

**ИМПЛЕМЕНТАЦИЈА ПОДРШКЕ ЗА ПРЕНОС ДАТОТЕКА ПОМОЋУ CANOPEN И HTTP ПРОТОКОЛА У ИНДУСТРИЈСКОМ ИНТЕРНЕТУ СТВАРИ****IMPLEMENTATION OF SUPPORT FOR DATA TRANSFER USING CANOPEN AND HTTP PROTOCOL IN INDUSTRIAL INTERNET OF THINGS**Саша Бојанић, Иван Каштелан, *Факултет техничких наука, Нови Сад***Област – РАЧУНАРСТВО И АУТОМАТИКА**

**Кратак садржај** – У овом раду описан је приједлог рјешења за пренос датотеке са дијагностичким подацима у оквиру једног IoT система који укључује машину са посредничким уређајем и удаљени сервер. Пренос унутар машине одвија се преко Controller Area Network (CAN) магистрала, односно помоћу CANOpen протокола, док се пренос до удаљеног сервера одвија помоћу Hypertext Transfer Protocol (HTTP) протокола. Приликом преноса датотеке до удаљеног сервера уочен је недостатак уколико је временско ограничење за пренос фиксно. Да би се недостатак превазишао, имплементирано је израчунавање временског ограничења на основу ширине пропусног опсега и величине датотеке. Рад укључује и анализу резултата израчунавања временског ограничења.

**Кључне речи:** *Индустријски интернет ствари, CANOpen, HTTP, посреднички уређај, Fog computing*

**Abstract** – This paper presents an implementation of the support for data transfer with diagnostic data within an IoT system that includes the fog device and cloud. Data transfer within the device is performed using the CAN bus, following the CANOpen protocol, while the data transfer to the cloud follows the HTTP protocol. During the data transfer, a deficiency is noticed if the time constraint is fixed. To overcome this deficiency, the calculation of the time constraint based on the bandwidth and file size is implemented. The paper includes the analysis of the time constraint calculation results.

**Keywords:** *Industrial internet of things, CANOpen, HTTP, fog device, fog computing*

**1. УВОД**

Развојем и усавршавањем рачунарских технологија, као и комуникације путем интернета, створени су услови за интеграцију разних електронских уређаја са циљем побољшања квалитета живота, унапређења сигурности, као и уштеде енергије. С тим у вези настао је појам под именом „Интернет ствари“ (енгл. *Internet of Things – IoT*). Интернет ствари представља велики број умрежених механичких, електричних и виртуелних уређаја који међусобно комуницирају и на тај начин формирају аутоматизоване системе

**НАПОМЕНА:**

Овај рад проистекао је из мастер рада чије ментор је био др Иван Каштелан, ванр. проф.

којима је потребна минимална или никаква интеракција са људима. Интернет ствари је пронашао широку примјену у индустрији, те се тако развила и засебна грана – индустријски интернет ствари. Уређаји у оквиру индустријског интернета ствари умрежени су у оквиру фабрика или машина. Интернет ствари у оваквим машинама нашао је примјену у виду сензора и актуатора, као и посредничких уређаја и сервиса облака који су задужени за прикупљање, праћење и анализу података из саме машине.

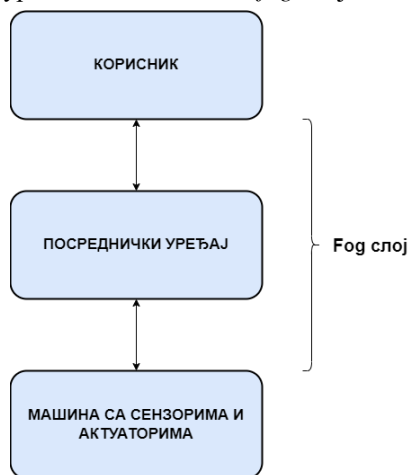
У овом раду је описана имплементација подршке за пренос података у индустријском интернету ствари. Циљ истраживања је остварење веће контролабилности периферних уређаја и стварање могућности за искоришћавање већих потенцијала које они посједују. Сама форма датотеке која учествује у преносу доприноси читљивости и прегледности и самим тим крајњем кориснику олакшава комуникацију, као и комплетан процес управљања машином, а израчунавање временског ограничења на основу ширине пропусног опсега обезбјеђује поузданост система, то јест даје својеврсну гаранцију да пренос датотеке неће бити прекинут због истека временског рока.

**2. КОНЦЕПТ КОМУНИКАЦИЈЕ У IoT СИСТЕМУ**

Основни концепт комуникације у оквиру интернета ствари заснован је на раду сензора и актуатора и њихове спреге са управљачком јединицом која може, али и не мора да буде под контролом корисника. Најчешће коришћен начин корисничке интеракције јесте путем система облака (енгл. *cloud*) који кориснику даје директан приступ машини у којој су смјештени сензори и актуатори.

Међутим, како је услед убрзаног развоја рачунарских технологија дошло до наглог повећања броја појединачних, али и читавих мрежа сензорских и актуаторских уређаја повезаних на интернет, а самим тим и до значајног пораста у количини генерисаних података, открили су се недостаци система облака. Због превелике количине података постало је немогуће да се сви подаци обраде, чиме су изазвана кашњења, преоптерећења мреже, то јест, систем је изгубио поузданост. Да би се превазишао овај проблем, у постојећи систем уведени су посреднички уређаји на којима се врши обрада, агрегација и селекција података за слање кориснику, то јест формира се међуслој који носи назив *fog* (енглеска

ријеч за маглу) [1]. На слици 1 приказана је архитектура система облака са fog слојем.



Слика 1. Резултати симулације

### 3. CANOPEN ПРОТОКОЛ

*Controller Area Network Open* (CANOpen) је комуникациони протокол заснован на CAN магистралаи, који помоћу сета стандарда дјелимично имплементира више нивое *Open Systems Interconnection* (OSI) модела. Уређаји у оквиру CANOpen протокола представљени су као чворови, а све информације везане за конфигурацију и функционалност чвора уписане су у табели која се назива рјечник објеката. Пренос података се одвија помоћу два главна комуникациона протокола: PDO (*Process Data Object*) и SDO (*Service Data Object*) [2].

#### 3.1. PDO протокол

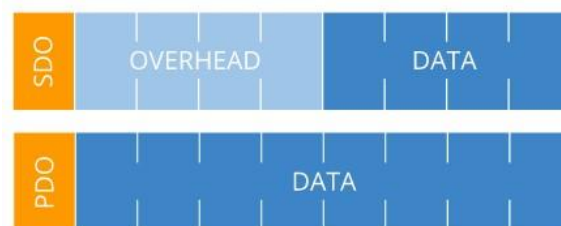
PDO протокол служи за дистрибуцију контролних и статусних информација високог приоритета, као и података који се преносе у реалном времену. Оквир се састоји од 8 бајтова чистих апликативних података. Погодан је за трансфер података који често мијењају вриједност и шаљу се периодично (подаци са сензора или подаци за специфичне актуаторе).

#### 3.2. SDO протокол

SDO протокол омогућава приступ свим уносима рјечника објеката и заснован је на принципу клијент-сервер комуникације са потврдом. Чвор чијем се рјечнику објеката приступа представља сервер, а чвор који приступа рјечнику објеката другог чвора представља клијента. За разлику од PDO протокола, оквир SDO протокола носи 4 бајта апликативних података, док остала 4 припадају заглављу. Разлика у садржају оквира PDO и SDO протокола приказана је на слици 2.

Постоје три врсте преноса података у оквиру SDO протокола: 1) убрзани – користи се када подаци могу да стану у једну поруку, 2) нормални – назива се још и сегментирани и користи се када подаци не могу да стану у једну поруку и 3) блок трансфер – оптимизовани режим преноса за уносе у рјечнику објеката који садрже велику количину података. Један

блок може да садржи до 889 бајтова (127 порука по 7 бајтова) и тек након трансфера читавог блока обавља се потврда о пријему.



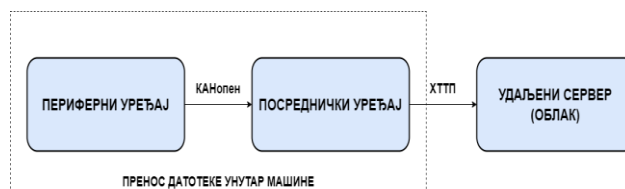
Слика 2. Оквири SDO и PDO протокола [3]

### 4. HTTP ПРОТОКОЛ

*Hypertext Transfer Protocol* (HTTP) протокол је један од најзаступљенијих и најпоузданијих протокола за комуникацију са сервисом облака. HTTP је протокол апликативног слоја који служи за пренос хипертекста, односно кода написаног помоћу *Hypertext Markup Language* (HTML) језика. Овај протокол ради на принципу захтјев-одговор. Клијентски захтјев се састоји из два дијела, а то су заглавље и тијело. У заглављу су садржане информације о верзији протокола, команда, име траженог документа итд. Тип захтјева дефинисан је командом односно наредбом. Неке од наредби су: *get*, *post*, *put*, *delete* итд. Одговор сервера се такође састоји из два дијела, а то су извјештај о статусу захтјева и сам садржај одговора. Извјештај о статусу је у форми троцифреног броја и кратког текста, док одговор има исту форму као и захтјев [4].

### 5. ПРЕНОС ДАТОТЕКЕ

Пренос датотеке је инициран од стране периферног уређаја који се налази у машини и повезан је CAN магистралом са посредничким уређајем. По пријему датотеке, посреднички уређај исту просљеђује удаљеном серверу и на тај начин датотека постаје доступна кориснику. Према томе, процес преноса датотеке може се подијелити у два дијела (слика 3): 1) дио за пренос датотеке унутар машине – пренос између периферног и посредничког уређаја и 2) дио за пренос датотеке од машине до удаљеног сервера.



Слика 3. Концепт рјешења

#### 5.1. Пренос датотеке унутар машине

Пренос датотеке унутар машине обавља се помоћу CANOpen протокола, односно помоћу блок трансфера који представља један од типова преноса у оквиру SDO протокола. Рјечник објеката посредничког уређаја конфигуриран је тако да омогући писање датотеке од стране периферног уређаја, а самим тим дефинишу се

улоге оба уређаја – посреднички уређај има улогу SDO сервера, а периферни уређај има улогу SDO клијента. Пренос датотеке почиње приступом клијента рјечнику објеката сервера. Прво се провјерава тренутни статус преноса и уколико је сервер доступан, то јест ако нема недовршеног преноса, уписује се име датотеке и започиње пренос по блоковима. У једном захтјеву могућ је пренос 127 блокова по 7 бајтова односно 889 бајтова података, након чега сервер шаље потврду клијенту. Наведена процедура се понавља све док се не пренесе читава датотека.

## 5.2. Пренос датотеке од машине до удаљеног сервера

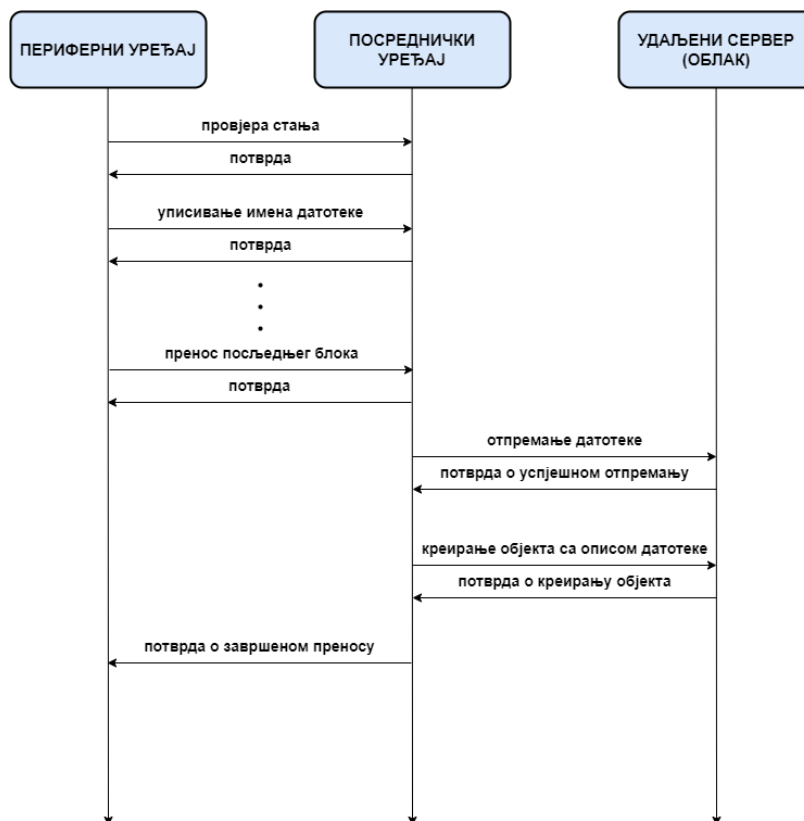
Процес преноса започиње уписом датотеке у директоријум. Након што је датотека у потпуности уписана, активира се механизам за провјеру и потенцијално просљеђивање датотеке на удаљени сервер. Механизам за провјеру обухвата провјеру величине датотеке, провјеру везе са удаљеним сервером и провјеру доступности CANOpen канала за пренос. Уколико су све провјере задовољене, процес преноса прелази у наредну фазу - израчунавање временског ограничења.

## 5.3. Израчунавање временског ограничења

Временско ограничење представља максималан период трајања отпремања датотеке, а његова вриједност добија се као количник величине датотеке и експериментално утврђеног коефицијента (1). Свака ширина пропусног опсега има одговарајући коефицијент, а његова вриједност дефинисана је као најнижа брзина преноса при оптималном раду комуникационог канала за слање података на интернет (енгл. *uplink*).

$$timeout = \frac{file\_size}{bandwidth\_coefficient} \quad (1)$$

Процес преноса наставља се отпремањем датотеке у репозиторијум који се налази на удаљеном серверу, а као одговор уређај добија адресу на којој се налази датотека. Након тога, креира се објекат који садржи информације о датотеци – име датотеке и адреса на којој се налази. Обје операције (пренос датотеке и креирање објекта) реализују се помоћу POST захтјева у оквиру HTTP протокола. Комплетан процес преноса приказан је дијаграмом комуникације на слици 4.



Слика 4. Дијаграм комуникације

## 6. РЕЗУЛТАТИ РАЧУНАЊА ВРЕМЕНСКОГ ОГРАНИЧЕЊА

Резултати тестова израчунавања временског ограничења на основу пропусног опсега помоћу наведених испитних случајева садржани су у табели 1. На основу резултата тестирања приказаних у табели 1, могуће је извести закључак о успјешном раду система. У свим испитним случајевима измјерено вријеме преноса мање је од израчунатог временског ограничења чиме је одговорено на захтјеве тестирања.

Међусобним поређењем резултата тестирања за *EDGE* и *UMTS* ширине пропусног опсега, могуће је уочити својеврсну аномалију. У сва три испитна случаја измјерено вријеме за *UMTS* ширину пропусног опсега је веће, иако ову ширину карактерише већа просјечна брзина од ширине *EDGE*. Овакви резултати настају због чињенице да је уређај који је слао датотеку у *EDGE* ширини пропусног опсега имао већу јачину сигнала. Дакле у процесу преноса датотеке осим фактора ширине пропусног опсега фигурише и

фактор јачине сигнала што је потврђено на основу поређења наведених испитних случајева.

Табела 1. Резултати рачунања временског ограничења

Величина Датотеке (Mb)	Ширина Пропусног Опсега	Изрaчунато Временско Ограничење (S)	Измјерено Вријеме Преноса (S)
1	EDGE	273	6.117
	UMTS	40	8.314
	LTE	8	6.147
10	EDGE	2730	28.906
	UMTS	409	36.945
	LTE	81	25.075
40	EDGE	10923	85.980
	UMTS	1638	135.021
	LTE	327	83.704

## 7. ЗАКЉУЧАК

Рад је показао да је могуће креирати поуздан систем за пренос дијагностичких података од периферног уређаја унутар машине до корисника који подацима приступа на сервису облака. За разлику од најчешће коришћене примјене CANOpen протокола за слање појединачних података, ово рјешење омогућава њихово слање у форми датотеке. На овај начин остварује се већа контролабилност периферних уређаја и ствара се могућност за искоришћавање већих потенцијала које они посједују. Сама форма датотеке доприноси читљивости и прегледности и самим тим крајњем кориснику олакшава комуникацију, као и комплетан процес управљања машином. Кључни резултати тестирања односе се на адекватно израчунато временско ограничење за пренос датотеке и на непромијењен садржај саме датотеке.

## 8. ЛИТЕРАТУРА

- [1] H. -J. Hong, "From Cloud Computing to Fog Computing: Unleash the Power of Edge and End Devices," 2017 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), Hong Kong, China, 2017, pp. 331-334, doi: 10.1109/CloudCom.2017.53
- [2] Olaf Pfeiffer, Andrew Ayre, Christian Keydel, *Embedded Networking with CAN and CANopen*, Copperhill Technologies Corporation, 2008.
- [3] *CSS Electronics CANLogger1000*: [canlogger1000.csselectronics.com/img/CANopen-PDO-SDO-Difference-Example-Message-Format-CAN-Bus.svg](http://canlogger1000.csselectronics.com/img/CANopen-PDO-SDO-Difference-Example-Message-Format-CAN-Bus.svg)
- [4] И. Башичевић, М. Поповић, В. Ковачевић, „Основи рачунарских мрежа 1“, ФТН, 2019.

### Кратка биографија:



**Саша Бојанић** рођен је 1996. године у Бањалуци, Република Српска, Босна и Херцеговина. Звање дипл. инж. електротехнике и рачунарства стекао на Факултету техничких наука у Новом Саду. На истом факултету, године 2021. уписује се на мастер академске студије на студијском програму Рачунарство и аутоматика. Запослен као Софтвер инжињер на институту РТ-РК у Бањалуци контакт: [sasa-bojanic@hotmail.com](mailto:sasa-bojanic@hotmail.com)



**Иван Каштелан** рођен је у Београду, 1985. године. Докторску дисертацију на Факултету техничких наука из области Рачунарска техника и рачунарске комуникације одбранио је 2014. године. контакт: [ivan.kastelan@uns.ac.rs](mailto:ivan.kastelan@uns.ac.rs)