i (b) $T_F > T_{net}$ 150
5.22 Ilustracija koncepta dinamičkog pakovanja okvira pri *uplink* komunikaciji za (a) $T_F < T_{net}$ i (b) $T_F > T_{net}$ 151
5.23 Trajanje IRT faze u funkciji broja uređaja za: (a) linijsku i (b) topologiju zvijezde (veličina procesnih podataka 4 bajta) 153
5.24 Trajanje IRT faze u funkciji veličine procesnih podataka za: (a) linijsku i (b) topologiju zvijezde (broj uređaja 10) 154
5.25 Odnos $T_{irt}^{DFP}/T_{irt}^{PNIO}$ za dvije topologije u funkciji: (a) broja uređaja i (b) veličine procesnih podataka 155

Tabele

2.1 Pregled komandi definisanih ECAT specifikacijom 13
2.2 Pregled tipova EPL poruka 23
2.3 Pregled identifikatora čvorova 24
2.4 Pregled identifikatora PNIO okvira 41

4.1 Pregled ECAT telegrama koji se koriste u stvarnim instalacijama 64
4.2 Pregled statistike mjerenih vremenskih parametara 72
4.3 Pregled statistike mjerenih vremenskih parametara 76
4.4 Procjena vremenskih parametara terminala 76
4.5 Pregled statistike mjerenih vremenskih parametara 94
4.6 Pregled statistike mjerenih vremenskih parametara 109
4.7 Informacije o korišćenim uređajima 114
4.8 Organizacija cikličnog ECAT telegrama za razmatrani sistem 115
4.9 Uporedna analiza industrijskih komunikacionih sistema 116

5.1 Informacije o korišćenim uređajima 128
5.2 Poređenje rezultata optimizacije 129
5.3 Informacije o korišćenim uređajima 131
5.4 Pregled statistike za dužinu trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci (T_{poll} i T_{prc}) 143
5.5 Rezultati simulacija 146

Algoritmi

5.1 Algoritam za optimizaciju dužine okvira 125
5.2 Modifikovani algoritam za optimizaciju veličine okvira 130

1 Uvod

Moderni sistemi industrijske automatizacije su sve više orijentisani ka konvergenciji različitih tehnologija u cilju ostvarenja koncepta pod nazivom *inteligentna fabrika* (engl. *Smart Factory*) u okviru četvrte industrijske revolucije (engl. *Industrie 4.0*) [1], koja omogućava povezivanje svih nivoa proizvodnje u jedinstvenu virtuelnu mrežu u sklopu paradigme *Internet stvari* (engl. *Internet of Things* – IoT) [2, 3]. Najnovije prognoze pokazuju da će čak i rigidni industrijski sistemi, u skorijoj budućnosti, biti bazirani na servisno orijentisanoj paradigmi koja podrazumijeva koncept proizvodnje kao usluge. To znači da će budući proizvodni sistemi morati da budu rekonfigurabilni i adaptivni u smislu promjene procesa proizvodnje, kako bi se na brz i efikasan način prilagodili zahtijevima klijenata [4]. U tu svrhu, značajan napor se ulaže u omogućavanje koncepta automatske konfiguracije sistema prilikom dodavanja novih uređaja po tzv. *Plug and Produce* principu, inspirisanom sličnim konceptom (*Plug and Play*) koji se već dugo vremena primjenjuje u računarskim sistemima [5–7].

Industrijski komunikacioni sistemi zasnovani na *Ethernet* tehnologiji koji podržavaju rad u realnom vremenu, RTE (engl. *Real-Time Ethernet*) sistemi, čine osnovnu kariku u lancu koji omogućava realizaciju prethodno opisanih koncepata. Osnovne prednosti koje ovi sistemi imaju u odnosu na tradicionalne industrijske komunikacione sisteme, poznate pod nazivom *fieldbus*, su znatno širi propusni opseg, koji se sve više zahtijeva u nadolazećim aplikacijama, kao i inherentna podrška vertikalnoj integraciji industrijskih komunikacionih mreža na nivou procesa proizvodnje (engl. *field level*) sa klasičnim računarskim mrežama na nivou preduzeća (engl. *enterprise level*), kao što je opisano u [8]. Posljednja karakteristika omogućava povezivanje sektora proizvodnje sa sektorom uprave preduzeća, čime se povećava efikasnost poslovanja i maksimizuje iskorišćenje postojećih resursa [9]. Osim toga, ove mreže, u većini slučajeva, omogućavaju smanjenje troškova koji se odnose na kabliranje, dijagnosticiranje grešaka i održavanje sistema.

Osnovni nedostatak standardne *Ethernet* tehnologije ogleda se u tome što algoritam za arbitriranje pristupa zajedničkom medijumu nije deterministički pa stoga nije pogodan za sisteme

koji zahtijevaju rad u realnom vremenu [10]. Trenutno postoji veliki broj različitih RTE pristupa koji omogućavaju prilagođavanje *Ethernet* tehnologije na način da se omogući komunikacija u realnom vremenu uz očuvanje određenog nivoa kompatibilnosti sa standardnim *Ethernet* protokolom. Većina ovih pristupa je dio međunarodnih standarda [8, 11, 12]. Generalno, rješenja koja koriste standardne *Ethernet* komponente imaju lošije performanse u odnosu na rješenja koja uvode određene modifikacije. Odabir ispravne tehnologije, u opštem slučaju, zavisi od konkretne primjene. Nijedan od trenutno dostupnih RTE pristupa ne pokriva sve moguće scenarije uz istovremeno zadovoljenje temporalnih ograničenja koje nameće aplikacija [13]. Stoga se očekuje da budući komunikacioni sistemi u distribuiranim sistemima automatizacije budu izrazito heterogeni, kako bi mogli da se nose sa kompleksnošću ovih sistema [14].

U [15, 16], RTE protokoli su klasifikovani u tri različite kategorije u zavisnosti od nivoa performansi koje protokol može da obezbijedi. Protokoli iz treće kategorije (npr. *Profinet IO*, *Ethernet Powerlink* i *EtherCAT*) obezbjeđuju najviši nivo performansi koje zahtijevaju aplikacije za kontrolu kretanja (engl. *motion control*). Međutim, da bi postigli zahtjevani nivo performansi, navedena rješenja uvode modifikacije na sloju veze podataka *Ethernet* protokola, koje omogućavaju mehanizme za poboljšanje determinizma u komunikacionom sistemu. To neminovno dovodi do smanjenja kompatibilnosti sa standardnim komponentama, što, u određenoj mjeri, može da utiče na cijenu, ali i na mogućnost integracije komunikacionog rješenja sa klasičnim računarskim mrežama.

1.1 Pregled istraživanja

U posljednjih nekoliko godina, objavljen je niz studija u kojima su analizirane mogućnosti različitih industrijskih komunikacionih sistema i predložene modifikacije u cilju poboljšanja njihovog rada u kontekstu primjene u distribuiranim sistemima automatizacije. U većini radova, analiza je bazirana na analitičkim modelima koji su izvedeni za različite scenarije preuzete iz stvarnih sistema. Tako su, na primjer, u [17–19] autori teorijski analizirali poboljšanje performansi različitih RTE sistema pri povećanju bitske brzine sa 100 Mb/s na 1 Gb/s. Drugi autori su uveli dodatna ograničenja prilikom teorijske analize kako bi bolje analizirali efekte stvarnih aplikacija [20–23]. U [24–27], opisani su pristupi za primjenu različitih tehnika simulacije i formalne verifikacije industrijskih komunikacionih sistema. Takođe, da bi se karakterizovali vremenski parametri razvijenih analitičkih i simulacionih modela, sproveden je niz studija koje su za krajnji cilj imale eksperimentalnu procjenu karakteristika komponenata industrijskih komunikacionih sistema [28–33].

Osim teorijske analize, u literaturi su predloženi različiti pristupi za poboljšanje postojećih komunikacionih mehanizama navedenih RTE sistema. Tako je, na primjer, u [30] i [34] analizirana interna arhitektura *Profinet IO* kontrolera s ciljem poboljšanja vremenskih karakteristika sistema. U radu [35], predložen je DFP (engl. *Dynamic Frame Packing*) mehanizam za poboljšanje performansi *Profinet IO* protokola. Drugi pristupi (npr. [36–39]) definišu različite strategije raspoređivanja slanja poruka kako bi se optimizovale performanse mreže kada se koriste topologije kojima se obično realizuju stvarni sistemimi automatizacije. Takođe, u nekim studijama, analiziran je sličan problem, ali ako se primjeni prethodno pomenuti DFP mehanizam [40, 41]. S obzirom da, kod *Profinet IO* sistema, plan slanja poruka treba ponovo da se generiše pri svakoj promjeni strukture mreže, autori su u [42] predložili modifikaciju DFP mehanizma, koju su nazvali APM (engl. *Automatic Packing Mechanism*). APM mehanizam ne zahtijeva vremensko raspoređivanje poruka, čime se unapređuje fleksibilnost sistema uz istovremeno poboljšanje performansi. Međutim, navedeni pristupi su realizovani i testirani samo na nivou prototipa i još uvijek nisu dostupni u komercijalnoj izvedbi tako da nije moguće predvidjeti ponašanje predloženih rješenja u stvarnim instalacijama.

U slučaju *EtherCAT* pristupa, jedan dio istraživanja je usmjeren na optimizaciju dužine okvira, koja se obavlja prilikom konfiguracije i kao ulazni parametar koristi topologiju mreže [43]. Prema tome, struktura mreže, kao i sva fizička i prostorna ograničenja, moraju unaprijed da budu poznata kako bi se postigla optimalna konfiguracija čvorova. Druga istraživanja (npr. [44, 45]), orijentisana su ka različitim strategijama baziranim na hardverski podržanim arhitekturama nadređenog uređaja kako bi se obezbijedilo veoma kratko vrijeme komunikacionog ciklusa, ispod 100 μs. S obzirom da na efikasnost *EtherCAT* komunikacije značajno utiče odabrana mrežna topologija [46], autori su u [47] predložili rješenje bazirano na adaptaciji topologije s ciljem smanjenja ukupnog propagacionog kašnjenja mreže. Takođe, studije [48–50] obezbjeđuju različite mehanizme za poboljšanje efikasnosti razmjene asinhronih poruka kako bi se omogućila podrška za komunikaciju u realnom vremenu u slučaju aplikacija koje obrađuju neperiodične i sporadične događaje kao što su vremenski kritični alarmi.

Što se *Ethernet Powerlink* protokola tiče, autori različitih studija bavili su se analizom performansi ovog pristupa u industrijskim okruženjima. Tako, na primjer, u [51] autori su sproveli teorijsku analizu i studiju baziranu na simulacijama za različite mrežne konfiguracije kako bi istražili mogućnosti i ograničenja *Ethernet Powerlink* protokola prilikom razmjene cikličnih i acikličnih podataka. Takođe, u istom radu, autori su predložili modifikaciju protokola kako bi poboljšali njegovo ponašanje prilikom opsluživanja acikličnih podataka u realnom vremenu. U radu [52], data je opširna analiza različitih metrika definisanih u okviru IEC

61784-2 standarda [53] za scenario koji podrazumijeva koordinisanu kontrolu pokreta. Oba prethodno navedena rada analiziraju ponašanje sistema u idealnim radnim uslovima, kao i u prisustvu grešaka nastalih pri prenosu poruka. Mogućnosti proširenja *Ethernet Powerlink* protokola na primjenu u bežičnim mrežama baziranim na IEEE 802.11 standardu, analizirane su u radovima [54] i [55]. Analiza ponašanja u bežičnim primjenama je proširena eksperimentalnim rezultatima u radovima [55] i [56], pri čemu su mjerenja obavljena na prototipu stvarne bežične mreže koja je realizovana korišćenjem komercijalno dostupnih komponenata.

Konačno, jedan dio istraživanja (npr. [57–59]), orijentisan je ka istraživanju mogućnosti poboljšanja energetske efikasnosti mrežnih komponenata kada se one nalaze u neaktivnom stanju, u skladu sa IEEE 802.3az standardom [60].

1.2 Motivacija i predmet istraživanja

Analiza performansi bilo kojeg RTE sistema podrazumijeva definisanje različitih metrika koje omogućavaju kvantitativan opis mogućnosti različitih rješenja. Neke od ovih metrika su propisane odgovarajućim standardima, kao što je IEC-61784-2, dok su druge specifično definisane kako bi opisale ciklično ponašanje sistema [61, 62].

Najčešće korišćena metrika za opis cikličnog ponašanja RTE sistema je minimalno vrijeme komunikacionog ciklusa. Međutim, ova metrika ne uzima u obzir druge aspekte koji se obično zahtijevaju u distribuiranim sistemima automatizacije poput sinhronizacije, propusnog opsega rezervisanog za NRT saobraćaj, ili vremena potrebnog za obradu podataka u kontroleru i/ili distribuiranim uređajima. Iz navedenog razloga, definišu su metrike koje bolje opisuju karakteristike sistema, ili se pomenuta metrika koristi u kombinaciji sa drugim metrikama propisanim standardom IEC-61784-2.

Vrijeme odziva predstavlja objektivnu mjeru mogućnosti sistema po pitanju brzine njegovog reagovanja na promjene. Ova metrika objedinjuje više aspekata koji su potencijalno zanemareni kod minimalnog vremena komunikacionog ciklusa. Evaluacija industrijskih komunikacionih sistema u značajnom broju studija koristi vrijeme odziva kao metriku za kvantitatvinu procjenu mogućnosti sistema. U radovima [24, 28, 63], autori su analizirali vrijeme odziva *Profinet IO* sistema u slučaju nesinhronizovane klase saobraćaja, dok su u [64–66] prezentovani teorijski i eksperimentalni rezultati za slučaj generičkog RTE sistema. Studija [67] razmatra uticaj sinhronizacije uređaja na performanse sistema, u smislu minimalnog odstupanja vremena odziva.

Predmet istraživanja ove disertacije podrazumijeva teorijsku i eksperimentalnu analizu RTE sistema iz kategorije koja omogućava najviše performanse i visok nivo sinhronizacije svih

komponenata sistema. Kao osnovna metrika za kvantitativnu procjenu performansi sistema, korišćeno je vrijeme odziva. Ova metrika se obično koristi u kombinaciji sa ostalim relevantnim metrikama koje, na objektivan način, opisuju mogućnosti industrijskih komunikacionih sistema. Sprovedena analiza i eksperimentalna evaluacija omogućavaju uvođenje odgovarajućih mehanizama čiji je osnovni cilj optimizacija vremena odziva. Ovi mehanizmi podrazumijevaju promjenu topološke strukture mreže, optimizaciju vremenski kritične komunikacije protokola i precizno planiranje slanja poruka kako bi se izbjegla vremena čekanja zbog zauzeća resursa u mrežnim komponentama.

1.3 Kratak sadržaj i pregled rada

Rad se sastoji iz šest poglavlja i spiska korišćene literature. Nakon uvodnog razmatranja, u kojem je uvedena motivacija i dat pregled aktuelne literature koja tretira definisani predmet istraživanja, u drugom poglavlju su opisana tri, u praksi najčešće korišćena, pristupa iz treće kategorije RTE sistema (*EtherCAT*, *Ethernet Powerlink* i *Profinet IO*). Detaljno su opisani mehanizmi komunikacije za svaki od pristupa, kao i ograničenja koja su posljedica navedenih mehanizama.

U trećem poglavlju, razmatraju se metrike za kvantitativnu i kvalitativnu ocjenu mogućnosti industrijskih komunikacionih sistema. Definisane su dvije osnovne grupe metrika: metrike opisane standardom IEC 61784-2 i specifično definisane metrike koje detaljnije opisuju ciklično ponašanje pomenutih sistema. Posebna pažnja je posvećena analizi međusobnog uticaja metrika, kao i specifičnostima koje se odnose na odabrane industrijske komunikacione sisteme prilikom njihove procjene.

U četvrtom poglavlju su izvedeni analitički modeli za pomenuta tri pristupa s ciljem analize uticaja veličine i strukture mreže na prethodno definisane metrike. Izvedeni su modeli za linijsku i topologiju zvijezde, pri čemu je posebna pažnja posvećena parametrima koji uzimaju u obzir stvarne uslove u kojima se mrežne komponente koriste. U istom poglavlju je data karakterizacija vremenskih parametara modela kojima se kvalitativno i kvantitativno opisuju ponašanje komponenata sistema. Eksperimentalni rezultati, korišćeni za karakterizaciju pomenutih parametara, dobijeni su mjerenjem na prototipskim mrežama koje se sastoje od komponenata različitih proizvođača.

U petom poglavlju su razmotrene tehnike i mehanizmi za optimizaciju vremena odziva prethodno analiziranih RTE sistema. Uzimajući u obzir specifičnosti svakog protokola, u ovom radu su predložena tri pristupa za optimizaciju *EtherCAT* i *Ethernet Powerlink* protokola u smislu

minimizacije vremena odziva, uvažavajući ograničenja koja postoje u svakom od protokola. Predloženi algoritmi i mehanizmi za optimizaciju su eksperimentalno verifikovani, a dobijeni rezultati su upoređeni sa rezultatima drugih autora. U slučaju *Profinet IO* protokola, detaljno je opisan DFP mehanizam i analizirane mogućnosti primjene ovog koncepta u stvarnim uslovima.

Konačno, u zaključku su sumirani dobijeni rezultati i doprinosi disertacije i predložene smjernice za dalje istraživanje u ovoj oblasti.

2 Industrijski komunikacioni sistemi

Industrijski komunikacioni sistemi zasnovani na *Ethernet* tehnologiji, poznatiji pod skraćenim nazivom RTE (engl. *Real-Time Ethernet*), postaju sve zastupljeniji u distribuiranim sistemima automatizacije. Osnovni razlog za ekspanziju ovih sistema može se pronaći u povećanju stepena fleksibilnosti uz istovremeno smanjenje troškova realizacije i održavanja. Ovo je direktna posljedica primjene komunikacione tehnologije koja je poznata dugo vremena i koja je veoma dobro prihvaćena u oblasti računarskih mreža, čime je, inherentno, olakšana integracija između distribuiranih uređaja, koji se koriste u polju primjene (engl. *field-level*), sa standardnim računarskim mrežama iz poslovnog sektora. Osim toga, *Ethernet* tehnologija podržava veoma velike bitske brzine, što omogućava prenos različitog saobraćaja (npr. upravljačke poruke, procesni podaci, multimedijalni sadržaj itd.) sa zahtjevanim nivoom kvaliteta putem istog fizičkog medijuma. Konačno, ova tehnologija se neprestano razvija i unapređuje tako da se očekuje da ostane *de-facto* standard u oblasti lokalnih mreža još dugi niz godina.

RTE sistemi najviših performansi iz treće kategorije, prema klasifikaciji datoj u [15, 16], zahtijevaju izmjene na sloju veze podataka *Ethernet* protokola kako bi se omogućio visok nivo determinizma komunikacije. Najznačajniji predstavnici ove kategorije, *EtherCAT*, *Ethernet Powerlink* i *Profinet IO*, predmet su ovog poglavlja. Detaljno su opisani servisi svakog protokola, struktura okvira koji se enkapsulira u standardne *Ethernet* okvire, kao i specifičnosti vezane za samu konfiguraciju sistema i organizaciju procesnih podataka unutar cikličnih okvira.

Sadržaj poglavlja daje jasnu predstavu o načinu rada svakog od rješenja, što omogućava detaljnu analizu njihovog rada i izvođenje odgovarajućih analitičkih modela koji dovoljno precizno opisuju njihovo ponašanje za scenarije prisutne u stvarnim sistemima automatizacije.

Poglavlje je organizovano na sljedeći način. U okviru odjeljka 2.1, opisan je *EtherCAT* protokol. Prikazane su osnovne karakteristike protokola koje uključuju princip rada, pregled komandi i načina adresiranja koji se koriste u sistemu, kao i postupak konfiguracije sistema. Posebna pažnja posvećena je strukturi fajla koji opisuje konfiguraciju mreže. U odjeljku 2.2,

date su osnove *Ethernet Powerlink* protokola. Prvo je opisan princip rada i struktura komunikacionog ciklusa, a zatim su detaljno objašnjeni načini rada koje podržava ovaj protokol. Takođe, komentarisana su ograničenja po pitanju topologije mreže, kao i specifičnosti vezane za njenu konfiguraciju. Osnove *Profinet IO* komunikacije, date su u odjeljku 2.3. Nakon opisa vrsta uređaja i tipičnog toka projektovanja i konfigurisanja *Profinet IO* mreža, detaljno je opisan model i načini identifikacije uređaja, kao i komunikacioni servisi koji su podržani u ovom sistemu. Posebna pažnja je posvećena organizaciji komunikacionog ciklusa, te opisu različitih klasa saobraćaja koje ovaj protokol podržava. Konačno, u odjeljku 2.4, izvedeni su osnovni zaključci.

2.1 Osnove EtherCAT protokola

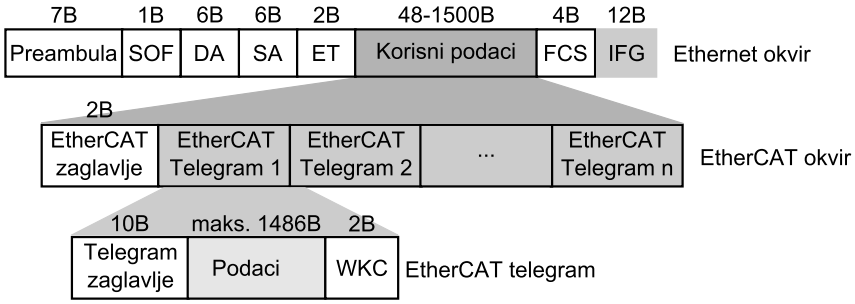
EtherCAT (ECAT) je RTE protokol, razvijen 2003. godine od strane njemačke kompanije *Beckhoff Automation*, koji tenutno podžava i unapređuje ETG (engl. *EtherCAT Technology Group*) organizacija. Protokol se koristi za efikasan prenos podataka između ulazno/izlaznih uređaja distribuiranih u pogonu (engl. *field devices*). Takođe, ECAT obezbjeđuje veoma preciznu sinhronizaciju čvorova (znatno ispod 1 μs, tipično reda 10 ns) što ga čini pogodnim za primjenu u tzv. *motion control* aplikacijama. Sistem podržava osnovne topologije (kao i bilo koju njihovu kombinaciju) koje se tipično koriste u sistemima automatizacije: linijsku topologiju, te topologije zvijezde, prstena ili stabla.

ECAT koristi tzv. *koncept sumacionog okvira* za prenos podataka između umreženih čvorova. Ovaj koncept podrazumijeva prenos procesnih podataka više uređaja korišćenjem jednog *Ethernet* okvira, za razliku od pristupa sa indvidualnim okvirima (npr. *Ethernet Powerlink*, *Profinet IO*, itd.) kod kojih svaki uređaj šalje zaseban (individualan) *Ethernet* okvir sa procesnim podacima. S obzirom da je minimalan *Ethernet* okvir veličine 64 bajta (46 bajtova korisnih podataka) i da su procesni podaci u sistemima automatizacije veličine samo nekoliko bajtova koji se razmjenjuju periodično frekevencijom koju definiše odgovarajuća primjena, jasno je da se korišćenjem ovog koncepta postiže bolje iskorišćenje propusnog opsega jer se izbjegava dopunjavanje bajtovima do minimalne veličine *Ethernet* okvira. Kako bi se podaci namijenjeni određenom uređaju mogli identifikovati unutar sumacionog okvira, koristi se posebna jedinica podataka, *ECAT telegram* (ili datagram), sa zasebnim zaglavljem veličine 10 bajtova koje sadrži kontrolne infromacije poput adrese uređaja, zahtijevane komande itd. Osim zaglavlja, svaki ECAT telegram na svom kraju sadrži WKC (engl. *Working Counter*) polje koje se koristi za dijagnostičke potrebe. Naime, svaki uređaj u mreži uvećava sadržaj ovog polja nakon uspješnog

prenosa podataka za unaprijed definisanu vrijednost. S obzirom da je očekivana vrijednost WKC polja u slučaju uspješnog prenosa podataka unaprijed poznata, prostim poređenjem stvarne vrijednosti polja sa očekivanom moguće je na jednostavan način detektovati eventualne greške u radu sistema. Struktura jednog ECAT okvira prikazana je na slici 2.1.

Kao što može da se vidi sa slike, ECAT okvir se enkapsulira u polje korisnih podataka *Ethernet* okvira čije zaglavlje čine preambula (56-bitno polje alternativnih jedinica i nula), bajt za označavanje početka okvira ili SOF (engl. *Start of Frame*), odredišna i izvorna MAC (engl. *Media Access Control*) adresa veličine 6 bajtova, kao i ET (engl. *EtherType*) polje veličine 2 bajta koje označava tip protokola koji se enkapsulira u *Ethernet* okviru (0x88a4 u slučaju ECAT protokola). *Ethernet* okvir se završava poljem za provjeru ispravnosti okvira, FCS (engl. *Frame Check Sequence*), čija je veličina 4 bajta. Ukoliko je veličina korisnih podataka manja od 46 bajtova, okvir se dopunjava tako da njegova minimalna veličina (bez preambule i SOF polja) bude 64 bajta. Prilikom uzastopnog slanja okvira, potrebno je obezbijediti da minimalno rastojanje između kraja prethodnog i početka narednog okvira ne bude manje od 12 bajtova. Opisano rastojanje se naziva IFG (engl. *Interframe Gap*), i u slučaju 100 Mb/s *Ethernet* komunikacije iznosi 960 ns.

ECAT okvir čine zaglavlje veličine 2 bajta i jedan ili više ECAT telegrama. Svaki telegram adresira jedan ili više (zavisno od tipa adresiranja) uređaja u mreži. Komunikacija se inicira slanjem ECAT okvira koji generiše centralni uređaj u mreži pod nazivom *nadređeni uređaj* (engl. *master*). Ovaj uređaj obezbjeđuje i kontroliše sve komunikacione servise u sistemu čime omogućava razmjenu podataka između *podređenih uređaja* (engl. *slave*). Svaki ECAT okvir koji šalje nadređeni uređaj prolazi kroz sve podređene uređaje u mreži a zatim se vraća nadređenom uređaju nakon što je prošao posljednji podređeni uređaj formirajući na taj način logičku topologiju prstena bez obzira na fizičku topologiju koja je implementirana u sistemu. ECAT okvire obrađuje posebna hardverska jedinica pod nazivom ESC (engl. *EtherCAT Slave Controller*) koja je sastavni dio svakog podređenog uređaja. Ova jedinica omogućava razmjenu procesnih podataka



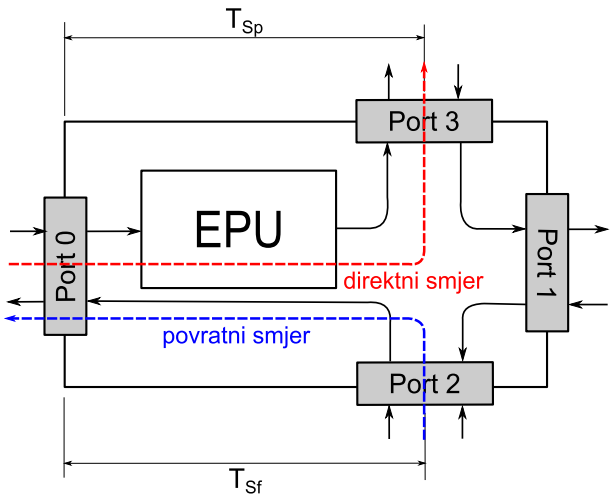
Slika 2.1: Struktura i način enkapsulacije ECAT okvira

između interne memorije uređaja i ECAT okvira dok okvir prolazi kroz podređeni uređaj (tzv. *on-the-fly* mehanizam), čime se minimizuje vrijeme zadržavanja okvira u podređenom uređaju što značajno smanjuje vrijeme propagacije kroz uređaje u mreži. Pojednostavljena struktura ESC jedinice je prikazana na slici 2.2. U osnovi, ona se sastoji iz četiri porta. Portovi mogu da budu omogućeni ili neomogućeni. Kada je port onemogućen, on samo prosljeđuje okvir sa zanemarljivim kašnjenjem sljedećem portu u nizu, čime je omogućena realizacija podređenih uređaja sa dva, tri ili četiri porta. Port 0 je uvijek ulazni port. Okvir koji dolazi na ovaj port se prosljeđuje preko EPU (engl. *EtherCAT Processing Unit*) jedinice za obradu njegovog sadržaja u tzv. *direktnom smjeru*. Prilikom vraćanja okvira, u *povratnom smjeru*, okvir prolazi samo kroz portove i ne zadržava se u EPU jedinici. Ukupno vrijeme propagacije kroz ESC, može se definisati kao zbir kašnjenja u direktnom (T_{Sp}) i povratnom (T_{Sf}) smjeru, tj. $T_S = T_{Sp} + T_{Sf}$. Vremena propagacije u direktnom i povratnom smjeru zavise od strukture samog uređaja kao i od realizacije interfejsa na fizičkom sloju. S obzirom da se u direktnom smjeru okvir zadržava nešto duže zbog obrade njegovog sadržaja u EPU jedinici, uvijek postoji asimetrija u kašnjenju okvira u ova dva smjera koja se obično kvantifikuje parametrom $T_{diff} = T_{Sp} - T_{Sf}$.

ECAT protokol trenutno podržava dva tipa protokola na fizičkom sloju: standardni *Ethernet 100BASE-TX* i *E-bus* interfejs. E-bus interfejs se koristi kod modularnih uređaja za povezivanje modula i u potpunosti je kompatibilan sa *Ethernet* standardom. Ovaj interfejs obezbjeđuje manje vrijeme zadržavanja okvira u ESC jedinici.

2.1.1 Načini adresiranja

Kao što je prethodno opisano, ECAT koristi koncept sumacionog okvira na način da se jednim *Ethernet* okvirom prenosi više telegrama koji sadrže procesne podatke za jedan ili



Slika 2.2: Struktura ESC jedinice

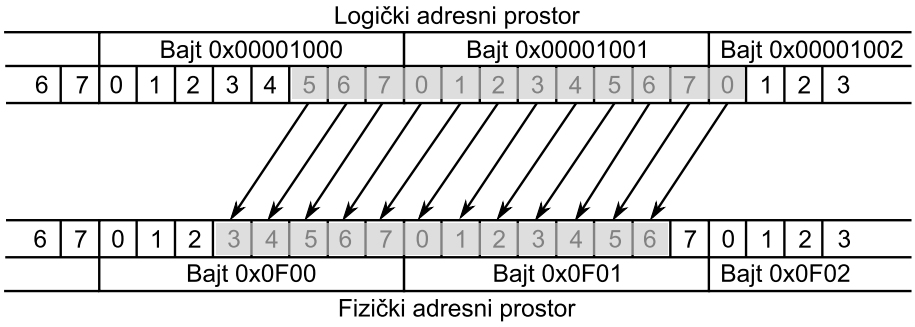
više podređenih uređaja. Ovaj koncept podrazumijeva definisanje načina adresiranja na nivou telegrama, što znači da se standardne MAC adrese u *Ethernet* okviru ne mogu koristiti za adresiranje ECAT uređaja. Struktura zaglavlja ECAT telegrama je prikazana na slici 2.3. Brojevi u zagradama određuju veličinu svakog polja izraženu u bitima.

Bilo koji podređeni uređaj u ECAT mreži se adresira korišćenjem jednog od dva specifična načina adresiranja: *čvornog* ili *logičkog*. Čvornim adresiranjem se podaci u okviru telegrama dostavljaju podređenom uređaju koji je određen posebnim poljem adrese u zaglavlju telegrama. Polje adrese može da definiše poziciju uređaja u logičkom prstenu (tzv. *poziciono adresiranje* ili *adresiranje sa automatskim inkrementovanjem adrese*) ili fiksnu adresu dodijeljenu čvoru prilikom konfiguracije sistema (tzv. *konfigurisano* ili *fiksno adresiranje*). Poziciono adresiranje se tipično koristi prilikom inicijalizacije sistema kada fiksne adrese još uvijek nisu definisane i dodijeljene čvorovima. Ono omogućava automatsko skeniranje i detekciju mrežne topologije koja se koristi u sistemu. Naime, polje adrese se podešava tako da odgovara fizičkoj poziciji podređenog uređaja u logičkom prstenu. Svaki uređaj uvećava vrijednost adresnog polja koje, kada postane jednako nuli, ukazuje na to da je ECAT okvir stigao do odredišnog uređaja. Fiksno adresiranje se koristi za konfiguraciju svakog čvora pojedinačno i za prenos asinhronih poruka u sistemu. Adresno polje kod ovog tipa adresiranja predstavlja jedinstvenu adresu koju postavlja nadređeni uređaj prilikom inicijalizacije podređenih uređaja. Kod oba tipa adresiranja, adresa čvora je definisana 16-bitnim ADP (engl. *Address Position*) poljem u zaglavlju telegrama u kojem se specificira pozicija uređaja u logičkom prstenu (poziciono adresiranje), odnosno njegova konfigurisana adresa (fiksno adresiranje). Na taj način je omogućeno adresiranje do 65536 podređenih uređaja u mreži. Dodatno 16-bitno ADO (engl. *Address Offset*) polje se koristi za adresiranje konkretnog registra u lokalnoj memoriji (64 kB adresnog prostora) podređenog uređaja.

Logičko adresiranje koristi 32-bitno polje u zaglavlju pod nazivom ADR (engl. *Logical Address*) umjesto polja ADP i ADO. Ovaj tip adresiranja se tipično koristi za razmjenu cikličnih procesnih podataka. Koncept logičkog adresiranja omogućava pristup zajedničkom adresnom prostoru ukupne veličine 4 GB koji se naziva procesna slika i koju dijele svi uređaji u mreži. U svakom podređenom uređaju se nalazi FMMU (engl. *Fieldbus Memory Management Unit*) jedinica koja je zadužena za transliranje logičke adrese u odgovarajuću fizičku adresu u uređaju

<table>
<tr><td rowspan="2">Cmd
(8)</td><td rowspan="2">Idx
(8)</td><td>ADP (16)</td><td>ADO (16)</td><td rowspan="2">Len
(11)</td><td rowspan="2">R
(3)</td><td rowspan="2">C
(1)</td><td rowspan="2">M
(1)</td><td rowspan="2">IRQ
(16)</td></tr>
<tr><td colspan="2">ADR (32)</td></tr>
</table>

Slika 2.3: Struktura zaglavlja ECAT telegrama



Slika 2.4: Ilustracija translacije logičke u fizičku adresu

kao što je ilustrovano slikom 2.4. U primjeru sa slike se 12-bitni podatak iz 32-bitnog logičkog adresnog prostora kojeg čine najviša 3 bita bajta na adresi 0x00001000, kompletan bajt na adresi 0x00001001 i najniži bit bajta na adresi 0x00001002 transliraju u 12-bitni podatak u 16-bitnom fizičkom adresnom prostoru kojeg čine 5 najviših bita bajta na adresi 0x0F00 i 7 nižih bita bajta na adresi 0x0F01. Opisani koncept je sličan konceptu virtuelne memorije koji se koristi u računarskim sistemima. Ovim načinom adresiranja se dodatno povećava iskorišćenje propusnog opsega jer se procesni podaci više različitih podređenih uređaja mogu mapirati u isti ECAT telegram.

ECAT protokol definiše i specijalan tip adresiranja, tzv. sveodredišno (engl. *broadcast*) adresiranje, kod kojeg svi uređaji učestvuju u razmjeni podataka. Ovaj tip adresiranja se koristi za dijagnostičke poruke (npr. provjera trenutnog stanja svih podređenih uređaja).

2.1.2 Komande

Bilo kojem podređenom uređaju u ECAT mreži se može pristupiti korišćenjem četiri osnovna komunikaciona servisa definisana na sloju veze podataka: jedno čitanje (RD), jedan upis (WR), jedno čitanje sa jednim upisom (RW) i jedno čitanje sa višestrukim upisom (RMW). Komanda kojom se određuje vrsta servisa kao i način adresiranja se specificira u polju *Cmd* u okviru zaglavlja ECAT telegrama (slika 2.3).

Pristup sa jednim čitanjem koristi nadređeni uređaj prilikom preuzimanja vrijednosti iz nekog od ulaznih registara podređenog uređaja, dok pristup sa jednim upisom omogućava upis vrijednosti u neki od izlaznih registara podređenog uređaja. Pristup sa jednim čitanjem i jednim upisom kombinuje prethodna dva servisa tako da se podaci mogu razmijeniti dvosmjerno korišćenjem samo jedne komande. Posljednji servis omogućava čitanje vrijednosti adresiranog uređaja (korišćenjem pozicionog ili fiksnog adresiranja) i upis pročitane vrijednosti u registre uređaja koji se nalaze poslije adresiranog uređaja. U ECAT sistemu, ovaj servis se tipično koristi za sinhronizaciju čvorova u mreži ili, preciznije, za distribuciju sistemskog vremena referentnog

uređaja ostalim uređajima.

Svaki od opisanih servisa može se kombinovati sa bilo kojim od prethodno opisanih načina adresiranja, s izuzetkom RMW servisa koji može da se koristi samo sa pozicionim i fiksnim adresiranjem. Pregled svih dostupnih komandi definisanih ECAT specifikacijom je prikazan u tabeli 2.1.

Tabela 2.1: Pregled komandi definisanih ECAT specifikacijom

Kôd	Naziv	Opis
0	NOP	Uređaj ignoriše komandu
1	APRD	Uređaj inkrementuje ADP polje a zatim u ECAT telegram smješta podatak iz memorijske lokacije adresirane ADO poljem ako je ADP jednako nuli.
2	APWR	Uređaj inkrementuje ADP polje a zatim upisuje podatak iz ECAT telegrama na memorijsku lokaciju adresiranu ADO poljem ako je ADP jednako nuli.
3	APRW	Uređaj inkrementuje ADP polje a zatim u ECAT telegram smješta podatak iz memorijske lokacije adresirane ADO poljem i upisuje podatak iz ECAT telegrama na istu memorijsku lokaciju ako je ADP jednako nuli.
4	FPRD	Uređaj u ECAT telegram smješta podatak iz memorijske lokacije adresirane ADO poljem ako ADP polje odgovara fiksnoj adresi uređaja.
5	FPWR	Uređaj upisuje podatak iz ECAT telegrama na memorijsku lokaciju adresiranu ADO poljem ako ADP polje odgovara fiksnoj adresi uređaja.
6	FPRW	Uređaj u ECAT telegram smješta podatak iz memorijske lokacije adresirane ADO poljem i upisuje podatak iz ECAT telegrama na istu memorijsku lokaciju ako ADP polje odgovara fiksnoj adresi uređaja.
7	BRD	Svi uređaji u ECAT telegram smještaju rezultat logičke ILI operacije između sadržaja ECAT telegrama i memorijske lokacije adresirane ADO poljem. Svi uređaji inkrementuju ADP polje.
8	BWR	Svi uređaji upisuju podatak iz ECAT telegrama na memorijsku lokaciju adresiranu ADO poljem. Svi uređaji inkrementuju ADP polje.
9	BRW	Svi uređaji u ECAT telegram smještaju rezultat logičke ILI operacije između sadržaja ECAT telegrama i memorijske lokacije adresirane ADO poljem, a zatim upisuju podatak iz ECAT telegrama na istu memorijsku lokaciju. Svi uređaji inkrementuju ADP polje.
10	LRD	Uređaj u ECAT telegram smješta ulazne podatke ako ADR polje odgovara logičkoj adresi podešenoj u nekoj od FMMU jedinica.
11	LWR	Uređaj upisuje izlazni podatak iz ECAT telegrama ako ADR polje odgovara logičkoj adresi podešenoj u nekoj od FMMU jedinica.
12	LRW	Uređaj obavlja dvosmjernu razmjenu podataka između interne memorije i ECAT telegrama ako ADR polje odgovara logičkoj adresi podešenoj u nekoj od FMMU jedinica.
13	ARMW	Uređaj inkrementuje ADP polje a zatim u ECAT telegram smješta podatak iz memorijske lokacije adresirane ADO poljem ako je ADP jednako nuli, inače upisuje podatak iz ECAT telegrama na istu memorijsku lokaciju.
14	FRMW	Uređaj u ECAT telegram smješta podatak iz memorijske lokacije adresirane ADO poljem ako ADP polje odgovara fiksnoj adresi uređaja, inače upisuje podatak iz ECAT telegrama na istu memorijsku lokaciju.

2.1.3 ECAT mašina stanja

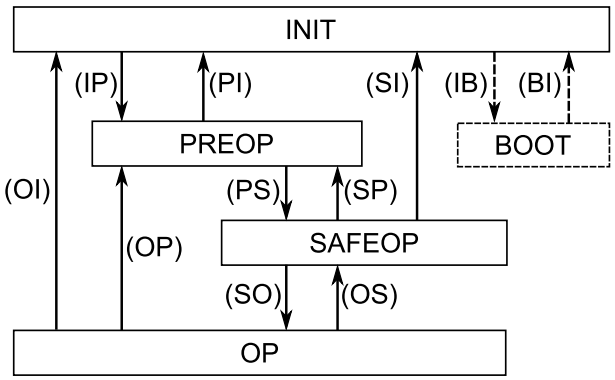
ECAT mašina stanja ili ESM (engl. *EtherCAT State Machine*) obezbjeđuje servise neophodne za upravljanje mrežom, tj. obavlja funkcije koordinacije nadređenog i podređenih uređaja prilikom pokretanja i normalnog rada sistema. Prelazak iz jednog u drugo stanje inicira i kontroliše nadređeni uređaj slanjem odgovarajućeg zahtjeva lokalnim aplikacijama u podređenim uređajima. Koncept upravljanja mrežom korišćenjem mašine stanja je naslijeđen iz *CANopen* protokola, s tim da je struktura mašine stanja u ECAT sistemu nešto drugačija.

ESM definiše četiri obavezna i jedno opciono stanje (slika 2.5):

- stanje inicijalizacije (INIT),
- predoperaciono stanje (PREOP),
- bezbjedno operaciono stanje (SAFEOP),
- operaciono stanje (OP) i
- opciono stanje ažuriranja sistema (BOOT).

Slika 2.5 takođe prikazuje doozvoljene tranzicije između pojedinih stanja, npr. IP tranzicija odgovara prelazu iz INIT u PREOP stanja, i sl. Zahtjev za promjenu trenutnog stanja podređeni uređaj inicira upisivanjem odgovarajuće vrijednosti u kontrolni registar koji se nalazi u internoj memoriji podređenog uređaja. Aplikacija u podređenom uređaju prelazi u zahtijevano stanje (ukoliko je to moguće) a zatim mijenja sadržaj statusnog registra u internoj memoriji kako bi reflektovala trenutno stanje uređaja, odnosno prikazuje informaciju o greški ukoliko nije moguće obaviti traženu tranziciju.

Početno stanje prilikom pokretanja sistema je *stanje inicijalizacije*. Ovim stanjem se ukazuje na to da je uspostavljena osnovna komunikacija na nivou aplikativnog sloja između nadređenog



Slika 2.5: ECAT mašina stanja

i jednog ili više podređenih uređaja. Direktna komunikacija između uređaja u ovom stanju nije moguća. Nadređeni uređaj u INIT stanju inicijalizuje skup konfiguracionih registara podređenih uređaja kako bi se uspostavili osnovni komunikacioni servisi i omogućio prelazak u naredno stanje.

Predoperaciono stanje se koristi za prenos parametara i konfiguracionih podataka koje koriste servisi aplikativnog sloja komunikacije kako bi se omogućila razmjena procesnih podataka. U ovom stanju se uspostavljaju osnovni servisi koji omogućavaju prenos pomenutih parametara konfiguracije.

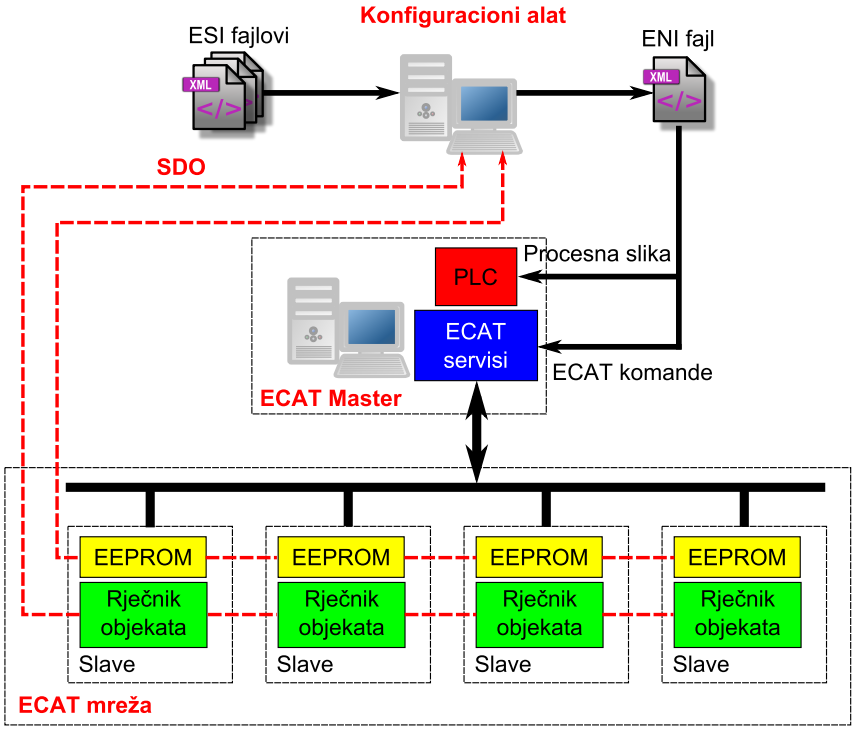
U *bezbjednom operacionom stanju* je moguća razmjena procesnih podataka. Međutim, dozvoljeno je slanje samo ulaznih procesnih podataka dok se izlazni procesni podaci nalaze u tzv. *bezbjednom stanju* koje garantuje da u sistemu neće doći do pojave hazardnih situacija koje bi mogle da dovedu do oštećenja opreme ili ugrožavanja ljudskih života.

Posljednje stanje u koje prelazi aplikacija je *operaciono stanje*. U ovom stanju nadređeni uređaj šalje izračunate vrijednosti za izlazne procesne podatke koji se dostavljaju aplikacijama odgovarajućih podređenih uređaja i nastavlja sa razmjenom ulaznih procesnih podataka. Operaciono stanje je radno stanje sistema.

Stanje ažuriranja sistema je opciono ali se preporučuje ako je potrebno obezbjediti instalaciju nove verzije softvera unutar podređenog uređaja. Kopiranje svih fajlova koji su neophodni za ažuriranje podređenog uređaja se obavlja korišćenjem specifičnih komunikacionih servisa koji su aktivni samo u ovom stanju. Prealazak u ovo stanje i izlazak iz njega je dozvoljeno samo ako se sistem nalazi u INIT stanju.

2.1.4 Konfiguracija ECAT mreže

U ECAT sistemu, proces konfiguracije mreže je odvojen od dijela sistema koji realizuje komunikacione servise (slika 2.6). Alat za konfiguraciju mreže (obično specijalan softver koji se pokreće na standardnom PC računaru) može da bude povezan na mrežu i da na taj način automatski, korišćenjem odgovarajućih komunikacionih servisa, detektuje sve prisutne podređene uređaje i prikuplja relevantne informacije iz EEPROM memorije uređaja i/ili lokalnog rječnika objekata (engl. *Object Dictionary*) putem SDO servisa u slučaju uređaja koji podržavaju *CANopen* protokol. Ako alat za konfiguraciju mreže nije povezan na mrežu, informacije o podređenim uređajima se mogu dobiti iz XML (engl. *eXtensible Markup Language*) baziranih ESI (engl. *EtherCAT Slave Information*) fajlova koji sadrže opis karakteristika podređenog uređaja.



Slika 2.6: Konfiguracija ECAT mreže

Na osnovu informacija prikupljenih od podređenih uređaja, alat za konfiguraciju generiše XML bazirani ENI (engl. *EtherCAT Network Information*) fajl koji opisuje konfiguraciju ECAT mreže. Ovaj fajl sadrži opšte informacije o nadređenom uređaju, kao i konfiguraciju svakog podređenog uređaja u mreži (pozicija u logičkom prstenu, konfigurisana adresa uređaja, veličina procesnih podataka, itd.). Osim toga, ENI fajl definiše komande koje se šalju u određenim stanjima nadređenog uređaja. Nadređeni uređaj učitava ovaj fajl prilikom incijalizacije sistema i interpretira njegov sadržaj kako bi mogao da inicijalizuje sistem i konfiguriše sve podređene uređaje prisutne u ECAT mreži.

2.1.5 Struktura ENI fajla

Kao što je prethodno objašnjeno, ENI fajl sadrži informacije neophodne nadređenom uređaju da incijalizuje i konfiguriše sve podređene uređaje prisutne u mreži. ENI fajl se sastoji iz niza XML elemenata i atributa definisanih odgovarajućim specifikacijama. Strukturu svakog ENI fajla čine XML oznake prikazane u listingu 2.1.

Kompleksni XML element definisan oznakom `<Master>` sadrži konfiguraciju nadređenog uređaja. Detaljniji prikaz ovog elementa je ilustrovan primjerom konfiguracije u okviru listinga 2.2. Konfiguracija nadređenog uređaja sadrži osnovne informacije o uređaju (oznaka `<Info>`) kao što su naziv uređaja (`<Name>`), izvorna i odredišna MAC adresa (`<Source>` i `<Destination>`), te tip protokola (`<EtherType>`) koji je enkapsuliran u *Ethernet* okvir (u slučaju

Listing 2.1: Osnovna struktura ENI fajla

```
<EtherCATConfig>
        <Config>
                <Master>
                        ...
                </Master>
                <Slave>
                        ...
                </Slave>
                        ...
                <Cyclic>
                        ...
                </Cyclic>
                <ProcessImage>
                        ...
                </ProcessImage>
        </Config>
</EtherCATConfig>
```

ECAT protokola ovaj element uvijek ima vrijednost 0xa488).

Lista komandi kojim nadređeni uređaj incijalizuje sve podređene uređaje u mreži se nalazi u okviru elementa sa XML oznakom `<InitCmds>`. Svaka komanda (`<InitCmd>`) sadrži informacije

Listing 2.2: Primjer konfiguracije nadređenog uređaja

```
<Master>
        <Info>
                <Name>Device 1 (EtherCAT)</Name>
                <Destination>010105010000</Destination>
                <Source>000cf1dea5fc</Source>
                <EtherType>a488</EtherType>
        </Info>

        <InitCmds>
                ...
                <InitCmd>
                        <Transition>IP</Transition>
                        <BeforeSlave>1</BeforeSlave>
                        <Comment>clear configured addresses</Comment>
                        <Requires>cycle</Requires>
                        <Cmd>8</Cmd>
                        <Adp>0</Adp>
                        <Ado>16</Ado>
                        <Data>0000</Data>
                        <Retries>3</Retries>
                </InitCmd>
                ...
        </InitCmds>
</Master>
```

Listing 2.3: Primjer konfiguracije podređenog uređaja

```
<Slave>
        <Info>
                <Name>Term 22 (EL3403)</Name>
                <PhysAddr>1022</PhysAddr>
                <AutoIncAddr>65515</AutoIncAddr>
                <Physics>KK</Physics>
                <VendorId>2</VendorId>
                <ProductCode>223031378</ProductCode>
                <RevisionNo>1245184</RevisionNo>
                <SerialNo>0</SerialNo>
                <ProductRevision>EL3403-0000-0019</ProductRevision>
        </Info>
        <ProcessData>
                <Send>
                        <BitStart>568</BitStart>
                        <BitLength>24</BitLength>
                </Send>
                <Recv>
                        <BitStart>568</BitStart>
                        <BitLength>496</BitLength>
                </Recv>
                ...
        </ProcessData>
                ...
        <InitCmds>
                ...
        </InitCmds>
        <PreviousPort Selected="1">
                <Port>B</Port>
                <PhysAddr>1021</PhysAddr>
        </PreviousPort>
        ...
</Slave>
```

o tranziciji (`<Transition>`) pri kojoj se šalje odgovarajuća ECAT komanda (`<Cmd>`) sa definisanim adresama uređaja i interne registarske mape (`<Adp>` i `<Ado>`), kao i podacima (`<Data>`) koji trebaju da budu upisani u željenu lokaciju. Opis komande definiše još neke dodatne informacije kao što je zahtjev da se data komanda šalje prije komandi koje su specifične za podređeni uređaj (oznaka `<BeforeSlave>`), odnosno parametar koji definiše da li komanda treba da se šalje u odvojenom ciklusu ili okviru (oznaka `<Requires>`). Opciono, u okviru opisa komande može da se specificira komentar (`<Comment>`) u kojem se bliže opisuje šta data komanda treba da konfiguriše u sistemu. Konačno, u okviru elementa `<Retries>` se definiše koliko puta se ponavlja slanje komande prije proglašenja inicijalizacije neuspješnom.

Opis komande u navedenom primjeru definiše komandu sa kodom 8 (BWR) koja upisuje

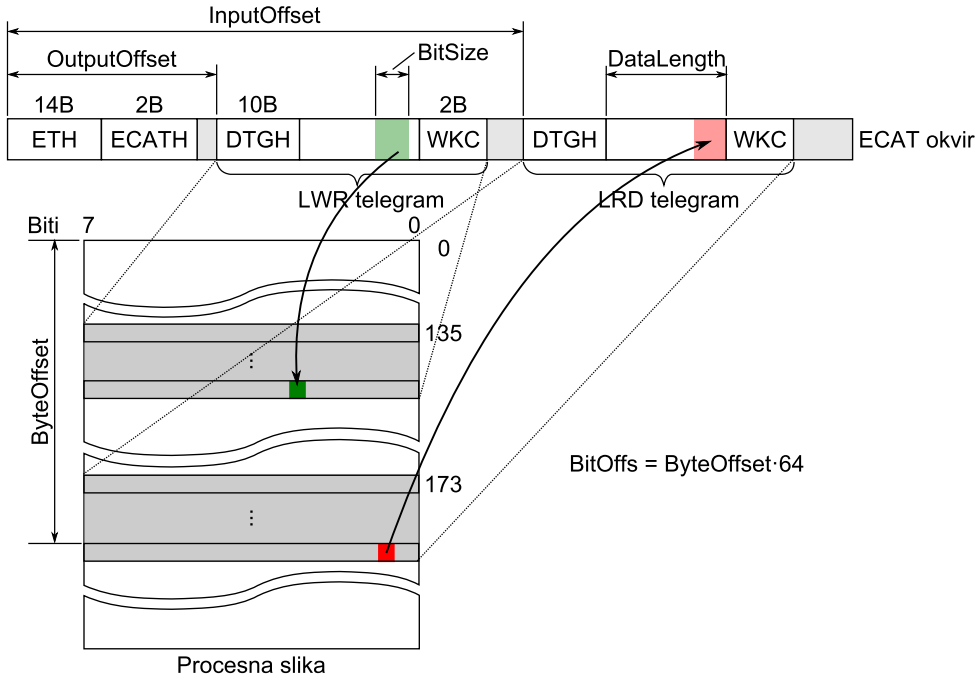
Listing 2.4: Primjer strukture ECAT okvira u ENI fajlu

```
<Cyclic>
        <Frame>
                ...
                <Cmd>
                        <State>SAFEOP</State>
                        <State>OP</State>
                        <Comment>cyclic cmd</Comment>
                        <Cmd>11</Cmd>
                        <Addr>16779264</Addr>
                        <DataLength>26</DataLength>
                        <Cnt>11</Cnt>
                        <InputOffset>135</InputOffset>
                        <OutputOffset>135</OutputOffset>
                </Cmd>
                <Cmd>
                        <State>SAFEOP</State>
                        <State>OP</State>
                        <Comment>cyclic cmd</Comment>
                        <Cmd>10</Cmd>
                        <Addr>16781312</Addr>
                        <DataLength>79</DataLength>
                        <Cnt>6</Cnt>
                        <InputOffset>173</InputOffset>
                        <OutputOffset>173</OutputOffset>
                </Cmd>
                ...
        </Frame>
</Cyclic>
```

vrijednost 0x0000 u registar sa adresom 0x0010 svih podređenih uređaja. Komanda se šalje u zasebnom ciklusu prije specifičnih komandi podređenog uređaja i pri IP tranziciji, tj. kada se prelazi iz stanja inicijalizacije u predoperaciono stanje. Nadređeni uređaj će pokušati tri puta da pošalje ovu inicijalizacionu komandu prije nego što inicijalizaciju proglasi neuspješnom. Iz priloženog komentara može da se zaključi da ova komanda briše konfigurisane (fiksne) adrese svih podređenih uređaja.

U okviru kompleksnog XML elementa sa oznakom `<Slave>` definišu se parametri konfiguracije kao i specifične inicijalizacione komande koje se odnose na definisani podređeni uređaj. Broj ovih elemenata u ENI fajlu je određen brojem aktivnih čvorova u ECAT mreži. Struktura elementa koji sadrži konfiguraciju podređenog uređaja je slična `<Master>` elementu, a jedan primjer njegovog izgleda za konkretan uređaj je dat u listingu 2.3.

Kao i kod nadređenog uređaja, u okviru `<Slave>` elementa, definišu se osnovne informacije o uređaju (`<Info>`) kao što su naziv uređaja (`<Name>`), poziciona (`<AutoIncAddr>`) i fiksna



Slika 2.7: Primjer mapiranja telegrama u procesnu sliku ECAT sistema

(`<PhysAddr>`) adresa, interfejs korišćen na fizičkom sloju komunikacije (`<Physics>`), kao i informacije o proizvođaču i samom proizvodu. Specifične ECAT komande za inicijalizaciju uređaja, definišu se u ekvivalentnom XML elementu (`<InitCmds>`) koji sadrži XML oznake sa istom strukturom i formatom kao kod nadređenog uređaja.

Najznačajniji segment u ovom dijelu ENI fajla predstavlja definisanje parametara procesnih podataka (`<ProcessData>`) koje razmjenjuje podređeni uređaj. U tom smislu, definišu se pozicija (`<BitStart>`) i veličina podatka (`<BitLength>`) izražena brojem bita relativno u odnosu na početak procesne slike u nadređenom uređaju. Ulazni podaci se definišu u okviru `<Send>` a izlazni u okviru `<Recv>` XML elementa.

Informacija o topologiji mreže je sadržana u `<PreviousPort>` elementu koji definiše port (`<Port>`) i fiksnu adresu (`<PhysAddr>`) podređenog uređaja sa kojim je povezan uređaj koji se konfiguriše.

Kao što može da se vidi iz listinga, svaki okvir (`<Frame>`) prenosi jedan ili više ECAT telegrama, od kojih svaki sadrži podatke za jedan ili više podređenih uređaja. Struktura svakog telegrama je opisana kompleksnim XML elementom `<Cmd>` koji sadrži informacije o ECAT stanjima (`<State>`) u kojima se telegram šalje, komandi (`<Cmd>`) koja se definiše u zaglavlju, logičkoj adresi (`<Addr>`), veličini podataka (`<DataLength>`) izraženu brojem bajtova, kao i očekivanoj vrijednosti WKC (`<Cnt>`). Relativna pozicija podataka u odnosu na početak okvira se specificira `<InputOffset>` i `<OutputOffset>` elementima. Kao i kod definisanja komandi za incijalizaciju sistema, korisnik može navesti komentar (`<Comment>`) u kojem bliže definiše ulogu

Listing 2.5: Primjer strukture procesne slike u ENI fajlu

```
<ProcessImage>
        <Inputs>
                <ByteSize>1536</ByteSize>
                ...
                <Variable>
                        <Name>Term 2 (EL1012).Channel 1.Input</Name>
                        <DataType>BOOL</DataType>
                        <BitSize>1</BitSize>
                        <BitOffs>2088</BitOffs>
                </Variable>
                ...
        </Inputs>
        <Outputs>
        <ByteSize>1536</ByteSize>
        ...
                <Variable>
                        <Name>Term 3 (EL2002).Channel 1.Output</Name>
                        <DataType>BOOL</DataType>
                        <BitSize>1</BitSize>
                        <BitOffs>1352</BitOffs>
                </Variable>
        ...
        </Outputs>
</ProcessImage>
```

datog telegrama.

U primjeru iz listinga 2.4 definisana su dva ECAT telegrama čija je struktura i način mapiranja podataka na dijeljenu procesnu sliku ilustrovana na slici 2.7. Kao što može da se zaključi iz navedenog primjera, oba telegrama nose ciklične podatke koji se razmjenjuju kada se sistem nalazi u bezbijednom operacionom (SAFEOP) i operacionom (OP) stanju.

Organizacija procesnih podataka unutar procesne slike sistema (definicija promjenljivih koje čuvaju vrijednosti procesnih podataka na nivou sistema automatizacije), kao i način mapiranja promjenljivih u ECAT telegrame opisan je u okviru XML elementa `<ProcessImage>`. U osnovi, procesna slika se sastoji iz dva dijela: ulaznih (`<Inputs>`) i izlaznih (`<Outputs>`) promjenljivih. Svaki dio sadrži određeni broj promjenljivih čija je struktura opisana XML elementom `<Variable>`. Veličina svakog dijela procesne slike definiše se `<ByteSize>` elementom izraženo brojem bajtova.

Svaka promjenljiva u procesnoj slici sadrži naziv promjenljive (`<Name>`), tip podatka (`<DataType>`), veličinu (`<BitSize>`) izraženu brojem bita, kao i relativni bitski pomjeraj promjenljive (`<BitOffs>`) u odnosu na početak ECAT okvira.

Jedan primjer strukture procesne slike stvarnog ECAT sistema je prikazan listingom 2.5. U

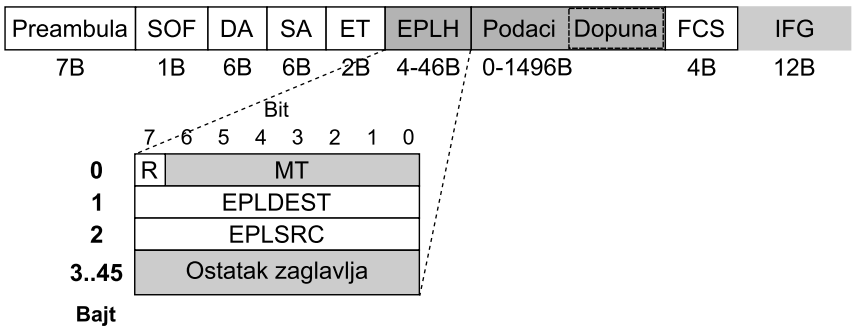
navedenom primjeru, prikazana je definicija dvije promjenljive, a grafička ilustracija mapiranja ovih promjenljivih u ECAT telegrame, prikazana je na slici 2.7.

2.2 Osnove Ethernet Powerlink protokola

Ethernet Powerlink (EPL) je industrijski komunikacioni protokol koji omogućava determinističku razmjenu podataka između uređaja koji koriste standardne *Ethernet* mrežne interfejse. EPL koncept je predložen 2001. godine od strane austrijske kompanije *Bernecker+Rainer Industrie-Elektronik*. Razvoj i promociju EPL protokola danas podržava i obezbjeđuje EPSG (engl. *Ethernet Powerlink Standardization Group*) organizacija. Za razliku od ECAT koncepta, EPL koristi individualne okvire za prenos cikličnih procesnih podataka i asinhronih (necikličnih) poruka. Osnovni princip rada svodi se na prozivanje (engl. *polling*) čvorova prema unaprijed isplaniranom rasporedu. Karakterističnost EPL protokola u odnosu na druge pristupe ogleda se u tome da on koristi poludupleksnu *Ethernet* komunikaciju, što znači da se umjesto mrežnih komutatora (engl. *network switch*) kao infrastrukturne komponente za povezivanje krajnjih uređaja koriste *koncentratori* ili *regeneratori signala* (engl. *repeating hub*). Time se postiže kraće vrijeme propagacije vremenski kritičnih poruka sa veoma malom varijacijom kašnjenja, *džiterom* (engl. *jitter*), što omogućava preciznu sinhronizaciju čvorova (ispod 1 μs) i postizanje visokog nivoa determinizma komunikacije u sistemu.

Da bi obezbijedio determinističku komunikaciju, EPL protokol koristi tehniku pod nazivom *upravljanje mrežom sa vremenskim slotovima* ili skraćeno SCNM (engl. *Slot Communication Network Management*). Komunikacijom u sistemu upravlja specijalan uređaj koji se naziva *upravljački čvor*, MN (engl. *Managing Node*), dok se svi ostali uređaji u mreži nazivaju *upravljani čvorovi*, CN (engl. *Controlled Node*). Svaki EPL segment mreže sastoji se iz jednog MN i maksimalno 239 CN čvorova. Kao što je prethodno pomenuto, upravljački čvor proziva svaki upravljani čvor unutar predefinisanog vremenskog slota u komunikacionom ciklusu koji se dodjeljuje odgovarajućem čvoru prilikom konfiguracije mreže. Na taj način, slanje podataka dozvoljava se samo jednom čvoru unutar datog vremenskog slota. Trajanje svakog vremenskog slota je ograničeno vremenskim intervalom koji se naziva *PResTimeout* (T_{TO}).

EPL poruke se enkapsuliraju u *Ethernet* okvire kao što je prikazano na slici 2.8. Za poruke koje pripadaju EPL protokolu rezervisana je vrijednost 0x88ab u okviru polja koje označava tip protokola (ET). Svaka EPL poruka sadrži zaglavlje (EPLH) u kojem se uvijek nalaze polja koja definišu tip poruke (MT), izvornu (EPLSRC) i odredišnu (EPLDEST) EPL adresu, kao i dodatne informacije koje zavise od tipa poruke. EPL standard definiše različite tipove poruka



Slika 2.8: Struktura okvira EPL protokola

Tabela 2.2: Pregled tipova EPL poruka

MT	Naziv	Značenje
0x01	*Start of Cycle* (SoC)	Određuje početak novog komunikacionog ciklusa i sinhronizuje sve čvorove u mreži.
0x03	*PollRequest* (PReq)	Zahtijeva od upravljanog čvora slanje PRes okvira i dostavlja izlazne procesne podatke prozvanom čvoru.
0x04	*PollResponse* (PRes)	Odgovara na PReq okvir i dostavlja procesne podatke svim čvorovima u mreži.
0x05	*Start of Asynchronous* (SoA)	Određuje početak asinhrone faze i dodjeljuje asinhroni slot nekom od čvorova.
0x06	*Asynchronous Send* (ASnd)	Prenosi asinhrone podatke.
0x07	*Active Managing Node Indication* (AMNI)	Određuje aktivan upravljački čvor.
0x0d	*Asynchronous Invite* (AInv)	Dodjeljuje asinhroni slot nekom od čvorova.

čija je uloga da obezbijede komunikacione servise neophodne za razmjenu procesnih podataka, konfiguraciju i upravljanje sistemom, kao i razmjenu asinhronih poruka. Pregled naziva i značenja tipova poruka koje su definisane standardom je dat u tabeli 2.2.

Svakom EPL uređaju se prilikom konfiguracije mreže dodjeljuje jedinstven 8-bitni identifikator (*NodeID*) čija vrijednost predstavlja adresu uređaja u EPL mrežnom segmentu koja se specificira u EPLSRC i EPLDEST poljima EPL zaglavlja (slika 2.8). Pregled predefinisanih identifikatora EPL čvorova sa opisom je dat u tabeli 2.3.

Osnovni EPL komunikacioni ciklus je podijeljen u tri faze: izohronu, asinhronu i neaktivnu kao što je ilustrovano na slici 2.9. Vremenskim razdavajanjem sinhronih (cikličnih) i asinhronih poruka se postiže determinizam u mreži na način da asinhrone poruke ne mogu da ometaju odvijanje vremenski kritičnog saobraćaja koji čine sinhrone poruke.

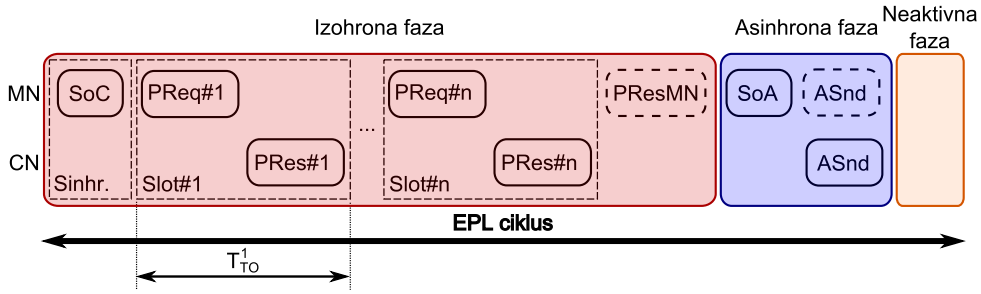
Na početku svakog ciklusa, upravljački čvor šalje višeodredišni (engl. *multicast*) SoC okvir svim upravljanim čvorovima kojim sinhronizuje čvorove i označava početak komunikacionog ciklusa i *izohrone faze*. Tokom izohrone faze razmjenjuju se ciklični procesni podaci između

Tabela 2.3: Pregled identifikatora čvorova

NodeID	Opis
0	Neispravna adresa
1–239	Upravljani čvorovi
240	Trenutno aktivan upravljački čvor
241–250	Redundantni upravljački čvorovi
251	Čvor adresira sam sebe
252	Virtuelni čvor (čvor koji ne obavlja nikakvu funkciju)
253	Dijagnostički uređaj
254	Uređaj za rutiranje standardnog *Ethernet* saobraćaja
255	Sveodredišna adresa

čvorova. U ovoj fazi, upravljački čvor proziva odgovarajući upravljani čvor tako što šalje jednoodredišni (engl. *unicast*) PReq okvir koji mu istovremeno dostavlja izlazne procesne podatke. Upravljani čvor adresiran PReq okvirom odgovara višeodredišnim PRes okvirom koji ujedno sadrži ulazne procesne podatke datog čvora. Kao što je ranije pomenuto, svakom upravljanom čvoru se dodjeljuje vremenski slot ograničenog trajanja (npr. T_{TO}^{1} za prvi čvor sa slike 2.9) koje je određeno vremenom potrebnim za slanje PReq i prijem PRes okvira datog čvora, kao i ukupnim vremenom propagacije ovih okvira kroz mrežu. Izohrona faza traje sve dok upravljački čvor ne prozove sve konfigurisane i aktivne upravljane čvorove u EPL mrežnom segmentu. Opcionalno, na kraju ove faze, upravljački čvor može da pošalje sopstvenu PRes poruku (PResMN) svim čvorovima.

Asinhrona faza služi za razmjenu neperiodičnih poruka koje se tipično šalju prilikom detekcije nekog događaja. Početak ove faze u okviru komunikacionog ciklusa, označava se slanjem višeodredišnog SoA okvira svim upravljanim čvorovima od strane upravljačkog čvora, koji odobrava pristup mreži samom sebi ili jednom upravljanom uređaju. U okviru asinhrone faze, može



Slika 2.9: EPL komunikacioni ciklus

se poslati jedna asinhrona poruka (ASnd) ili standardni *Ethernet* okvir u koji je enkapsulirana poruka nekog protokola sa višeg sloja komunikacionog modela (npr. TCP/IP datagram). Čvor kojem se odobrava slanje asinhrone poruke označen je posebnim poljem u okviru SoA poruke koje se naziva *odredište zahtjevanog servisa* (engl. *RequestedServiceTarget*). Ovdje je važno napomenuti da posljednja verzija EPL standarda [68] omogućava slanje više asinhronih poruka unutar asinhrone faze koje se inicira slanjem specifično definisane poruke pod nazivom AInv (engl. *Asynchronous Invite*).

Neaktivnu fazu čini preostali vremenski interval između kraja asinhrone faze i početka novog komunikacionog ciklusa. Tokom ovog intervala nema komunikacije i svi mrežni uređaji čekaju naredni ciklus. S obzirom da je trajanje komunikacionog ciklusa definisano prilikom konfiguracije mreže i ne mijenja se tokom rada sistema, vrijeme trajanja neaktivne faze obično varira od ciklusa do ciklusa i tipično zavisi od trajanja asinhrone faze (preciznije, od dužine asinhrone poruke koja se razmjenjuje u ovoj fazi).

Jedan od najznačajnijih parametara koji određuje efikasnost komunikacije u EPL sistemu je *vrijeme odziva* EPL čvora (engl. *node response time*). Ovo vrijeme se definiše kao vrijeme potrebno da EPL čvor (CN ili MN) obradi i odgovori na primljeni zahtijev (PReq ili PRes okvir).

U slučaju upravljanog čvora, vrijeme odziva (T_R^{CN}) je određeno intervalom od prijema PReq okvira (ili SoA okvira u slučaju asinhrone poruke) do slanja odgovarajućeg PRes (ASnd) okvira. Vrijeme odziva upravljanog čvora se može smanjiti ako se slanje PRes okvira realizuje hardverski. To se postiže korišćenjem specifičnog EPL mehanizma sa automatskim slanjem odgovora koji se naziva *auto-response* mehanizam. U osnovi, hardver mrežnog interfejsa se projektuje tako da sadrži skup filtera koji omogućavaju detekciju EPL okvira na koje je potrebno odgovoriti sa minimalnim kašnjenjem. Svaki filter definiše masku za označavanje relevatnih polja u okviru, i vrijednosti koje se porede sa stvarnim vrijednostima polja u primljenom okviru definisanim maskom. Okviri čije se vrijednosti relevantnih polja poklapaju sa vrijednostima definisanim filterom iniciraju slanje okvira kojim čvor treba da odgovori na zahtijev. Minimalno kašnjenje odgovora koje se na ovaj način može postići odgovara minimalnom rastojanju između *Ethernet* okvira.

Kod upravljačkog uređaja, vrijeme odziva (T_R^{MN}) se definiše kao vrijeme koje protekne od prijema PRes okvira prethodno prozvanog upravljanog čvora do slanja PReq okvira kojim se proziva naredni upravljani čvor. Dominantno kašnjenje u reakciji upravljanog čvora čini vrijeme opsluživanja prekida koji se generiše prilikom prijema okvira. Kako bi se ovo vrijeme eliminisalo i na taj način smanjilo vrijeme odziva upravljačkog čvora, prijemna i predajna strana upravljačkog čvora se razdvajaju tako što se slanje PReq okvira vremenski inicira korišćenjem vremenskog

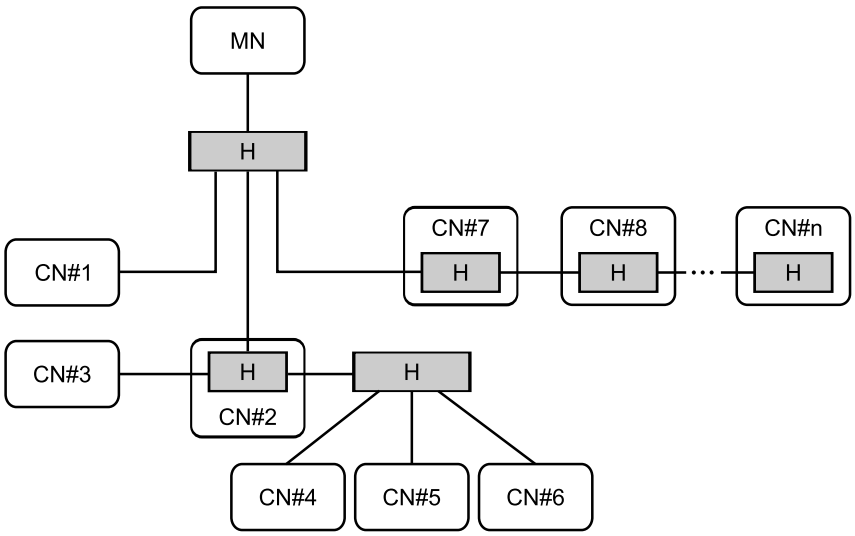
brojača visoke rezolucije, HRT (engl. *High-Resolution Timer*), ili preciznije, sljedeći PReq okvir se šalje nakon isteka predefinisanog vremenskog intervala koji odgovara trajanju vremenskog slota datog čvora. Pažljivim izborom trajanja vremenskih slotova, vrijeme odziva upravljačkog čvora može se svesti na vrijednost oko 1 μs.

2.2.1 Osnovne topologije EPL mreže

Kao što je ranije pomenuto, EPL standard preporučuje korišćenje koncentratora umjesto mrežnih komutatora kako bi se smanjilo kašnjenje poruka kroz infrastrukturne komponente i povećao determinizam komunikacije. Iako se mrežni komutatori mogu koristiti za realizaciju EPL mreže, oni unose znatno veća kašnjenja sa značajnom varijacijom (džiterom), što onemogućava preciznu sinhronizaciju čvorova. Tipično kašnjenje poruka kroz koncentrator je oko 500 ns sa maksimalnim apsolutnim odstupanjem od 10% (50 ns). S druge strane, kašnjenje mrežnih komutatora je oko 5 μs u slučaju direktnih ili CT (engl. *cut-through*) komutatora, odnosno oko 10 μs u slučaju komutatora sa pohranjivanjem prije prosljeđivanja ili SF (engl. *store-and-forward*) komutatora. Apsolutno odstupanje kašnjenja kod mrežnih komutatora, tipično se kreće oko 1 μs. Međutim, u slučaju SF komutatora, ovo kašnjenje dodatno zavisi od dužine okvira i može da varira značajno iznad 1 μs.

EPL protokol podržava sve osnovne topologije koje se koriste u sistemima automatizacije kao i bilo koju njihovu kombinaciju. Ovo podrazumijeva linijsku topologiju, topologiju zvijezde i stabla, kao i topologiju prstena koja se realizuje korišćenjem specijalnih uređaja za obezbjeđenje redundantnih veza u mreži pod nazivom *selektori veze* ili LS (engl. *link selector*). Za realizaciju topologije zvijezde, direktno se može koristiti višeportni koncentrator. Linijska i topologija stabla se realizuju tako što se u čvorove integriše koncentrator sa dva ili više protova. Primjer jedne karakteristične EPL topologije je dat na slici 2.10.

Kao što može da se vidi sa slike, linijska i topologija stabla se realizuju kaskadnim vezivanjem koncentratora koji su integrisani u same uređaje. Važno je napomenuti da je maksimalan broj kaskadno vezanih koncentratora ograničen apsolutnim odstupanjem kašnjenja korišćenih koncentratora i zahtijevanim nivoom preciznosti sinhronizacije. U slučaju tipičnog odstupanja kašnjenja od 50 ns i zahtijevanog nivoa sinhronizacije čvorova ispod 1 μs, ovaj broj mora da bude manji od 20.



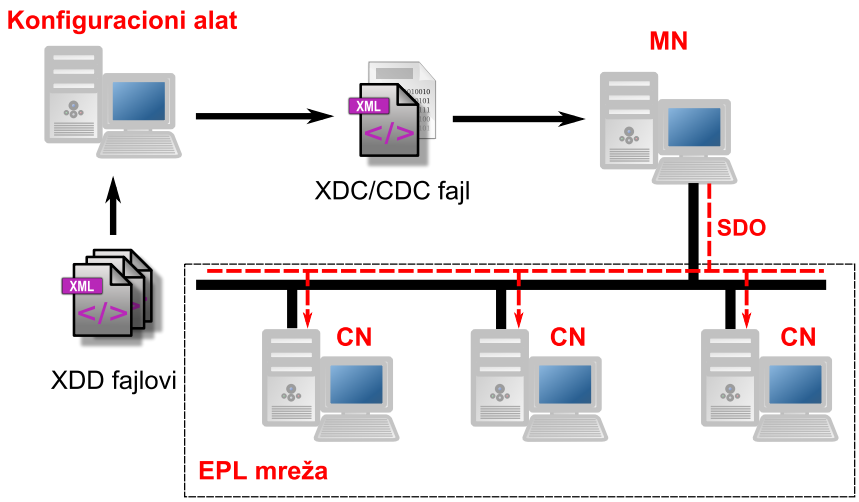
Slika 2.10: Primjer topologije EPL mreže

2.2.2 Konfiguracija EPL mreže

Slično kao kod ECAT sistema, EPL mreža mora da se konfiguriše prije dovođenja u operativno stanje. Postupak kreiranja konfiguracionog fajla obično se obavlja nezavisno od sistema, korišćenjem specifično definisanih fajlova za opis čvorova i odgovarajućeg alata za konfiguraciju. Tipičan tok konfiguracije EPL mreže prikazan je na slici 2.11.

Karakteristike svakog EPL čvora opisuju se u okviru elektronske specifikacije uređaja koju proizvođač dostavlja krajnjem korisniku u vidu XML fajla pod nazivom XDD (engl. *XML Device Description*) [69]. Alat za konfiguraciju (obično standardni PC računar) učitava ove fajlove i daje korisniku mogućnost da definiše parametre neophodne za ispravan rad mreže. Ovo tipično podrazumijeva podešavanje različitih komunikacionih parametara, kao i definisanje načina mapiranja procesnih podataka koji se dostavljaju PReq i PRes okvirima, sa lokalnim promjenljivim definisanim unutar aplikacionog procesa čvora.

Kao rezultat podešavanja parametara čvorova, alat za konfiguraciju mreže generiše izlazni XML fajl pod nazivom XDC (engl. *XML Device Configuration*) koji sadrži konkretne vrijednosti konfiguracionih parametara. Ovaj fajl se dostavlja upravljačkom uređaju kako bi inicijalizovao upravljane čvorove i doveo mrežu u operativno stanje. Upravljani čvor prenosi parametre tokom asinhrone faze komunikacionog ciklusa, korišćenjem SDO komunikacionog mehanizma naslijeđenog iz *CANopen* protokola. Važno je napomenuti da se, ukoliko je upravljani čvor implementiran na platformi sa ograničenim resursima, umjesto XDC fajla koristi jednostavniji binarni CDC (engl. *Concise Device Configuration*) fajl koji ima manju veličinu.

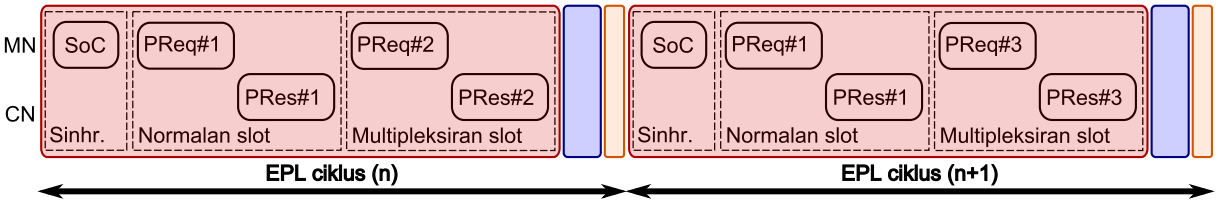


Slika 2.11: Konfiguracija EPL mreže

2.2.3 Multipleksirani način rada

Osim prethodno opisanog koncepta kontinualnog prozivanja svih upravljanih čvorova, EPL protokol podržava i tzv. *multipleksirani način rada*. U ovom načinu rada dva ili više uređaja dijele isti vremenski slot pri čemu se različiti uređaji prozivaju u različitim komunikacionim ciklusima kao što je ilustrovano na slici 2.12.

Sa slike je uočljivo da se razmjena podataka između multipleksiranih čvorova (čvorovi 2 i 3) odvija u dva osnovna komunikaciona ciklusa, što znači da je efektivno trajanje komunikacionog ciklusa ovih čvorova jednako dvostrukoj vrijednosti osnovnog komunikacionog ciklusa (npr. ukoliko je trajanje osnovnog ciklusa 1 ms, ciklus razmjene podataka multipleksiranih čvorova je 2 ms). U opštem slučaju, komunikacioni ciklus multipleksiranih čvorova jednak je umnošku trajanja osnovnog ciklusa brojem čvorova koji dijele isti vremenski slot. Na ovaj način se smanjuje broj prozivanja čvorova čime se skraćuje izohrona faza. Važno je napomenuti da je opisani pristup moguć samo ukoliko u mreži postoje čvorovi čiji je zahtijevani period razmjene podataka nekoliko puta duži od perioda osnovnog ciklusa, što zavisi od konkretne primjene.



Slika 2.12: EPL komunikacioni ciklus sa multipleksiranim načinom rada

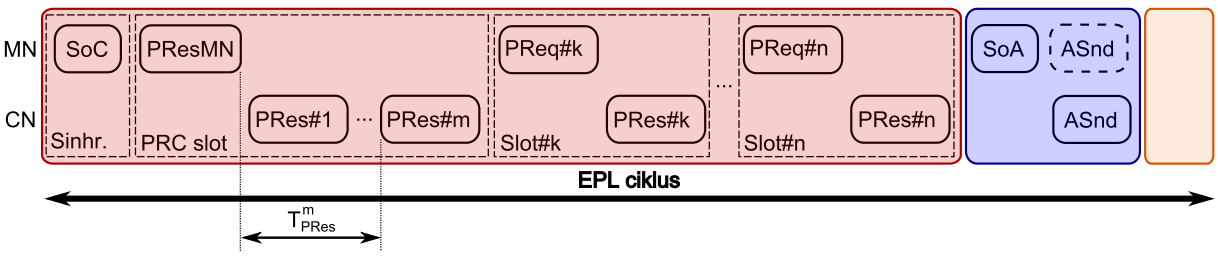
2.2.4 Način rada sa ulančavanjem PRes okvira

Prethodno opisani mehanizam prozivanja čvorova nije efikasan po pitanju iskorišćenja propusnog opsega, naročito u slučaju linijske topologije i uređaja čiji su podaci nekoliko bajtova, što je tipičan slučaj u sistemima automatizacije. Stoga je u okviru EPL standarda predloženo proširenje standardnog načina rada tzv. *mehanizmom sa ulančavanjem PRes okvira* ili kraće PRC (engl. *PollResponse Chaining*) [70].

Kod PRC načina rada, PReq okviri se izostavljaju jer se slanje PRes okvira u čvorovima inicira internim vremenskim brojačima koji se pokreću nakon prijema PResMN okvira. Ovaj okvir šalje upravljački uređaj na početku izohrone faze odmah nakon SoC okvira. Izlazni podaci, koji se standardno dostavljaju PReq okvirima, sada se mapiraju u PResMN okvir. Standardni i ulančani čvorovi mogu da se nalaze u istom mrežnom segmentu s tim da se razmjena podataka odvija u odvojenim dijelovima izohrone faze, što je ilustrovano na slici 2.13. Sa slike se vidi da svi upravljani čvorovi koji podržavaju PRC način rada razmjenjuju podatake u zasebnom slotu pod nazivom *PRC slot*. Ukupno trajanje PRC slota definiše se parametrom koji se u EPL terminologiji naziva *PResMNTimeout*.

Da bi se izbjegle kolizije okvira, upravljački čvor konfiguriše svaki upravljani čvor tokom inicijalizacije sistema tako da se PRes okvir konfigurisanog čvora šalje u vremenskom trenutku koji je određen T_{PRes} intervalom. Ovaj interval odgovara vremenu odziva čvora sa konfigurabilnim kašnjenjem, odnosno definiše vrijeme koje upravljani čvor treba da čeka nakon prijema PResMN okvira prije nego što inicira slanje sopstvenog PRes okvira. U primjeru na slici 2.13, ovaj koncept je ilustrovan za m-ti čvor. Kao što može da se zaključi sa slike, vrijeme odziva i-tog čvora, T^i_{PRes}, određeno je zbirom vremena potrebnog sa slanje svih prethodnih PRes okvira, kao i vremena propagacije ovih okvira kroz mrežu. Vrijeme odziva u PRC načinu rada se uvijek određuje u odnosu na kraj PResMN okvira.

Čvorovi se konfigurišu tako da se ulančani PRes okviri šalju sa minimalnim rastojanjem koje odgovara IFG intervalu. U tu svrhu je potrebno modifikovati prethodno opisani *auto-response* mehanizam tako da podrži automatsko slanje odgovora sa zadrškom (*auto-response*



Slika 2.13: EPL komunikacioni ciklus sa ulačavanjem PRes okvira

delay mehanizam). Kašnjenje pri slanju odgovora može se podešavati i odgovara prethodno definisanom T_{PRes} parametru.

Da bi upravljački čvor mogao da generiše vremenski plan slanja PRes okvira, potrebno je da ima informaciju o vremenima propagacije okvira između svih čvorova koji rade u PRC režimu. Propagaciona kašnjenja između upravljanih čvorova mjere se automatski tokom inicijalizacije sistema. U osnovi, upravljački čvor šalje specijalni SoA okvir pod nazivom *SyncReq* svakom od ulančanih čvorova u mreži. Čvor čija je adresa specificirana u *RequestedServiceTarget* polju ovog okvira odmah odgovara (korišćenjem *auto-response* mehanizma) specifično definisanim ASnd okvirom pod nazivom *SyncRes*. S obzirom da se oba okvira (*SyncReq* i *SyncRes*) šalju kao višeodredišni okviri, svi ulančani čvorovi primaju ove okvire i registruju vremena kada su oni primljeni. Nakon toga, svaki čvor izračunava razliku između vremena prijema *SyncReq* i *SyncRes* okvira za određeni upravljani čvor. Kada se od izračunate razlike oduzme vrijeme potrebno za slanje *SyncReq* okvira, dobija se *SyncDelay* vrijednost koja odgovara propagacionom kašnjenju između čvora koji je adresiran *SyncReq* okvirom i čvora koji izračunava vrijeme propagacije. Svaki ulančani čvor dostavlja izračunate *SyncDelay* vrijednosti upravljačkom čvoru korišćenjem zasebnog *SyncRes* okvira. Na osnovu izračunatih vremena propagacije između čvorova, upravljački čvor može da odredi vrijeme slanja PRes okvira za svaki ulančani čvor u mreži. Ovo vrijeme se svakom čvoru dostavlja korišćenjem dodatnog *SyncReq* okvira. Uzimajući u obzir prethodno rečeno, može se zaključiti da svaki ulančani čvor mora da razmijeni sa upravljačkim čvorom najmanje 3 para *SyncReq–SyncRes* okvira (ukupno 6 okvira) kako bi se realizovao proces mjerenja propagacionih kašnjenja (jedan par za mjerenje *SyncDelay* parametra, jedan da bi se izmjerena vrijednost dostavila upravljačkom čvoru, i jedan za dostavljanje izračunatog vremena slanja PRes okvira ulančanom čvoru).

2.3 Osnove Profinet IO protokola

Profinet IO (PNIO) je industrijski komunikacioni sistem zasnovan na *Ethernet* protokolu koji omogućava razmjenu vremenski kritičnih podataka u realnom vremenu (tzv. *Real-Time* ili RT komunikacija) paralelno sa saobraćajem koji nije vremenski kritičan (tzv. *Non Real-Time* ili NRT komunikacija). Ovo se postiže razdvajanjem različitih vrsta saobraćaja u vremenski izolovane intervale slijedeći koncept višestrukog pristupa sa dijeljenjem u vremenu ili TDMA (engl. *Time Division Multiple Access*), čime se dio propusnog opsega rezerviše za RT poruke (tzv. kanal za prenos podataka u realnom vremenu ili *RT kanal*) što smanjuje mogućnost da one budu zakašnjene zbog slanja NRT poruka. Poruke koje nisu vremenski kritične prenose se u

okviru tzv. *standardnog kanala*. Ovaj kanal tipično se koristi za konfiguraciju i parametrizaciju sistema korišćenjem UDP/IP protokola.

PNIO mreža se sastoji iz krajnjih uređaja koji su povezani korišćenjem infrastrukturnih komponenata, najčešće mrežnih komutatora. U zavisnosti od zahtjevanog nivoa determinizma, mogu da se koriste standardni mrežni komutatori sa prioritetizacijom saobraćaja u skladu sa IEEE 802.1Q standardom [71], odnosno specijalni mrežni komutatori koji omogućavaju sinhronizaciju čvorova i minimizaciju vremena zadržavanja poruka u komutatoru primjenom vremenskog planiranja slanja poruka, čime se postiže tzv. *izohrona* komunikacija u realnom vremenu ili IRT (engl. *Isochronous Real-Time*) komunikacija.

Standardna topologija u mrežama zasnovanim na *Ethernet* komunikaciji sa mrežnim komutatorima je topologija zvijezde, međutim, kod PNIO sistema mogu se realizovati i druge topologije uobičajene u sistemima automatizacije, kao što su linijska, te topologije stabla i prstena. Ove topologije najčešće se realizuju kaskadnim povezivanjem mrežnih komutatora koji se obično integrišu u same krajnje uređaje. Važno je napomenuti da je broj mrežnih komutatora koji se mogu kaskadno vezati ograničen preciznošću vremenske sinhronizacije koja se zahtijeva u sistemu.

Uzimajući u obzir skalabilnost sistema, kao i nivo determinizma koji je potrebno obezbijediti, PNIO definiše različite klase RT saobraćaja. Komunikacija u različitim klasama može da bude sinhronizovana ili nesinhronizovana, kao i da nudi različite nivoe fleksibilnosti sistema. Poruke koje pripadaju bilo kojoj klasi RT saobraćaja uvijek imaju viši prioritet u odnosu na UDP/IP saobraćaj. PNIO poruke je moguće identifikovati inspekcijom polja za označavanje protokola unutar *Ethernet* okvira (za PNIO protokol je rezervisana vrijednost 0x8892).

RTC1 (engl. *Real-Time Class 1*) klasa saobraćaja predstavlja nesinhronizovanu komunikaciju u realnom vremenu unutar jednog mrežnog segmenta. Za adresiranje se koriste standardne MAC adrese definisane *Ethernet* protokolom, što znači da se za ovu klasu saobraćaja mogu koristiti standardni mrežni komutatori. Determinizam komunikacije se može poboljšati ako se koriste mrežni komutatori sa VLAN (engl. *Virtual Local Area Network*) opcijom koja omogućava prioritetizaciju saobraćaja prema prethodno pomenutom IEEE 802.1Q standardu. S obzirom da uređaji nisu sinhronizovani, ne postoji vremenska relacija između početka komunikacionog ciklusa u uređajima. Ovo znači da slanje RT poruke u mrežnom komutatoru može da bude zakašnjeno zbog toga što je odredišni port zauzet slanjem neke druge RT poruke ili NRT poruke. Ova klasa saobraćaja nudi najviši nivo fleksibilnosti sistema na uštrb determinizma komunikacije koji se može postići. Tipično vrijeme komunikacionog ciklusa kreće se oko 100 ms sa maksimalnim odstupanjem od 15%. Pažljivim planiranjem saobraćaja i konfiguracijom

mreže može se postići vrijeme komunikacionog ciklusa čiji je red veličine 10 ms.

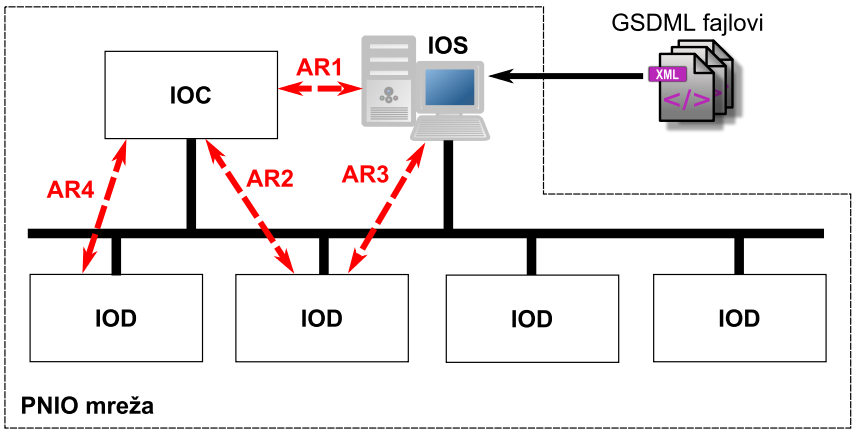
RTC2 (engl. *Real-Time Class 2*) klasa saobraćaja podržava sinhronizovanu i nesinhronizovanu komunikaciju. Nesinhronizovana komunikacija u potpusnosti odgovara RTC1 klasi saobraćaja. Kod sinhronizovane komunikacije, svi čvorovi koji učestvuju u komunikaciji dijele istu vremensku bazu čime se omogućava veoma precizno označavanje početka komunikacionog ciklusa. Ovim se sprečava da NRT poruke odgode slanje RT poruka u mrežnim komutatorima, jer je slanje poruka koje pripadaju različitim klasama saobraćaja odvojeno u vremenu. Međutim, i dalje postoji mogućnost da RT poruke koje dijele isti odredišni port međusobno interferiraju i time uzrokuju zadržavanje jedne od poruka u mrežnom komutatoru. Sve infrastrukturne komponente, kao i svi krajnji uređaji, moraju da podržavaju mehanizme za sinhronizaciju lokalnog vremena. I u ovoj klasi saobraćaja prosljeđivanje okvira bazira se na standardnim MAC adresama. Tipično vrijeme komunikacionog ciklusa je oko 1 ms sa maksimalnim odstupanjem koje je manje od 1 μs.

RTC3 (engl. *Real-Time Class 3*) klasa saobraćaja obezbjeđuje najviši nivo determinizma tako što se slanje poruka u svim uređajima u mreži, uključujući i infrastrukturne komponente, precizno planira prilikom konfiguracije mreže. Optimizovani vremenski plan slanja poruka dostavlja se svim uređajima korišćenjem odgovarajućih PNIO servisa. Da bi se ovo moglo postići, u mreži moraju da se koriste specijalni mrežni komutatori koji obezbjeđuju visok nivo preciznosti sinhronizacije čvorova, kao i prosljeđivanje poruka na osnovu kreiranog vremenskog plana, a ne na osnovu odredišne adrese poruke, čime se dodatno smanjuje vrijeme zadržavanja poruka u komutatorima. Nedostatak opisanog pristupa ogleda se u tome što je za optimizaciju vremenskog plana potrebno poznavati topologiju mreže. Prema tome, ovaj pristup je manje fleksibilan u odnosu na prethodne klase saobraćaja, jer promjena strukture sistema (bilo da se radi o promjeni topologije ili skupa poruka koje se periodično razmjenjuju) zahtijeva kreiranje novog vremenskog plana. Ova klasa saobraćaja se koristi kada je potrebno postići vremena ciklusa značajno ispod 1 ms sa odstupanjem koje je mnogo manje od 1 μs.

2.3.1 Vrste uređaja u PNIO mreži

PNIO specifikacija definiše tri vrste uređaja (slika 2.14): PNIO kontroler (engl. *IO-Controller* – IOC), PNIO uređaj u pogonu (engl. *IO-Device* – IOD) i nadzorni PNIO uređaj (engl. *IO-Supervisor* IOS).

PNIO kontroler je obično PLC (engl. *Programmable Logic Controller*) na kojem se izvršavaju zadaci neophodni za automatizaciju procesa. Svaka PNIO konfiguracija sadrži bar jedan



Slika 2.14: Vrste PNIO uređaja i konfiguracija mreže

IOC uređaj.

PNIO uređaj u pogonu je distribuirani uređaj sa kojim su direktno povezani senzori i/ili aktuatori za upravljanje procesom. Ovaj uređaj razmjenjuje podatke sa jednim ili više IOC uređaja korišćenjem odgovarajućih PNIO mehanizama komunikacije. Svaka PNIO konfiguracija sadrži bar jedan IOD uređaj.

Nadzorni PNIO uređaj služi za konfiguraciju i dijagnostiku rada svih uređaja u mreži. To je obično standardni PC računar ili HMI (engl. *Human Machine Interface*) uređaj. Najčešće se povezuje privremeno u mrežu (npr. PC računar) prilikom njene konfiguracije i pokretanja, odnosno u svrhu dijagnostikovanja kvarova. Međutim, može i trajno da bude povezan sa mrežom (npr. HMI uređaj) kada je potrebno kontinualno nadzirati proces upravljanja i definisati različite parametre tokom rada sistema.

Kao što je pomenuto, konfiguraciju PNIO mreže obavlja IOS uređaj koji u tu svrhu koristi informacije iz specifično definisanih XML fajlova pod nazivom GSDML (engl. *General Station Description Markup Language*). Slično kao u slučaju XDD fajlova kod EPL sistema, GSDML fajlove distribuiraju proizvođači uz sam uređaj i oni mogu da se smatraju specifikacijom uređaja u elektronskom obliku. U osnovi, GSDML fajl opisuje karakteristike uređaja i sadrži informacije o parametrima (kao što su vrijeme zadržavanja poruke u uređaju, veličina procesnih podataka itd.) koji su IOS uređaju neophodni za konfiguraciju mreže i pripremanje vremenskog plana slanja poruka koje se razmjenjuju u realnom vremenu. Konfiguracioni fajl, koji generiše IOS uređaj, učitava se u IOC uređaj pomoću odgovarajućih servisa PNIO protokola. IOC uređaj koristi informacije iz konfiguracionog fajla za inicijalizaciju mreže i podešavanje parametara svakog pojedinačnog IOD uređaja. Kao što će biti detaljnije objašnjeno u nastavku ovog poglavlja, to podrazumijeva uspostavljanje različitih relacija između aplikacija pojedinih uređaja, AR (engl. *Application Relation*), koje tipično sadrže više komunikacionih relacija, CR (engl.

Communication Relation). Svaka komunikaciona relacija podrazumijeva virtuelni kanal za komunikaciju između uređaja sa zahtjevanim nivoom kvaliteta usluge.

2.3.2 Model PNIO uređaja

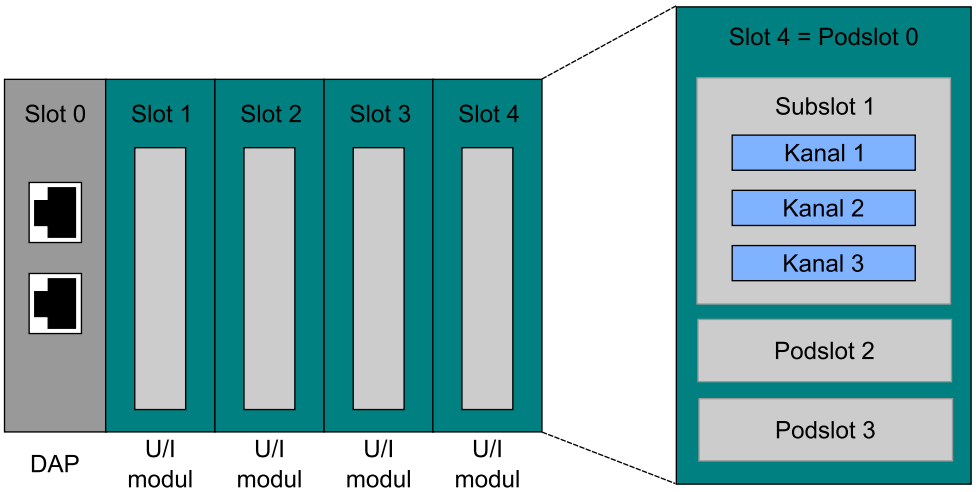
PNIO uređaji mogu da se realizuju kao modularni ili kao kompaktni. Obje vrste uređaja se opisuju kao uređaji koji se sastoje iz određenog broja modula s tom razlikom da kompaktni uređaji imaju fiksnu konfiguraciju, dok kod modularnih uređaja konfiguraciju može da definiše i mijenja krajnji korisnik. Model PNIO uređaja čine sljedeći elementi: slot (engl. *slot*), podslot (engl. *subslot*) i kanal (engl. *channel*) (slika 2.15). Za definisanje različitih parametara unutar slotova i podslotova, kojima se pristupa tipično korišćenjem asinhronih komunikacionih servisa, koriste se specifični elementi koji se nazivaju indeksi (engl. *Index*). Struktura i mogućnosti svakog PNIO uređaja je definisana unutar prethodno pomenutog GSDML fajla.

Slot opisuje strukturu komponenata ili funkcija, npr. hardverske module ili logičke jedinice, unutar PNIO uređaja. Slotovi su određeni brojevima u opsegu od 1 do 32767. Slot sa brojem 0 uvijek definiše karakteristike samog uređaja i rezervisan je za tzv. *pristupnu tačku uređaja* ili DAP (engl. *Device Access Point*) koja obezbjeđuje povezivanje uređaja na mrežu.

Slično kao slot, *podslot* opisuje strukturu komponenata ili funkcija unutar slota i takođe je određen brojevima u opsegu od 1 do 32767. Podslot sa brojem 0 uvijek adresira slot u kojem se nalazi. Takođe, mogu se definisati i podslotovi u opsegu do 32768 do 36863 za adresiranje specijalnih aplikacija. Svaki podslot može da ima jedan ili više kanala.

Kanal predstavlja stvarnu strukturu ulaznih i izlaznih (analognih ili digitalnih) procesnih podataka koji su vezani za fizičke tačke komunikacije sa senzorima i/ili aktuatorima.

Osim prethodno navedenih elemenata, svaki PNIO uređaj obezbjeđuje specijalan element



Slika 2.15: Struktura modularnog PNIO uređaja

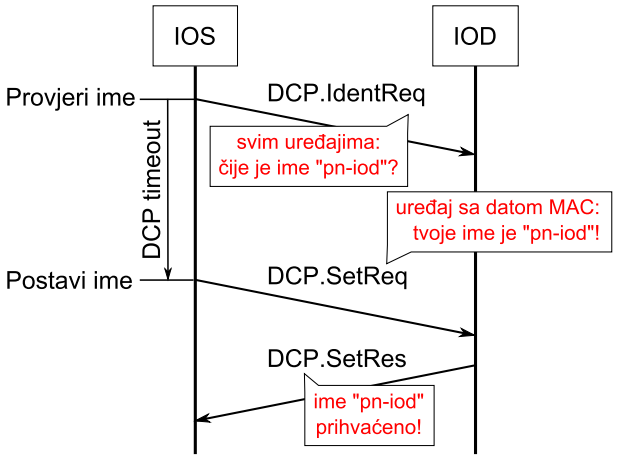
pod nazivom API (engl. *Application Programming Interface*) koji omogućava jednostavnu integraciju aplikacionih profila unutar uređaja. Na ovaj način, jedan uređaj može da podrži više aplikacionih profila korišćenjem različitih API elemenata.

2.3.3 Adresiranje PNIO uređaja

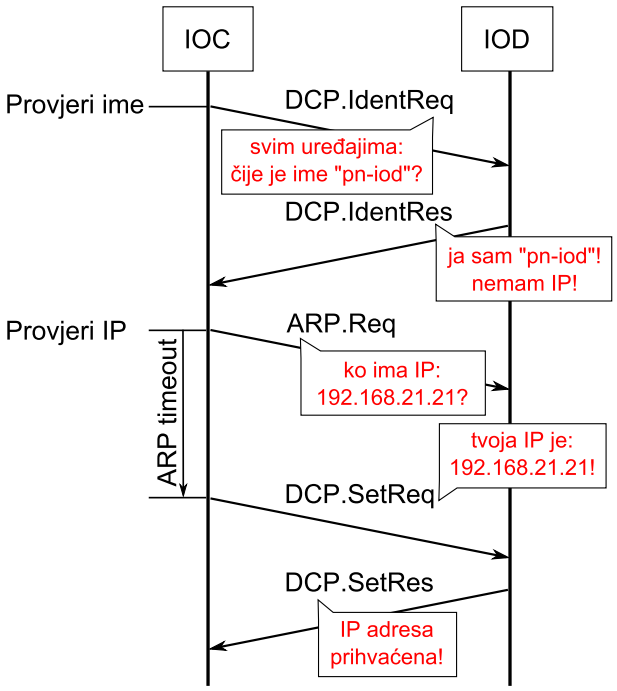
Prilikom konfiguracije PNIO sistema, svakom PNIO uređaju dodjeljuje se simboličko ime kojem se pridružuje jedinstvena fizička adresa mrežnog interfejsa (MAC adresa). Za pronalaženje i dodjeljivanje simboličkih imena uređajima, PNIO koristi posebne mehanizme definisane DCP (engl. *Discovery and Configuration Protocol*) protokolom. U primjeru na slici 2.16, prikazana je sekvenca DCP poruka koja ilustruje proces identifikacije i dodjele simboličkog imena *pnio-id* uređaju sa definisanom MAC adresom.

S obzirom da PNIO za parametrizaciju i konfigurisanje IOD uređaja koristi UDP/IP poruke, svakom uređaju mora da se dodijeli jedinstvena IP adresa unutar mrežnog segmenta. Ovu adresu dodjeljuje IOC uređaj prilikom inicijalizacije sistema tako što identifikuje uređaje na osnovu njihovog simboličkog imena, a zatim im pridružuje sljedeću slobodnu IP adresu iz definisanog opsega. Kao i u slučaju dodjeljivanja simboličkog imena, za definisanje IP adrese mogu se koristiti DCP mehanizmi kao što je ilustrovano dijagramom sekvenci na slici 2.17. U primjeru sa slike, uređaju kojem je prethodno dodjeljeno simboličko ime *pnio-iod* dodjeljuje se IP adresa 192.168.21.21. Prije dodjeljivanja IP adrese, mreži se šalje ARP (engl. *Address Resolution Protocol*) zahtjev koji ispituje da li je navedena IP adresa zauzeta.

Alternativno, za dodjeljivanje i razrješavanje IP adresa u PNIO mreži, može se koristiti i DHCP (engl. *Dynamic Host Configuration Protocol*) server, koji predstavlja konvencionalan pristup naslijeđen iz standardnih računarskih mreža.



Slika 2.16: Sekvenca DCP poruka za dodjelu simboličkog imena PNIO uređaju



Slika 2.17: Sekvenca DCP poruka za dodjelu IP adrese PNIO uređaju

2.3.4 PNIO komunikacioni servisi

Kako bi mogao da obezbijedi različite vrste servisa, koji su neophodni za realizaciju komunikacije između komponenata sistema automatizacije različitog nivoa kompleksnosti, PNIO definiše nekoliko vrsta objekata koji se mogu grupisati u tri cjeline: objekti ulazno/izlaznih podataka (engl. *IO-Data Objects* – IODO), objekti zapisa (engl. *Record Data Objects* – RDO) i objekti alarma (engl. *Alarm Data Objects* – ADO).

Objekti ulazno/izlaznih podataka predstavljaju ulazno/izlazne procesne podatke koji se razmjenjuju ciklično (periodično) u realnom vremenu u okviru RT kanala. Svaki objekat ovog tipa je jednoznačno određen specificiranjem uređaja, slota i podslota. Podaci se razmjenjuju po modelu *davalac-potrošač* (engl. *provider-consumer*) bez potvrde prijema podataka. Interval koji određuje period razmjene podataka definiše se prilikom konfiguracije mreže, a komunikacija se kontinualno nadzire tokom rada sistema.

Objekti zapisa se prenose u vidu neperiodičnih (asinhronih) poruka sa podacima koji se odnose na parametrizaciju i konfiguraciju sistema, te statusne i dijagnostičke informacije. Ovi podaci se prenose u okviru standardnog kanala u kojem se prenose poruke koje nisu vremenski kritične. S obzirom da se komunikacija u standardnom kanalu obavlja bez garantovanog nivoa usluge (engl. *best-effort*), za prenos ovih podataka se koriste protokoli iz standardnih računarskih mreža (tipično UDP/IP).

Objekti alarma se odnose na prenos asinhronih poruka u realnom vremenu koje se prenose

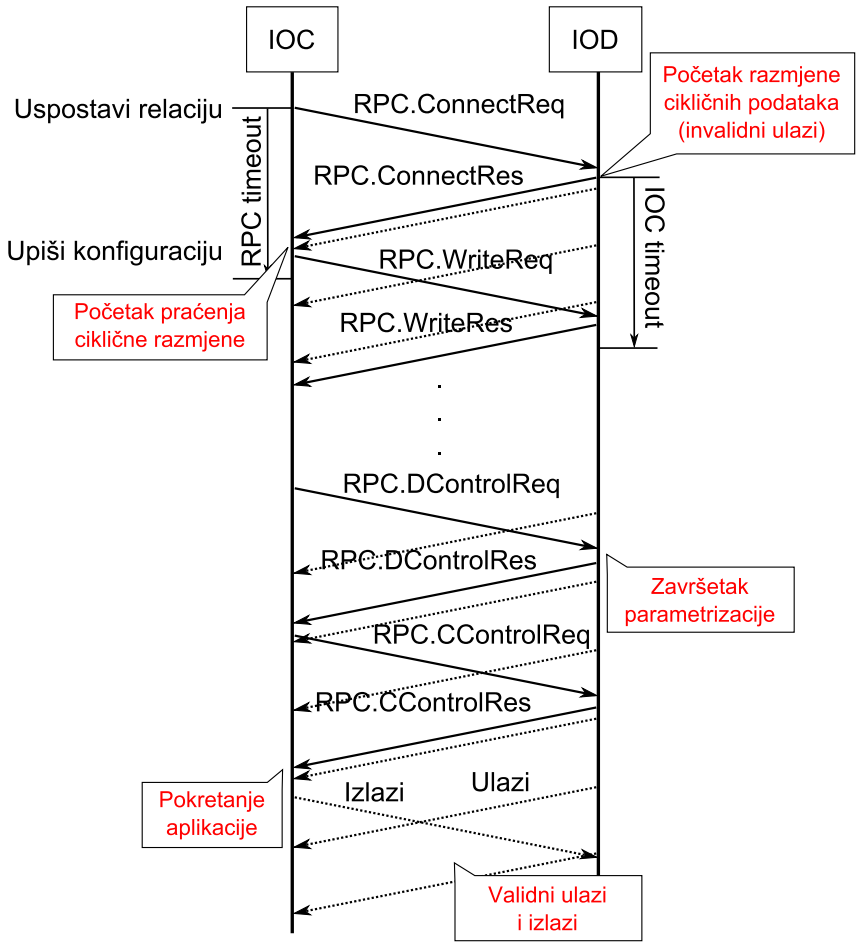
u okviru RT kanala. Prenos poruka se inicira različitim događajima u sistemu čija je uloga da alarmiraju aplikacije u uređajima i/ili službu održavanja sistema. Ovo može da se odnosi na događaje poput zamjene modula u uređaju, detekciju različitih kvarova i grešaka u radu sistema, kao i korisnički definisane alarme koji se odnose na proces upravljanja, kao što su detekcija prekoračenja definisanog nivoa tečnosti ili vrijednosti trenutne temperature.

Kako bi se omogućio prenos prethodno opisanih objekata, PNIO koristi različite servise za kreiranje AR relacija, koje se izvršavaju u PNIO uređajima. Svaka AR relacija, koja se uspostavlja između IOC, IOS i IOD uređaja, sadrži jednu ili više CR relacija, koje predstavljaju virtuelne komunikacione kanale između uređaja preko kojih se prenose prethodno opisani PNIO objekti (slika 2.18). U tom smislu, definišu se tri tipa CR relacija: za prenos ulazno/izlaznih podataka (engl. *IO-Data CR*), za prenos zapisa (engl. *Record Data CR*) i za prenos alarma (engl. *Alarm Data CR*). Postupak kreiranja i brisanja AR relacija, u PNIO terminologiji, naziva se *upravljanje kontekstom*, CM (engl. *Context Management*).

Upravljanje kontekstom, u PNIO sistemu, obezbjeđuje RPC (engl. *Remote Procedure Call*) protokol, čije se poruke prenose unutar UDP/IP datagrama u okviru standardnog kanala. Ovaj protokol definiše različite servise za uspostavljanje veze na nivou AR, kao i prenos različitih podataka neophodnih za parametrizaciju i konfiguraciju sistema. Pojednostavljen prikaz jedne sekvence za uspostavljanje virtuelnih kanala i parametrizaciju modula u uređaju prikazan je dijagramom na slici 2.19.

Ovaj dijagram predstavlja sekvencu poruka koje se, po pravilima RPC protokola, razmjenjuju između IOC i IOD uređaja. Proces uspostavljanja AR i CR relacija inicira IOC (ili IOS) uređaj slanjem okvira za uspostavljanje veze (*RPC.ConnectReq* okvir). Nakon prihvatanja ovog zahtjeva, IOD šalje pozitivnu potvrdu putem *RPC.ConnectRes* okvira i započinje proces cikličnog slanja ulaznih podataka koji su označeni kao invalidni, jer još uvijek nije uspostavljena relacija na nivou aplikacije. Na IOC strani, nakon prijema potvrde, počinje proces praćenja ciklične razmjene ulaznih podataka i inicira se parametrizacija (upis trenutno definisane konfiguracije) svih modula u PNIO uređaju. Parametrizacija se obavlja posredstvom *RPC.WriteReq* i *RPC.WriteRes* okvira koji se razmjenjuju između uređaja. Završetak parametrizacije na strani IOC uređaja označava se slanjem *RPC.DControlReq* okvira. Konačno, kraj parametrizacije na strani IOD

Slika 2.18: Ilustracija koncepta virtuelnih PNIO kanala (AR i CR)



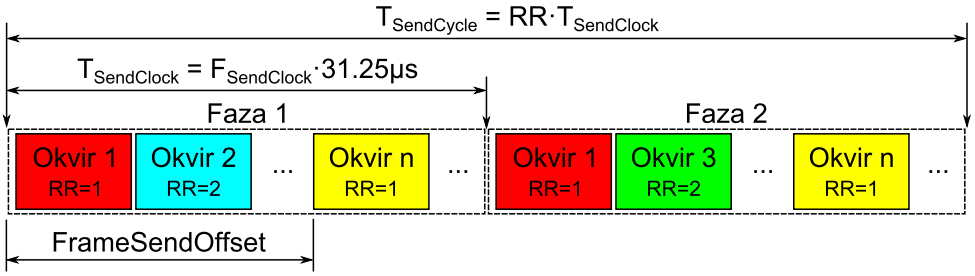
Slika 2.19: Sekvenca RPC poruka za uspostavljanje CR kanala i parametrizaciju uređaja

uređaja i označavanje uspostavljanja AR relacije definisano je razmjenom *RPC.CControlReq* i *RPC.CControlRes* okvira. Nakon toga, oba uređaja mogu da razmjenjuju validne ulazne i izlazne procesne podatke.

2.3.5 Organizacija PNIO komunikacionog ciklusa

Slično kao kod ECAT i EPL protokola, prenos cikličnih procesnih podataka u PNIO sistemu organizuje se unutar ponavljajućeg vremenskog perioda koji se naziva *komunikacioni ciklus*. S obzirom da uređaji mogu da razmjenjuju ciklične podatke sa različitim periodom, u cilju optimizacije iskorišćenja propusnog opsega, PNIO definiše nekoliko parametara koji određuju vremenske relacije poruka unutar komunikacionog ciklusa kao što je prikazano na slici 2.20.

Trajanje osnovnog perioda PNIO komunikacionog ciklusa ($T_{SendClock}$) određeno je cjelobrojnim umnoškom osnovnog vremenskog intervala od 31.25 μs i faktora komunikacionog ciklusa ($F_{SendClock}$). U slučaju sinhronizovane komunikacije, svakom okviru sa RT porukom, pridružuje se parametar koji definiše početak slanja okvira unutar komunikacionog ciklusa, FSO (engl. *FrameSendOffset*), čime se onemogućava interferencija okvira i obezbjeđuje efikasno is-



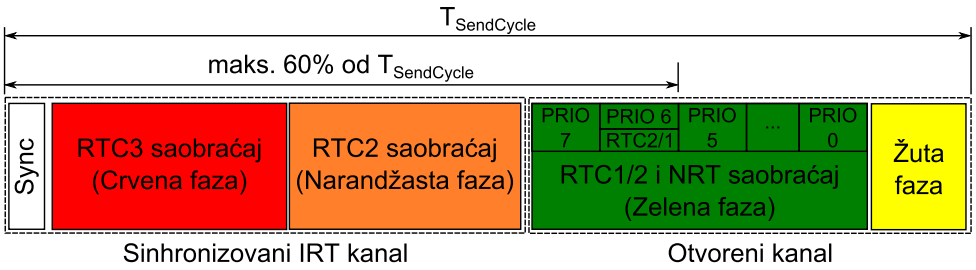
Slika 2.20: Ilustracija slanja periodičnih PNIO okvira

korišćenje propusnog opsega. Uređaji koji razmjenjuju podatke sa dužim periodom, dodatno optimizuju iskorišćenje propusnog opsega uz obezbjeđenje zahtjevane propusne moći tako što svoje okvire šalju sa predefinisanim *faktorom redukcije*, *RR* (engl. *Reduction Ratio*), koji odgovara umnošku osnovnog perioda komunikacionog ciklusa. Na taj način, ukupno vrijeme komunikacionog ciklusa ($T_{SendCycle}$) dijeli se u više faza, pri čemu je trajanje svake faze jednako osnovnom komunikacionom ciklusu. Ukupan broj faza odgovara najvećem faktoru redukcije koji je definisan u sistemu. U primjeru sa slike 2.20, ukupni PNIO ciklus čine dvije faze, jer je faktor redukcije uređaja koji šalju okvire 2 i 3 podešen na vrijednost 2.

Kao što je već pomenuto, osnovni komunikacioni ciklus je podijeljen u različite intervale unutar kojih se razmjenjuju poruke koje pripadaju različitim klasama saobraćaja. Izgled osnovnog PNIO komunikacionog ciklusa je prikazan na slici 2.21.

Kao što može da se vidi sa slike, početak svakog PNIO komunikacionog ciklusa je označen sinhronizacionom porukom (*Sync* interval) i sastoji se iz dva kanala: *sinhronizovanog IRT kanala* u kojem se razmjenjuju sinhronizovane RTC2/3 poruke, i *otvorenog kanala* u kojem se razmjenjuju asinhrone alarmne poruke (prioriteta 7 i 5), te nesinhronizovane RTC2/1 poruke (prioriteta 6) i NRT saobraćaj.

Tokom *crvenog intervala* razmjenjuju se samo poruke koje pripadaju RTC3 klasi saobraćaja (tzv. *IRTtop* klasa saobraćaja). Kao što je već pomenuto, ova klasa saobraćaja zahtijeva da raspored slanja poruka od strane umreženih čvorova bude unaprijed isplaniran za svaku poruku koja se prenosi po predefinisanoj putanji. Raspored slanja poruka zavisi od topologije mreže,



Slika 2.21: PNIO komunikacioni ciklus

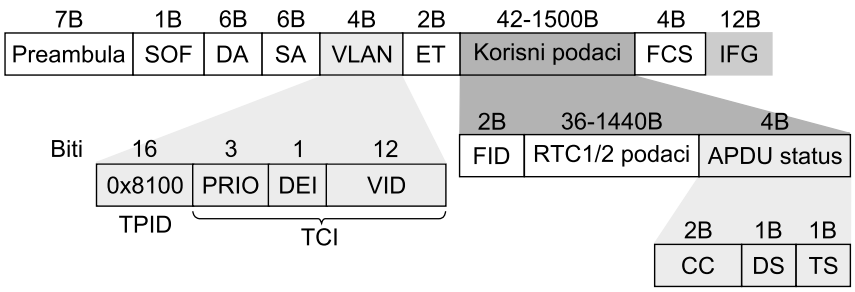
kao i od broja i veličine poruka koje se razmjenjuju i definiše se prilikom projektovanja sistema u okviru specijalnog alata za konfiguraciju mreže (obično je ova funkcionalnost smještena u IOS uređaju). Da bi se mogao kreirati optimalan plan slanja poruka, sva kašnjenja u putanji svakog okvira moraju biti unaprijed poznata i precizno određena. Takođe, svi uređaji u mrežnom segmentu moraju da budu precizno sinhronizovani kako bi akcije slanja poruka u svakom uređaju bile precizno koordinisane na nivou čitave mreže. Prosljeđivanje poruke se bazira na osnovu identifikatora PNIO okvira, o kojem će više riječi biti u nastavku poglavlja. Ukoliko se u ovom intervalu pojave poruke koje ne pripadaju IRT klasi saobraćaja, one se privremeno skladište u IRT mrežnom komutatoru i šalju se u okviru otvorenog kanala.

Narandžasti interval je rezervisan za saobraćaj iz sinhronizovane RTC2 klase (tzv. *IRTflex* klasa saobraćaja). Ova klasa saobraćaja predstavlja fleksibilniju klasu IRT komunikacije, jer nije neophodno kreiranje vremenskog plana koji zavisi od strukture mreže. Narandžasti interval počinje odmah nakon crvenog intervala. S obzirom da se poruke u ovom intervalu prosljeđuju na osnovu odredišne MAC adrese, iskorišćenje propusnog opsega nije optimizovano kao u slučaju prethodnog intervala, što se mora uzeti u obzir prilikom određivanja trajanja ovog intervala.

Zeleni interval se primarno koristi za PNIO saobraćaj koji pripada nesinhronizovanoj RTC1 klasi saobraćaja. Dodatno, u ovoj fazi se razmjenjuju alarmne asinhrone poruke koje se prenose u realnom vremenu, kao i NRT saobraćaj. Takođe, tokom ovog intervala mogu da se šalju poruke iz RTC2 klase ukoliko je to potrebno. Prioritet slanja poruka u mrežnom komutatoru je određen VLAN oznakom unutar okvira. Svi okviri koji pripadaju IRT komunikaciji se uništavaju ukoliko se pojave tokom ovog intervala.

Žuti interval predstavlja prelaznu oblast, koju koriste uređaji iz zelenog intervala, čiji se okviri mogu u potpunosti poslati do završetka komunikacionog ciklusa. Okviri čije bi slanje zakasnilo početak narednog ciklusa smještaju se u interne redove čekanja svakog komutatora i njihovo slanje se odgađa do početka zelenog intervala u sljedećem komunikacionom ciklusu.

Struktura PNIO okvira koji se koristi za prenos RT poruka iz RTC1/2 klasa saobraćaja prikazana je na slici 2.22. Kao što može da se vidi sa slike, kontrolne informacije protokola čine



Slika 2.22: Struktura PNIO RT okvira

Tabela 2.4: Pregled identifikatora PNIO okvira

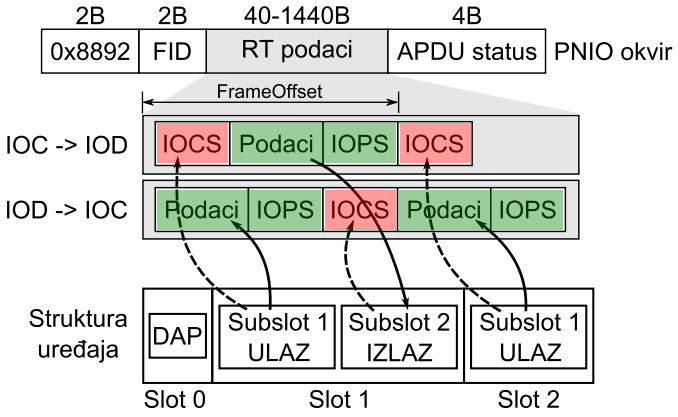
Opseg FID	Značenje
0x0000–0x00FF	Okviri za sinhronizaciju
0x0100–0x7FFF	RTC3 okviri
0x8000–0xBFFF	RTC2 okviri
0xC000–0xFBFF	RTC1 okviri
0xFC00–0xFCFF	Alarmi visokog prioriteta
0xFD00–0xFDFF	Rezervisani identifikatori
0xFE00–0xFEFC	Alarmi niskog prioriteta
0xFEFD–0xFEFF	DCP okviri
0xFF00–0xFFFF	Rezervisani identifikatori

dva bajta identifikatora okvira (FID) koji određuje tip poruke, i 4 bajta statusnih podataka: dva bajta za brojač ciklusa, CC (engl. *Cycle Counter*), jedan bajt za označavanje statusa podataka, DS (engl. *Data Status*), i jedan bajt za označavanje statusa prenosa, TS (engl. *Transfer Status*). Tipovi poruka definisani FID poljem, koje podržava PNIO protokol, prikazani su u tabeli 2.4.

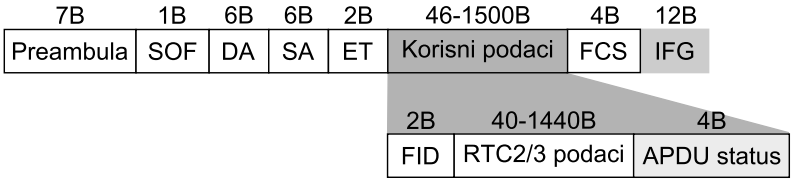
Kako bi se definisali prioriteti poruka u okviru zelene faze, *Ethernet* okviri u koje se enkapsuliraju PNIO poruke sadrže 32-bitne VLAN oznake koje se sastoje iz 16 bita fiksnog identifikatora tipa poruke, TPID (engl. *Tag Protocol Identifier*) i 16-bitnog TCI (engl. *Tag Control Information*) polja koje sadrži kontrolne informacije. TCI polje čine 3-bitno polje za definisanje prioriteta (PRIO), rezervisani DEI (engl. *Drop Eligible Indicator*) bit, te polje jedinstvenog identifikatora VLAN poruke koje se specificira u 12-bitnom polju VID (engl. *VLAN Identifier*).

Korisni podaci, unutar PNIO okvira, organizuju se u zavisnosti od strukture modularnog uređaja i njihova veličina zavisi od broja modula. Kako bi se obezbijedile dijagnostičke informacije na nivou modula, korisnim podacima svakog modula pridružuju se statusne informacije koje se prenose poljima IOPS (engl. *IO-Data Producer Status*) i IOCS (engl. *IO-Data Consumer Status*). Veličina ovih polja definisana je opisom uređaja u okviru GSDML fajla i obično iznosi jedan bajt. Ukoliko neki modul uređaja ne prenosi ciklične procesne podatke (npr. DAP), onda se za ovaj modul prenose samo IOPS i IOCS polja. Način mapiranja procesnih podataka i statusnih polja za jedan primjer IOD uređaja ilustrovan je na slici 2.23.

Parametar *FrameOffset* označava udaljenost početka polja podataka određenog modula u odnosu na početak polja korisnih podataka *Ethernet* okvira. Ovaj parametar se određuje prilikom konfiguracije mreže i dostavlja se IOD uređaj prilikom uspostavljanja virtuelnog kanala u



Slika 2.23: Primjer mapiranja procesnih podataka u PNIO okvir



Slika 2.24: Struktura PNIO IRT okvira

segmentu parametrizacije.

Kao što je ranije pomenuto, prosljeđivanje poruka kod IRT komunikacije bazirano je na vremenskom planu koje koristi FID polje za identifikaciju poruka. Prema tome, MAC adrese se ne korsite za identifikaciju uređaja i poruka, pa je VLAN označavanje suvišno. Slično, poruke koje pripadaju sinhronizovanoj RTC2 klasi saobraćaja mogu, ali ne moraju da sadrže VLAN oznake, jer je prosljeđivanje okvira bazirano na odredišnim MAC adresama. Struktura PNIO okvira za ove dvije klase saobraćaja prikazana je na slici 2.24.

2.4 Zaključak

U ovom poglavlju opisana su tri predstavnika RTE sistema iz klase protokola sa najvišim performansama: *EtherCAT*, *Ethernet Powerlink* i *Profinet IO*. Svaki od protokola definiše različite mehanizme koji omogućavaju efikasnu razmjenu vremenski kritičnih podataka u sistemu.

EtherCAT se zasniva na konceptu sumacionog okvira kod kojeg se procesni podaci grupišu u jedan *Ethernet* okvir. Ovim se postiže bolje iskorišćenje propusnog opsega komunikacionog sistema, jer su procesni podaci tipično veličine samo nekoliko bajtova.

Ethernet Powerlink koristi metod prozivanja upravljanih uređaja, čime jedan upravljački uređaj kontroliše pristup dijeljenom medijumu uz obezbjeđenje visokog nivoa determinizma u mreži. Da bi se smanjila kašnjenja mrežnih komponenata, za povezivanje krajnjih čvorova tipično se koriste koncentratori.

Profinet IO definiše nekoliko klasa saobraćaja koji se vremenski multipleksiraju u okviru komunikacionog ciklusa. Svaka klasa saobraćaja nudi različit nivo determinizma i vremenske sinhronizacije, što omogućava primjenu ovog komunikacionog sistema u izrazito heterogenim okruženjima.

3 Metrike u industrijskim komunikacionim sistemima

U ovom poglavlju, opisane su i analizirane metrike definisane standardom IEC 61784-2 [53]. Iako su neke od najznačajnijih metrika, koje opisuju mogućnosti RTE sistema sa aspekta kapaciteta podataka, vremenskih karakteristika i topologije, definisane u okviru pomenutog standarda, one ne karakterišu u dovoljnoj mjeri ciklično ponašanje RTE sistema, što je od ključnog značaja sa aspekta njihove primjene u distribuiranim sistemima automatizacije. Iz tog razloga, u literaturi su definisane specifične metrike, *minimalno vrijeme komunikacionog ciklusa* i *vrijeme odziva*.

Precizna definicija pomenutih metrika je od ključnog značaja kako bi se izbjegla pogrešna interpretacija odgovarajuće metrike prilikom projektovanja sistema. Opisane metrike analizirane su sa aspekta prethodno razmatranih RTE sistema, pri čemu su date smjernice za njihovu pravilnu interpretaciju i kvantifikaciju prilikom analize i projektovanja sistema.

Osim uvodnih razmatranja, poglavlje sadrži tri odjeljka. U odjeljku 3.1 su opisane i analizirane metrike definisane standardom IEC 61784-2. Detaljno su obrađene specifičnosti svake metrike u smislu razmatranih RTE sistema. U odjeljku 3.2 razmatrane su specifično definisane metrike, minimalno vrijeme komunikacionog ciklusa i vrijeme odziva, kao i postupak njihove kvantifikacije za odabrane RTE sisteme. Konačno, u odjeljku 3.3 izvedeni su osnovni zaključci.

3.1 Metrike definisane standardom IEC 61784-2

Standardom IEC 61784-2 definisan je skup metrika koje služe za kvantitativno i kvalitativno opisivanje mogućnosti industrijskog komunikacionog sistema zasnovanog na *Ethernet* komunikaciji. U terminologiji ovog standarda, za označavanje određene metrike, koristi se naziv *indikator performansi* (engl. *Performance Indicator* – PI). Takođe, standard propisuje da svaki komunikacioni profil treba da definiše konzistentan skup metrika kojima se opisuju mogućnosti datog RTE komunikacionog sistema koji podrazumijeva i eventualne međuzavisnosti između

pojedinih metrika.

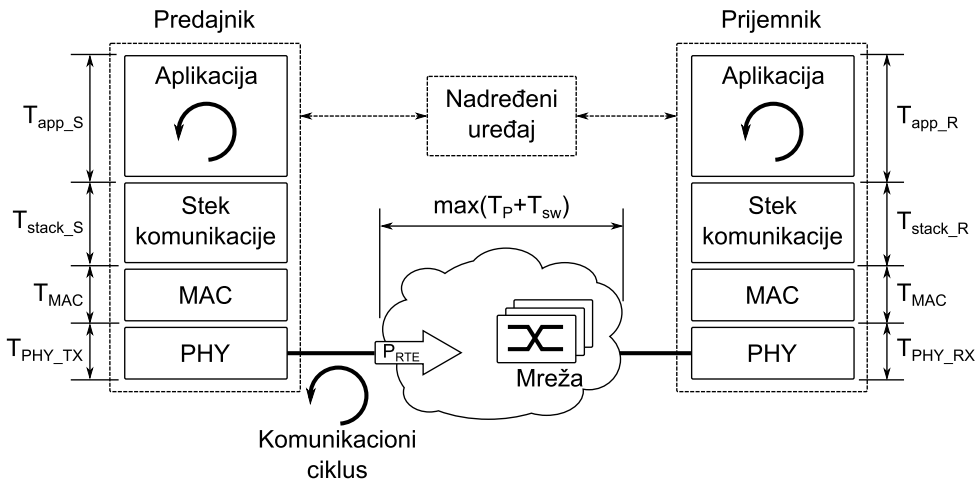
Definisano je ukupno devet metrika koje se mogu raspodijeliti u tri grupe. Prvu grupu, koja opisuje kapacitet podataka u RTE sistemu, čine sljedeće metrike: vrijeme isporuke (engl. *delivery time*), propusna moć (engl. *throughput RTE*) i propusni opseg rezervisan za NRT saobraćaj (engl. *non-RTE bandwidth*). Druga grupa se odnosi na vremensku karakterizaciju komponenata sistema i definiše: preciznost vremenske sinhronizacije (engl. *time synchronization accuracy*), preciznost sinhronizacije koja nije bazirana na globalnom vremenu (engl. *non-time-based synchronization accuracy*) i vrijeme oporavka od greške (engl. *redundancy recovery time*). Posljednja grupa opisuje topologiju mreže i čine je: broj krajnjih stanica (engl. *number of RTE end-stations*), osnovna mrežna topologija (engl. *basic network topology*) i broj mrežnih komutatora između krajnjih stanica (engl. *number of switches between RTE end-stations*).

3.1.1 Vrijeme isporuke

Vrijeme isporuke (T_D) se definiše kao vrijeme potrebno za prenos poruke sa vremenski kritičnim podacima od izvornog do odredišnog čvora. Prema IEC 61784-2 standardu, specificira se maksimalno vrijeme isporuke za dva slučaja: (1) prenos poruke bez grešaka i (2) gubitak jednog okvira prilikom prenosa sa oporavkom od greške. Vrijeme slanja se mjeri na aplikacionom sloju i uključuje sva kašnjenja koja se pojavljuju u procesu generisanja i dostavljanja podatka, slanja odgovarajuće poruke, kao i sva vremena čekanja u infrastrukturnim komponentama mreže prilikom prenosa poruke. Ovo vrijeme zavisi od mnogo faktora kao što su topologija mreže, opterećenje mreže (npr. NRT saobraćajem) ali i aplikacije koja generiše, odnosno konzumira podatak koji se prenosi u realnom vremenu. Vrijeme isporuke dodatno zavisi i od toga da li je obezbijeđena sinhronizacija između aplikacionih procesa u krajnjim čvorovima i komunikacionog ciklusa mreže.

Iako se u okviru IEC 61784-2 standarda vrijeme isporuke za različita rješenja interpretira i izračunava na različite načine (što je i razumljivo s obzirom na različitost pristupa samih komunikacionih profila), uzimajući u obzir prethodnu definiciju, ova metrika se u opštem slučaju može predstaviti kao što je ilustrovano na slici 3.1.

Sa slike je vidljivo da vrijeme isporuke, u opštem slučaju, određuju: period izvršavanja ciklične aplikacije na predajnoj i prijemnoj strani (T_{app_S} i T_{app_R}), kašnjenje kroz komunikacioni stek na predajnoj i prijemnoj strani (T_{stack_S} i T_{stack_R}), vrijeme potrebno za prosljeđivanje generisanog okvira kroz MAC (T_{MAC}) i PHY (engl. *PHYsical layer*) komponente na predajnoj i prijemnoj strani (T_{PHY_TX} i T_{PHY_RX}), te maksimalno kašnjenje kroz mrežne komponente



Slika 3.1: Komponente komunikacionog sistema koje utiču na vrijeme isporuke

koje zavisi od topologije mreže i propagacionog kašnjenja kroz kablove (T_P) i infrastrukturne komponente (T_{sw}). Takođe, važno je napomenuti da, osim prethodno pomenutih komponenata kašnjenja, prilikom izračunavanja vremena isporuke treba uzeti u obzir i ukupnu propusnu moć koju sistem zahtijeva za vremenski kritične podatke (P_{RTE}). Propusna moć određuje minimalno trajanje komunikacionog ciklusa koje se može postići u mreži.

U slučaju da komunikaciju kontroliše nadređeni uređaj (crtkano označen na slici 3.1), kao što je slučaj kod ECAT i EPL protokola, potrebna su najmanje dva komunikaciona ciklusa za razmjenu podataka između uređaja, jedan za slanje ulaznog podatka koji se dobija od predajne strane, i drugi za dostavljanje izlaznog podatka prijemnoj strani.

3.1.2 Propusna moć

Propusna moć (P_{RTE}) predstavlja ukupnu količinu vremenski kritičnih podataka koji se prenesu u jednoj sekundi izraženo brojem okteta ili bajtova u sekundi (B/s). Pri izračunavanju propusne moći u obzir se uzima samo količina stvarnih podataka bez kontrolnih informacija neophodnih za njihov prenos (zaglavlja protokola i drugi mehanizmi za obezbjeđenje pouzdanog prenosa informacija).

Prema definiciji, uzimajući u obzir cikličnu prirodu prenosa podataka u industrijskim komunikacionim sistemima, propusna moć se trivijalno može izraziti kao odnos vremena komunikacionog ciklusa (T_C) i ukupne količine vremenski kritičnih podataka koji se prenesu u okviru jednog ciklusa (L_{RTE}), tj.

$$P_{RTE} = \frac{L_{RTE}}{T_C}, \tag{3.1}$$

što pokazuje da se propusna moć povećava s povećanjem ukupne količine podataka ili smanjenjem perioda komunikacionog ciklusa. To znači da se povećanje propusne moći može različito

manifestovati na vrijeme isporuke, jer smanjenje vremena komunikacionog ciklusa u opštem slučaju vodi ka smanjenju vremena isporuke, dok se pri povećanju ukupne količine podataka dobija suprotan efekat.

Ukupna količina podataka, u opštem slučaju, zavisi od broja krajnjih čvorova, dok je vrijeme komunikacionog ciklusa određeno različitim faktorima koji uključuju topologiju mreže, broj infrastrukturnih komponenata, kao i veličinu podataka koji se u realnom vremenu razmjenjuju u mreži, te procenat propusnog opsega rezervisanog za NRT saobraćaj.

Važno je napomenuti da izraz (3.1) određuje trenutnu, a ne maksimalnu, vrijednost propusne moći, koja se može postići u datom sistemu za fiksirano vrijeme komunikacionog ciklusa. Kod RTE sistema kod kojih su RT i NRT saobraćaj razdvojeni u različite intervale (EPL i PNIO), maksimalna propusna moć se može izraziti formulom

$$P_{RTE(max)} = \frac{T_{RT} - (L_{ovh} \cdot 8)/B_{tot}}{T_C}, \tag{3.2}$$

gdje T_{RT} odgovara trajanju intervala rezervisanog za RT saobraćaj, dok L_{ovh} određuje veličinu kontrolnih informacija protokola izraženu u bajtovima, a B_{tot} ukupnu širinu propusnog opsega kanala (npr. 100 Mb/s).

U slučaju ECAT protokola, takođe, za određivanje maksimalne propusne moći, primjenljiv je izraz (3.2), pri čemu je član T_{RT} određen sa

$$T_{RT}^{ECAT} = T_{C(min)}, \tag{3.3}$$

gdje $T_{C(min)}$ odgovara minimalnom vremenu komunikacionog ciklusa koje je neophodno za razmjenu svih vremenski kritičnih podataka. Kao što će biti detaljnije objašnjeno u nastavku ovog poglavlja, ovo vrijeme podrazumijeva vrijeme potrebno za slanje ECAT okvira, kao i njegovu propagaciju kroz sve čvorove i povratak nadređenom čvoru.

3.1.3 Propusni opseg rezervisan za NRT saobraćaj

Propusni opseg rezervisan za NRT saobraćaj (B_{NRT}) određen je procentom ukupnog propusnog opsega koji se može koristiti za prenos poruka koje pripadaju saobraćaju koji nije vremenski kritičan. Očigledno je da su propusna moć i propusni opseg rezervisan za NRT saobraćaj međusobno zavisne metrike i da se svi faktori koji utiču na propusnu moć istovremeno odražavaju i na ovu metriku.

U opštem slučaju, prethodno definisana metrika se može kvantifikovati jednačinom

$$B_{NRT}(\%) = \left[1 - \frac{B_{RTE} + B_{ovh}}{B_{tot}}\right] \cdot 100, \tag{3.4}$$

gdje je: $B_{RTE} = P_{RTE} \cdot 8$ – propusni opseg, koji odgovara trenutnoj propusnoj moći i $B_{ovh} = (L_{ovh} \cdot 8)/T_C$ – propusni opseg, koji odgovara kontrolnim informacijama protokola.

Analizom prethodne jednačine može se zaključiti da propusni opseg, koji se može koristiti za prenos NRT poruka, zavisi od trenutne propusne moći, kao i od vremena komunikacionog ciklusa (uzimajući u obzir i veličinu kontrolnih informacija protokola). Za fiksirano vrijeme komunikacionog ciklusa, propusni opseg rezervisan za NRT saobraćaj varira u zavisnosti od zahtijevane propusne moći.

U stvarnim RTE sistemima, interval komunikacionog ciklusa rezervisan za NRT saobraćaj obično se fiksira na određenu vrijednost prilikom konfiguracije mreže, pa se ova metrika može jednostavnije izraziti jednačinom

$$B_{NRT}(\%) = \frac{T_{NRT}}{T_C} \cdot 100, \tag{3.5}$$

gdje T_{NRT} odgovara trajanju intervala rezervisanog za NRT saobraćaj.

Različiti komunikacioni profili propisuju različite zahtjeve po pitanju procenta rezervacije propusnog opsega za NRT saobraćaj. Tako, na primjer, PNIO specifikacija preporučuje da se za NRT saobraćaj rezerviše najmanje 40% ukupnog propusnog opsega. Takođe, alati za konfiguraciju PNIO mreže obično ne dozvoljavaju korisniku podešavanje trajanja zelenog intervala ispod 125 μs kako bi se garantovao prenos jednog kompletnog *Ethernet* okvira maksimalne veličine tokom ovog intervala, jer PNIO protokol ne definiše mehanizme koji omogućavaju fragmentaciju okvira maksimalne dužine.

S druge strane, ECAT i EPL specifikacije omogućavaju fragmentaciju *Ethernet* okvira korišćenjem specijalnih uređaja koji podržavaju posebno definisane komunikacione mehanizme. S tim u vezi, EPL specifikacija omogućava prenos asinhronih poruka sa minimalnom veličinom polja korisnih podataka *Ethernet* okvira koja iznosi 300 bajtova. ECAT specifikacija ne definiše nikakva ograničenja po ovom pitanju i ostavlja slobodu korisniku da sistem konfiguriše u skladu sa zahtjevima koje nameće njegova primjena uzimajući u obzir ograničenja definisanih mehanizama NRT komunikacije [72].

3.1.4 Preciznost vremenske sinhronizacije

Preciznost vremenske sinhronizacije definiše maksimalno odstupanje lokalnog vremena internih časovnika bilo koja dva čvora. Različita komunikaciona rješenja obezbjeđuju različite mehanizme koji omogućavaju veoma preciznu sinhronizaciju svih uređaja. Najčešće se koriste mehanizmi sinhronizacije zasnovani na standardu IEEE 1588 koji definiše PTP (engl. *Precision Time Protocol*) protokol za sinhronizaciju lokalnog vremena umreženih čvorova [73, 74].

PNIO specifikacijom definiše se unapređena verzija PTP protokola, pod nazivom PTCP (engl. *Precision Transparent Clock Protocol*), koja omogućava kompenzaciju kašnjenja kroz infrastrukturne komponente uvođenjem tzv. transparentnih časovnika, TC (engl. *Transparent Clock*), čime se postiže viši nivo sinhronizacije čak i kada mreža sadrži veliki broj kaskadno vezanih mrežnih komutatora pri formiranju linijske topologije [75]. Na taj način se može postići sinhronizacija koja omogućava da odstupanje lokalnog vremena bilo koja dva čvora bude znatno ispod 1 μs uz uslov da maksimalan broj kaskadno vezanih mrežnih komutatora između bilo koja dva krajnja čvora nije veći od 20. Eksperimentalni rezultati, prezentovani u [30, 34], pokazuju da je u slučaju manjeg broja uređaja moguće postići sinhronizaciju koja je znatno ispod 100 ns.

Sinhronizacija čvorova u EPL protokolu zasniva se na periodičnom slanju SoC okvira čija je uloga da označi početak komunikacionog ciklusa i sinhronizuje sve čvorove u mreži. Ovaj okvir sadrži polje sa trenutnim apsolutnim vremenom lokalnog časovnika upravljačkog čvora koje upravljani čvorovi mogu koristiti za korekciju sopstvenog lokalnog vremena. Ukupno odstupanje zavisi od toga koliko se precizno generišu i šalju SoC okviri, kao i od ukupnog odstupanja mreže kroz infrastrukturne komponente. S obzirom da EPL za infrastrukturne komponente preporučuje korišćenje koncentratora čije je odstupanje kašnjenja ispod 100 ns, u EPL mreži se može postići precizna sinhronizacija čvorova (ispod 1 μs) [76], što je i eksperimentalno potvrđeno u [32]. Važno je napomenuti da, slično kao kod PNIO protokla, preciznost sinhronizacije čvorova zavisi od broja koncentratora između dva najudaljenija čvora.

Konačno, ECAT protokol definiše poseban mehanizam za sinhronizaciju čvorova, koji se zasniva na konceptu *distribuiranih časovnika*, DC (engl. *Distributed Clocks*). U osnovi, prvi podređeni uređaj u logičkoj topologiji prstena ima ulogu *referentnog časovnika* (engl. *reference clock*). Periodičnim slanjem odgovarajućih telegrama sa ARMW komandom, nadređeni uređaj omogućava distribuciju lokalnog vremena referentnog čvora svim ostalim čvorovima u mreži. Kompenzacija propagacionog kašnjenja između čvorova obezbjeđuje se odgovarajućim DC mehanizmima zasnovanim na mjerenju ovog kašnjenja prilikom inicijalizacije sistema. Na taj način, može se ostvariti preciznost sinhronizacije ECAT čvorova znatno ispod 1 μs [76]. Eksperimen-

talni rezultati prezentovani u [77] pokazuju da je sa komercijalno dostupnim komponentama, moguće postići nivo sinhronizacije čije je odstupanje ispod 100 ns.

3.1.5 Preciznost sinhronizacije koja nije bazirana na globalnom vremenu

Preciznost sinhronizacije koja nije bazirana na globalnom vremenu predstavlja maksimalno odstupanje kašnjenja prilikom prenosa cikličnih poruka koje se naziva i *komunikacioni džiter*. Ovo odstupanje zavisi od strukture mreže, kao i od broja i vrste korišćenih infrastrukturnih komponenata, ali i sposobnosti i brzine mreže da reaguje na pojavu eventualnih grešaka koje za posljedicu mogu da imaju gubitak jednog ili više okvira, ili, u najgorem slučaju, kompletnog komunikacionog ciklusa.

Komunikacioni džiter, takođe, zavisi i od sposobnosti krajnjih čvorova da generišu i šalju ciklične poruke sa preciznim vremenskim intervalima. Kod EPL i ECAT protokola, ciklična razmjena poruka je kontrolisana upravljačkim, odnosno nadređenim uređajem, tako da komunikacioni džiter dominatno zavisi od mogućnosti navedenog uređaja da inicira komunikacione cikluse sa visokim nivoom preciznosti. Ovo najčešće podrazumijeva hardversku realizaciju najkritičnijih komponenata nadređenog uređaja.

3.1.6 Vrijeme oporavka od greške

Vrijeme oporavka od greške definiše maksimalno vrijeme koje protekne od pojave greške do poptpuno operativnog stanja u slučaju jednog permanentnog otkaza (prekid jedne linije veze ili otkaz jednog čvora). Da bi se obezbijedio oporavak od permanentnog otkaza, mreža se obično konfiguriše tako da bude organizovana u vidu topologije prstena uz definisanje specifičnih mehanizama koji omogućavaju detekciju otkaza i rekonfiguraciju mreže kako bi se omogućio nastavak normalnog rada mreže bez prekida.

U PNIO specifikaciji, definisana su dva protokola koji omogućavaju rekonfiguraciju mreže u slučaju jednog permanentog otkaza, MRP (engl. *Media Redundancy Protocol*) i MRPD (engl. *Media Redundancy for Planned Duplication*) protokol. Oba protokola koriste topologiju prstena kao neophodan preduslov za omogućenje mehanizama redundanse. MRP protokol omogućava vrijeme oporavka koje ne prelazi 200 ms [78, 79]. MRPD predstavlja unapređenu verziju protokola koja podrazumijeva unaprijed isplaniranu komunikaciju unutar prstena koja se odvija u okviru sinhronog IRT kanala [80]. Ovo podrazumijeva primjenu specijalnih IRT mrežnih komutatora i pažljivo planiranje vremenskog rasporeda slanja poruka u mreži. Kao rezultat dobijaju se veoma efikasni mehanizmi redundanse sa nultim vremenom oporavka.

EPL i ECAT takođe koriste topologiju prstena za realizaciju mehanizama oporavka od jednog permanentnog otkaza. Prema specifikacijama protokola, vrijeme oporavka je određeno vremenom komunikacionog ciklusa, jer se jedan kompletan ciklus gubi za detekciju nastale greške. U slučaju EPL protokola, vrijeme detekcije i oporavka od greške može se dodatno smanjiti korišćenjem drugih mehanizama koji podrazumijevaju replikaciju kompletne topologije mreže uz upotrebu specijalnih uređaja koji se nazivaju *selektori veze*, LS (engl. *Link Selector*), i redundantnih upravljačkih čvorova [26].

3.1.7 Broj krajnjih stanica

Broj krajnjih stanica ili *krajnjih čvorova* predstavlja maksimalan broj čvorova koji je podržan u mrežnom segmentu, ne računajući infrastrukturne komponente poput premosnika (engl. *bridge*), komutatora, koncentratora itd. Ova metrika svakako utiče na ostale metrike s obzirom da svaki čvor razmjenjuje određenu količinu podataka, što neminovono dovodi do povećanja zahtijevane propusne moći za vremenski kritične podatke, čime se indirektno utiče na vrijeme isporuke i propusni opseg rezervisan za NRT saobraćaj.

Ograničenje po pitanju maksimalnog broja krajnjih čvorova zavisi od načina adresiranja čvorova koje definiše odgovarajući komunikacioni profil, kao i drugih fizičkih ograničenja koja su posljedica strukture mreže, odnosno zahtijevanih performansi sistema.

Kao što je opisano u poglavlju 2, ECAT koristi 16-bitno ADP polje za definisanje jedinstvene adrese svakog podređenog uređaja. Prema tome, ECAT u jednom mrežnom segmentu teoretski može da sadrži maksimalno 65536 čvorova. Međutim, jasno je da, s obzirom na koncept rada i organizaciju mreže po principu logičkog prstena, nije realno očekivati da se mreža sa ovolikim brojem čvorova može praktično realizovati a da istovremeno budu zadovoljeni zahtijevi po pitanju performansi sistema.

S druge strane, EPL dozvoljava maksimalno 240 čvorova u mrežnom segmentu: jedan upravljački čvor i do 239 upravljanih čvorova. Međutim, slično kao kod ECAT komunikacije, princip prozivanja čvorova može da bude neefikasan u smislu iskorišćenja propusnog opsega, što utiče na maksimalan broj krajnjih čvorova ukoliko je potrebno realizovati sistem visokih performansi.

Konačno, adresiranje u PNIO protokolu je bazirano na MAC adresama. Prema tome, logički gledano, u jednom mrežnom segmentu može da se nalazi veoma veliki broj krajnjih čvorova. Međutim, kao i u prethodnim slučajevima, zahtijevane performanse sistema igraju ključnu ulogu po pitanju stvarnog maksimalnog broja čvorova u mreži.

3.1.8 Osnovna mrežna topologija

Osnovna mrežna topologija podrazumijeva jednu od tri osnovne topologije definisane standardom: hijerarhijski organizovanu topologiju zvijezde (engl. *hierarchical star*), topologiju prstena (engl. *ring*), ili linijsku (linearnu) topologiju (engl. *linear topology*); odnosno bilo koju kombinaciju ovih topologija. Pomenute osnovne mrežne topologije definisane IEC 61784-2 standardom prikazane su na slici 3.2.

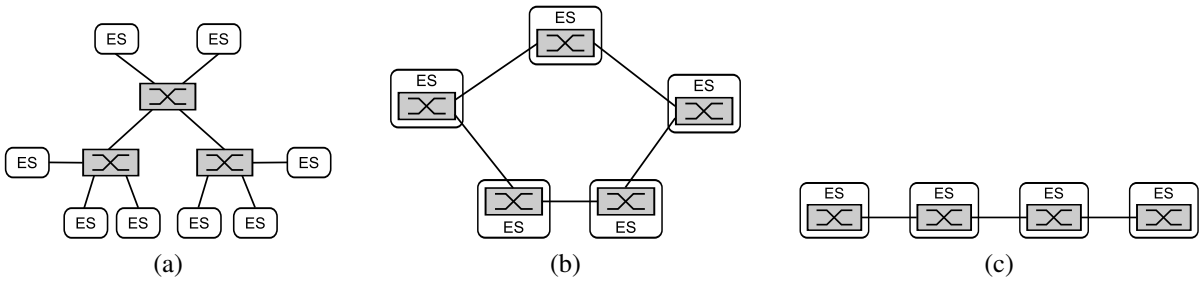
Kao što može da se vidi sa slike, *hijerarhijska topologija prstena* se realizuje korišćenjem višeportnih infrastrukturnih komponenata (komutatora ili koncentratora) koji se međusobno hijerarhijski povezuju tako da formiraju topologiju stabla ili zvijezde. Na svaku od ovih infrastrukturnih komponenata povezuje se određeni broj krajnjih uređaja koji formiraju lokalnu topologiju zvijezde.

Linijska topologija najčešće se realizuje kroz kaskadno (lančano) povezivanje uređaja (engl. *daisy chain*). Za realizaciju ove topologije koriste se uređaji u kojima je integrisana funkcionalnost infrastrukturne komponente sa tri porta.

Topologija prstena predstavlja specijalan tip linijske topologije kod koje se posljednji čvor povezuje sa prvim čvorom u liniji. Ova vrsta topologije tipično se koristi ukoliko je potrebno obezbijediti mehanizme za oporavak od greške u sistemu kroz uvođenje tzv. *redundantnog kabliranja* (engl. *cable redundancy*).

Kao što je već pomenuto u prethodnom poglavlju, sva tri komunikaciona rješenja, koja se razmatraju u ovoj disertaciji, omogućavaju realizaciju bilo koje od tri osnovne topologije, kao i bilo koju njihovu kombinaciju. Na taj način, korisniku se obezbjeđuje značajna fleksibilnost prilikom projektovanja sistema.

Topologija, u određenim slučajevima, može značajno da utiče na druge metrike definisane standardom. Jasno je da, kod rješenja koja koriste infrastrukturne komponente (EPL i PNIO), postoji ograničenje po pitanju maksimalnog broja infrastrukturnih komponenata između dva



Slika 3.2: Osnovne mrežne topologije: (a) hijerarhijski organizovana topologija zvijezde, (b) topologija prstena i (c) linijska topologija

krajnja čvora, ukoliko se želi postići zahtijevani nivo sinhronizacije čvorova. Za topologije bazirane na linijskoj ili topologiji prstena, ovo ograničenje se ujedno odražava i na maksimalan broj krajnjih čvorova. Takođe, s obzirom da je ukupno kašnjenje između dva najudaljenija čvora u mreži određeno topologijom mreže, ona u opštem slučaju utiče i na vrijeme isporuke koje se može ostvariti u sistemu, kao i na ukupan džiter koji unose infrastrukturne komponente.

3.1.9 Broj mrežnih komutatora između krajnjih stanica

Broj mrežnih komutatora između krajnjih stanica definiše maksimalan broj komutatora između bilo koja dva krajnja čvora u mrežnom segmentu. Ovaj broj zavisi od topologije mreže i zahtijevanog nivoa preciznosti sinhronizacije čvorova, kao i od zahtijevanih performansi sistema.

Kao što je opisano u odjeljku 3.1.4, PNIO standardom je definisano da broj mrežnih komutatora između bilo koja dva krajnja uređaja (uključujući i komutatore koji su integrisani u same krajnje uređaje) ne smije da bude veći od 20 ukoliko se želi postići sinhronizacija čvorova čija je preciznost značajno ispod 1 μs.

S druge strane, EPL specifikacija preporučuje korišćenje koncentratora umjesto komutatora kako bi se poboljšao determinizam u mreži. Iako specifikacijom nije propisan maksimalan broj koncentratora koji smije da se nalazi između krajnjih čvorova, jasno je da ovo ograničenje zavisi od maksimalnog odstupanja kašnjenja infrasturkturnih komponenata i zahtijevanog nivoa preciznosti sinhronizacije. Za maksimalno odstupanje od 70 ns, broj koncentratora između bilo koja dva krajnja čvora (uključujući koncentratore integrisane u same uređaje) ne smije da bude veći od 14, ukoliko se zahtijeva preciznost sinhronizacije ispod 1 μs. Ako se u EPL mreži koriste mrežni komutatori kao infrastrukturne komponente, ovaj broj mora da bude još manji.

Konačno, važno je istaći da za određene komunikacione profile, kao što je ECAT, ova metrika ne može da se primjeni, jer se infrastrukturne komponente tipično ne koriste pri realizaciji sistema. Kod ovih rješenja, infrastrukturna komponenta zadužena za prosljeđivanje okvira je integrisana u sam uređaj i ne utiče na sinhronizaciju čvorova, jer se uvode mehanizmi za preciznu kompenzaciju kašnjenja svih komponenata u sistemu.

3.2 Specifično definisane metrike

Iako IEC 81784-2 standard definiše temeljan skup metrika za opis mogućnosti industrijskih komunikacionih sistema, često se koriste i druge metrike koje bolje opisuju ciklično ponašanje RTE sistema. Kako ne postoji formalni dokument koji sadrži definiciju ovih metrika, ovdje će

se koristiti terminologija i definicije koje se poklapaju sa onim kod većine autora radova koji tretiraju ovu oblast.

3.2.1 Minimalno vrijeme komunikacionog ciklusa

Minimalno vrijeme komunikacionog ciklusa ($T_{C(min)}$) je jedna od najčešće korišćenih metrika za opis RTE sistema koji podržavaju cikličnu komunikaciju. Kao što je definisano u [17, 18, 52, 61], ova metrika predstavlja vrijeme potrebno da svi umreženi čvorovi razmijene ulazno/izlazne ciklične podatke i, prema tome, može da se smatra minimalnim vremenom odmjeravanja koje se može postići u mreži. Međutim, treba napomenuti da je struktura ciklusa komunikacije kod mnogih RTE sistema (npr. EPL i PNIO) takva da se RT i NRT poruke razmjenjuju u odvojenim vremenskim intervalima. To znači da se kod ovih sistema, u stvarnim uslovima, nikada ne može postići minimalno vrijeme komunikacionog ciklusa opisano prethodnom definicijom. Bez obzira na to, ova metrika predstavlja važan parametar svake mreže jer određuje donju granicu komunikacionog ciklusa i, na taj način, može da pomogne u pronalaženju uskog grla (engl. *bottleneck*) prilikom konfiguracije mreže.

S obzirom na datu definiciju i na princip rada, minimalno vrijeme komunikacionog ciklusa kod ECAT protokola određeno je vremenom potrebnim da sumacioni okvir koji šalje nadređeni uređaj na početku ciklusa prođe kroz sve čvorove u mreži i vrati se ovom uređaju. Jasno je da ovako određeno vrijeme zavisi od vremena potrebnog za slanje kompletnog okvira, kao i od propagacionih kašnjenja u mreži.

U slučaju EPL protokola, minimalno vrijeme komunikacionog ciklusa odgovara dužini trajanja izohrone faze pa, prema tome, direktno zavisi od broja čvorova koji se prozivaju u ovoj fazi. Ovo vrijeme, takođe, zavisi i od strukture mreže, kao i od propagacionih kašnjenja koje unose mrežne komponente. Najkritičniji parametar u tom smislu predstavlja vrijeme odziva EPL čvora.

Slično kao kod EPL protokola, minimalno vrijeme komunikacionog ciklusa kod PNIO komunikacije, određeno je trajanjem IRT kanala. Kod sistema visokih performansi, koji su primarno predmet ove disertacije, relevantno je samo trajanje crvenog intervala, jer se u njrmu razmjenjuju vremenski isplanirane poruke koje omogućavaju minimalno zadržavanje u infrastrukturnim komponentama. Dužina trajanja ovog intervala zavisi od broja čvorova, broja i dužine poruka, kao i od algoritma koji generiše vremenski plan slanja poruka u zavisnosti od topologije.

3.2.2 Vrijeme odziva

Vrijeme odziva (T_{RT}) ili *vrijeme reakcije* je veoma važna metrika koja se koristi za opisivanje sposobnosti sistema da u datim uslovima odreaguje na događaje inicirane spolja. U kontekstu industrijskih komunikacionih sistema, ova metrika se definiše kao kašnjenje između pojave događaja u okviru određenog procesa i odgovora kao reakciju na dati događaj koji proizvodi kontroler unutar istog procesa [63–66].

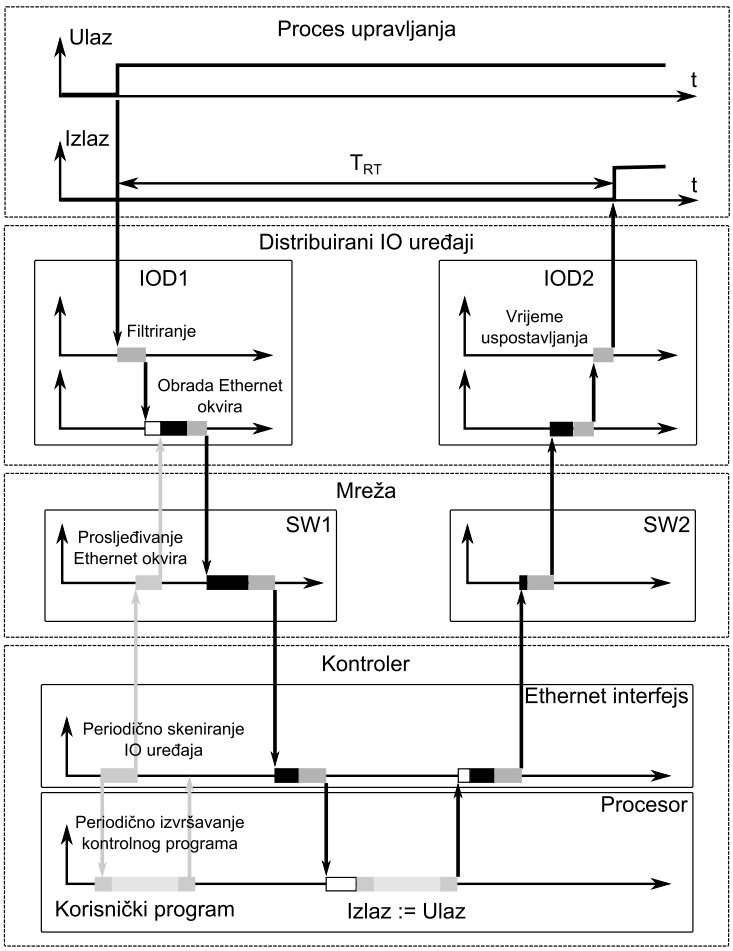
U opštem slučaju, vrijeme odziva zavisi od različitih faktora koji zavise od konkretnih komponenata koje se koriste za realizaciju sistema. Prema [64], ovo vrijeme se može razvrstati u sljedeće tri komponente kašnjenja: kašnjenje zbog obrade podataka, kašnjenje zbog sinhronizacije i kašnjenje zbog zauzetosti dijeljenih resursa.

Komponenta kašnjenja koja nastaje zbog obrade podataka zavisi od karakteristika samog uređaja i obično se definiše u okviru specifikacije proizvođača. U slučaju ulaznog uređaja, ovo kašnjenje može. na primjer, da bude posljedica filtriranja ulaznog signala ili da se odnosi na vrijeme potrebno za pretvaranje analogne vrijednosti ulaznog signala u digitalni oblik. Slično, kod izlaznih uređaja, kašnjenje signala je određeno vremenom uspostavljanja izlaznog nivoa. Konačno, komponenta ovog kašnjenja može da nastane kao posljedica kopiranja podataka u interni komunikacioni bafer čiji se sadržaj šalje unutar odgovarajućih poruka.

Kašnjenje zbog sinhronizacije može da nastane kao posljedica interne i eksterne sinhronizacije. Interna sinhronizacija podrazumijeva sinhronizaciju između komponenata unutar samog uređaja (npr. vrijeme čekanja u kontroleru do početka izvršavanja novog ciklusa aplikacije), dok eksterna sinhronizacija podrazumijeva sinhronizaciju između mrežnih komponenata (npr. vrijeme čekanja do prijema zahtjeva za slanje novih podataka od kontrolera). Treba napomenuti da RTE sistemi kod kojih je komunikacioni ciklus sinhronizovan sa ciklusima izvršavanja aplikacije u kontroleru i distribuiranim uređajima, mogu u potpunosti da eliminišu ovo kašnjenje.

Treći tip kašnjenja se odnosi na trenutno zauzeće resursa unutar same komponente. To može da podrazumijeva zauzeće procesora koji obrađuje neki drugi zadatak, ili zauzeće izlaznog porta mrežnog komutatora. Vremenskim planiranjem zadataka na nivou aplikacija koje se izvršavaju u kontroleru i distribuiranim uređajima, kao i na nivou mreže, ovo kašnjenje se može značajno redukovati.

Na slici 3.3 prikazan je jedan primjer koji ilustruje način određivanja vremena odziva u industrijskoj mreži sa označenim komponentama kašnjenja koje su prethodno opisane. U prikazanoj mreži, pretpostavljeno je da periodično skeniranje distribuiranih uređaja inicira kontorler slanjem specifičnih zahtjeva koji zavise od konkretnog RTE sistema. Komponente kašnjenja



Slika 3.3: Primjer određivanja vremena odziva u industrijskoj mreži (bazirano na [64])

označene sivom bojom odnose se na kašnjenja zbog obrade podataka, bijelom bojom na kašnjenja zbog sinhronizacije i crnom bojom na kašnjenja zbog zauzetosti dijeljenih resursa. Kao što može da se vidi na slici 3.3, detekcija promjene ulaznog signala u distribuiranom uređaju IOD1 zakašnjenja je zbog obrade signala pri procesu filtriranja. Nakon toga, podatak se smješta u interni bafer uređaja i spreman je za slanje. Međutim, slanje ne može da se inicira dok ne stigne zahtjev od kontrolera, što je naznačeno kašnjenjem koje je posljedica eksterne sinhronizacije. Slanje okvira se dalje odgađa, jer je potrebno čekati da se oslobode resursi unutar uređaja. Konačno, nakon kašnjenja obrade podatka koje podrazumijeva kopiranje iz internog bafera i enkapsulaciju u odgovarajući okvir, podaci se dostavljaju mreži. U mreži, okvir je dodatno zakašnjen zbog zauzeća resursa mrežnog komutatora SW1 i vremena potrebnog za prosljeđivanja okvira. Konačno, okvir stiže do prijemnog bafera u okviru *Ethernet* interfejsa kontrolera koji, kada se oslobode resursi, prosljeđuje podatke u dijeljenu memoriju kontrolera kako bi mogli biti obrađeni u okviru korisničkog programa koji se ciklično izvršava sa definisanim periodom. Međutim, s obzirom da korisnički program u opštem slučaju nije sinhronizovan sa komunikacionim ciklusom, dolazi do dodatnog kašnjenja uslijed interne sinhronizacije. Kada korisnički

program obradi ulazni podatak (u konkretnom primjeru izlaz prosto prati promjenu na ulazu) i smjesti rezultat u dijeljenu memoriju, u narednom komunikacionom ciklusu šalje se odgovor izlaznom uređaju kao što je ilustrovano na slici 3.3.

Iz navedenog primjera, uočljivo je vrijeme odziva ne zavisi samo od kašnjenja unutar mreže, već i od drugih efekata, kao što je sinhronizacija, kako eksternih komponenata sistema, tako i internih komponenata samih uređaja. U opštem slučaju, prema [67], minimalno i maksimalno vrijeme odziva može se opisati sljedećim jednačinama:

$$T_{RT(min)} = \sum_{\forall x} T_{(x)}, \tag{3.6}$$

$$T_{RT(max)} = \sum_{\forall x} T_{p(x)} + \sum_{\forall x} T_{w(x)}, \tag{3.7}$$

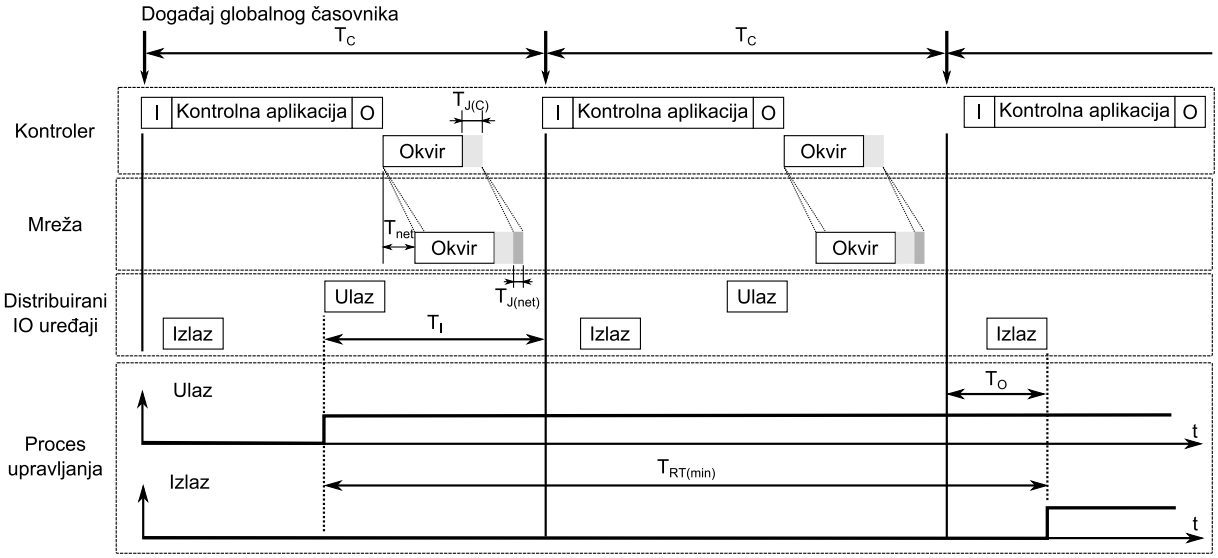
gdje $T_{p(x)}$ čine komponente kašnjenja obrade, a $T_{w(x)}$ kašnjenja uzrokovana zauzećem resursa ili kao posljedica nedostatka sinhronizacije komponenata sistema. Iz prethodnih jednačina, može se zaključiti da je apsolutno odstupanje vremena odziva (džiter) jednako

$$\Delta T_{RT} = T_{RT(J)} = T_{RT(max)} - T_{RT(min)} = \sum_{\forall x} T_{w(x)}, \tag{3.8}$$

što znači da varijacija vremena odziva u sistemu direktno zavisi od ukupnog zbira vremena kašnjenja svih komponenata sistema.

S obzirom da RTE sistemi, koji se analiziraju u ovom radu, podržavaju precizne mehanizme sinhronizacije, kako između mreže i mrežnih interfejsa kontrolera i distribuiranih uređaja, tako i na nivou samih aplikacija, može se smatrati da je vrijeme odziva dominantno određeno postavljenim vremenom komunikacionog ciklusa [67, 81]. Ovo znači da su kašnjenja zbog sinhronizacije u potpunosti eliminisana. Preciznim planiranjem i strukturiranjem mreže, takođe se mogu otkloniti i kašnjenja koja nastaju kao posljedica zauzetosti resursa u infrastrukturnim komponentama. Jedan primjer potpuno sinhronizovanog industrijskog komunikacionog sistema, ilustrovan je na slici 3.4. S obzirom da se sve aktivnosti iniciraju istovremeno, ovakvi sistemi se nazivaju i *izohroni sistemi*.

Kao što može da se vidi sa slike, trenutak aktiviranja kontrolne aplikacije, odmjeravanja ulaznih i aktuelizacije izlaznih vrijednosti je sinhronizovan na sistemskom nivou posredstvom globalnog časovnika. Nakon aktivacije, kontrolna aplikacija čita trenutne ulazne vrijednosti iz procesne slike, izračunava nove vrijednosti izlaza u skladu sa algoritmom upravljanja i upisuje izračunate vrijednosti na lokacije izlaza u procesnoj slici. Zatim se inicira slanje okvira kojim



Slika 3.4: Vrijeme odziva u potpuno sinhronizovanom (izohronom) RTE sistemu

se dostavljaju izračunate vrijednosti izlaza od kontrolera prema distriburiranim uređajima i novih ulaznih vrijednosti od distribuiranih uređaja prema kontroleru. Prilikom slanja, okviri su zakašnjeni za vrijeme potrebno da se oni prenesu kroz mrežne uređaje (T_{net}) čija vrijednost može da varira zbog komunikacionog džitera ($T_{J(net)}$). Osim komponente džitera koju unose mrežni uređaji, postoji i komponenta džitera koju unosi sam kontroler prilikom slanja okvira ($T_{J(C)}$), a koja se mora uzeti u obzir prilikom podešavanja vremena komunikacionog ciklusa (T_C) sistema. Vrijeme potrebno za odmjeravanje ulaza (T_I) i vrijeme potrebno za aktuelizaciju izlaza (T_O) podešavaju se relativno u odnosu na početak komunikacionog ciklusa na način da se novi ulazni podaci odmjeravaju tako da budu spremni prije slanja odgovarajućih okvira, odnosno da se izlazne vrijednosti aktuelizuju kada svi uređaji prime nove izlazne vrijednosti izračunate od strane kontrolera. Sa slike 3.4 se vidi da je minimalno vrijeme odziva određeno sa

$$T_{RT(min)} = T_I + T_C + T_O. \tag{3.9}$$

U najlošijem slučaju, odmjeravanje ulaza može da bude odgođeno za sljedeći ciklus komunikacije pa je maksimalno vrijeme odziva

$$T_{RT(max)} = T_I + 2T_C + T_O, \tag{3.10}$$

što znači da je odstupanje u ovom slučaju jednako komunikacionom ciklusu sistema, tj. $\Delta T_{RT} = T_{RT(J)} = T_C$.

Vrijeme ciklusa je određeno vremenom izvršavanja kontrolne aplikacije, vremenom slanja svih okvira, kašnjenjem koje unose mrežni uređaji, kao i svim komponentama komunikacionog

džitera. Važno je napomenuti da se prilikom konfiguracije sistema određeni dio komunikacionog ciklusa rezerviše za NRT saobraćaj, što značajno utiče na vrijeme ciklusa. Vrijeme izvršavanja kontrolne aplikacije može se smanjiti povećanjem računarskih resursa što, s obzirom na aktuelne trendove razvoja računarske opreme, obično ne predstavlja problem. S druge strane, pravilnom konfiguracijom mreže, pažljivim planiranjem topologije i optimizacijom mehanizama komunikacije koji se koriste u industrijskim komunikacionim sistemima, moguće je postići smanjenje dužine trajanja intervala potrebnog za razmjenu vremenski kritičnih poruka, a samim tim i vremena odziva sistema.

3.3 Zaključak

U cilju poređenja različitih industrijskih komunikacionih sistema, definisane su odgovarajuće metrike, kao objektivne mjere njihovih mogućnosti. Specifičan doprinos poglavlja predstavlja opis i kvantifikacija navedenih metrika u smislu RTE sistema opisanih u prethodnom poglavlju. Metrike su obično međusobno zavisne, što nameće zaključak da se one ne mogu i ne trebaju posmatrati izolovano prilikom projektovanja određenog sistema. Iz navedenog razloga, metrike se obično grupišu tako da čine konzistentan skup koji opisuje određene karakteristike industrijskih komunikacionih sistema.

Prvi skup metrika, definisan standardom IEC 61784-2, opisuje vremenske karakteristike, kapacitet podataka i topologiju mreže. Pokazano je da tri RTE sistema, koja su predmet istraživanja ove disertacije, unose različite specifičnosti i ograničenja prilikom određivanja pomenutih metrika. Tako se, na primjer, kod RTE sistema sa rezervacijom propusnog opsega za vremenski kritičan saobraćaj (EPL i PNIO) definiše trenutna i maksimalna propusna moć. S druge strane, kod koncepta sa sumacionim okvirom (ECAT), maksimalna i trenutna propusna moć su uvijek jednake. Osim toga, EPL i PNIO unose značajna ograničenja po pitanju broja infrastrukturnih komponenata i topologije mreže, kada se želi postići zahtijevani nivo preciznosti vremenske sinhronizacije krajnjih čvorova (ispod 1 μs). Važno je pomenuti da EPL i PNIO, takođe, uvijek rezervišu određeni dio komunikacionog ciklusa za razmjenu poruka koje nisu vremenski kritične, što nije slučaj sa ECAT sistemom. Konačno, broj krajnjih čvorova je, osim topologijom i zahtijevanim nivoom sinhronizacije, određen načinom adresiranja. Tako, na primjer, u jednom EPL mrežnom segmentu ne može da se nalazi više od 240 čvorova.

Specifično definisane metrike, minimalno vrijeme komunikacionog ciklusa i vrijeme odziva, bolje opisuju ciklično ponašanje RTE sistema. Minimalno vrijeme komunikacionog ciklusa predstavlja donju granicu vremena odmjeravanja signala koji se obrađuju u sistemu i, prema

tome, određuje njegove dinamičke karakteristike. Ova metrika, međutim, ne uzima u obzir realne karakteristike i ograničenja analiziranog RTE sistema, kao što su uticaj saobraćaja koji se ne odvija u realnom vremenu i efekti sinhronizacije komponenata sistema. S druge strane, vrijeme odziva, osim kašnjenja usljed obrade podataka, uzima u obzir i dodatne efekte poput zauzeća mrežnih i računarskih resursa, te vremena čekanja zbog nedostatka sinhronizacije između različitih komponenata sistema. Navedeni efekti proizvode apsolutno odstupanje (džiter) čija je maksimalna vrijednost određena ukupnim zbirom vremena čekanja svih komponenata sistema. U slučaju potpuno sinhronizovanog (izohronog) sistema, vremena čekanja se eliminišu, čime je apsolutno odstupanje vremena odziva određeno konfigurisanim vremenom ciklusa RTE sistema. Ovo omogućava pojednostavljenje analize vremena odziva i stvara prostor za njegovu optimizaciju.

4 Modelovanje industrijskih komunikacionih sistema

Prilikom analize nekog industrijskog komunikacionog sistema potrebno je obezbijediti odgovarajuće analitičke modele koji omogućavaju kvantitativnu ocjenu metrika koje su prezentovane u prethodnom poglavlju. Analitički modeli obično zahtijevaju karakterizaciju relevatnih vremenskih parametara sistema koji određuju njegovo ponašanje i samim tim znatno utiču na preciznost samog modela. Vrijednosti ovih parametara mogu da se procijene teorijskim putem ili odgovarajućim mjerenjima na prototipskim mrežama sa manjim brojem uređaja.

U ovom poglavlju izvedeni su analitički modeli koji opisuju ponašanje RTE sistema koji su predmet istraživanja ove disertacije. Teorijski modeli su koncipirani tako da u obzir uzmu parametre i karakteristike stvarnih sistema, koje su obično zanemarene ili znatno pojednostavljene u analitičkim modelima drugih autora, uključujući one koje je autor prezentovao u svom dosadašnjem istraživanju [23, 46, 82].

Modeli predloženi u ovom poglavlju, predstavljaju jedan od glavnih doprinosa ove disertacije. Ovo se prvenstveno odnosi na specifičnosti konfiguracije mreže korišćenjem komercijalno dostupnih konfiguracionih alata. Izvedene su jednačine za dvije osnovne topologije definisane standardom IEC 61784-2, linijsku i topologiju zvijezde. Topologija prstena je izostavljena iz analize, jer se ona primarno koristi za realizaciju redundatnih topoloških struktura, čije je razmatranje izvan opsega ove disertacije. Relevantni vremenski parametri predloženih modela, dobijeni su ekperimentalnim putem na prototipskim mrežama, koje čine uređaji različitih proizvođača, što je dodatni doprinos ovog rada. Konačno, sprovedena je analiza tri razmatrana RTE sistema u smislu definisanih metrika za nekoliko različitih scenarija koji podrazumijevaju varijaciju broja uređaja i veličine procesnih podataka za prethodno pomenute topologije. S obzirom da sva tri rješenja podržavaju izohronu komunikaciju na sistemskom nivou, odnosno da obezbjeđuju mehanizme sinhronizacije na nivou aplikacije, sprovedena je uporedna analiza pomenuta tri pristupa na primjeru konfiguracije čija struktura odgovara mreži koja se može naći u stvarnim sistemima.

Strukturu poglavlja, osim uvodnog, čine odjeljci u kojima su izvedeni prethodno pomenuti analitički modeli za ECAT (odjeljak 4.1), EPL (odjeljak 4.4) i PNIO (odjeljak 4.7) protokol, zatim odjeljci koji sadrže ekperimentalnu karakterizaciju pomenutih protokola (4.2, 4.5 i 4.8), te odjeljci 4.3, 4.6 i 4.9 u kojima su analizirane mogućnosti datih komunikacionih rješenja sa aspekta korišćenih metrika. Uporedna analiza ECAT, EPL i PNIO sistema za scenario mrežne konfiguracije srednje složenosti koja odgovara stvarnom sistemu automatizacije je data u odjeljku 4.10. Konačno, u odjeljku 4.11 su sumirani dobijeni rezultati i izvedeni osnovni zaključci.

4.1 Analitički model ECAT protokola

Kao što je prethodno opisano, ECAT princip rada podrazumijeva slanje sumacionog okvira svim podređenim čvorovima u mreži. S obzirom da ne postoje smjernice o tome koliki procenat propusnog opsega treba da se rezerviše za NRT saobraćaj, analitički model ECAT protokola će biti ograničen na opis minimalnog vremena komunikacionog ciklusa. U skladu sa definicijom, ovo vrijeme se može odrediti izrazom

$$T_{C(min)} = T_{frames} + T_{net}, \tag{4.1}$$

gdje je T_{frames} vrijeme neophodno za slanje jednog ili više ECAT okvira, dok se T_{net} odnosi na ukupna propagaciona kašnjenja mrežnih komponenata.

Bez umanjenja opštosti analize, može se smatrati da količina podataka koja se prenosi u sistemu (uključujući kontrolne informacije ECAT protokola) ne prelazi maksimalnu veličinu polja korisnih podataka *Ethernet* okvira (1500 bajtova), odnosno da je $T_{frames} = T_F$, gdje T_F odgovara vremenu potrebnom za slanje jednog ECAT okvira. Ovo je učinjeno iz razloga što način raspoređivanja telegrama u više ECAT okvira nije definisan specifikacijom i veoma zavisi od algoritma koji se koristi u konfiguracionom alatu. Kao što će kasnije biti pokazano, izborom odgovarajućeg algoritma raspoređivanja telegrama unutar ECAT okvira, može se postići određeni nivo optimizacije njegove veličine.

U opštem slučaju, ECAT okvir može da sadrži više telegrama. Prema tome, vrijeme potrebno za njegovo slanje može se odrediti korišćenjem jednačine

$$T_{frames} = T_F = \frac{L_F \cdot 8}{B_{tot}}, \tag{4.2}$$

gdje L_F predstavlja dužinu ECAT okvira izraženu brojem bajtova. Ova dužina se može odrediti

pomoću sljedeće formule:

$$L_F = \max\left[L_{ovh} + k(L_{th} + L_{wkc}) + L_{data}, L_{F(min)}, \right], \tag{4.3}$$

gdje je: k – ukupan broj telegrama, $L_{ovh} = 28$ – ukupna dužina kontrolnih informacija ECAT okvira (uključujući *Ethernet* kontrolne informacije), $L_{th} = 10$ – dužina zaglavlja telegrama, $L_{wkc} = 2$ – dužina WKC polja, L_{data} – ukupna veličina podataka koji se prenose telegramima i $L_{F(min)} = 64$ – dužina minimalnog *Ethernet* okvira.

Ukupan broj telegrama najviše zavisi od algoritma za mapiranje procesnih podataka u telegrame koji je primijenjen u konfiguracionom alatu, ali i od načina na koji su procesni podaci organizovani unutar uređaja, kao i skupa podržanih ECAT komandi. U većini slučajeva, ulazni procesni podaci se prenose u okviru jednog LRD telegrama potrebne veličine, dok se izlazni procesni podaci prenose odgovarajućim LWR telegramom. Ukoliko postoji simetrija između procesnih podataka za odgovarajući uređaj, može se koristiti efikasnija LRW komanda ukoliko je uređaj podržava. Ovom komandom, istim telegramom se uređaju dostavljaju izlazni podaci uz istovremeno preuzimanje ulaznih vrijednosti. Na taj način, ukupna veličina telegrama može se smanjiti i do 50% [82].

Ukoliko se radi o kompleksnijem uređaju, koji podržava mehanizme sinhronizacije čvorova i obezbjeđuje parametrizaciju putem CoE protokola, ECAT specifikacija definiše odgovarajuće telegrame neophodne za realizaciju pomenutih funkcionalnosti. U tabeli 4.1 dat je pregled telegrama koji se tipično koriste u stvarnim instalacijama.

Ukupno kašnjenje mrežnih komponenata zavisi od korišćene topologije i od karakteristika samih uređaja. S obzirom da se ECAT okvir obrađuje pri prolasku kroz čvor, vrijeme propagacije je svedeno na minimalnu vrijednost, koja zavisi od interne strukture ESC jedinice i fizičkog interfejsa za povezivanje uređaja.

4.1.1 Linijska topologija

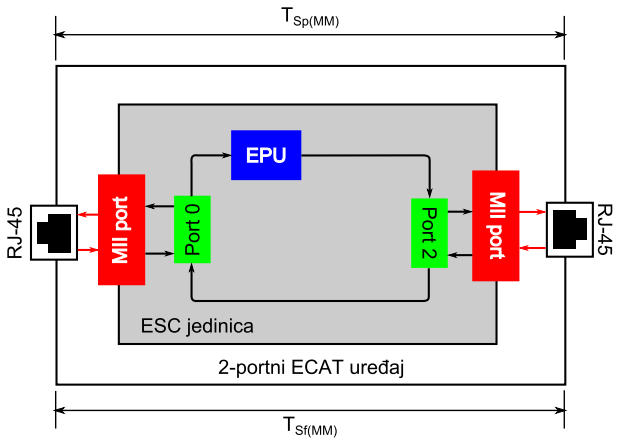
U slučaju klasične linijske topologije, kod koje se koriste kompaktni uređaji povezani standardnim mrežnim interfejsom (npr. RJ-45 priključak), ESC zahtijeva upotrebu posebnih komponenata na fizičkom sloju koje se nazivaju PHY komponente. Povezivanje se tipično obavlja preko standardizovanog MII (engl. *Media Independent Interface*) interfejsa kao što je ilustrovano na slici 4.1. Sa slike se može vidjeti da se, od ukupno četiri dostupna, koriste dva porta koja omogućavaju kaskadno povezivanje uređaja, čime se formira linijska topologija. Kašnjenje kroz uređaj je određeno zbirom kašnjenja u direktnom ($T_{Sp(MM)}$) i povratnom ($T_{Sf(MM)}$) smjeru

Tabela 4.1: Pregled ECAT telegrama koji se koriste u stvarnim instalacijama

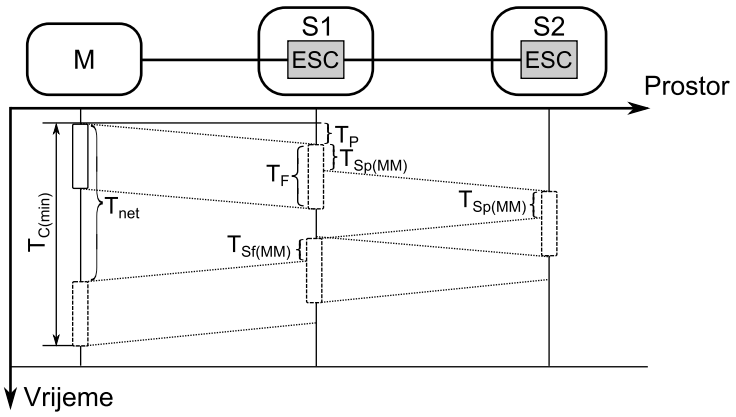
Komanda	Veličina	Opis
NOP	4 B	Koristi se za komande koje obezbjeđuju sinhronizaciju. Rezerviše prostor u ECAT okviru za stvarne komande koje se generišu tokom rada sistema.
ARMW	4 B	Distribuira lokalno vrijeme referentnog čvora ostalim čvorovima u mreži.
LRD	Promjenljiva	Prenosi informaciju o stanju CoE bafera. Veličina zavisi od broja uređaja koji podržavaju CoE komunikaciju (N_{CoE}) i može se izračunati formulom $8 \cdot \lceil N_{CoE}/8 \rceil$ (svakom uređaju se pridružuje jedan bit u telegramu).
LRW	Promjenljiva	Obezbjeđuje efikasan prenos ulazno/izlaznih procesnih podataka u oba smjera. Veličina zavisi od ukupne veličine procesnih podataka uređaja koji koriste ovu komandu.
LWR	Promjenljiva	Obezbjeđuje dostavljanje izlaznih procesnih podataka uređajima. Veličina zavisi od ukupne veličine procesnih podataka uređaja koji koriste ovu komandu.
LRD	Promjenljiva	Obezbjeđuje preuzimanje ulaznih procesnih podataka od uređaja. Veličina zavisi od ukupne veličine procesnih podataka uređaja koji koriste ovu komandu.
BRD	2 B	Čitanje stanja statusnih registara svih uređaja.

između MII portova uređaja. U slučaju da se drugi port uređaja ne koristi (npr. posljednji uređaj u linijskoj topologiji), kašnjenje uređaja je određeno samo kašnjenjem u direktnom smjeru.

Prostorno-vremenski dijagram komunikacije između jednog nadređenog i dva podređena uređaja vezana linijskom topologijom prikazan je na slici 4.2. Sa slike se vidi da ukupno propagaciono kašnjenje čine vrijeme propagacije kroz oba uređaja u direktnom smjeru, vrijeme propagacije kroz prvi uređaj u povratnom smjeru i vrijeme propagacije kroz kablove (T_P) koje iznosi oko 5 ns po metru korišćenog kabla, odnosno $T_{net} = 2T_{Sp(MM)} + T_{Sf(MM)} + 4T_P$. Ukoliko se prethodno razmatranje uopšti za n uređaja, dobija se opšti izraz za vrijeme propagacije kroz



Slika 4.1: Struktura standardnog ECAT uređaja sa dva MII porta



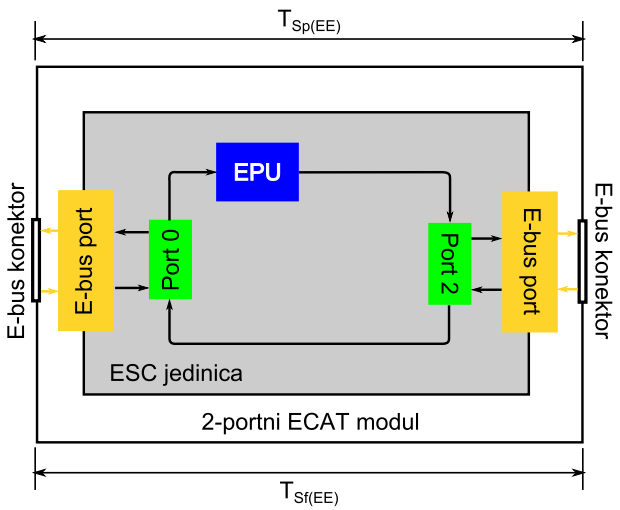
Slika 4.2: Prostorno-vremenski dijagram ECAT komunikacije

mrežu u obliku

$$T_{net}^{line} = \max\left[(n-1)T_{S(MM)} + T_{Sp(MM)} + 2nT_P, T_{ifg}\right], \tag{4.4}$$

gdje $T_{S(MM)} = T_{Sp(MM)} + T_{Sf(MM)}$ predstavlja ukupno kašnjenje kroz uređaj sa dva MII porta, dok T_{ifg} odgovara minimalnom razmaku između *Ethernet* okvira (960 ns u slučaju 100 Mb/s *Ethernet* komunikacije). Takođe, treba napomenuti da je pri izvođenju jednačine (4.4) pretpostavljeno da su svi uređaji povezani kablovima iste dužine.

Kao što je pomenuto u odjeljku 2.1, ECAT protokol podržava E-bus interfejs za povezivanje modula koji čine modularan uređaj, čija struktura u suštini odgovara linijskoj topologiji s tom razlikom da se uređaji ne povezuju klasičnim UTP (engl. *Unshielded Twisted Pair*) ili STP (engl. *Shielded Twisted Pair*) kablovima, već specijalnim E-bus konektorima. Ovo znači da se vrijeme propagacije signala kroz kablove može zanemariti. Struktura modula ovakvog uređaja prikazana je na slici 4.3.



Slika 4.3: Struktura ECAT modula sa dva E-bus porta

Kao što može da se vidi sa slike, kašnjenje kroz ovaj uređaj može se odrediti pomoću izraza

$$T_{S(EE)} = T_{Sp(EE)} + T_{Sf(EE)}, \tag{4.5}$$

gdje $T_{Sp(EE)}$ i $T_{Sf(EE)}$ predstavljaju kašnjenje kroz uređaj u direktnom i povratnom smjeru, redom. Slično kao kod kompaktnog uređaja sa MII interfejsom, izraz (4.5) sadrži samo kašnjenje u direktnom smjeru ako je drugi port zatvoren (tj. ako na port nije povezan naredni uređaj).

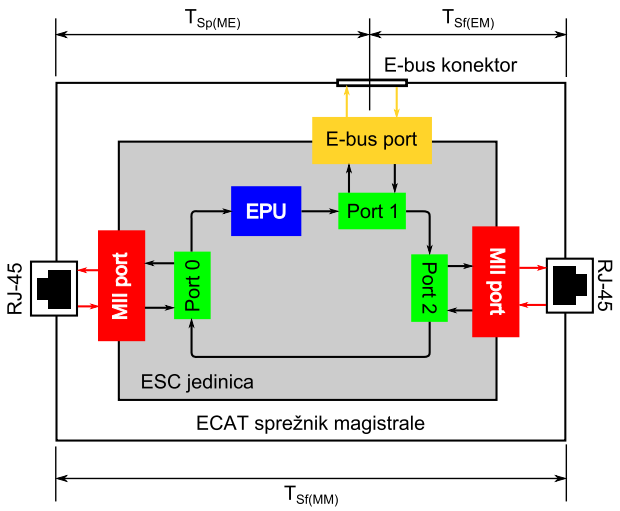
Važno je napomenuti da je neophodno obezbijediti povezivanje modula koji čine modularni uređaj sa ostatkom mreže putem standardnih UTP/STP kablova. U ovu svrhu se koristi poseban ECAT uređaj pod nazivom *sprežnik magistrale* (engl. *bus coupler*) koji ne prenosi procesne podatke i ima ulogu infrastrukturne komponente u ECAT sistemu. Struktura sprežnika magistrale sa dva MII i jednim E-bus portom, prikazana je na slici 4.4.

Ukupno kašnjenje kroz sprežnik magistrale određeno je izrazom

$$T_{S(ME)} = T_{Sp(ME)} + T_{Sf(EM)} + T_{Sf(MM)}, \tag{4.6}$$

gdje $T_{Sp(ME)}$ i $T_{Sf(EM)}$ predstavljaju kašnjenje kroz uređaj u direktnom i povratnom smjeru, redom, između MII i E-bus interfejsa. U slučaju da neki od portova nije otvoren, njegovo kašnjenje u izrazu (4.6) se izostavlja prilikom izračunavanja vrijednosti kašnjenja sprežnika magistrale.

Ukupno kašnjenje kroz mrežu, kod koje se koriste modularni uređaji, određuje se jednačinom (4.4), uzimajući u obzir činjenicu da je propagacija signala između dva modula unutar istog modularnog uređaja jednaka nuli.



Slika 4.4: Struktura ECAT sprežnika magistrale

4.1.2 Topologija zvijezde

S obzirom da ECAT uređaji ne interpretiraju polja MAC adrese prilikom obrade okvira, direktno korišćenje mrežnih komutatora za realizaciju topologije zvijezde nije podržano. S druge strane, fizička topologija zvijezde može se realizovati korišćenjem tzv. modularnog komutatora koji se dobija povezivanjem više ESC jedinica pomoću E-bus magistrale kao što je ilustrovano na slici 4.5.

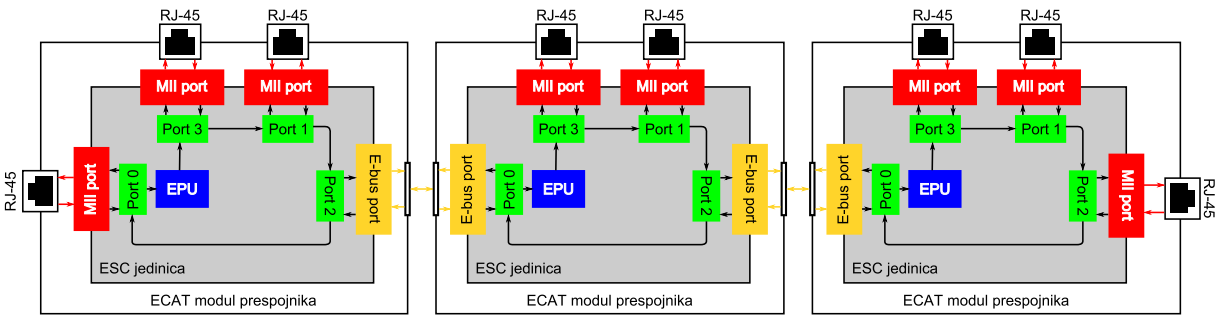
Ova slika prikazuje realizaciju modularnog komutatora sa osam MII portova, pri čemu su primjenjene tri ESC jedinice. Po potrebi, modularni komutator se može proširiti dodavanjem više modula, koji imaju istu strukturu kao srednja ESC jedinica, između krajnjih ESC jedinica. Svaki dodatni modul uvećava broj MII portova komutatora za dva, tj. vrijedi $N = 6 + 2m$, gdje je N broj portova modularnog komutatora, a m predstavlja ukupan broj srednjih modula.

Analizom strukture sa slike 4.5, ukupno propagaciono kašnjenje kroz modularni komutator je određeno sa:

$$\begin{aligned} T_{sw} &= T_{S(L)} + T_{S(M)} + T_{S(R)} = \left[T_{Sp(MM)} + T_{Sf(MM)} + T_{Sf(ME)} + T_{Sf(EM)}\right] \\ &+ \left[T_{Sp(EM)} + T_{Sf(MM)} + T_{Sf(ME)} + T_{Sf(EE)}\right] \\ &+ \left[T_{Sp(EM)} + T_{Sf(MM)} + T_{Sf(MM)} + T_{Sf(ME)}\right] \\ &= T_{Sp(MM)} + 4T_{Sf(MM)} + 2T_{Sp(EM)} + T_{Sf(EM)} + 3T_{Sf(ME)} + T_{Sf(EE)}, \end{aligned} \tag{4.7}$$

gdje $T_{S(L)}$, $T_{S(M)}$ i $T_{S(R)}$ određuju, redom, propagaciono kašnjenje kroz lijevu, srednju i desnu ESC jedinicu u modularnom komutatoru.

U opštem slučaju, kada se koristi m srednjih modula unutar komutatora, dobija se sljedeći



Slika 4.5: Struktura modularnog ECAT komutatora sa 8 portova

izraz:

$$\begin{aligned}
T_{sw} &= T_{S(L)} + mT_{S(M)} + T_{S(R)} = \left[T_{Sp(MM)} + T_{Sf(MM)} + T_{Sf(ME)} + T_{Sf(EM)}\right] \\
&+ m\left[T_{Sp(EM)} + T_{Sf(MM)} + T_{Sf(ME)} + T_{Sf(EE)}\right] \\
&+ \left[T_{Sp(EM)} + T_{Sf(MM)} + T_{Sf(MM)} + T_{Sf(ME)}\right] \\
&= T_{Sp(MM)} + (m+3)T_{Sf(MM)} + (m+1)T_{Sp(EM)} + T_{Sf(EM)} \\
&+ (m+2)T_{Sf(ME)} + mT_{Sf(EE)}.
\end{aligned} \tag{4.8}$$

Konačno, ako pretpostavimo simetričnu strukturu ESC jedinice, tj. da je zadovoljena jednačina $T_{Sf(ME)} = T_{Sf(EM)}$, dobija se

$$T_{sw} = T_{Sp(MM)} + (m+3)T_{Sf(MM)} + (m+1)T_{Sp(EM)} + (m+3)T_{Sf(EM)} + mT_{Sf(EE)}. \tag{4.9}$$

Korišćenjem jednačine (4.9), ukupno propagaciono kašnjenje kroz mrežu sa n krajnjih uređaja povezanih topologijom zvijezde, određeno je sa:

$$T_{net}^{star} = \max\left[2(n+1)T_P + T_{sw} + nT_{Sp(MM)}, T_{ifg}\right]. \tag{4.10}$$

Važno je napomenuti da član T_{sw} zavisi od broja korišćenih portova modularnog komutatora ($n+1$ u datom slučaju) i da krajnji uređaji unose kašnjenje samo u direktnom smjeru. Kao i u slučaju linijske topologije, pretpostavljeno je da su svi uređaji povezani sa modularnim komutatorom kablovima iste dužine.

Iako standardni mrežni komutatori nisu direktno podržani u ECAT sistemu, moguće je konfigurisati mrežu tako da se kreiraju zasebni okviri kojima se dostavljaju podaci svakom mrežnom segmentu koji je povezan na odgovarajući port komutatora. U tu svrhu se mogu koristiti upravljivi mrežni komutatori (engl. *managed network switch*), koji imaju mogućnost interpretacije i kreiranja VLAN oznaka. Na taj način se ECAT mreža logički segmentira u više odvojenih ECAT mreža. Broj okvira odgovara broju mrežnih segmenata, odnosno broju korišćenih portova komutatora. Ukoliko se za realizaciju topologije zvijezde u ECAT sistemu koristi konvencionalni mrežni komutator, ukupno vrijeme propagacije signala kroz mrežu je takođe određeno izrazom (4.10). Razlika je u tome što se član T_{sw} određuje formulom

$$T_{sw} = T_{fabric} + T_{store} + T_{queue}, \tag{4.11}$$

gdje je: T_{fabric} – vrijeme prenosa okvira kroz logiku za prosljeđivanje (tipična vrijednost 5-10 μs), T_{store} – vrijeme potrebno za pohranjivanje okvira koje odgovara njegovoj dužini, i T_{queue} – vrijeme zadržavanja okvira u redu čekanja određenog izlaznog porta.

Vrijeme zadržavanja okvira postoji samo ako se dva (ili više) pristigla okvira istovremeno prosljeđuju na isti izlazni port. Ovo vrijeme zavisi od opterećenja mreže i ne može se deterministički odrediti. Pažljivim planiranjem mreže i kontrolom saobraćaja, ono se može u potpunosti eliminisati. U slučaju ECAT sistema, s obzirom da okvire generiše jedan uređaj, vrijeme čekanja je uvijek jednako nuli.

Važno je napomenuti da, s obzirom da je potrebno slati više odvojenih okvira (za svaki mrežni segment ili uređaj povezan na komutator), se ne može koristiti formula (4.2) kod ECAT mreže sa topologijom zvijezde i standardnim komutatorom, već se mora koristiti formula

$$T_{frames} = \sum_{i=1}^{n} T_F^i + (n-1)T_{ifg}, \tag{4.12}$$

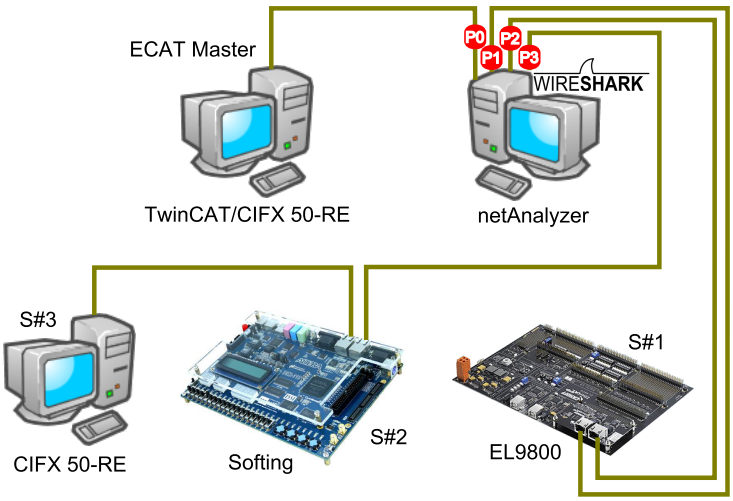
gdje T_F^i predstavlja vrijeme potrebno za slanje ECAT okvira i-tom uređaju. U slučaju homogene mreže, kod koje svi uređaji imaju istu veličinu podataka i istu strukturu ECAT okvira, tj. gdje vrijedi $T_F^i = T_F$ za $i = 1, .., n$; ovo vrijeme se može trivijalno izraziti kao $T_{frames} = nT_F + (n-1)T_{ifg}$. Očigledno, s ciljem smanjenja vremena T_{frames}, okviri se šalju uzastopno jedan iza drugog sa obaveznim razmakom između okvira (član T_{ifg}) definisanim standardom. Konačno, za izračunavanje vremena kašnjenja mreže može se koristiti jednačina (4.10) uz napomenu da se kod standardnog komutatora član T_{sw} mora uračunati dva puta jer okviri prolaze dva puta kroz komutator.

4.2 Karakterizacija vremenskih parametara ECAT analitičkog modela

Na osnovu prethodnog razmatranja, može se zaključiti da izvedeni analitički model ECAT protokola sadrži različite vremenske parametre koji uglavnom predstavljaju kašnjenja kroz mrežne komponente i vrijeme potrebno za slanje okvira. Iz jednačine (4.2) je očigledno da se vrijeme slanja okvira može deterministički odrediti ako je poznata njegova dužina L_F. S druge strane, većina vremenskih parametara, koji se odnose na kašnjenja mrežnih komponenata, ima nedeterministički karakter, što znači da postoji varijacija kašnjenja unutar komponente koja zavisi od različitih faktora.

Karakterizacija vremenskih parametara ECAT analitičkog modela podrazumijeva postupak statističkog opisa ovih parametara, kroz analizu rezultata dobijenih mjerenjem na odgovarajućoj eksperimentalnoj postavci ECAT mreže. U prethodnom odjeljku je objašnjeno da postoje dvije vrste ECAT uređaja: modularni uređaji, čiji se moduli (terminali) povezuju E-bus magistralom, i kompaktni uređaji, koji se povezuju standardnim mrežnim kablovima. Za mjerenje relevantnih vremenskih parametara kompaktnih uređaja, korišćena je eksperimentalna postavka prikazana na slici 4.6.

U datom slučaju, analizirana su kašnjenja uređaja tri različita proizvođača: *Beckhoff* razvojni sistem EL9800 (baziran na FB1111 modulu), *Softing* implementacija na razvojnoj ploči Altera DE2-115 i *Hilscher* implementacija na PCI kartici CIFX 50-RE koja se ugrađuje u standardan PC računar. Kao nadređeni uređaj, korišćene su dvije implementacije: *Beckhoff TwinCAT* softverska realizacija na standardnom PC računaru sa *Windows XP* operativnim sistemom i *Hilscher* implementacija na prethodno pomenutoj PCI kartici. Svi uređaji su povezani kablovima dužine 2 m, što znači da se propagacija signala kroz kablove može zanemariti. Za mjerenje vremenskih parametara korišćen je *Hilscher netAnalyzer*, specijalna mjerna sonda, koja omogućava preciznu evaluaciju RTE sistema, sa greškom koja ne prelazi 10 ns. Ova sonda se ugrađuje kao PCI kartica u standardni PC računar, pri čemu se za prikupljanje i analizu rezultata mjerenja koristi poseban softver koji dolazi uz samu karticu. U konkretnom slučaju, u cilju smanjenja uticaja dijela za mjerenje na ostatak mreže, korišćen je poseban PC računar za mjerenje vremenskih parametara, odvojen od ostatka postavke kao što je prikazano na slici 4.6. Strukturu *netAnalyzer* kartice čine četiri *Ethernet* porta (P0–P3). Portovi P0 i P1 (kao i P2 i P3) su interno povezani korišćenjem pasivnog premosnika (engl. *passive tap*), što znači da se okviri primljeni na P0 (P2) preslikavaju na P1 (P3), i obrnuto, sa nultim kašnjenjem. Trajanje svakog eksperimenta je



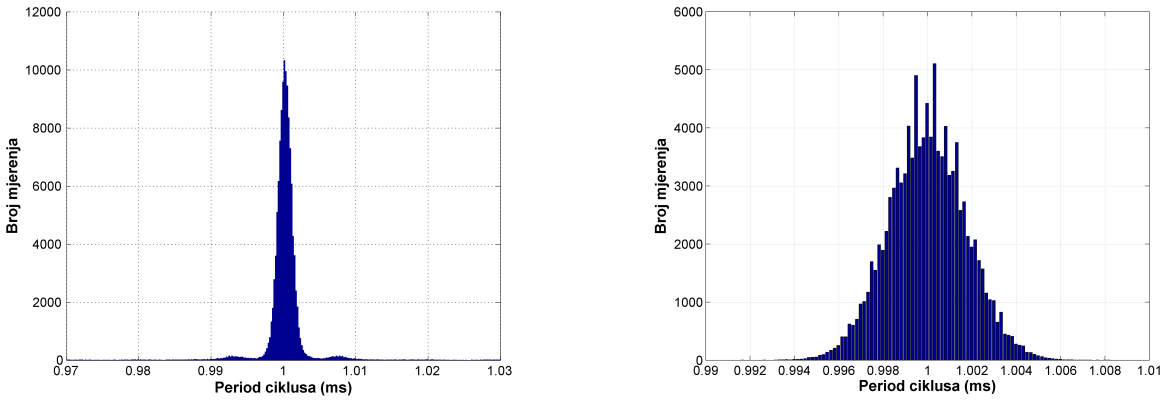
Slika 4.6: Eksperimentalna postavka ECAT mreže

postavljeno tako da se dobije najmanje 30000 mjerenja.

Prvi skup mjerenja je imao za cilj da karakterizuje ciklično ponašanje nadređenog uređaja za dvije pomenute implementacije. U tu svrhu, analiziran je ciklični saobraćaj koji dolazi od nadređenog uređaja (port P0). Sve asinhrone poruke, koje se šalju sporadično kako bi se dobila informacija o stanju sistema, su filtrirane kako bi se otklonio njihov uticaj na statističku analizu dobijenih rezultata. Period ponavljanja cikličnih okvira odgovara vremenu ciklusa (T_C) koje je podešeno u okviru odgovarajućeg softvera za konfiguraciju mreže, a samim tim i podređenog uređaja. U konkretnom slučaju, ova vrijednost je podešena tako da bude jednaka 1 ms. Vrijednost izmjerenog perioda se može odrediti ako se oduzme vrijeme prijema dva uzastopna ciklična okvira na portu P0. Kvalitativni rezultati za dvije implementacije nadređenog uređaja su prikazani na slici 4.7, dok je pregled statistike dat u tabeli 4.2.

Dobijeni rezultati pokazuju da, kod obje implementacije, vrijeme komunikacionog ciklusa odgovara konfigurisanoj vrijednosti (srednja vrijednost je veoma blizu 1 ms). Međutim, znatno bolja preciznost se postiže kod CIFX implementacije. Ovo se može objasniti samim načinom implementacije jednog i drugog rješenja. Naime, *TwinCAT* implementacija je softverska implementacija bazirana na optimizaciji softverskih komponenti standardnog PC računara u okviru *Windows XP* operativnog sistema, dok se CIFX bazira na specifičnom hardveru, koji se nalazi u okviru PCI kartice i koji obavlja vremenski kritične funkcionalnosti ECAT komunikacionog steka. Posljedica toga je za red veličine manja vrijednost apsolutnog odstupanja.

Nakon kvalitativne i kvantitativne analize vremena komunikacionog ciklusa, realizovan je skup eksperimenata s ciljem da se procijeni kašnjenje koje unose podređeni uređaji različitih proizvođača. Kao što je opisano u prethodnom odjeljku, svaki podređeni uređaj unosi kašnjenje



(a) (b)

Slika 4.7: Izmjereni period komunikacionog ciklusa za (a) *TwinCAT* i (b) CIFX implementaciju nadređenog uređaja

Tabela 4.2: Pregled statistike mjerenih vremenskih parametara

Parametar	Min.	Max.	μ	σ
$T_C^{Beckhoff}$	899.23	1099.99	1000.209	6.812
$T_C^{Hilscher}$	991.56	1008.24	999.998	1.736
$T_{Sp(MM)}^{Beckhoff}$	0.623	0.673	0.647	0.012
$T_{Sf(MM)}^{Beckhoff}$	0.571	0.621	0.601	0.012
$T_{Sp(MM)}^{Softing}$	0.695	0.785	0.744	0.017
$T_{Sf(MM)}^{Softing}$	0.631	0.711	0.671	0.017
$T_{Sp(MM)}^{Hilscher}$	1.109	1.179	1.145	0.014
$T_{Sf(MM)}^{Hilscher}$	1.063	1.133	1.108	0.012

Sve vrijednosti su izražene u μs

u direktnom i povratnom smjeru. Sa slike 4.6, može da se vidi da je kašnjenje EL9800 uređaja u direktnom smjeru ($T_{Sp(MM)}$) određeno razlikom vremena prijema istog okvira na portovima P2 i P0. Ekvivalentno, kašnjenje u povratnom smjeru ($T_{Sf(MM)}$) se može odrediti ako se vrijeme prijema okvira na portu P3 oduzme od vremena prijema istog okvira na portu P1. Kašnjenje ostala dva uređaja može se dobiti na isti način ako se mreža modifikuje tako da su portovi uređaja čija se kašnjenja mjere povezana na portove *netAnalyzer* kartice kao kod EL9800 uređaja. Rezultati dobijeni mjerenjem za tri uređaja prikazani su na slikama 4.8–4.10, dok su rezultati kvantitativne analize dati u tabeli 4.2.

Dobijeni rezultati pokazuju da je kašnjenje kod svih uređaja veoma precizno sa apsolutnim odstupanjem ispod 100 ns. Međutim, srednja vrijednost kašnjenja razlikuje se od uređaja do uređaja. Najveće kašnjenje unosi *Hilscher* implementacija bazirana na CIFX rješenju (preko 1 μs u oba smjera), dok najbolje karakteristike postiže *Beckhoff* implementacija podređenog uređaja (i po preciznosti i po srednjoj vrijednosti kašnjenja).

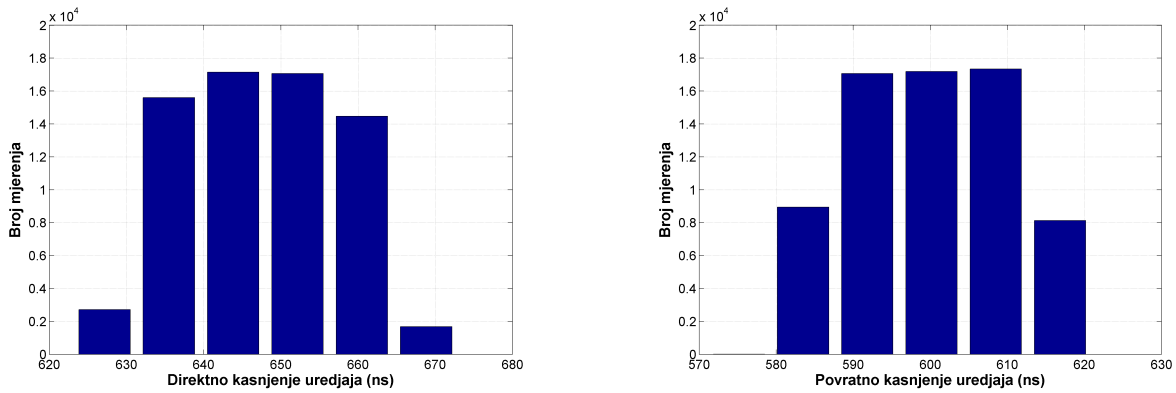
Osim kompaktnih uređaja, ECAT protokol definiše i modularnu strukturu uređaja. Kao što je već pomenuto, modularni uređaj se sastoji iz modula koji se u ECAT terminologiji nazivaju terminali (slika 4.3). Terminali se međusobno povezuju putem E-bus interfejsa koji predstavlja internu magistralu modularnog uređaja. Za povezivanje modularnog uređaja sa ostatkom mreže koristi se posebna vrsta terminala koji se naziva sprežnik magistrale (slika 4.4). Takođe, kao što je već objašnjeno, korišćenjem odgovarajućih terminala može se realizovati modularni komutator, koji omogućava primjenu topologije zvijezde. Prema tome, veoma je važno analizirati kašnjenja koja unose terminali koji čine strukturu modularnog uređaja. Eksperimentalna postavka pogodna za mjerenje kašnjenja terminala modularnog uređaja je prezentovana na slici 4.11. Sve

komponente ECAT mreže, u prikazanoj postavci, čine terminali proizvođača *Beckhoff*.

Na slici se uočava da strukturu modularnog uređaja čini sprežnik magistrale (EK1100), određeni broj ulazno/izlaznih terminala (u primjeru sa slike EL2002 predstavlja izlazni, a EL1012 ulazni terminal) povezanih E-bus interfejsom i terminal 2-portnog komutatora (EK1122), kao završni terminal u modularnom uređaju. Struktura terminala 2-portnog komutatora odgovara strukturi srednjeg modula modularnog komutatora sa slike 4.5. Kada se ne koristi drugi E-bus port (Port 2 srednjeg modula sa slike 4.5), interna struktura ovog terminala je u potpunosti ekvivalentna strukturi sprežnika magistrale (slika 4.4), pa se može smatrati da je veza EK1100-EK1122 u prikazanoj konfiguraciji potpuno simetrična. Ulazno/izlazni terminali su prikazani crtkano zato što, prilikom mjerenja, nije korišćena fiksna, već promjenljiva konfiguracija sa varijabilnim rasporedom i brojem, kao i tipom korišćenih terminala. Takođe, terminal označen kao *Term 5* služi samo za kompletiranje logičkog prstena, kako bi se moglo mjeriti kašnjenje modularnog uređaja u direktnom i povratnom smjeru.

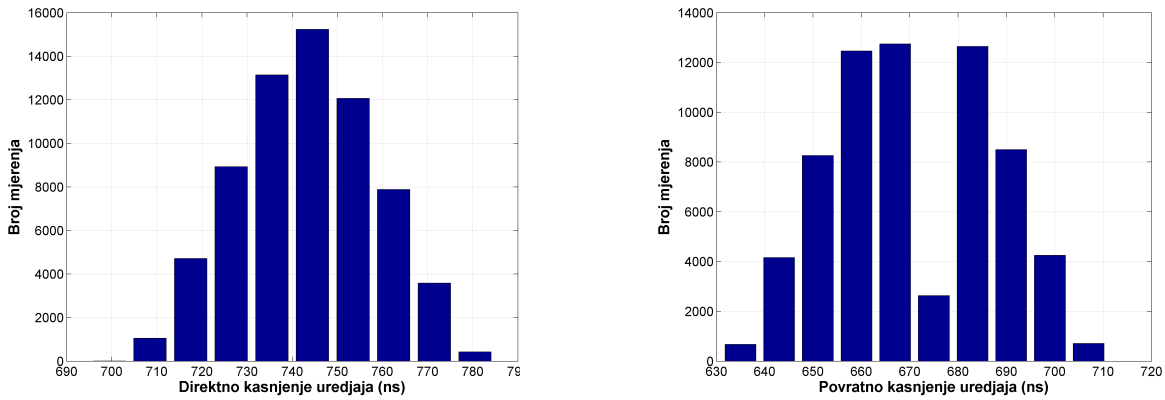
U cilju karakterizacije kašnjenja terminala modularnih uređaja, obavljeno je nekoliko eksperimenata na postavci sa slike 4.11. Prvo mjerenje podrazumijeva procjenu kašnjenja koje unose sprežnik magistrale i terminal 2-portnog komutatora. Preciznije, mjerenje kašnjenja u direktnom i povratnom smjeru je realizovano kada nema ulazno/izlaznih terminala, tj. sprežnik magistrale i terminal 2-portnog komutatora su direktno povezani E-bus interfejsom (EK1100-EK1122 struktura). Kašnjenje, u oba smjera, može se odrediti na isti način kao kod kompaktnih uređaja (slika 4.6). Rezultati mjerenja za oba smjera su prikazani na slici 4.12, dok je detaljnija kvantifikacija data u tabeli 4.3.

S obzirom na pretpostavljenu simetriju modularne strukture, moguće je procijeniti vremenske parametre kašnjenja sprežnika magistrale. Naime, kako je ukupno kašnjenje u direktnom



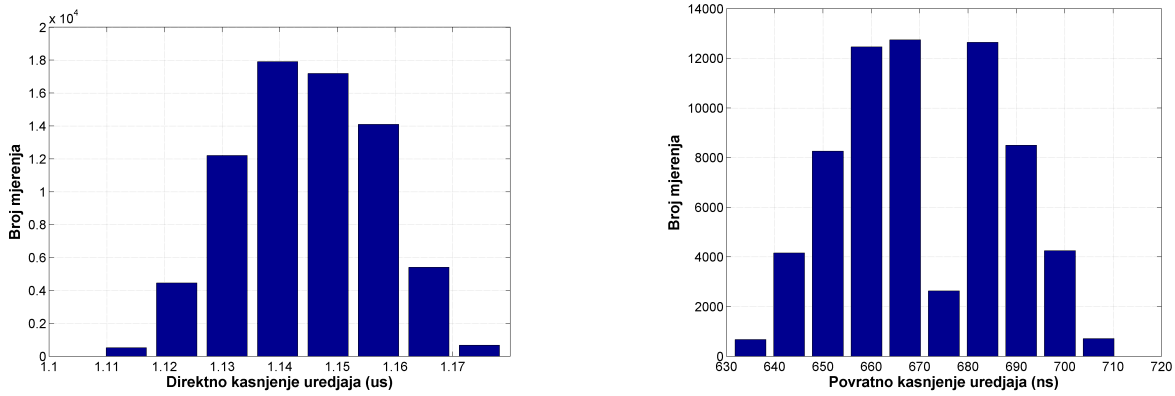
(a) (b)

Slika 4.8: Kašnjenje *Beckhoff* uređaja u (a) direktnom i (b) povratnom smjeru



(a) (b)

Slika 4.9: Kašnjenje *Softing* uređaja u (a) direktnom i (b) povratnom smjeru



(a) (b)

Slika 4.10: Kašnjenje *Hilscher* uređaja u (a) direktnom i (b) povratnom smjeru

i povratnom smjeru određeno zbirom pojedinačnih kašnjenja, uz pretpostavku da slučajne promjenljive T_{SpME} i $T_{Sp(EM)}$, odnosno T_{SfME} i $T_{Sf(EM)}$, imaju istu raspodjelu vjerovatnoće sa identičnim parametrima, možemo smatrati da vrijedi:

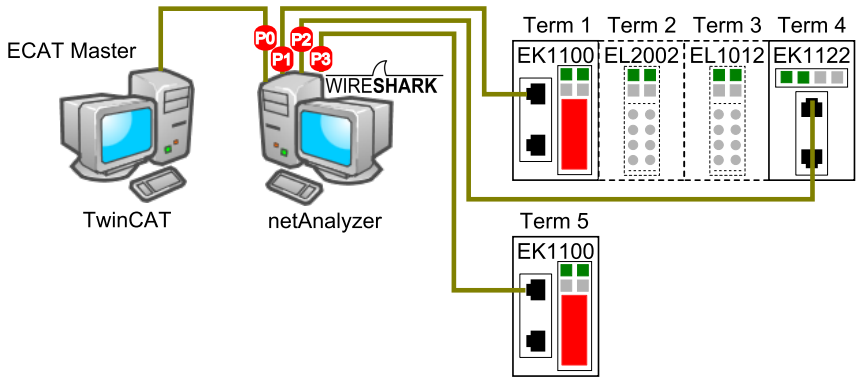
$$\mu_{Mp} = \mu_{T_{Sp(ME)}} + \mu_{T_{Sp(EM)}} = 2\mu_{T_{Sp(EM)}}, \tag{4.13}$$

$$\sigma^2_{Mp} = \sigma^2_{T_{Sp(ME)}} + \sigma^2_{T_{Sp(EM)}} = 2\sigma^2_{T_{Sp(EM)}}, \tag{4.14}$$

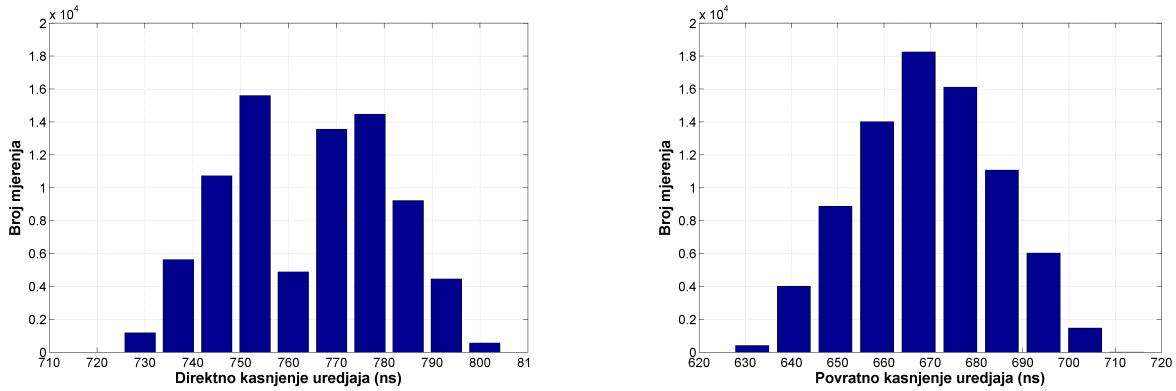
$$\mu_{Mf} = \mu_{T_{Sf(ME)}} + \mu_{T_{Sf(EM)}} = 2\mu_{T_{Sf(EM)}}, \tag{4.15}$$

$$\sigma^2_{Mf} = \sigma^2_{T_{Sf(ME)}} + \sigma^2_{T_{Sf(EM)}} = 2\sigma^2_{T_{Sf(EM)}}, \tag{4.16}$$

gdje μ_{Mx} i σ_{Mx} predstavljaju srednje vrijednosti i standardne devijacije, redom, dobijene nakon statističke analize rezultata mjerenja (tabela 4.3). Korišćenjem izvedenih izraza mogu se izračunati srednje vrijednosti i standardne devijacije za vremenske parametre $T_{Sp(EM)}$ i $T_{Sf(EM)}$ koji



Slika 4.11: Eksperimentalna postavka ECAT mreže sa modularnim uređajem



(a) (b)

Slika 4.12: Kašnjenje modularne strukture EK1100-EK1122 u (a) direktnom i (b) povratnom smjeru

se koriste u izvedenom analitičkom modelu. Izračunate vrijednosti date su u tabeli 4.4. Iz tabele se vidi da je srednja vrijednost kašnjenja između MII i E-bus interfejsa za oko 40% manja u odnosu na strukturu sa dva MII porta (tabela 4.2), dok je standardna devijacija ista u oba slučaja.

Nakon mjerenja kašnjenja koja unose sprežnik magistrale i terminal 2-portnog komutatora, može se pristupiti mjerenju, odnosno procjeni kašnjenja ulazno/izlaznih terminala, koji čine strukturu modularnog uređaja. Da bi se procijenio trend kašnjenja u oba smjera, kao i zavisnost kašnjenja od vrste terminala, obavljeno je nekoliko eksperimenata sa različitim vrstama i brojem terminala kao što je prikazano u tabeli 4.3. Iz tabele je vidljivo da su korišćeni obični digitalni ulazni (EL1012) i izlazni (EL2002) terminali, digitalni ulazni (E1252) i izlazni (EL2252) terminali sa sinhronizacijom, te analogni ulazni (EL3102) i izlazni (EL4132) terminali. Kvalitativni rezultati mjerenja za strukture EK1100-EL2002-EK1122 i EK1100-EL2002-EL1012-EK1122 prikazani su na slikama 4.13 i 4.14, redom. Za ostale konfiguracije se dobijaju kvalitativno slični rezultati pa su odgovarajući grafici izostavljeni.

Sličnim rezonovanjem kao i u slučaju procjene vremenskih parametara $T_{Sp(EM)}$ i $T_{Sf(EM)}$,

Tabela 4.3: Pregled statistike mjerenih vremenskih parametara

Modularna struktura	Min.	Max.	μ	σ
Direktan smjer				
EK1100-EK1122	0.725	0.805	0.764	0.017
EK1100-18T-EK1122	3.465	3.605	3.538	0.021
EK1100-EL2002-EK1122	0.845	0.945	0.894	0.017
EK1100-EL2002-EL1012-EK1122	1.035	1.125	1.081	0.017
EK1100-EL2252-EK1122	0.875	0.975	0.926	0.017
EK1100-EL2252-EL1252-EK1122	1.015	1.115	1.064	0.017
EK1100-EL4132-EK1122	0.845	0.945	0.895	0.017
EK1100-EL4132-EL3102-EK1122	1.015	1.115	1.066	0.017
Povratni smjer				
EK1100-EK1122	0.627	0.717	0.669	0.017
EK1100-18T-EK1122	3.027	3.177	3.105	0.021
EK1100-EL2002-EK1122	0.757	0.847	0.803	0.017
EK1100-EL2002-EL1012-EK1122	0.887	0.987	0.937	0.017
EK1100-EL2252-EK1122	0.787	0.877	0.835	0.017
EK1100-EL2252-EL1252-EK1122	0.907	1.017	0.961	0.017
EK1100-EL4132-EK1122	0.767	0.857	0.812	0.017
EK1100-EL4132-EL3102-EK1122	0.907	1.017	0.962	0.017

Sve vrijednosti su izražene u μs

Tabela 4.4: Procjena vremenskih parametara terminala

Parametar	μ	σ
$T_{Sp(EM)}$	0.382	0.012
$T_{Sf(EM)}$	0.335	0.012
$T_{Sp(EE)}$	0.154	0.003
$T_{Sf(EE)}$	0.135	0.003

Sve vrijednosti su izražene u μs

mogu se izvesti ekvivalentne jednačine za vremenske parametre $T_{Sp(EE)}$ i $T_{Sf(EE)}$:

$$\mu_{Mp} = 2\mu_{T_{Sp(EM)}} + k\mu_{T_{Sp(EE)}}, \tag{4.17}$$

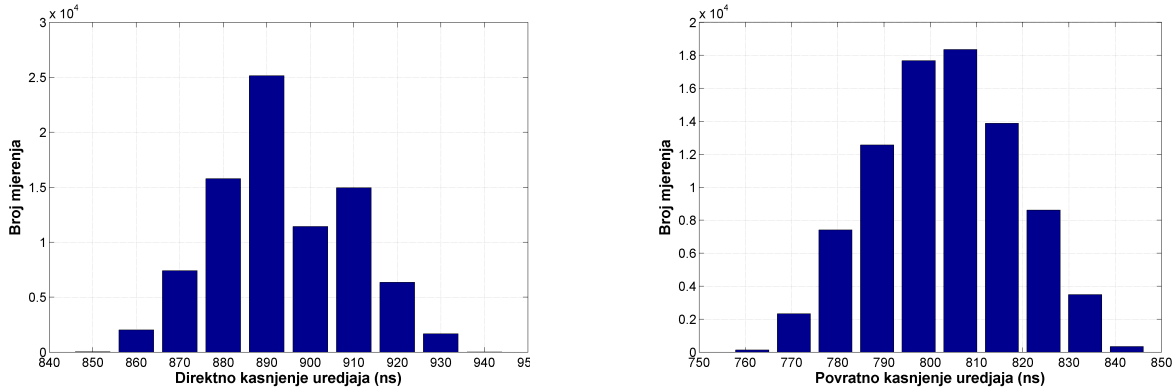
$$\sigma^2_{Mp} = 2\sigma^2_{T_{Sp(EM)}} + k\sigma^2_{T_{Sp(EE)}}, \tag{4.18}$$

$$\mu_{Mf} = 2\mu_{T_{Sf(EM)}} + k\mu_{T_{Sf(EE)}} \tag{4.19}$$

$$\sigma^2_{Mf} = 2\sigma^2_{T_{Sf(EM)}} + k\sigma^2_{T_{Sf(EE)}}, \tag{4.20}$$

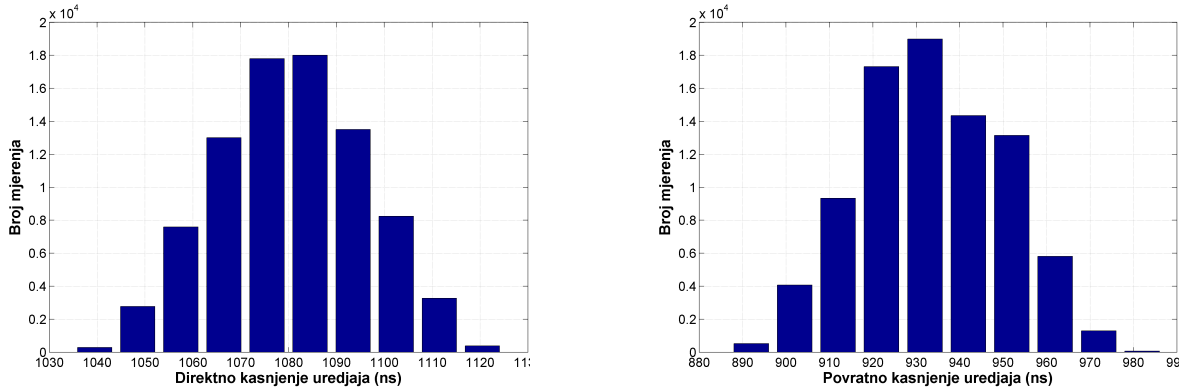
gdje k određuje broj ulazno/izlaznih terminala.

Uvrštavanjem rezultata mjerenja iz tabele 4.3 u prethodno izvedene jednačine, može se



(a) (b)

Slika 4.13: Kašnjenje modularne strukture EK1100-EL2002-EK1122 u (a) direktnom i (b) povratnom smjeru



(a) (b)

Slika 4.14: Kašnjenje modularne strukture EK1100-EL2002-EL1012-EK1122 u (a) direktnom i (b) povratnom smjeru

zaključiti da srednja vrijednost kašnjenja terminala varira između 0.13 i 0.162 μs u direktnom, odnosno 0.134 i 0.166 μs u povratnom smjeru, dok se standardna devijacija uopšte ne može procijenti, jer je njena promjena tako mala kada se dodaju jedan ili dva terminala, da se ne može detektovati korišćenom mjernom opremom.

Varijacija srednje vrijednosti može se objasniti činjenicom da PHY komponenta na prijemnoj strani može da unese dodatno kašnjenje prilikom uspostavljanja linka zbog različitog poravnanja faze između takt signala 25 MHz i 125 MHz, kao što je opisano u [83]. Ovo kašnjenje se manifestuje u vidu rastera od 5 pozicija čije je međusobno rastojanje 8 ns. Na taj način se dobija konstantan pomjeraj ± 16 ns u odnosu na nominalno kašnjenje PHY komponente čija je vrijednost različita pri svakom uspostavljanju linka.

Opisani efekti mogu se, do određene mjere, smanjiti ako se prilikom mjerenja koristi veći broj ulazno/izlaznih terminala. U tu svrhu je izmjereno kašnjenje modularnog uređaja kada

se koristi 18 ulazno/izlaznih terminala (konfiguracija EK1100-18T-EK1122). Kvantitativni rezultati mjerenja su dati u tabeli 4.3, dok je procjena vremenskih parametara $T_{Sp(EE)}$ i $T_{Sf(EE)}$ izračunatih korišćenjem jednačina (4.17)–(4.20) prikazana u tabeli 4.4. Iz tabele se vidi da je srednja vrijednost kašnjenja za skoro 60% manja u odnosu na kašnjenja sprežnika magistrale, dok standardna devijacija iznosi samo 3 ns.

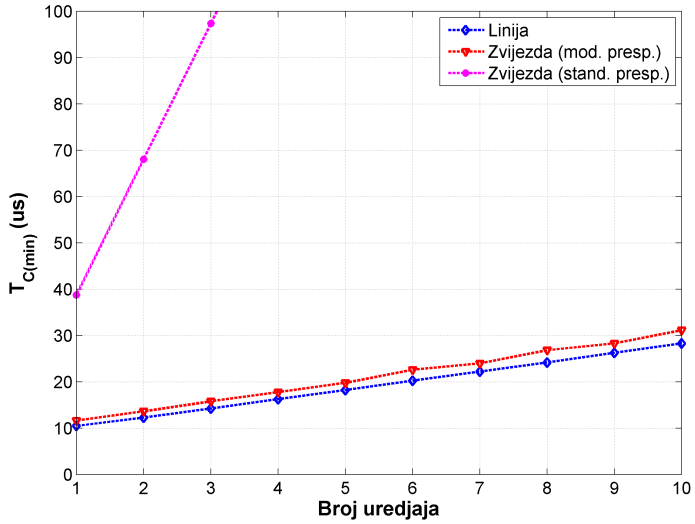
4.3 Teorijska analiza ECAT protokola

Izvedeni teorijski analitički model može se iskoristiti za procjenu mogućnosti komunikacionog sistema u skladu sa prethodno definisanim metrikama. Ovo omogućava identifikaciju strategija pomoću kojih je moguće postići određena poboljšanja. S obzirom da vrijeme komunikacionog ciklusa dominantno utiče na vrijeme odziva sistema, sve analizirane metrike kao krajnji cilj imaju procjenu ovog parametra. S tim u vezi, minimalno vrijeme ciklusa ($T_{C(min)}$) predstavlja prvu metriku koja može kvantitativno da odredi mogućnosti ECAT sistema.

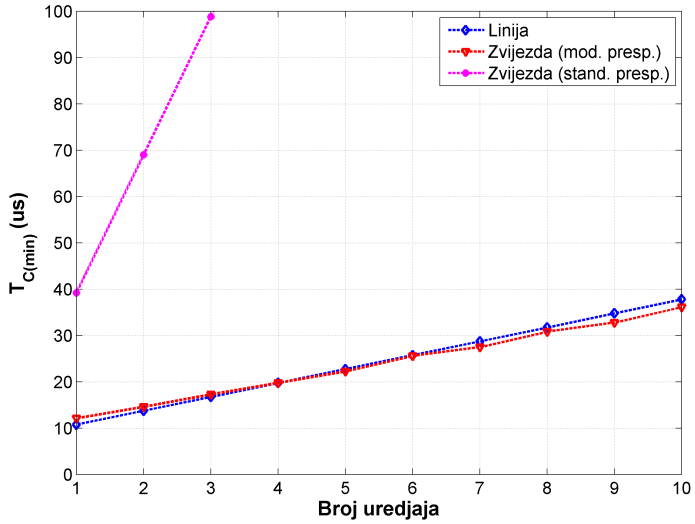
S obzirom da analitički model ECAT protokola podržava dvije osnovne topologije, linijsku i topologiju zvijezde, analiza u ostatku poglavlja će biti koncentrisana na ispitivanje definisanih metrika za ove dvije topologije. Osim toga, važno je napomenuti da vrsta uređaja značajno utiče na performanse sistema pa su, u modelu, korišćene srednje vrijednosti vremenskih parametara za dvije implementacije, *Beckhoff* i *Hilscher*, prezentovane u prethodnom odjeljku.

Prilikom svake analize korišćene su sljedeće pretpostavke: svi uređaji su povezani kablovima dužine 10 m, ukupna veličina procesnih podataka je 8 bajtova sa 75% asimetrijom između ulaza i izlaza (tj. veličina ulaznih podataka je 3 puta veća od veličine izlaznih podataka), maksimalan broj uređaja u mreži je ograničen na 10, svi uređaji podržavaju CoE komunikaciju (koristi se dodatni LRD telegram za čitanje statusa CoE bafera) i omogućena je sinhronizacija između čvorova, što podrazumijeva dodatne telegrame u cikličnom okviru za ovu svrhu. Asimetrija između ulaza i izlaza je uvedena kako bi se forsirala razmjena podataka u odvojenim LRD i LWR telegramima. Broj uređaja je ograničen na 10 kako bi se obezbijedila usklađenost sa rezultatima za druga dva protokola, jer EPL i PNIO protokoli, koji će biti analizirani u nastavku ovog poglavlja, uvode ograničenje po pitanju broja infrastrukturnih komponenata kao što je objašnjeno u odjeljku 3.1.9. Za realizaciju topologije zvijezde, korišćen je modularni komutator opisan u odjeljku 4.1 i standardni komutator sa pohranjivanjem prije prosljeđivanja *D-Link DES-1100-16* čija je srednja vrijednost kašnjenja 4.65 μs [84].

U prvoj analizi kvantifikovano je minimalno vrijeme komunikacionog ciklusa za različite strukture mreže u zavisnosti od broja uređaja. Rezultati ove analize za dvije različte imple-



(a)

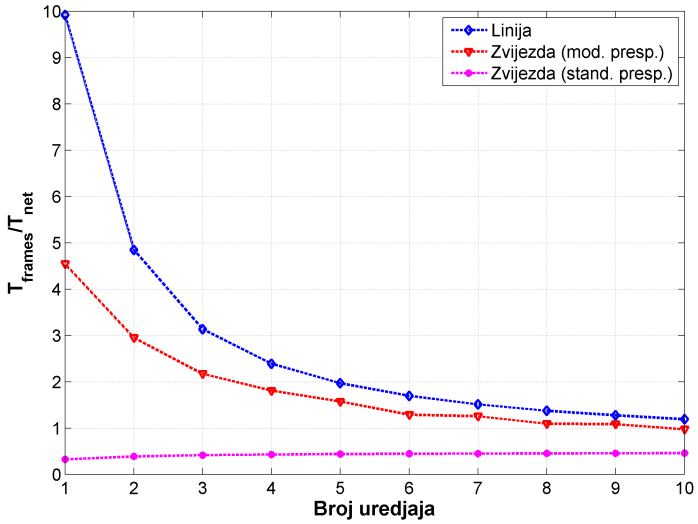


(b)

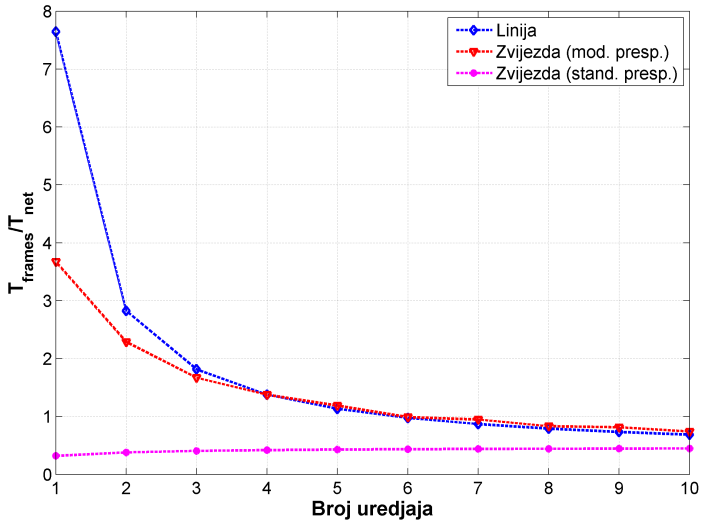
Slika 4.15: $T_{C(min)}$ u funkciji broja uređaja za različite topologije: (a) *Beckhoff* i (b) *Hilscher* implementacija

mentacije uređaja prikazani su na slici 4.15. Analiza je sprovedena za dvije strukture topologije zvijezde, sa standardnim i sa modularnim komutatorom. Iz dobijenih rezultata može se zaključiti da, u oba slučaja, linijska i topologija zvijezde sa modularnim komutatorom postižu približno iste rezultate. Kod *Hilscher* implementacije, kašnjenja uređaja, za slučaj linijske topologije, postaju dominantna kada je broj uređaja veći od 6. Posljedica toga je veća vrijednost minimalnog vremena komunikacionog ciklusa u odnosu na topologiju sa modularnim komutatorom, koja blago raste s povećanjem broja uređaja. S druge strane, topologija zvijezde realizovana sa standardnim komutatorom ima značajno lošije karakteristike, što je i očekivano s obzirom da komutator unosi veoma veliko kašnjenje zbog pohranjivanja okvira.

Kao što je ranije objašnjeno, minimalno vrijeme komunikacionog ciklusa je određeno vri-



(a)

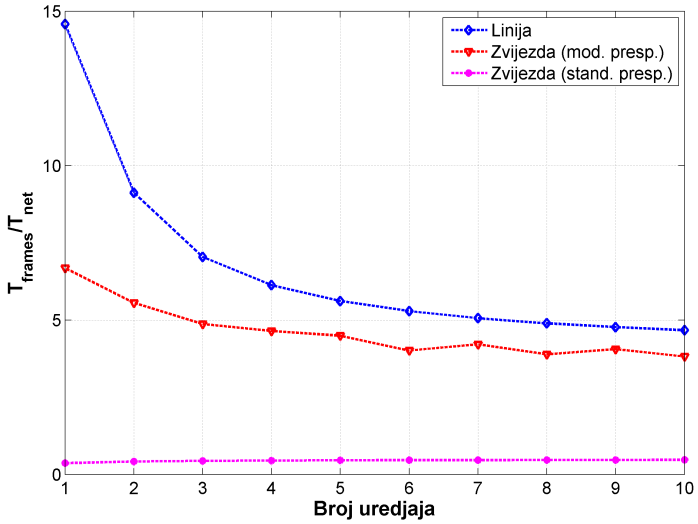


(b)

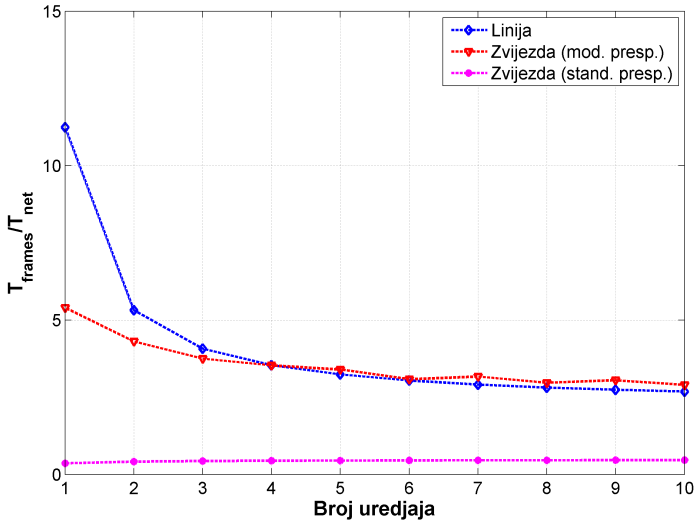
Slika 4.16: Odnos T_{frames}/T_{net} u funkciji broja uređaja za (a) *Beckhoff* i (b) *Hilscher* implementaciju (ukupna veličina procesnih podataka 8 bajtova)

jednošću dva člana: vremena potrebnog za slanje okvira i vremena propagacije signala kroz mrežu. Da bi se procijenio uticaj ovih članova na minimalno vrijeme komunikacionog ciklusa dobijeno u prethodnoj analizi, na slici 4.16 je prikazan odnos ova dva člana u funkciji broja uređaja.

Sa slike se vidi da je vrijeme slanja, kod obje implementacije, dominantno za manji broj uređaja, dok propagaciono kašnjenje preovladava pri većem broju uređaja. Zanimljivo je primjetiti da ovaj zaključak ne važi u slučaju topologije zvijezde sa standardnim komutatorom, kod koje je pomenuti odnos uvijek približno konstantan i manji od jedinice, što znači da je vrijeme propagacije uvijek veće od vremena slanja. Ovo se može objasniti činjenicom da član koji se odnosi na vrijeme slanja figuriše i u vremenu propagacije jer komutator pohranjuje okvir prije



(a)



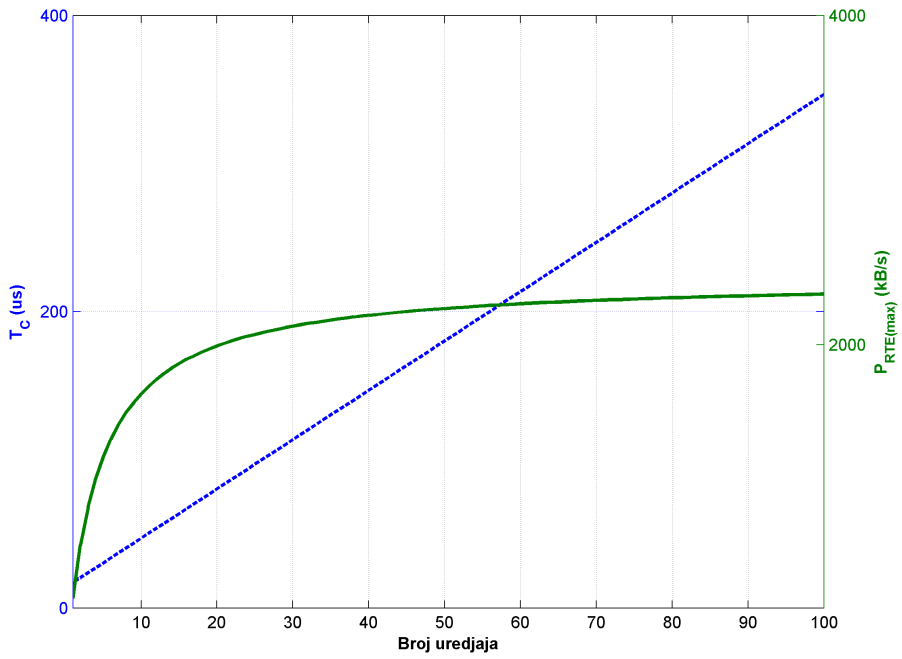
(b)

Slika 4.17: Odnos T_{frames}/T_{net} u funkciji broja uređaja za (a) *Beckhoff* i (b) *Hilscher* implementaciju (ukupna veličina procesnih podataka 64 bajta)

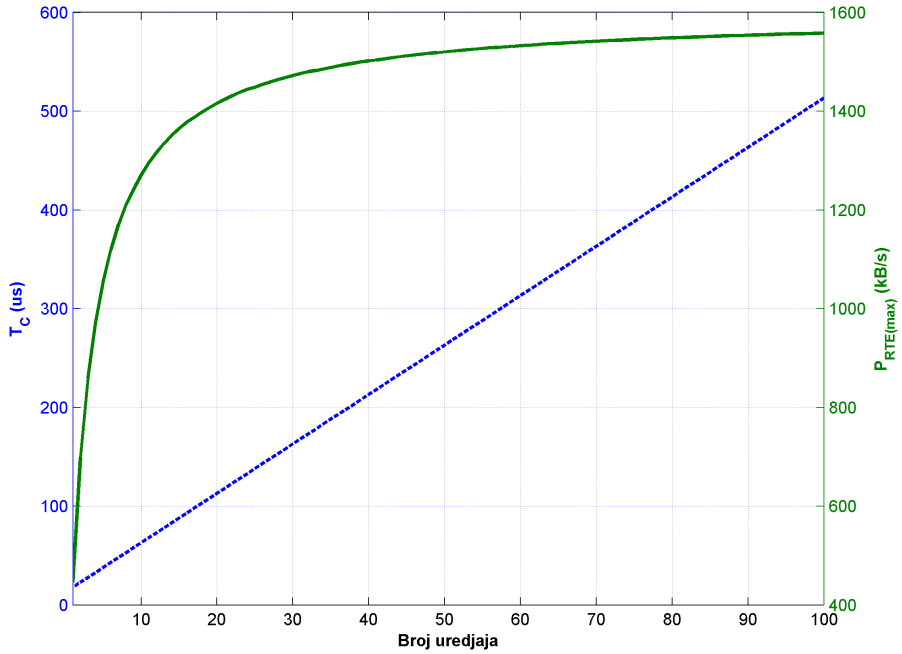
prosljeđivanja.

Kako bi se analizirao uticaj veličine procesnih podataka, prethodna analiza je ponovljena za slučaj kada je ukupna veličina podataka 64 bajta. Rezultati su dati na slici 4.17. Iz dobijenih rezultata se vidi da povećanje veličine procesnih podataka do određene mjere umanjuje uticaj vremena propagacije na minimalno vrijeme komunikacionog ciklusa sa povećanjem broja uređaja, što je i očekivano, jer povećanje veličine podataka direktno utiče na povećanje vremena slanja okvira.

Iz prethodnih analiza, može se zaključiti da je vrijeme slanja okvira kritičan faktor kada se koristi manji broj uređaja, dok vrijeme propagacije signala kroz mrežu postaje dominantno pri veoma velikom broju čvorova. To znači da se poboljšanje performansi može obezbijediti



(a)



(b)

Slika 4.18: T_C i $P_{RTE(max)}$ u funkciji broja uređaja za (a) *Beckhoff* i (b) *Hilscher* implementaciju (linijska topologija)

rješenjima koja omogućavaju smanjenje veličine okvira i/ili smanjenje propagacionog kašnjenja, naročito u slučaju velikih mreža koje koriste linijsku topologiju.

Vrijeme ciklusa industrijskog komunikacionog sistema podrazumijeva rezervaciju određenog dijela propusnog opsega za komunikacione servise protokola koji ne prenose vremenski kritične

podatke. Prema tome, u stvarnim sistemima nikada nije moguće postići vrijeme ciklusa koje odgovara minimalnom vremenu ciklusa. S druge strane, ovaj parametar utiče na druge značajne metrike poput vremena odziva, propusnog opsega rezervisanog za NRT saobraćaj i maksimalne propusne moći. Da bi se analizirao uticaj vremena ciklusa na navedene metrike, urađena je analiza kod koje je propusni opseg rezervisan za NRT saobraćaj fiksiran na 40% ukupne vrijednosti, pri čemu je vrijeme ciklusa određeno minimalnom vrijednošću komunikacionog ciklusa dobijenom u prethodnim analizama. Promjena vremena ciklusa i maksimalne propusne moći u zavisnosti od broja uređaja za linijsku topologiju i dvije implementacije uređaja prikazana je na slici 4.18. Analiza je ponovljena i za ostale dvije topologije, pri čemu su dobijeni kvalitativno slični rezultati koji su izostavljeni zbog uštede u prostoru.

Na slici 4.18 se vidi da promjena vremena ciklusa odgovara promjeni minimalnog vremena ciklusa, jer propusni opseg rezervisan za NRT saobraćaj uvijek ima fiksan procenat ukupnog opsega. S druge strane, maksimalna propusna moć, takođe, raste s povećanjem broja uređaja, što je očekivano, jer se povećava količina RT podataka. Međutim, povećanje maksimalne propusne moći odlazi u zasićenje i ima približno konstantnu vrijednost kada se dostigne određeni broj uređaja. Ovo se može objasniti činjenicom da je povećanje količine podataka, koje uzrokuje porast propusne moći, kompenzovano istovremenim povećanjem vremena ciklusa koje ima suprotan efekat na maksimalnu propusnu moć, što je vidljivo iz jednačine (3.2).

S obzirom da ECAT protokol omogućava sinhronizaciju na nivou aplikacija, vrijeme odziva dominantno zavisi od konfigurisanog vremena ciklusa komunikacionog sistema. Ukoliko je trajanje faze u kojoj se ramjenjuju NRT poruke fiksirano na prethodno opisan način, ukupno vrijeme ciklusa može se redukovati ako se na neki način minimizuje faza komunikacionog ciklusa u kojoj se razmjenjuju vremenski kritične poruke. U ECAT protokolu, trajanje ove faze odgovara minimalnom vremenu ciklusa. Prema tome, strategije za optimizaciju ove metrike direktno dovode do poboljšanja vremena odziva komunikacionog sistema.

4.4 Analitički model EPL protokola

Kao što je objašnjeno u odjeljku 2.2, EPL komunikacioni ciklus se sastoji iz tri faze, pa se vrijeme komunikacionog ciklusa, u opštem slučaju, može izraziti jednačinom

$$T_C = T_{ip} + T_{ap} + T_{id}, \tag{4.21}$$

gdje je: T_{ip} – dužina trajanja izohrone, T_{ap} – dužina trajanja asinhrone i T_{id} – dužina trajanja neaktivne faze.

Dužina trajanja asinhrone faze zavisi od dužine najveće poruke koja se razmjenjuje tokom ove faze. U opštem slučaju, ona može da se opiše sljedećom formulom

$$T_{ap} = T_{F(SoA)} + T_{F(ASnd)} + T_{RTT}, \tag{4.22}$$

gdje je: $T_{F(SoA)}$ – vrijeme potrebno za slanje SoA okvira, koje obično odgovara dužini minimalnog *Ethernet* okvira $T_{F(min)}$, $T_{F(ASnd)}$ – vrijeme potrebno za slanje asinhrone poruke ASnd okvirom i T_{RTT} – vrijeme povratnog puta, RTT (engl. *Round Trip Time*), čvora kojem je dozvoljeno slanje jedne asinhrone poruke. Vrijeme povratnog puta zavisi od toga da li asinhronu poruku šalje upravljački ili upravljani čvor. U slučaju da poruku šalje upravljački čvor, ovo vrijeme odgovara minimalnom rastojanju između *Ethernet* okvira, tj. $T_{RTT} = T_{ifg}$. Ako asinhronu poruku šalje upravljani čvor, onda vrijeme povratnog puta zavisi od više faktora, koji uključuju topologiju mreže, odnosno udaljenosti adresiranog od upravljačkog čvora, kao i vremena odziva čvora. U nastavku ovog poglavlja biće prikazana detaljna analiza vremena povratnog puta za različite topologije. S obzirom da je standardom propisano da minimalna ASnd poruka sadrži 300 bajtova korisnih podataka (uključujući EPL specifične kontrolne informacije), minimalno trajanje asinhrone faze (u slučaju bitske brzine 100 Mb/s) određeno je sa:

$$T_{ap(min)} = 31.84\mu s + T_{RTT(max)}, \tag{4.23}$$

gdje vrijednost $T_{RTT(max)}$ odgovara maksimalnom vremenu povratnog puta koje odgovara najudaljenijem čvoru u mreži, dok se vrijeme od 31.84 μs odnosi na slanje SoA okvira (5.76 μs) i minimalnog ASnd okvira (26.08 μs) definisanog standardom.

Dužina trajanja neaktivne faze varira od ciklusa do ciklusa i zavisi od toga da li se tokom asinhrone faze razmjenjuju poruke, preciznije, ona predstavlja preostali interval unutar asinhrone faze koji čvorovi ne mogu da iskoriste za slanje poruka.

Trajanje izohrone faze komunikacionog ciklusa EPL protokola zavisi od različitih faktora kao što su topologija mreže, veličina cikličnih podataka i vrijeme odziva čvorova. Uzimajući u obzir definiciju iz odjeljka 3.2.1, minimalno vrijeme komunikacionog ciklusa EPL protokola odgovara dužini trajanja ove faze, tj. $T_{C(min)} = T_{ip}$. Kako je pomenuta metrika vrlo značajna za procjenu karakteristika industrijskih komunikacionih sistema, u nastavku ovog poglavlja posebna pažnja će biti posvećena analizi trajanja izohrone faze za različite načine rada.

4.4.1 Standardni način rada

Na slici 4.19 prikazan je prostorno-vremenski dijagram izohrone faze EPL mreže koja se sastoji iz jednog upravljačkog i dva upravljana čvora povezana linijskom topologijom. U opštem slučaju, vrijeme trajanja izohrone faze može se odrediti jednačinom

$$T_{ip} = T_{sync} + T_{poll}, \tag{4.24}$$

gdje T_{sync} i T_{poll} predstavljaju dio izohrone faze u kojem se vrši sinhronizacija, odnosno prozivanje čvorova. Ovo vrijeme može dalje da se izrazi kao

$$T_{sync} = T_{F(SoC)} + T_W, \tag{4.25}$$

$$T_{poll} = T_{frames} + T_{net}, \tag{4.26}$$

gdje je: $T_{F(SoC)}$ – vrijeme potrebno za slanje SoC okvira, T_W – vrijeme čekanja potrebno da svi upravljani čvorovi prime i obrade SoC okvir, T_{frames} – vrijeme potrebno za slanje svih okvira u dijelu izohrone faze u kojem se prozivaju upravljani čvorovi, T_{net} – ukupno kašnjenje koje unose mrežne komponente.

U standardnom načinu rada, unutar svakog vremenskog slota, razmjenjuju se dva okvira (PReq i PRes) između upravljačkog i upravljanog čvora. Prema tome, član T_{frames} u jednačini (4.26) za mrežu koja ima n upravljanih čvorova može da se izrazi kao

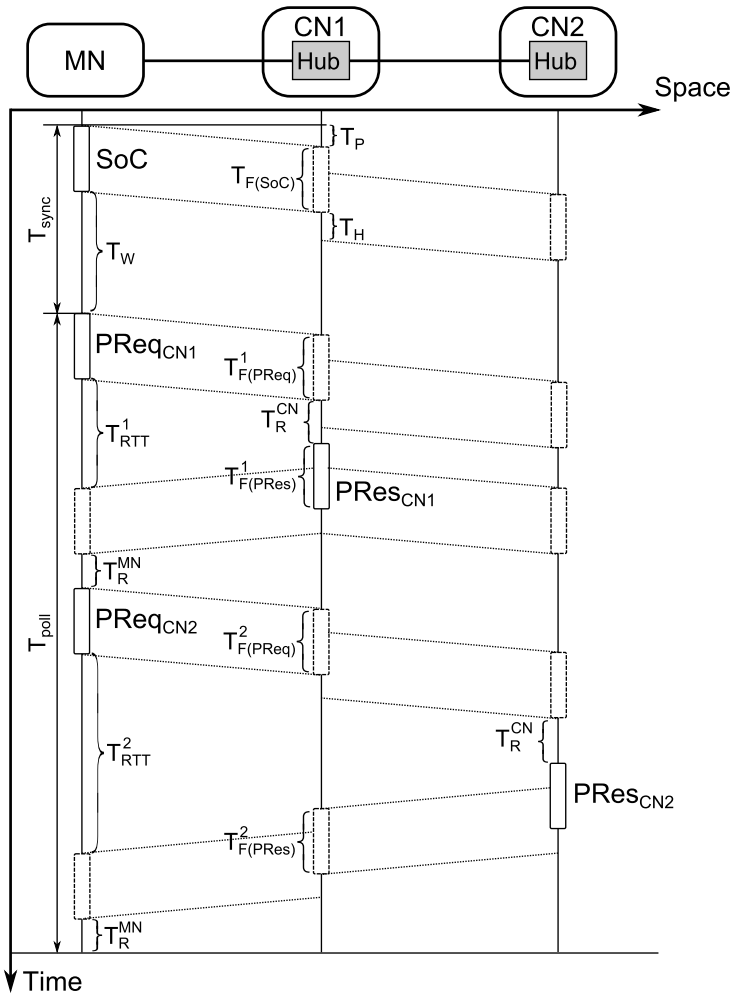
$$T_{frames} = \sum_{i=1}^{n} (T^i_{F(PReq)} + T^i_{F(PRes)}) = 2nT_F \tag{4.27}$$

uz pretpostavku da svi okviri imaju isto vrijeme slanja jednako T_F. Ovo je razumna pretpostavka, jer su procesni podaci koji se šalju PReq i PRes okvirima obično veličine samo nekoliko bajtova, što znači da mogu da se smjeste u minimalan *Ethernet* okvir.

Ukupno kašnjenje koje unose mrežne komponente (T_{net}) jednako je zbiru vremena povratnog puta svih upravljanih čvorova. Vrijeme povratnog puta za čvorove CN1 i CN2 sa slike 4.19 je definisano sa T^1_{RTT} i T^2_{RTT}, redom. Ovoj sumi je još potrebno dodati član nT^{MN}_R, što konačno daje opšti izraz

$$T_{net} = \sum_{i=1}^{n} T^i_{RTT} + nT^{MN}_R, \tag{4.28}$$

gdje T^i_{RTT} predstavlja vrijeme povratnog puta i-tog uređaja u mreži, a T^{MN}_R vrijeme odziva upravljanog čvora. Značajno je napomenuti da ovaj parametar isključivo zavisi od topologije



Slika 4.19: Prostorno-vremenski dijagram izohrone faze EPL mreže u standardnom režimu

mreže i vrijednosti vremenskih parametara korišćenih mrežnih komponenata.

Za primjer mreže sa slike 4.19, formule koje određuju vrijeme povratnog puta mogu se lako izvesti direktno sa slike, tj. $T^1_{RTT} = 2T_P + T_H + T^{CN}_R$ i $T^2_{RTT} = 4T_P + 3T_H + T^{CN}_R$, gdje T_P predstavlja vrijeme propagacije signala kroz kabl (tipično oko 5 ns/m), T_H kašnjenje koncentratora (tipično oko 500 ns), a T^{CN}_R vrijeme odziva upravljanog čvora. Za linijsku topologiju, prethodno razmatranje može se uopštiti za i-ti čvor korišćenjem sljedeće relacije:

$$T^i_{RTT} = 2iT_P + (2i-1)T_H + T^{CN}_R. \tag{4.29}$$

Važno je napomenuti da je, zbog jednostavnosti, pretpostavljeno da su svi čvorovi identični i povezani kablovima jednake dužine, tj. da su parametri T_P, T_H i T^{CN}_R jednaki. Ukoliko se (4.29) uvrsti u jednačinu (4.28) i primjeni identitet $\sum_{i=1}^{n} i = n(n+1)/2$, dobija se opšta formula za T_{net},

koja važi za slučaj linijske topologije:

$$T_{net}^{line} = \sum_{i=1}^{n}[2iT_P + (2i-1)T_H + T_R^{CN}] + nT_R^{MN}$$
$$= n(n+1)T_P + n^2T_H + n(T_R^{CN} + T_R^{MN}). \quad (4.30)$$

U slučaju topologije zvijezde, vrijeme povratnog puta upravljanog čvora se može odrediti korišćenjem jednačine

$$T_{RTT}^{i} = 4T_P + 2T_H + T_R^{CN}. \quad (4.31)$$

Ovaj izraz je izveden na osnovu činjenice da dva okvira (PReq i PRes) u topologiji zvijezde prelaze put koji čine dva kabla (od upravljačkog čvora do koncentratora i od koncentratora do upravljanog čvora i obrnuto) i jedan koncentrator. Ukoliko se posljednja jednačina uvrsti u (4.28), dobija se opšta formula za T_{net} koja važi u slučaju topologije zvijezde

$$T_{net}^{star} = 4nT_P + 2nT_H + n(T_R^{CN} + T_R^{MN}). \quad (4.32)$$

Vrijeme trajanja vremenskog slota i-tog čvora u EPL mreži, koja radi u standardnom načinu, odgovara njegovoj *PResTimeout* vrijednosti (T_{TO}^{i}), koja se može izraziti jednačinom

$$T_{TO}^{i} = T_{F(PReq)}^{i} + T_{F(PRes)}^{i} + T_{RTT}^{i} + T_R^{MN} + T_J, \quad (4.33)$$

gdje T_J određuje ukupan džiter, koji unose mrežne komponente. S obzirom da T_{TO}^{i} određuje vrijeme trajanja vremenskog slota i-tog čvora, ova vrijednost, takođe, se može koristiti za izračunavanje vremena trajanja dijela izohrone faze u kojem se prozivaju upravljani čvorovi, tj. $T_{poll} = \sum_{i=1}^{n} T_{TO}^{i}$. Ukoliko se zanemari komponenta džitera u (4.33), prethodna jednačina daje isti rezultat koji je dobijen u prethodnoj analizi.

4.4.2 PRC način rada

Na slici 4.20 prikazan je prostorno-vremenski dijagram izohrone faze EPL mreže u PRC režimu koju čine jedan upravljački i dva upravljana čvora povezani linijskom topologijom. Analizom ovog dijagrama zaključuje se da trajanje PRC slota za EPL mrežu sa m ulančanih čvorova može da se odredi korišćenjem izraza

$$T_{prc} = T_{F(PResMN)} + T_{F(PRes)}^{m} + T_{PRes}^{m} + T_{RTT}^{m}, \quad (4.34)$$

gdje $T_{F(PResMN)}$ predstavlja vrijeme potrebno za slanje PResMN okvira, dok su $T^m_{F(PRes)}$, T^m_{PRes} i T^m_{RTT}, vrijeme slanja PRes okvira, vrijeme odziva i vrijeme povratnog puta posljednjeg čvora u PRC slotu, redom. Vrijeme povratnog puta (T^m_{RTT}) može se izračunati korišćenjem odgovarajućih jednačina, koje su izvedene za standardni način rada, uz uslov da je $T^{CN}_R = T_{ifg}$. Osim toga, važno je napomenuti da za PResMN ne važi pretpostavka o veličini okvira, koja je korišćena u standardnom načinu rada, jer se u ovaj okvir smještaju izlazni podaci svih ulančanih čvorova. To znači da je veličina PResMN okvira značajno veća od bilo kojeg PReq/PRes okvira koji se razmjenjuje u standardnom načinu rada. Vrijeme potrebno za slanje PResMN okvira može se odrediti korišćenjem sljedeće formule

$$T_{F(PResMN)} = \frac{L_{F(PResMN)} \cdot 8}{B_{tot}}, \tag{4.35}$$

gdje $L_{F(PResMN)}$ predstavlja dužinu PResMN okvira izraženu brojem bajtova koja se može odrediti pomoću sljedeće formule

$$L_{F(PResMN)} = \max\left[L_{ovh} + L_{data}, L_{F(min)}\right], \tag{4.36}$$

gdje $L_{ovh} = 36$ sadrži ukupnu dužinu kontrolnih informacija EPL protokola (uključujući *Ethernet* kontrolne informacije), a L_{data} ukupnu veličinu izlaznih procesnih podataka koji se prenose PResMN okvirom.

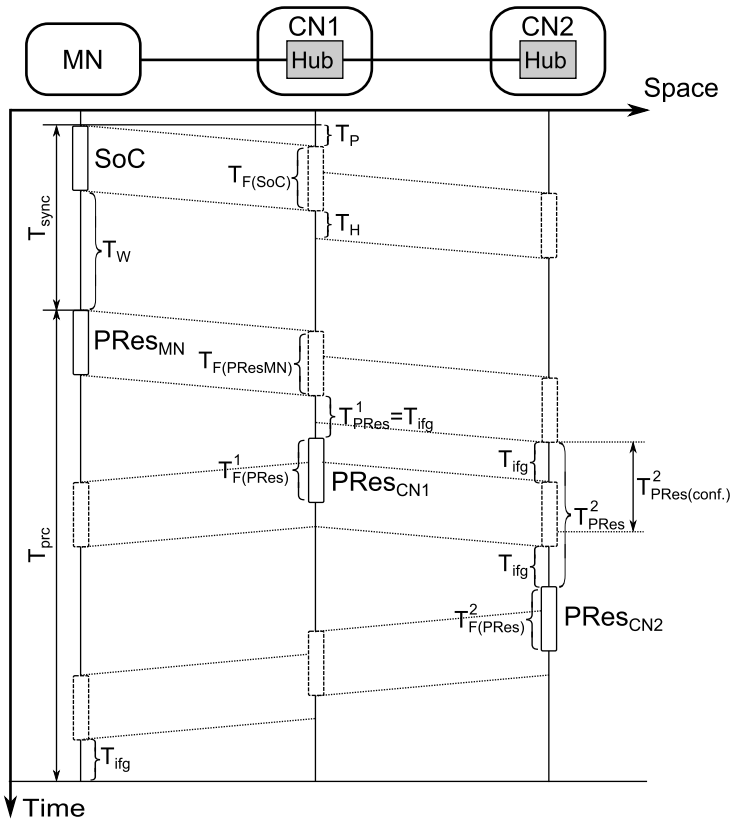
Vrijeme odziva i-tog čvora u PRC slotu može se odrediti pomoću sljedeće jednačine

$$T^i_{PRes} = T^{i-1}_{PRes} + T^{i,i-1}_D + T^{i-1}_{F(PRes)}, \qquad i = 2, \ldots, m, \tag{4.37}$$

gdje: $T^1_{PRes} = T_{ifg}$, $T^{i,i-1}_D$ – propagaciono kašnjenje (*SyncDelay*) između i-tog i $(i-1)$-og čvora, a $T^{i-1}_{F(PRes)}$ – vrijeme potrebno za slanje PRes okvira $(i-1)$-og čvora. Međutim, da bi se kompenzovao uticaj džitera mrežnih komponenata i nepreciznosti mjerenja, upravljački uređaj svakom čvoru dostavlja konfigurisanu vrijednost ovog vremena, $T^i_{PRes(conf.)}$, koja sadrži dodatni član δ^i_C za korekciju vremena odziva, tj.

$$T^i_{PRes(conf.)} = T^i_{PRes} + \delta^i_C. \tag{4.38}$$

Korekcioni član se određuje tako da čvor šalje sopstveni PRes okvir tokom prijema PRes okvira od prethodnog čvora (kao što je ilustrovano na slici 4.20). CSMA (engl. *Carrier Sense Multiple Access*) mehanizam, koji je standardno podržan u svakom *Ethernet* mrežnom interfejsu,



Slika 4.20: Prostorno-vremenski dijagram izohrone faze EPL mreže u PRC režimu

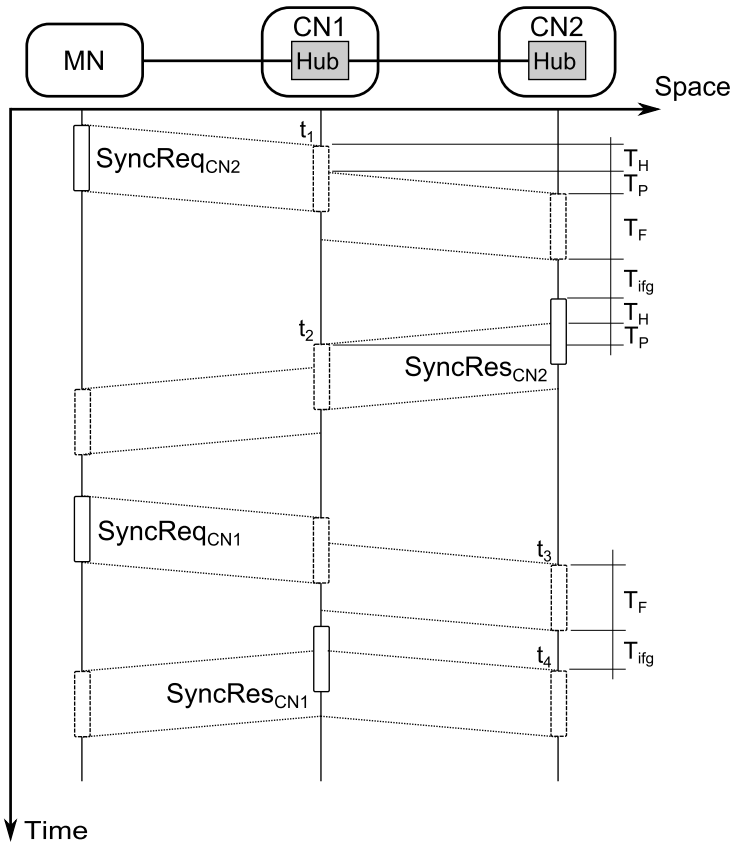
odgađa slanje PRes okvira dok se ne završi prijem trenutnog okvira i, na taj način, osigurava minimalno rastojanje između ulančanih PRes okvira. Značajno je napomenuti da se prvi čvor u PRC slotu uvijek konfiguriše tako da bude $T_{PRes(conf.)} = 0$, tj. ovaj čvor šalje sopstveni PRes okvir odmah po prijemu PResMN okvira (sa obaveznim IFG rastojanjem između okvira).

Kao što je opisano u odjeljku 2.2.4, mjerenje propagacionih kašnjenja između čvorova, koja su neophodna za izračunavanje $T_{PRes(conf.)}$ parametara, obavlja se automatski tokom inicijalizacije mreže. Ovo podrazumijeva razmjenu specifično definisanih okvira, *SyncReq* i *SyncRes*, između čvorova tokom asinhrone faze. Prostorno-vremenski dijagram sa slike 4.21 ilustruje opisani koncept na primjeru dva upravljana čvora povezana linijskom topologijom.

Propagaciona kašnjenja između čvorova CN1 and CN2 sa slike 4.21 mogu se izračunati pomoću sljedećih jednačina:

$$T_D^{1,2} = t_2 - t_1 - T_F = 2(T_H + T_P) + T_{ifg}, \tag{4.39}$$

$$T_D^{2,1} = t_4 - t_3 - T_F = T_{ifg}. \tag{4.40}$$



Slika 4.21: Ilustracija mjerenja propagacionog kašnjenja (*SyncDelay*)

U slučaju linijske topologije, jednačine (4.39) i (4.40) mogu se generalizovati kao:

$$T_D^{i,j} = \begin{cases} 2(j-i)(T_H + T_P) + T_{ifg}, & i < j \\ T_{ifg}, & i > j \end{cases} \tag{4.41}$$

uz pretpostavku da svi upravljani uređaji imaju iste vremenske parametre (T_H i T_P). U slučaju topologije zvijezde, propagaciono kašnjenje između bilo koja dva čvora je određeno sa $2T_P + T_H + T_{ifg}$.

Značajno je pomenuti da redoslijed slanja PRes okvira utiče na efikasnost komunikacije u smislu ukupnog trajanja PRC slota. U slučaju linijske topologije, optimalan raspored slanja odgovara fizičkom redoslijedu ulančanih čvorova. Ovo se može zaključiti na osnovu jednačine (4.41) koja pokazuje da susjedni čvorovi uvijek imaju najmanje moguće propagaciono kašnjenje. što rezultuje manjoj vrijednosti T_{PRes}, a samim tim i kraćem trajanju PRC slota (T_{prc}).

Rješavanjem rekurzije (4.37) po T_{PRes}^m, uz pretpostavku da se koristi optimalan raspored slanja PRes okvira i uz iste uslove kao u slučaju standardnog načina rada (vrijeme slanja svih PRes okvira je jednako T_F i mrežne komponente imaju identične parametre), dobijaju se formule:

$$T_{PRes}^m = (m-1)T_F + mT_{ifg}, \tag{4.42}$$

$$T^{m}_{PRes} = 2(m-1)T_P + (m-1)(T_F + T_H) + mT_{ifg}, \tag{4.43}$$

koje važe za linijsku i topologiju zvijezde, redom.

Vrijeme trajanja PRC slota, takođe, se može izraziti na ekvivalentan način kao T_{poll} član u (4.26), tj. $T_{prc} = T_{frames} + T_{net}$, pri čemu se vremena

$$T_{frames} = (m+1)T_F, \tag{4.44}$$

$$T^{line}_{net} = 2mT_P + (2m-1)T_H + (m+1)T_{ifg}, \tag{4.45}$$

$$T^{star}_{net} = 2(m+1)T_P + (m+1)(T_H + T_{ifg}) \tag{4.46}$$

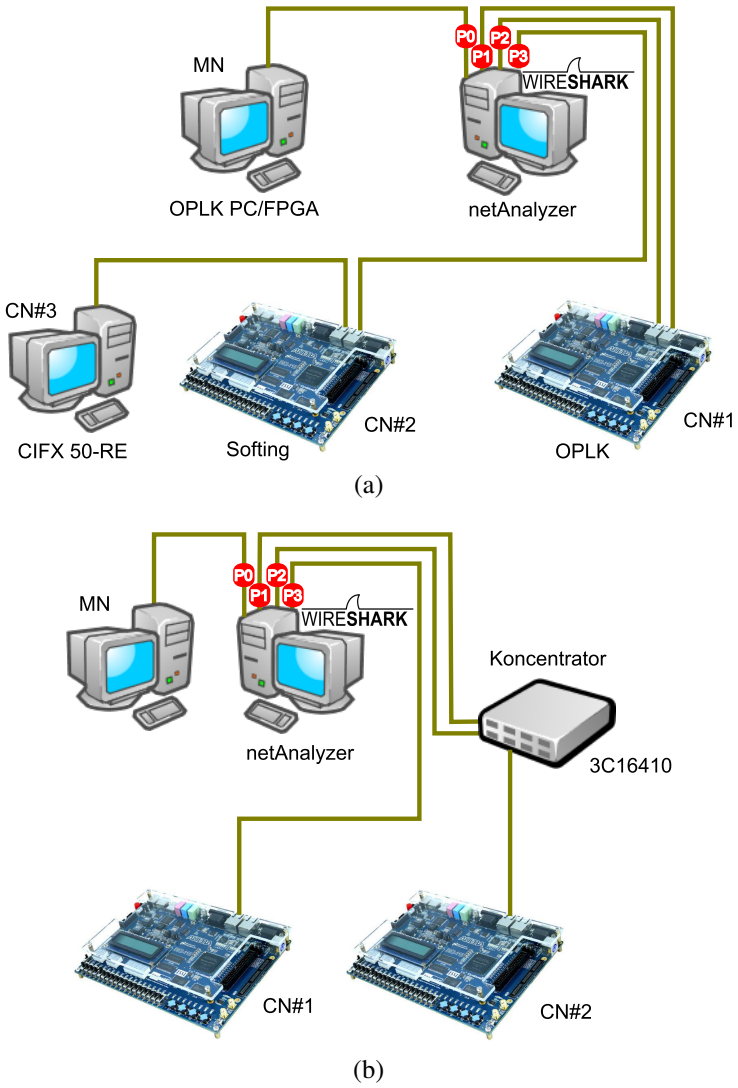
mogu odrediti kada se T^{m}_{PRes} i T^{m}_{RTT} u (4.34) zamijene prethodno izvedenim izrazima uz pretpostavku da vrijedi $T_{F(PResMN)} = T^{m}_{FPRes} = T_F$.

4.5 Karakterizacija vremenskih parametara EPL analitičkog modela

U cilju karakterizacije vremenskih parametara prethodno izvedenog analitičkog modela, realizovana je eksperimentalna postavka prototipa EPL mreže za dvije mrežne topologije, kao što je prikazano na slici 4.22. Kao i u slučaju ECAT sistema, za mjerenje različitih vremenskih parametara EPL mrežnih komponenata korišćen je *netAnalyzer*. Prva postavka (slika 4.22a) se koristi za karakterizaciju vremenskih parametara MN čvora i CN čvorova različitih proizvođača, dok se druga postavka (slika 4.22b) koristi za karakterizaciju vremenskih parametara koncentratora koji omogućava formiranje topologije zvijezde.

U okviru eksperimentalne analize, korišćene su dvije implementacije MN čvora: softverska implementacija zasnovana na klasičnom PC računaru i implementacija bazirana na specifičnom hardveru realizovanom na FPGA (engl. *Field Programmable Gate Array*) platformi. Obje implementacije su zasnovane na *openPOWERLINK v2.2.1* softverskom paketu otvorenog koda koji sadrži sve neophodne komponente za realizaciju EPL komunikacionog steka. Konfiguraciju MN čvora zasnovanog na klasičnom PC računaru čine Linux Mint 17.1, Kernel 3.12.24-rt38 x86_64, 7.5 GB RAM, Intel Core i5-4300M CPU @2.60 GHz, dok je FPGA varijanta zasnovana na razvojnom sistemu Altera DE2-115.

Kao infrastrukturna komponenta korišćen je 16-portni *3Com* koncentrator (3C16410), a analizirana su i tri upravljana čvora različitih proizvođača: *openPOWERLINK* (OPLK) i *Softing*



Slika 4.22: Eksperimentalna postavka EPL mreže: (a) linijska i (b) topologija zvijezde

implementacija na Altera DE2-115 platformi, te *Hilscher* implementacija na PCI kartici CIFX 50-RE, koja se ugrađuje u standardan PC računar. Sve tri implementacije upravljanog čvora imaju integrisan koncentrator sa tri porta, čime je omogućeno povezivanje uređaja u linijsku topologiju.

Za povezivanje uređaja u mreži, korišćeni su kablovi dužine 2 m, što znači da se kašnjenja uslijed propagacije signala kroz kablove mogu zanemariti prilikom analize. Svi upravljani čvorovi su konfigurisani tako da razmjenjuju 4 bajta ulaznih i 4 bajta izlaznih procesnih podatka. Trajanje EPL ciklusa je podešeno tako da bude jednako 1 ms (10 ms u slučaju PC implementacije MN čvora zbog ograničenja po pitanju džitera), a dužina svakog eksperimenta je definisana tako da se obezbijedi najmanje 30000 mjerenja za svaki od parametara.

Vremenske parametre koji se odnose na MN čvor čine period SoC okvira ($T_{SoC-SoC}$) i *PResTimeout* vrijednosti (T_{TO}). Varijacija perioda SoC okvira predstavlja važan parametar, koji opisuje sposobnost MN čvora da precizno definiše početak komunikacionog ciklusa i može da

se koristi kao mjera preciznosti sinhronizacije čvorova.

PResTimeout parametar određuje trajanje vremenskog slota datog CN čvora i, prema tome, uključuje kašnjenja svih mrežnih komponenata prilikom prozivanja odgovarajućeg čvora kao što je definisano jednačinom (4.33). Iako se ovaj parametar podešava prilikom konfiguracije mreže, a njegova vrijednost zavisi od vremenskih parametara mrežnih komponenata, mogućnost MN čvora da precizno generiše vremenske intervale definisane *PResTimeout* vrijednostima ima ključan uticaj na trajanje izohrone faze. U suštini, s obzirom na način rada MN čvora, prilikom slanja PReq okvira u standardnom režimu, *PResTimeout* parametar određuje maksimalno vrijeme odziva ovog čvora (T_R^{MN}). Vremenski parametri korišćeni pri izvođenju EPL analitičkog modela podrazumijevaju različita kašnjenja mrežnih komponenti, kao što su kašnjenje koncentratora (T_H) i vrijeme odziva upravljanog čvora (T_R^{CN}).

Period SoC okvira dobija se kada se oduzme vrijeme prijema dva uzastopna SoC okvira na portu P0 mjerne sonde. Rezultat ovog mjerenja, za slučaj PC realizacije MN čvora, prikazan je na slici 4.23a. Minimalna vrijednost perioda koja se može podesiti za ovu implementaciju uz istovremeno očuvanje stabilnosti rada mreže iznosi 10 ms. Detaljnija kvantifikacija ovog perioda je data u tabeli 4.5. Iz tabele se vidi da standardna devijacija (σ) SoC perioda iznosi 3.187 μs (apsolutno odstupanje manje od 60 μs) sa srednjom vrijedošću (μ), koja je veoma bliska konfigurisanoj vrijednosti komunikacionog ciklusa (10.000389 ms). Mjerenje ovog perioda je ponovljeno za slučaj kada se koristi FPGA implementacija MN čvora, pri čemu je EPL ciklus podešen na 1 ms. S obzirom da FPGA implementacija omogućava precizniju kontrolu vremenskih intervala jer se koriste specifične hardverske komponenta, moguće je postići znatno manje vrijednosti ciklusa u sistemu. Rezultati dobijeni za ovaj slučaj, dati su na slici 4.23b i u tabeli 4.5. Očigledno je da, iz prethodno pomenutih razloga, FPGA implementacija MN čvora postiže značajno bolje rezultate po pitanju performansi u odnosu na PC implementaciju.

Kao što je ranije pomenuto, kod *openPOWERLINK* implementacije funkcionalnosti MN čvora, slanje PReq okvira u standardnom načinu rada se inicira korišćenjem sistemskog tajmera koji ima visoku preciznost kako bi se izbjegla kašnjenja potrebna za obradu prekida, pri čemu interval aktivacije tajmera odgovara T_{TO} intervalu odgovarajućeg CN čvora. Vrijednost ovog intervala se za svaki čvor u mreži može inicijalno procijeniti korišćenjem jednačine (4.33). Međutim, da bi se uračunao uticaj komponente džitera (član T_J u jednačini), ovu vrijednost je potrebno dodatno podesiti eksperimentalnim putem.

Nakon nekoliko pokušaja, zaključeno je da za mrežu koja se sastoji iz dva OPLK upravljana čvora, koji su povezani sa PC baziranim MN čvorom linijskom topologijom, vrijednost od 25 μs za oba čvora (CN#1 i CN#2) osigurava stabilan rad sistema bez grešaka. Vrijednost

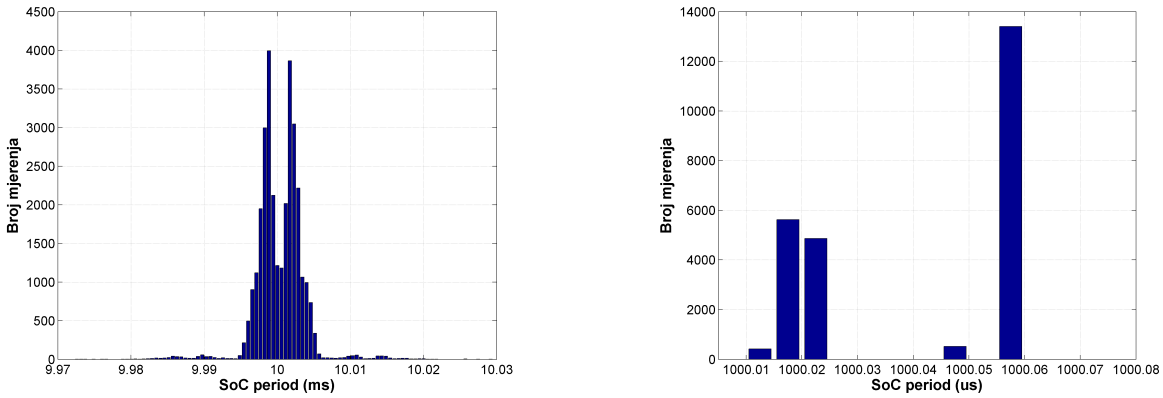
Tabela 4.5: Pregled statistike mjerenih vremenskih parametara

Parametar	Min.	Max.	μ	σ
$T^{PC}_{SoC-SoC}$	9972.31	10029.44	10000.389	3.187
$T^{FPGA}_{SoC-SoC}$	1000.01	1000.06	1000.42	0.02
$T^{CN\#1}_{TO}$	47.92	55.61	55.376	0.248
$T^{CN\#2}_{TO}$	29.2	50.4	29.47	0.454
T^{FPGA}_{TO}	17.72	17.81	17.762	0.006
T^{OPLK}_{H}	2.101	2.171	2.137	0.014
$T^{Softing}_{H}$	2.015	2.085	2.046	0.014
$T^{Hilscher}_{H}$	0.549	0.599	0.572	0.012
T^{3Com}_{H}	0.413	0.463	0.434	0.012
$T^{CN}_{R(OPLK)}$	0.945	0.995	0.977	0.02
$T^{CN}_{R(Softing)}$	0.925	0.975	0.968	0.009
$T^{CN}_{R(Hilscher)}$	1.719	1.769	1.746	0.02

Sve vrijednosti su izražene u μs

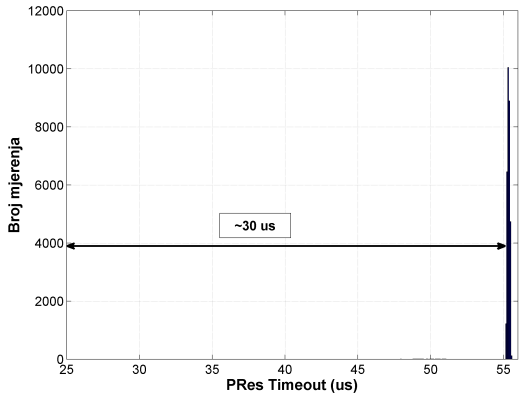
T_{TO} intervala za pomenuta dva čvora, može se odrediti tako što se oduzme vrijeme prijema PReq$_{\text{CN\#1}}$ i PReq$_{\text{CN\#2}}$ okvira (za $T^{CN\#1}_{TO}$), odnosno PReq$_{\text{CN\#2}}$ i SoA okvira (za $T^{CN\#2}_{TO}$) na portu P0. Kvalitativni rezultati mjerenja su prikazani na slici 4.24, dok je kvantitativna statistička analiza data u tabeli 4.5.

Sa slike 4.24 se vidi da je većina mjerenja, kod oba intervala, koncentrisana oko srednjih vrijednosti (sa neznatnim brojem izvan datog opsega) koje su date u prethodno pomenutoj tabeli. Ovdje treba skrenuti pažnju na to da $T^{CN\#1}_{TO}$ i $T^{CN\#2}_{TO}$ imaju konstantne pomjeraje od približno 30 μs i 5 μs, redom, u odnosu na konfigurisane vrijednosti. Oni nastaju kao posljedica baferovanja okvira unutar softvera koji kontroliše rad mrežnog interfejsa, jer se koristi softverska

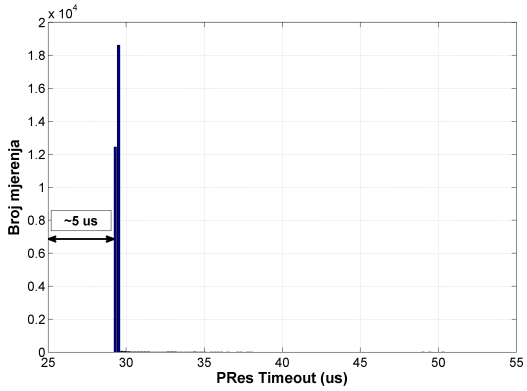


(a) (b)

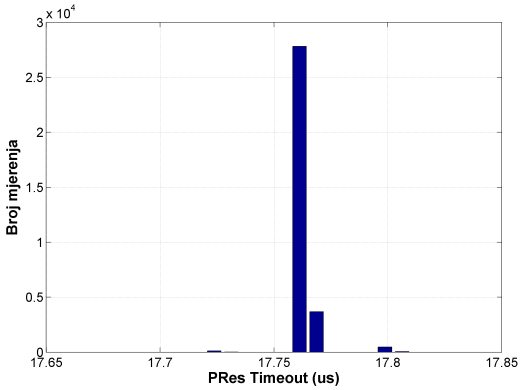
Slika 4.23: Izmjereni period SoC okvira za (a) PC i (b) FPGA implementaciju MN čvora



(a)



(b)



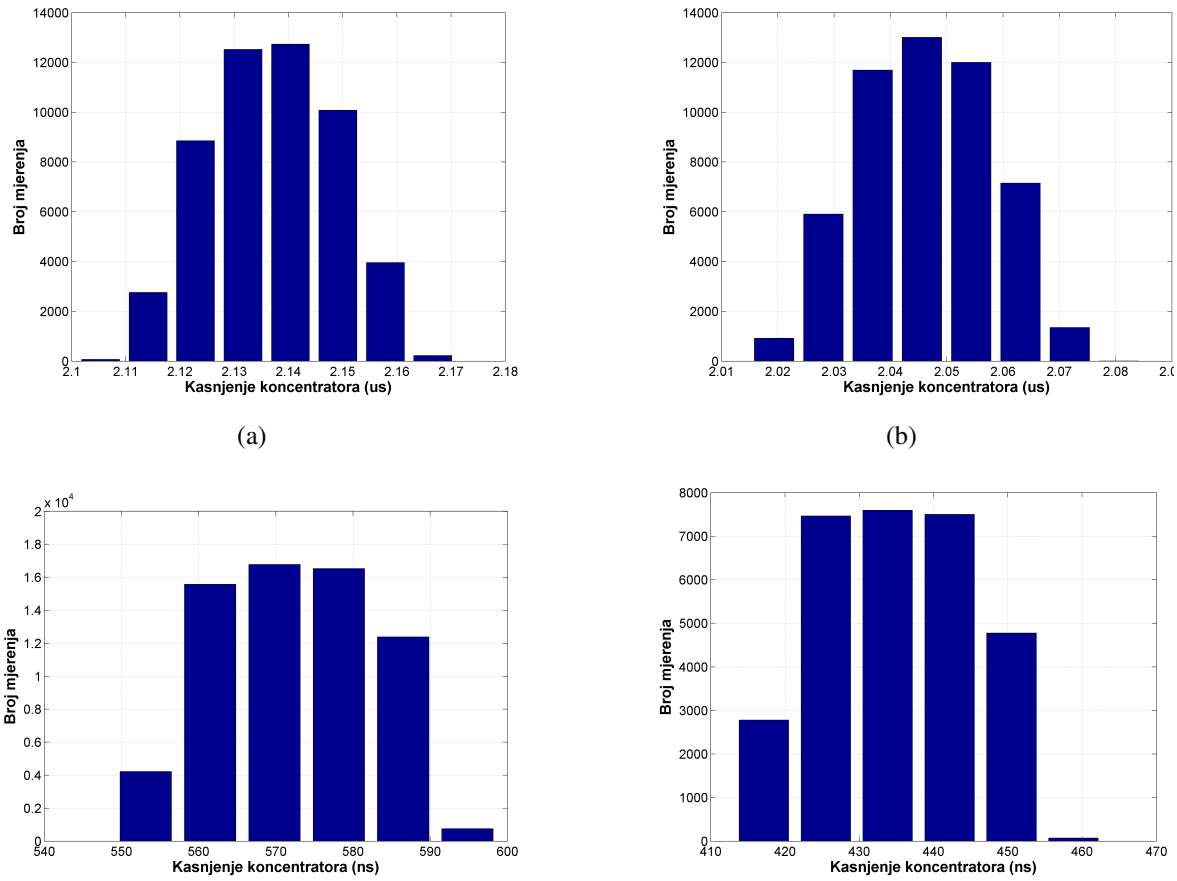
(c)

Slika 4.24: Izmjerene vrijednosti *PResTimeout* parametra (linijska topologija): (a) $T_{TO}^{CN\#1}$ i (b) $T_{TO}^{CN\#2}$ za PC, i (c) $T_{TO}^{CN\#1}$ za FPGA implementaciju MN čvora

implementacija MN čvora bazirana na PCAP biblioteci. Ovakva implementacija unosi značajno kašnjenje prilikom slanja okvira i na taj način pogoršava performanse. Performanse se mogu poboljšati optimizacijom softverskih komponenti koje su specijalno razvijene samo za određene mrežne interfejse. Osim toga, dodatno poboljšanje performansi može se postići korišćenjem specijalnih hardverskih komponenti koje su realizovane na FPGA platformi.

Da bi ocijenili mogućnosti rješenja sa specifičnim hardverom, prethodna mjerenja su ponovljena za slučaj implementacije MN čvora na FPGA platformi, pri čemu je eksperimentalno određena vrijednost T_{TO} intervala podešena na 17.7 μs. Rezultati mjerenja su prikazani na slici 4.24c, dok su statistički brojni rezultati dati u tabeli 4.5. Jasno je da FPGA implementacija MN čvora obezbjeđuje znatno precizniju kontrolu T_{TO} intervala, što omogućava da vrijeme odziva ovog čvora bude reda veličine 1 μs.

Kašnjenje koncentratora se može izraziti razlikom između vremena prijema istog okvira na portovima P0 i P2 (slika 4.22). Mjerena su kašnjenja za četiri različita tipa koncentratora, tri integrisana u CN čvoru različitih proizvođača (OPLK, *Softing* i *Hilscher*) i jedan 16-portni ek-



Slika 4.25: Izmjereno kašnjenje za (a) OPLK, (b) *Softing*, (c) *Hilscher* i (d) *3Com* koncentrator

sterni koncentrator (*3Com*). Kvalitatitvni rezultati i pregled statistike za navedene koncentratore prikazani su na slici 4.25 i u tabeli 4.5.

Iz prikazanih rezultata, može se zaključiti da srednja vrijednost kašnjenja koncentratora integrisanih u OPLK i *Softing* čvorove iznosi 2.137 μs i 2.046 μs, redom, sa apsolutnim odstupanjem od 70 ns. S druge strane, tipična vrijednost ovog kašnjenja je ispod 500 ns, kao što je slučaj sa kašnjenjem *Hilscher* i *3Com* koncentratora (slike 4.25d i 4.25c). Ovo se može objasniti analizom interne arhitekture OPLK hardvera. Naime, *openMAC* modul, koji obezbjeđuje funkcionalnost koncentratora, dizajniran je za RMII (*Reduced Media Independent Interface*) interfejs, dok DE2-115 razvojna ploča koristi *Marvell* 88E1111 PHY komponente, koje koriste MII (*Media Independent Interface*) interfejs. Da bi ovaj modul bio kompatibilan sa razvojnom pločom, OPLK posjeduje posebnu hardversku komponentu (*rmii2miiconv*) koja konvertuje signale između dva interfejsa. Ova komponenta unosi značajno kašnjenje prilikom prosljeđivanja okvira kroz koncentrator. Korišćenjem logičkog analizatora za Altera FPGA komponente (*SignalTap Logic II Embedded Logic Analyzer*) potvrđeno je da navedena komponenta unosi kašnjenje u opsegu 1.7 μs do 1.74 μs, što znači da bi kašnjenje koncentratora bilo oko 413 ns,

kada bi se koristile PHY komponente sa RMII interfejsom.

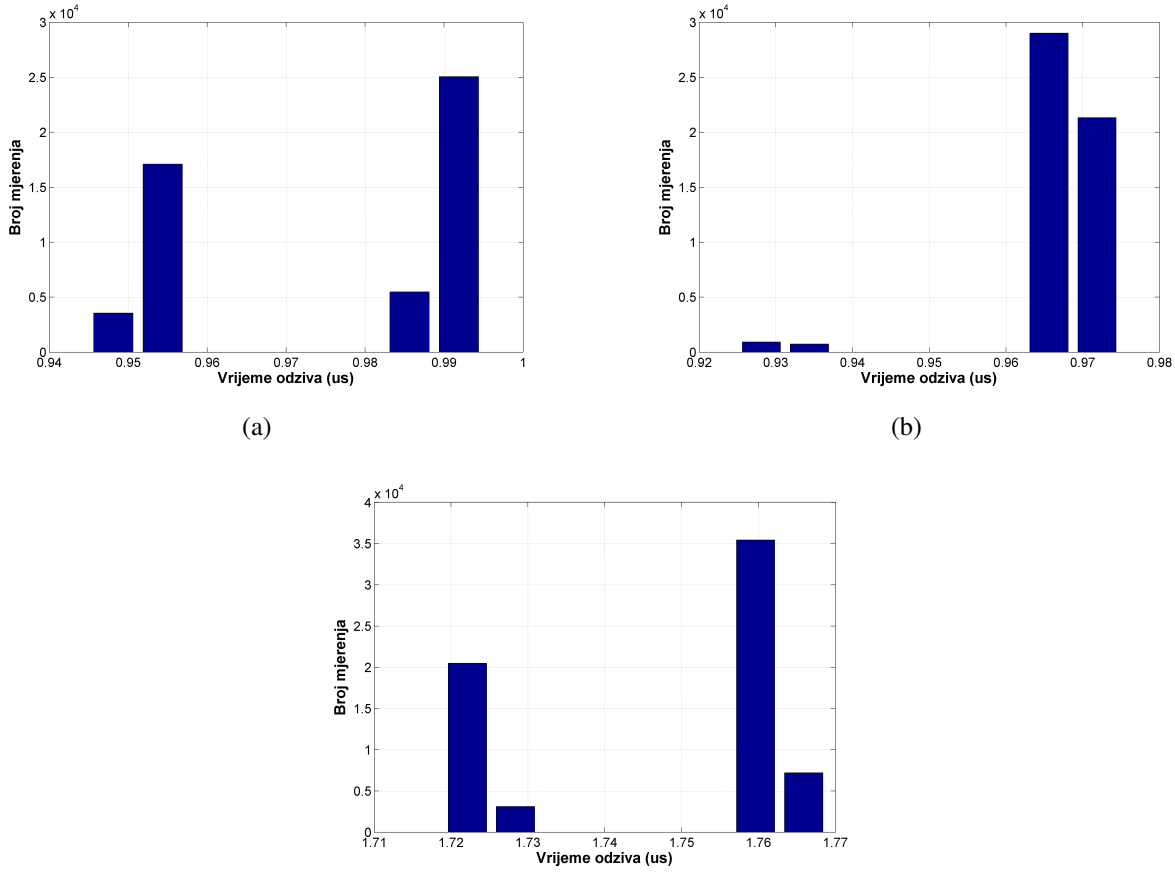
Sličan efekat se dobija i u slučaju *Softing* realizacije CN čvora, jer je ovaj dizajn baziran na OPLK implementaciji. Međutim, kao što može da se zaključi iz rezultata dobijenih mjerenjem, kašnjenje je nešto manje, što znači da je dizajn konvertora između interfejsa u ovom slučaju dodatno optimizovan. Konačno, važno je napomenuti da se eliminisanjem pomenutog konvertora devijacija kašnjenja takođe smanjuje na 50 ns, što odgovara vrijednostima druga dva koncentratora.

Vrijeme odziva upravljanog čvora se može izračunati ako se vrijeme prijema PReq okvira kojim se proziva odgovarajući čvor na portu P0 oduzme od vremena prijema PRes okvira istog čvora na portu P1 (slika 4.22). Ovo vrijeme je eksperimentalno kvantifikovano za tri pomenute realizacije upravljanog čvora. Rezultati mjerenja su prikazani na slici 4.26, dok je detaljniji prikaz statistike dat u tabeli 4.5.

Kao što je očekivano, vrijeme odziva za slučaj OPLK i Softing implementacije je veoma precizno sa veoma malim apsolutnim odstupanjem (50 ns), zbog toga što se koristi prethodno opisani *auto-response* mehanizam implementiran u *openMAC* modulu. S druge strane, *Hilscher* implementacija ima nešto veće kašnjenje u odzivu čvora, ali sa istim nivoom preciznosti. Iz prezentovanih rezultata može se zaključiti da komercijalno dostupne realizacije CN čvorova mogu da obezbijede veoma precizno vrijeme odziva čije je kašnjenje ispod 2 μs.

Osim karakterizacije vremenskih parametara EPL analitičkog modela, sprovedena je analiza procesa mjerenja *SyncDelay* parametara, koji je opisan u odjeljku 4.4. U cilju određivanja vrijednosti $T_{PRes(conf.)}$ i T_D parametara za dva upravljana čvora (CN#1 i CN#2) povezana linijskom topologijom, prvo su iz ukupnog saobraćaja izolovane *SyncReq* i *SyncRes* okviri, a zatim je analiziran njihov sadržaj. Nakon analize, zaključeno je da propagaciona kašnjenja (*SyncDelays*) između čvorova, dobijena pomenutim procesom, imaju vrijednosti $T_D^{1,2} = 5.24\mu s$ i $T_D^{2,1} = 1\mu s$, što je u skladu sa jednačinama (4.39) i (4.40). Takođe, konfigurisano vrijeme odziva za čvorove CN#1 i CN#2 iznosi $T^1_{PRes(conf.)} = 0$ i $T^2_{PRes(conf.)} = 6.31\mu s$, redom.

Korišćenjem (4.38) i prethodno datih vrijednosti, može se zaključiti da korekcioni član za čvor CN#2 ima vrijednost $\delta_C = -0.45\mu s$. Analiziranjem izvornog koda koji je dostupan u okviru OPLK softverskog paketa, zaključeno je da se korekcioni član za kompenzaciju greške mjerenja i ukupnog džitera u mreži za i-ti upravljani čvor ($i > 1$) može izraziti u obliku $\delta_C^i = (i-1)\cdot 50 - 500$ [ns]. Važno je napomenuti da ova formula vrijedi samo u slučaju linijske topologije.



Slika 4.26: Izmjereno vrijeme odziva za (a) OPLK, (b) *Softing* i (c) *Hilscher* CN

4.6 Teorijska analiza EPL protokola

U slučaju EPL protokola, analitički model pokriva analizu izohrone i asinhrone faze komunikacionog ciklusa. Dužina trajanja asinhrone faze zavisi od maksimalne dužine asinhrone poruke koja se razmjenjuje u sistemu. S obzirom da EPL protokol podržava fragmentaciju dužih poruka, ova faza može da se podesi na vrijednost koja odgovara potrebama korisnika. Kao što je pomenuto, standard [85] propisuje da dužina asinhrone poruke ne može da bude manja od 300 bajtova uključujući EPL kontrolne informacije, što znači da, kada se uzme u obzir slanje SoA poruke, ova faza ne može da bude kraća od 31.84 μs kao što je dato formulom (4.23).

Izohrona faza EPL protokola odgovara minimalnom trajanju komunikacionog ciklusa i direktno zavisi od broja uređaja, veličine procesnih podataka i topologije koja se koristi za realizaciju mreže. Rad sa ulančavanjem PRes okvir omogućava smanjenje trajanja ove faze, jer se izbjegava prozivanje čvorova koje inherentno zauzima značajan procenat propusnog opsega. Da bi se procijenile mogućnosti EPL protokola, sprovedena je analiza korišćenjem izvedenog analitičkog modela za dva načina rada (standardni i PRC) i dvije topologije (linijsku i topologiju

zvijezde). Prilikom analize, uvedene su sljedeće pretpostavke:

- kašnjenje koncentratora integrisanog u čvor, odnosno eksternog koncentratora za realizaciju topologije zvijezde, odgovara srednjem kašnjenju OPLK implementacije, odnosno srednjem kašnjenju *3Com* koncentratora (tabela 4.5),
- broj upravljanih čvorova je ograničen na 10 kako bi čvorovi bili sinhronizovani sa preciznošću ispod 1 μs,
- veličina procesnih podataka je varirana u opsegu od 1 do 100 bajtova,
- vrijeme potrebno za prijem i sinhronizaciju SoC okvira odgovara minimalnom razmaku između okvira, tj. $T_W = T_{ifg}$ i
- svi čvorovi su povezani kablovima dužine 10 m.

Promjena minimalnog vremena komunikacionog ciklusa (dužine trajanja izohrone faze) u zavisnosti od broja čvorova za prethodno pomenute načine rada i topologije, data je slici 4.27. Veličina procesnih podataka je fiksirana na 4 bajta.

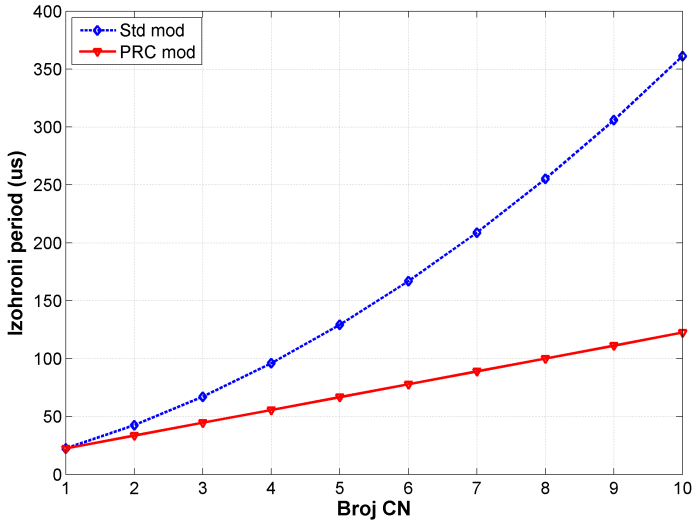
Na slici 4.27 se može uočiti da, za obje topologije, PRC način rada omogućava znatno kraće vrijeme trajanja izohrone faze. U slučaju linijske topologije, kada se koristi 10 čvorova, trajanje izohrone faze pri PRC načinu rada je skoro 3 puta manje u odnosu na standardni način rada. Poboljšanje je nešto manje u slučaju topologije zvijezde, ali je i dalje značajno.

Kako bi se analizirao uticaj veličine procesnih podataka, analiza je ponovljena za slučaj kada se veličina procesnih podataka mijenja u opsegu od 1 do 100 bajtova, pri čemu je broj čvorova fiksiran na 10. Rezultati analize su prikazani na slici 4.28.

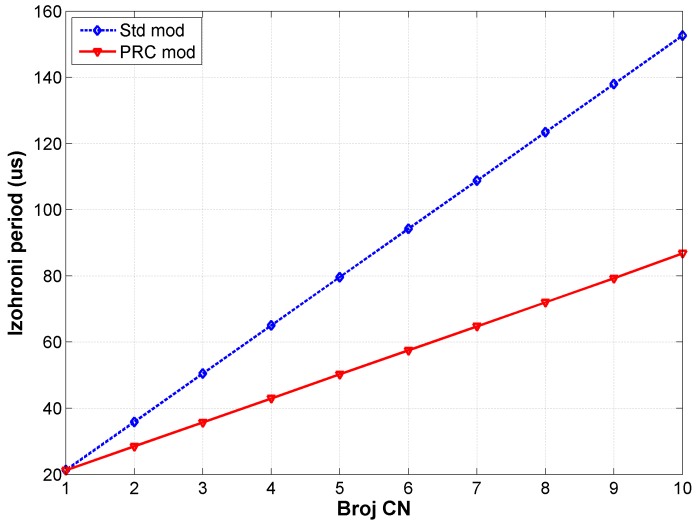
Sa slike se vidi da PRC način rada omogućava bolje performanse za cijeli opseg promjene veličine procesnih podataka, bez obzira na korišćenu topologiju. Poboljšanje performansi je veće kada je veličina procesnih podataka manja od 36 bajtova.

Da bi se bolje prikazalo poboljšanje performansi, prethodno dobijeni rezultati su prezentovani tako da prikazuju odnos dužine trajanja izohrone faze pri PRC i standardnom načinu rada ($T_{ip}^{PRC}/T_{ip}^{Std}$). Slika 4.29 prikazuje ovaj odnos u funkciji broja čvorova i veličine procesnih podataka, pri čemu uvedene iste pretpostavke kao i u prethodne dvije analize.

Kao što može da se vidi, prednost PRC načina rada povećava se sa povećanjem broja čvorova, naročito u slučaju linijske topologije kada je veličina procesnih podataka samo nekoliko bajtova, što je čest slučaj u stvarnim sistemima. Iz prethodne analize je jasno da se značajno poboljšanje performansi postiže uvođenjem PRC načina rada. Međutim, ovaj način ima određene nedostatke



(a)

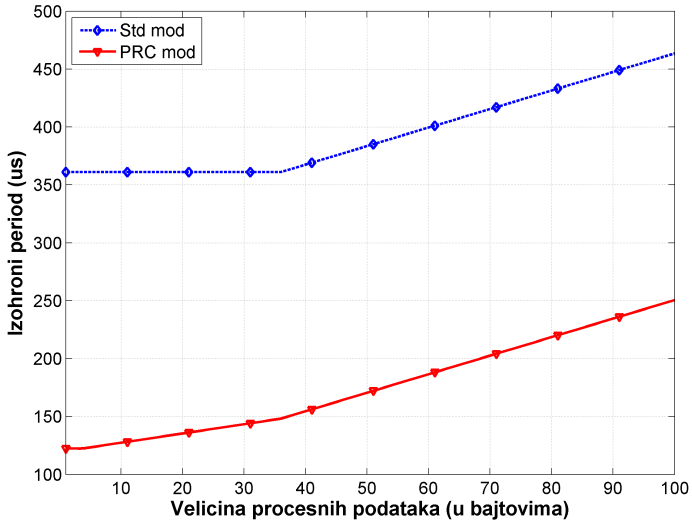


(b)

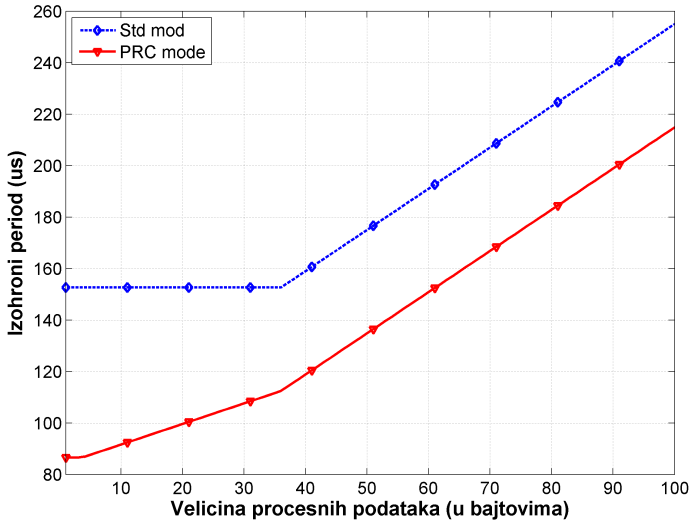
Slika 4.27: Trajanje izohrone faze u funkciji broja uređaja za: (a) linijsku i (b) topologiju zvijezde (veličina procesnih podataka 4 bajta)

čiji efekat može da se ublaži u određenoj mjeri odgovarajućim modifikacijama koje će biti opisane u narednom poglavlju. Takođe, predložene modifikacije omogućavaju dodatno poboljšanje performansi u slučaju kompleksnijih mrežnih struktura kod kojih se kombinuju čvorovi sa PRC i standardnim načinom rada.

Kao i kod ECAT protokola, minimalno vrijeme komunikacionog ciklusa samo po sebi ne određuje ostale metrike. Iz navedenog razloga je sprovedena analiza ukupnog vremena komunikacionog ciklusa i maksimalne propusne moći, pri čemu je propusni opseg asinhrone faze fiksiran na 40% vremena ciklusa kao i u prethodnim analizama. Rezultati analize, za linijsku topologiju i dva načina rada, prikazani su na slici 4.30. Slika prikazuje zavisnost vremena ciklusa i maksimalne propusne moći u funkciji broja uređaja, pri čemu je veličina procesnih podataka 4



(a)

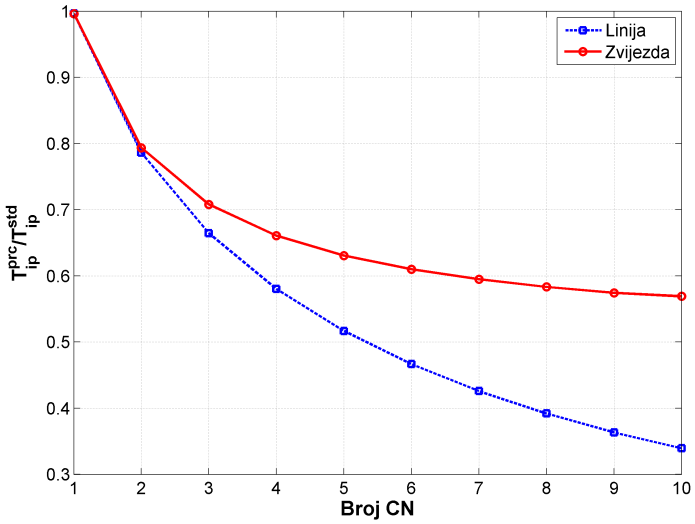


(b)

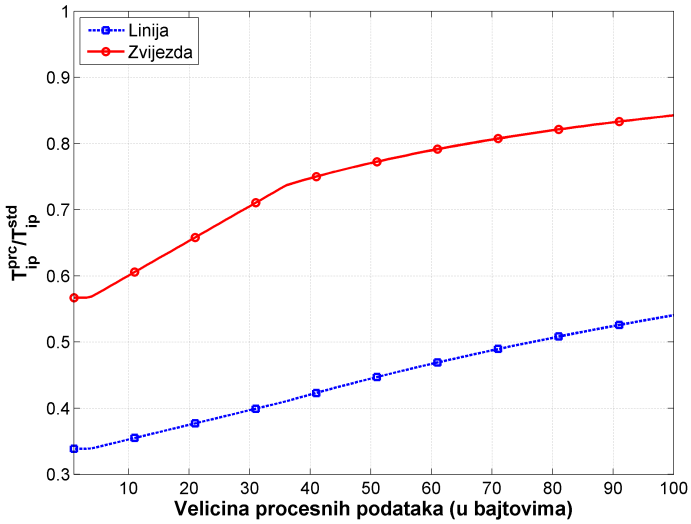
Slika 4.28: Trajanje izohrone faze u funkciji veličine procesnih podataka za: (a) linijsku i (b) topologiju zvijezde (broj čvorova 10)

bajta. Rezultati za topologiju zvijezde su kvalitativno slični rezultatima sa slike 4.30b.

Analizom rezultata sa slike 4.30b, može se zaključiti da oni kvalitativno odgovaraju rezultatima ECAT protokola (s tim da ECAT omogućava mnogo manje vrijeme ciklusa i veću propusnu moć za iste uslove). S druge strane, kod standardnog načina rada (slika 4.30a), uočljivo je da se maksimalna vrijednost propusne moći (nešto iznad 200 kB/s) dostiže kada se koriste dva čvora, nakon čega dolazi do smanjenja njene vrijednosti. Ovo je posljedica uticaja propagacionog kašnjenja kod linijske topologije koje, kod standardnog načina rada, ima kvadratnu zavisnost od broja čvorova. To znači da je, pri većem broju čvorova, dužina trajanja izohrone faze dominantno određena vremenom propagacije signala kroz mrežu, dok je uticaj ukupne veličine procesnih podataka zanemariv. Opisani efekat prvenstveno zavisi od kašnjenja infrastrukurnih kompo-



(a)

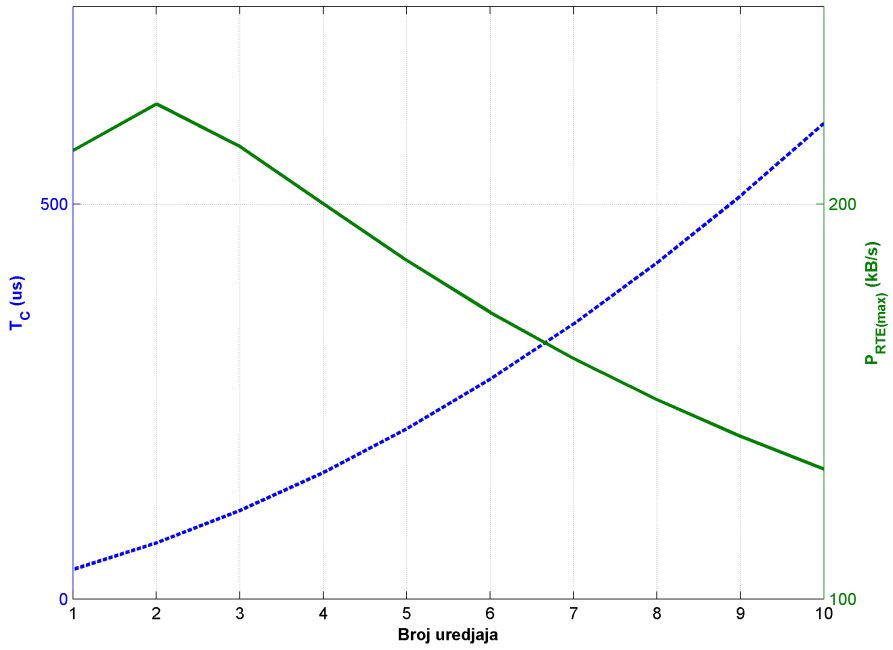


(b)

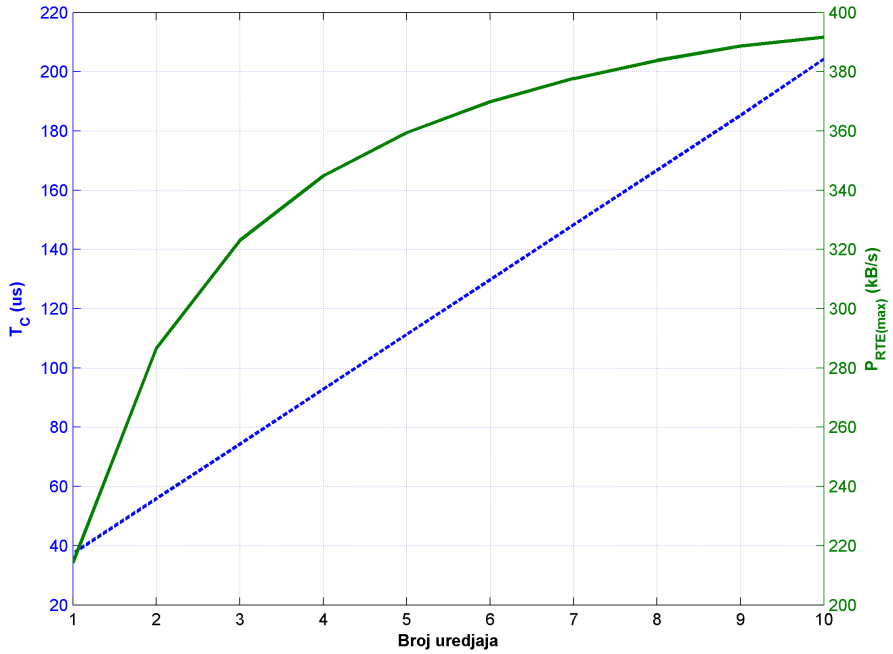
Slika 4.29: Odnos $T_{ip}^{PRC}/T_{ip}^{Std}$ za dvije topologije u funkciji: (a) broja čvorova i (b) veličine procesnih podataka

nenata i nešto je manje izražen pri povećanju veličine procesnih podataka. To je evidentno iz rezultata koji su dobijeni za slučaj topologije zvijezde i kod linijske topologije za PRC način rada (slika 4.30b).

Konačno, kao i u slučaju ECAT protokola, EPL protokol omogućava preciznu sinhronizaciju aktivacije procesa na aplikativnom nivou, tako da vrijeme odziva dominantno zavisi od ukupnog vremena komunikacionog ciklusa koji se postiže u mreži. Važno je napomenuti da, pri manjem broju uređaja, vrijeme ciklusa ima nešto veću vrijednost u odnosu na rezultate sa slike 4.30, jer EPL standard ograničava trajanje asinhrone faze s donje strane na vrijednost datu izrazom (4.23), kao što je prethodno objašnjeno. Ova vrijednost je veća od pretpostavljenih 40% propusnog opsega, što dodatno pogoršava teorijski dobijene rezultate prezentovane na slici 4.30.



(a)



(b)

Slika 4.30: T_C i $P_{RTE(max)}$ u funkciji broja čvorova za linijsku topologiju i (a) standardni i (b) PRC način rada

4.7 Analitički model PNIO protokola

Kao što je objašnjeno u odjeljku 2.3, u PNIO komunikacionom sistemu, kao infrastrukturne mrežne komponente koriste se mrežni komutatori, bilo da se radi o zasebnim, ili o kompo-

nentama integrisanim u same krajnje uređaje. Iz standardnih računarskih mreža, poznato je da mrežni komutatori razdvajaju kolizione domene uređaja i time inherentno omogućavaju potpunu dupleksnu komunikaciju. To znači da se prenos cikličnih podataka, od IOC uređaja prema IOD uređajima i obrnuto, može odvijati istovremeno. Smjer prenosa poruka od IOC uređaja ka IOD uređajima se uobičajeno naziva *downlink* komunikacija, dok se prenos poruka u obrnutom smjeru naziva *uplink* komunikacija.

Minimalno vrijeme komunikacionog ciklusa PNIO protokola, određeno je vremenom potrebnim da se razmjene sve vremenski kritične poruke u sistemu, odnosno odgovara dužini trajanja IRT kanala u slučaju kada se koristi mreža visokih performansi. U opštem slučaju, ovo vrijeme se može izraziti sljedećom jednačinom:

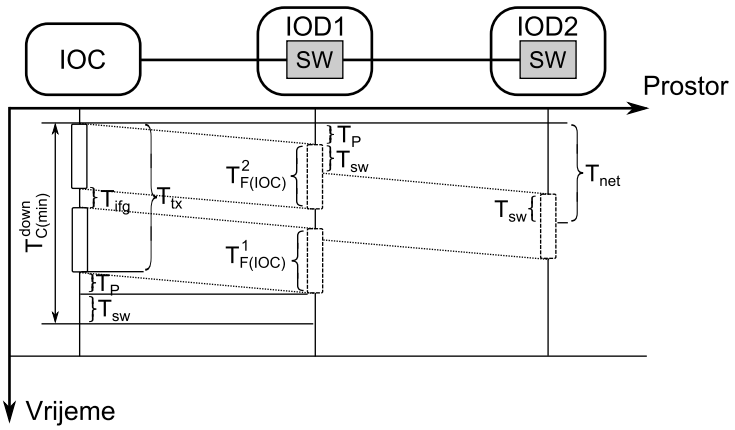
$$T_{C(min)} = T_{sync} + \max\left[T^{down}_{C(min)}, T^{down}_{C(min)}\right], \tag{4.47}$$

gdje T_{sync} označava trajanje sinhronizacionog intervala unutar komunikacionog ciklusa (obično odgovara vremenu slanja sinhronizacione poruke čija je dužina 146 bajtova (vrijeme slanja 11.64 μs) uključujući kontrolne informacije [86]), a $T^{down}_{C(min)}$ i $T^{up}_{C(min)}$ određuju minimalno vrijeme potrebno za razmjenu svih poruka u *downlink* i *uplink* smjeru, redom. Vrijednosti $T^{down}_{C(min)}$ i $T^{up}_{C(min)}$ zavise od više faktora, kao što su topologija mreže, vremenski plan slanja poruka, veličina poruka i kašnjenje infrastrukturnih komponenata. Ukupno vrijeme kašnjenja može se minimizovati ako se za datu topologiju i skup poruka optimizuje vremenski plan slanja tako da sve poruke imaju minimalno vrijeme zadržavanja u mrežnim komutatorima.

U slučaju linijske topologije, trajanje komunikacionog ciklusa može se optimizovati korišćenjem tehnike poznate kao *slipstreaming* mehanizam [17]. U slučaju *downlink* komunikacije, ovaj mehanizam podrazumijeva kreiranje vremenskog plana kod kojeg IOC uređaj prvo šalje poruke najudaljenijim IOD uređajima u mreži kao što je ilustrovano na slici 4.31. U obrnutom smjeru, vremenski plan se optimizuje tako što IOD uređaji koji su najbliži IOC uređaju prvi šalju svoje poruke (slika 4.32).

Parametar T_{FSO} (*FrameSendOffset*) određuje konstantan pomjeraj relativno u odnosu na početak komunikacionog ciklusa pri slanju poruke od strane odgovarajućeg uređaja kako bi se izbjegli konflikti na odredišnim portovima koji dijele različite poruke, čime se minimizuje vrijeme čekanja zbog zauzeća mrežnih resursa.

U slučaju linijske topologije, kada je slanje poruka optimizovano na prethodno opisani način,



Slika 4.31: Ilustracija *downlink* slanja okvira kada se koristi *slipstreaming* mehanizam

minimalno vrijeme komunikacionog ciklusa, za mrežu sa slike 4.31, određeno je sa

$$T_{C(min)}^{down} = T_{tx}^{down} + T_P + T_{sw}, \tag{4.48}$$

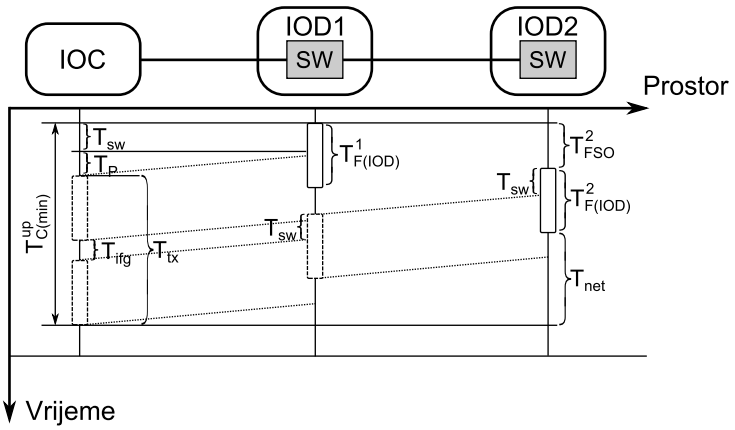
gdje je T_{tx}^{down} vrijeme potrebno za slanje svih okvira od IOC prema IOD uređajima, a T_{sw} vrijeme prosljeđivanja poruke kroz mrežni komutator IOD1 uređaja.

Analizom prostorno-vremenskog dijagrama sa slike 4.32, može se izvesti zaključak da se dobija ekvivalentan izraz, tj.

$$T_{C(min)}^{up} = T_{tx}^{up} + T_P + T_{sw}. \tag{4.49}$$

U opštem slučaju, vrijeme potrebno za slanje svih okvira može da se razlikuje za dva smjera, tj. za mrežu sa n uređaja važe sljedeći izrazi:

$$T_{tx}^{down} = \sum_{i=1}^{n} T_{F(IOC)}^{i} + (n-1)T_{ifg}, \tag{4.50}$$



Slika 4.32: Ilustracija *uplink* slanja okvira kada se koristi *slipstreaming* mehanizam

$$T_{tx}^{up} = \sum_{i=1}^{n} T_{F(IOD)}^{i} + (n-1)T_{ifg}, \tag{4.51}$$

gdje su $T_{F(IOC)}^{i}$ i $T_{F(IOD)}^{i}$, redom, vremena slanja okvira od IOC uređaja prema i-tom IOD uređaju i obrnuto, od i-tog IOD uređaja prema IOC uređaju. S obzirom da su procesni podaci male veličine (nekoliko bajtova) i da mogu da se smjeste u minimalan *Ethernet* okvir, opravdano se može pretpostaviti da je veličina okvira svih uređaja ista, tj. da važi $T_{F(IOC)}^{i} = T_{F(IOD)}^{i} = T_F$ za svako $i = 1, ..., n$. U tom slučaju, dobija se da je vrijeme potrebno za slanje svih poruka, u oba smjera, isto pa se, konačno, uzimajući u obzir jednačinu (4.47), dobija izraz za linijsku topologiju:

$$T_{C(min)} = T_{sync} + nT_F + (n-1)T_{ifg} + T_P + T_{sw}. \tag{4.52}$$

U opštem slučaju, vrijeme potrebno za slanje PNIO okvira određeno je jednačinom (4.2), pri čemu se za izračunavanje dužine okvira L_F koriste sljedeće formule:

$$L_{F(IOC)} = \max(L_{ovh} + m_s(L_{IOCS} + L_{IOPS}) + m_i L_{IOCS} + m_o L_{IOPS} + L_{data}, L_{F(min)}), \tag{4.53}$$

$$L_{F(IOD)} = \max(L_{ovh} + m_s(L_{IOCS} + L_{IOPS}) + m_i L_{IOPS} + m_o L_{IOCS} + L_{data}, L_{F(min)}), \tag{4.54}$$

gdje $L_{F(IOC)}$ i $L_{F(IOD)}$ određuju dužinu odgovarajućih okvira izraženu brojem bajtova. Ostali parametri iz jednačina (4.53) i (4.54) imaju sljedeće značenje: $L_{ovh} = 32$ – ukupna dužina kontrolnih informacija protokola (uključujući kontrolne informacije *Ethernet* protokola), m_s – broj sistemskih modula koji ne prenose procesne podatke (npr. DAP, modul za napajanje itd.), m_i – broj ulaznih modula, m_o – broj izlaznih modula, L_{IOCS} – dužina IOCS polja (tipično 1 bajt), L_{IOPS} – dužina IOPS polja (tipično 1 bajt) i L_{data} – ukupna veličina procesnih podataka. Prethodne formule su izvedene na osnovu organizacije procesnih podataka modularnog uređaja kao što je ilustrovano na slici 2.23.

U slučaju topologije zvijezde, takođe, može da se primijeni *slipstreaming* mehanizam za kreiranje vremenskog plana. Izraz koji opisuje minimalno vrijeme ciklusa u tom slučaju je:

$$T_{C(min)} = T_{sync} + nT_F + (n-1)T_{ifg} + 2(T_P + T_{sw}). \tag{4.55}$$

Razlog zbog kojeg se posljednja dva člana u jednačini (4.52) množe sa dva u prethodnom izrazu je činjenica da se za realizaciju topologije zvijezde koristi dodatni mrežni komutator i kabl za povezivanje svih uređaja.

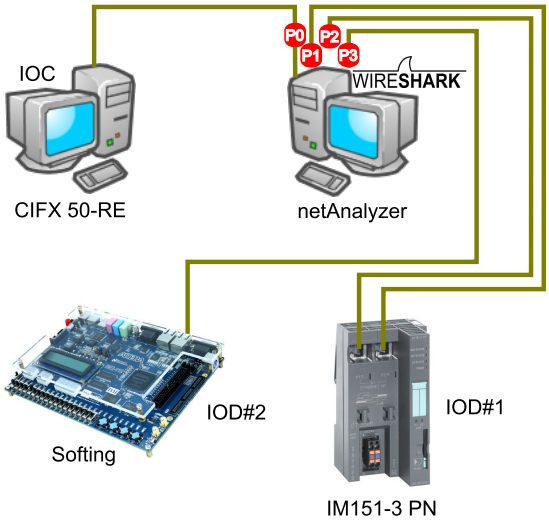
Kompleksnije topologije koje se realizuju kombinacijom prethodnih topologija zahtijevaju

precizniju analizu, jer minimalno vrijeme ciklusa zavisi od algoritma kojim se optimizuje vremenski plan slanja poruka. Problem postaje još kompleksniji kada se koriste različite dužine okvira, ili kada se uređaji povezuju u topologiju prstena pri čemu se sve poruke u mreži dupliraju kako bi se obezbijedio visok nivo redundanse koja omogućava efikasan oporavak mreže od grešaka.

4.8 Karakterizacija vremenskih parametara PNIO analitičkog modela

Kao što može da se vidi iz jednačina analitičkog modela izvedenog u prethodnom odjeljku, za analizu karakteristika i procjenu metrika PNIO mreže potrebno je kvantifikovati odgovarajuće vremenske parametre PNIO komponenata. To podrazumijeva kašnjenje koje unose infrastrukturne komponente (T_{sw}), kao i kvalitativnu analizu cikličnog ponašanja IOC i IOD uređaja. S ciljem karakterizacije pomenutih vremenskih parametara realizovano je nekoliko mjerenja na eksperimentalnoj postavci prikazanoj na slici 4.33.

Prototipsku mrežu čine komponente različitih proizvođača, dok je za mjerenje relevantnih vremenskih parametara korišćen *netAnalyzer* kao i kod prethodna dva protokola. U konkretnom slučaju, kao IOC uređaj korišćena je *Hilscher* implementacija bazirana na prethodno pomenutoj PCI kartici CIFX 50-RE ugrađenoj u PC računar. Kao IOD uređaji, korišćene su implementacije dva proizvođača: *Softing* realizacija IOD uređaja na razvojnom sistemu *Altera* DE2-115 i komercijalno dostupna IM151-3 PN modularna struktura iz serije ET200S, proizvođača *Siemens*. Uređaji su povezani kablovima dužine 2 m, što znači da se, pri analizi, vrijeme propagacije signala



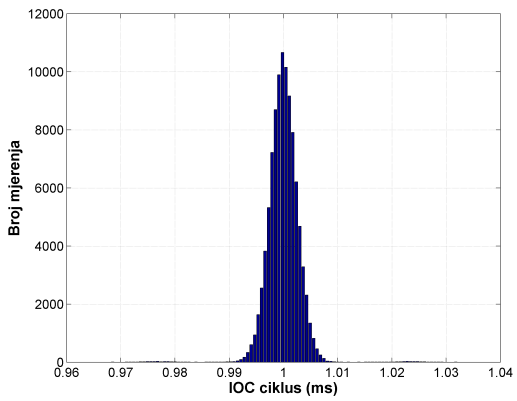
Slika 4.33: Eksperimentalna postavka PNIO mreže

kroz kablove može zanemariti.

Vrijeme komunikacionog ciklusa je konfigurisano tako da bude jednako 1 ms, a dužina trajanja svakog eksperimenta je definisana tako da se prikupi najmanje 30000 mjerenja. Ovdje je važno napomenuti da CIFX 50-RE implementacija IOC uređaja ne podržava sinhronizovanu IRT komunikaciju, što treba imati u vidu prilikom analize rezultata mjerenja, naročito ako se radi o kašnjenju komutatora, jer NRT okviri, zbog nedostatka sinhronizacije vremena ciklusa u uređajima, mogu da interferiraju sa RT okvirima koje prosljeđuje komutator. Ovo ni na koji način ne utiče na validnost mjerenih rezultata, jer oba IOD uređaja podržavaju IRT komunikaciju i način rada komutatora sa direktnim prosljeđivanjem okvira.

Prvi eksperiment se odnosi na kvalitativnu i kvantitativnu analizu cikličnog ponašanja PNIO uređaja. S tim u vezi, izmjerene su vrijednosti perioda cikličnih RT poruka koje šalju IOC i IOD uređaji. Period komunikacionog ciklusa IOC uređaja (T_C^{IOC}) dobija se oduzimanjem vremena prijema dvije uzastopne ciklične poruke određenog identifikatora okvira (polje FID) na portu P0. Kvalitativni rezultati mjerenja ovog perioda su prikazani na slici 4.34, dok je pregled statistike dat u tabeli 4.6.

Kao što može da se zaključi iz dobijenih rezultata, srednja vrijednost vremena ciklusa IOC uređaja je bliska konfiurisanoj vrijednosti, dok je standardna devijacija ispod 3 μs (apsolutno odstupanje ispod 70 μs). Na sličan način, može se ocijeniti vrijeme komunikacionog ciklusa IOD uređaja, s tim da je potrebno izolovati okvire sa FID poljem koje odgovara RT porukama svakog uređaja koje se primaju portovima P1 i P3 za IM151-3 PN i *Softing* implementaciju, redom. Rezultati dobijeni mjerenjem za ova dva uređaja su prikazani na slici 4.35 i u tabeli 4.6. Analizom dobijenih rezultata i njihovim poređenjem sa ciklusom IOC uređaja može se zaključiti da IOD uređaji omogućavaju mnogo preciznije generisanje cikličnih okvira (apsolutno odstupanje je manje od 15 μs kod IM151-3 PN, odnosno 1 μs kod *Softing* implementacije). Ovakvi rezultati



Slika 4.34: Izmjereni period komunikacionog ciklusa IOC uređaja

Tabela 4.6: Pregled statistike mjerenih vremenskih parametara

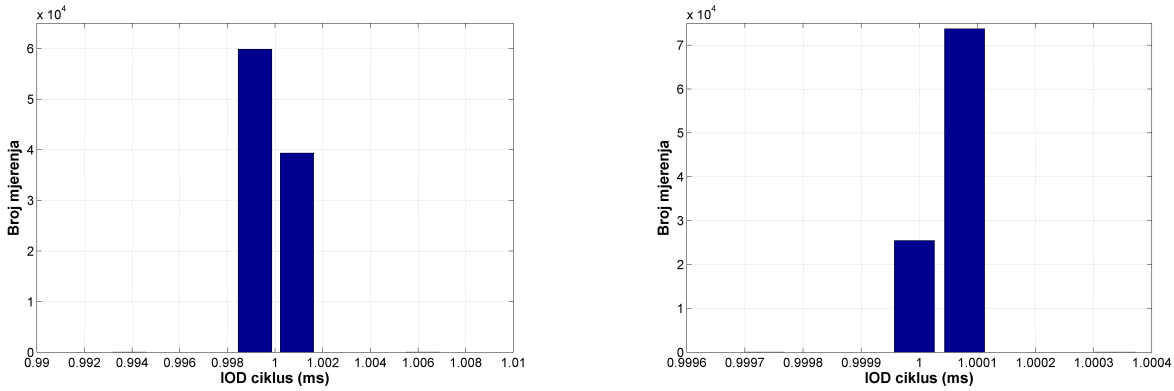
Parametar	Min.	Max.	μ	σ
T_C^{IOC}	968.19	1032.08	999.998	2.878
$T_{C(Softing)}^{IOD}$	999.69	1000.38	1000.044	0.019
$T_{C(Siemens)}^{IOD}$	993	1007.08	1000.04	0.061
$T_{sw}^{Softing}$	3.909	4.579	4.371	0.094
$T_{sw}^{Siemens}$	5.711	5.791	5.753	0.012

Sve vrijednosti su izražene u μs

se mogu objasniti manjom kompleksnošću zahtijevanih funkcionalnosti IOD uređaja u odnosu na IOC uređaj.

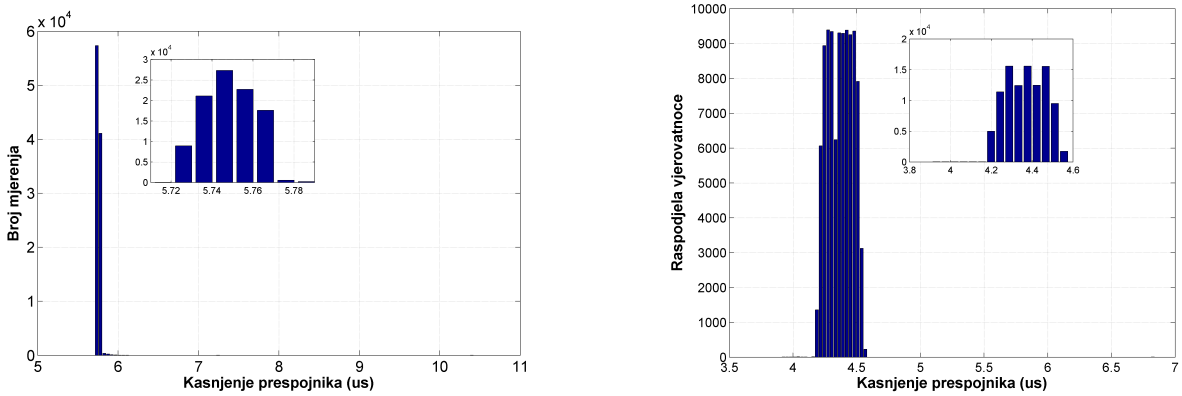
Kašnjenje komutatora može se odrediti ako se oduzme vrijeme prijema istog okvira na portovima P0 i P2 za slučaj IM151-3 PN (slika 4.33). Ekvivalentno, kašnjenje komutatora *Softing* implementacije dobija se na isti način, pri čemu je topologiju mreže sa slike 4.33 potrebno modifikovati tako da ova dva uređaja zamijene mjesto. Kvalitativni rezultati, za oba komutatora, prikazani su na slici 4.36.

Analizom rezultata, zaključuje se da je maksimalno kašnjenje kod IM151-3 PN komutatora 10.421 μs, a kod *Softing* implementacije 6.839 μs. Međutim, detaljnijom analizom saobraćaja, utvrđeno je da su navedene vrijednosti posljedica interferencije RT i NRT okvira zbog nedostatka sinhronizacije uređaja kao što je ranije objašnjeno. Jasno je da se pomenuta kašnjenja javljaju sporadično kao efekat zauzeća odredišnog porta prilikom prosljeđivanja određenih RT okvira. Kada se iz rezultata filtriraju ova mjerenja, dobijaju se konzistentniji rezultati za kašnjenje, kao što je ilustrovano histogramima unutar slika. Kvantitativna ocjena ovih rezultata je data u tabeli 4.6.



(a) (b)

Slika 4.35: Izmjereni period komunikacionog ciklusa IOD uređaja: (a) IM151-3 PN i (b) *Softing* implementacija



(a) (b)

Slika 4.36: Kašnjenje komutatora integrisanog u (a) IM151-3 PN i (b) *Softing* IOD uređaj

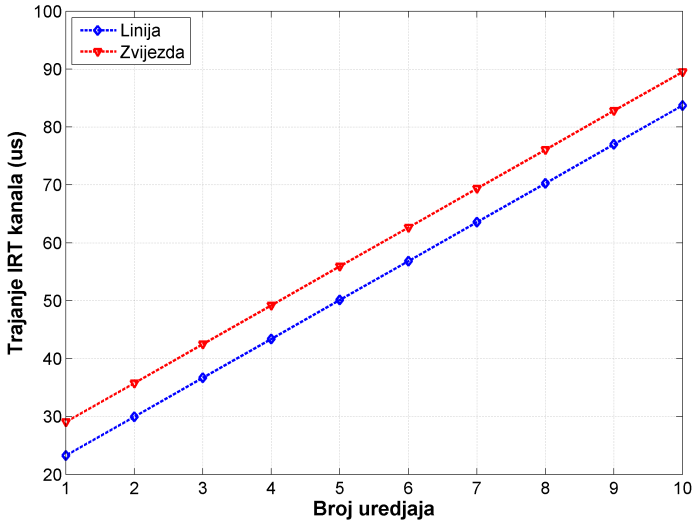
Kao što može da se vidi, *Softing* implementacija ima manju srednju vrijednost kašnjenja u odnosu na IM151-3 PN uređaj, ali je standardna devijacija veća skoro za red veličine.

4.9 Teorijska analiza PNIO protokola

Karakteristike distribuiranog sistema automatizacije zasnovanog na PNIO komunikacionom sistemu zavise od različitih parametara definisanih PNIO standardom. Kao što je opisano u odjeljku 2.3, ovaj sistem podržava različite klase saobraćaja koje mogu da se kombinuju u sistemu, pri čemu RTC3 klasa omogućava najbolje performanse sa preciznom sinhronizacijom uređaja. Osim toga, PNIO sistem uvijek rezerviše određeni dio komunikacionog ciklusa za prenos NRT poruka, što može dodatno da komplikuje analizu.

Iako standardom nije definisana količina propusnog opsega, koji se rezerviše za NRT saobraćaj, preporuka je da to bude bar 40%. S druge strane, većina konfiguracionih alata zahtijeva od korisnika da dužina trajanja ovog intervala bude najmanje 125 μs, kako bi se garantovao prenos jednog *Ethernet* okvira maksimalne dužine (1518 bajtova sa kontrolnim informacijama). Ovo može značajno da utiče na određene metrike, kao što je vrijeme odziva. Tako, na primjer, u mreži kod koje je dužina trajanja IRT kanala 75 μs, ukupno vrijeme ciklusa iznosi 200 μs, što znači da je minimalno vrijeme odziva koje se može postići u najlošijem slučaju oko 400 μs. Međutim, posljednja verzija PNIO standarda podržava fragmentaciju NRT okvira, čime je ovaj problem prevaziđen.

U odjeljku 2.3 je objašnjeno da PNIO protokol podržava različite klase saobraćaja. Međutim, s obzirom da je predmet istraživanja ove disertacije koncentrisan na protokole koji omogućavaju najviše performanse, prilikom analize će se smatrati da je PNIO sistem konfigurisan tako



Slika 4.37: Trajanje IRT faze u funkciji broja uređaja za dvije topologije

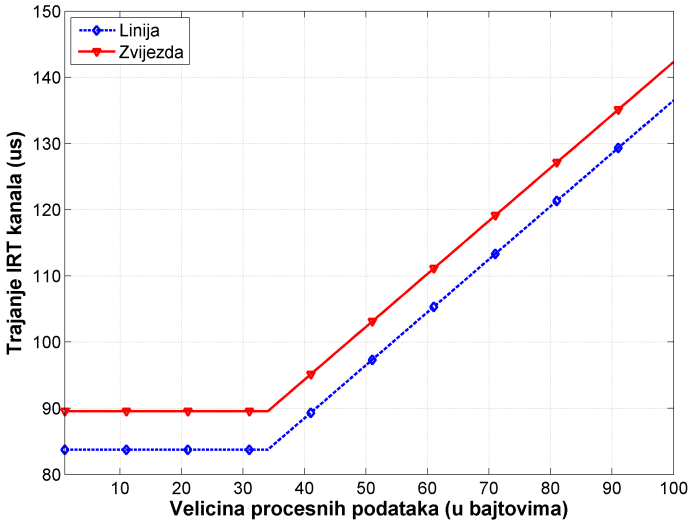
da postoji samo IRT faza sa RTC3 klasom saobraćaja. Dužina trajanja IRT faze odgovara minimalnom komunikacionom ciklusu i zavisi od različitih faktora poput topologije mreže, broja čvorova i veličine procesnih podataka. Osim toga, ova metrika zavisi i od vremenskog plana slanja vremenski kritičnih poruka.

Pri analizi PNIO sistema, usvojene su sljedeće pretpostavke: maksimalan broj IOD uređaja je ograničen na 10, kao i u prethodnim slučajevima, veličina procesnih podataka se mijenja u opsegu od 1 do 100 bajtova, vremenski plan slanja poruka je organizovan u skladu sa opisanim *slipstreaming* efektom, što predstavlja optimalan izbor za razmatrane topologije (linijska i topologija zvijezde), kašnjenje komutatora je određeno srednjim kašnjenjem IM151-3 PN uređaja (tabela 4.6), dužina IOCS i IOPS polja je po jedan bajt. Konačno, pretpostavljeno je da su korišćeni kompaktni IOD uređaji sa fiksnom konfiguracijom modula (dva sistemska modula, jedan ulazni i jedan izlazni modul) i da su svi uređaji povezani kablovima dužine 10 m.

Promjena dužine trajanja IRT faze u zavisnosti od broja uređaja, za analizirane topologije, prikazana je na slici 4.37. Veličina procesnih podataka je postavljena na 4 bajta. Slika 4.37 pokazuje da linijska topologija omogućava nešto kraće vrijeme ove faze, jer se ne koristi dodatni komutator za povezivanje krajnjih uređaja.

Na slici 4.38, prikazana je zavisnost ove faze od veličine procesnih podataka, pri čemu je broj uređaja postavljen na 10. Sa slike 4.38 je jasno da, za veličinu procesnih podataka manju od 36 bajtova, dužina trajanja IRT kanala ostaje nepromijenjena. S povećanjem veličine podataka, počinje linearno da se povećava i vrijeme trajanja ove faze. Ovo je posljedica *Ethernet* dopune za slučaj kada je veličina procesnih podatka nedovoljna da se dobije minimalna veličina okvira.

Kao i u slučaju prethodnih protokola, sprovedena je analiza ukupnog vremena komunika-



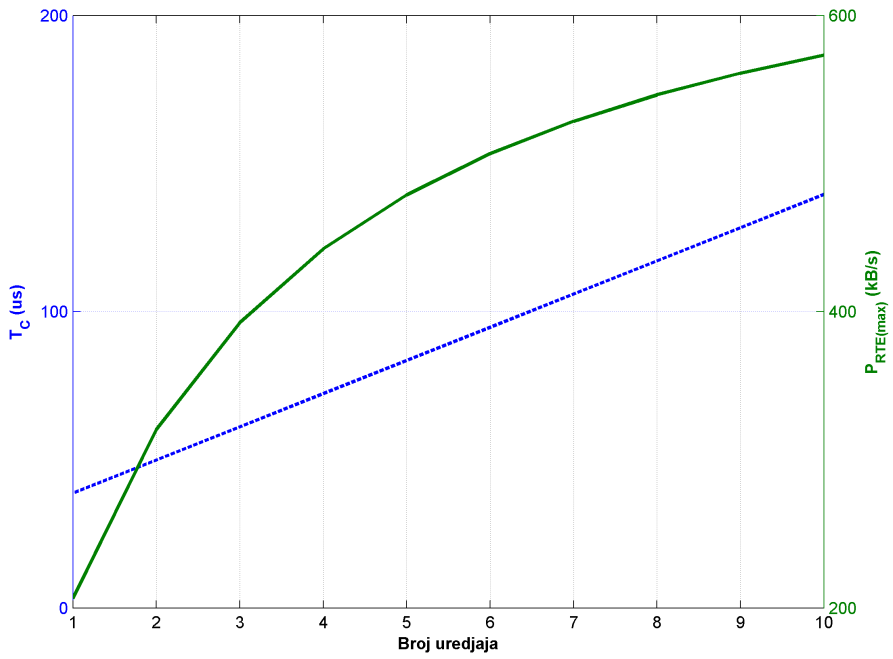
Slika 4.38: Trajanje IRT faze u funkciji veličine procesnih podataka za dvije topologije

cionog ciklusa i maksimalne propusne moći za PNIO protokol, pri čemu je interval rezervisan za NRT saobraćaj fiksiran na 40% ukupnog propusnog opsega ekvivalentno prethodnim razmatranjima. Rezultati ove analize, za dvije topologije, prikazani su na slici 4.39, koja prikazuje zavisnost vremena ciklusa i maksimalne propusne moći u funkciji broja uređaja, pri čemu je veličina procesnih podataka 4 bajta.

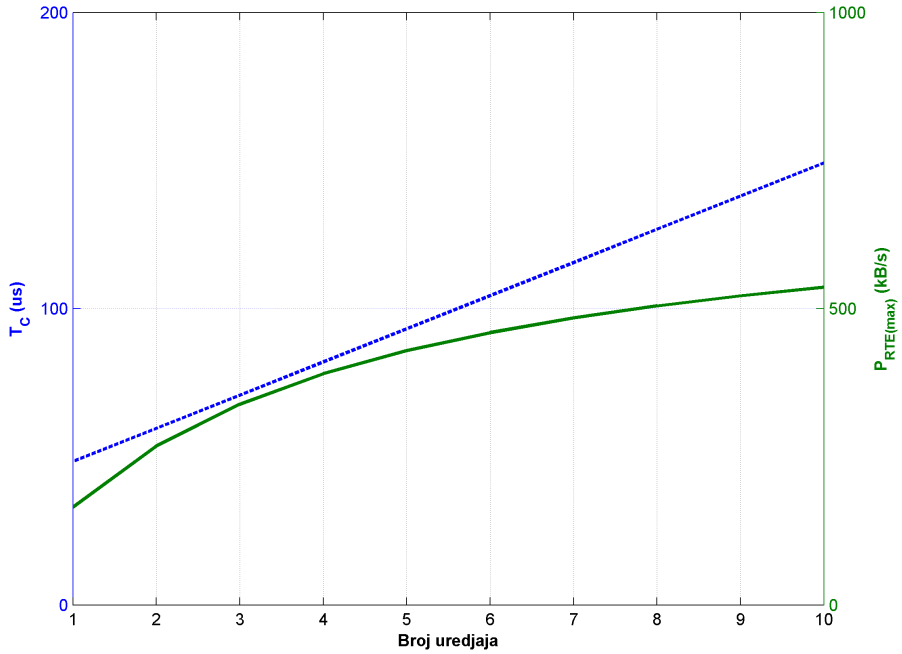
Rezultati analize pokazuju da PNIO sistem ima slične karakteristike za dvije topologije, pri čemu su rezultati kvalitativno slični rezultatima iz prethodnih analiza. Performanse PNIO sistema, u smislu analiziranih metrika, nalaze se u oblasti između ECAT i EPL sistema.

PNIO protokol obezbjeđuje precizne nivoe sinhronizacije između uređaja uz uslov da broj kaskadno vezanih mrežnih komutatora ne prelazi određenu vrijednost, kao što je opisano u odjeljku 3.1.9. Međutim, da bi se postigla sinhronizacija i na nivou aplikacije, interna arhitektura kontrolera i distribuiranih uređaja mora da bude organizovana na odgovarajući način, pri čemu se za aktivaciju lokalnih aplikacija koriste globalno sinhronizovani hardverski signali [30, 34]. Sinhronizacijom svih uređaja na nivou aplikacije omogućava da vrijeme odziva, kao i kod prethodna dva rješenja, dominantno zavisi od vremena ciklusa.

Iz prethodne analize se zaključuje da se optimizacija vremena ciklusa, a samim tim i vremena odziva, može ostvariti optimizacijom vremenskog plana slanja poruka unutar sinhronizovanog IRT kanala, uz pretpostavku da je mrežna konfiguracija takva da se duži NRT okviri mogu fragmentisati, čime je dio propusnog opsega sveden na razumnu mjeru. Takođe, jasno je da kritičan faktor, kod PNIO komunikacije, predstavlja neefikasno iskorišćenje propusnog opsega pri maloj veličini procesnih podataka, što je tipičan slučaj u distribuiranim sistemima automatizacije.



(a)

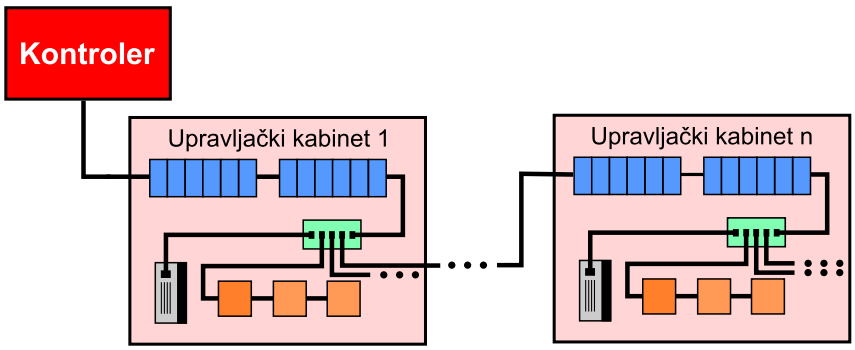


(b)

Slika 4.39: T_C i $P_{RTE(max)}$ u funkciji broja uređaja za (a) linijsku i (b) topologiju zvijezde

4.10 Uporedna analiza industrijskih komunikacionih sistema

Tipična struktura industrijskog komunikacionog sistema koji se koristi u stvarnim sistemima automatizacije, prikazana je na slici 4.40. Mreža se sastoji iz upravljačkih ormara (engl. *control cabinet*) povezanih linijskom topologijom. Svaki upravljački ormar obično sadrži više različitih



Slika 4.40: Tipična struktura mreže u sistemima automatizacije

uređaja koji obavljaju određene zadatke u procesu proizvodnje. Ovi uređaji se mogu povezivati različitim topološkim strukturama, u zavisnosti od vrste komunikacionog sistema.

U slučaju ECAT sistema, na primjer, koriste se modularni uređaji povezani u linijsku topologiju sa eventualnim ograncima koji su povezani na centralni modularni komutator. Kod EPL i PNIO sistema, u upravljačkom ormaru se obično nalazi jedna ili više infrastrukturnih komponenata (koncentratora ili komutatora) u hijerarhijskoj topologiji zvijezde kojom se povezuju krajnji uređaji, čija struktura može da bude kompaktna ili modularna.

U cilju poređenja mogućnosti prethodno opisanih i analiziranih komunikacionih protokola, kada se oni koriste u distribuiranim sistemima automatizacije, analiziran je scenario sistema čija je topološka struktura slična strukturi prikazanoj na slici 4.40.

Prilikom analize, uvedene su sljedeće pretpostavke: sistem se sastoji iz 10 upravljačkih ormara povezanih kablovima dužine 50 m, svaki ormar sadrži 5 uređaja (dva modularna uređaja sa odgovarajućim brojem ulazno/izlaznih modula i tri servo ose realizovane kao kompaktni uređaji sa fiksnom konfiguracijom), koji su međusobno povezani kablovima dužine 5 m. Struktura procesnih podataka korišćenih uređaja, prikazana je u tabeli 4.7 i obezbjeđeni su mehanizmi za sinhronizaciju svih uređaja u mreži.

Tabela 4.7: Informacije o korišćenim uređajima

Uređaj/modul	**Broj modula**	**Ibits**	**Obits**
Modularni uređaj 1			
Analogni ulaz	4	16	-
Analogni izlaz	2	-	16
Modularni uređaj 2			
Digitalni ulaz	8	4	-
Digitalni izlaz	16	-	4
Servo osa	-	96	96
Ukupno:	300	3840	3840

Kao što može da se vidi iz tabele, prvi modularni uređaj sadrži 4 analogna ulazna modula sa 16-bitnim ulaznim procesnim podacima i 2 analogna izlazna modula sa 16-bitnim izlaznim procesnim podacima. Drugi modularni uređaj se sastoji iz 8 digitalnih ulaznih modula, čija je veličina ulaznih procesnih podataka 4 bita, i 16 digitalnih izlaznih modula sa izlaznim procesnim podacima veličine 4 bita. Uređaj koji ima funkcionalnost servo ose ima 96 bita ulaznih i 96 bita izlaznih procesnih podataka. Struktura svih uređaja odgovara strukturi stvarnih uređaja, koji su komercijalno dostupni.

Kompletan sistem sadrži 50 uređaja, od čega je 20 modularnih uređaja koji ukupno imaju 300 modula, što odgovara sistemu automatizacije srednje složenosti. Ukupna veličina ulazno-/izlaznih procesnih podataka izračunata na osnovu prethodnih informacija iznosi 960 bajtova (480 bajtova ulaznih i 480 bajtova izlaznih procesnih podataka).

Kao što je već opisano, modularni uređaji kod ECAT sistema se povezuju E-bus magistralom tako da svaki modul čini zaseban ECAT uređaj. Ulazni podaci modularnih uređaja mapiraju se u isti telegram sa LRD komandom, dok su izlazni podaci mapirani u zaseban LWR telegram. S obzirom da servo ose imaju simetričnu strukturu procesnih podataka (isti broj ulaza i izlaza), njihovi podaci se mogu razmjenjivati unutar jednog LRW telegrama u oba smjera. Pretpostavljeno je da svi uređaji, osim digitalnih modula, podržavaju CoE komunikaciju pa je potrebno koristiti dodatni LRD telegram za čitanje statusa CoE bafera.

Kako bi se poboljšala dijagnostika u sistemu, uređaji su organizovani u logičke grupe ili domene (engl. *domains*). U konkretnom slučaju, definisano je 10 domena, po jedan za svaki kontrolni ormar. Koncept domena podrazumijeva razdvajanje procesnih podataka, koji pripadaju različitim domenina, u različite telegrame. Ovaj pristup će biti detaljnije objašnjen u narednom poglavlju. S obzirom na prethodno navedeno, struktura ECAT okvira se može opisati kao što je prikazano u tabeli 4.8.

Iz tabele se može zaključiti da ECAT okvir sadrži ukupno 34 telegrama, dok je ukupna

Tabela 4.8: Organizacija cikličnog ECAT telegrama za razmatrani sistem

Telegram	**Veličina**	**Broj telegrama**
NOP	4 B	1
ARMW	4 B	1
LRD	12 B	1
LRW	36 B	10
LWR	12 B	10
LRD	12 B	10
BRD	2 B	1
Ukupno:	622 B	34

veličina polja korisnih podataka telegrama jednaka 622 bajta. Ukupna veličina okvira, prema tome, iznosi 1058 bajtova, što odgovara vremenu slanja od 84.64 μs.

Kod EPL i PNIO sistema, modularni i kompaktni uređaji unutar upravljačkog ormara povezani su na centralnu infrastrukturnu komponentu (koncentrator kod EPL, odnosno mrežni komutator kod PNIO), čime se lokalno formira topologija zvijzde.

Korišćenjem jednačina izvedenih analitičkih modela i eksperimentalno dobijenih vremenskih parametara, sprovedena je uporedna analiza za navedena tri protokola u smislu metrika definisanih u okviru poglavlja 3. Rezultati dobijeni navedenom analizom dati su u tabeli 4.9.

Prilikom analize, pretpostavljeno je da sistem rezerviše dovoljno propusnog opsega kako bi se omogućio prenos jednog *Ethernet* okvira maksimalne dužine (125 μs). Važno je napomenuti da, u slučaju ECAT protokola, maksimalan okvir ne može da se prenese bez fragmentacije, jer se za prenos NRT poruka koristi koncept tunelovanja okvira, što podrazumijeva njihovu enkapsulaciju unutar ECAT okvira. To znači da *Ethernet* okvir maksimalne dužine ne može da se prenese u okviru jednog komunikacionog ciklusa. Kod PNIO i EPL sistema to nije slučaj, jer se NRT poruke prenose u rezervisanom intervalu.

S obzirom da sva tri rješenja omogućavaju preciznu sinhronizaciju na nivou aplikacija, vrijeme odziva mreže odgovara dvostrukoj vrijednosti ukupnog komunikacionog ciklusa (T_C) uz zanemarenje kašnjenja obrade signala u ulazno/izlaznim uređajima.

Kao što može da se vidi iz tabele 4.9, najbolje performanse u smislu prenosa vremenski kritičnih podataka postiže ECAT protokol. S druge strane, iako je za NRT saobraćaj rezervisan veći procenat propusnog opsega u odnosu na druga dva rješenja, prenos NRT poruka kod ECAT protokola je manje efikasan zbog efekata koje unosi koncept tunelovanja okvira. Metrika koja se odnosi na broj komutatora (N_{sw}), primjenjiva je samo u slučaju PNIO protokola, jer druga dva rješenja ne koriste ove uređaje. Takođe, iz tabele se vidi da je u slučaju ECAT sistema korišćeno

Tabela 4.9: Uporedna analiza industrijskih komunikacionih sistema

Parametar	**ECAT**	**EPL_STD**	**EPL_PRC**	**PNIO**
$T_{C(min)}$ (μs)	209.059	927.72	708.406	358.501
T_{NRT} (μs)	125	145.056	145.056	125
T_C (μs)	334.059	1072.776	853.462	483.501
B_{NRT} (%)	37.41	13.52	16.99	25.85
P_{RTE} (MB/s)	2.874	0.894	1.125	1.986
T_{RT} (μs)	667.318	2145.552	1706.924	967.002
N_{sw}	–	–	–	10
N_{es}	330	50	50	50

znatno više krajnjih stanica (N_{es}), što je posljedica načina implementacije modularnog uređaja u ECAT sistemu.

4.11 Zaključak

U ovom poglavlju su predloženi novi analitički modeli RTE sistema koji su predmet istraživanja ove disertacije. Predloženi analitički modeli predstavljaju proširenje istraživanja autora koje je već bilo izloženo kritici javnosti kroz nekoliko radova prezentovanih na međunarodnim i nacionalnim konferencijama [23, 46, 72, 82, 90].

Vremenski parametri modela su određeni i karakterizovani eksperimentalnim putem na prototipima stvarnih mreža, koji sadrže uređaje različitih proizvođača. Pokazano je da je, za sve razmatrane slučajeve, srednja vrijednost kašnjenja podređenog ECAT uređaja manja od 1.2 μs sa standardnom devijacijom ispod 20 ns. Srednja vrijednost kašnjenja ECAT sprežnika na E-bus magistralu je za oko 40% manja u odnosu na kompaktni uređaj, dok je srednja vrijednost kašnjenja ECAT terminala povezanih E-bus interfejsom manja od 200 ns sa standardnom devijacijom od 3 ns. S druge strane, period komunikacionog ciklusa nadređenog uređaja, zavisi od realizacije vremenski kritičnih komponenata njegovog komunikacionog steka. Kod softverske implementacije bazirane na operativnom sistemu opšte namjene sa podrškom za rad u realnom vremenu, standardna devijacija komunikacionog ciklusa je nekoliko mikrosekundi (ispod 10 μs) sa apsolutnim odstupanjem oko 200 μs. Kod hardverski podržane realizacije, standardna devijacija je oko 1.7 μs, pri čemu apsolutno odstupanje nije veće od 20 μs.

U slučaju EPL sistema, tipično kašnjenje koncentratora je ispod 600 ns sa standardnom devijacijom 12 ns. Izuzetak je *openPOWERLINK* implementacija, koja unosi dodatno kašnjenje od oko 1.7 μs zbog konverzije između MII i RMII interfejsa na posmatranoj platformi. EPL čvorovi koji podržavaju tzv. *auto-response* mehanizam omogućavaju vrijeme odziva ispod 1 μs sa standardnom devijacijom od 20 ns. Komunikacioni džiter upravljačkog uređaja je dominatno određen načinom implementacije i, u slučaju softverske realizacije iznosi oko 3 μs, dok se kod hardverski podržane realizacije može smanjiti i do vrijednosti 20 ns.

Kod PNIO protokola je pokazano da kašnjenje mrežnog komutatora iznosi oko 6 μs u svim razmatranim slučajevima, dok je komunikacioni džiter IOC uređaja oko 3 μs, a IOD uređaja ispod 100 ns. Takođe, značajno je napomenuti da vrijeme prosljeđivanja okvira kroz mrežni komutator dominantno zavisi od mogućnosti sinhronizacije svih čvorova u mreži. Kada se koriste klase saobraćaja koje ne podržavaju sinhronizaciju, vrijeme zadržavanja RT okvira u komutatoru može da bude produženo, jer je odredišni port blokiran zbog slanja NRT poruke iz

prethodnog komunikacionog ciklusa.

Nakon izvođenja analitičkih modela i ekperimentalne karakterizacije njihovih parametara, sprovedena je teorijska analiza razmatranih RTE sistema za dvije osnovne mrežne topologije (linijsku i topologiju zvijezde) u smislu prethodno definisanih metrika, pri čemu je variran broj uređaja i veličina procesnih podataka u sistemu.

Pokazano je da, u slučaju ECAT sistema, vrijeme potrebno za slanje okvira predstavlja kritičan faktor pri manjem broju uređaja, dok vrijeme propagacije signala kroz mrežu postaje dominantno pri veoma velikom broju čvorova, nezavisno od korišćene topologije.

Kod EPL protokola, način rada sa ulančavanjem PRes okvira (PRC) postiže za preko 60% bolje performanse u odnosu na standardni način rada ako se koristi linijska topologija sa 10 čvorova i malom veličinom procesnih podataka (4 bajta).

Osnovni nedostatak PNIO protokola odražava se u činjenici da se, pri smanjenju veličine procesnih podataka, smanjuje efikasnost iskorišćenja propusnog opsega, bez obzira na broj korišćenih uređaja. Takođe, značajno je pomenuti da vrijeme odziva zavisi od vremenskog plana slanja okvira unutar IRT kanala.

Konačno, u cilju kvantitativne ocjene mogućnosti različitih RTE sistema, sprovedena je uporedna analiza pomenutih RTE sistema na primjeru mreže čija struktura odgovara konfiguracijama stvarnih sistema automatizacije. Rezultat ove analize pokazuje da ECAT protokol postiže najbolje performanse po pitanju prenosa vremenski kritičnih podataka (za oko 40% kraće vrijeme odziva u odnosu na PNIO, i preko 70% kraće vrijeme odziva u odnosu na EPL protokol).

5 Mehanizmi optimizacije vremena odziva

Vrijeme odziva predstavlja veoma značajnu metriku u industrijskim komunikacionim sistemima koji se koriste za realizaciju distribuiranih sistema automatizacije. S obzirom da RTE sistemi, analizirani u prethodnom poglavlju, omogućavaju preciznu sinhronizaciju na nivou čitavog sistema, uključujući lokalne aplikacije na strani kontrolera i distribuiranih uređaja, vrijeme odziva je dominantno određeno komunikacionim ciklusom koji može da se postigne u sistemu.

Kao što je opisano u odjeljku 3.2.2, ovo vrijeme je određeno različitim komponentama kašnjenja kontrolera, mreže i distribuiranih uređaja, koje su posljedica obrade, zauzeća resursa i nedostatka sinhronizacije. Vremena obrade u kontroleru (dužina izvršavanja upravljačkog zadatka lokalne aplikacije) i u distribuiranim uređajima (npr. vrijeme aktivacije procesa lokalne aplikacije u cilju odmjeravanja i konverzije analognog signala u digitalni oblik), mogu se svesti na minimalnu mjeru adekvatnim projektovanjem softverskih i hardverskih komponenata. Problemi sa zauzećem resursa i efektima, koji nastaju zbog nedostatka sinhronizacije, rješavaju se pažljivim planiranjem komunikacije i obezbjeđenjem odgovarajućih mehanizama za sinhronizaciju svih komponenata sistema. Kašnjenje, koje nastaje kao posljedica obrade u mrežnim komponentama (npr. vrijeme potrebno za prosljeđivanje okvira u mrežnom komutatoru), može se optimizovati modifikacijom odgovarajućih komunikacionih mehanizama protokola, što je predmet ovog poglavlja.

Poglavlje je organizovano na sljedeći način. U odjeljku 5.1, opisana su dva pristupa za optimizaciju ECAT protokola. Prvi pristup opisuje optimizaciju propagacionog kašnjenja ECAT mreže, dok se drugi koncentriše na mogućnost optimizacije dužine sumacionog okvira. Odjeljak 5.2 uvodi mehanizam baziran na ulančavanju PRes okvira za optimizaciju EPL mreže. U odjeljku 5.3, opisan je i detaljno analiziran DFP (engl. *Dynamic Frame Packing*) koncept koji omogućava optimizaciju PNIO protokola. Konačno, odjeljkom 5.4, zaključuje se poglavlje i komnetarišu prezentovani rezultati.

5.1 Optimizacija ECAT protokola

Iako koncepti sumacionog okvira i logičkog adresiranja kod ECAT protokola omogućavaju veoma dobru efikasnost komunikacije, naročito po pitanju iskorišćenja ukupnog propusnog opsega, ukupne performanse mogu da postanu kritičan faktor u slučaju veoma velikih mreža koje sadrže stotine distribuiranih uređaja. Ovo je posljedica logičke topologije prstena koja zahtijeva da okvir, koji generiše nadređeni uređaj, prolazi kroz svaki čvor dva puta, u direktnom i povratnom smjeru, što značajno utiče na ukupno vrijeme propagacije mreže, koje direktno zavisi od broja mrežnih uređaja. Osim toga, brojem uređaja, određena je ukupna veličina sumacionog okvira, od koje direktno zavisi vrijeme potrebno za slanje okvira. Svi ovi faktori utiču na minimalno vrijeme komunikacionog ciklusa koje se može postići u ECAT sistemu, a samim tim i na ostale metrike definisane u poglavlju 3.

U cilju poboljšanja performansi ECAT sistema, mogu se primjeniti različiti pristupi. Najjednostavniji pristup podrazumijeva povećanje bitske brzine sa 100 Mb/s na 1 Gb/s. Tada je očekivano da se vrijeme potrebno za slanje okvira smanji za jedan red veličine kao što je pretpostavljeno u [17–19]. Međutim, ovo je vrlo gruba i pojednostavljena procjena poboljšanja performansi sistema, jer, prilikom analize, nisu uzeti u obzir svi efekti gigabitskog *Ethernet* sistema propisani standardom.

Prvi efekat se odnosi na dužinu trajanja minimalnog okvira. Naime, da bi se omogućila kompatibilnost rada sa prethodnim verzijama standarda, uveden je koncept *produženja nosioca* (engl. *Carrier Extension*) za kratke okvire [87]. Efektivno, dobija se da je minimalna veličina okvira proširena na 512 bajtova, što, u slučaju okvira koji ne prenose veliku količinu podataka, dovodi do još značajnijeg smanjenja efikasnosti propusnog opsega. Ovaj efekat se, do određene mjere, može umanjiti ukoliko se okviri šalju u naletu (engl. *burst*) korišćenjem tehnike pod nazivom *Frame Bursting* [88].

Drugi efekat se odnosi na mogućnost korišćenja tzv. *jumbo* okvira (okviri maksimalne veličine 9000 bajtova), koji dodatno poboljšavaju iskorišćenje propusnog opsega kada se koristi koncept sumacionog okvira.

Konačno, mrežne komponente koje se koriste u gigabitskim *Ethernet* sistemima su osjetljivije na smetnje, što sužava oblast njihove primjene u industrijskim sistemima.

U drugom pristupu poboljšanja performansi ECAT sistema, razmatra se mogućnost smanjenja vremena propagacije signala kroz mrežne komponente korišćenjem tehnike paralelizacije linijskih ogranaka, čime se efektivno jedan logički prsten velike dužine ”razbija” u više logičkih prstenova sa manjim vremenom propagacije. Okviri koje šalje nadređeni uređaj se raspoređuju

paralelno prema linijskim ograncima koji su povezani na centralni komutator. Ovaj pristup pod nazivom *Switched EtherCAT* je predložen i analiziran u [89], dok je hardverska realizacija specifično razvijenog mrežnog komutatora predložena u [47].

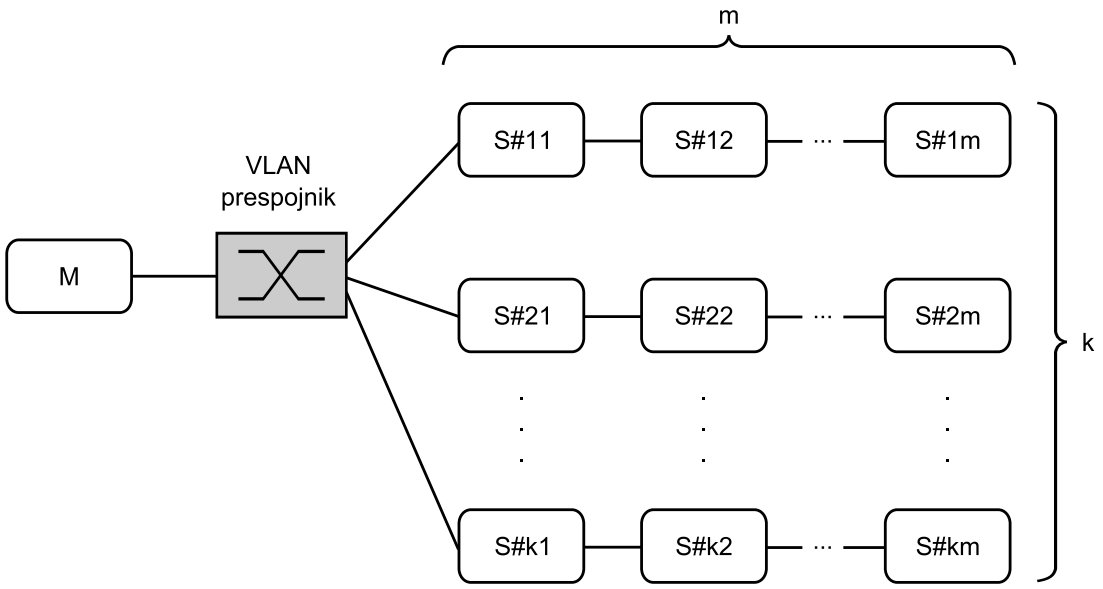
Iako omogućava značajnu redukciju vremena propagacije, rješenje sa specijalnim komutatorom nije optimalno rješenje sa aspekta dostupnosti zahtjevanih komponenata, jer je, pri realizaciji mreže, potrebno koristiti komponente koje nisu standardne. Kako bi se prevazišao ovaj problem, u [90], predloženo je rješenje sa standardnim mrežnim komutatorima koji podržavaju VLAN oznake. Međutim, ni u jednoj od pomenutih studija, ne analizira se izbor optimalnog broja portova komutatora koji u opštem slučaju omogućava minimizaciju vremena komunikacionog ciklusa.

Treći pristup se koncentriše na optimizaciju veličine sumacionog okvira. U tu svrhu koristi se osobina prostorne simetričnosti uređaja u ECAT mreži i logičko adresiranje sa LRW komandom. U slučaju idealno simetrične mreže, veličina okvira može da se smanji i do 50% kao što je opisano u [82]. U [43] je predložen heuristički algoritam za optimizaciju veličine sumacionog okvira za proizvoljnu mrežnu konfiguraciju koja može da sadrži kompaktne i modularne uređaje, dok je u [91] ovaj algoritam proširen tako da podrži grupisanje uređaja u fizičke cjeline u cilju definisanja fizičkih ograničenja koje postoje u stvarnim uslovima, kao i logičke jedinice ili domene koji poboljšavaju dijagnostiku grešaka u sistemu. U navedenim radovima, međutim, nije analizirana optimalnost, kao ni procjena kompleksnosti predloženih algoritama, što se daje u ovom poglavlju.

5.1.1 Optimizacija propagacionog kašnjenja ECAT mreže

Iz analize u okviru odjeljka 4.3, može se zaključiti da kašnjenje koje unose mrežne komponente postaje dominantno u odnosu na vrijeme slanja okvira kada mreža sadrži veći broj uređaja, naročito ako je veličina procesnih podataka malena i ako se koriste komponente koje imaju veće kašnjenje (npr. *Hilscher* implementacija bazirana na CIFX 50-RE). Linijska topologija sa velikim brojem uređaja se može restrukturirati tako da se čvorovi grupišu u više linijskih ogranaka koji se povezuju sa centralnom infrastrukturnom komponentom kao što je prikazano na slici 5.1. Na ovaj način se veoma dugačaka linija dijeli u više segmenata manje dužine, čime se omogućava smanjenje propagacionog kašnjenja za $k = n/m$ puta, gdje k odgovara broju portova mrežnog komutatora, m broju uređaja u svakom ogranku, a n ukupnom broju uređaja.

S obzirom da ECAT koristi specijalne načine adresiranja i da podređeni uređaji ne interpretiraju MAC adrese, pri realizaciji mrežne strukture sa slike 5.1 nije moguće koristiti konven-



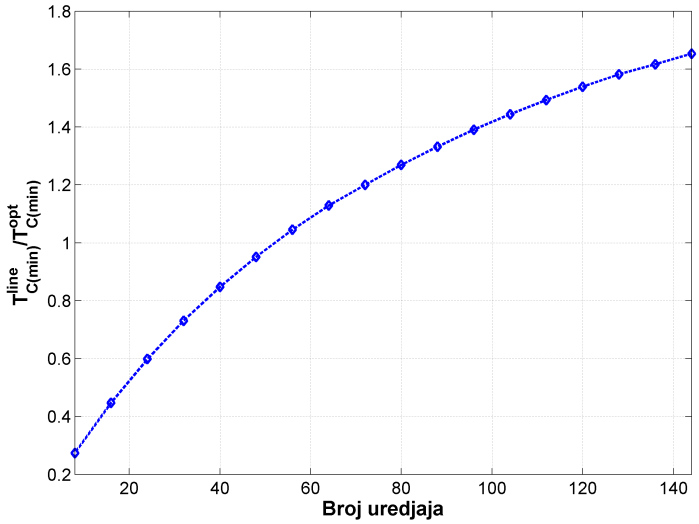
Slika 5.1: Topologija ECAT mreže sa mrežnim komutatorom i linijskim ograncima

cionalne mrežne komutatore. Kako bi se ovaj problem prevazišao, koriste se komutatori koji podržavaju VLAN označavanje okvira. Naime, nadređeni uređaj može da ostvari komunikaciju u više sumacionih okvira (po jedan za svaki linijski ogranak) koji sadrže VLAN oznake. Mrežni komutator interpretira označene okvire i prosljeđuje ih odgovarajućim ograncima. Podređeni uređaji ne modifikuju ove oznake, tako da mrežni komutator u povratnom smjeru zna gdje treba proslijediti svaki od okvira.

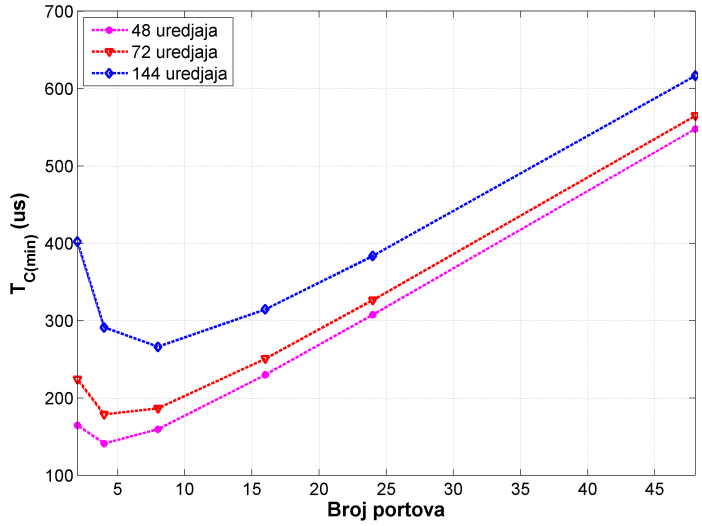
S obzirom da opisani pristup ne zahtijeva modifikaciju hardvera i da se može realizovati komercijalno dostupnim komponentama uz odgovarajuće softverske modifikacije nadređenog uređaja, opisani pristup je mnogo povoljniji u odnosu na rješenje opisano u [47].

Korišćenjem jednačina analitičkog modela ECAT protokola koje su izvedene u odjeljku 4.1, može se sprovesti teorijska analiza efikasnosti opisanog rješenja. U tom smislu, analizirana je mogućnost minimizacije minimalnog vremena ciklusa, kao metrike koja dominatno utiče na druge karakteristike sistema uključujući i vrijeme odziva. Prilikom analize, uvedene su sljedeće pretpostavke: uređaji su povezani kablovima dužine 10 m i ukupna veličina procesnih podataka je 8 bajtova sa 75% asimetrije između ulaza i izlaza, kao što je bio slučaj u prethodnim analizama. Osim toga, maksimalan broj uređaja je ograničen na 144, a prosječna vremena kašnjenja podređenog uređaja odgovaraju vrijednostima *Hilscher* implementacije datim u tabeli 4.2. Konačno, pretpostavljeno je da mrežni komutator ima 8 portova.

Odnos vrijednosti minimalnog vremena ciklusa, koji se dobija linijskom i topologijom zvijezde sa ograncima, u funkciji broja uređaja, prikazan je na slici 5.2. Sa slike se vidi da predloženo rješenje ima prednost u odnosu na linijsku topologiju, ako je broj uređaja veći od 48. Takođe, minimalno vrijeme komunikacionog ciklusa je manje za više od 30% u odnosu na linijsku



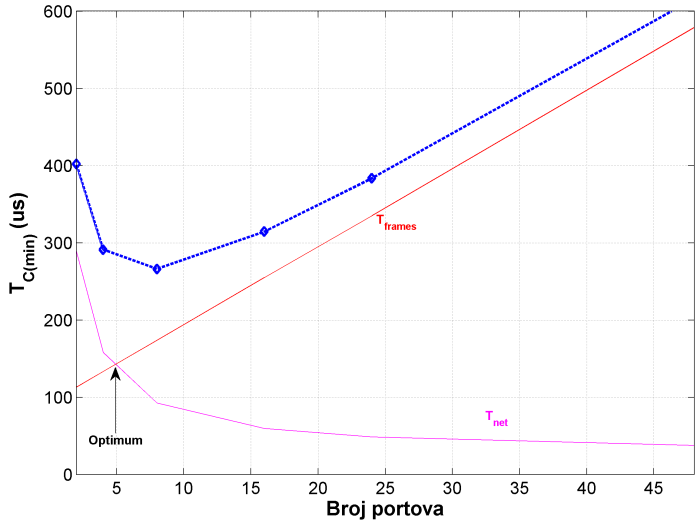
Slika 5.2: Zavisnost odnosa $T_{C(min)}$ linijske i poboljšane topologije od broja uređaja



Slika 5.3: Zavisnost $T_{C(min)}$ od broja portova i broja uređaja

topologiju, ako je broj uređaja veći od 100. Broj portova mrežnog komutatora, takođe, utiče na efikasnost komunikacije. Kako bi se ovaj efekat detaljnije istražio, sprovedena je analiza za scenarije sa različitim brojem uređaja, pri čemu je broj portova variran u opsegu 2 do 48. Rezultati analize su prikazani na slici 5.3.

Kao što može da se vidi sa slike 5.3, za svaki scenario, može se uočiti lokalni minimum za koji se postižu najbolje performanse. Za scenario sa 48 i 72 uređaja, optimalan broj portova mrežnog komutatora je 4, dok se za scenario sa 144 uređaja ova vrijednost povećava na 8. Ovaj efekat se može objasniti činjenicom da veći broj portova podrazumijeva veći broj okvira koje šalje nadređeni uređaj. Iako se vrijeme propagacije signala kroz mrežu smanjuje pri povećanju broja portova (odnosno linijskih ogranaka), vrijeme slanja okvira postaje dominantno, što dovodi do narušavanja efikasnosti rješenja.



Slika 5.4: Ilustracija određivanja lokalnog minimuma za slučaj sa 144 uređaja

Detaljnijom analizom dobijenih rezultata i predloženih matematičkih modela, može se zaključiti da se lokalni minimum nalazi u presječnoj tački funkcija koje predstavljaju vrijeme potrebno za slanje poruka i propagaciono kašnjenje mreže, kao što je ilustrovano na slici 5.4. Ovo ujedno predstavlja mjeru za određivanje optimalnog broja portova mrežnog komutatora s ciljem minimizacije komunikacionog ciklusa.

U opštem slučaju, algoritam za raspoređivanje uređaja u linijske ogranke treba da uzme u obzir ukupno vrijeme slanja okvira i vrijeme propagacije kroz komutator i linijske ogranke i da na prethodni način optimizuje broj ogranaka i broj uređaja u svakom ogranku.

5.1.2 Optimizacija dužine ECAT okvira

U ECAT mrežama, veličina okvira se može smanjiti iskorišćenjem potencijalne simetrije uređaja u smislu veličine ulaznih i izlaznih procesnih podataka. U mreži sa simetričnom prostornom raspodjelom uređaja koji imaju jednak broj ulaza i izlaza, veličina ECAT telegrama, potrebnog za razmjenu podataka, može se smanjiti za pola upotrebom LRW komande.

Na primjer, ako modularni ECAT uređaj, koji se na mrežu povezuje pomoću sprežnika (npr. EK1100), sadrži 4 ulazna terminala sa 4-bitnim procesnim podacima (npr. EL1004) i 4 izlazna terminala sa 4-bitnim izlaznim podacima (npr. EL2004), veličina LRW telegrama bez kontrolnih informacija mora da bude minimalno 4 bajta, kada se ulazni terminali nalaze fizički prije izlaznih terminala. U slučaju zasebnih komandi za čitanje (LRD) i upis (LWR), dobija se još gori slučaj, jer su kontrolne informacije ECAT telegrama (12 bajtova) dvostruko veće. Ukoliko se promijeni redoslijed terminala u modularnom uređaju, tako da izlazni terminali budu fizički prije ulaznih, za LRW telegram je dovoljno da se rezerviše prostor korisnih podataka veličine 2 bajta. Naime,

Algoritam 5.1: Algoritam za optimizaciju dužine okvira

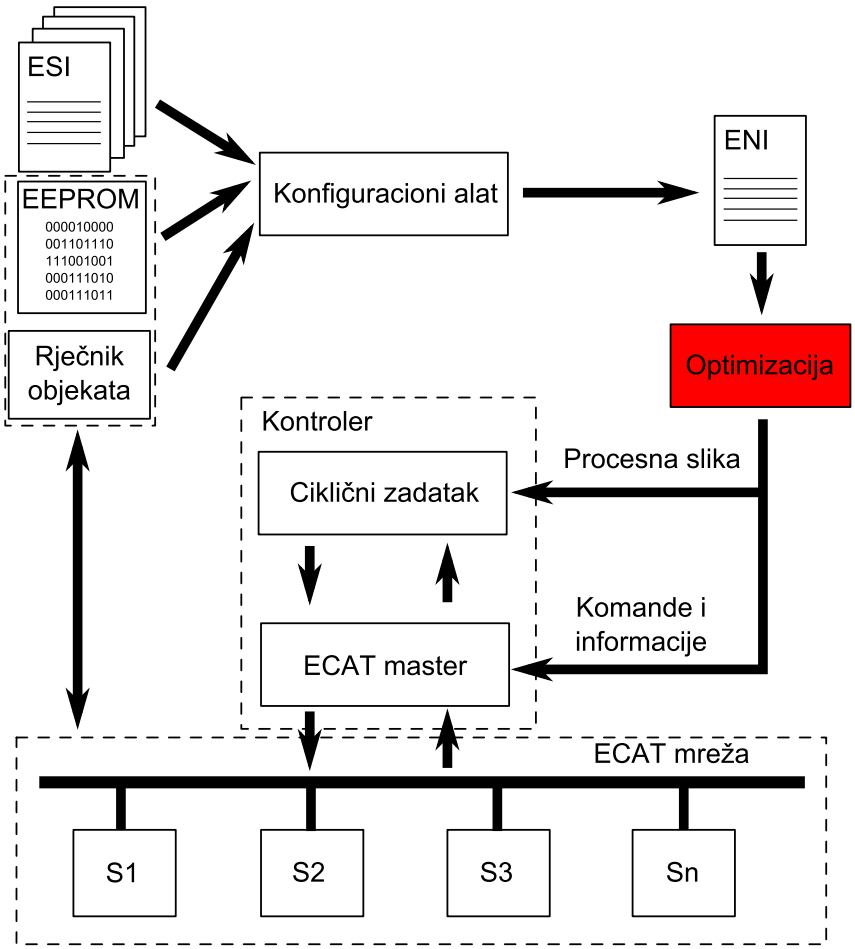
```
1:  Postavi LogAddrI = LogAddrO na početnu vrijednost
2:  for i = 1 → n do
3:      Diff = LogAddrI − LogAddrO
4:      if Diff > Slave(i).Diff then
5:          Pronađi i < k ≤ n za koje je Diff ≤ Slave(k).Diff
6:          if k nije pronađeno then
7:              Pronađi i < k ≤ n za koje je min(|Slave(k).Diff|)
8:          end if
9:          Zamijeni uređaje Slave(i) i Slave(k)
10:     end if
11:     if Slave(i).Obits ≠ 0 then
12:         Izračunaj FMMU parametre izlaza za i-ti uređaj
13:         if Diff > 0 then
14:             LogAddrO = LogAddrI + Slave(i).Obits
15:         else
16:             LogAddrO = LogAddrO + Slave(i).Obits
17:         end if
18:     end if
19:     if Slave(i).Ibits ≠ 0 then
20:         Izračunaj FMMU parametre ulaza za i-ti uređaj
21:         LogAddrI = LogAddrI + Slave(i).Ibits
22:     end if
23: end for
```

kako se dostavljeni izlazni podaci ne koriste u ostatku mreže, njihov prostor u telegramu može da se iskoristi za dostavljanje ulaznih podataka koji se čitaju iz ulaznih terminala.

Algoritam za optimizaciju veličine okvira (algoritam 5.1), realizovan je u specifično razvijenom konfiguracionom alatu koji koristi informacije o trenutnoj konfiguraciji ECAT mreže, koja je definisana u okviru ENI fajla. Ovaj fajl može da se generiše korišćenjem nekog od komercijalno dostupnih alata za konfiguraciju ECAT sistema (npr. *Beckhoff TwinCAT System Manager* [92] and *KPA Studio EtherCAT* [93]).

Kao što je opisano u odjeljku 2.1.5, ENI fajl sadrži informacije o konfiguraciji svih podređenih uređaja (komande za inicijalizaciju parametara uređaja, veličinu procesnih pdataka, itd.), kao i o strukturi sumacionog okvira koji se razmjenjuje ciklično u mreži. Izmjenom sadržaja ovog fajla, moguće je prilagoditi konfiguraciju mreže tako da se optimizuje dužina okvira uz promjenu načina povezivanja ECAT uređaja. Integracija procesa optimizacije veličine okvira u standardni tok konfiguracije ECAT sistema, ilustrovana je na slici 5.5.

Osnovna ideja, na kojoj se algoritam 5.1 zasniva, je u tome da se fizičke lokacije uređaja u mreži organizuju tako da se prilikom prolaska okvira kroz uređaje prvo upisuju izlazni podaci, a zatim čitaju ulazni podaci od uređaja, tj. da se upis/čitanje obavlja naizmjenično. Na taj način



Slika 5.5: Proces optimizacije veličine okvira prilikom konfiguracije

se prostor koji u telegramu zauzimaju izlazni podaci, koristi za transport ulaznih podataka.

Na početku algoritma, definišu se vrijednosti logičkih adresa za ulaze ($LogAddrI$) i izlaze ($LogAddrO$). Inicijalno, ove vrijednosti su jednake i obično mogu da se postave na nulu. Zatim, u svakoj iteraciji, izračunava se razlika između logičke adrese izlaza i ulaza ($Diff$). Ova razlika se poredi sa razlikom između veličine izlaznih i ulaznih procesnih podataka i-tog uređaja ($Slave(i).Diff$), koja se može izračunati korišćenjem $Slave(i).Obits - Slave(i).Ibits$, gdje $Ibits$ i $Obits$ predstavljaju veličinu ulaznih i izlaznih procesnih podataka, redom, izraženu brojem bita. Ako je $Diff > Slave(i).Diff$, to znači da ulazni podaci uređaja ne mogu da se smjeste u prostor rezervisan za izlazne podatke i da se moraju poravnati sa logičkom adresom ulaza iz prethodne iteracije kako bi se spriječilo preklapanje podataka. Ovim se smanjuje iskorišćenje propusnog opsega, a time i efikasnost komunikacije. Da bi prevazišao opisani problem, predloženi algoritam pretražuje ostatak mreže kako bi pronašao k-ti uređaj za koji važi $Diff \leq Slave(k).Diff$. Ako takav uređaj postoji u mreži, algoritam ga zamjenjuje sa uređajem iz trenutne iteracije (i-ti uređaj), u suprotnom, zamjenjuje se uređaj sa minimalnom razlikom $|Slave(k).Diff|$.

Drugi dio algoritma je zadužen za izračunavanje logičkih adresa $LogAddrI$ i $LogAddrO$ sljedećeg uređaja. Kao što je već pomenuto, za uređaj kod kojeg je $Diff \leq Slave.Diff$, logička adresa izlaza mora da se poravna sa završetkom logičke adrese ulaza. Logičke adrese ulaza i izlaza se izračunavaju jednostavno dodavanjem veličine ulaznih/izlaznih procesnih podataka vrijednosti logičke adrese iz prethodne iteracije (algoritam 5.1). Parametri FMMU jedinice uređaja, mogu se izračunati na sljedeći način:

$$LogStartAddr = \left\lceil \frac{LogAddrX}{8} \right\rceil, \tag{5.1}$$

$$Length = \left\lceil \frac{Xbits}{8} \right\rceil, \tag{5.2}$$

$$LogStartBit = LogAddrX - 8\left(\left\lceil \frac{LogAddrX}{8}, \right\rceil - 1\right), \tag{5.3}$$

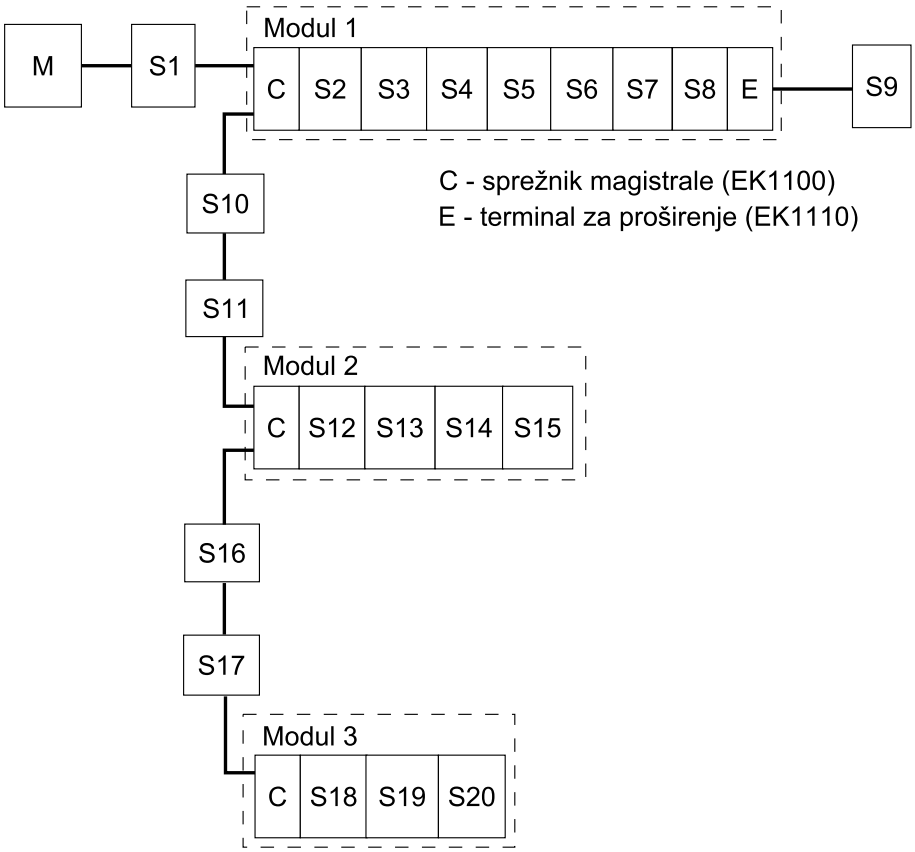
$$LogStopBit = NewLogAddrX - 8\left(\left\lceil \frac{NewLogAddrX}{8}, \right\rceil - 1\right) - 1, \tag{5.4}$$

gdje je: X – I (ulazi) ili O (izlazi), $NewLogAddrX$ – vrijednost $LogAddrX$ za sljedeću iteraciju, $LogStartAddr$ – početna logička adresa, $Length$ – veličina procesnih podataka izražena brojem bajtova, $LogStartBit$ – početni bit logičke adrese, $LogStopBit$ – posljednji bit logičke adrese. Ostali parametri FMMU jedinice uređaja (početna fizička adresa, početni bit fizičke adrese, tip i fleg aktivacije FMMU jedinice) mogu ostati neizmjenjeni.

Važno je napomenuti da se, prethodno opisani algoritam, primjenjuje hijerarhijski, tj. prvo se primjenjuje na nivou ECAT modularnog uređaja (fizička preraspodjela terminala od kojih se uređaj sastoji), a zatim na kompletnu mrežu u smislu fizičkog preraspoređivanja uređaja povezanih mrežnim kablovima. Na taj način, zadržava se fizička lokacija terminala unutar odgovarajućeg modularnog uređaja (terminali ne mogu da se izmjeste van uređaja kojem pripadaju). Ovo je veoma bitno sa aspekta očuvanja konzistencije i integriteta strukture modularnih uređaja u sistemu.

Opisani algoritam je primjenjen na relativno jednostavnu mrežu čija je topologija prikazana na slici 5.6. Detaljnije informacije o uređajima koji su korišćeni za realizaciju date mreže se mogu pronaći u tabeli 5.1.

Sa slike 5.6, uočljivo je da se mreža sastoji iz tri modularna uređaja (moduli 1, 2 i 3) različite strukture i broja terminala i 6 kompaktnih uređaja sa različitom organizacijom procesnih podataka. Inicijalni ENI fajl je generisan korišćenjem *TwinCAT System Manager* konfiguracionog alata kompanije *Beckhoff*. Algoritam predložen u ovom radu, primjenjen je za dva scenarija.



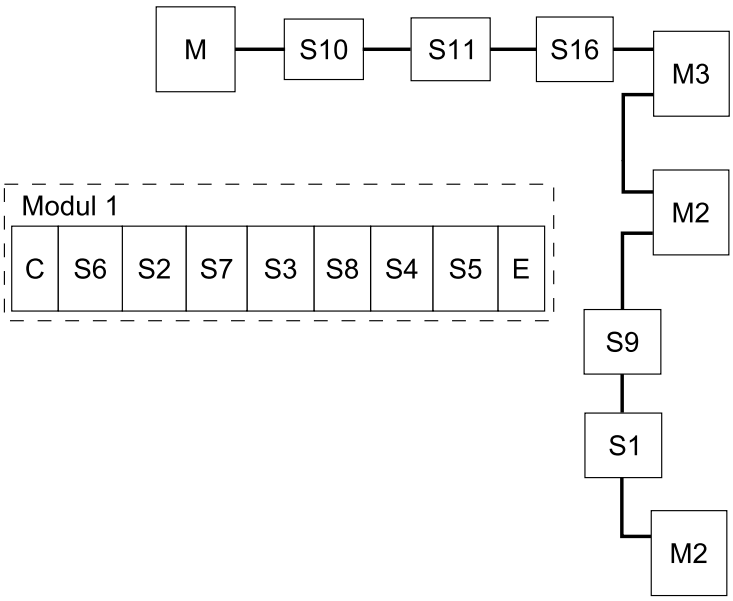
Slika 5.6: Primjer mrežne topologije

U prvom scenariju, dozvoljene su modifikacije topologije za sve uređaje u mreži uz ograničenje da struktura modularnih uređaja mora da bude sačuvana. Topologija dobijena nakon optimizacije za ovaj scenario je prikazan na slici 5.7.

Drugi scenario podrazumijeva promjenu topologije samo unutar modularnog uređaja, dok se struktura ostatka mreže ne smije mijenjati. Ovo je čest slučaj u sistemima automatizacije, gdje je fizička lokacija uređaja diktirana spoljnim faktorima koji zavise od samog procesa proizvodnje. Očekivano, rezultat optimizacije za opisani scenario daje nešto lošije rezutate, ali je osnovna topologija mreže sa slike u potpunosti sačuvana (fizička pozicija uređaja se mijenja samo za

Tabela 5.1: Informacije o korišćenim uređajima

Uređaj	**Naziv**	**Ibits**	**Obits**
S1	AX2000-B110	64	48
S2-S5, S12-S15	EL1004	4	-
S6-S8	EL2004	-	4
S9	EL9800-SPI	256	128
S10, S11	AX5101	48	48
S16	AX5203	96	96
S17	FB1111 Dig. In/Out	16	16
S18-S20	EL4034	-	64
Ukupno:		560	588



Slika 5.7: Mrežna topologija nakon optimizacije

modul 1, kao što je prikazano na slici 5.7).

Rezultati dobijeni optimizacijom, za navedene scenarije, kao i uporedna analiza ovih rezultata sa rezultatima dobijenim korišćenjem komercijalno dostupnih konfiguracionih alata, data je u tabeli 5.2.

Iz tabele 5.2, vidljivo je da se, u odnosu na konfiguraciju dobijenu TwinCAT alatom, predloženim postupkom optimizacije postiže oko 32% kraće vrijeme slanja okvira, kada je dozvoljena promjena mrežne topologije, odnosno oko 21%, kada su promjene dozvoljene na nivou modularnih uređaja. U slučaju alata KPA Studio EtherCAT, rezultati su još bolji (preko 45% kraće vrijeme slanja okvira za oba slučaja).

Osnovni nedostatak prethodno opisanog algoritma je u tome što on zahtijeva modifikaciju fizičke topologije mreže, što nije moguće postići u određenim slučajevima, jer je fizička struktura mreže određena spoljnim faktorima. Ovaj problem se može ublažiti modifikacijom algoritma na način da se algoritam primjenjuje bez modifikacije topologije, ili da se promjena topologije ograniči na određenu grupu uređaja koji se nalaze fizički blizu jedni drugima (npr. u istom upravljačkom ormaru), tako da se mogu lako preraspodijeliti ukoliko je to potrebno. Osim toga,

Tabela 5.2: Poređenje rezultata optimizacije

Konfiguracioni alat	L_F **(B)**	T_F **(μs)**
Algoritam primjenjen na cijelu mrežu	126	10.08
Algoritam primjenjen samo na module	146	11.68
TwinCAT System Manager	185	14.80
KPA EtherCAT Studio	273	21.84

Algoritam 5.2: Modifikovani algoritam za optimizaciju veličine okvira

```
1:  for each group ∈ groupList do
2:     for each domain ∈ domainList do
3:        LogAddrO = LogAddrI = domain.StartLogAddr
4:     end for
5:     for i = 1 → n = group.NumberOfSlave do
6:        Diff = LogAddrO − LogAddrI
7:        if Diff + group.Slave(i).Diff < 0 then
8:           Pronađi i < k ≤ n za koje je Diff + group.Slave(k).Diff ≥ 0
9:           if k nije pronađeno then
10:             Pronađi i < k ≤ n za koje je min(|group.Slave(k).Diff|)
11:          end if
12:          Zamijeni group.Slave(i) i group.Slave(k)
13:       end if
14:    end for
15:    LogAddrI = LogAddrI + Slave(i).Ibits
16:    if Diff < 0 then
17:       LogAddrO = LogAddrI + Slave(i).Obits
18:    else
19:       LogAddrO = LogAddrO + Slave(i).Obits
20:    end if
21: end for
```

grupisanje podataka svih uređaja u jedan ECAT telegram negativno utiče na dijagnostiku mreže, jer pogrešan rad samo jednog uređaja u mreži uzrokuje zaustavljanje čitavog sistema zato što vrijednost WKC polja telegrama nije ispravna. Poboljšanje dijagnostike u mreži, može se postići raspoređivanjem uređaja u logički izolovane grupe koje se nazivaju *domeni* (neki proizvođači koriste i termin *SyncUnit*).

U skladu sa prethodno rečenim, opisani algoritam je proširen tako da podrži organizaciju uređaja u grupe i domene. Na ovaj način, promjena fizičke pozicije uređaja se ograničava na određenu grupu (npr. upravljački ormar ili modularni uređaj), dok broj domena odgovara broju LRW telegrama u okviru kojih se razmjenjuju procesni podaci uređaja iz istog domena. Modifikovana verzija predloženog algoritma, koja uključuje prethodno opisane funkcionalnosti, prikazana je pseudokodom algoritma 5.2, pri čemu je izostavljen dio za određivanje FMMU parametara. Ovdje treba napomenuti da modifikovani algoritam, u slučaju jedne grupe i jednog domena, u potpunosti odgovara algoritmu 5.1.

Prije optimizacije, podređeni uređaji se organizuju u grupe i domene. Ovaj proces korisnik obavlja ručno u okviru specifično razvijenog konfiguracionog alata koji implementira algoritam za optimizaciju okvira. Za svaki domen, korisnik mora da definiše početnu logičku adresu ($domain.StartLogAddr$) koja se koristi za inicijalizaciju logičkih adresa ulaza i izlaza. Kao što je već pomenuto, svaka grupa sadrži uređaje koji su smješteni u isti kontrolni ormar, što znači

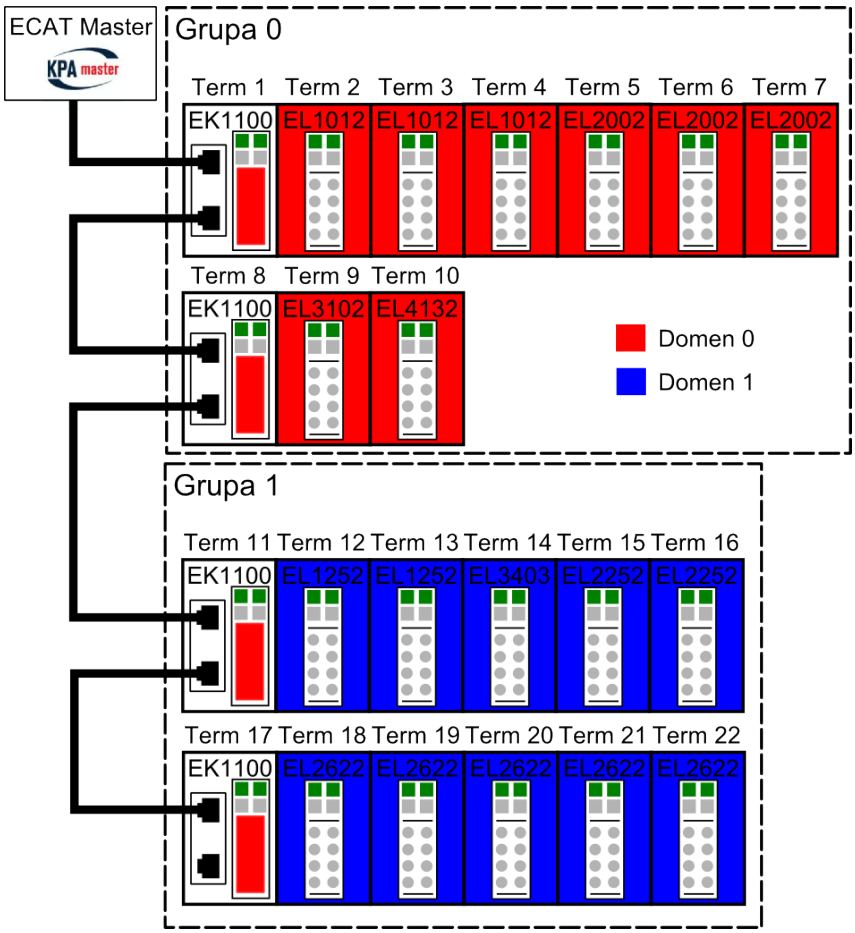
da je, unutar grupe, lako promijeniti redoslijed povezivanja uređaja, ukoliko je potrebno.

Domen predstavlja skup podređenih uređaja čiji se procesni podaci razmjenjuju na konzistentan i sinhronizovan način, tj. svi uređaji pripadaju istoj logičkoj grupi, koja dijeli zajedničko WKC polje unutar telegrama. Ovaj pristup značajno unapređuje dijagnostiku grešaka u sistemu i omogućava izolaciju uređaja koji su u kvaru na nivou domena. U slučaju otkaza nekog uređaja, kompletan sadržaj telegrama, kojim uređaj razmjenjuje podatke, će biti označen kao neispravan (npr. tako što se detektuje neispravna vrijednost WKC polja u telegramu), što znači da će svi uređaji koji se nalaze u istom domenu prestati sa razmjenom podataka. S druge strane, nastali otkaz neće uticati na rad uređaja koji se nalaze u drugom domenu, jer WKC polje telegrama koji prenosi podatke ovih uređaja ima ispravnu vrijednost. Kada bi se koristio samo jedan domen, otkaz jednog uređaja bi zaustavio rad svih ostalih uređaja u mreži iako se oni nalaze u ispravnom stanju. Grupisanjem uređaja u domene postiže se da samo uređaji koji obavljaju međusobno zavisne zadatke (pa prema tome pripadaju istom domenu) prestaju sa radom u slučaju greške, dok ostali uređaji mogu nesmetano da nastave sa radom.

Modifikovani algoritam je testiran za dva različita scenarija na stvarnoj mreži sa slike 5.8. Detaljnije informacije o korišćenim uređajima, date su u tabeli 5.3. Prvi scenario podrazumijeva slučaj kada se svi uređaji nalaze u istoj grupi i dijele isti domen, dok je u drugom scenariju mreža podijeljena u dvije grupe i dva domena, tako da se uređaji iz iste grupe nalaze u istom domenu, kao što je ilustrovano na slici 5.8. Kao nadređeni uređaj korišćena je evaluaciona verzija *KPA EtherCAT Master* softverske implementacije za standardni PC računar sa *Windows XP* operativnim sistemom, koja je uključena u *KPA Studio EtherCAT* softverski paket, dok je za generisanje inicijalnog ENI fajla korišćen *TwinCAT System Manager*, kao i u slučaju prvog algoritma. Važno je napomenuti da navedena softverska realizacija nadređenog uređaja ne uključuje ekstenziju za rad u realnom vremenu pod *Windows XP* operativnim sistemom, kao u

Tabela 5.3: Informacije o korišćenim uređajima

Uređaj	**Naziv**	**Ibits**	**Obits**
Term 2-4	EL1012	2	-
Term 5-7	EL2002	-	2
Term 9	EL3102	64	-
Term 10	EL4132	-	32
Term 12-13	EL1252	280	-
Term 14	EL3403	496	24
Term 15-16	EL2252	-	80
Term 18-22	EL2622	-	2
Ukupno:		1126	232



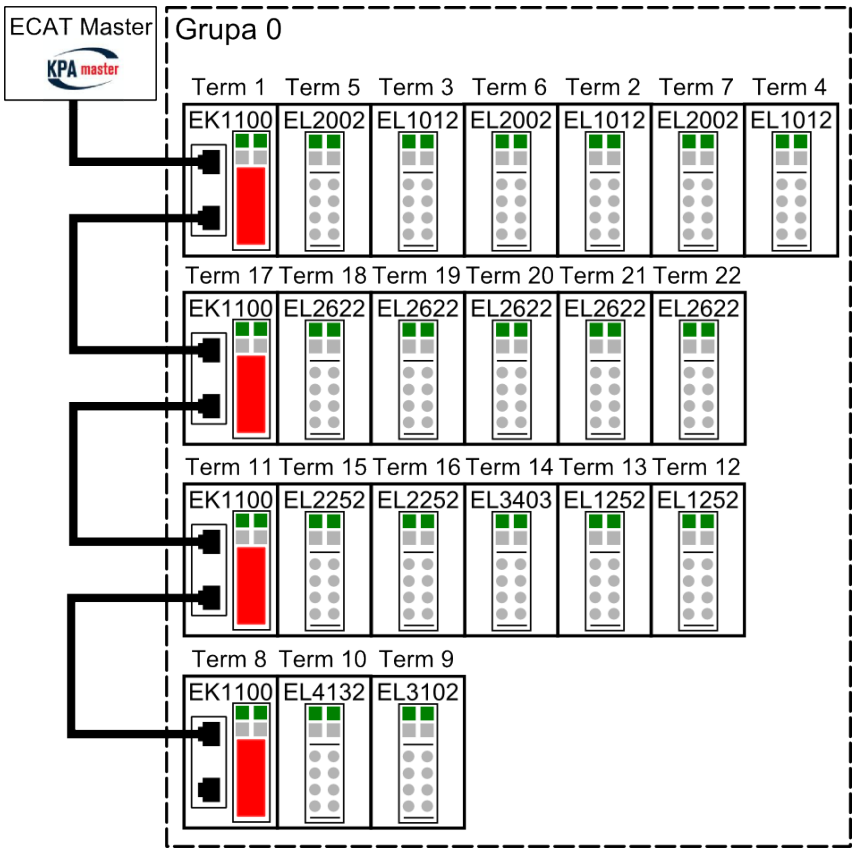
Slika 5.8: Mrežna postavka korišćena pri testiranju

slučaju *TwinCAT* sotvera, ali omogućava učitavanje ENI fajla modifikovanog nakon optimizacije, čime se omogućava testiranje optimizovane strukture cikličnih okvira u stvarnim uslovima.

Mrežne topologije, za dva scenarija, dobijene nakon optimizacije, prikazane su na slikama 5.9 i 5.10. Dio komunikacije, u okviru *Wireshark* softverskog alata za hvatanje i analizu mrežnog saobraćaja, sa označenim ulaznim podacima dobijenim od digitalnih ulaznih terminala (EL1012), analognog ulaznog terminala (EL3102) i terminala za mjerenje snage (EL3403), za prvi i drugi scenario, prikazan je na slikama 5.11 i 5.12, redom.

Rezultati dobijeni nakon optimizacije, upoređeni su sa rezultatima dobijenim pomoću prethodno pomenutih komercijalnih konfiguracionih alata. Detalji ovog poređenja, dati su na slici 5.13. Sa slike se vidi da, u slučaju prvog scenarija, modifikovani algoritam postiže 14%, odnosno 20%, bolje rezultate u poređenju sa *KPA Studio* i *TwinCAT* konfiguracionim alatom, redom. Kod drugog scenarija, veličina okvira je za 23% kraća u odnosu na konfiguraciju dobijenu *TwinCAT* alatom. Rezultati poređenja sa konfiguracionim alatom *KPA Studio* su izostavljeni sa slike 5.13, jer ovaj alat ne podržava koncept organizacije uređaja u domene.

U cilju procjene efikasnosti predloženog algoritma za optimizaciju veličine okvira, za kompleksniju mrežnu konfiguraciju, dizajnirana je mreža korišćenjem *TwinCAT System Manager*



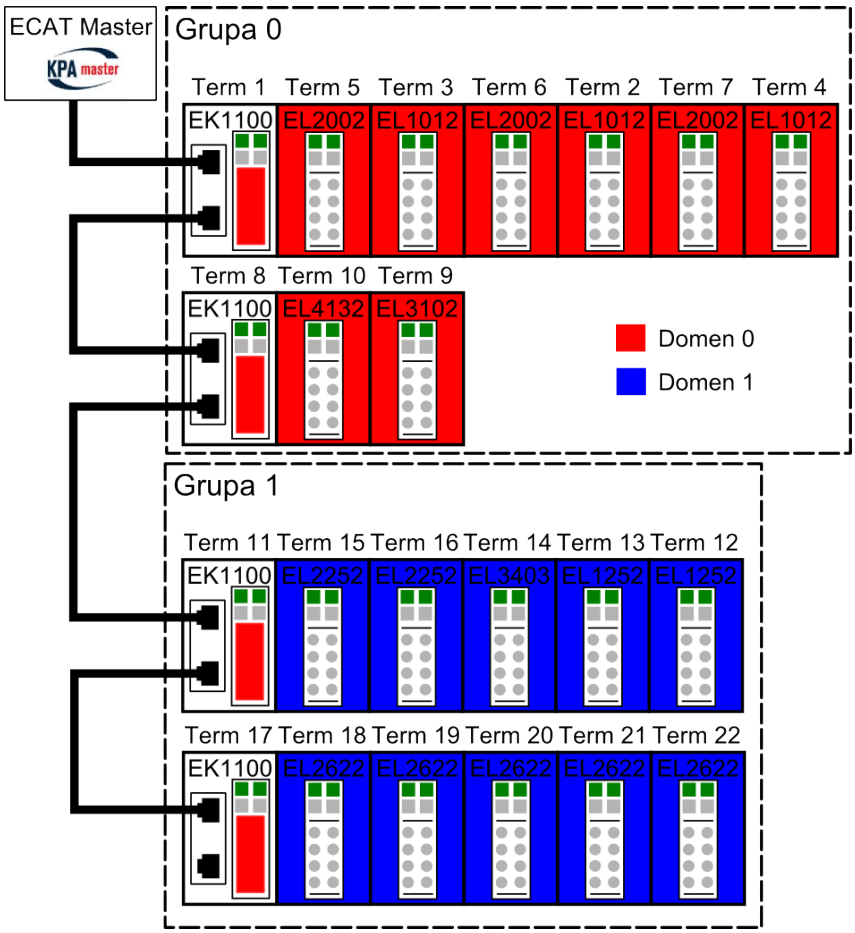
Slika 5.9: Mrežna topologija dobijena nakon optimizacije za prvi scenario

konfiguracionog alata.

Analizirana mreža se sastoji iz 5 upravljačkih ormara od kojih svaki sadrži jedan modularni uređaj, čiju strukturu čini 8 digitalnih ulaznih terminala sa procesnim podacima veličine 2 bita, 8 digitalnih izlaznih terminala sa procesnim podacima veličine 2 bita, 4 analogna ulazna terminala sa 2 kanala i 4 analogna izlazna terminala sa 2 kanala, kao i 10 individualnih servo osa. Na taj način, dobijena je mreža koja ima ukupno 175 podređenih uređaja (uključujući sprežnike magistrale). Takođe, uvedena je pretpostavka da uređaji unutar svakog upravljačkog ormara čine jednu grupu i pripadaju istom domenu (tj. postoji ukupno 5 domena).

Kao rezultat, *TwinCAT System Manager* softver je generisao konfiguraciju kod koje je veličina okvira 880 bajtova. S druge strane, nakon optimizacije, dobijena je veličina okvira od 670 bajtova, što predstavlja poboljšanje od 24% u odnosu na prethodnu konfiguraciju.

Kompleksnost algoritma 5.1 zavisi od mogućnosti brzog pronalaženja uređaja koji zadovoljava uslov da je razlika između logičkih adresa ulaza i izlaza manja ili jednaka nuli. U najgorem slučaju, kada se pretražuju svi uređaji, algoritam ima $O(n^2)$ kompleksnost u vremenu. Međutim, ako se uređaji organizuju u binarnu *heap* strukturu, koja je uređena u skladu sa vrijednošću $Slave.Diff$, kompleksnost se može svesti na $O(n \log n)$. Takođe, uređaji se mogu sortirati po $Slave.Diff$, pri čemu se opisani algoritam može direktno primjeniti na sortirani niz. U ovom



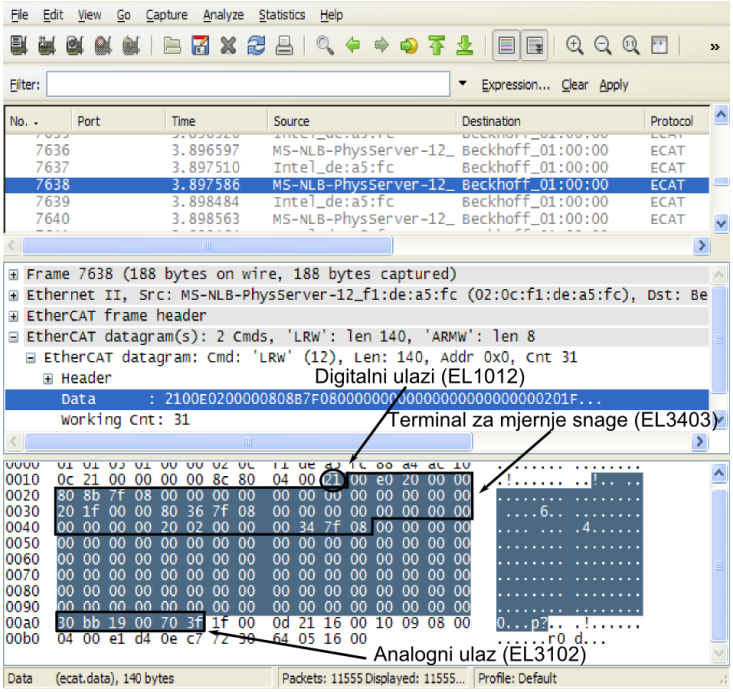
Slika 5.10: Mrežna topologija dobijena nakon optimizacije za drugi scenario

slučaju, kompleksnost algoritma se svodi na kompleksnost primjenjenog algoritma za sortiranje.

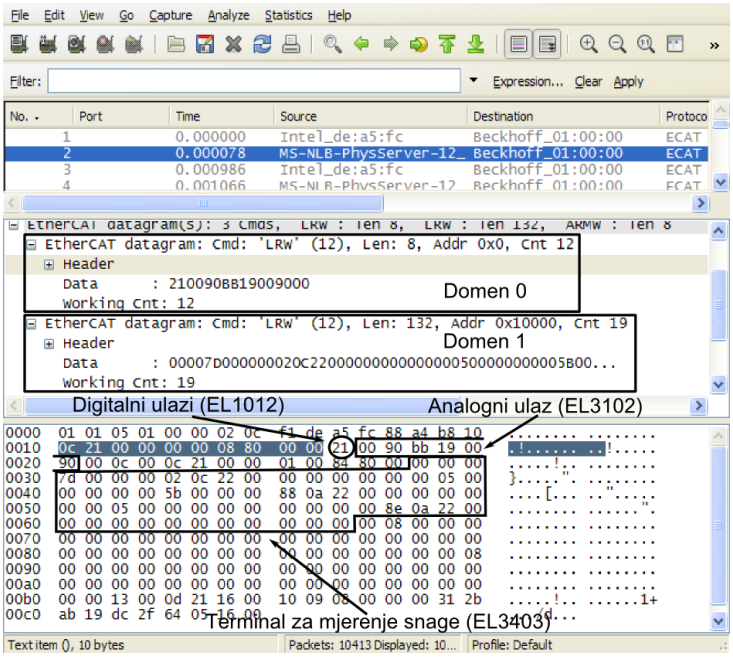
Optimalnost algoritma se može dokazati matematičkom indukcijom. Ako se pretpostavi da je rezultat dobijen predloženim algoritmom $A = \{a_1, \ldots, a_i\}$, a da je $O = \{o_1, \ldots, o_j\}$ optimalno rješenje, kod kojeg je minimizovana dužina okvira, tj. ne postoji prazan prostor kod uzastopnih logičkih adresa izlaza između dva susjedna uređaja. U cilju poređenja rezultata, definiše se mjera $f(\cdot)$, koja, za parcijalno rješenje i-te iteracije algoritma, definiše ukupnu veličinu neiskorišćenog prostora u rješenju. Cilj je da se dokaže da je za svako $r \leq k$ zadovoljeno $f(a_1, \ldots, a_r) \leq f(o_1, \ldots, o_r)$.

Inicijalno, za $r = 1$, logičke adrese ulaza i izlaza su poravnate, pa predloženi algoritam bira uređaj kod kojeg je zadovoljeno $Slave.Diff \leq 0$ ili kod kojeg je ova razlika minimalna. Prema tome, veličina neiskorišćenog prostora je minimizovana, što znači da je zadovoljeno $f(a_1) \leq f(o_1)$. U cilju dokazivanja optimalnosti algoritma za $r > 1$, uvodi se pretpostavka da $A_r = \{a_1, \ldots, a_r\}$ nije optimalno rješenje. Ovo znači da je odabrano parcijalno rješenje kod kojeg veličina neiskorišćenog prostora nije minimizovana.

Na slici 5.14 prikazan je proces izbora dva moguća rješenja u i-toj iteraciji algoritma za slučaj kada je $Diff > 0$ i kada je $Diff < 0$. Kao što može da se vidi sa slike 5.14b, izbor

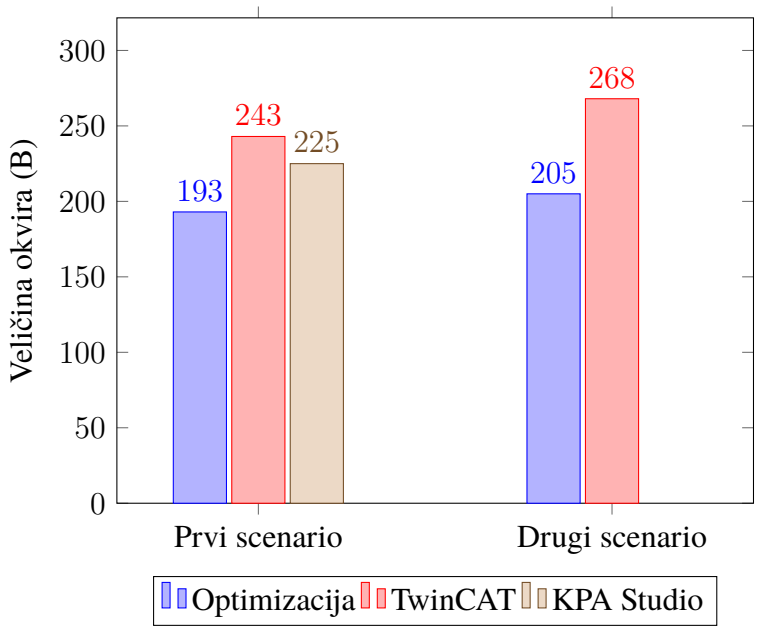


Slika 5.11: Prikaz komunikacije u okviru *Wireshark* softvera sa označenim ulaznim podacima za prvi scenario



Slika 5.12: Prikaz komunikacije u okviru *Wireshark* softvera sa označenim ulaznim podacima za drugi scenario

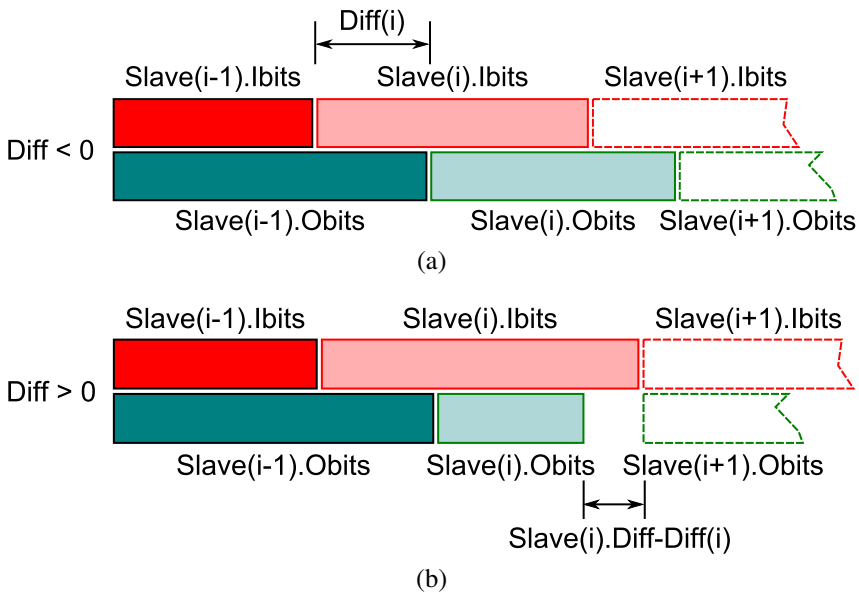
rješenja za koje vrijedi $Diff > 0$ dovodi do pojave neiskorišćenog prostora u logičkom prostoru izlaza čija je veličina određena sa $Slave(i).Diff - Diff(i)$. Prema tome, veličina neiskorišćenog prostora se minimizuje kada se odabere uređaj sa minimalnom vrijednošću $Slave(i).Diff$. To znači da algoritam uvijek bira rješenje koje minimizuje veličinu neiskorišćenog prostora, što je u kontradikciji sa prethodno uvedenom pretpostavkom.



Slika 5.13: Veličina okvira (u bajtovima) dobijena različitim konfiguracionim alatima

Konačno, može da se zaključi da $f(a_1, \ldots, a_r) \leq f(o_1, \ldots, o_r)$ vrijedi za svako $r \leq k$, čime je dokazana optimalnost predloženog algoritma.

Iako prethodno opisani algoritmi omogućavaju optimalnu reorganizaciju ECAT okvira, postoje određeni nedostaci u njihovoj primjeni. Kao prvo, da bi opisani koncept bio primjenjiv u stvarnim uslovima, svi podređeni uređaji u mreži moraju da podržavaju LRW servis. Ovo ne predstavlja ozbiljan problem, jer većina komercijalno dostupnih implementacija zadovoljava ovaj uslov. S druge strane, važno je napomenuti da postoji određeni broj proizvođača čiji uređaji ne podržavaju pomenuti servis. Drugo, algoritam treba da se primjeni u ranoj fazi projektova-



Slika 5.14: Ilustracija izbora sljedećeg uređaja za (a) $Diff < 0$ i (b) $Diff > 0$

nja sistema kako bi se izbjegli troškovi ponovnog ožičavanja realizovanog sistema. Konačno, postoje određeni scenariji (npr. kada se koriste samo ulazni moduli) koji mogu drastično da smanje efikasnost rezultata algoritma.

5.2 Optimizacija EPL protokola

Osnovni nedostatak EPL protokola ogleda se u proceduri prozivanja čvorova koja kao rezultat ima veoma slabo iskorišćenje propusnog opsega, kao što je pokazano u poglavlju 4. Osim toga, EPL koristi koncentratore koji rade u poludupleksnoj komunikaciji, što onemogućava paralelizaciju razmjene podataka u oba smjera.

Jedan od načina za prevazilaženje ovog problema podrazumijeva prelazak na mrežne komutatore uz istovremeno povećanje bitske brzine na 1 Gb/s. Jedno takvo rješenje, opisano je u [94]. Međutim, standardni mrežni komutatori unose značajno kašnjenje sa odstupanjem koje se ne može zanemariti. Osim toga, prethodno pomenuti prijedlog zahtijeva značajne izmjene trenutne verzije EPL protokola, što dovodi u pitanje očuvanje kompatibilnosti sistema.

Drugo rješenje, predloženo u ovom radu, koristi ulančavanje PRes okvira (PRC način rada), čime se izbjegava prozivanje čvorova, što povećava iskorišćenje propusnog opsega i poboljšanje efikasnosti komunikacije. Originalno rješenje, predloženo u [70], definiše vremenski kontrolisano ulančavanje PRes okvira. To znači da svaki čvor, nakon prijema PResMN okvira od upravljanog uređaja, čeka da istekne predefinisani interval prije nego što pošalje sopstveni PRes okvir. Ovaj pristup je zbog toga nazvan TT-PRC (engl. *Time-Triggered PollResponse Chaining*).

EPL mreža koja radi u TT-PRC režimu pruža mnoge prednosti u odnosu na mrežu koja radi u standardnom režimu rada, od kojih su najvažnije: kraće vrijeme komunikacionog ciklusa, jer se izostavlja slanje PReq okvira, brže vrijeme reakcije, jer se svi izlazni podaci istovremeno dostavljaju unutar jednog (PResMN) okvira i veće mogućnosti smanjenja potrošnje prilikom primjene strategija opisanih u [57, 58, 95], s obzirom da se neaktivno stanje MN čvora u okviru izohrone faze može značajno produžiti. Međutim, kao što je objašnjeno u odjeljku 2.2.4, ovaj mehanizam zahtijeva da svi ulančani čvorovi razmjenjuju vremenski kritične poruke u okviru rezervisanog PRC slota. S druge strane, iz jednačine (4.41), zaključuje se da se optimalne performanse u sistemu postižu ako redoslijed slanja PRes okvira odgovara fizičkom rasporedu ulančanih čvorova, jer se na taj način dobija najmanje vrijeme propagacije između dva čvora.

Prethodno zaključivanje, analitički je potvrđeno u odjeljku 4.4, za slučaj linijske topologije. U opštem slučaju, optimalan redoslijed slanja PRes okvira može se odrediti jednostavnim heurističkim algoritmom kod kojeg je kriterijum selekcije sljedećeg ulančanog čvora određen

minimalnim propagacionim kašnjenjem. U kompleksnijim mrežnim strukturama, ovaj uslov obično ne može da bude zadovoljen, ako se zahtijeva da svi ulančani čvorovi razmjenjuju podatke u istom vremenskom slotu, što dovodi do određenog smanjenja performansi sistema. Osim toga, neki EPL mehanizmi, kao što su multipleksirani način rada i različite opcije redundanse (npr. redundantni MN čvor), nisu direktno podržani standardom. Konačno, TT-PRC mehanizam je osjetljiv na poremećaje koji mogu da uzrokuju neželjene kolizije PRes okvira, zbog nesavršenosti korišćenih mrežnih komponenata na koje utiču različiti faktori okruženja (npr. temperatura, vibracije, starenje komponenata, itd.).

Pomenute nesavršenosti određuju varijaciju kašnjenja korišćenih komponenata koja može da se promijeni tokom vremena. S druge strane, TT-PRC mehanizam predviđa mjerenje propagacionih kašnjenja samo prilikom inicijalizacije sistema, što znači da on ne obezjeđuje mehanizme za detekciju eventualne promjene propagacionih kašnjenja tokom rada sistema. Takođe, važno je napomenuti da u određenim aplikacijama koje su orijentisane ka obezbjeđenju visokog nivoa pouzdanosti, poruke mogu da se šalju različitim putanjama u različitim trenucima, što predstavlja dodatni izazov prilikom korišćenja TT-PRC pristupa.

5.2.1 Ulančavanje PRes okvira kontrolisano događajima

U cilju otklanjanja pomenutih nedostataka TT-PRC mehanizma, u ovoj disertaciji, predlaže se modifikacija originalnog pristupa na način da se slanje PRes okvira inicira događajem pa je predloženi mehanizam nazvan *ulančavanje PRes okvira kontrolisano događajima*, ET-PRC (engl. *Event-Triggered PollResponse Chaining*). Kao i u slučaju TT-PRC prsitupa, PResMN okvir, koji se šalje odmah nakon SoC okvira, sadrži izlazne podatke koji se dostavljaju ulančanim čvorovima. Naredni čvorovi šalju sopstvene PRes okvire po prijemu PResMN okvira ili PRes okvira koje su poslali prethodni čvorovi. Da bi se ovo postiglo, svaki čvor, koji podržava ET-PRC način rada, ima posebno definisan parametar *PrevNodeID* koji se inicijalizuje prilikom konfiguracije sistema. Kada neki čvor primi PRes okvir, on poredi izvornu EPL adresu iz primljenog okvira (polje EPLSRC) sa konfigurisanom vrijednošću *PrevNodeID* parametra i odgovara slanjem sopstvenog PRes okvira, samo ako se ove dvije vrijednosti poklapaju. Poređenje može da se realizuje softverski ili, ako se zahtijevaju maksimalne performanse, hardverski korišćenjem prethodno pomenutog *auto-response* mehanizma. Mjerenje propagacionih kašnjenja više nije potrebno, što pojednostavljuje proces inicijalizacije sistema i eliminiše probleme koji nastaju kao posljedica nesavršenosti vremenskih parametara korišćenih mrežnih komponenata.

Optimalne performanse se postižu ako vrijednost parametra *PrevNodeID* odgovara EPL

adresi uređaja (*NodeID*) koji ima najmanje propagaciono kašnjenje u odnosu na čvor koji se konfiguriše. U slučaju linijske topologije, optimalna konfiguracija ET-PRC čvorova je veoma jednostavna i može se opisati sa $PrevNodeID^{i+1} = NodeID^{i}$.

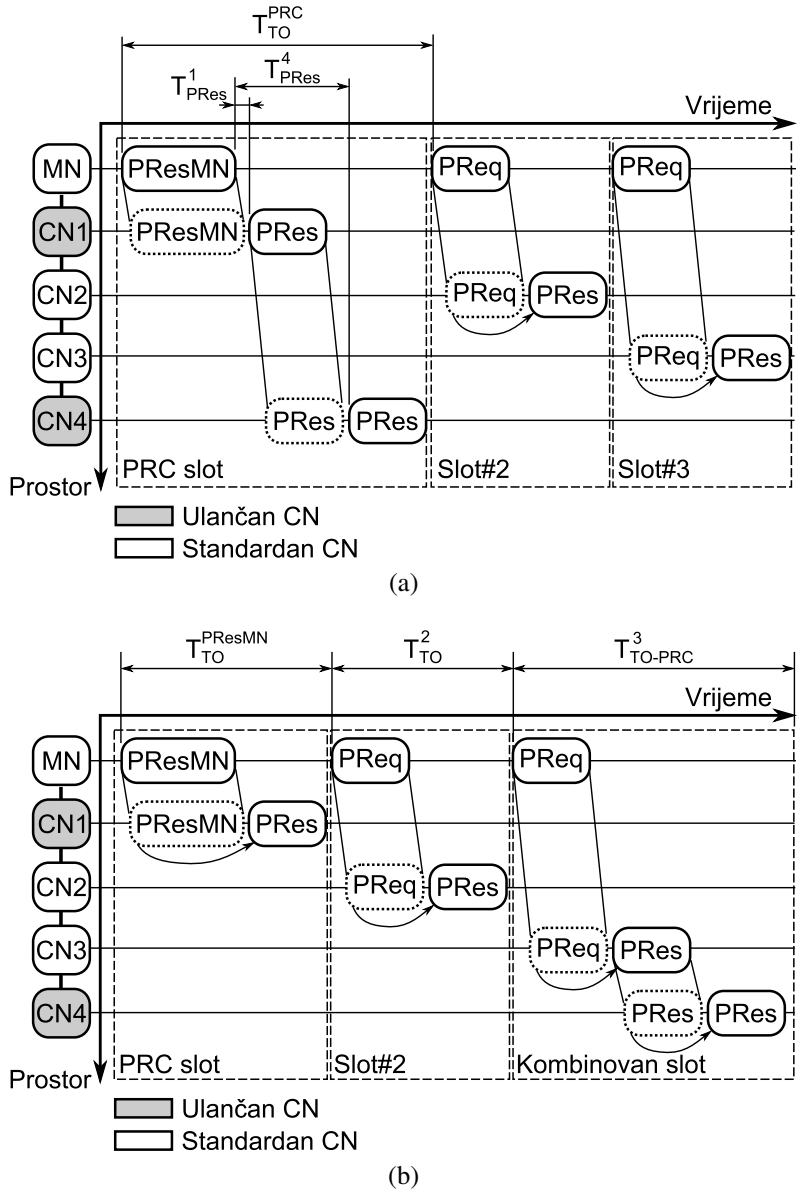
Za razliku od TT-PRC pristupa, ET-PRC mehanizam omogućava kombinovanje standardnih i ulančanih čvorova u okviru istog vremenskog slota, koji se naziva *kombinovani slot*, čime se dobija više prostora za optimizaciju dužine trajanja izohrone faze u slučaju komplesknijih mrežnih konfiguracija. Pojednostavljeni prostorno-vremenski dijagrami, prikazani na slici 5.15, ilustruju prethodno pomenuti koncept na primjeru mreže koja se sastoji iz četiri upravljana čvora (dva standardna i dva ulančana čvora) povezana linijskom topologijom.

U slučaju TT-PRC mehanizma, izohrona faza se sastoji iz jednog PRC slota i dva standardna vremenska slota (*Slot#2* i *Slot#3*) rezervisana za standardne čvorove CN2 i CN3, kao što je prikazano na slici 5.15a. Prilikom izračunavanja vremena odziva za čvor CN4 (T^{4}_{PRes}) u obzir se mora uzeti kašnjenje PResMN okvira kroz čvorove CN2 i CN3, što znači da slanje ulančanih PRes okvira nije optimalno. S druge strane, kod ET-PRC načina rada, čvorovi CN1 i CN4 se konfigurišu tako da bude $PrevNodeID^{1} = 240$ i $PrevNodeID^{4} = 3$, tj. CN1 šalje sopstveni PRes okvir po prijemu PResMN okvira, dok se CN4 aktivira po prijemu PRes okvira čvora CN3.

Čvorove CN2 i CN3 eksplicitno proziva MN čvor slanjem odgovarajućih PReq okvira. Kao što može da se vidi sa slike 5.15b, komunikacija u ovom načinu rada, takođe, podijeljena je u tri vremenska slota.

Prvi slot ima istu strukturu kao PRC slot u TT-PRC režimu. Jedina razlika je u tome što se u ovom slotu šalju samo PResMN i PRes čvora CN2 za razmatrani primjer. Drugi slot (*Slot#2*) je standardni slot rezervisan za čvor CN2. Konačno, posljednji slot je kombinovani slot u kojem se PRes okvirom čvora CN3, takođe, inicira slanje PRes okvira čvora CN4. Kombinovani slot se može smatrati specijalnom vrstom PRC slota. Jedina razlika je u tome što se prvi ulančani čvor u slotu aktivira PRes okvirom standardnog čvora umjesto PResMN okvirom.

Dužina trajanja kombinovanog slota (npr. T^{3}_{TO-PRC} na slici 5.15b), može se odrediti korišćenjem jednačine (4.34), s tim da je potrebno zamijeniti $T_{F(PResMN)}$ sa $T_{F(PReq)} + T_{F(PRes)}$ odgovarajućeg standardnog čvora. Prednost predloženog pristupa ogleda se u tome što on omogućava postizanje najmanjeg vremena propagacije između susjednih čvorova u PRC slotu i u kombinovanim slotovima. To se jasno može vidjeti na slici 5.15b, s obzirom da čvorovi CN3 i CN4 imaju manje propagaciono kašnjenje u odnosu na čvorove CN1 i CN4 u slučaju TT-PRC načina rada (slika 5.15a). Konačno, treba napomenuti da je analitički model predložen za TT-PRC režim rada u potpunosti primjenjiv i kod ET-PRC režima. Kao što je već objašnjeno, jedina razlika leži u tome što se kod ET-PRC mehanizma uvode kombinovani slotovi, čija se

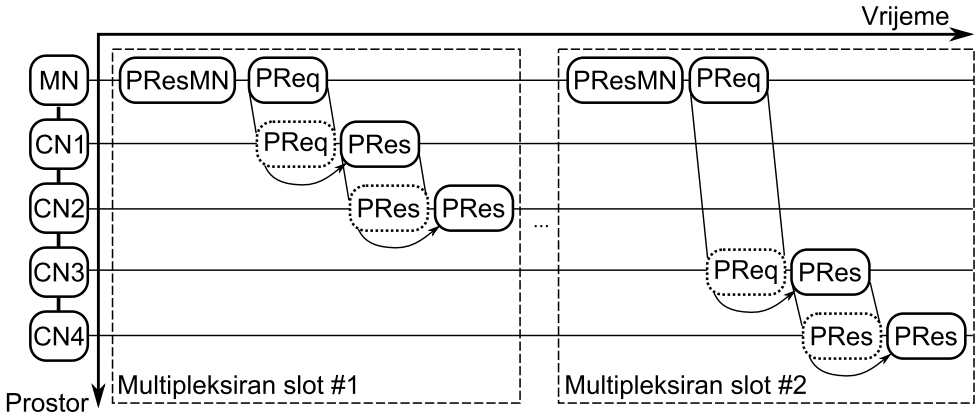


Slika 5.15: Primjer EPL komunikacije u (a) TT-PRC i (b) ET-PRC režimu

dužina trajanja može odrediti na prethodno opisani način.

Koncept kombinovanih slotova može se iskoristiti za omogućenje multipleksiranog načina rada, koji nije direktno podržan TT-PRC načinom rada. Kako bi se ovo postiglo, prvi čvor u multipleksiranom slotu mora da bude konfigurisan tako da radi u standardnom režimu rada. Naredni ulančani čvorovi se aktiviraju PRes okvirima prethodnih čvorova kao što je ilustrovano na slici 5.16.

Slika prikazuje pojednostavljeni prostorno-vremenski dijagram za mrežu sa četiri ulančana čvora povezana linijskom topologijom. Kako bi se omogućio multipleksiran način rada, CN1 i CN3 su konfigurisani tako da rade u standardnom režimu. Upravljački čvor proziva ova dva čvora u različitim komunikacionim ciklusima, kao i kod standardnog multipleksiranog načina rada. S druge strane, čvorovi CN2 i CN4 su podešeni da rade u ET-PRC režimu sa parametrima



Slika 5.16: Primjer EPL komunikacije u multipleksiranom ET-PRC režimu

$PrevNodeID^2 = 1$ i $PrevNodeID^4 = 3$, tj. CN2 se aktivira PRes okvirom čvora CN1 u prvom multipleksiranom slotu, dok se CN4 aktivira PRes okvirom čvora CN3 u drugom multipleksiranom slotu.

Radi kompletnosti razmatranja, važno je napomenuti da se multipleksirani način rada u TT-PRC režimu, iako nije direktno podržan standardom, može obezbijediti odgovarajućim podešavanjem $T_{PRes(conf.)}$ parametara svakog čvora. Međutim, kako bi se izbjegla situacija da više ulančanih čvorova istovremeno šalje PRes okvire, svaki upravljani čvor mora da prati vrijednost brojača ciklusa i da omogući PRC režim rada samo u multipleksiranim slotovima u kojim dati čvor smije da komunicira. Nasuprot tome, ET-PRC pristup ne zahtijeva implementaciju opisane funkcionalnosti u upravljanom čvoru pa se, prema tome, lakše integriše u standardni tok EPL konfiguracije.

Osim prethodno opisanih prednosti, važno je pomenuti da ET-PRC mehanizam zahtijeva samo manje softverske modifikacije postojeće implementacije EPL komunikacionog steka, što znači da može da se koristi sa bilo kojim standardnim mrežnim interfejsom. Međutim, vrijeme odziva upravljanog čvora u slučaju softverske implementacije ovog mehanizma može da bude ograničavajući faktor u najzahtjevnijim primjenama. Stoga, kako bi se postigle najveće performanse sistema uporedive sa TT-PRC pristupom, potrebno je koristiti specijalne hardverske komponente koje podržavaju *auto-response* mehanizam. Trenutno, postoji veliki broj različitih proizvođača koji nude ovakve komponente bilo u vidu predprojektovanih modula za implementaciju na FPGA platformi [96], ili u vidu rješenja integrisanih u okviru SoC (engl. *System on Chip*) čipa [97].

Očigledno, i ET-PRC ima određene nedostatke. Kao što je prethodno objašnjeno, upravljani čvor, koji radi u ET-PRC režimu, ne šalje nikakav odgovor dok ne primi PRes okvir sa predefinisanom izvornom EPL adresom. Ukoliko se neki od PRes okvira izgubi prilikom slanja, čvor koji se aktivira datim okvirom, kao i svi naredni ulančani čvorovi, neće poslati nikakve podatke,

što rezultuje gubitkom dijela komunikacionog ciklusa. U najlošijem slučaju (u slučaju gubitka PResMN okvira), kompletan komunikacioni ciklus može biti izgubljen.

Upravljački čvor može detektovati gubitak PRes okvira tokom komunikacije (npr. korišćenjem posebnog tajmera čija vrijednost odbrojavanja se podešava za svaki čvor) i preduzeti odgovarajuće mjere. Jedan način da se opisani problem prevaziđe, jeste da MN pošalje PRes okvir sa odgovarajućom izvornom EPL adresom, kako bi aktivirao naredni čvor, pri čemu se u ovom okviru podaci označavaju kao neispravni (postavljanjem odgovarajuće vrijednosti *Ready* bita u PRes okviru) kako bi se izbjegla interpretacija pogrešnih ulaznih podataka od strane drugih uređaja. Važno je napomenuti da će opisani pristup ispravno funkcionisati samo u slučaju otkaza nekog od čvorova, pri čemu je greška takva da čvor ne šalje nikakve okvire. Tranzijentne greške na liniji veze mogu da proizvedu neželjene kolizije, jer MN čvor može da zaključi da je PRes okvir izgubljen (na vezi koja je pogođena tranzijentnom greškom), a da taj okvir bude dostavljen sljedećem ulančanom čvoru (preko veze koja nije pogođena tranzijentnom greškom). Kako bi se ovaj problem otklonio, potrebno je koristiti druge pristupe (npr. kablovsku redundansu u mreži).

5.2.2 Eksperimentalna verifikacija

U cilju verifikacije funkcionalnosti predloženog pristupa, kao i ispravnosti analitičkog modela izvedenog u odjeljku 4.4, *openPOWERLINK V2.2.1* implementacija EPL komunikacionog steka je modifikovana tako da podrži prethodno opisani ET-PRC mehanizam. Nakon toga, obavljeno je više eksperimenata na prototipima mrežnih konfiguracija prikazanim na slici 4.22. Eksperimentalni rezultati su prikupljeni za slučajeve kada mreža radi u svakom od opisanih režima: standardnom, TT-PRC i ET-PRC. Ovdje treba napomenuti da je struktura mreže sa slike 4.22a modifikovana tako da sadrži dva OPLK čvora, jer *Softing* i *Hilscher* implementacije predstavljaju zatvorena rješenja kod kojih nije moguće modifikovati kôd tako da podržava ET-PRC način rada.

Kao što je već pomenuto, primarni kriterijum prilikom optimizacije vremena odziva predstavlja mogućnost minimizacije komunikacionog ciklusa. S obzirom da se dio komunikacionog ciklusa rezervisan za asinhronu komunikaciju uvijek može fiksirati, prostor za optimizaciju je potrebno potražiti u obezbjeđenju mehanizama koji omogućavaju minimizaciju izohrone faze u okviru EPL komunikacionog ciklusa. Prilikom mjerenja analizirana je samo dužina trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci (T_{poll} i T_{prc}), s obzirom da je dio koji se odnosi na sinhronizaciju (T_{sync}) isti u sva tri režima rada. Rezultati sprovedene eksperimentalne analize su sumirani u tabeli 5.4. Kvalitativni rezultati u vidu histograma za tri

Tabela 5.4: Pregled statistike za dužinu trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci (T_{poll} i T_{prc})

Način rada	Min.	Max.	μ	σ
Struktura mreže sa slike 4.22a				
Standardni	66.796	74.546	74.248	0.249
TT-PRC	25.566	25.716	25.632	0.024
ET-PRC	25.544	25.684	25.626	0.024
Struktura mreže sa slike 4.22b				
Standardni	64.716	71.116	70.837	0.218
TT-PRC	24.682	24.892	24.789	0.024
ET-PRC	24.728	24.858	24.798	0.03

Sve vrijednosti su izražene u μs

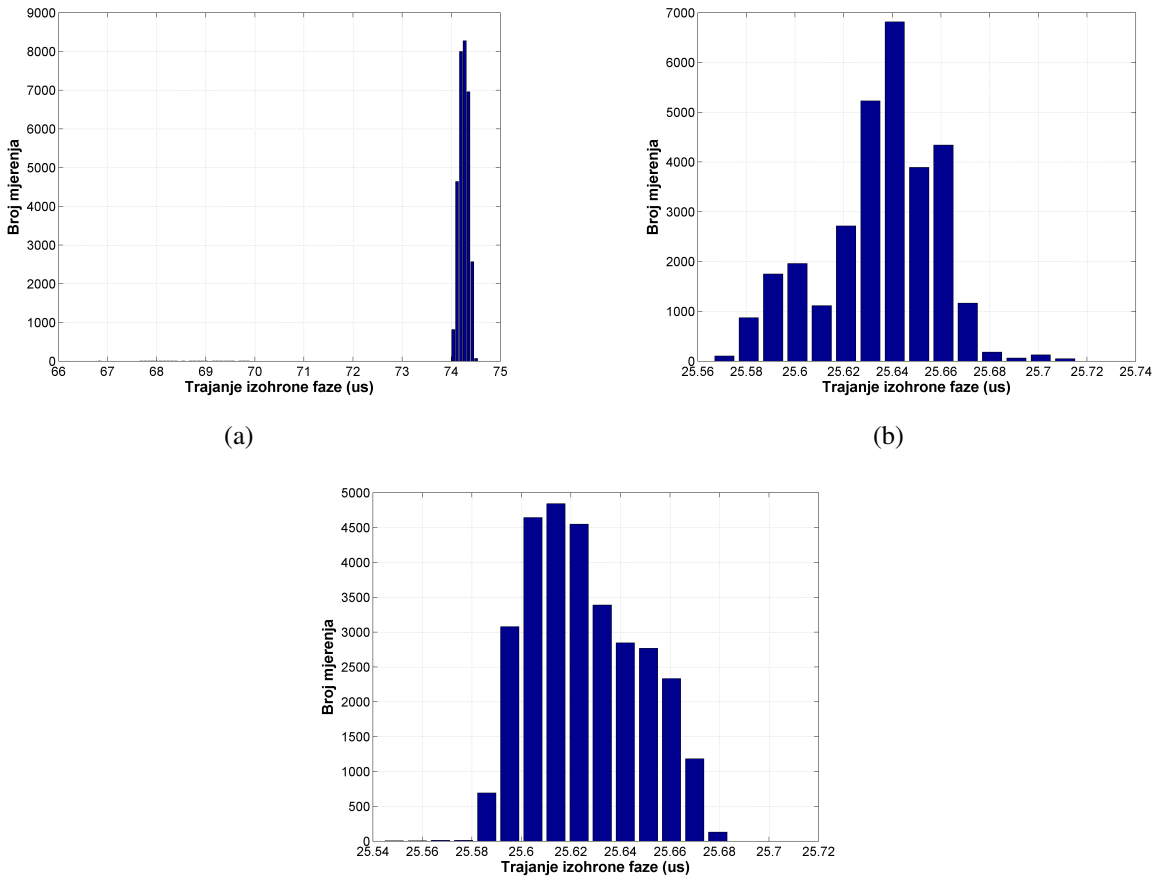
načina rada u slučaju linijske topologije su prikazani na slici 5.17. Kvalitativno slični rezultati, dobijeni su i u slučaju topologije zvijezde (slika 5.18).

Očekivano, mehanizam ulančavanja čvorova poboljšava efikasnost komunikacije u smislu dužine trajanja. Iz dobijenih eksperimentalnih rezultata, takođe, može se zaključiti da TT-PRC i ET-PRC mehanizmi obezbjeđuju približno isti nivo performansi. Naravno, ovo umnogome zavisi od vremena odziva čvorova koji rade u ET-PRC režimu. ET-PRC način rada može da obezbijedi isti nivo performansi, kao kod TT-PRC režima, samo ako se koriste čvorovi koji podržavaju *auto-response* mehanizam.

Kao što se može vidjeti na slikama 5.17a i 5.18a, histogrami dužine trajanja analiziranog dijela izohrone faze u potpunosti odgovaraju histogramima dobijenim prilikom karakterizacije T_{TO} parametara (slika 4.24). To znači da su performanse sistema koji radi u standardnom načinu rada dominantno određene mogućnostima MN čvora da precizno generiše ove vremenske intervale. S obzirom da je u eksperimentalnoj postavci korišćena PC implementacija MN čvora, performanse sistema su značajno degradirane u odnosu na teorijski dobijenu vrijednost. Ukoliko bi se koristila FPGA implementacija MN čvora, dužina trajanja analiziranog dijela izohrone faze može dodatno da se redukuje tako da bude blizu teorijskog minimuma, koji se može odrediti korišćenjem EPL analitičkog modela.

5.2.3 Simulaciona analiza

U cilju evaluacije predloženog ET-PRC mehanizma i njegovog poređenja sa drugim rješenjima, u slučaju složenijih mrežnih konfiguracija, u ovom radu je razvijen simulacioni model za OMNeT++ simulaciono okruženje koji podržava sva tri režima rada EPL porotokola analizirana u prethodnom odjeljku. Svi vremenski parametri mrežnih komponenata, analizirani u

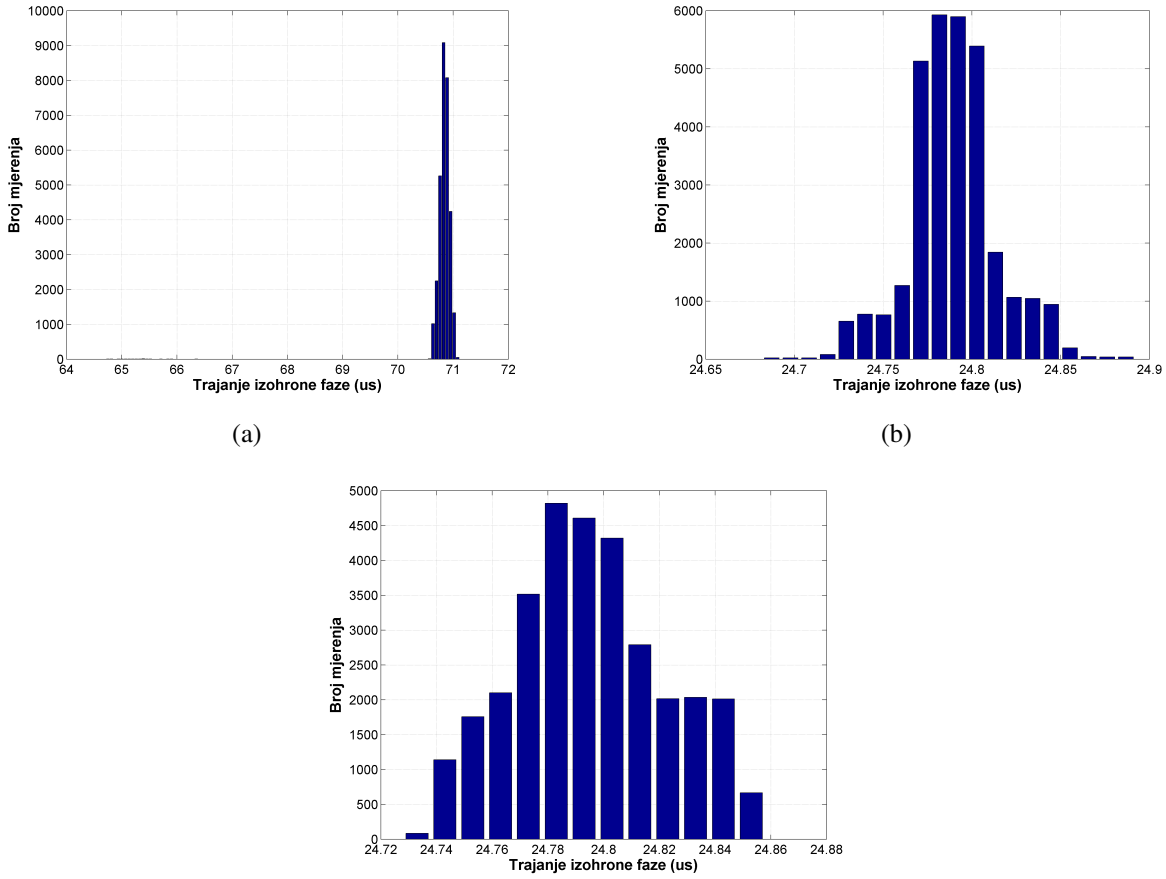


Slika 5.17: Izmjerena dužina trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci za linijsku topologiju sa dva upravljana čvora u (a) standardnom, (b) TT-PRC i (c) ET-PRC režimu

odjeljku 4.5, modelovani su normalnom raspodjelom $\mathcal{N}(\mu, \sigma)$, pri čemu su korišćene srednje vrijednosti (μ) i standardne devijacije (σ) iz tabele 4.5.

Simulacioni model je verifikovan tako što su rezultati dobijeni simulacijom mrežnih konfiguracija iz prethodnog odjeljka upoređeni sa eksperimentalnim rezultatima iz tabele 5.4. Osim toga, u cilju dodatne verifikacije, rezultati simulacije su upoređeni sa teorijskim vrijednostima izračunatim korišćenjem jednačina analitičkog modela, koji je izveden u odjeljku 4.4. Pri tome korišćene su srednje vrijednosti vremenskih parametara iz tabele 4.5 (uz pretpostavku da je $T_R^{MN} = 1\mu s$). Rezultati simulacija su prezentovani u tabeli 5.5.

Poređenjem rezultata simulacija sa eksperimentalnim rezultatima, može se zaključiti da je njihovo odstupanje manje od 1.5%, za sve posmatrane slučajeve. Takođe, očigledno je da teorijske vrijednosti, dobijene analitičkim putem, odgovaraju srednjim vrijednostima dobijenim mjerenjem. Jedini izuzetak predstavlja standardni način rada, kod kojeg se dobija značajna razlika u odnosu na teorijski izračunate vrijednosti, zbog nemogućnosti PC implementacije MN čvora da generiše precizne T_{TO} intervale.



Slika 5.18: Izmjerena dužina trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci za topologiju zvijezde sa dva upravljana čvora u (a) standardnom, (b) TT-PRC i (c) ET-PRC režimu

Simulacijom istih mrežnih konfiguracija, pri čemu su u modelu korišćeni vremenski parametri FPGA implementacije MN čvora, dobijeni su rezultati koji su u potpunosti usklađeni sa teorijski dobijenim vrijednostima, kao što može da se vidi u tabeli 5.5.

Kako bi se razmotrile mogućnosti predloženog ET-PRC mehanizma u cilju poboljšanja performansi sistema, obavljena je simulacija jedne složenije mrežne strukture koja kombinuje linijsku i topologiju zvijezde (što je čest slučaj u stvarnim instalacijama) na način da je 8 linijskih ogranaka povezano na centralni koncentrator, kao što je prikazano na slici 5.19.

Mrežni uređaji su modelovani vremenskim parametrima FPGA implementacije MN čvora. Svi upravljani čvorovi su modelovani parametrima OPLK implementacija, dok je centralni koncentrator (H) modelovan parametrima *3Com* koncentratora. Prilikom simulacije, usvojene su sljedeće pretpostavke: uređaji su povezani kablovima dužine 10 m, ulazno/izlazni procesni podaci svakog uređaja su veličine 4 bajta i trajanje EPL komunikacionog ciklusa je 5 ms. Trajanje svake simulacije definisano je tako da se obezbijedi najmanje 30000 mjerenja. Parametri modela, neophodni za različite režime rada, T_{TO} i T_{PRes}, procijenjeni su korišćenjem jednačina (4.33) i

Tabela 5.5: Rezultati simulacija

Način rada	μ	σ	Teorijski
Struktura mreže sa slike 4.22a			
Standardni (PC MN)	74.496	0.249	34.632
Standardni (FPGA MN)	34.869	0.025	
TT-PRC	25.799	0.024	25.639
ET-PRC	25.879	0.025	25.719
Struktura mreže sa slike 4.22b			
Standardni (PC MN)	71.093	0.219	32.122
Standardni (FPGA MN)	32.471	0.023	
TT-PRC	25.068	0.029	24.828
ET-PRC	25.148	0.29	24.908
Struktura mreže sa slike 5.19			
Standardni	2908.455	0.088	2908.44
TT-PRC	2498.14	0.082	2499.11
ET-PRC	2233.313	0.085	2233.304
Standardni (multipleksiran)	362.695	0.072	362.68
ET-PRC (multipleksiran)	283.998	0.069	283.988

Sve vrijednosti su izražene u μs

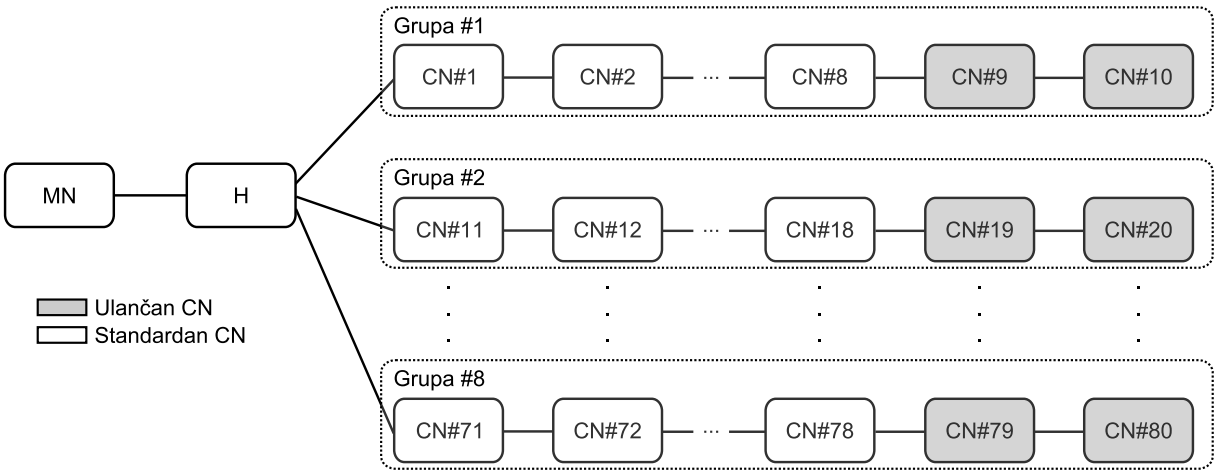
(4.37), redom.[1]

Rezultati simulacija su prezentovani u tabeli 5.5. Očigledno je da TT-PRC postiže bolje rezultate u odnosu na standardni način rada (14% kraća dužina trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci). Takođe, može se zaključiti da ET-PRC omogućava dodatnu optimizaciju dužine trajanja ovog intervala (oko 10%) u odnosu na TT-PRC. Ovo je posljedica korišćenja kombinovanih slotova, koji omogućavaju smanjenje ukupnog propagacionog kašnjenja između ulančanih čvorova. Rezultati dobijeni simulacijama su dodatno potvrđeni analitičkim putem, korišćenjem jednačina predloženog teorijskog modela.

Kako bi se analizirala mogućnost primjene multipleksiranih slotova u ET-PRC režimu, mrežna konfiguracija je promijenjena tako što su upravljani čvorovi organizovani u logičke grupe, kao što je ilustrovano na slici 5.19. Čvorovi iz iste grupe razmjenjuju vremenski kritične podatke u odvojenom EPL komunikacionom ciklusu u odnosu na čvorove koji pripadaju drugoj grupi (tj. linijski ogranci su multipleksirani faktorom 8).

Prethodno opisana mrežna konfiguracija, simulirana je za dva radna režima, standardni i ET-PRC. Rezultati simulacija prikazani su u tabeli 5.5. Kao što je očekivano, performanse u smislu dužine trajanja dijela izohrone faze u kojem se razmjenjuju vremenski kritični podaci, značajno se poboljšavaju ako se koristi multipleksirani način rada (ugrubo, poboljšanje odgovara faktoru

[1]Svi parametri modela kao i kompletan OMNeT++ izvorni kod su dostupni u okviru repozitorijuma https://bitbucket.org/mknezic/epl4inet.



Slika 5.19: Složenija EPL mrežna konfiguracija

multipleksiranja). Takođe, iz dobijenih rezultata se jasno može izvesti zaključak da ET-PRC omogućava dodatno poboljšanje performansi (oko 20%) u odnosu na standardni način rada.

5.3 Optimizacija PNIO protokola

Analizom teorijskih rezultata dobijenih u odjeljku 4.9, može se identifikovati nekoliko karakterističnih segmenata koji znatno utiču na performanse koje se mogu postići u PNIO baziranom industrijskom sistemu.

Prvi efekat ogleda se u uticaju veličine podataka na iskorišćenost propusnog opsega, što ujedno pogoršava performanse u smislu minimalnog vremena komunikacionog ciklusa. Naime, sa slike 4.38, jasno se vidi da opadajući trend dužine trajanja IRT faze, pri smanjenju veličine procesnih podataka, ulazi u oblast zasićenja za vrijednosti ispod 36 bajtova, što znači da je u ovoj oblasti $T_{C(min)}$ konstantno, bez obzira na veličinu podataka i korišćenu topologiju. S obzirom da se veličina procesnih podataka u sistemima automatizacije tipično kreće u opsegu od nekoliko bajtova, jasno je da mogućnost ublažavanja opisanog efekta može da dovede do značajnog poboljšanja performansi.

Jedno takvo rješenje, poznato pod nazivom *dinamičko pakovanje okvira*, DFP (engl. *Dynamic Frame Packing*), predloženo je u [98] i eksperimentalno verifikovano u [35]. Međutim, sinhronizacija uređaja predstavlja kritičan faktor koji ograničava mogućnost primjene predloženog rješenja u stvarnim uslovima, kao što je pokazano u [99]. Pažljivim planiranjem redoslijeda slanja poruka, u zavisnosti od korišćene topologije, ovaj efekat se može redukovati, kao što je opisano u [40, 41]. Modifikacijom osnovnog DFP koncepta, autori su u [42] predložili *mehanizam za automatsko pakovanje okvira*, APM (engl. *Automatic Packing Mechanism*), kojim se prevazilazi potreba za specifičnim algoritmima raspoređivanja poruka.

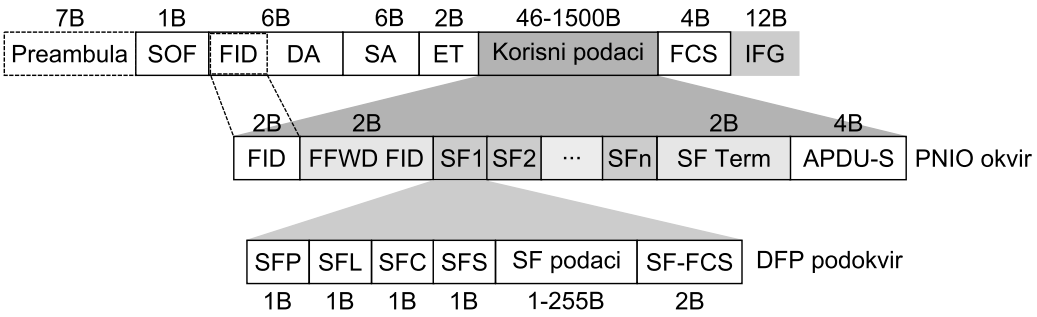
Drugi aspekt se odnosi na optimizaciju vremenskog plana slanja izohronih poruka koji direktno određuje dužinu trajanja IRT faze. Algoritam za kreiranje pomenutog vremenskog plana u komercijalnim alatima, poput *Siemens Simatic Step 7*, nije javno dostupan. Jedino je poznato da se bazira na brzom algoritmu za pronalaženje najkraće putanje u mreži koji je predložen u [100].

U radu [17], pokazano je da se optimalan plan, u slučaju linijske topologije, dobija primjenom *slipstreaming* efekta opisanog u prethodnom poglavlju. Međutim, kod kompleksnijih mrežnih konfiguracija, kreiranje optimalnog vremenskog plana može da bude vrlo zahtjevan zadatak. Koncept *slipstreaming* efekta može se proširiti na mrežu proizvoljne topologije, čija konfiguracija ne sadrži ciklične putanje, kao što je predloženo u radu [38]. U istom radu, autori su pokazali da predloženi algoritam daje optimalan vremenski plan za pomenute topološke strukture, što je i verifikovano na primjeru topologija linije, stabla i linije sa ograncima. Mogućnost proširenja pomenutog algoritma na redundantne topološke strukture je razmatrana u [39, 80].

5.3.1 Dinamičko pakovanje okvira

Kod dinamičkog pakovanja okvira, uvodi se mehanizam pakovanja više podokvira (engl. *subframe*) u jedan PNIO okvir, čime se neutrališe negativan efekat zahtijevane minimalne veličine *Ethernet* okvira od 64 bajta. Koncept je sličan konceptu sumacionog okvira u ECAT protokolu, s tom razlikom da je pakovanje okvira dinamičko, tj. uređaji, po prijemu okvira, uklanjaju podatke koji su im namijenjeni i na taj način smanjuju veličinu okvira. Ovo dodatno povećava efikasnost sistema, naročito u slučaju linijske topologije. Način enkapsulacije i izgled strukture DFP podokvira prikazani su na slici 5.20.

Kao što može da se vidi sa slike, područje podokvira je omeđeno poljima FFWD FID (engl. *Fast Forward Frame ID*) i SF Term (engl. *Subframe Terminator*), svako veličine 2 bajta, čija je uloga da definišu identifikator podokvira koji omogućava brzo prosljeđivanje podokvira i označe kraj segmenta u kojem se nalaze podokviri, redom. S ciljem dodatnog poboljšanja



Slika 5.20: Struktura DFP podokvira

komunikacije, uveden je koncept ranog prosljeđivanja PNIO okvira (engl. *Fast Forwarding*).

Kod ovog koncepta, identifikator okvira je smješten u prva dva bajta odredišne MAC adrese, što omogućava ranu detekciju vrste okvira i brže prosljeđivanje na odredišni port. Osim toga, predloženo je da se izostavi preambula koju čini 7 bajtova, kako bi se dodatno smanjilo vrijeme slanja okvira.

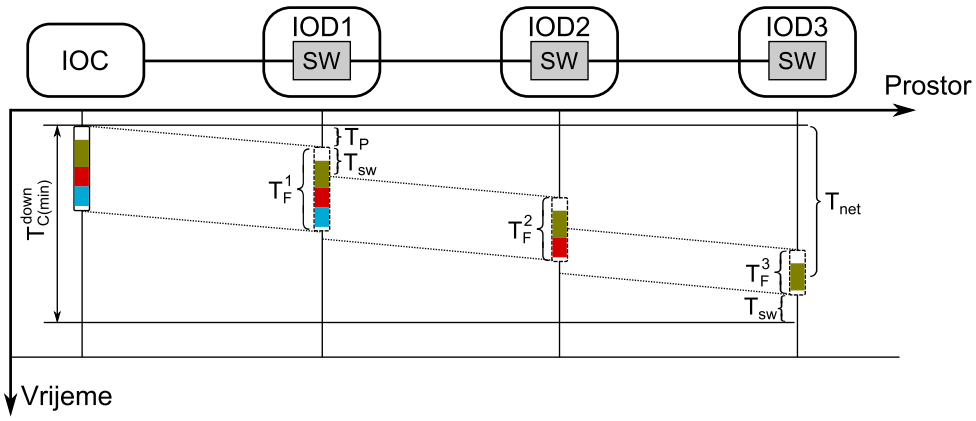
Zaglavlje svakog okvira čine sljedeća polja: SFP (engl *Subframe Position*), koje određuje poziciju odgovarajućeg uređaja, SFL (engl. *Subframe Length*), koje predstavlja dužinu korisnih podataka, SFC (engl. *Subframe Cycle Counter*), koje se uvećava u svakom ciklusu i na taj način omogućava praćenje vremenskog slijeda podokvira, te SFS (engl. *Subframe Data Status*), koje daje informaciju o trenutnom statusu podataka koji se prenose. Nakon zaglavlja, slijede korisni podaci čija je maksimalna dužina 255 bajtova. Konačno, svaki podokvir se završava 16-bitnim poljem SF-FCS za provjeru ispravnosti sadržaja.

Način rada PNIO sistema, koji koristi DFP koncept, u slučaju *downlink* komunikacije, ilustrovan je na slici 5.21. Važno je napomenuti da se, kod DFP komunikacije, razlikuju dva slučaja: kada je vrijeme slanja okvira manje od vremena propagacije signala kroz mrežu (slika 5.21a) i obrnuto, kada je vrijeme slanja okvira veće od vremena propagacije signala kroz mrežu (slika 5.21b). Koji od uslova će biti zadovoljen zavisi od kašnjenja mrežnih komponenata, topologije, dužine kablove, kao i od veličine procesnih podataka.

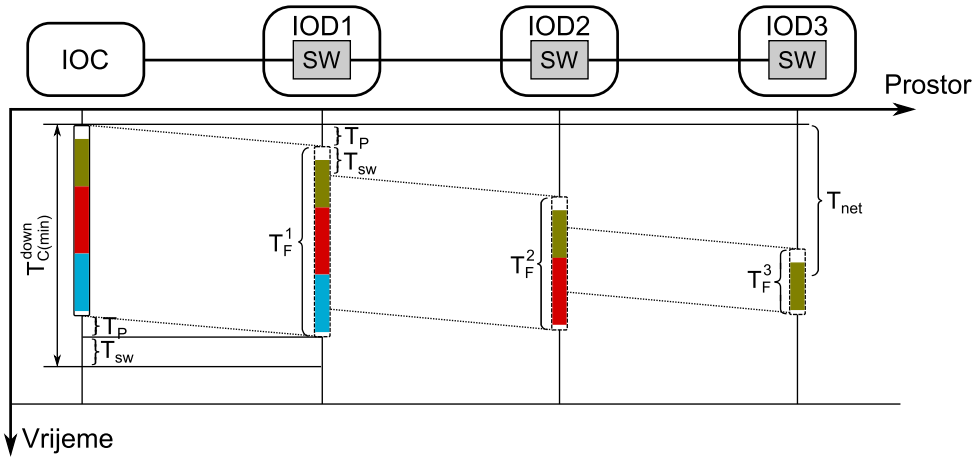
Sličan prostorno-vremenski dijagram se dobija i u slučaju *uplink* komunikacije, kao što je prikazano na slici 5.22. Bitna razlika je u tome što je slanje okvira u svim uređajima kontrolisano događajem (*Syncevent*), koji je sinhronizovan na nivou čitavog sistema, dok se kod *downlink* komunikacije slanje okvira inicira u kontroleru.

Zanimljivo je primjetiti da, u slučaju male veličine procesnih podataka (slika 5.22a), sumirajući okvir sadrži praznine (označeno sivom bojom na slici), koje nastaju kao rezultat dužeg vremena propagacije u odnosu na vrijeme potrebno za slanje svih procesnih podataka. Naime, u primjeru sa slike 5.22, svi uređaji istovremeno šalju okvire. S obzirom da uređaj IOD2 šalje okvir prije nego što se prime podaci od IOD3 uređaja, pakovanje procesnih podataka IOD3 uređaja u okvir IOD2 uređaja mora da se odgodi dok ne stignu ovi procesni podaci. To uzrokuje praznine u okviru koje utiču na efikasnost komunikacije. Ove praznine se mogu izbjeći uvođenjem pomjeraja pri slanju okvira IOD2 uređaja, tako da podaci od IOD3 uređaja stignu u trenutku kada se završi slanje podataka IOD2 uređaja. Međutim, ovim se ne postiže smanjenje minimalnog vremena komunikacionog ciklusa.

U opštem slučaju, vrijeme potrebno za slanje okvira kod DFP mehanizma može se odrediti



(a)



(b)

Slika 5.21: Ilustracija koncepta dinamičkog pakovanja okvira pri *downlink* komunikaciji za (a) $T_F < T_{net}$ i (b) $T_F > T_{net}$

pomoću jednačine (4.2), pri čemu je dužina okvira određena sa

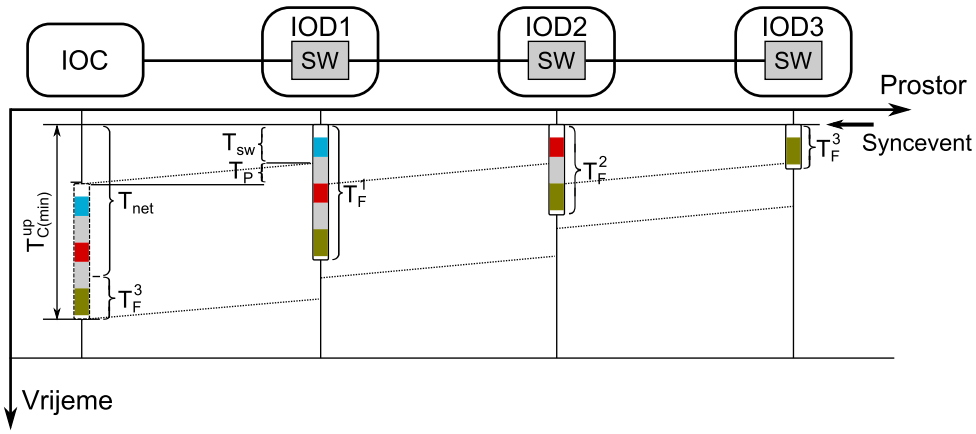
$$L_F = \max(L_{ovh} + L_{dfpov} + n(L_{sfov} + L_{payload}), L_{F(min)}), \tag{5.5}$$

gdje je: $L_{ovh} = 25$ – ukupna dužina kontrolnih informacija protokola (uključujući kontrolne informacije *Ethernet* protokola), $L_{dfpov} = 4$ – dužina kontrolnih informacija DFP protokola, n – broj podokvira koji odgovara broju uređaja u mreži, $L_{sfov} = 6$ – dužina kontrolnih informacija podokvira i $L_{payload}$ – veličina procesnih podataka jednog uređaja.

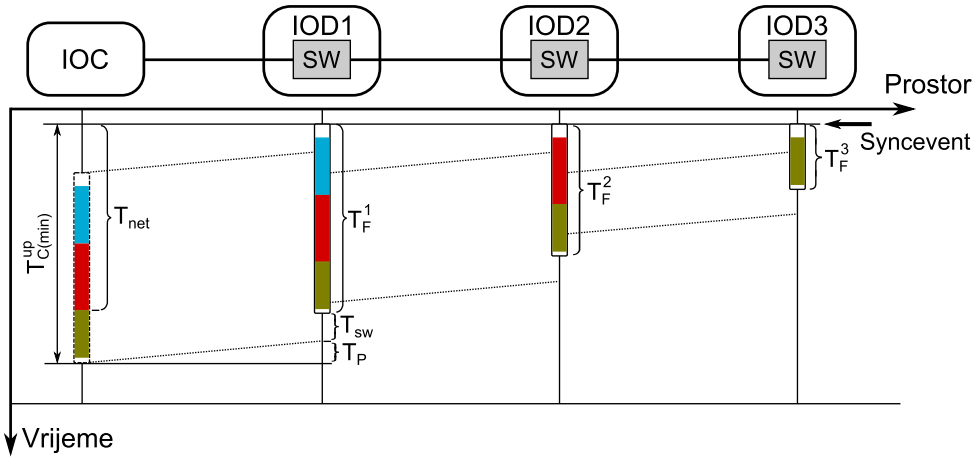
S druge strane, vrijeme kašnjenja zbog propagacije kroz mrežu, za linijsku i topologiju zvijezde, određeno je sa:

$$T_{net}^{line} = n(T_P + T_{sw}), \tag{5.6}$$

$$T_{net}^{star} = 2(T_P + T_{sw}). \tag{5.7}$$



(a)



(b)

Slika 5.22: Ilustracija koncepta dinamičkog pakovanja okvira pri *uplink* komunikaciji za (a) $T_F < T_{net}$ i (b) $T_F > T_{net}$

U slučaju kada je $T_F < T_{net}$, dužina okvira je određena na sljedeći način:

$$L_F^1 = \max(L_{ovh} + L_{dfpov} + L_{sfov} + L_{payload}, L_{F(min)}), \tag{5.8}$$

gdje L_F^1 označava dužinu okvira kada se prenose podaci samo jednog uređaja. Uzimajući u obzir prethodne jednačine, vrijeme minimalnog komunikacionog ciklusa, koje odgovara trajanju IRT faze, se za linijsku i topologiju zvijezde može opisati sljedećim formulama:

$$T_{C(min)}^{line} = T_{sync} + \max(T_{net}^{line} + T_F^1, T_F + T_{sw} + T_P), \tag{5.9}$$

$$T_{C(min)}^{star} = T_{sync} + T_F + T_{net}^{star}. \tag{5.10}$$

Kao što može da se zaključi iz izvedenih jednačina, DFP pristup, u slučaju linijske topologije, omogućava dinamičku redukciju veličine okvira prilikom njegovog prolaska kroz uređaje, što se pozitivno odražava na performanse sistema.

U cilju poređenja DFP koncepta sa standardnom PNIO komunikacijom, sprovedena je teorijska analiza korišćenjem jednačina analitičkog modela PNIO protokola iz odjeljka 4.7 i jednačina (5.9) i (5.10). Prilikom analize, uvedene su sljedeće pretpostavke: sistem se sastoji iz maksimalno 10 uređaja povezanih u linijsku ili topologiju zvijezde kablovima dužine 10 m, veličina procesnih podataka se mijenja u opsegu 1 do 100 bajtova, preambula je izostavljena prilikom slanja okvira i kašnjenje mrežnog komutatora, koji koristi DFP koncept, zbog implementiranih mehanizama brzog prosljeđivanja okvira, svedeno je na vrijednost od 1.2 μs, što je u skladu sa mjerenjima ostvarenim na prototipu [35].

Prva analiza podrazumijeva promjenu broja uređaja, pri čemu je veličina procesnih podataka fiksirana na 4 bajta. Rezultat ove analize, za standardni i DFP način rada i pomenute dvije topologije, prikazan je na slici 5.23.

Sa slike je jasno da DFP pristup obezbjeđuje kraće trajanje IRT faze za obje topologije. Takođe, može se zaključiti da se ova prednost povećava s brojem uređaja, što je i bilo očekivano, s obzirom da DFP poboljšava iskorišćenje propusnog opsega, naročito kada se koristi veliki broj uređaja.

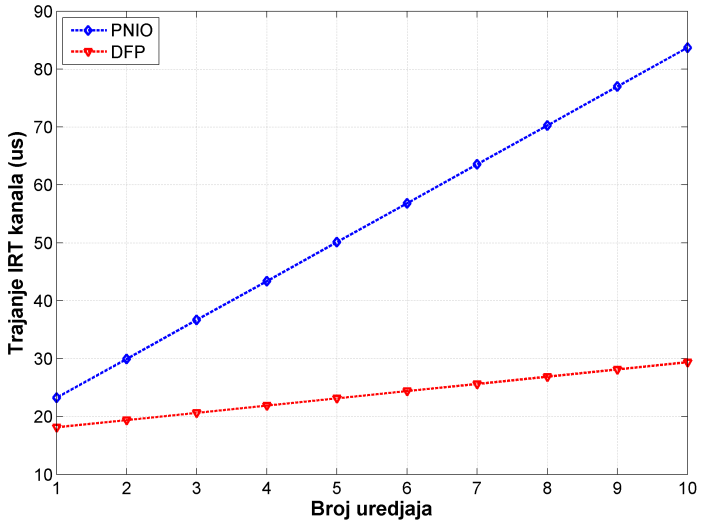
U drugoj analizi je, u mreži sa 10 uređaja, varirana veličina procesnih podataka u opsegu od 1 do 100 bajtova. Dobijeni rezultati, za dvije topologije, datisu na slici 5.24.

Kao što može da se vidi sa slike 5.24, kod linijske topologije, zadržava se trend povećanja performansi i za veću dužinu procesnih podataka. S druge strane, kod topologije zvijezde, prednost DFP pristupa dostiže fiksnu vrijednost kada veličina podataka pređe 36 bajtova. Takođe, interesantno je primjetiti da, za procesne podatke čija je veličina ispod 12 bajtova, DFP pristup, u slučaju topologije zvijezde, postiže bolje rezultate u odnosu na linijsku topologiju.

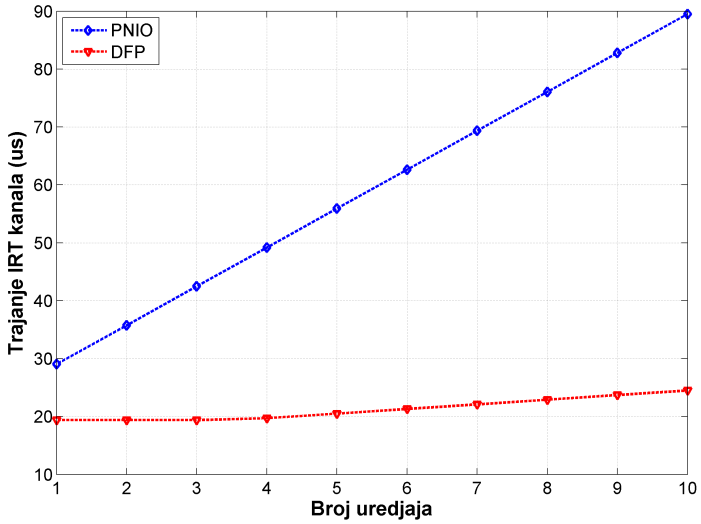
Pomenuta prednost ne zavisi od broja uređaja, kao što pokazuje slika 5.23. Ovo se može objasniti činjenicom da je kod linijske topologije, u navedenom slučaju, vrijeme propagacije signala kroz mrežne komponente dominatan član pri određivanju minimalnog vremena komunikacionog ciklusa. Međutim, pri većoj dužini procesnih podataka, do izražaja dolazi efekat dinamičkog smanjenja veličine okvira koji se koristi kod linijske topologije. S obzirom da ovaj koncept nije primjenjiv u slučaju topologije zvijezde, jasno je da vrijeme slanja okvira postaje dominantno, što se negativno odražava na performanse.

Opisani efekti se bolje mogu opisati i analizirati ako se prikaže zavisnost odnosa trajanja IRT faze za dva načina rada od broja uređaja (slika 5.25a) i veličine procesnih podataka (slika 5.25b).

Sa slike 5.25a, može da se vidi da topologija zvijezde ima približno fiksnu prednost u odnosu na linijsku topologiju, bez obzira na broj uređaja, kada je veličina procesnih podataka malena (u konkretnom slučaju je postavljena na 4 bajta). Takođe, može se zaključiti da se prednost u



(a)



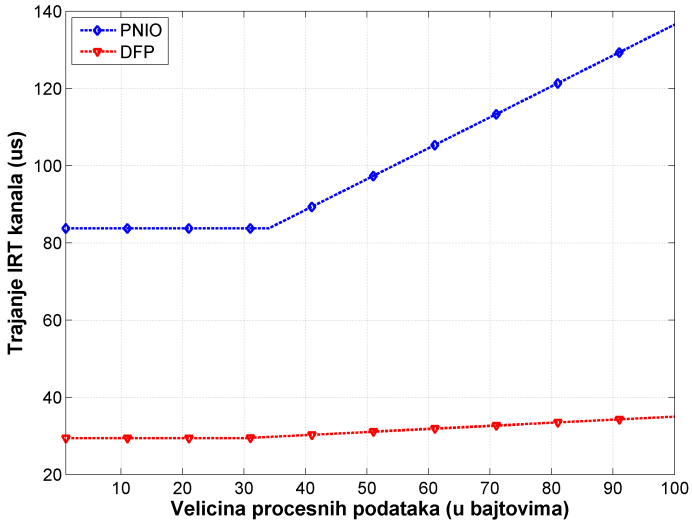
(b)

Slika 5.23: Trajanje IRT faze u funkciji broja uređaja za: (a) linijsku i (b) topologiju zvijezde (veličina procesnih podataka 4 bajta)

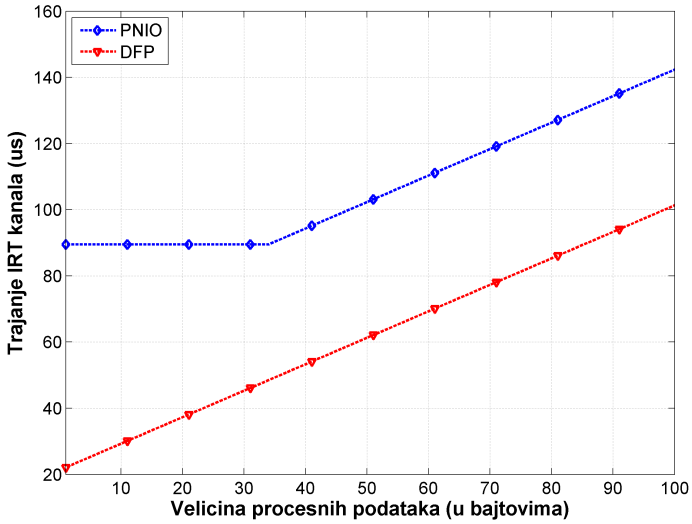
odnosu na standardni način rada, kod obje topologije, povećava s porastom broja uređaja (oko 35% kraće trajanje IRT faze u slučaju linijske i oko 28% u slučaju topologije zvijezde, za mrežu sa 10 uređaja).

Slika 5.25b opisuje efekte koji nastaju povećanjem veličine procesnih podataka. Sa slike se vidi da, u slučaju linijske topologije i za procesne podatke manje veličine, DFP pristup ima konstantnu prednost (oko 35% kraće trajanje IRT faze za mrežu sa 10 uređaja koja je analizirana u konkretnom slučaju) u odnosu na standardni način rada. Za veće procesne podatke, ova prednost počinje blago da raste i, pri veličini procesnih podataka od 100 bajtova, dostiže vrijednost koja odgovara oko 26% kraćem trajanju IRT faze u odnosu na standardni način rada.

S druge strane, topologija zvijezde inicijalno postiže bolje performanse u odnosu na linijsku



(a)



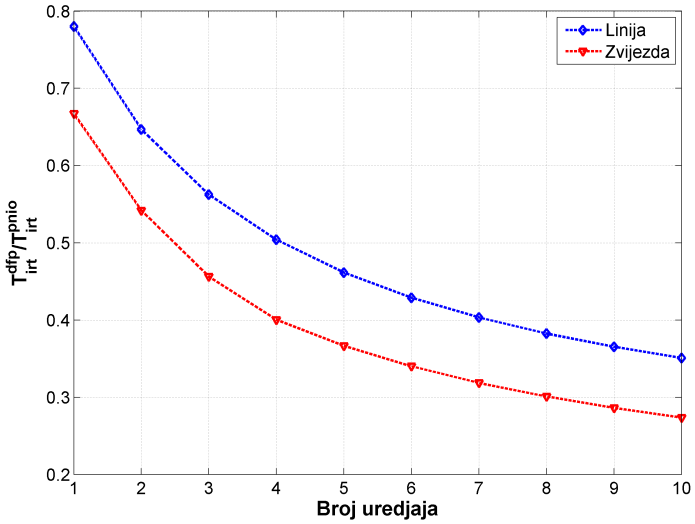
(b)

Slika 5.24: Trajanje IRT faze u funkciji veličine procesnih podataka za: (a) linijsku i (b) topologiju zvijezde (broj uređaja 10)

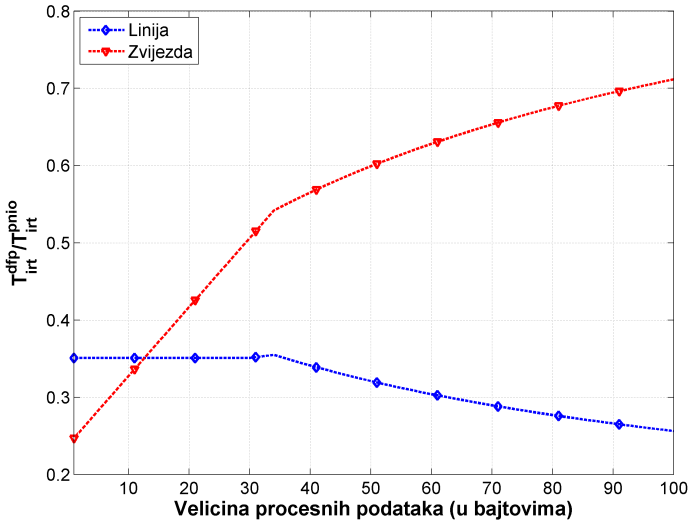
topologiju (oko 25% kraće trajanje IRT faze kada je veličina procesnih podataka 1 bajt). Međutim, kada veličina procesnih podataka postane veća od 12 bajtova, navedena prednost postaje sve manja. U slučaju procesnih podataka veličine 100 bajtova, dužina trajanja IRT faze dostiže oko 71% dužine trajanja ove faze u standardnom načinu rada.

Kao što može da se vidi iz prethodnog razmatranja, DFP mehanizam omogućava značajno poboljšanje performansi PNIO sistema, naročito kada se koristi linijska topologija. Kada je veličina procesnih podataka relativno mala, topologija zvijezde ima nešto izraženiju prednost u odnosu na linijsku topologiju bez obzira na broj uređaja u mreži.

Osnovni nedostatak predloženog rješenja se odnosi na pitanje kompatibilnosti sa prethodnim verzijama protokola i sa standardnim *Ethernet* komponentama. Naime, određene izmjene,



(a)



(b)

Slika 5.25: Odnos $T_{irt}^{DFP}/T_{irt}^{PNIO}$ za dvije topologije u funkciji: (a) broja uređaja i (b) veličine procesnih podataka

kao što je izostavljanje preambule, nisu kompatibilne sa *Ethernet* specifikacijama koje većina proizvođača slijedi doslovce, što može da izazove određene probleme pri integraciji uređaja koji podržavaju DFP način rada sa uređajima drugih proizvođača. Takođe, s obzirom da su navedene izmjene još u fazi razvoja i da ne postoje iskustva korisnika u stvarnim sistemima automatizacije, nije moguće predvidjeti ponašanje ovih komponenata sistema u stvarnim uslovima, niti uočiti eventualne probleme koji mogu nastati prilikom proširenja sistema.

5.4 Zaključak

U poglavlju su predloženi i analizirani algoritmi i metode za optimizaciju vremena odziva kod RTE sistema koji su predmet istraživanja ove disertacije. Predloženi pristupi za poboljšanje performansi RTE sistema su eksperimentalno verifikovani.

U slučaju ECAT protokola, predložena su dva pristupa. Prvi pristup podrazumijeva modifikaciju topološke strukture mreže kako bi se smanjilo ukupno propagaciono kašnjenje. Pokazano je da se, u razmatranom scenariju, može postići smanjenje minimalnog vremena komunikacionog ciklusa za preko 30% u odnosu na linijsku topologiju. Takođe, predložen je analitički pristup kojim se može odrediti optimalan broj portova mrežnog komutatora. Drugi pristup omogućava optimizaciju dužine sumacionog okvira. U tu svrhu je predložen algoritam koji u obzir uzima fizička ograničenja definisana proizvodnim procesom, kao i poboljšanje dijagnostike sistema uvođenjem logičkih grupa ili domena. Ispravnost koncepta je eksperimentalno verifikovana na prototipu mreže sa manjim brojem uređaja. Optimalnost algoritma je formalno dokazana matematičkom indukcijom. Postignuto je smanjenje dužine sumacionog okvira do 24% u odnosu na rezultate dobijene komercijalnim konfiguracionim alatima.

U slučaju EPL pristupa, predložen je ET-PRC mehanizam koji predstavlja modifikaciju originalnog TT-PRC mehanizma. Predložena modifikacija je eksperimentalno verifikovana na prototipu mreže, a poboljšanje performansi je evaluirano simulacijama za kompleksniju mrežnu strukturu. Predloženi scenario ET-PRC postiže oko 10% kraću dužinu trajanja izohrone faze u odnosu na originalni (TT-PRC) pristup.

Kod PNIO protokola, analizirana je efikasnost DFP mehanizma za slučaj linijske i topologije zvijezde. Rezultati analize pokazuju da je, za procesne podatke manje veličine, moguće ostvariti i do 35% kraće trajanje izohronog intervala u odnosu na standardni način rada, kada se koristi topologija zvijezde. Linijska topologija omogućava bolje performanse u odnosu na topologiju zvijezde kada je veličina procesnih podataka iznad 12 bajtova. Ovo je posljedica dinamičkog smanjenja ukupne veličine okvira, koje se može postići kod linijske topologije.

6 Zaključak

U ovoj disertaciji data je teorijska i eksperimentalna analiza industrijskih komunikacionih sistema baziranih na *Ethernet* tehnologiji iz kategorije koja nudi najviši nivo performansi. Analizirane su mogućnosti tri komunikaciona rješenja koja su najčešće korišćena u stvarnim sistemima automatizacije: *EtherCAT*, *Ethernet Powerlink* i *Profinet IO*. Sprovedena analiza se oslanja na specifično definisane metrike za ocjenu performansi industrijskih komunikacionih sistema koje su propisane standardom IEC 61784-2, odnosno predložene od strane različitih autora u okviru studija u kojima se razmatra i ocjenjuje ciklično ponašanje ovih sistema. U tom smislu, predloženi su analitički modeli, koji na odgovarajući način opisuju ponašanje pomenutih komunikacionih protokola, uzimajući u obzir specifičnost konfiguracije ovakvih sistema, pri čemu su vremenski parametri pomenutih modela određeni eksperimentalno. Teorijska analiza je primjenjena na različite scenarije, a rezultati analize su prezentovani za različit broj uređaja i veličinu procesnih podataka koji se razmjenjuju u mreži.

Dodatnom uporednom analizom pomenuta tri protokola, pokazano je da, u smislu pomenutih metrika, najbolje performanse postiže *EtherCAT* sistem, pri čemu je razmatran scenario jednog tipičnog distribuiranog sistema automatizacije, koji može da se nađe u stvarnim uslovima. Prilikom analize, uzete su u obzir realne karakteristike komponenata sistema, kao i specifičnosti konfiguracije koje utiču na veličinu i redoslijed slanja okvira kojima se razmjenjuju vremenski kritični podaci. Iz dobijenih rezultata može se zaključiti da *EtherCAT* postiže za oko 30% kraće vrijeme komunikacionog ciklusa u odnosu na *Profinet IO*, dok je u slučaju *Ethernet Powerlink* protokola ova razlika još veća (oko 68% u slučaju standardnog i oko 60% u slučaju ulančanog načina rada). S obzirom da je kod analiziranih sistema, zbog mogućnosti ostvarivanja potpune sinhronizacije, vrijeme odziva dominantno određeno vremenom komunikacionog ciklusa, prethodno zaključivanje može se primijeniti i na vrijeme odziva.

Osim teorijske i ekperimentalne analize, u radu su predloženi i evaluirani pristupi koji omogućavaju optimizaciju *EtherCAT* i *Ethernet Powerlink* protokola i analizirane prednosti

i ograničenja DFP mehanizma, koji je nedavno predložen za optimizaciju komunikacije kod *Profinet IO* protokola.

U slučaju *EtherCAT* protokola predložena su dva pristupa. Prvi pristup predstavlja modifikaciju linijske topologije sa velikim brojem uređaja na način da sadrži više manjih linijskih segmenata, čime se smanjuje kašnjenje usljed propagacije okvira kroz mrežu. U tu svrhu je iskorišćena topologija zvijezde koja se bazira na mrežnom komutatoru sa VLAN oznakama. Teorijskom analizom pokazano je da se može postići vrijeme ciklusa koje je za više od 30% kraće u odnosu na klasičnu linijsku topologiju. Takođe, potvrđeno je da izbor broja portova komutatora određuje efikasnost primjenjenog metoda. U razmatranim slučajevima utvrđeno je da se optimalan izbor svodi na broj koji se dobija prilikom izjednačavanja propagacionog kašnjenja linijskog ogranka mreže (uključujući kašnjenje mrežnog komutatora) sa ukupnim vremenom potrebnim za slanje okvira svim ograncima.

Drugi pristup podrazumijeva primjenu predloženog algoritma kojim se može optimizovati dužina sumacionog okvira. Naime, u ovu svrhu je iskorišćena karakteristika *EtherCAT* protokola da, u okviru jednog telegrama, može dvosmjerno razmjenjivati podatke. Pravilnom konfiguracijom, moguće je značajno smanjiti ukupnu veličinu okvira, naročito ako je ostvarena određena simetrija veličine procesnih podataka u sistemu. Predložena modifikacija algoritma uključuje koncept grupisanja uređaja u fizičke grupe, kako bi se uvažila postojeća ograničenja koje uvodi sam proizvodni proces, kao i u logičke grupe (domene), kako bi se poboljšala dijagnostika sistema. Algoritam je primjenjen za različite scenarije koji podrazumijevaju stvarne mrežne konfiguracije, a ispravnost rada koncepta je demonstrirana na prototipu mreže manje veličine. Dobijeni rezultati pokazuju da se u stvarnim uslovima veličina okvira može smanjiti i do 20% u odnosu na komercijalno dostupne konfiguracione alate. Takođe, zaključeno je da se, u slučaju kompleksnijih mrežnih stuktura, postižu bolji rezultati.

Nedostatak oba prethodno opisana pristupa leži u tome što je za ostvarenje maksimalnih rezultata neophodno modifikovati topologiju mreže, što u određenim slučajevima nije izvodivo ili unosi dodatne troškove. U slučaju algoritma za optimizaciju dužine okvira, ovo je moguće svesti na minimalnu mjeru, a da se ipak dobiju zadovoljavajući rezultati.

Kod *Ethernet Powerlink* pristupa predložena je modifikacija originalnog pristupa sa ulančavanjem PRes okvira (PRC način rada) kod kojeg se čvorovi aktiviraju korišćenjem vremenski-inicirane paradigme (TT-PRC). Uvedena modifikacija podrazumijeva ulančavanje PRes okvira kontrolisano događajima (ET-PRC). Ovaj način rada zahtijeva neznatne softverske modifikacije dostupnih impementacija *Ethernet Powerlink* komunikacionog steka. Opisano rješenje je eksperimentalno verifikovano na prototipu mreže za različite topologije, a dobijeni rezultati su

upoređeni sa standardnim i TT-PRC načinom rada. U cilju evaluacije ET-PRC mehanizma za kompleksnije mrežne strukture, razvijen je simulacioni model za OMNeT++ simulator. Simulacionim analizama je pokazano da ET-PRC ima prednost od oko 10% u odnosu na TT-PRC pristup. Nedostatak ET-PRC mehanizma ogleda se u činjenici da gubitak PRes okvira dovodi do gubitka čitavog komunikacionog ciklusa, što pogoršava determinizam koji se može postići u mreži.

U slučaju *Profinet IO* protokola analiziran je mehanizam dinamičkog pakovanja okvira, koji je nedavno predložen s ciljem poboljšanja perfromansi sistema. Pomenuti mehanizam je detaljno opisan, a zatim je sprovedena teorijska analiza za dvije topološke strukture. Rezultati analize pokazuju da primijenjeni mehanizam omogućava i do 35% kraću dužinu trajanja izohronog intervala, u odnosu na standardni način rada. Takođe, pri veličini procesnih podataka ispod 12 bajtova, topologija zvijezde postiže bolje rezultate u odnosu na linijsku topologiju, bez obzira na broj uređaja koji se koristi u mreži. Međutim, za veće procesne podatke, linijska topologija ima prednost zbog dinamičke promjene veličine okvira. Iako nudi određene prednosti po pitanju efikasnosti komunikacije, DFP pristup ima i određene nedostake. Kao prvo, da bi se postigli maksimalni rezultati, u sistemu mora da se obezbijedi visok nivo sinhronizacije između uređaja (ispod 100 ns), što može da predstavlja izazov u slučaju kompleksnijih mrežnih konfiguracija. Drugo, vremenski plan slanja poruka (naročito pri *uplink* komunikaciji) mora da bude dodatno optimizovan kako bi se izbjegle praznine između susjednih podokvira. Konačno, predloženo rješenje je još uvijek u fazi razvoja i nije testirano u stvarnim okruženjima, tako da je teško predvidjeti efekte koji mogu da se pojave prilikom proširenja sistema.

Budući pravci istraživanja podrazumijevaju proširenje predloženih rješenja u cilju otklanjanja uočenih nedostataka. Tako, na primjer, kod *EtherCAT* protokola postoji prostor za optimizaciju u slučajevima kada veličina procesnih podataka prevazilazi opseg jednog *Ethernet* okvira maksimalne dužine. Takođe, optimalan izbor broja portova mrežnog komutatora, kod pristupa kojim se smanjuje propagaciono kašnjenje *EtherCAT* mreže, može se analitički odrediti na prethodno opisani način samo u slučaju homogene mreže kod koje svi uređaji imaju iste karakteristike i istu veličinu procesnih podataka. Kod kompleksnijih konfiguracija, koje mogu da imaju izraženu asimetriju, potrebno je koristiti drugačiji pristup. U tom smislu se može formalizovati algoritam koji uzima u obzir navedene efekte i kao rezultat daje optimalnu mrežnu strukturu.

U slučaju *Ethernet Powerlink* protokola, ET-PRC mehanizam se može unaprijediti tako da omogući ispravan rad i u slučaju gubitka okvira, na način da upravljački uređaj kontinualno nadzire komunikaciju i obezbjeđuje odgovarajuće signalne poruke koje omogućavaju nastavak komunikacije u slučaju greške. Dalje istraživanje, takođe, može da se fokusira i na mogućnost

integracije opisanog mehanizma sa konceptima koji podrazumijevaju automatsku konfiguraciju sistema.

Kod *Profinet IO* komunikacije postoji prostor za unapređenje algoritama koji kreiraju vremenski plan slanja poruka u standardnom i DFP načinu rada. U tom smislu, potrebno je analizirati specifičnosti postojećih algoritama kako bi se identifikovao prostor za eventualna unapređenja.

Bibliografija

[1] M. Hermann, T. Pentek, and B. Otto, "Design principles for industrie 4.0 Scenarios," in *Proc. 49th Hawaii Int. Conf. System Sciences (HICSS)*, Jan. 2016, pp. 3928–3937.

[2] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, oct 2010. [Online]. Available: https://doi.org/10.1016%2Fj.comnet.2010.05.010

[3] Y.-C. Li, S. H. Hong, X. Huang, G. Chen, and X. Liang, "Implementation of a powerlink-wirelesshart gateway for industrial automation," in *Proc. Telecommunications and Information Technology (ECTI-CON) 2016 13th Int. Conf. Electrical Engineering/Electronics, Computer*, Jun. 2016, pp. 1–6.

[4] L. Wisniewski, S. Chahar, and J. Jasperneite, "Seamless reconfiguration of time triggered ethernet based protocols," in *Proc. IEEE World Conf. Factory Communication Systems (WFCS)*, May 2015, pp. 1–4.

[5] G. Reinhart, S. Krug, S. Hüttner, Z. Mari, F. Riedelbauch, and M. Schlögel, "Automatic configuration (plug amp; produce) of industrial ethernet networks," in *Proc. 9th IEEE/IAS Int. Conf. Industry Applications - INDUSCON 2010*, Nov. 2010, pp. 1–6.

[6] L. Duerkop, H. Trsek, J. Jasperneite, and L. Wisniewski, "Towards autoconfiguration of industrial automation systems: A case study using profinet IO," in *Proc. IEEE 17th Int. Conf. Emerging Technologies Factory Automation (ETFA 2012)*, Sep. 2012, pp. 1–8.

[7] L. Dürkop, J. Jasperneite, and A. Fay, "An analysis of real-time ethernets with regard to their automatic configuration," in *Proc. IEEE World Conf. Factory Communication Systems (WFCS)*, May 2015, pp. 1–8.

[8] T. Sauter, "The three generations of field-level networks —evolution and compatibility issues," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3585–3595, Nov. 2010.

[9] T. Sauter, "The continuing evolution of integration in manufacturing automation," *IEEE Industrial Electronics Magazine*, vol. 1, no. 1, pp. 10–19, May 2007.

[10] J. D. Decotignie, "Ethernet-based real-time and industrial communications," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1102–1117, Jun. 2005.

[11] M. Felser, "Real-time ethernet - industry prospective," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1118–1129, Jun. 2005.

[12] M. Felser and T. Sauter, "Standardization of industrial ethernet - the next battlefield?" in *Proc. IEEE Int. Workshop Factory Communication Systems*, Sep. 2004, pp. 413–420.

[13] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems – a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 403–416, feb 2013. [Online]. Available: https://doi.org/10.1109%2Ftii.2012.2219540

[14] P. Gaj, J. Jasperneite, and M. Felser, "Computer communication within industrial distributed environment —a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 182–189, Feb. 2013.

[15] J. Jasperneite, J. Imtiaz, M. Schumacher, and K. Weber, "A proposal for a generic real-time ethernet system," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 2, pp. 75–85, May 2009.

[16] P. Neumann, "Communication in industrial automation—what is going on?" *Control Engineering Practice*, vol. 15, no. 11, pp. 1332–1347, nov 2007. [Online]. Available: https://doi.org/10.1016%2Fj.conengprac.2006.10.004

[17] J. Jasperneite, M. Schumacher, and K. Weber, "Limits of increasing the performance of industrial ethernet protocols," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 17–24.

[18] G. Prytz, "A performance analysis of ethercat and profinet irt," in *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*, Sep. 2008, pp. 408–415.

[19] J. Robert, J.-P. Georges, E. Rondeau, and T. Divoux, "Minimum cycle time analysis of ethernet-based real-time protocols," *International Journal of Computers Communications & Control*, vol. 7, no. 4, p. 744, sep 2014. [Online]. Available: https://doi.org/10.15837%2Fijccc.2012.4.1372

[20] M. Sung, I. Kim, and T. Kim, "Toward a holistic delay analysis of EtherCAT synchronized control processes," *International Journal of Computers Communications & Control*, vol. 8, no. 4, p. 608, aug 2013. [Online]. Available: https://doi.org/10.15837%2Fijccc.2013.4.384

[21] R. Schlesinger, A. Springer, and T. Sauter, "New approach for improvements and comparison of high performance real-time ethernet networks," in *Proc. IEEE Emerging Technology and Factory Automation (ETFA)*, Sep. 2014, pp. 1–4.

[22] A. Soury, M. Charfi, D. Genon-Catalot, and J. M. Thiriet, "Performance analysis of ethernet powerlink protocol: Application to a new lift system generation," in *Proc. IEEE 20th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2015, pp. 1–6.

[23] M. Knezic, B. Dokic, and Z. Ivanovic, "Performance analysis of the ethernet powerlink pollresponse chaining mechanism," in *Proc. IEEE World Conf. Factory Communication Systems (WFCS)*, May 2015, pp. 1–4.

[24] P. Ferrari, A. Flammini, D. Marioli, A. Taroni, and F. Venturini, "New simulation models to evaluate performance of profinet IO class 1 Systems," in *Proc. 5th IEEE Int. Conf. Industrial Informatics*, vol. 1, Jun. 2007, pp. 237–242.

[25] L. Seno and S. Vitturi, "A simulation study of ethernet powerlink networks," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 740–743.

[26] S. Limal, S. Potier, B. Denis, and J. J. Lesage, "Formal verification of redundant media extension of ethernet powerlink," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 1045–1052.

[27] L. Seno and C. Zunino, "A simulation approach to a real-time ethernet protocol: Ethercat," in *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*, Sep. 2008, pp. 440–443.

[28] P. Ferrari, A. Flammini, D. Marioli, A. Taroni, and F. Venturini, "Experimental analysis to estimate jitter in profinet IO class 1 networks," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation*, Sep. 2006, pp. 429–432.

[29] P. Ferrari, A. Flammini, F. Venturini, and A. Augelli, "Large profinet IO RT networks for factory automation: A case study," in *Proc. ETFA2011*, Sep. 2011, pp. 1–4.

[30] P. Ferrari, A. Flammini, D. Marioli, A. Taroni, and F. Venturini, "Evaluation of timing characteristics of a prototype system based on profinet IO RT_class 3," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 1254–1261.

[31] M. Cereia, I. C. Bertolotti, and S. Scanzio, "Performance of a real-time ethercat master under linux," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 679–687, Nov. 2011.

[32] K. Erwinski, M. Paprocki, L. M. Grzesiak, K. Karwowski, and A. Wawrzak, "Application of ethernet powerlink for communication in a linux rtai open cnc system," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 2, pp. 628–636, Feb. 2013.

[33] M. Knezic, B. Dokic, and Z. Ivanovic, "Theoretical and experimental evaluation of ethernet powerlink pollresponse chaining mechanism," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, p. 1, 2016.

[34] C. Felser, M. Felser, and H. Kaghazchi, "Improved architecture for profinet irt devices," in *Proc. IEEE 17th Int. Conf. Emerging Technologies Factory Automation (ETFA 2012)*, Sep. 2012, pp. 1–8.

[35] D. Gunzinger, C. Kuenzle, A. Schwarz, H. D. Doran, and K. Weber, "Optimising profinet irt for fast cycle times: A proof of concept," in *Proc. IEEE Int Workshop Factory Communication Systems*, May 2010, pp. 35–42.

[36] O. Graeser and O. Niggemann, "Planning of time triggered communication schedules," in *Software-intensive verteilte Echtzeitsysteme*. Springer Nature, 2009, pp. 21–30. [Online]. Available: https://doi.org/10.1007%2F978-3-642-04783-1_3

[37] Z. Hanzalek, P. Burget, and P. Sucha, "Profinet IO irt message scheduling with temporal constraints," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 3, pp. 369–380, Aug. 2010.

[38] L. Wisniewski, M. Schumacher, J. Jasperneite, and C. Diedrich, "Increasing flexibility of time triggered ethernet based systems by optimal greedy scheduling approach," in *Proc. IEEE 20th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2015, pp. 1–6.

[39] L. Wisniewski, V. Wendt, J. Jasperneite, and C. Diedrich, "Scheduling of profinet irt communication in redundant network topologies," in *Proc. IEEE World Conf. Factory Communication Systems (WFCS)*, May 2016, pp. 1–4.

[40] L. Wisniewski, M. Schumacher, J. Jasperneite, and S. Schriegel, "Fast and simple scheduling algorithm for profinet irt networks," in *Proc. 9th IEEE Int. Workshop Factory Communication Systems*, May 2012, pp. 141–144.

[41] L. Wisniewski, J. Jasperneite, and C. Diedrich, "Effective and fast approach to schedule communication in profinet irt networks," in *Proc. IEEE Int. Symp. Industrial Electronics*, May 2013, pp. 1–6.

[42] R. Schlesinger, A. Springer, and T. Sauter, "Automatic packing mechanism for simplification of the scheduling in profinet irt," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1822–1831, Oct. 2016.

[43] M. Knezic, B. Dokic, and Z. Ivanovic, "Increasing ethercat performance using frame size optimization algorithm," in *Proc. IEEE 16th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2011, pp. 1–4.

[44] T. Maruyama and T. Yamada, "Hardware acceleration architecture for ethercat master controller," in *Proc. 9th IEEE Int Factory Communication Systems (WFCS) Workshop*, May 2012, pp. 223–232.

[45] D. Orfanus, R. Indergaard, G. Prytz, and T. Wien, "Ethercat-based platform for distributed control in high-performance industrial applications," in *Proc. IEEE 18th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2013, pp. 1–8.

[46] M. Knezic, B. Dokic, and Z. Ivanovic, "Topology aspects in ethercat networks," in *Proc. 14th Int. Power Electronics and Motion Control Conf. (EPE/PEMC)*, Sep. 2010, pp. T1–1–T1–6.

[47] G. Cena, S. Scanzio, A. Valenzano, and C. Zunino, "A distribute-merge switch for ethercat networks," in *Proc. 8th IEEE Int Factory Communication Systems (WFCS) Workshop*, May 2010, pp. 121–130.

[48] G. Cena, A. Valenzano, and C. Zunino, "An arbitration-based access scheme for ethercat networks," in *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*, Sep. 2008, pp. 416–423.

[49] G. Cena, I. C. Bertolotti, A. Valenzano, and C. Zunino, "A high-performance can-like arbitration scheme for ethercat," in *Proc. IEEE Conf. Emerging Technologies Factory Automation*, Sep. 2009, pp. 1–8.

[50] L. L. Bello, E. Bini, and G. Patti, "Priority-driven swapping-based scheduling of aperiodic real-time messages over ethercat networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 741–751, Jun. 2015.

[51] G. Cena, L. Seno, A. Valenzano, and S. Vitturi, "Performance analysis of ethernet powerlink networks for distributed control and automation systems," *Computer Standards & Interfaces*, vol. 31, no. 3, pp. 566–572, mar 2009. [Online]. Available: https://doi.org/10.1016%2Fj.csi.2008.03.022

[52] S. Vitturi, L. Peretti, L. Seno, M. Zigliotto, and C. Zunino, "Real-time ethernet networks for motion control," *Computer Standards & Interfaces*, vol. 33, no. 5, pp. 465–476, sep 2011. [Online]. Available: https://doi.org/10.1016%2Fj.csi.2011.01.005

[53] "IEC 61784, Digital data communications for measurement and control, Part 2: Additional profiles for ISO/IEC 8802-3 based communication networks in real-time applications," 2007.

[54] L. Seno, S. Vitturi, and C. Zunino, "Analysis of ethernet powerlink wireless extensions based on the IEEE 802.11 WLAN," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 2, pp. 86–98, May 2009.

[55] G. Gamba, L. Seno, and S. Vitturi, "Theoretical and experimental evaluation of polling times for wireless industrial networks using commercially available components," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (ETFA)*, Sep. 2010, pp. 1–8.

[56] L. Seno, F. Tramarin, and S. Vitturi, "Performance of industrial communication systems: Real application contexts," *IEEE Industrial Electronics Magazine*, vol. 6, no. 2, pp. 27–37, jun 2012. [Online]. Available: https://doi.org/10.1109%2Fmie.2012.2193292

[57] J. A. Maestro and P. Reviriego, "Energy efficiency in industrial ethernet: The case of powerlink," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 8, pp. 2896–2903, Aug. 2010.

[58] F. Tramarin and S. Vitturi, "Energy efficient ethernet for the industrial communication scenario," in *Proc. IEEE 18th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2013, pp. 1–8.

[59] F. Tramarin and S. Vitturi, "Strategies and services for energy efficiency in real-time ethernet networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 841–852, Jun. 2015.

[60] "IEEE Std. 802.3az-2009, Energy Efficient Ethernet Standard Draft 1.2.1," 2009.

[61] L. Seno, S. Vitturi, and C. Zunino, "Real time ethernet networks evaluation using performance indicators," in *Proc. IEEE Conf. Emerging Technologies Factory Automation*, Sep. 2009, pp. 1–8.

[62] G. Gamba, L. Seno, and S. Vitturi, "Performance indicators for wireless industrial communication networks," in *Proc. 8th IEEE Int Factory Communication Systems (WFCS) Workshop*, May 2010, pp. 3–12.

[63] P. Ferrari, A. Flammini, and S. Vitturi, "Response times evaluation of profinet networks," in *Proc. IEEE Int. Symp. Industrial Electronics ISIE 2005*, vol. 4, Jun. 2005, pp. 1371–1376.

[64] G. Marsal, B. Denis, J. M. Faure, and G. Frey, "Evaluation of response time in ethernet-based automation systems," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation*, Sep. 2006, pp. 380–387.

[65] B. Denis, S. Ruel, J. M. Faure, G. Frey, and G. Marsal, "Measuring the impact of vertical integration on response times in ethernet fieldbuses," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 532–539.

[66] B. Addad, S. Amari, and J. J. Lesage, "Analytic calculus of response time in networked automation systems," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 4, pp. 858–869, Oct. 2010.

[67] M. Felser, "Fieldbus based isochronous automation application," in *Proc. IEEE Conf. Emerging Technologies Factory Automation*, Sep. 2009, pp. 1–7.

[68] "EPSG Draft Standard 302-B, Ethernet Powerlink, Part B: Multiple ASnd Version 1.0.0," 2012. [Online]. Available: http://www.ethernet-powerlink.org

[69] "EPSG Draft Standard 311, Ethernet Powerlink, XML Device Description Version 1.2.0," 2016. [Online]. Available: http://www.ethernet-powerlink.org

[70] "EPSG Draft Standard 302-C, Ethernet Powerlink, Part C: PollResponse Chaining Version 1.0.0," 2012. [Online]. Available: http://www.ethernet-powerlink.org

[71] "IEEE Std. 802.1Q-2003, Virtual Bridged Local Area Networks," 2003.

[72] M. Knezic and Z. Ivanovic, "Evaluation of ethernet over ethercat protocol efficiency," in *Proc. INFOTEH-JAHORINA*, vol. 12, Mar. 2013, pp. 455–459.

[73] "Standard IEEE 1588. Standard for a precision clock synchronization protocol for networked measurement and control systems," 2002.

[74] G. Cena, I. C. Bertolotti, S. Scanzio, A. Valenzano, and C. Zunino, "Synchronize your watches: Part i: General-purpose solutions for distributed real-time control," *IEEE Industrial Electronics Magazine*, vol. 7, no. 1, pp. 18–29, Mar. 2013.

[75] J. Jasperneite, K. Shehab, and K. Weber, "Enhancements to the time synchronization standard IEEE-1588 for a system of cascaded bridges," in *Proc. IEEE Int. Workshop Factory Communication Systems*, Sep. 2004, pp. 239–244.

[76] G. Cena, I. C. Bertolotti, S. Scanzio, A. Valenzano, and C. Zunino, "Synchronize your watches: Part ii: Special-purpose solutions for distributed real-time control," *IEEE Industrial Electronics Magazine*, vol. 7, no. 2, pp. 27–39, Jun. 2013.

[77] G. Cena, I. C. Bertolotti, S. Scanzio, A. Valenzano, and C. Zunino, "Evaluation of ethercat distributed clock performance," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 1, pp. 20–29, Feb. 2012.

[78] M. Felser, "Media redundancy for profinet IO," in *Proc. IEEE Int. Workshop Factory Communication Systems*, May 2008, pp. 325–330.

[79] A. Giorgetti, F. Cugini, F. Paolucci, L. Valcarenghi, A. Pistone, and P. Castoldi, "Performance analysis of media redundancy protocol (mrp)," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 218–227, Feb. 2013.

[80] L. Wisniewski, M. Schumacher, J. Jasperneite, and C. Diedrich, "Linear time, possibly disjoint path search approach for ethernet based industrial automation networks," in *Proc. IEEE Emerging Technology and Factory Automation (ETFA)*, Sep. 2014, pp. 1–9.

[81] S. Höme, S. Palis, and C. Diedrich, "Design of communication systems for networked control system running on profinet," in *Proc. 10th IEEE Workshop Factory Communication Systems (WFCS 2014)*, May 2014, pp. 1–8.

[82] M.Knezic, B. Dokic, and Z. Ivanovic, "Influence of the ethercat frame transmission time on network efficiency in case of asymmetric traffic," in *Proc. 54th ETRAN Conf.*, Mar. 2010, pp. 1–4.

[83] N. Simanic, R. Exel, P. Loschmidt, T. Bigler, and N. Kerö, "Compensation of asymmetrical latency for ethernet clock synchronization," in *Proc. Control and Communication 2011 IEEE Int. Symp. Precision Clock Synchronization for Measurement*, Sep. 2011, pp. 19–24.

[84] F. Savic, M. Knezic, and B. Dokic, "Omnet + + simulation model for evaluation of scheduling algorithms in ftt-se networks," in *Proc. Int. Symp. Industrial Electronics (INDEL)*, Nov. 2016, pp. 1–5.

[85] "EPSG Draft Standard 301, Ethernet Powerlink, Communication Profile Specification Version 1.3.0," 2016. [Online]. Available: http://www.ethernet-powerlink.org

[86] Z. Hanzálek, P. Burget, and P. Sucha, "Profinet IO irt message scheduling," in *Proc. 21st Euromicro Conf. Real-Time Systems*, Jul. 2009, pp. 57–65.

[87] H. Frazier and H. Johnson, "Gigabit ethernet: from 100 to 1,000 Mbps," *IEEE Internet Computing*, vol. 3, no. 1, pp. 24–31, Jan. 1999.

[88] M. Molle, M. Kelkunte, and J. Kadambi, "Frame bursting: a technique for scaling CSMA/CD to gigabit speeds," *IEEE Network*, vol. 11, no. 4, pp. 6–15, Jul. 1997.

[89] G. Cena, S. Scanzio, A. Valenzano, and C. Zunino, "Performance analysis of switched ethercat networks," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (ETFA)*, Sep. 2010, pp. 1–4.

[90] M. Knezic, B. Dokic, and Z. Ivanovic, "Performance evaluation of the switched EtherCAT networks with VLAN tagging," *Serbian Journal of Electrical Engineering*, vol. 9, no. 1, pp. 33–42, 2012. [Online]. Available: https://doi.org/10.2298%2Fsjee1201033k

[91] M. Knezic, B. Dokic, and Z. Ivanovic, "A software tool for efficient configuration of ethercat networks," in *Proc. IEEE 17th Int. Conf. Emerging Technologies Factory Automation (ETFA 2012)*, Sep. 2012, pp. 1–4.

[92] "Beckhoff Automation GmbH. TwinCAT System Manager." [Online]. Available: http://www.beckhoff.com/

[93] "Koenig-PA GmbH. KPA Studio EtherCAT." [Online]. Available: https://koenig-pa.com

[94] N. Simanic, P. Loschmidt, G. Gaderer, J. Serrano, and N. Kerö, "Efficient large-scale oriented real-time network—-white rabbit based powerlink."

[95] S. Vitturi and F. Tramarin, "Energy efficient ethernet for real-time industrial networks," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 228–237, Jan. 2015.

[96] "POWERLINK Enhanced Ethernet MAC IP core," 2010. [Online]. Available: http://www.port.de

[97] F. Walzer, "POWERLINK on TI Sitara processors," 2015. [Online]. Available: http://www.ti.com

[98] M. Schumacher, J. Jasperneite, and K. Weber, "A new approach for increasing the performance of the industrial ethernet system profinet," in *Proc. IEEE Int. Workshop Factory Communication Systems*, May 2008, pp. 159–167.

[99] M. Schumacher, L. Wisniewski, J. Jasperneite, and S. Schriegel, "Node to node synchronization accuracy requirements of dynamic frame packing," in *Proc. IEEE Int. Symp. Precision Clock Synchronization for Measurement Control and Communication (ISPCS)*, Sep. 2013, pp. 53–58.

[100] U. Lauther, "An extremely fast, exact algorithm for finding shortest paths in static networks with geographical background," *Geoinformation und Mobilität-von der Forschung zur praktischen Anwendung*, vol. 22, pp. 219–230, 2004.

Biografija autora

Mladen Knežić je rođen 24.07.1981. godine u Jajcu, Bosna i Hercegovina. Osnovnu i srednju elektrotehničku školu, završio je 2000. godine u Brčkom odličnim uspjehom, a iste godine, upisao je osnovni studij elektrotehnike, odsjek za *Elektroniku i komunikacije* na Elektrotehničkom fakultetu u Banjoj Luci. Petogodišnje studije elektrotehnike, završio je 09.12.2005. godine odbranom diplomskog rada pod nazivom *"Primjena VHDL-a i FPGA tehnologija na primjeru projektovanja komponenata softverski definisanog radija"* sa prosječnom ocjenom 8,40. Od 01.02.2006. godine, zaposlen je na Elektrotehničkom fakultetu u Banjoj Luci u zvanju *saradnika sa punim radnim vremenom*. U zvanje *asistenta* na *Katedri za elektroniku*, izabran je 16.03.2006. godine na period od pet godina, a od 16.04.2009. godine ovo zvanje je ekvivalentirano sa izborima na uže naučne oblasti *Elektronika* i *Računarski hardver*.

Odmah poslije diplomiranja, upisao je postdiplomske studije na Elektrotehničkom fakultetu u Banjoj Luci, smjer *Elektronika*, u okviru kojih je položio sve ispite predviđene programom sa prosječnom ocjenom 10,00. Na istom fakultetu, 16.12.2010. godine, odbranio je magistarsku tezu pod nazivom *"Prilog analizi performansi EtherCAT sistema na bazi vremena komunikacionog ciklusa"* i time stekao akademsko zvanje *magistra elektrotehničkih nauka*. U zvanje *višeg asistenta* na užu naučnu oblast *Elektronika i elektronski sistemi*, izabran je 27.01.2011. godine.

Osim nastavnih aktivnosti, autor je, u okviru naučnog rada, objavio preko 20 naučnih radova u zbornicima radova konferencija i u naučnim časopisima. Predmet njegovog istraživanja uključuje različite aspekte simulacije i analize industrijskih komunikacionih sistema, te projektovanja ugrađenih računarskih sistema.

Образац -3

УНИВЕРЗИТЕТ У БАЊОЈ ЛУЦИ
ЕЛЕКТРОТЕХНИЧКИ ФАКУЛТЕТ

УНИВЕРЗИТЕТ У БАЊОЈ ЛУЦИ
ЕЛЕКТРОТЕХНИЧКИ ФАКУЛТЕТ
БАЊА ЛУКА
Број: 215
Датум: 06.03.2017.

ИЗВЈЕШТАЈ

о оцјени урађене докторске дисертације

I ПОДАЦИ О КОМИСИЈИ

На основу одлуке бр. 20/3.31-54/17 коју је донијело Наставно-научно вијеће Електротехничког факултета у Бањој Луци дана 31.01.2017. године, именована је Комисија за оцјену урађене докторске дисертације у сљедећем саставу:

1. **Др Предраг Петковић, редовни професор**, ужа научна област Електроника, Универзитет у Нишу, Електронски факултет, **предсједник**
2. **Др Бранко Докић, редовни професор**, ужа научна област Електроника и електронски системи, Универзитет у Бањој Луци, Електротехнички факултет, **ментор и члан**
3. **Др Бранко Блануша, ванредни професор**, ужа научна област Електроника и електронски системи, Универзитет у Бањој Луци, Електротехнички факултет, **члан**
4. **Др Гордана Гардашевић, ванредни професор**, ужа научна област Телекомуникације, Универзитет у Бањој Луци, Електротехнички факултет, **члан**
5. **Др Игор Крчмар, доцент**, ужа научна област Аутоматика и роботика, Универзитет у Бањој Луци, Електротехнички факултет, **члан**

II ПОДАЦИ О КАНДИДАТУ

МЛАДЕН (Бранко) КНЕЖИЋ
Рођен 24.07.1981. године у Јајцу, Босна и Херцеговина.
Послиједипломске студије је завршио на Електротехничком факултету, Универзитета у Бањој Луци.
Магистарску тезу под називом *Прилог анализи перформанси EtherCAT система на бази времена комуникационог циклуса* из уже научне области Електроника и електронски системи, одбранио је 16.12.2010. године на Електротехничком факултету, Универзитета у Бањој Луци и тиме стекао звање магистра електротехничких наука.

III УВОДНИ ДИО ОЦЈЕНЕ ДОКТОРСКЕ ДИСЕРТАЦИЈЕ

Наслов докторске дисертације мр Младена Кнежића је

ОПТИМИЗАЦИЈА ВРЕМЕНА ОДЗИВА ИНДУСТРИЈСКИХ КОМУНИКАЦИОНИХ МРЕЖА

Израду дисертације је одобрио Сенат Универзитета у Бањој Луци Одлуком бр. 02/04-3.75-56/13 од 31.01.2013. године.[1)]

Садржај докторске дисертације изложен је у сљедећим поглављима:

1. Увод (стр. 1-6),
2. Индустријски комуникациони системи (стр. 7-43),
3. Метрике у индустријским комуникационим системима (стр. 44-60),
4. Моделовање индустријских комуникационих система (стр. 61-118),
5. Механизми оптимизације времена одзива (стр. 119-157) и
6. Закључак (стр. 158-161).

Свако поглавље садржи више одјељака који су нумерисани у складу са уваженим стандардима писања научних радова. Осим општег закључка на крају рада (поглавље 6), свако поглавље се завршава одговарајућим закључком са јасно издвојеним резултатима истраживања тог поглавља. На крају рада, дат је списак коришћене литературе са 100 референци. Дисертација је написана на 161 страници текста (укупно 172 стране са литературом) и садржи 6 поглавља, 93 слике и 18 табела.

[1)] Сенат је одобрио тезу под радним насловом *Оптимизација времена одзива код индустријских комуникационих мрежа*. Након прегледа радне верзије дисертације, неки чланови Комисије су указали да је ријеч "код" сувишна, што је кандидат прихватио.

IV УВОД И ПРЕГЛЕД ЛИТЕРАТУРЕ

IV.1. Предмет, циљ и хипотезе истраживања

Предмет истраживања ове дисертације подразумијева теоријску и експерименталну анализу индустријских комуникационих система из категорије која омогућава највише перформансе и висок ниво синхронизације свих компонената система. Основну метрику за квантитативну процјену перформанси представља вријеме одзива система. Ова метрика се обично користи у комбинацији са осталим релевантним метрикама, које на објективан начин описују могућности индустријских комуникационих система. Спроведена анализа и експериментална евалуација омогућавају увођење одговарајућих механизама чији је основни циљ оптимизација времена одзива. Ови механизми подразумијевају промјену тополошке структуре мреже, оптимизацију временски критичне комуникације протокола и прецизно планирање слања порука како би се избјегла времена чекања услијед заузећа ресурса у мрежним компонентама.

Основни циљ докторске дисертације је да се, на основу резултата добијених симулацијом помоћу развијених аналитичких модела разматраних индустријских комуникационих система, добије одговор на питање да ли је одговарајућим планирањем топологије мреже и/или распореда слања порука могуће оптимизовати постојеће комуникационе механизме тако да се минимизује вријеме одзива индустријског комуникационог система, што је била и основна претпоставка (радна хипотеза) докторске дисертације.

IV.2. Преглед претходних истраживања

Модерни системи индустријске аутоматизације су све више оријентисани ка конвергенцији различитих технологија у циљу остварења концепта који је познат под називом *интелигентна фабрика* (енгл. *Smart Factory*) у оквиру *Индустрија 4.0* (енгл. *Industrie 4.0*) иницијативе [1] која подразумијева повезивање свих нивоа производње у јединствену виртуелну мрежу у склопу IoT (енгл. *Internet of Things*) парадигме [2, 3]. Најновије прогнозе показују да ће чак и ригидни системи аутоматизације у скоријој будућности бити базирани на сервисно оријентисаној парадигми која подразумијева концепт производње као услуге. То значи да ће будући производни системи морати да буду реконфигурабилни и адаптивни у смислу промјене процеса производње како би се на брз и ефикасан начин прилагодили захтјевима клијената [4]. У ту сврху, значајан напор се улаже у омогућавање концепта аутоматске конфигурације система приликом додавања нових уређаја по тзв. *Plug and Produce* принципу који је инспирисан сличним концептом (енгл. *Plug and Play*) који се већ дуго времена користи у рачунарским системима [5, 6, 7].

Индустријски комуникациони системи засновани на *Ethernet* технологији који подржавају рад у реалном времену, RTE (енгл. *Real-Time Ethernet*) системи, чине основну карику у ланцу који омогућава реализацију претходно описаних концепата. Основна предност коју ови системи имају у односу на традиционалне индустријске комуникационе системе познате под називом *fieldbus* су знатно шири пропусни опсег, који се све више захтијева у надолазећим апликацијама, као и инхерентна подршка за вертикалну интеграцију мрежа, које се користе у аутоматизацији на нивоу процеса производње (енгл. *field level*) са класичним рачунарским мрежама на нивоу предузећа (енгл. *enterprise level*), као што је описано у [8]. Посљедња карактеристика омогућава повезивање сектора производње са сектором управе предузећа, чиме се повећава ефикасност пословања и максимизује искоришћење постојећих ресурса [9]. Осим тога, ове мреже типично подразумијевају смањење трошкова који се односе на каблирање, дијагностиковање грешака и одржавање система.

Основни недостатак стандардне *Ethernet* технологије огледа се у томе што алгоритам за арбитрирање приступа заједничком медијуму није детерминистички, па стога није погодан за системе који захтијевају рад у реалном времену [10]. Тренутно постоји велики број различитих RTE приступа који омогућавају прилагођавање *Ethernet* технологије, тако да се омогући комуникација у реалном времену, од којих је већина дио међународних стандарда као што је описано у [8, 11, 12]. Сваки од приступа нуди одређени ниво компатибилности са стандардним *Ethernet* протоколом. Генерално, рјешења која користе стандардне *Ethernet* компоненте имају лошије перформансе у односу на рјешења која подразумијевају одређене модификације. Одабир исправне технологије генерално зависи од примјене. Ниједан од тренутно доступних RTE приступа не може да покрије све могуће сценарије уз истовремено задовољење темпоралних ограничења које намеће апликација [13]. Стога се очекује да будући комуникациони системи у дистрибуираним системима аутоматизације буду изразито хетерогени како би могли да се носе са све већом комплексношћу ових система [14].

У [15, 16], RTE протоколи су класификовани у три различите категорије, у зависности од нивоа перформанси које протокол може да обезбиједи. Протоколи из категорије 3, као што су нпр. *Profinet IRT*, *Ethernet Powerlink* и *EtherCAT*, обезбјеђују највиши ниво перформанси које захтијевају одређене апликације (нпр. апликације за

контролу кретања или *motion control*). Међутим, да би постигли захтијевани ниво перформанси, наведена рјешења уводе модификације на слоју података *Ethernet* протокола, који омогућавају механизме за побољшање детерминизма у комуникационом систему. То неминовно доводи до смањења компатибилности са стандардним компонентама, што у одређеној мјери може да утиче на цијену, али и на могућност интеграције комуникационог рјешења са класичним рачунарским мрежама.

У посљедњих неколико година, објављен је низ студија у којима су анализиране могућности различитих индустријских комуникационих система и предложене модификације у циљу побољшања њиховог рада, у контексту примјене у дистрибуираним системима аутоматизације. У већини радова, анализа је базирана на аналитичким моделима који су изведени за различите сценарије преузете из стварних система. Тако су, на примјер, у [17, 18, 19] аутори теоријски анализирали побољшање перформанси различитих RTE система при повећању битске брзине са 100 Mb/s на 1 Gb/s. Други аутори су увели додатна ограничења приликом теоријске анализе како би боље анализирали ефекте стварних апликација [20, 21, 22]. У [23, 24, 25, 26], описани су приступи за примјену различитих техника симулације и формалне верификације индустријских комуникационих система. Такође, да би се карактеризовали временски параметри развијених аналитичких и симулационих модела, спроведен је низ студија које су за крајњи циљ имале експерименталну анализу мрежних компонената индустријских комуникационих система [27, 28, 29, 30, 31].

Осим теоријске анализе, у литератури су предложени различити приступи за побољшање постојећих комуникационих механизама наведених RTE система. Тако је, на примјер, у [28] и [32] анализирана интерна архитектура *Profinet IO* контролера у циљу побољшања временских карактеристика система. У раду [33], предложен је DFP (енгл. *Dynamic Frame Packing*) механизам за побољшање перформанси код *Profinet IO* протокола. Други приступи (нпр. [34, 35, 36, 37]) дефинишу различите стратегије распоређивања слања порука како би се оптимизовале перформансе мреже у случају када се користе топологије којима се обично реализују стварни системими аутоматизације. Такође, у неким студијама анализиран је сличан проблем, али код примјене претходно поменутог DFP механизма [38, 39]. С обзиром да код *Profinet IO* система план слања порука треба поново да се генерише сваки пут када се промијени структура мреже, аутори су у [40] предложили модификацију DFP механизма, који су назвали APM (енгл. *Automatic Packing Mechanism*). APM механизам не захтијева временско распоређивање порука, чиме се унапређује флексибилност система уз истовремено побољшање перформанси. Међутим, наведени приступи су реализовани и тестирани само на нивоу прототипа и још увијек нису доступни у комерцијалној изведби, тако да није могуће предвидјети понашање предложених рјешења у стварним инсталацијама.

У случају *EtherCAT* pristupa, један дио истраживања је усмјерен на оптимизацију дужине оквира с обзиром на топологију мреже, која се обавља приликом конфигурације мреже. Према томе, структура мреже као и сва физичка и просторна ограничења морају да буду унапријед позната да би се постигла оптимална конфигурација чворова. Друга истраживања, на примјер [41, 42], оријентисана су ка различитим стратегијама базираним на хардверски подржаним архитектурама надређеног уређаја како би се обезбиједило веома кратко вријеме комуникационог циклуса, испод 100 μs. С обзиром да на ефикасност *EtherCAT* комуникације значајно утиче одабрана мрежна топологија, аутори [43] предложили су рјешење базирано на адаптацији топологије с циљем смањења укупног пропагационог кашњења мреже.

Такође, студије [44, 45, 46] обезбјеђују различите механизме за побољшање ефикасности размјене асинхроних порука како би се омогућила подршка за комуникацију у реалном времену, у случају апликација које обрађују непериодичне и спорадичне догађаје, као што су временски критични аларми.

Што се *Ethernet Powerlink* протокола тиче, аутори различитих студија бавили су се анализом перформанси овог приступа у индустријским окружењима. Тако, на примјер, у [47] аутори су спровели теоретску анализу и студију базирали на симулацијама за различите мрежне конфигурације како би истражили могућности и ограничења *Ethernet Powerlink* протокола приликом размјене цикличних и ацикличних података. Такође, у истом раду, аутори су предложили модификацију протокола како би побољшали његово понашање приликом опслуживања ацикличних података у реалном времену (нпр. алармних порука). У раду [48] је дата опширна анализа различитих метрика, које су дефинисане у оквиру IEC 61784-2 стандарда, за сценарио који подразумијева координисану контролу покрета. Оба претходно наведена рада анализирају понашање система у идеалним радним условима, као и у присуству грешака насталих при преносу порука. Могућности проширења *Ethernet Powerlink* протокола на примјену у бежичним мрежама, базираним на IEEE 802.11 стандарду, анализиране су у радовима [49] и [50]. Анализа понашања у бежичним примјенама проширена је експерименталним резултатима у радовима [50] и [51], при чему су мјерења обављена на прототипу стварне бежичне мреже која је реализована коришћењем комерцијално доступних компонената.

Коначно, један дио претходних истраживања (нпр. [52, 53, 54]) оријентисан је ка истраживању могућности побољшања енергетске ефикасности мрежних компонената, када се оне налазе у неактивном стању у складу са IEEE 802.3az стандардом [55].

Анализа перформанси било којег RTE система подразумијева дефинисање различитих метрика које омогућавају квантитативан опис могућности различитих рјешења. Неке од ових метрика су прописане одговарајућим стандардима [56], док су друге специфично дефинисане како би описале циклично понашање система [57, 58].

Најчешће коришћена метрика за опис цикличног понашања RTE система је минимално вријеме комуникационог циклуса. Међутим, ова метрика не узима у обзир друге аспекте који се обично захтијевају у дистрибуираним системима аутоматизације, попут синхронизације, пропусног опсега резервисаног за NRT саобраћај, или времена потребног за обраду података у контролеру и/или дистрибуираним уређајима. Из наведеног разлога, дефинишу се метрике које боље описују карактеристике система, или се поменута метрика користи у комбинацији са другим метрикама.

Вријеме одзива представља објективну мјеру могућности система по питању брзине реаговања на промјене. Ова метрика обједињује више аспеката који су потенцијално занемарени код минималног времена комуникационог циклуса. Евалуација индустријских комуникационих система у значајном броју студија користи вријеме одзива као метрику за квантитатвину процјену могућности система. У [27, 23, 59] је анализирано вријеме одзива за *Profinet IO* систем са несинхронизованом класом саобраћаја, док су у [60, 61, 62] презентовани теоријски и експериментални резултати за примјер генеричког RTE система. У студији [63] разматран је утицај синхронизације уређаја на перформансе система, у смислу минималног одступања времена одзива.

Литература цитирана у IV.2

[1] M. Hermann, T. Pentek, and B. Otto, "Design principles for industrie 4.0 Scenarios," in *Proc. 49th Hawaii Int. Conf. System Sciences (HICSS)*, Jan. 2016, pp. 3928–3937.

[2] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, oct 2010.

[3] Y.-C. Li, S. H. Hong, X. Huang, G. Chen, and X. Liang, "Implementation of a powerlink-wirelesshart gateway for industrial automation," in *Proc. Telecommunications and Information Technology (ECTI-CON) 2016 13th Int. Conf. Electrical Engineering/Electronics, Computer*, Jun. 2016, pp. 1–6.

[4] L. Wisniewski, S. Chahar, and J. Jasperneite, "Seamless reconfiguration of time triggered ethernet based protocols," in *Proc. IEEE World Conf. Factory Communication Systems (WFCS)*, May 2015, pp. 1–4.

[5] G. Reinhart, S. Krug, S. Hüttner, Z. Mari, F. Riedelbauch, and M. Schlögel, "Automatic configuration (plug & produce) of industrial ethernet networks," in *Proc. 9th IEEE/IAS Int. Conf. Industry Applications - INDUSCON 2010*, Nov. 2010, pp. 1–6.

[6] L. Duerkop, H. Trsek, J. Jasperneite, and L. Wisniewski, "Towards autoconfiguration of industrial automation systems: A case study using profinet IO," in *Proc. IEEE 17th Int. Conf. Emerging Technologies Factory Automation (ETFA 2012)*, Sep. 2012, pp. 1–8.

[7] L. Dürkop, J. Jasperneite, and A. Fay, "An analysis of real-time ethernets with regard to their automatic configuration," in *Proc. IEEE World Conf. Factory Communication Systems (WFCS)*, May 2015, pp. 1–8.

[8] T. Sauter, "The three generations of field-level networks — evolution and compatibility issues," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3585–3595, Nov. 2010.

[9] T. Sauter, "The continuing evolution of integration in manufacturing automation," *IEEE Industrial Electronics Magazine*, vol. 1, no. 1, pp. 10–19, 2007.

[10] J. D. Decotignie, "Ethernet-based real-time and industrial communications," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1102–1117, Jun. 2005.

[11] M. Felser, "Real-time ethernet - industry prospective," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1118–1129, Jun. 2005.

[12] M. Felser and T. Sauter, "Standardization of industrial ethernet - the next battlefield?" in *Proc. IEEE Int. Workshop Factory Communication Systems*, Sep. 2004, pp. 413–420.

[13] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems – a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 403–416, feb 2013.

[14] P. Gaj, J. Jasperneite, and M. Felser, "Computer communication within industrial distributed environment —a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 182–189, Feb. 2013.

[15] J. Jasperneite, J. Imtiaz, M. Schumacher, and K. Weber, "A proposal for a generic real-time ethernet system," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 2, pp. 75–85, May 2009.

[16] P. Neumann, "Communication in industrial automationwhat is going on?" *Control Engineering Practice*, vol. 15, no. 11, pp. 1332–1347, nov 2007.

[17] J. Jasperneite, M. Schumacher, and K. Weber, "Limits of increasing the performance

of industrial ethernet protocols," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 17–24.

[18] G. Prytz, "A performance analysis of ethercat and profinet irt," in *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*, Sep. 2008, pp. 408–415.

[19] J. Robert, J.-P. Georges, E. Rondeau, and T. Divoux, "Minimum cycle time analysis of ethernet-based real-time protocols," *International Journal of Computers Communications & Control*, vol. 7, no. 4, p. 744, sep 2014.

[20] M. Sung, I. Kim, and T. Kim, "Toward a holistic delay analysis of EtherCAT synchronized control processes," *International Journal of Computers Communications & Control*, vol. 8, no. 4, p. 608, aug 2013.

[21] R. Schlesinger, A. Springer, and T. Sauter, "New approach for improvements and comparison of high performance real-time ethernet networks," in *Proc. IEEE Emerging Technology and Factory Automation (ETFA)*, Sep. 2014, pp. 1–4.

[22] A. Soury, M. Charfi, D. Genon-Catalot, and J. M. Thiriet, "Performance analysis of ethernet powerlink protocol: Application to a new lift system generation," in *Proc. IEEE 20th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2015, pp. 1–6.

[23] P. Ferrari, A. Flammini, D. Marioli, A. Taroni, and F. Venturini, "New simulation models to evaluate performance of profinet IO class 1 Systems," in *Proc. 5th IEEE Int. Conf. Industrial Informatics*, vol. 1, Jun. 2007, pp. 237–242.

[24] L. Seno and S. Vitturi, "A simulation study of ethernet powerlink networks," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 740–743.

[25] S. Limal, S. Potier, B. Denis, and J. J. Lesage, "Formal verification of redundant media extension of ethernet powerlink," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 1045–1052.

[26] L. Seno and C. Zunino, "A simulation approach to a real-time ethernet protocol: Ethercat," in *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*, Sep. 2008, pp. 440–443.

[27] P. Ferrari, A. Flammini, D. Marioli, A. Taroni, and F. Venturini, "Experimental analysis to estimate jitter in profinet IO class 1 networks," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation*, Sep. 2006, pp. 429–432.

[28] P. Ferrari, A. Flammini, D. Marioli, A. Taroni, and F. Venturini, "Evaluation of timing characteristics of a prototype system based on profinet IO RT_class 3," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 1254–1261.

[29] P. Ferrari, A. Flammini, F. Venturini, and A. Augelli, "Large profinet IO RT networks for factory automation: A case study," in *Proc. ETFA2011*, Sep. 2011, pp. 1–4.

[30] M. Cereia, I. C. Bertolotti, and S. Scanzio, "Performance of a real-time ethercat master under linux," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 679–687, Nov. 2011.

[31] K. Erwinski, M. Paprocki, L. M. Grzesiak, K. Karwowski, and A. Wawrzak, "Application of ethernet powerlink for communication in a linux rtai open cnc system," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 2, pp. 628–636, Feb. 2013.

[32] C. Felser, M. Felser, and H. Kaghazchi, "Improved architecture for profinet irt devices," in *Proc. IEEE 17th Int. Conf. Emerging Technologies Factory Automation*

(ETFA 2012), Sep. 2012, pp. 1–8.

[33] D. Gunzinger, C. Kuenzle, A. Schwarz, H. D. Doran, and K. Weber, "Optimising profinet irt for fast cycle times: A proof of concept," in *Proc. IEEE Int Workshop Factory Communication Systems*, May 2010, pp. 35–42.

[34] O. Graeser and O. Niggemann, "Planning of time triggered communication schedules," in *Software-intensive verteilte Echtzeitsysteme*. Springer Nature, 2009, pp. 21–30.

[35] Z. Hanzalek, P. Burget, and P. Sucha, "Profinet IO irt message scheduling with temporal constraints," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 3, pp. 369–380, Aug. 2010.

[36] L. Wisniewski, M. Schumacher, J. Jasperneite, and C. Diedrich, "Increasing flexibility of time triggered ethernet based systems by optimal greedy scheduling approach," in *Proc. IEEE 20th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2015, pp. 1–6.

[37] L. Wisniewski, V. Wendt, J. Jasperneite, and C. Diedrich, "Scheduling of profinet irt communication in redundant network topologies," in *Proc. IEEE World Conf. Factory Communication Systems (WFCS)*, May 2016, pp. 1–4.

[38] L. Wisniewski, M. Schumacher, J. Jasperneite, and S. Schriegel, "Fast and simple scheduling algorithm for profinet irt networks," in *Proc. 9th IEEE Int. Workshop Factory Communication Systems*, May 2012, pp. 141–144.

[39] L. Wisniewski, J. Jasperneite, and C. Diedrich, "Effective and fast approach to schedule communication in profinet irt networks," in *Proc. IEEE Int. Symp. Industrial Electronics*, May 2013, pp. 1–6.

[40] R. Schlesinger, A. Springer, and T. Sauter, "Automatic packing mechanism for simplification of the scheduling in profinet irt," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1822–1831, Oct. 2016.

[41] T. Maruyama and T. Yamada, "Hardware acceleration architecture for ethercat master controller," in *Proc. 9th IEEE Int Factory Communication Systems (WFCS) Workshop*, May 2012, pp. 223–232.

[42] D. Orfanus, R. Indergaard, G. Prytz, and T. Wien, "Ethercat-based platform for distributed control in high-performance industrial applications," in *Proc. IEEE 18th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2013, pp. 1–8.

[43] G. Cena, S. Scanzio, A. Valenzano, and C. Zunino, "A distribute-merge switch for ethercat networks," in *Proc. 8th IEEE Int Factory Communication Systems (WFCS) Workshop*, May 2010, pp. 121–130.

[44] G. Cena, A. Valenzano, and C. Zunino, "An arbitration-based access scheme for ethercat networks," in *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*, Sep. 2008, pp. 416–423.

[45] G. Cena, I. C. Bertolotti, A. Valenzano, and C. Zunino, "A high-performance can-like arbitration scheme for ethercat," in *Proc. IEEE Conf. Emerging Technologies Factory Automation*, Sep. 2009, pp. 1–8.

[46] L. L. Bello, E. Bini, and G. Patti, "Priority-driven swapping-based scheduling of aperiodic real-time messages over ethercat networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 741–751, Jun. 2015.

[47] G. Cena, L. Seno, A. Valenzano, and S. Vitturi, "Performance analysis of ethernet powerlink networks for distributed control and automation systems," *Computer Standards & Interfaces*, vol. 31, no. 3, pp. 566–572, mar 2009.

[48] S. Vitturi, L. Peretti, L. Seno, M. Zigliotto, and C. Zunino, "Real-time ethernet

networks for motion control," *Computer Standards & Interfaces*, vol. 33, no. 5, pp. 465–476, sep 2011.

[49] L. Seno, S. Vitturi, and C. Zunino, "Analysis of ethernet powerlink wireless extensions based on the IEEE 802.11 WLAN," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 2, pp. 86–98, May 2009.

[50] G. Gamba, L. Seno, and S. Vitturi, "Theoretical and experimental evaluation of polling times for wireless industrial networks using commercially available components," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (ETFA)*, Sep. 2010, pp. 1–8.

[51] L. Seno, F. Tramarin, and S. Vitturi, "Performance of industrial communication systems: Real application contexts," *IEEE Industrial Electronics Magazine*, vol. 6, no. 2, pp. 27–37, jun 2012.

[52] J. A. Maestro and P. Reviriego, "Energy efficiency in industrial ethernet: The case of powerlink," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 8, pp. 2896–2903, Aug. 2010.

[53] F. Tramarin and S. Vitturi, "Energy efficient ethernet for the industrial communication scenario," in *Proc. IEEE 18th Conf. Emerging Technologies Factory Automation (ETFA)*, Sep. 2013, pp. 1–8.

[54] F. Tramarin and S. Vitturi, "Strategies and services for energy efficiency in real-time ethernet networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 841–852, Jun. 2015.

[55] "IEEE Std. 802.3az-2009, Energy Efficient Ethernet Standard Draft 1.2.1," 2009.

[56] "IEC 61784, Digital data communications for measurement and control, Part 2: Additional profiles for ISO/IEC 8802-3 based communication networks in real-time applications," 2007.

[57] L. Seno, S. Vitturi, and C. Zunino, "Real time ethernet networks evaluation using performance indicators," in *Proc. IEEE Conf. Emerging Technologies Factory Automation*, Sep. 2009, pp. 1–8.

[58] G. Gamba, L. Seno, and S. Vitturi, "Performance indicators for wireless industrial communication networks," in *Proc. 8th IEEE Int Factory Communication Systems (WFCS) Workshop*, May 2010, pp. 3–12.

[59] P. Ferrari, A. Flammini, and S. Vitturi, "Response times evaluation of profinet networks," in *Proc. IEEE Int. Symp. Industrial Electronics ISIE 2005*, vol. 4, Jun. 2005, pp. 1371–1376.

[60] G. Marsal, B. Denis, J. M. Faure, and G. Frey, "Evaluation of response time in ethernet-based automation systems," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation*, Sep. 2006, pp. 380–387.

[61] B. Denis, S. Ruel, J. M. Faure, G. Frey, and G. Marsal, "Measuring the impact of vertical integration on response times in ethernet fieldbuses," in *Proc. IEEE Conf. Emerging Technologies and Factory Automation (EFTA 2007)*, Sep. 2007, pp. 532–539.

[62] B. Addad, S. Amari, and J. J. Lesage, "Analytic calculus of response time in networked automation systems," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 4, pp. 858–869, Oct. 2010.

[63] M. Felser, "Fieldbus based isochronous automation application," in *Proc. IEEE Conf. Emerging Technologies Factory Automation*, Sep. 2009, pp. 1–7.

IV.3. Очекивани доприноси

У пријави дисертације, кандидат је навео сљедеће очекиване резултате:

1. преглед стања научног истраживања у предметној области,
2. постављање теоретске основе за анализу параметара индустријских комуникационих мрежа,
3. модели којима се описује понашање различитих типова индустријских комуникационих мрежа,
4. нови алгоритми и методе за оптимизацију времена одзива индустријских комуникационих мрежа и
5. софтвер за конфигурацију индустријских комуникационих мрежа.

Као што ће бити показано, сви су у потпуности остварени.

IV.4. Остварени доприноси

Прегледом докторске дисертације кандидата, Комисија је установила да су остварени сљедећи резултати и доприноси:

1. Кандидат је у уводном поглављу дао опсежан преглед актуелног научног истраживања у предметној области.
2. У дисертацији су дате теоретске основе разматраних индустријских комуникационих мрежа и детаљно описане метрике које карактеризују ове системе.
3. На основу презентованих теоретских основа, кандидат је развио оригиналне аналитичке моделе који описују понашање претходно поменутих индустријских комуникационих система. Параметри модела су добијени мјерењем на екпериментaлним поставкама које је кандидат реализовао посебно за ову намјену. Упоредном анализом разматрана три протокола, показано је да *EtherCAT* постиже за око 30% краће вријеме комуникационог циклуса у односу на *Profinet IO*, док је у случају *Ethernet Powerlink* протокола ова разлика још већа (око 68% у случају стандардног и око 60% у случају уланчаног начина рада).
4. Кандидат је предложио нове алгоритме и методе за оптимизацију времена одзива индустријских комуникационих мрежа. Добијени резултати показују да се, примјеном предложених алгоритама, величина *EtherCAT* оквира (а самим тим и вријеме одзива) може смањити и до 20%, у односу на комерцијално доступне конфигурационе алате. Такође, унапређени механизам уланчавања PRes оквира код *Ethernet Powerlink* протокола (ET-PRC), који је кандидат предложио и експериментално верификовао, омогућава побољшање перформанси од око 10% у односу на рјешења других аутора.
5. Како би могао тестирати предложене алгоритме, кандидат је реализовао специфичан софтвер који омогућава аутоматизовану конфигурацију мреже са оптималном тополошком структуром.

V МАТЕРИЈАЛ И МЕТОД РАДА

V.1. Материјал и критеријуми

Фокус истраживања, обрађен у дисертацији, јесте оптимизација времена одзива индустријских комуникационих мрежа.

Оптимизација комуникационе ефикасности индустријских комуникационих система, нарочито у погледу објективних метрика, као што је вријеме одзива, веома је актуелна, како у области академског истраживања, тако и у области примјене. Вријеме одзива представља једну од најзначајнијих метрика у индустријским комуникационим системима. Оно директно одређује ефикасност производног процеса, што даље утиче на цијену и квалитет крајњег производа. Код протокола највиших перформанси, који су предмет истраживања ове дисертације, вријеме одзива зависи од времена комуникационог циклуса које се може постићи у систему јер је остварена прецизна синхронизација свих компонената система.

V.2. Метод и актуелност истраживања

Кандидат у уводном поглављу указује на актуелност области истраживања. Затим прелази на појединости које нису довољно истражене из чега произилазе садржаји који су научни допринос његове докторске дисертације.

С обзиром да се постављени циљ дисертације односи на оптимизацију времена одзива индустријских комуникационих система, кандидат је прво указао на потребу за прецизним аналитичким моделима. На основу предложених модела, представљени су резултати симулације и компаративне анализе три разматрана RTE комуникациона система. Ови резултати омогућавају идентификацију простора за побољшање комуникационих механизама протокола који ће као резултат довести до побољшања времена одзива система.

На основу претходног произишли су оригинални приједлози техника и механизама који омогућавају оптимизацију времена одзива система. Кандидат предлаже три механизма за оптимизацију комуникације у *EtherCAT* и *Ethernet Powerlink* систему. Анализира њихове предности, недостатке и ограничења.

Теоретске поставке провјерене су рачунарским симулацијама и мјерењима на прототипима мрежа које је кандидат реализовао посебно за ове потребе. Експерименталне поставке укључују три разматрана комуникациона протокола. Резултати симулације и мјерења су представљени табеларно и одговарајућим дијаграмима. На основу тога кандидат извлачи одговарајуће закључке и препоруке. Дакле, кандидат је користио савремене научне методе и технике држећи се плана истраживања дефинисаног у пријави докторске дисертације.

Осим што је проблематика актуелна, примјењена методологија и технике рјешавања дефинисаног задатка су научно и методолошки савремени.

VI РЕЗУЛТАТИ И НАУЧНИ ДОПРИНОС ИСТРАЖИВАЊА

VI.1. Резултати истраживања

Поређењем са резултатима других аутора, Комисија закључује да су у раду остварени сљедећи научни доприноси:

Моделовање индустријских комуникационих система. Предложени су оригинални аналитички модели који на одговарајући начин описују понашање поменутих комуникационих протокола узимајући у обзир специфичност конфигурације оваквих система. Временски параметри поменутих модела одређени су експериментално. Теоријска анализа обухвата различите сценарије, а резултати анализе су презентовани с обзиром на број уређаја у мрежи и величину процесних података који се размјењују.

Додатном упоредном анализом поменута три протокола, показано је да, у смислу поменутих метрика, најбоље перформансе постиже *EtherCAT* систем, при чему је разматран сценарио једног типичног дистрибуираног система аутоматизације који може да се нађе у стварним инсталацијама. Приликом анализе, узете су у обзир реалне карактеристике компонената система, као и специфичности конфигурације које утичу на величину и редослијед слања оквира којима се размјењују временски критични подаци. Добијени резултати показују да *EtherCAT* постиже за око 30% краће вријеме комуникационог циклуса у односу на *Profinet IO*, док је у случају *Ethernet Powerlink* протокола ова разлика још већа (око 68% у случају стандардног и око 60% у случају уланчаног начина рада). С обзиром да је код анализираних система, због могућности остваривања потпуне синхронизације, вријеме одзива доминантно одређено временом комуникационог циклуса, претходно закључивање се може примјенити и на вријеме одзива.

Технике и механизми за оптимизацију времена одзива. У случају *EtherCAT* протокола предложена су два приступа. Први приступ представља модификацију линијске топологије са великим бројем уређаја на начин да садржи више мањих линијских сегмената, чиме се смањује кашњење усљед пропагације оквира кроз мрежу. У ту сврху искоришћена је топологија звијезде која се базира на мрежном преспојнику са VLAN ознакама. Теоријском анализом је показано да се може постићи вријеме циклуса које је за више од 30% краће у односу на класичну линијску топологију. Такође, потврђено је да избор броја портова преспојника одређује ефикасност примјењеног метода. У разматраним случајевима утврђено је да се оптималан избор своди на број који се добија приликом изједначавања пропагационог кашњења линијског огранка мреже (укључујући кашњење мрежног преспојника) са укупним временом потребним за слање оквира свим огранцима.

Други приступ подразумијева примјену алгоритма којим се може оптимизовати дужина сумационог оквира. Наиме, у ову сврху је искоришћена карактеристика *EtherCAT* протокола да може у оквиру једног телеграма двосмјерно размјењивати податке. Правилном конфигурацијом могуће је значајно смањити укупну величину оквира, нарочито ако је остварена одређена симетрија величине процесних података у систему. Предложена модификација алгоритма укључује два концепта груписања уређаја у:

- физичке групе, како би се уважила постојећа ограничења које уводи сам производни процес и

- логичке групе или домене, како би се побољшала дијагностика система.

Алгоритам је примјењен за различите сценарије који подразумијевају стварне мрежне конфигурације. Исправност рада концепта је демонстрирана на прототипу мреже мање величине. Добијени резултати показују да се у стварним условима величина оквира може смањити и до 20%, у односу на комерцијално доступне конфигурационе алате. Такође, аутор закључује да се, у случају комплекснијих мрежних стуктура, постижу бољи резултати.

Код *Ethernet Powerlink* приступа предложена је модификација оригиналног приступа са уланчавањем PRes оквира (PRC начин рада), код којег се чворови активирају коришћењем временски-иницираних парадигме (TT-PRC). Уведена модификација подразумијева уланчавање PRes оквира контролисано догађајима (ET-PRC). Овај начин рада захтијева незнатне софтверске модификације доступних имплементација *Ethernet Powerlink* комуникационог стека. Предложено рјешење је експериментално верификовано на прототипу мреже за различите топологије, а добијени резултати су упоређени са стандардним и TT-PRC начином рада. У циљу евалуације ET-PRC механизма за комплексније мрежне структуре, кандидат је развио симулациони модел за OMNeT++ симулатор. Симулационим анализама, кадидат је показао да ET-PRC има предност од око 10% у односу на TT-PRC приступ.

У случају *Profinet IO* протокола анализиран је механизам динамичког паковања оквира, који је недавно предложен, с циљем побољшања перформанси система. Поменути механизам је детаљно описан, а затим је спроведена теоријска анализа за двије тополошке структуре. Резултати анализе показују да примјењени механизам омогућава и до 35% краћу дужину трајања изохроног интервала у односу на стандардни начин рада. Такође, при величини процесних података испод 12 бајтова, топологија звијезде постиже боље резултате у односу на линијску топологију без обзира на број уређаја који се користи у мрежи. Међутим, за веће процесне податке, линијска топологија има предност због динамичке промјене величине оквира.

Кандидат је објавио девет радова који су директно везани са дисертацијом, а који су у дисертацији презентовани у цјелости или дјелимично. Један рад је објављен у часопису са SCI-E листе и фактором утицаја 4.708, један рад је објављен у часопису националног значаја, пет радова је презентовано на међународним конференцијама, док су три рада презентована на националним конференцијама, од којих је један проглашен за најбољи рад младог аутора у секцији Електроника. У свим овим радовима кандидат је први аутор. Такође, дисертација представља наставак истраживања започет у магистарском раду кандидата.

Хронолошки списак кандидатових публикација из области дисертације је сљедећи:

[1] M. Knezic, B. Dokic, and Z. Ivanovic, "Topology aspects in ethercat networks," in Proc. 14th Int. Power Electronics and Motion Control Conf. (EPE/PEMC), Sep. 2010, pp. T1–1–T1–6.

[2] M. Knezic, B. Dokic, and Z. Ivanovic, "Influence of the ethercat frame transmission time on network efficiency in case of asymmetric traffic," in Proc. 54th ETRAN Conf., June 2010, pp. 1–4.

[3] M. Knezic, B. Dokic, and Z. Ivanovic, "Increasing ethercat performance using frame size optimization algorithm," in Proc. IEEE 16th Conf. Emerging Technologies Factory Automation (ETFA), Sep. 2011, pp. 1–4.

[4] **M. Knezic** and B. Dokic, **"Performance aspects of the Switched EtherCAT networks," in Proc. 55th ETRAN Conf.**, June 2011, pp. 1–4 **(проглашен за најбољи рад младог аутора у секцији Електроника)**.

[5] M. Knezic, B. Dokic, and Z. Ivanovic, "Performance evaluation of the switched EtherCAT networks with VLAN tagging," Serbian Journal of Electrical Engineering, vol. 9, no. 1, pp. 33–42, 2012.

[6] M. Knezic, B. Dokic, and Z. Ivanovic, "A software tool for efficient configuration of ethercat networks," in Proc. IEEE 17th Int. Conf. Emerging Technologies Factory Automation (ETFA 2012), Sep. 2012, pp. 1–4.

[7] M. Knezic and Z. Ivanovic, "Evaluation of ethernet over ethercat protocol efficiency," in Proc. INFOTEH-JAHORINA, vol. 12, Mar. 2013, pp. 455–459.

[8] M. Knezic, B. Dokic, and Z. Ivanovic, "Performance analysis of the ethernet powerlink pollresponse chaining mechanism," in Proc. IEEE World Conf. Factory Communication Systems (WFCS), May 2015, pp. 1–4.

[9] **M. Knezic**, B. Dokic, and Z. Ivanovic, **"Theoretical and experimental evaluation of ethernet powerlink pollresponse chaining mechanism," IEEE Transactions on Industrial Informatics**, 2016.

VI.2. Критичност и коректност тумачења резултата

Комисија оцјењује да је кандидат добијене резултате правилно, логично и јасно тумачио, испољавајући при томе довољно критичности. Он прецизно идентификује проблеме из предметне области који нису ријешени или су недовољно истражени од стране других аутора. Кандидат коректно пореди резултате сопственог истраживања са резултатима других аутора.

VI.3. Нови истраживачки задаци

Кандидат у својој дисертацији истиче даљи правац истраживања у смјеру проширења предложених рјешења у циљу отклањања уочених недостатака. Тако, на примјер, он уочава да код *EtherCAT* протокола постоји простор за оптимизацију у случајевима када величина процесних података превазилази опсег једног *Ethernet* оквира максималне дужине. Такође, кандидат истиче да се оптималан избор броја портова мрежног преспојника, код приступа којим се смањује пропагационо кашњење *EtherCAT* мреже, може аналитички одредити на начин изложен у дисертацији само у случају хомогене мреже, код које сви уређаји имају исте карактеристике и исту величину процесних података. У том смислу, он предлаже да се, у будућем раду, формализује алгоритам који узима у обзир ефекте асиметрије у структури мреже и да се као резултат добије оптимална мрежна структура.

У случају *Ethernet Powerlink* протокола, кандидат указује да се ET-PRC механизам може унаприједити тако да омогући исправан рад и у случају губитка оквира, на начин да управљачки уређај континуално надзире комуникацију и обезбјеђује одговарајуће сигналне поруке које омогућавају наставак комуникације у случају грешке.

Код *Profinet IO* комуникације, кандидат примјећује да постоји простор за унапређење алгоритама који креирају временски план слања порука како у стандардном, тако и у начину рада са динамичким паковањем оквира. У том смислу, он указује на потребу анализирања постојећих алгоритама како би се идентификовао простор за евентуална унапређења.

VII ЗАКЉУЧАК И ПРИЈЕДЛОГ

Докторска дисертација мр Младена Кнежића под називом

ОПТИМИЗАЦИЈА ВРЕМЕНА ОДЗИВА ИНДУСТРИЈСКИХ КОМУНИКАЦИОНИХ МРЕЖА

садржи више оригиналних научних доприноса у овој области. Главни научни допринос се односи на моделовање и технике оптимизације времена одзива у индустријским комуникационим системима. Материја је обрађена на веома коректан научно утемељен начин. Предложени аналитички модели су провјерени рачунарским симулацијама и мјерењима. Теоретске поставке су експериментално потврђене. Реализовани су прототипови разматраних индустријских комуникационих мрежа. Резултати симулације и мјерења су представљени табелама и дијаграмима. Кандидат указује на ограничења предложених рјешења, као и на даље правце истраживања. Већи дио резултата истраживања већ је био изложен научној критичкој јавности. Прво, ова дисертација представља наставак истраживања који је кандидат започео у свом магистарском раду. Друго, послије магистарског рада, кандидат је први аутор седам објављених радова из ове области. Један од тих радова је објављен у међународном часопису са **SCI-E листе са фактором утицаја 4.708**. Један рад презентован на националној Конференцији ЕТРАН, проглашен је за најбољи рад младог аутора у секцији Електроника.

С обзиром на све што је изложено у овом Извјештају, Комисија са великим задовољством

П Р Е Д Л А Ж Е

Наставно-научном вијећу Електротехничког факултета и Сенату Универзитета у Бањој Луци да прихвати докторску дисертацију ОПТИМИЗАЦИЈА ВРЕМЕНА ОДЗИВА ИНДУСТРИЈСКИХ КОМУНИКАЦИОНИХ МРЕЖА мр Младена Кнежића и кандидату одобри јавну усмену одбрану.

Бања Лука, 04.03.2017. године

ПОТПИС ЧЛАНОВА КОМИСИЈЕ

1. Проф. др Предраг Петковић, предсједник
2. Проф. др Бранко Докић, ментор и члан
3. Проф. др Бранко Блануша, члан
4. Проф. др Гордана Гардашевић, члан
5. Доц. др Игор Крчмар, члан

Прилог 3.

Изјава 1

ИЗЈАВА О АУТОРСТВУ

Изјављујем
да је докторска дисертација

Наслов рада Оптимизација времена одзива индустријских комуникационих мрежа

Наслов рада на енглеском језику Response time optimization of industrial communication networks

- ☒ резултат сопственог истраживачког рада,
- ☒ да докторска дисертација, у цјелини или у дијеловима, није била предложена за добијање било које дипломе према студијским програмима других високошколских установа,
- ☒ да су резултати коректно наведени и
- ☒ да нисам кршио/ла ауторска права и користио интелектуалну својину других лица.

Потпис докторанта

У Бањој Луци, дана 31.03.2017. године

Младен Кнежић

_____

Изјава 2

Изјава којом се овлашћује Универзитет у Бањој Луци да докторску дисертацију учини јавно доступном

Овлашћујем Универзитет у Бањој Луци да моју докторску дисертацију под насловом Оптимизација времена одзива индустријских комуникационих мрежа

која је моје ауторско дјело, учини јавно доступном.

Докторску дисертацију са свим прилозима предао/ла сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у дигитални репозиторијум Универзитета у Бањој Луци могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (*Creative Commons*) за коју сам се одлучио/ла.

- ○ Ауторство
- ○ Ауторство – некомерцијално
- ○ Ауторство – некомерцијално – без прераде
- ○ Ауторство – некомерцијално – дијелити под истим условима
- ○ Ауторство – без прераде
- ◉ Ауторство – дијелити под истим условима

(Молимо да заокружите само једну од шест понуђених лиценци, кратак опис лиценци дат је на полеђини листа).

У Бањој Луци, дана 31.03.2017. године

Потпис докторанта
Младен Кнежић

Изјава 3

Изјава о идентичности штампане и електронске верзије докторске дисертације

Име и презиме аутора	Младен Кнежић
Наслов рада	Оптимизација времена одзива индустријских комуникационих мрежа
Ментор	проф. др Бранко Докић

Изјављујем да је штампана верзија моје докторске дисертације идентична електронској верзији коју сам предао/ла за дигитални репозиторијум Универзитета у Бањој Луци.

У Бањој Луци, дана 31.03.2017. године

Потпис докторанта
Младен Кнежић