

Edukativni primer korišćenja internet komunikacionih protokla i platformi baziranih na M2M elektronskim sistemima

Lenka Brestovački, Vladimir Rajs
Departman za energetiku, elektroniku i telekomunikacije
Fakultet tehničkih nauka Novi Sad
Novi Sad, Srbija
lenkabrestovacki@uns.ac.rs, vladimir@uns.ac.rs

Sažetak—Sa razvojem tehnologije postoji sve veća potreba za udaljenom kontrolom objekata. Jedan od načina udaljene kontrole objekata jeste Internet of things (IoT). Pojavom 4.0 industrijske revolucije javila se potreba i za obrazovanjem studenata u pravcu razvoja IoT sistema. Uz ograničene hardverske resurse i određene uslove koji su svakom studentu na raspolaganju osmišljen je sistem za edukativne svrhe koji će obuhvatiti sve osobine IoT sistema. U ovom radu je opisan način implementacije jednog takvog sistema sa minimalnim hardverskim resursima i njegova nadgradnja do realne primene.

Ključne reči—IoT; HTTP; MQTT; Node-red; ThingSpeak

I. UVOD

Uslovi izazvani pandemijom virusa COVID-19 postavili su visoke zahteve u pogledu organizovanja *online* nastave na fakultetima. Posebno je do izražaja došla potreba za udaljenim pristupom uređajima na fakultetu.

Povezivanje udaljenih uređaja preko mreže postalo je globalni trend budući da je omogućavalo beskontaktno obavljanje različitih zadataka.

Kao logična potreba javilo se obrazovanje studenata u smeru izgradnje *IoT* (eng. *Internet-of-Things*) sistema. Kako su studenti smera *Primenjena elektronika na Fakultetu tehničkih nauka Univerziteta u Novom Sadu* pretežno naklonjeni radu sa hardverom, bilo je potrebno osmisliti jednostavne obrade podataka koje se mogu izvršavati na objekatskoj (eng. *edge*) strani sistema, kao i jednostavne pristupe *IoT* platformama.

IoT sistemi omogućavaju potpuno automatizovano upravljanje uređajima, ali i uključivanje čoveka u upravljanje po potrebi. Pametni gradovi i pametne zgrade su danas najaktuelniji primeri *IoT* sistema. Automatizovan sistem osvetljenja čija je implementacija prikazana u ovom radu primenljiv je i u slučaju automatizovanog upravljanja, ali i u slučaju uključivanja čoveka u upravljanje po potrebi. Ovakva realizacija automatizovanog rešenja dodatno omogućava minimizaciju potrošnje energije budući da se osvetljenje uključuje samo onda kada je to potrebno.

II. STRUKTURA SISTEMA

A. *IoT* platforma

IoT predstavlja mrežu objekata (korisničkih aplikacija, računara, senzora i drugo) koji ostvaruju komunikaciju putem interneta [1]. Cilj je efikasna obrada podataka očitanih sa senzora i što jednostavniji prikaz rezultata korisniku. *IoT* platforma je internet servis namenjen za prikupljanje i grafičko prikazivanje podataka. U nastavi je korišćena *ThingSpeak* platforma [2] kojoj se pristupa jednostavnim *GET* i *PUT* metodama *HTTP* (eng. *Hypertext Transfer Protocol*) protokola. Ona obezbeđuje jednostavan API (eng. *Application Programming Interface*) putem koga se podaci na jednostavan način prosleđuju platformi. U *ThingSpeak* je moguće izvršavati *MATLAB* kod, samim tim i *online* analizu podataka u realnom vremenu. Često se koristi za izradu prototipova koji zahtevaju analitiku. Sa *ThingSpeak* platformom moguće je skladištiti i analizirati podatke bez konfigurisanja web servisa. Takođe se mogu kreirati sofisticirana obaveštenja putem elektronske pošte o događajima koji pristižu sa elektronskih uređaja povezanih na platformu.

B. Protokoli za komunikaciju između podsistema

Budući da se za razvoj sistema koriste *Raspberry Pi* i *ESP8266* koji koriste programski jezik *Python* kao jezik za razvoj softvera, a poznajući biblioteke ovog programskog jezika koje implementiraju jednostavne funkcije za komunikaciju putem *HTTP* protokola, u nastavi je najviše korišćen upravo ovaj protokol. *HTTP* je protokol koji počiva na klijent-server arhitekturi i pogodan je za učenje kod *IoT* sistema zbog jednostavnosti korišćenja. Neophodno je poznavati *API* odgovarajućeg servisa i korišćenje biblioteke *urllib* za funkcije *HTTP* protokola. Mana ovog protokola je što ne odgovara ideji stvarnog *IoT* sistema gde je potrebno da veliki broj uređaja komunicira ravnopravno. *HTTP* je korišćen za razmenu podataka između platformi *ThingSpeak* i *Node-red* zbog jednostavnosti implementacije *HTTP* metoda u *Node-red*-u.

MQTT (eng. *Message Queuing Telemetry Transport*) protokol počiva na *publish-subscribe* arhitekturi i time omogućava kačenje neograničenog broja uređaja na mrežu. Mana je što je osim navedenih servisa potrebno instalirati i broker koji će vršiti usmeravanje poruka. Jedan od uređaja postavlja određenu temu i na tu temu objavljuje poruke, uređaji kojima su ti podaci potrebni pretplate se na tu temu, a usmeravanje poruka vrši upravo broker (Sl. 1). Svaki od klijenata može da objavljuje poruke na određenu temu, ali i da se prijavi za primanje poruka sa neke druge teme. *MQTT* protokol je zbog svoje jednostavnosti najpogodniji protokol za korišćenje u *IoT* sistemima.

C. Node-red

Node-red je servis za kontrolu toka podataka i kreiranje jednostavnih interfejsa. Omogućava povezivanje hardverskih uređaja, *API*-ja i *online* servisa, kao sastavnih delova *IoT* sistema [3]. *Node-Red* obezbeđuje kreiranje *JavaScript* funkcija kroz editor tokova podataka baziran na web pregledaču. Kreirani tokovi se skladište u *JSON* formatu. *Node-red* je izabran kao jednostavan alat, koji zahteva minimalno poznavanje servisa za dozajiranje web stranica.

Node-red pruža online grafički editor za kreiranje aplikacija. Sadrži nodove - blokove koda za izvršavanje zadataka:

- čitanje i pisanje podataka u bazu,
- slanje podataka na *cloud*,
- čitanje podataka sa senzora, slanje komandi uređajima,
- uspostavljanje *MQTT* komunikacije sa različitim pločama (*Arduino*, *ESP32*...)
- kreiranje korisničkih grafičkih interfejsa,
- preuzimanje podataka sa web servisa pozivom odgovarajućih *API*-ja,
- tempiranje događaja...

U sklopu editora mogu biti pisane *JavaScript* funkcije koje obavljaju određene zadatke. Međusobnim povezivanjem nodova dobijaju se tokovi.

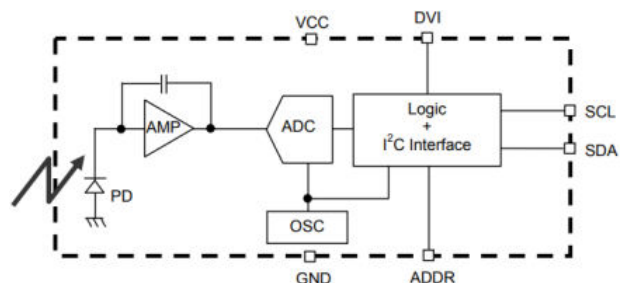
D. Raspberry Pi

U nastavi je korišćen *Raspberry Pi* različitih generacija, zbog skalabilnosti i jednostavnog pristupa njegovim pinovima putem *Node-red* servisa. *Node-red* omogućava jednostavnu akciju jer poseduje alate za konfigurisanje pinova *Raspberry Pi* računara u zavisnosti od toga da li su oni ulazni ili izlazni, da li koristimo logičke vrednosti ili impulsno-

širinsku modulaciju...

E. BH1750 senzor osvetljenosti

Senzor *BH1750* je digitalni senzor koji sa kontrolerom komunicira putem *I2C* komunikacije i za njegovo povezivanje potrebne su svega četiri linije. Senzor je jeftin i pristupačan, omogućava brzo merenje ambijentalnog osvetljenja. Nivo osvetljenosti se dobija kao 16-bitni broj zahvaljujući ugrađenom *AD* konvertoru. Ovaj senzor se najčešće koristi u mobilnim telefonima, *LCD* televizorima i digitalnim kamerama.



Slika 2. Blok šema BH1750 senzora [3]

Senzor *BH1750* čija je blok šema prikazana na slici 2 sadrži:

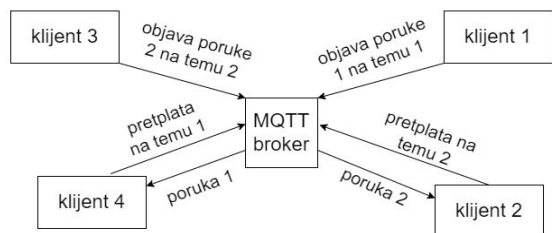
- Fotodiodu – reaguje na spoljašnje osvetljenje
- Pojačavač – pojačava napon sa diode
- *AD* konvertor – podatak sa senzora konvertuje u 16-bitnu vrednost
- Logički i *I2C* interfejs, uključujući registar podataka (skladištenje podataka o osvetljenosti) i vremenski registar (za setovanje merenja vremena)
- Interni oscilator – vrednost frekvencije oscilatora je tipično 320kHz

Proizvođač preporučuje da se koristi *H-resolution mod*. Vreme merenja u ovom modu je dovoljno dugačko da odbacuje šum frekvencije 50 Hz - 60 Hz. Rezolucija 1 lx je pogodna za merenje i malih osvetljenosti, manje od 10 lx [4].

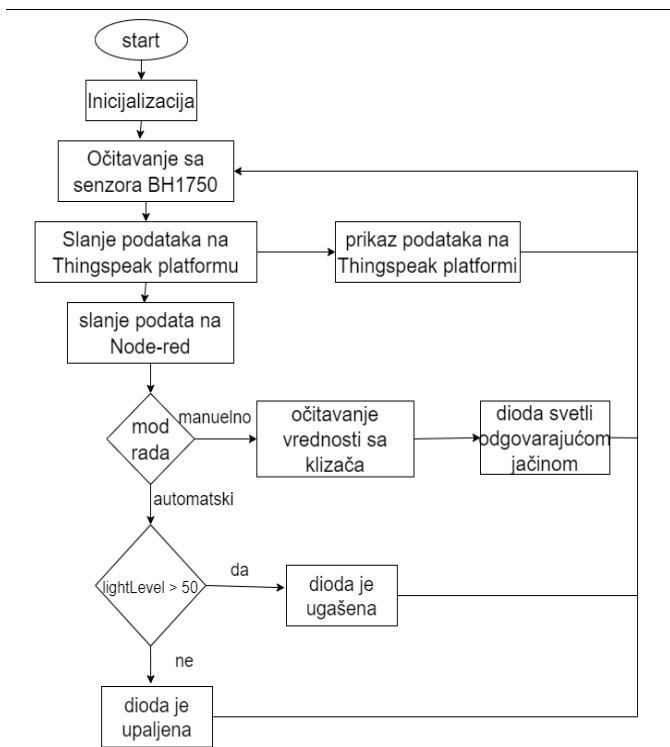
III. PRIMER AUTOMATIZOVANOG SISTEMA ZASNOVANOG NA M2M ELEKTRONSKIM SISTEMIMA

A. Pametna kuća: sistem za udaljeno upravljanje osvetljenjem u kući

Jedan od uspešno realizovanih projekata je implementacija sistema za udaljeno upravljanje osvetljenjem u kući. Algoritam rada prikazan je na sl. 3. Sistem je realizovan na *Raspberry Pi* platformi. Za merenje osvetljenja korišćen je senzor *BH1750*. Nakon iščitavanja podataka, oni se šalju na platformu *ThingSpeak* pomoću *request urlopen* biblioteke *urllib*. Kao parametar u funkciji koristi se *url* adresa u kojoj je sadržan *API Key* obezbeđen od strane *ThingSpeak*-a kako bi podaci bili prosledeni na odgovarajući kanal i u odgovarajuće polje. Na *ThingSpeak* platformi izabran je grafički prikaz.



Slika 1. MQTT protokol

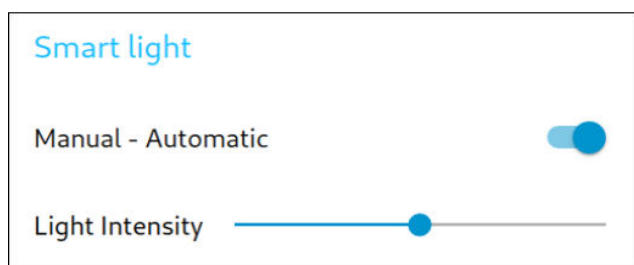


Slika 3. Algoritam rada sistema

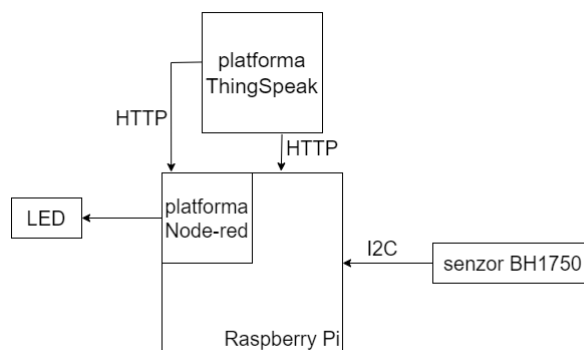
Ukoliko se za pregled izabere opcija *privatno* podaci koji pristižu sigurni su od nedozvoljenog pristupa stranih lica.

U razvojnom alatu *Node-red* kreiran je *dashboard* (Sl. 4) u kome se bira da li će se osvetljenje upravljati manuelno ili automatski. Ukoliko se osvetljenjem upravlja manuelno moguće je i podešavanje intenziteta uz pomoć klizača. Ukoliko se osvetljenjem upravlja automatski, na svakih 5 sekundi (podesiva veličina) šalje se *GET HTTP* zahtev sa *url*-om za iščitavanje koji sadrži odgovarajući *API Key* generisan od strane *ThingSpeak*-a. Pristigli podaci su string koji odgovara *JSON* objektu, zbog toga je potrebno pretvoriti ih u stvarni objekat i isparsirati odgovarajuću veličinu. Ako je jačina svetlosti manja od unapred definisanog praga, svetlo se automatski pali. Blok šema čitavog sistema prikazana je na Sl. 5.

Tok podataka prikazan je na Sl. 7. Ulaz predstavlja prekidač kojim se bira manuelno ili automatsko upravljanje. U slučaju manuelnog upravljanja sa klizača se uzima položaj i ta vrednost se prosleđuje priključku *Raspberry Pi* uređaja koji je konfigurisan da bude upravljani impulsno-širinski modulisanim vrednostima (eng. *Pulse-width modulation, pwm*). U slučaju automatskog upravljanja, generiše se signal koji se aktivira na svakih 5s. Ovo vreme je podesivo i izabrano je kao pogodno za reakciju u kućnim uslovima. Kada stigne signal šalje se *HTTP GET* zahtev čiji je atribut *url* u kome je sadržan *API Key* za čitanje sa *ThingSpeak* platforme. Pristigla poruka je



Slika 4. Izgled aplikativnog dela sistema kreiranog u *Node-red*

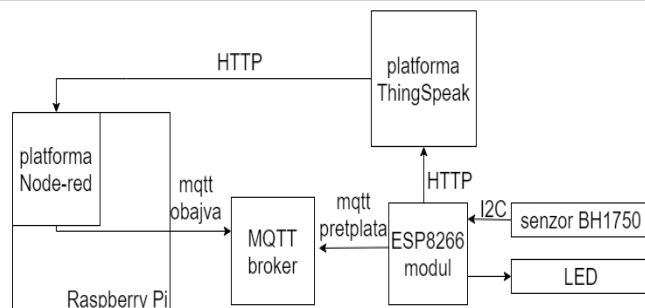


Slika 5. Blok šema prve verzije projekta

JSON objekat koji sadrži podatke o polju sa kog su očitani podaci, vrednost koja je očitana, vremenski trenutak u kom je očitana... Potrebno je isparsirati podatak o vrednosti i na osnovu toga da li je vrednost veća ili manja od praga na sledeći čvor proslediti 1 ako je veća i 0 ako je manja. Ova obrada vrši se *JavaScript* funkcijom koja je dobro podržana od strane *Node-red* alata. Nakon toga na priključak na koji je povezana LED se prosleđuje logička jedinica ili nula.

U prvoj verziji projekta, radi jednostavnosti ceo sistem je implementiran na *Raspberry Pi* platformi. *Node-red* alat omogućava direktno upravljanje pinovima logičkim vrednostima i *pwm* signalom. Ovakvo rešenje je centralizovano i stoga je potrebno da sva svetla budu direktno povezana sa *Raspberry Pi*-jem. To je praktično teško izvodljivo. Zbog čega je implementirana druga verzija (Sl. 6) u kojoj su senzori i svetla bili povezani na *ESP8266* modul koji je vršio očitavanje sa senzora i slanje podataka na *ThingSpeak* [5]. *Raspberry Pi* je korišćen kao server u tom slučaju, na kome je implementiran tok podataka u *Node-red* alatu. Ponovo je *GET* zahtevom *HTTP* protokola priman podatak o jačini osvetljenosti na svim sensorima. U koliko je jačina osvetljenosti manja od praga *Raspberry Pi* pomoću *MQTT* protokola objavljuje podatak na odgovarajuću temu. *ESP8266* se pretplati na tu temu i nakon pristizanja poruke vrši aktivaciju svetla [6][7]. Ukoliko je izabran manuelni mod na drugu temu objavljuje se poruka o stanju na *slider*-u i na osnovu toga se pali svetlo odgovarajućeg intenziteta.

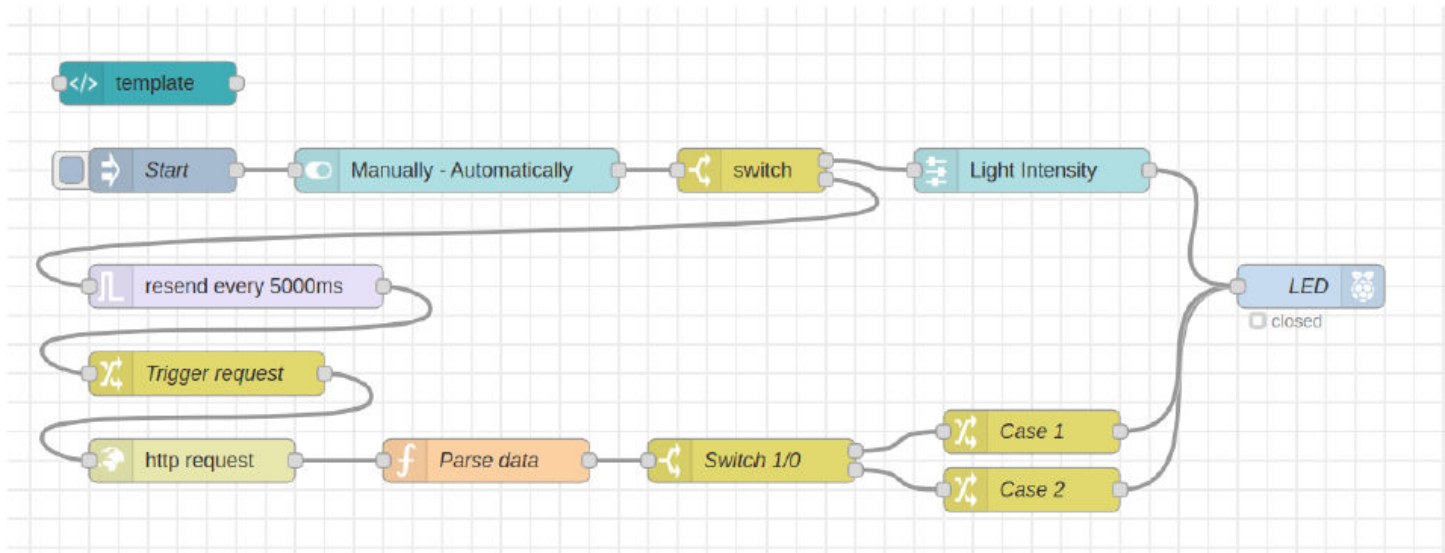
ThingSpeak platforma u ovom rešenju korišćena je kao *debug* platforma i u potpunosti se može zameniti prikazom podataka u okviru *Node-red* alata. Ovaj prikaz ima i kontrolnu svrhu budući da može da mu pristupi samo vlasnik kanala, ukoliko je njegov pregled postavljen na privatno. U složenijim primenama bila bi neophodna obrada podataka, koja kao što je rečeno može da se vrši u okviru *ThingSpeak* platforme i u tom slučaju ona bi bila neophodna za funkcionisanje sistema, a podaci bi se ka *Node-red* alatu prosleđivali na opisani način.



Slika 6. Blok šema druge verzije projekta

Kako je ovo edukativni primer, smatra se da je potpuna funkcionalnost iskorišćena.

Istraživanja u cilju unapređenja nastavnog procesa i razvoja naučno-stručnih oblasti Departmana za energetiku, elektroniku



Slika 7. Tok podataka kreiran u alatu Node-red

IV. ZAKLJUČAK

Sistem opisan u ovom projektu maksimalno pojednostavljuje hardverski deo zadatka čime se studenti upućuju da punu pažnju posvete komunikacijama i principima IoT sistema što je i cilj predmeta. Prikazano rešenje ima široku primenu u svakodnevnom životu. Kako *Node-red* alat sa standardnim podešavanjima funkcioniše samo u okviru iste mreže, obezbeđena je jednostavna zaštita podataka, jer ne postoji mogućnost da joj pristupi uređaj koji nije povezan na istu mrežu. *Node-red dashboard*-u može se pristupiti i sa druge mreže ukoliko se na oba uređaja omogući korišćenje VPN (*eng. Virtual Private Network*) [8]. Ovim bi se sačuvala privatnost, jer postoje servisi koji omogućavaju povezivanje samo unapred uparenih uređaja. Zainteresovani studenti upućuju se na proučavanje različitih mogućnosti povezivanja udaljenih uređaja. Moguće nadogradnje sistema u tom pogledu odnosile bi se na implementaciju *LoRa* (*eng. Long Range*) ili *NB-IoT* (*eng. Narrowband IoT*) radio signala. Kakos u student smera *Primenjena elektronika* već upoznati sa *GSM* mrežom, ovakva nadogradnja sistema je dobra prilika za uopštavanje njihovog znanja. Prilikom izlaska podataka na javnu mrežu, trebalo bi implementirati odgovarajući stepen zaštite [9]. To je još jedna od mogućnosti za dalje usavršavanje projekata.

ZAHVALNICA

Ovo istraživanje je podržano od strane Ministarstva prosvete, nauke i tehnološkog razvoja kroz projekat broj 451-03-68/202-314/ 200156 "Inovativna naučna i umetnička istraživanja iz domena delatnosti FTN-a

Ovaj rad je podržan od strane Fakulteta tehničkih nauka u Novom Sadu, Departmana za eneregтику, elektroniku i telekomunikacije u okviru realizacije projekta pod nazivom:"

i telekomunikacije.“

LITERATURA

- [1] Alexander S. Gillis, <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT> – 16.1.2023.
- [2] Uputstvo za upotrebu ThingSpeak platforme dostupno na adresi < <https://www.mathworks.com/help/thingspeak/>>, 3.3.2023.
- [3] Uputstvo za upotrebu Node-red razvojnog alata dostupno na adresi <https://nodered.org/docs/>, 16.1.2023.
- [4] Uputstvo za upotrebu senzora BH1750, <https://www.mouser.com/datasheet/2/348/bh1750fvi-e-186247.pdf>, 16.1.2023. K. Elissa, "Title of paper if known," unpublished.
- [5] N. Al-Husein, A.D. Salman, "IoT Monitoring System Based on MQTT Publisher/Subscriber Protocol", *Iraqi Journal of Computers, Communications, Control & Systems Engineering*, vol. 3, July 2020.
- [6] S.K. Selvaperumal, W. Al-Gumaei, R. Abdulla, V. Thiruchelvam "Integrated Wireless Monitoring System Using LoRa and Node/Red for University Building", *Journal of Computational and Theoretical Nanoscience*, vol. 16, August 2019.
- [7] T.A.Khoa, L.M.B.Nhu, H.H.Son, N.M.Trong, C.H.Phuc, N.T.H.Phuong, N.V.Dung, N.H.Nam, D.S.T.Chau, D.N.M.Duc, "Designing Efficient Smart Home Management with IoT Smart Lightening: A Case Study", *"Wireless Communications and Mobile Computing"*, vol.2020, November 2020.
- [8] S.Khanvilkar, A.Khokhar, "Virtual private networks: an overview with performance evaluation", *"IEEE Communications Magazine"*, vol. 42, October 2004
- [9] H.Lin, N.W.Bergman "IoT privacy and Security Challenges for smart home environments", *"Information"*, vol. 7, July 2016.

ABSTRACT

Due to the development of technology, there is an increasing need for remote control of facilities. This is what the Internet of Things (IoT) enables us to do. There was also a need to educate students in the direction of IoT systems. In the limited hardware conditions available to every student, it is necessary to design a system that will include all the principles of the IoT system. This paper describes how to implement one such work with minimal hardware resources and its development into a real application. There was implemented a system for remotely controlled house lighting using the *Raspberry Pi* platform, *BH1750* sensor, *ThingSpeak IoT* platform, and *Node-red* development tool. In the improved

version of the system, there was the *Raspberry Pi* platform as a server, and sensors were connected to the *ESP8266* module. The system uses *HTTP* and *MQTT* protocols. The authors suggest improvements to the system by using *LoRaWAN* and *NB-IoT* protocols and cyber security systems.

AN EDUCATIONAL EXAMPLE OF USING INTERNET COMMUNICATION PROTOCOLS AND PLATFORMS BASED ON M2M ELECTRONIC SYSTEMS

Lenka Brestovački, Vladimir Rajs