

Nadzor industrijskog procesa pomoću PLC-a opremljenog modulom Siemens IOT2040

Damjanović, Davor

Master's thesis / Diplomski rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:741591>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-07-16**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA**

Sveučilišni studij

**NADZOR INDUSTRIJSKOG PROCESA POMOĆU PLC-a
OPREMLJENOG MODULOM SIEMENS IOT2040**

Diplomski rad

Davor Damjanović

Osijek, 2020.

SADRŽAJ

| | |
|---------------------------------------------------------------------------------------------------------|----|
| 1. UVOD | 1 |
| 2. POJMOVI I TEHNOLOGIJE | 2 |
| 2.1 Internet stvari | 2 |
| 2.2 Računarstvo u oblak | 2 |
| 2.3 Industrija 4.0..... | 3 |
| 3. MAKETA I MODULI..... | 4 |
| 3.1 Siemens IOT2040..... | 4 |
| 3.2 Maketa održavanja temperature i razine tekućine u spremniku..... | 6 |
| 4. PROGRAMI UPOTRIJEBLJENI U RADU | 10 |
| 4.1 Node-RED | 10 |
| 4.2 MySQL baza podataka | 12 |
| 5. PROGRAMSKO RJEŠENJE | 13 |
| 5.1 Programsko rješenje na Siemens IOT2040 mrežnom modulu..... | 13 |
| 5.1.1 Mrežne postavke blokova za komunikaciju Siemens IOT2040 sa S7 PLC-om i MySQL bazom podataka | 14 |
| 5.1.2 Čitanje podataka iz memorije Siemens S7 – 1500 PLC-a i spremanje u MySQL bazu podataka..... | 19 |
| 5.1.3 Čitanje vrijednosti iz MySQL baze podataka i spremanje u memoriju Siemens S7 – 1500 PLC-a..... | 22 |
| 5.2 Programsko rješenje na serveru | 27 |
| 5.2.1 MySQL baza podataka | 27 |
| 5.2.2 Dohvaćanje podataka iz baze i njihov prikaz | 30 |
| 5.2.3 Upravljanje referentnim veličinama..... | 37 |
| 5.2.4 Postavljanje parametara procesa ovisno o dobu dana..... | 41 |
| 6. ZAKLJUČAK..... | 43 |
| LITERATURA | 44 |
| SAŽETAK..... | 45 |
| ABSTRACT | 45 |
| ŽIVOTOPIS..... | 46 |
| PRILOZI..... | 47 |

1. UVOD

Industrijska automatizacija sve više koristi programirljive logičke kontrolere (*eng. PLC-Programmable logic controller*) koji sadrže module koji omogućuju izgradnju rješenja koja se mogu svrstati u paradigmu internet objekata (*eng. Internet of Things*). Jedan takav inteligentni mrežni modul je Siemensov IOT2040 koji standardizira komunikaciju između različitih podatkovnih izvora te ih po potrebi distribuira različitim klijentima. Na taj način modul je moguće koristiti za udaljeni nadzor procesa i komunikaciju sa servisima u oblaku ili kao gateway između industrijske razine i razine upravljanja nekog poduzeća.

U ovom diplomskom radu prikazano je upravljanje i nadzor udaljenog postrojenja pomoću mrežnog modula Siemens IOT2040 i servera koji se može izvoditi na računalu ili na oblaku (*eng. Cloud*). Mrežni modul Siemens IOT2040 prosljeđuje podatke s PLC-a koji se koristi na nižoj razini upravljanja (koji upravlja sensorima i aktuatorima) te ih obrađuje i sprema u bazu podataka na serveru. Također Siemens IOT2040 čita podatke iz baze podataka sa servera i mijenja vrijednosti na PLC-u na nižoj razini kojim se upravlja. Aplikacija na serveru čita vrijednosti iz baze podataka, prikazuje ih i omogućava mijenjanje pojedinih varijabli procesa. Izmijenjene varijable aplikacija na serveru sprema u bazu podataka nakon čega ih s baze čita Siemens IOT2040 i na kraju Siemens IOT2040 izmijenjene varijable prosljeđuje PLC-u na nižoj razini.

Rad se dijeli na nekoliko poglavlja:

- U drugom su poglavlju opisane tehnologije i pojmovi upotrijebljeni u ovom diplomskom radu
- U trećem su poglavlju opisani moduli i maketa korišteni za izvršavanje nadzora i upravljanja
- U četvrtom poglavlju su opisani programi i način rada programa korištenih za izvršavanje zadatka u ovom diplomskom radu
- U petom poglavlju opisano je programsko rješenje na Siemens IOT2040 mrežnom modulu i serveru

2. POJMOVI I TEHNOLOGIJE

U ovom poglavlju opisane su tehnologije i pojmovi vezani za diplomski rad, njihova primjena, prednosti i manje.

2.1 Internet stvari

U današnjem svijetu sve je popularnije korištenje interneta stvari odnosno IoT(*eng. Internet of Things*). Internet stvari ili internet objekata je naziv za komunikaciju i upravljanje raznim uređajima iz svakodnevnog života preko interneta. Internet stvari pojednostavljuje i omogućava lakše korištenje raznih uređaja, omogućava upravljanje uređajima na daljinu čak i kada se nalazimo na drugom kraju svijeta. Internet stvari uz sve svoje prednosti otvara sigurnosni problem jer se uređaji koji prije nisu imali mogućnost spajanja sada spajaju na mrežu i upravljaju preko mreže te se tako otvara mogućnost da osobe s određenim znanjima mogu ući u takav sustav i na primjer, predstaviti se kao vlasnik kuće i otvoriti vrata garaže.

Neki od primjera interneta stvari su: automatizacija kuće, robotski usisivači, pametni televizori, pametni hladnjaci. Preko interneta možemo na primjer pokrenuti robotski usisivač da krene čistiti ili mu postaviti da čisti u određeno doba dana.

2.2 Računarstvo u oblak

Računarstvo u oblaku (*eng. Cloud computing*) se vrlo često koristi uz internet stvari. Računarstvo u oblaku omogućava osobama ili tvrtkama iznajmljivanje računalnih resurse, aplikacija, baza podataka i/ili nadgledanje postrojenja putem interneta. Najčešći model koji davatelji usluga omogućavaju jest „plati koliko koristiš“ (*eng. „Pay as you go“*).

Glavne podjele računarstva u oblaku su:

- Infrastruktura kao usluga (IaaS – *eng. Infrastructure as a service*) – usluge u oblaku (*eng. Cloud*) koje obuhvaćaju iznajmljivanje virtualnih strojeva i sučelja za programiranje aplikacija visokog nivoa, servera, pohrane podataka i slično.
- Platforma kao usluga (PaaS – *eng. Platform as a service*) – usluge u oblaku koje obuhvaćaju iznajmljivanje baza podataka, web servera, izvođenje programa. Omogućava

korisnicima da upravljaju, pokrenu i razvijaju programe bez da imaju vlastitu infrastrukturu.

- Aplikacija kao usluga (SaaS – *eng. Software as a service*) – usluge u oblaku koje obuhvaćaju iznajmljivanje aplikacija na zahtjev (*eng. Software on demand*). Radi na principu pretplate gdje korisnici plaćaju i nakon toga mogu pristupiti aplikaciji putem interneta koja se izvodi na udaljenom serveru.

2.3 Industrija 4.0

Kako je vrijeme prolazilo tako su uređaji interneta stvari postajali sve pouzdaniji. Vidjelo se da se internet stvari može primijeniti i za ozbiljnije primjene zadržavajući svoje svojstvo lakšeg korištenja i korištenja na daljinu. Tako se internet stvari polagano počeo širiti i na industriju. Primjena interneta stvari u industriji se naziva Industrijom 4.0. Industrija 4.0 ili kako je mnogi zovu 4. industrijska revolucija svojim nazivom obuhvaća:

- digitalizaciju industrije koja obuhvaća povezanost strojeva, skladišta, logistike i opreme,
- integraciju - vertikalnu koja je unutar sektora organizacije i horizontalnu koja se odnosi na poslovne partnere i dobavne lance,
- pametne strojeve s mogućnošću neovisne razmjene informacija,
- pametne tvornice, kontrolu i optimizaciju proizvodnje u realnom vremenu.[1]

Kao i svaka promjena tako i prijelaz na industriju 4.0 ima svoje prednosti i mana.

Prednosti industrije 4.0 prate prednosti interneta stvari kao što su lakše i efikasnije upravljanjem industrijskim procesima i pregled stanja cijelog postrojenja bez obzira na trenutnu lokaciju, veću konkurentnost na tržištu, prilagodljivu proizvodnju i smanjenje pritiska na radnike.

Jedan od najvećih nedostataka industrije 4.0 je što sustavi s izvanmrežnim (*eng. Offline*) načinom rada se počinju spajati na internet i tako se izlažu opasnostima interneta te se moraju ulagati dodatni resursi za zaštitu podataka i sigurnosti protiv upada u sustav.

3. MAKETA I MODULI

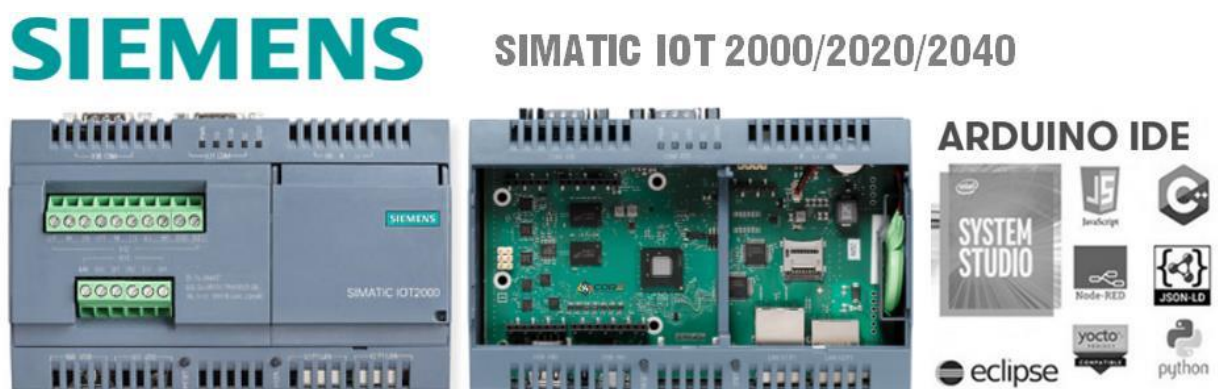
U ovom poglavlju opisana je maketa koju se nadgleda i kojom se upravlja, kao i opis mrežnog modula Siemens IOT2040 koji se koristi za nadgledanje i upravljanje procesa na višoj razini.

3.1 Siemens IOT2040

Siemens IOT2040 (prikazan na slici 3.1) je mrežni modul koji ima svojstvo da se može povezati na internet preko kojeg se može njime upravljati odnosno mijenjati parametre procesa. Također Siemens IOT2040 može spremati podatke na internet (baze podataka) koji se mogu nadzirati na daljinu. Ovaj mrežni modul je moguće programirati uz pomoć Eclipse IDE, Arduino IDE, Python i C++ programskih jezika te Node-RED-a.

Siemens IOT2040 sadrži:

- Intel Quark X1020 procesor,
- 1GB RAM,
- 2 Ethernet sučelja
- 2 RS232/422/485 sučelja
- Sat u stvarnom vremenu s baterijom[2]



Slika 3.1 Siemens IOT2000/2020/2040 [3]

U ovom diplomskom radu upotrijebljen je Siemens IOT2040 iz porodice IOT2000 (prikazan na slici 3.2). Ovaj mrežni modul ima svojstva koja su potrebna za rješavanje zadatka ovoga diplomskog rada.



Slika 3.2 Siemens IOT2040 na kojemu je napravljen zadatak

Siemens IOT2040 mrežni modul koristi se kao dodatak koji omogućava sekundarno sučelje za nadgledanje i upravljanje procesom s udaljene lokacije uz primarno sučelje koje se nalazi u postrojenju.

Iako bi se sučelje i baza podataka mogli izvoditi i na Siemens IOT2040 mrežnom modulu to se izbjegava iz nekoliko razloga:

- Relativno slab procesor koji ima problema s prikazom kompliciranijih grafičkih prikaza
- Količina memorije potrebna za spremanje dugoročnih podataka izvođenja procesa u bazu podataka koja se nalazi na mrežnom modulu je vrlo mala

- Brža degradacija i propadanje SD kartice koja se nalazi u mrežnom modulu zbog konstantnih zahtjeva za pisanje

3.2 Maketa održavanja temperature i razine tekućine u spremniku

Maketa na kojoj se održava temperatura i razina tekućine u spremniku prikazana na slici 3.3 sastoji se od nekoliko senzora i aktuatora kojima upravlja PLC Siemens S7-1500 serije. Na slici 3.4 prikazana je maketa održavanja temperature i razine tekućine u spremniku spojena s razvodnom pločom koja sadrži Siemens S7 – 1500 PLC.

Senzori koji se upotrebljavaju su:

- Senzor tlaka - koji se nalazi u spremniku. Pomoću njega se mjeri razina tekućine u spremniku kojeg se regulira. Mjeri razinu tekućine na osnovu pritiska.
- Senzor razine – mehanički mjeri razinu tekućine u drugom spremniku kojega se ne regulira. Služi za sigurnost ako nestane tekućine u spremniku prekida rad pumpe i ventila.
- Senzor temperature – mjeri temperaturu tekućine u spremniku
- Senzor protoka – mjeri brzinu protoka tekućine iz jednog spremnika u drugi

Aktuatora koji se upotrebljavaju su:

- Grijač – grije tekućinu dok se ne postigne tražena temperatura
- Pumpa – pumpa tekućinu iz jednog spremnika u drugi s reguliranim protokom
- Servo ventil – pomaže pri regulaciji protoka tekućine

PLC Siemens S7-1500 – programirajući logički sklop koji održava temperaturu i razinu tekućinu u spremniku pomoću navedenih senzora i aktuatora. Omogućena mu je promjena referentne temperatura i razine tekućine (tj. upravljane veličine) preko vanjskog zahtjeva putem mreže. Veličine koje se mjere te prosljeđuju na mrežu su: protok tekućine, temperatura tekućine, razina tekućine i otvorenost ventila. Na slici 3.5 prikazan je razvodna ploča sa Siemens S7 – 1500 PLC-om za maketu održavanja temperature i razine tekućine u spremniku

Za rješavanje zadatka ovog diplomskog primijenjen je jedna blok podataka na Siemens S7 – 1500 PLC-u koja je služila za izmjenu podataka između Siemens S7 – 1500 PLC-a i Siemens

IOT2040 mrežnog modula. Blok podataka na Siemens S7 – 1500 PLC-u prikazana je na slici 3.6.

Siemens S7 -1500 može direktno spremati podatke u bazu podataka te za njega Siemens IOT2040 je zapravo nepotreban, ali za PLC-ove starijih generacija koji ne mogu komunicirati s bazom podataka ugradnja IOT2000/2020/2040 je najjednostavniji i najjeftiniji način nadogradnje sustava i prilagodbu na industriju 4.0.



Slika 3.3 Maketa održavanja temperature i razine tekućine u spremniku



Slika 3.4 Maketa održavanja temperature i razine tekućine u spremniku s razvodnom pločom



Slika 3.5 Razvodna ploča sa Siemens S7 – 1500 PLC-om za maketu održavanja temperature i razine tekućine u spremniku

Diplomski_Rad ▸ PLC [CPU 1513-1 PN] ▸ Program blocks ▸ IOT2000 ▸ IOT2000DB [DB691]

Keep actual values Snapshot Copy snapshots to start values Load start values as actual values

IOT2000DB

| | Name | Data type | Offset | Start value | Retain | Accessible f... | Writa... | Visible in ... | Setpoint | Supervis... | Comment |
|----|------------------|--------------------|--------|-------------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|-------------|---------|
| 1 | Static | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | |
| 2 | FROM PLC | Struct | 0.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 3 | InputFlow | Real | 0.0 | 0.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 4 | FluidLevel | Int | 4.0 | 0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 5 | FluidTemperature | Real | 6.0 | 0.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 6 | ValveOpennes | Int | 10.0 | 0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 7 | Spare int | Array[0..3] of Int | 12.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 8 | TO PLC | Struct | 20.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 9 | SetLevel | Int | 20.0 | 0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 10 | SetTemperature | Real | 22.0 | 0.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 11 | Spare int | Array[0..5] of Int | 26.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |

Slika 3.6 Blok podataka na Siemens S7 -1500 PLC-u za izmjenu podataka sa Siemens IOT2040 mrežnim modulom

4. PROGRAMI UPOTRIJEBLJENI U RADU

U ovome poglavlju su opisani programi upotrijebljeni za realizaciju nadzora i upravljanja procesa

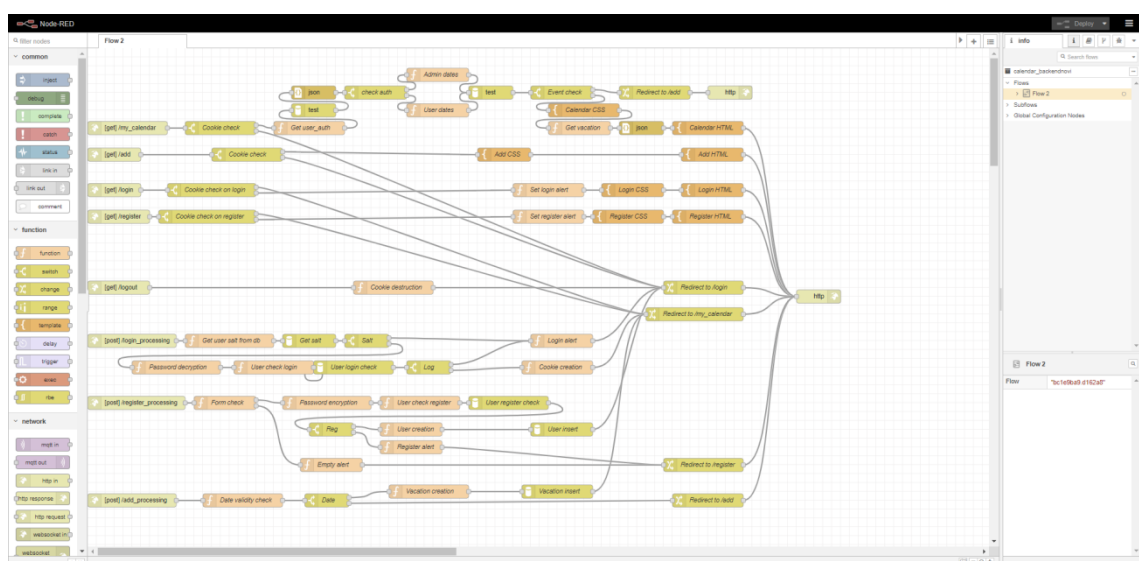
4.1 Node-RED

Node-RED je program izgrađen na platformi *Node.js*-a koji ne zahtijeva mnogo računalnih resursa za izvođenje, omogućava:

- spajanje raznih uređaja u mrežu,
- obradu podataka,
- izradu interaktivnog grafičkog sučelja,
- jednostavno korištenje sa serverima u oblaku.

Pristupa mu se putem web preglednika što omogućava programiranje i primjena na daljinu. Koristi tokove, blokovsko programiranje i JavaScript za programiranje koji omogućavaju vrlo jednostavno i intuitivno programiranje i preglednost napisanog koda. Svaki blok u kodu se naziva čvor (*eng. Node*) i izvodi određenu radnju koja mu piše u opisu.

Prebacivanje tokova s jednog uređaja na drugi je vrlo jednostavno zbog korištenja *.json* formata koji se može izvesti (generiranjem *.json* datoteke u kojoj se nalaze sve značajke jednog toka) s jednog uređaja koji koristi Node-RED te ga potom uvesti (učitati *.json* datoteku) na drugi uređaj koji koristi Node-RED. Na slici 4.1 prikazan je primjer jednog toka u Node-RED-u. [4]

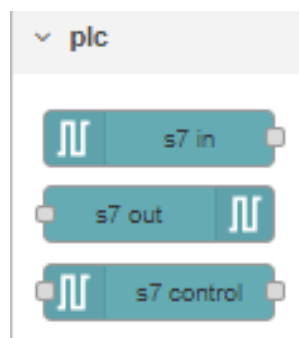


Slika 4.1 Primjer jednog toka u Node-RED-u

Uz standardne blokove za programiranje koji se nalaze u Node-RED-u na IOT2040 dodani su blokovi (u programu paleta *eng.pallet*) za komunikaciju sa Siemens S7 PLC-ovima (ime dodane palete: *node-red-contrib-s7*):

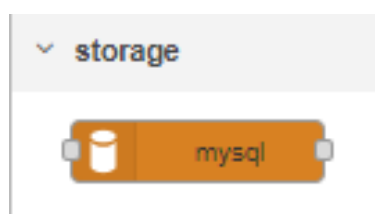
- *s7 in* – omogućava čitanje vrijednosti iz memorije PLC-a,
- *s7 out* – omogućava pisanje vrijednosti u memoriju PLC-a i
- *s7 control* – omogućava napredne kontrole nad PLC-om i mrežom između PLC-a i Node-RED uređaja.

Navedeni blokovi prikazani na slici 4.2 omogućavaju slanje i primanje vrijednosti i kontrolu Siemens S7 PLC-ova preko Node-RED uređaja (u ovome slučaju mrežnom modulu Siemens IOT2040).



Slika 4.2 Blokovi u Node-RED-u za komunikaciju s PLC-om

Na IOT2040 mrežnom modulu i serveru koji izvode Node-RED dodan je blok koji omogućava korištenje MySQL baze podataka (ime dodane palete: *node-red-node-mysql*) putem ip adrese ili stranice. Na slici 4.3 prikazan je izgled bloka koji omogućava korištenje MySQL baze podataka. S lijeve strane na ulaz u blok se pošalje SQL upit na bazu podataka. S desne strane se dobije traženi rezultat SQL upita.

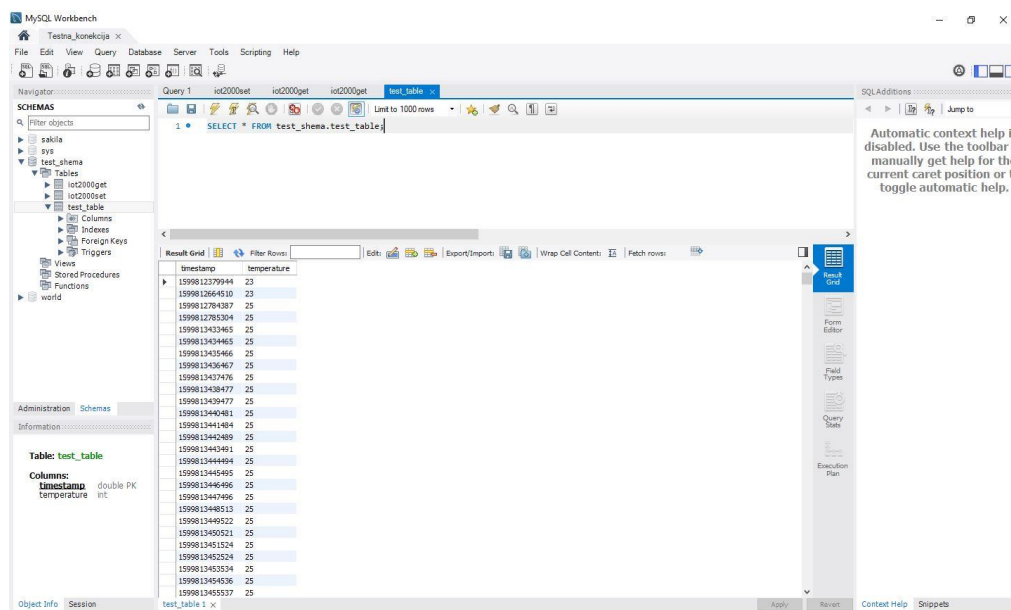


Slika 4.3 Blok za pristup bazi podataka

4.2 MySQL baza podataka

MySQL baza podataka je vrlo popularna baza podataka koja se koristi u mrežnim aplikacijama. MySQL baza podataka je relacijska baza podataka sa SQL (*eng. Structured Query Language*) upitima na bazu odnosno koristi strukturirani upitni jezik za rad s bazom podataka. Ima strukturirane zahtjeve na bazu poput stvaranja, brisanje ili izmjene tablica (*eng. Table*) u bazi, unosa, brisanja ili izmjene podataka unutar tablica (redaka tablice), stvaranje i brisanje okidača (*eng. Trigger*) koji se pokreću prilikom zadanih događaja u bazi podataka. Baza je relacijska što znači da tablice između sebe mogu biti povezane. Tablica u bazi može imati primarni ključ koji je jedinstven za svaki novi unos u tablicu (osim kada se rade izmjene na postojeći unos). Primarni ključ se postavlja na stupac u kojemu je sigurno da svaki unos ima drugačiji primarni ključ (npr. OIB osobe), ako je to nemoguće onda se najčešće stavlja ID koji kreće od 1 i nastavlja se povećavati za 1 svakim novim unosom. Tablica može imati strani ključ (*eng. Foreign key*) koji je zapravo primarni ključ neke druge tablice. Na taj način se podaci unutar baze podataka mogu povezati.

MySQL Workbench je sustav za upravljanje relacijskom bazom podataka (MySQL bazom podataka) koji omogućava postavljanje servera kojega se može dohvatiti lokalno ili putem ip adrese. Omogućava jednostavan pregled, uređivanje i korištenje MySQL baze podataka te promjene mrežnih postavki MySQL baze podataka na serveru. Na slici 4.4 je prikazan izgled MySQL Workbench-a.



Slika 4.4 Izgled MySQL Workbench-a

5. PROGRAMSKO RJEŠENJE

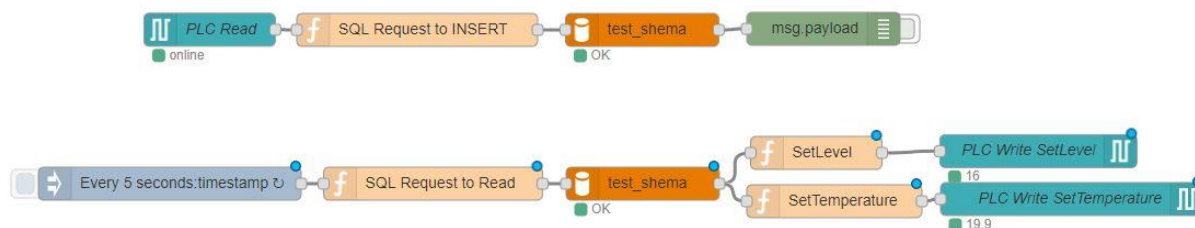
U ovom poglavlju opisano je programsko rješenje zadatka diplomskog rada. Rješenje se sastoji od dva dijela. Jedno je rješenje na Siemens IOT2040 mrežnom modulu, a drugo rješenje je na serveru koji može biti računalo ili usluga na oblaku. U ovome slučaju server se nalazi na računalu.

5.1 Programsko rješenje na Siemens IOT2040 mrežnom modulu

Zadatak Siemens IOT2040 mrežnog modula je pročitati podatke iz memorije Siemens S7 -1500 PLC-a, obraditi ih da se mogu spremiti u bazu podataka i na kraju poslati SQL upit na bazu podataka koja se nalazi na serveru. SQL upitom se šalju sve očitane vrijednosti sa Siemens S7 – 1500 PLC-a u bazu podataka svakih 5 sekundi.

Drugi zadatak Siemens IOT2040 mrežnog modula je očitati promjene traženih upravljanih veličina(*eng. Target value*) spremljenih u bazi podataka te ih poslati u memoriju Siemens S7 – 1500 PLC-a.

Oba zadatka su riješena pomoću Node-RED-a. Node-RED programsko rješenje na IOT2040 mrežnom modulu prikazano je na slici 5.1.



Slika 5.1 Prikaz programskog rješenja zadatka u Node-Red-u, čitanje iz memorije Siemens S7 – 1500 PLC-a i spremanje pročitanih vrijednosti u MySQL bazu podataka i obratno

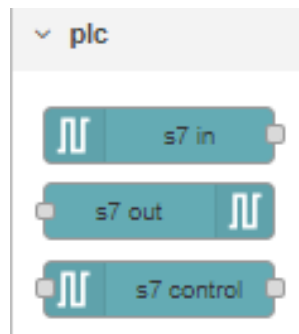
Siemens S7 – 1500 PLC na sebe ima blok podataka predviđenu za slanje i primanje podataka kojoj pristupa Siemens IOT2040 mrežni modul. Baza podataka na Siemens S7 – 1500 PLC-u prikazana je na slici 5.2.

| | Name | Data type | Offset | Start value | Retain | Accessible f... | Writa... | Visible in ... | Setpoint | Supervis... | Comment |
|----|------------------|--------------------|--------|-------------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|-------------|---------|
| 1 | Static | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | |
| 2 | FROM PLC | Struct | 0.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 3 | InputFlow | Real | 0.0 | 0.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 4 | FluidLevel | Int | 4.0 | 0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 5 | FluidTemperature | Real | 6.0 | 0.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 6 | ValveOpennes | Int | 10.0 | 0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 7 | Spare int | Array[0..3] of Int | 12.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 8 | TO PLC | Struct | 20.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 9 | SetLevel | Int | 20.0 | 0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 10 | SetTemperature | Real | 22.0 | 0.0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |
| 11 | Spare int | Array[0..5] of Int | 26.0 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | | |

Slika 5.2 Blok podataka na Siemens S7 -1500 PLC-u za izmjenu podataka sa Siemens IOT2040 mrežnim modulom

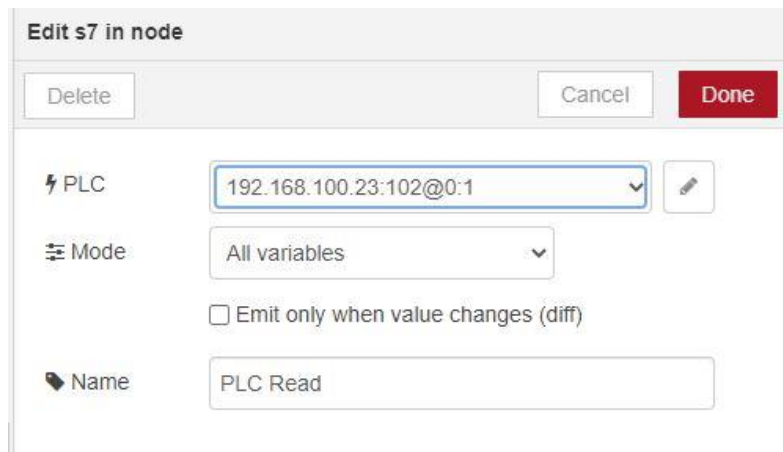
5.1.1 Mrežne postavke blokova za komunikaciju Siemens IOT2040 sa S7 PLC-om i MySQL bazom podataka

U Node-RED-u dodana je paleta koja omogućuje komunikaciju sa Siemens S7 PLC-ovima. Kako bi se omogućila komunikacija moraju se upisati tražene vrijednosti u opis blokova. Paleta u Node-RED-u za komunikaciju sa Siemens S7 PLC-ovima prikazana je na slici 5.3.



Slika 5.3 Blokovi u Node-RED-u za komunikaciju s PLC-om

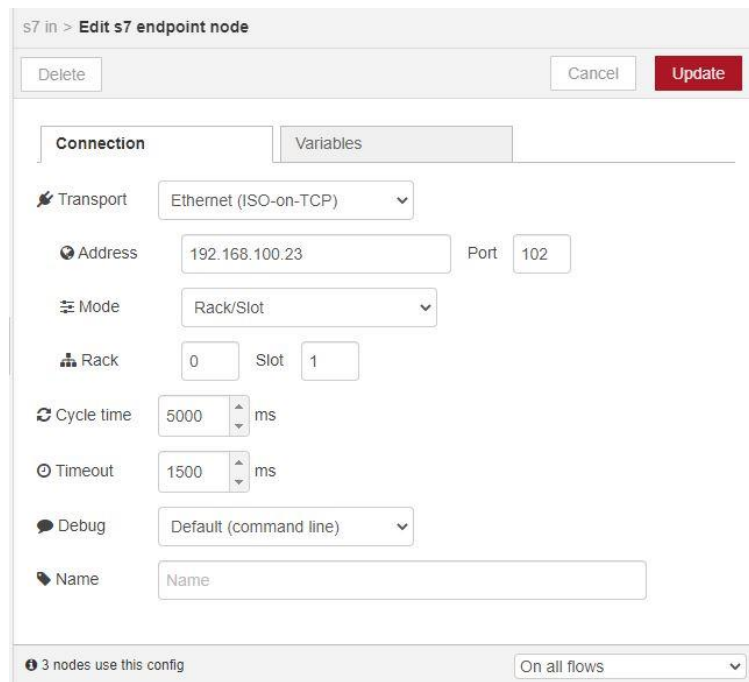
Otvaranjem bilo kojeg bloka od blokova prikazanih na slici 5.3 otvara se izbornik za odabir PLC-a. U padajućem izborniku prikazanom na slici 5.4 može se odabrati željeni PLC ili dodati novi. Klikom na olovku ulazi se u dodatne postavke veze.



Slika 5.4 Prikaz izbornika bloka za dohvaćanje podata s PLC-a („s7 in“ blok)

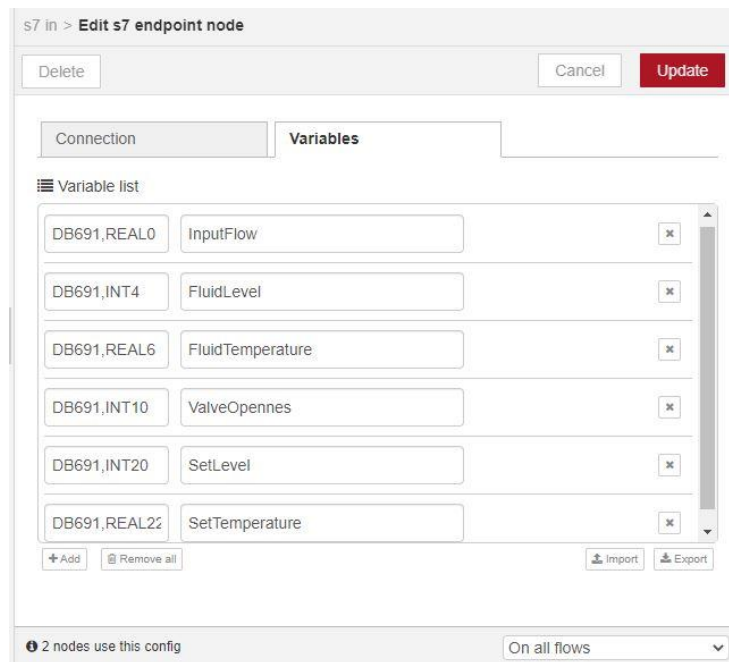
U dodatnim postavkama mreže koje se nalaze u *Connection* kartici prikazanoj na slici 5.5 može se promijeniti:

- protokol za komunikaciju,
- mrežna adresa kojoj želimo pristupiti (npr. ako PLC nije prikazan na padajućem izborniku na slici 5.4),
- port (za PLC je gotovo uvijek 102),
- mod kako je postavljen PLC,
- izbor postolja i mjesta na postolju PLC-a,
- frekvencija slanja zahtjeva za podatke,
- vrijeme nakon kojeg se podatak smatra izgubljenim ako se ne izvrši komunikacija
- mogućnost poruka za otklanjanje pogrešaka
- i postavljanje imena bloka (čvora).



Slika 5.5 Prikaz izbornika za uređivanje dodatnih opcija mrežne veze blokova za komunikaciju s PLC-om

U dodatnim postavkama varijabli koje se nalaze u *Variables* kartici prikazanoj na slici 5.6 mogu se postaviti adrese varijabli na PLC-u kojim želimo pristupiti tako da se u prvi stupac upiše ime bloka podataka na PLC-u, zatim zarez, tip podatka varijable i na kraju memorijska adrese na kojoj se nalazi vrijednost varijable. Između tipa podatka i memorijske adrese varijable nema razmaka. U drugi stupac se upisuje ime varijable koje se koristi u programu.



Slika 5.6 Prikaz izbornika za pristupanje memorijskom adresama PLC-a

Kada se jednom postave, mrežne postavke i varijable PLC-a se više ne moraju upisivati osim ako se dodaje novi PLC. Nakon dodavanja PLC se može odabrati u padajućem izborniku kao na slici 5.4.

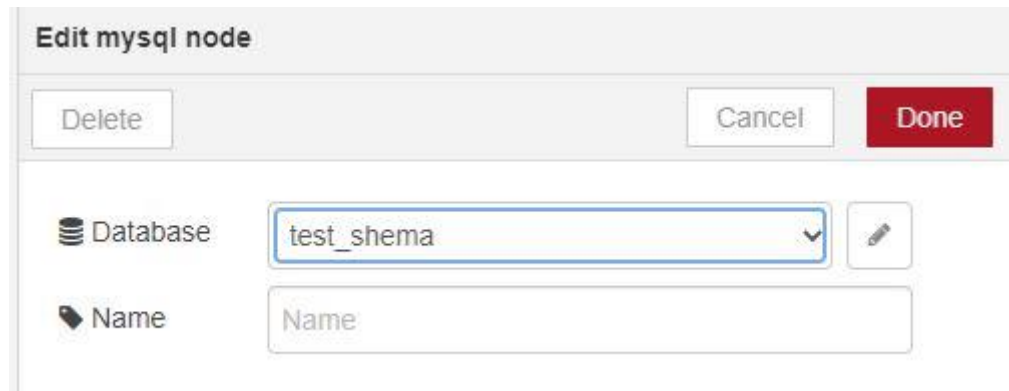
U Node-RED-u dodana je paleta koja nam omogućuje komunikaciju s MySQL bazom podataka na serveru. Kako bi se omogućila komunikacija upisane su tražene vrijednosti u opis bloka. Na slici 5.7 prikazan je izgled bloka za pristup bazi podataka.



Slika 5.7 Blok za pristup bazi podataka

Otvaranjem bloka prikazanog na slici 5.7 otvara se izbornik za odabir baze podataka i postavljanje imena bloka. Izbornik za odabir baze podataka i postavljanje imena bloka prikazani

su na slici 5.8. U padajućem izborniku na slici 5.8 može se odabrati željena baza podataka ili dodati nova. Klikom na olovku ulazi se u dodatne postavke veze.

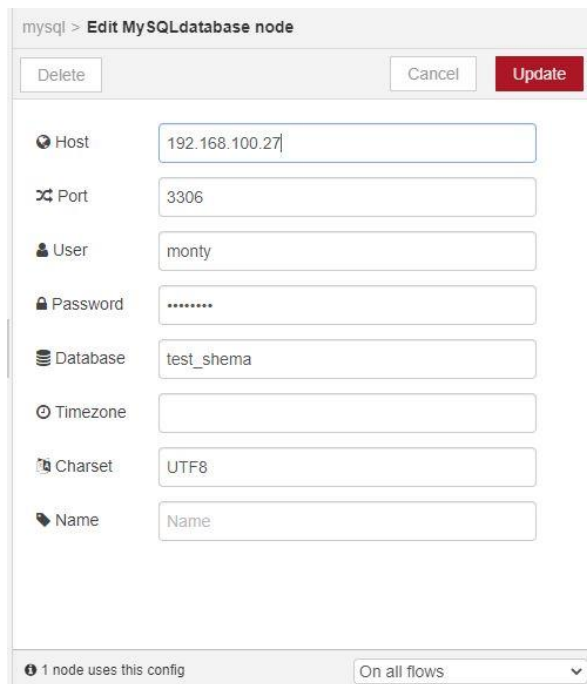


The image shows a dialog box titled "Edit mysql node". At the top, there are three buttons: "Delete", "Cancel", and "Done". Below the buttons, there are two input fields. The first is labeled "Database" and contains a dropdown menu with the text "test_shema" and a small edit icon to its right. The second is labeled "Name" and contains a text input field with the text "Name".

Slika 5.8 Opcije unutar bloka za komunikaciju s bazom podataka

U dodatnim postavkama veze prikazanim na slici 5.9 upisuje se:

- IP adresa servera na kojem se baza podataka nalazi
- Port (za MySQL bazu podataka gotovo uvijek je 3306)
- Ime i lozinka korisnika
- Vremenska zona
- Može se dodati ime bloka

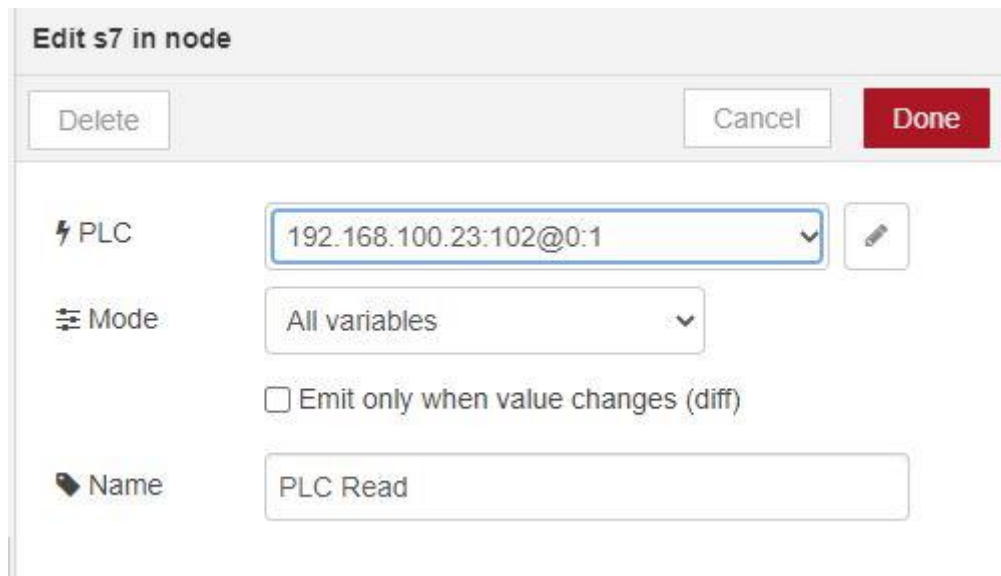


Slika 5.9 Prikaz izbornika za uređivanje opcija mrežne veze s bazom podataka

5.1.2 Čitanje podataka iz memorije Siemens S7 – 1500 PLC-a i spremanje u MySQL bazu podataka

Sa Siemens S7 - 1500 PLC-a se čitaju trenutne vrijednosti varijabli procesa iz memorije i spremaju se u MySQL bazu podataka svakih 5 sekundi.

Za čitanje podataka s PLC-a nakon što su postavljene opcije mreže i varijabli, u bloku za dohvaćanje podata s PLC-a („s7 in“ blok) odabiremo način slanja varijabli i željeno ime bloka. Za slanje vrijednosti varijabli mogu se odabrati sve varijable, jedna varijabla ili jedna varijabla po poruci. Za potrebe ovog zadatke odabrane su sve varijable. Ako se odabere potvrdni okvir sa slike 5.10 podaci će se slati jedino u slučaju kada se vrijednosti na memorijskim adresama na PLC-u promjene.



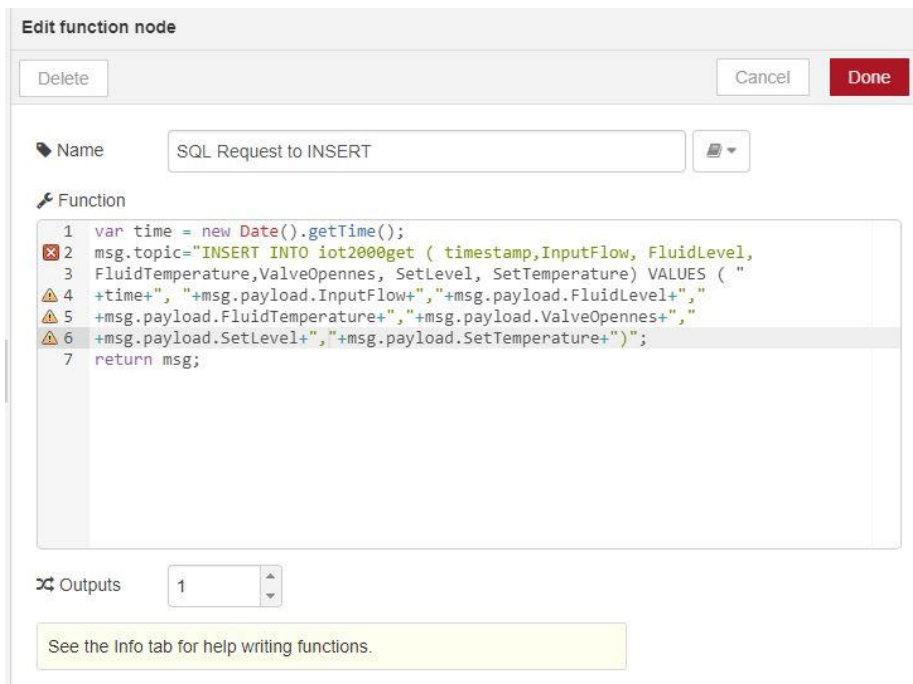
Slika 5.10 Prikaz izbornika za uređivanje veze s7 in bloka

Za izlaz iz bloka za dohvaćanje podata s PLC-a („s7 in“ bloka, preimenovanog u PLC Read) dobiva se objekt koji u svakom redu sadrži ime varijable, dvotočku i vrijednost s memorijske adrese. Izlaz iz bloka za dohvaćanje podata s PLC-a prikazan je na slici 5.11.



Slika 5.11 Izlaz bloka za dohvaćanje podata s PLC-a (objekt s imenom i vrijednostima varijabli)

Kako bi se vrijednosti s izlaza bloka za dohvaćanje podata s PLC-a („s7 in“ bloka) mogle upisati u bazu podataka moraju se obraditi i pretvoriti u SQL zahtjev. Tu pretvorbu obavlja funkcija „SQL Request to INSERT“. Uz prijašnje podatke dobivene iz bloka za dohvaćanje podata s PLC-a („s7 in“ bloka) dodaje se još i vrijeme i sve zajedno se pretvara u SQL zahtjev koji vrijednosti sprema u *iot2000get* tablicu MySQL baze. Kod „SQL Request to INSERT“ funkcije prikazan je na slici 5.12.



Slika 5.12 Funkcija „SQL Request to INSERT“ (zbog preglednosti slike ovaj dugački string je rascjepkan i prikazan u više redova)

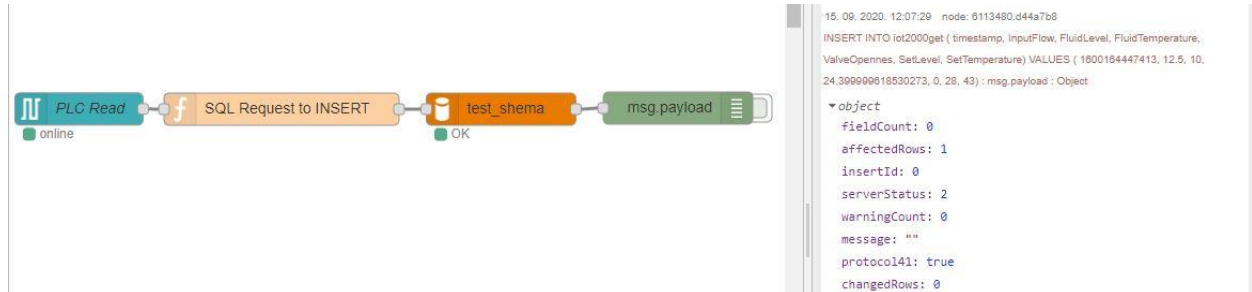
Nakon pretvorbe izlaza bloka za dohvaćanje podata s PLC-a pomoću „SQL Request to INSERT“ dobije se string koji predstavlja SQL zahtjev spreman za slanje u bazu podataka. SQL zahtjev generiran pomoću „SQL Request to INSERT“ funkcije prikazan je na slici 5.13.



Slika 5.13 Izlaz funkcije SQL Request to INSERT

Zahtjev generiran pomoću „SQL Request to INSERT“ funkcije se šalje na ulaz bloka za pristup bazi podataka. Blok za pristup bazi podataka prosljeđuje SQL zahtjev bazi podataka koja potom

izvršava SQL zahtjev i sprema vrijednosti s PLC-a u bazu podataka. Tok Node-RED programa za slanje podataka s PLC-a u bazu podataka je prikazan na slici 5.14.

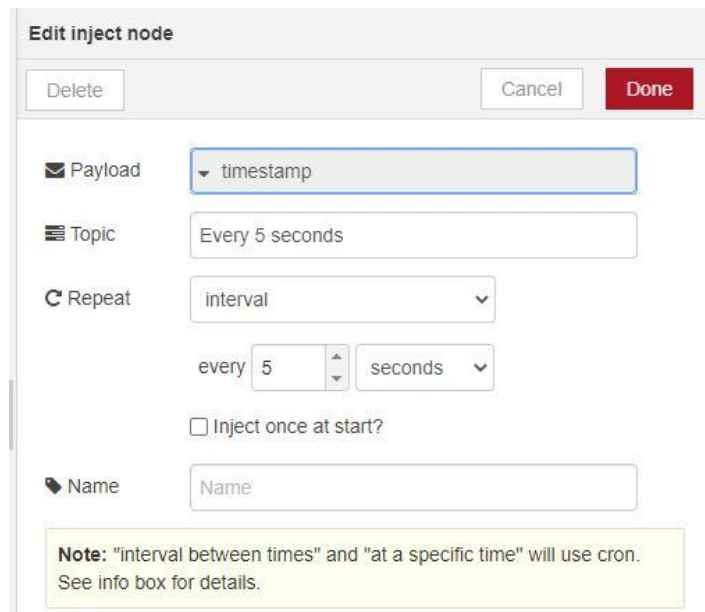


Slika 5.14 Slanje zahtjeva za upis podataka s PLC-a na MySQL bazu podataka (cijeli tok)

5.1.3 Čitanje vrijednosti iz MySQL baze podataka i spremanje u memoriju Siemens S7 – 1500 PLC-a

Iz MySQL baze podatak se dohvaćaju spremljene tražene vrijednosti upravljanih varijabli i spremaju se unutar memorije Siemens S7 – 1500 PLC-a svakih 5 sekundi.

Za vremensko ponavljanje svakih 5 sekundi koristi se inject blok. Kao vrijednost šalje trenutno vrijeme (iako se vrijeme ne koristi program se ne može odvijati kada je prazan inject). Postavljanjem opcije „Repeat“ na interval omogućava se ponavljanje radnje. U ovom primjeru, postavljeno je da se radnja ponavlja svakih 5 sekundi. Postavke inject bloka prikazane su na slici 5.15.



Slika 5.15 Opcije inject bloka

Za dobivanje vrijednosti iz baze potreban nam je SQL zahtjeva na bazu podataka. Funkcija „SQL Request to Read“ postavlja string sa zahtjevom na bazu koji čita vrijednosti iz *iot2000set* tablice u MySQL bazi. Kod „SQL Request to Read“ funkcije prikazan je na slici 5.16. Izlaz funkcije „SQL Request to Read“ prikazan je na slici 5.17.

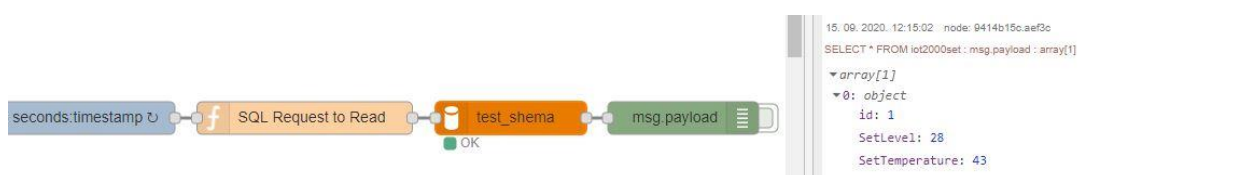


Slika 5.16 Funkcija SQL „Request to Read“



Slika 5.17 Izlaz funkcije „SQL Request to Read“

Nakon što se pošalje zahtjev za čitanje na bazu, baza vraća niz objekata dužine jedan. U objektu se nalaze referentne upravljane veličine procesa. Izlaz iz MySQL baze prikazan je na slici 5.18.

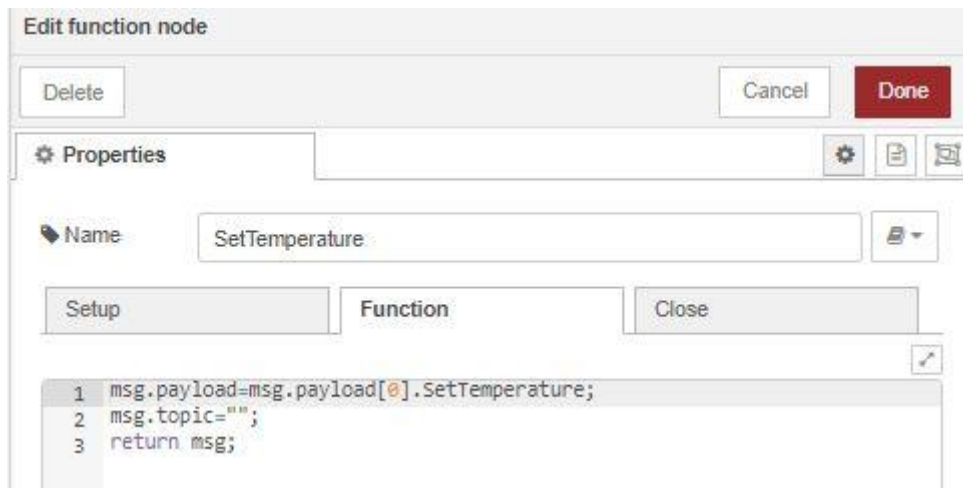


Slika 5.18 Izlaz iz MySQL baze podataka na SQL zahtjev „SQL Request to Read“ funkcije

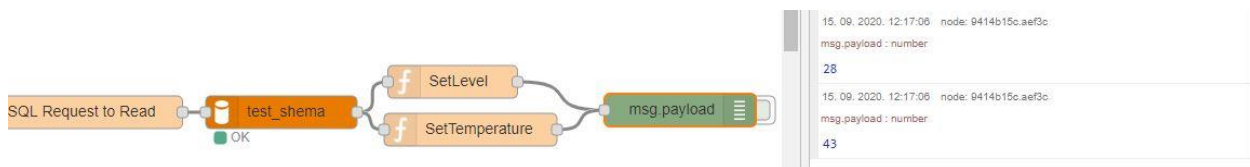
Nakon što je dobiven izlaz iz baze podataka potrebno je izvaditi pojedine vrijednosti iz dobivenog izlaza. To je napravljeno uz pomoć funkcija SetLevel i SetTemperature koje na izlazu daju vrijednost razine i temperature tekućine. Kod funkcije SetLevel prikazan je na slici 5.19. Kod funkcije SetTemperature prikazan je na slici 5.20. Izlaz funkcija SetLevel i SetTemperature prikazan je na slici 5.21.



Slika 5.19 Funkcija SetLevel koja izvlači vrijednost referentne razine tekućine u spremniku

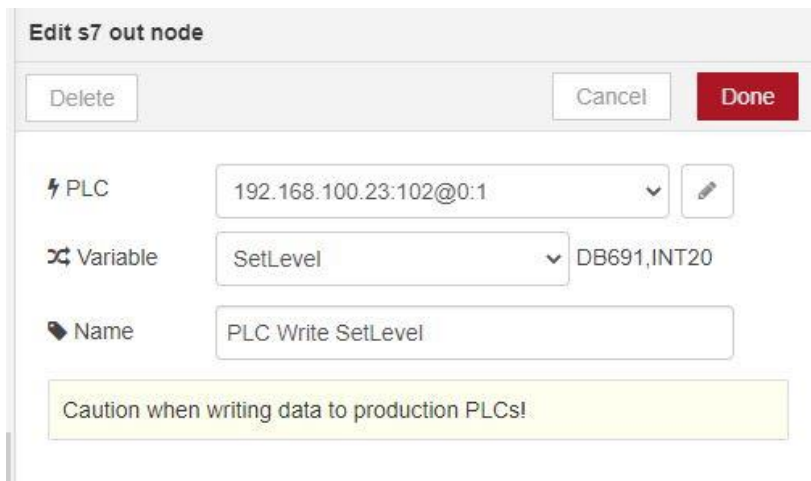


Slika 5.20 Funkcija SetTemperature koja izvlači vrijednost referentne temperature tekućine u spremniku

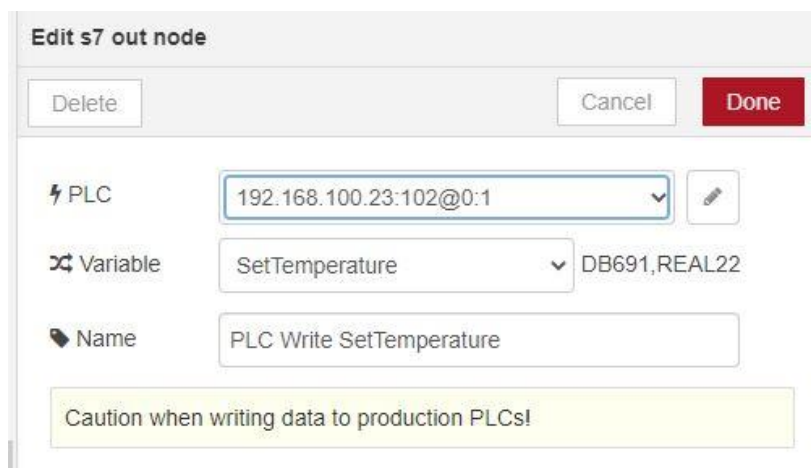


Slika 5.21 Izlaz funkcija SetLevel i SetTemperature

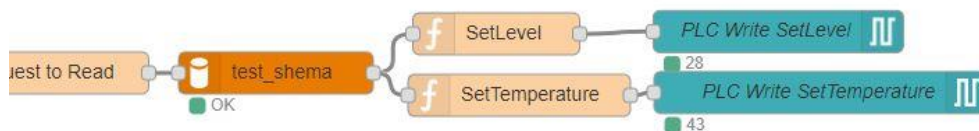
Vrijednosti dobivene iz funkcija SetLevel i SetTemperature se šalju u memoriju Siemens S7 – 1500 PLC-a uz pomoć dva bloka za upis podataka na PLC („s7 out“ bloka). Unutar svakog bloka za upis podataka na PLC („s7 out“ bloka) potrebno je naznačiti na koje memorijsko mjesto se spremaju vrijednosti. Jedan blok za upis podataka na PLC uzima vrijednost dobivenu funkcijom SetLevel i sprema ju na memorijsko mjesto SetLevel varijable na PLC-u. Drugi blok za upis podataka na PLC uzima vrijednost dobivenu funkcijom SetTemperature i sprema ju na memorijsko mjesto SetTemperature varijable na PLC-u. Postavke bloka za upis podataka na PLC SetLevel varijable prikazane su na slici 5.22. Postavke bloka za upis podataka na PLC SetTemperature varijable prikazane su na slici 5.23. Izgled toka za postavljanje referentnih vrijednosti varijabli na PLC prikazan je na slici 5.24.



Slika 5.22 Bloka za upis podataka na PLC koji sprema vrijednost iz SetLevel funkcije u SetLevel varijablu



Slika 5.23 Bloka za upis podataka na PLC koji sprema vrijednost iz SetTemperature funkcije u SetTemperature varijablu



Slika 5.24 Izlaz funkcija SetLevel i SetTemperature se šalje na blokove za upis podataka na PLC koji spremaju vrijednosti u memoriju Siemens S7 – 1500 PLC-a

5.2 Programsko rješenje na serveru

Server ima nekoliko zadataka. Server mora održavati bazu podataka. Aplikacija na serveru dohvaća podatke iz baze podataka i prikazuje ih na nadzornoj ploči (*eng. dashboard*). Aplikacija na serveru mijenja referentne upravljane veličine na zahtjev korisnika, referentne upravljane veličine se mijenjaju na nadzornoj ploči (*eng. dashboard*) i spremaju se u bazu podataka.

5.2.1 MySQL baza podataka

Server na sebi ima pokrenutu bazu podataka kojoj se može pristupiti lokalno i putem interneta. U ovom slučaju koristi se MySQL baza podataka.

U bazi podataka su napravljene dvije tablice:

- *iot2000get* - sadrži podatke iz memorije Siemens S7 – 1500 PLC-a i upisuju se nova očitavanja svakih 5 sekundi pomoću Siemens IOT2040 mrežnog modula. Spremljeni podaci se na serveru dohvaćaju i prikazuju na nadzornoj ploči (*eng. dashboard*). Varijable *iot2000get* tablice i njihove postavke prikazane su na slici 5.25. Vrijednosti nekoliko varijabli iz *iot2000get* tablice iz baze podataka prikazane su na slici 5.27.
- *iot2000set* - sadrži tražene vrijednosti upravljanih veličinama. Tablica ima samo jedan redak vrijednosti koji se mijenja pomoću upravljačkog zaslona na serveru. Siemens IOT2040 mrežni modul prima podatke iz baze podataka i sprema ih u memoriju Siemens S7 – 1500 PLC-a. Varijable *iot2000set* tablice i njihove postavke prikazane su na slici 5.26. Vrijednosti varijabli iz *iot2000set* tablice iz baze podataka prikazane su na slici 5.28.

test_table iot2000get iot2000set test_shema.iot2000get **iot2000get - Table** x

Table Name: Schema: **test_shema**

Charset/Collation: Engine: **InnoDB**

Comments:

| Column Name | Datatype | PK | NN | UQ | B | UN | ZF | AI | G | Default/Expression |
|------------------|----------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------|
| timestamp | DOUBLE | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| InputFlow | DOUBLE | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| FluidLevel | INT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| FluidTemperature | DOUBLE | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| ValveOpennes | INT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| SetLevel | INT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| SetTemperature | DOUBLE | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |

Slika 5.25 Prikaz varijabli u *iot2000get* tablici u bazi podataka

test_table iot2000get iot2000set test_shema.iot2000get iot2000get - Table **iot2000set - Table** x

Table Name: Schema: **test_shema**

Charset/Collation: Engine: **InnoDB**

Comments:

| Column Name | Datatype | PK | NN | UQ | B | UN | ZF | AI | G | Default/Expression |
|----------------|----------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------|
| id | INT | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| SetLevel | INT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| SetTemperature | DOUBLE | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |

Slika 5.26 Prikaz varijabli u *iot2000set* tablici u bazi podataka

test_table **iot2000get** x iot2000set test_shema.iot2000get iot2000get - Table iot2000set - Tat

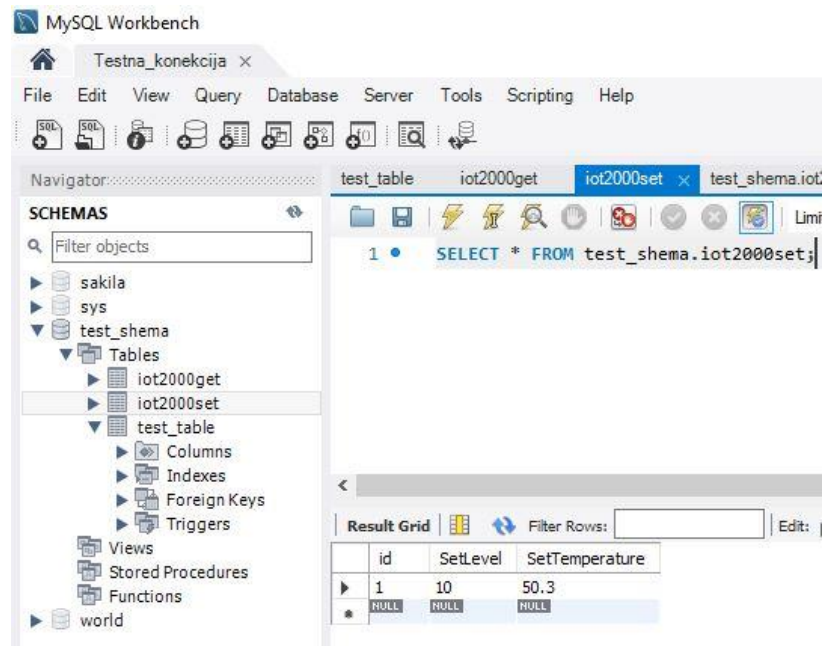
Limit to 1000 rows

```
1 • SELECT * FROM test_shema.iot2000get;
```

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell C

| | timestamp | InputFlow | FluidLevel | FluidTemperature | ValveOpennes | SetLevel | SetTemperature |
|---|---------------|-----------|------------|------------------|--------------|----------|----------------|
| ▶ | 1599820189390 | 25.3 | 10 | 50.3 | 0 | 10 | 50 |
| | 1599820198349 | 25.3 | 10 | 50.3 | 0 | 10 | 50 |
| | 1599820221830 | 25.3 | 10 | 50.3 | 0 | 10 | 50 |

Slika 5.27 Prikaz nekoliko vrijednosti varijabli iz *iot2000get* tablice iz baze podataka



Slika 5.28 Prikaz vrijednosti varijabli iz *iot2000set* tablice iz baze podataka

Kako bi se mogli slati SQL zahtjevi sa Siemens IOT2040 mrežnog modula potrebno je napraviti novog korisnika kojem je omogućen pristup MySQL bazi podataka i koji može pristupiti MySQL bazi podataka s druge IP adrese. To se postiže uz pomoć SQL zahtjeva pokrenutih u MySQL Workbench-u prikazanih na slici 5.29.

```

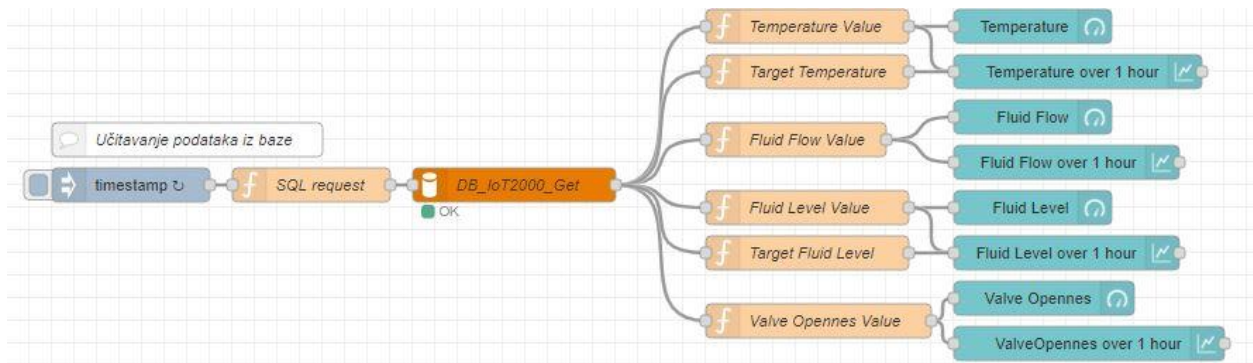
1 • CREATE USER 'monty'@'localhost' IDENTIFIED BY 'password';
2 • GRANT ALL PRIVILEGES ON *.* TO 'monty'@'localhost'
3   WITH GRANT OPTION;
4
5 • CREATE USER 'monty'@'%' IDENTIFIED BY 'password';
6 • GRANT ALL PRIVILEGES ON *.* TO 'monty'@'%'
7   WITH GRANT OPTION;
8
9 • ALTER USER 'monty'@'localhost' IDENTIFIED WITH mysql_native_password BY 'password';
10 • ALTER USER 'monty'@'%' IDENTIFIED WITH mysql_native_password BY 'password';
11
12 • FLUSH PRIVILEGES;

```

Slika 5.29 SQL zahtjevi na MySQL bazu podataka za stvaranje novog korisnika

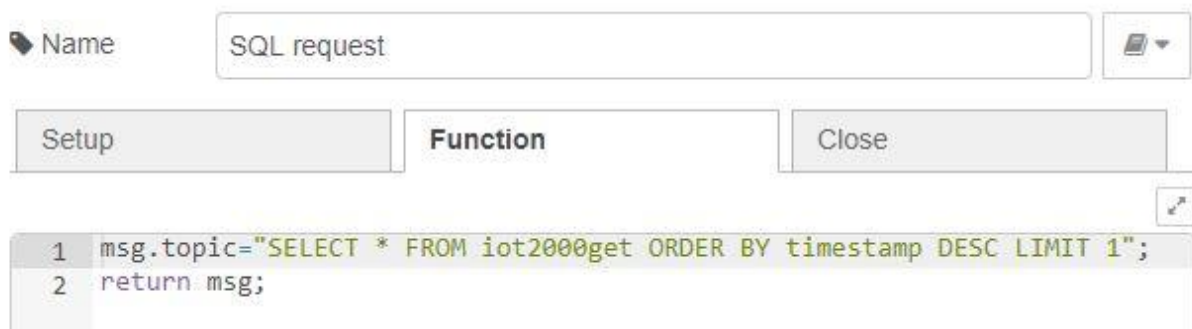
5.2.2 Dohvaćanje podataka iz baze i njihov prikaz

Svakih 5 sekundi na bazu podataka se šalje SQL zahtjev za dohvat najnovijeg unosa iz *iot2000get* tablice. Dohvaćeni podaci se nakon toga prikazuju na upravljačkoj ploči. Na slici 5.30 prikazan je cijeli tok za dohvaćanje i prikaz podataka iz baze podataka.



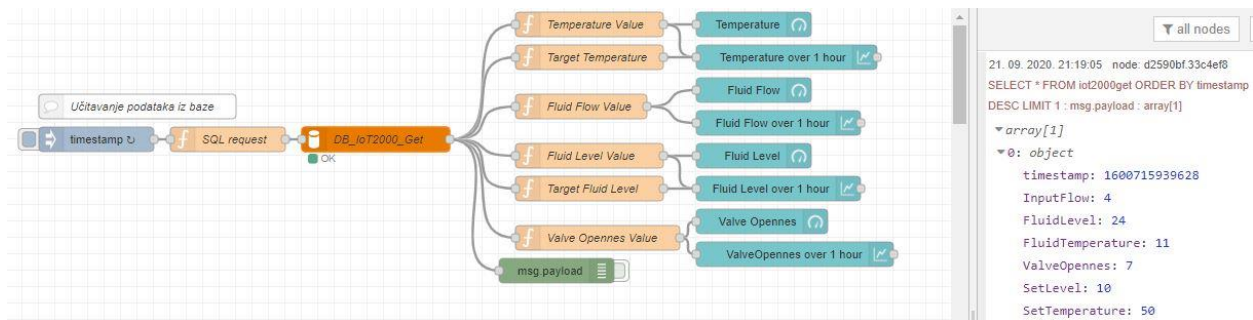
Slika 5.30 Tok za dohvaćanje podataka iz baze podataka i prikaz na upravljačkoj ploči

Za pokretanje svakih 5 sekundi koristi se inject blok s identičnim postavkama bloku na slici 5.15. Inject blok pokreće funkciju „SQL request“ koja generira SQL zahtjev za dohvat najnovijih podataka iz tablice *iot2000get* iz baze podataka. Na slici 5.31 prikazan je kod funkcije „SQL request“.



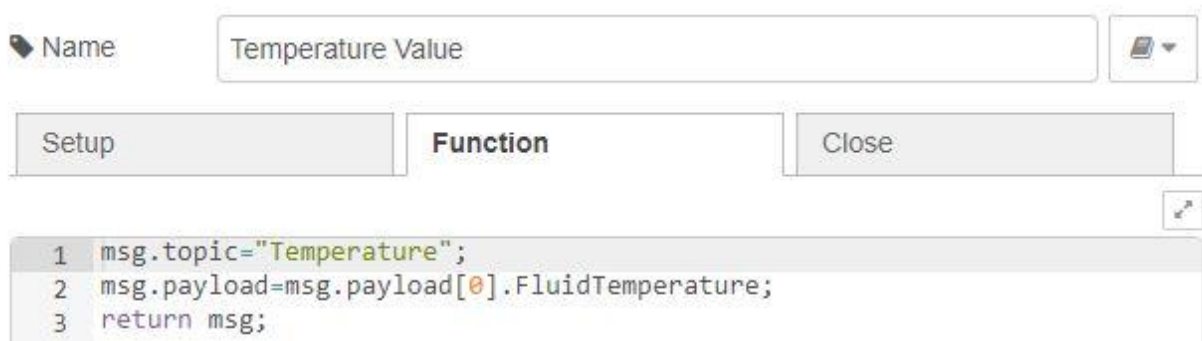
Slika 5.31 Kod funkcije „SQL request“

Generirani SQL zahtjev dobiven funkcijom „SQL request“ se šalje na bazu podataka koja za rezultat vraća tražene podatke. Dohvaćeni podaci iz baze podataka su prikazani na slici 5.32.



Slika 5.32 Dohvaćeni podaci iz baze podataka

Iz dohvaćenih podataka se pomoću funkcija „Temperature Values“, „Target Temperature“, „Fluid Flow Value“, „Fluid Level Value“, „Target Fluid Level“ i „Value Opennes Value“ odvajaju zasebne vrijednosti. Struktura svim funkcijama je ista. Jedina razlika između funkcija je koju vrijednost funkcija odvajaju i koje ime pridjeljuje toj vrijednosti. Na slici 5.33 je prikazana funkcija „Temperature Values“.



Slika 5.33 Kod funkcije „Temperature Value“

Odvojene vrijednosti s imenom se šalju na blokove mjerače i blokove za grafovski prikaz. Funkcije „Target Temperature“ i „Target Fluid Level“ se ne šalju na blokove mjerače nego jedino na blokove za grafovski prikaz.

Blokovi mjerači prikazuju zadnju primljenu vrijednost na skali zadanih dimenzija.

Blokovi mjerači imaju opcije:

- Grupa – odabire se kojoj grupi prikaza mjerač pripada
- Veličina – veličina prikaza
- Tip – odabire se izgled bloka na upravljačkoj ploči

- Oznaka – ime koje se prikazuje na upravljačkoj ploči
- Mjerna jedinica
- Gornja i donja granica vrijednosti
- Odabir boja i granica vrijednosti za pojedine regije

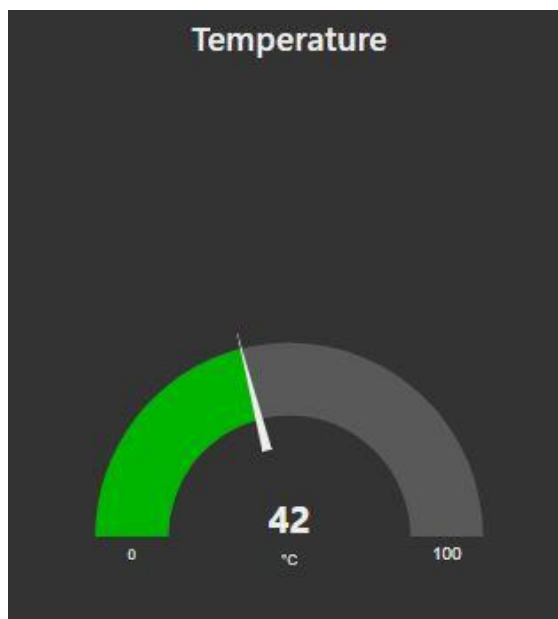
Izgled postavki bloka mjerača prikazan je na slici 5.34.

Izgled bloka mjerača na upravljačkoj ploči prikazan je na slici 5.35 i slici 5.36.

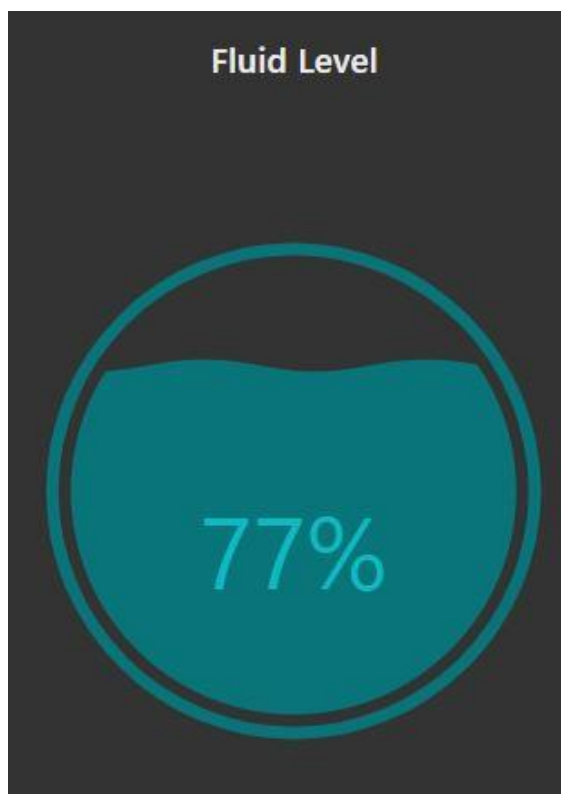
The image shows a configuration window titled "Edit gauge node". At the top, there are three buttons: "Delete", "Cancel", and "Done". Below this is a "Properties" section with several settings:

- Group:** A dropdown menu showing "[GUI] Temperature" with an edit icon.
- Size:** A text input field containing "6 x 10".
- Type:** A dropdown menu showing "Gauge".
- Label:** A text input field containing "Temperature".
- Value format:** A text input field containing "{{value}}".
- Units:** A text input field containing "°C".
- Range:** Two text input fields labeled "min" and "max" containing "0" and "100" respectively.
- Colour gradient:** A visual representation of a gradient with three colored segments: yellow, green, and red.
- Sectors:** A series of text input fields containing "0", "40", "60", and "100", separated by ellipses.
- Name:** An empty text input field.

Slika 5.34 Blok mjerač sa svim postavkama

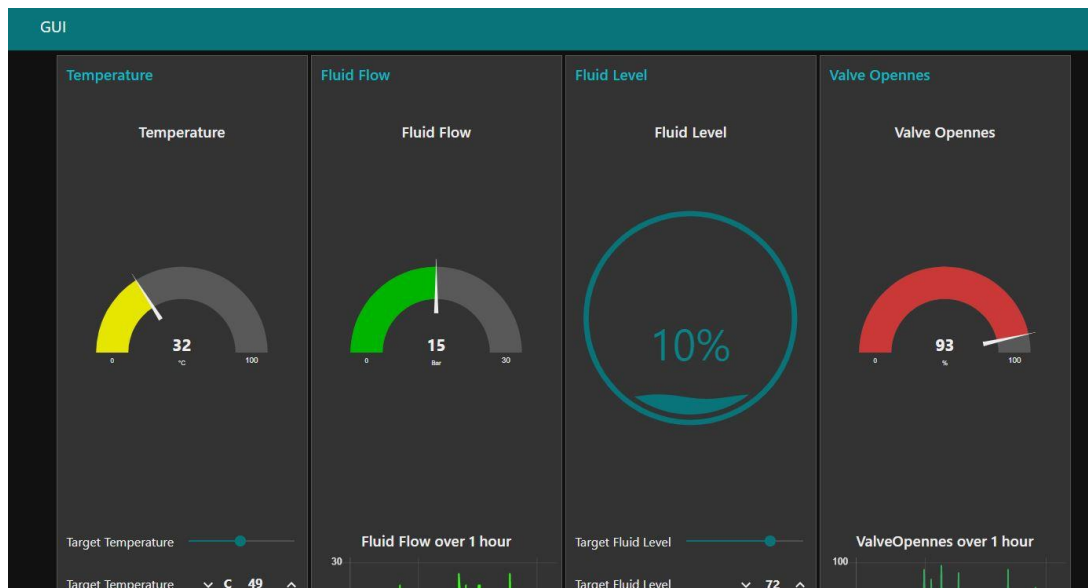


Slika 5.35 Izgled bloka mjerača na upravljačkoj ploči tip – *Gauge*



Slika 5.35 Izgled bloka mjerača na upravljačkoj ploči tip - *Level*

Svi mjerači korišteni u programu prikazani su na slici 5.36



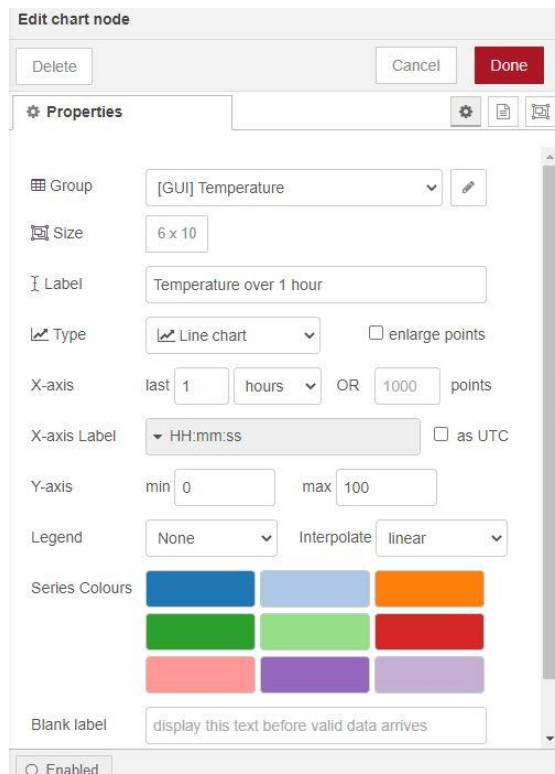
Slika 5.36 Prikaz gornjeg dijela upravljačke ploče

Blokovi za grafovski prikaz primaju vrijednosti na ulazu i prikazuju vrijednosti na grafu tijekom zadanog vremena ili broja unosa vrijednosti. Grafovi mogu prikazati jednu ili više vrijednosti na jednog grafu. Na ulazu *topic* označava ime varijable a *payload* vrijednost koja se prikazuje na grafu.

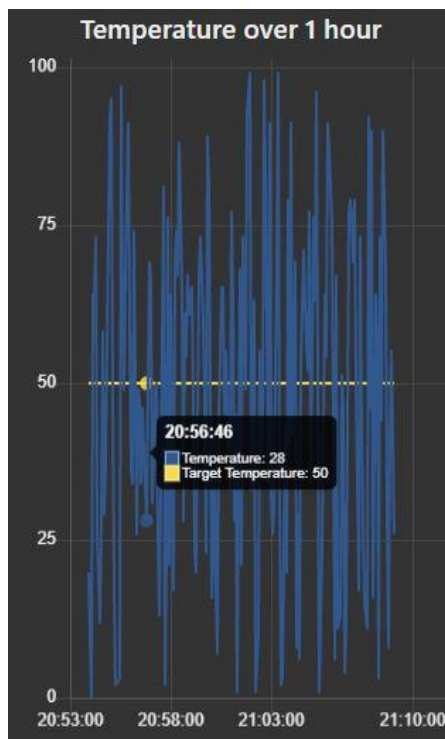
Blok za grafovski prikaz ima opcije:

- Odabira grupe kojoj pripadaju na upravljačkoj ploči
- Odabira veličine
- Ime koje se prikazuje iznad grafa
- Tip grafa – linijski, stupčasti, isječak itd.
- Razdoblje prikazivanje npr. zadnjih sat vremena ili zadnjih 1000 unosa

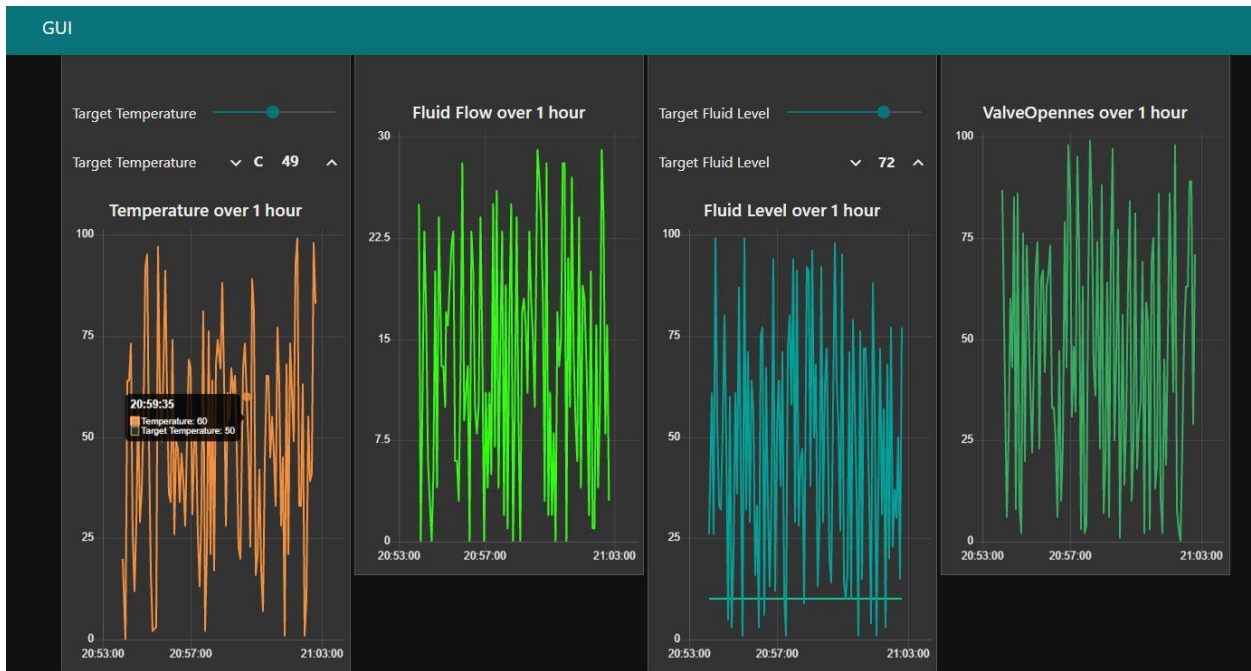
Opcije bloka za grafovski prikaz prikazane su na slici 5.37, dok je izgled bloka za grafovski prikaz s dvije varijable prikazan je na slici 5.38. Na slici 5.39 se nalazi prikaz drugog dijela upravljačke ploče.



Slika 5.37 Opcije bloka za grafovski prikaz

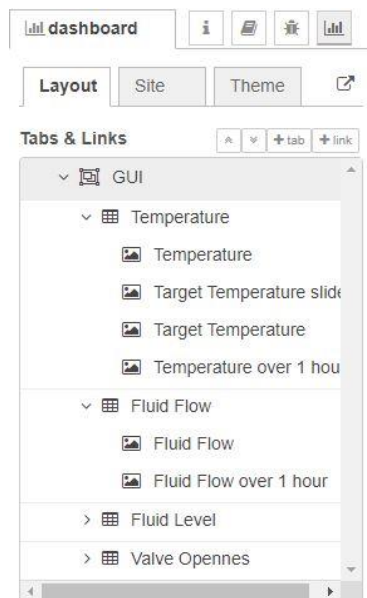


Slika 5.38 Izgled bloka za grafovski prikaz s dvije varijable



Slika 5.39 Prikaz drugog dijela upravljačke ploče

Za raspored i položaj svih vizualnih dijelova na upravljačkoj ploči koristi se *Layout* kartica u kojoj se mogu kreirati grupe i mijenjati razmještaj komponenti za prikaz. Na slici 5.40 prikazana je kartica *Layout*.

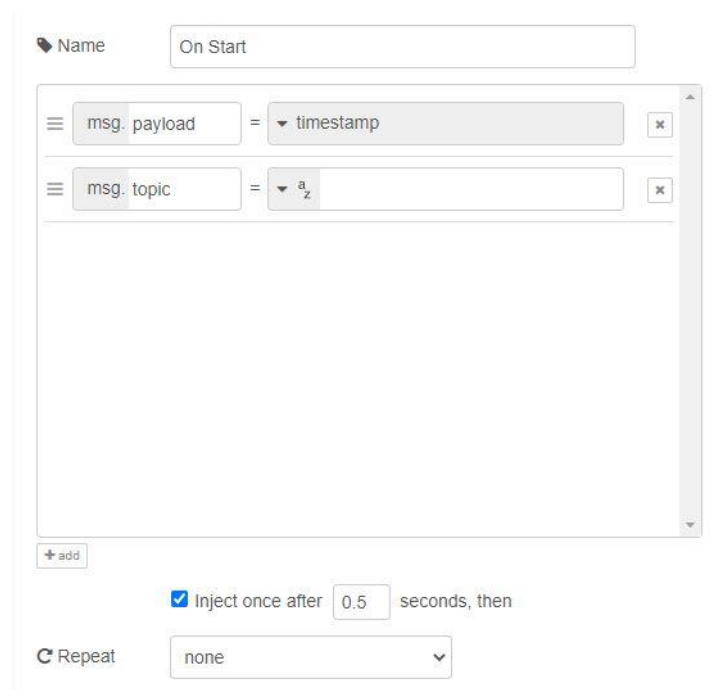


Slika 5.40 Prikaz *Layout* kartice upravljačke ploče

5.2.3 Upravljanje referentnim veličinama

Kako se proces ne bi naglo promijenio prilikom pokretanja upravljačke ploče potrebno je postaviti referentne veličinu u posljednje izmjerene veličine. Nakon toga se te vrijednosti proslijeđuju na ulaz bloku klizaču i brojčanik bloku tako se klizač i brojčanik na upravljačkoj ploči postavljaju na te vrijednosti. Nakon toga klizač i brojčanik su spremni za korištenje. Postavljanje vrijednosti je identično za varijable referentne temperature i razine tekućine.

Prvo je potrebno postaviti inject blok koji se pokreće jedino prilikom pokretanja toka. Klikom na kvačicu u opcijama inject bloka blok se pokreće jednom prilikom pokretanja toka. Na slici 5.41 prikazan je inject tok koji se pokreće samo prilikom pokretanja toka.



Slika 5.41 Inject blok za pokretanja ne početku toka

Inject blok pokreće funkcije „SQL request target Fluid Temperature“ i „SQL request target Fluid Level“. Funkcije generiraju SQL zahtjev koji se šalje na bazu podataka. Kod funkcije „SQL request target Fluid Temperature“ prikazan je na slici 5.42.



Slika 5.42 Kod funkcije „SQL request target Fluid Temperature“

Iz odgovora dobiveno iz baze podataka izdvaja se tražena referentna vrijednost uz pomoć funkcija „Set New Target Temperature“ i „Set New TargetLevel“. Kod funkcije „Set New Target Temperature“ prikazan je na slici 5.43.



Slika 5.43 Kod funkcije „Set New Target Temperature“

Izlaz iz funkcija „Set New Target Temperature“ i „Set New TargetLevel“ se šalje na ulaz bloka klizača te se tako postavljaju vrijednosti klizača i brojčanika . Blok klizač i blok brojčanik su spojeni u petlju tako da izlaz jednoga ide u ulaz drugoga i obratno te se tako promjenom vrijednosti jednog mijenja i vrijednost drugoga.

Blok klizač je blok koji na ulazu može primiti vrijednost i postaviti klizač na nadzornoj ploči na primljenu vrijednost. Promjenom položaja klizača na nadzornoj ploči na izlazu bloka klizača šalje se vrijednost položaja klizača.

Blok klizač ima opcije:

- Odabira grupe kojoj pripadaju na upravljačkoj ploči
- Odabira veličine
- Ime koje se prikazuje pokraj klizača
- Granice klizača i korak

- Način izlaza

Na slici 5.44 su prikazane opcije bloka klizača.

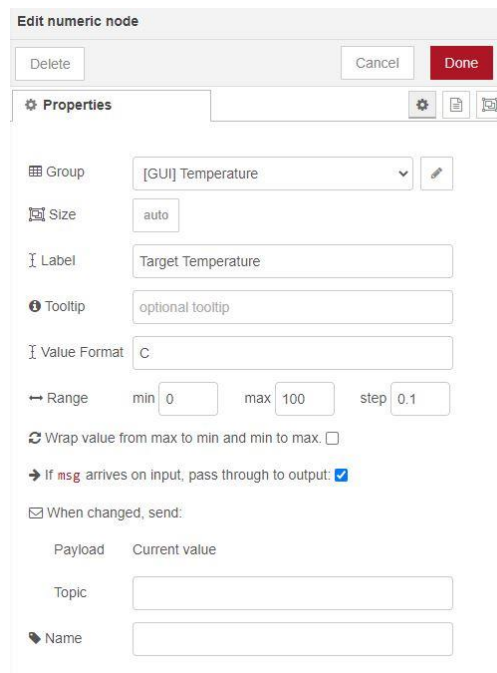
Slika 5.44 Opcije bloka klizača

Blok brojač je blok koji na ulazu može primiti vrijednost i postaviti svoju vrijednost na nadzornoj ploči na primljenu vrijednost. Klikom na strelicu gore ili dole na nadzornoj ploči na izlazu bloka brojača šalje se vrijednost s nadzorne ploče.

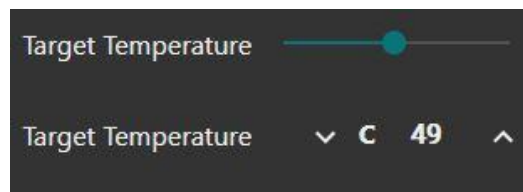
Blok brojač ima opcije:

- Odabira grupe kojoj pripadaju na upravljačkoj ploči
- Odabira veličine
- Ime koje se prikazuje pokraj brojača
- Granice brojača i korak
- Način izlaza

Na slici 5.45 su prikazane opcije bloka brojača. Na slici 5.46 prikazan je izgled bloka klizača i bloka brojača na upravljačkoj ploči.

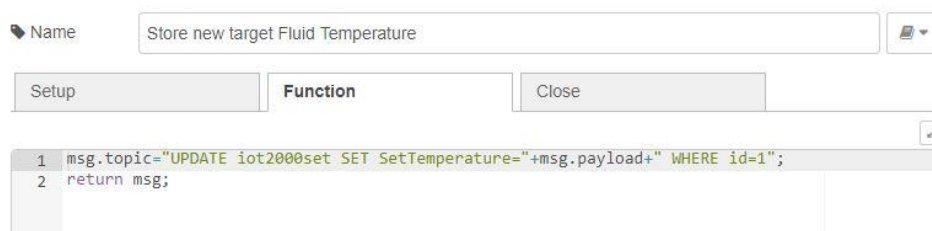


Slika 5.45 Opcije bloka brojača

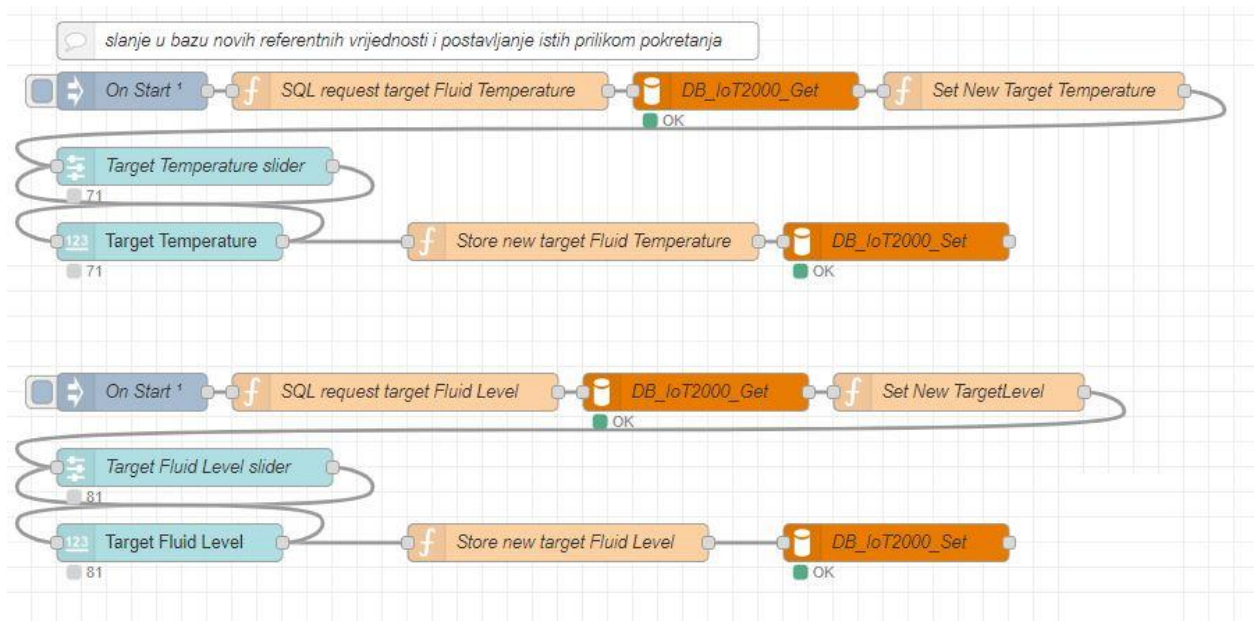


Slika 5.46 Izgled bloka klizača i bloka brojača na upravljačkoj ploči

Izlaz iz bloka brojača ulazi u funkciju „Store new target Temperature“ ili „Store new target Fluid Level“ koja generira SQL zahtjev na bazu podataka za spremanje vrijednosti referentne temperature ili razine tekućine u tablicu *iot2000set* u bazi podataka. Kod funkcije „Store new target Temperature“ prikazan je na slici 5.47. Na slici 5.48 prikazani su tokovi za postavljanje referentnih vrijednosti temperature i razine tekućine.



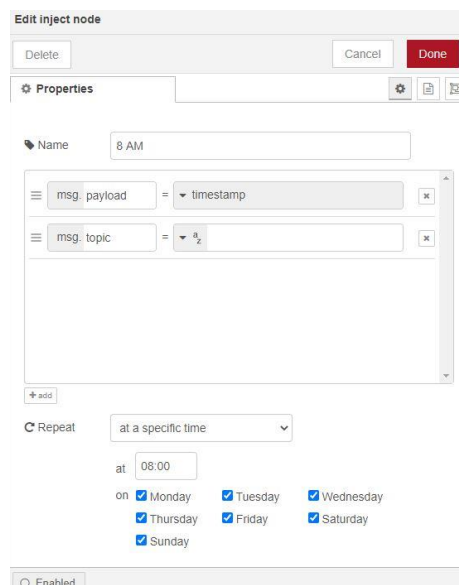
Slika 5.47 Kod funkcije „Store new target Temperature“



Slika 5.48 Prikaz tokova za upravljanje referentnim veličinama

5.2.4 Postavljanje parametara procesa ovisno o dobu dana

Za postavljanje parametara procesa ovisno o dobu dana potreban nam je inject blok koji je postavljen da se pokreće u određeno vrijeme (8 sati i 20 sati). U izborniku ponavljanje odabere se opcija „u određeno vrijeme“ i postavi se vrijeme izvođenja i kojim danima se izvodi. Slika 5.49 prikazuje opcije inject bloka koje su korištene za pokretanje bloka u 8 sati. Analogno tomu je napravljeno za 20 sati.



Slika 5.49 Prikaz opcija inject bloka za pokretanje u određeno vrijeme

Funkcija „Set Level =50“ se pokreće u 8 sati, a funkcija „Set Level =30“ u 20 sati. Funkcija „Set Level =50“ šalje SQL zahtjev na bazu podataka koji postavlja referentnu vrijednost razine tekućine na 50. Analogno tomu je napravljeno za „Set Level =30“. Slika 5.50 prikazuje kod funkcije „Set Level =50“.



The screenshot shows a function editor window with the title 'Set Level = 50'. It has three buttons: 'Setup', 'Function', and 'Close'. The 'Function' button is active. Below the buttons, there is a code editor with the following code:

```
1 msg.topic="UPDATE iot2000set SET SetLevel="+50+" WHERE id=1";  
2 return msg;
```

Slika 5.50 Kod funkcije „Set Level =50“

Na slici 5.51 Prikazani su tokovi za postavljanje parametara procesa ovisno o dobu dana



Slika 5.51 Prikaz tokova za postavljanje parametara procesa ovisno o dobu dana

Izlazi iz baze podataka pokreće funkciju koja osvježava vrijednosti bloka klizača i bloka brojača za upravljanje razinom tekućine.

6. ZAKLJUČAK

Diplomskim radom obuhvaćene su osnove korištenja Siemens IOT2040 mrežnog modula poput rada s bazama podataka, prikupljanja podataka s PLC-ova, prikaz mjerenih veličina i promjena traženih upravljanih veličina. Serverski dio zadatka se vrlo lako može prebaciti da radi na oblaku. Upravljačkoj ploči se može pristupiti s udaljene lokacije putem interneta uz upotrebu web preglednika.

Siemens-ov IOT2040 mrežni modul je sposoban za izvođenje gore navedenih zadataka. U industriji pronalazi mjesto kao sekundarna upravljačka ploča kojoj se može pristupiti na daljinu.

Siemens-ov IOT2040 mrežni modul pogodan je za nadogradnji sustava upravljanim PLC-ovima starije generacije koji nemaju pristup internetu. Na taj se način ne moraju kupovati mnogi novi PLC-ovi što je vrlo skupo nego se može vrlo jednostavno i jeftino nadograditi postojeći sustav s mrežnim modulom Siemens IOT2000/2020/2040 nakon čega se dobiva pristup internetu, bazama podataka i sekundarnoj upravljačkoj ploči kojoj se može pristupiti na daljinu.

LITERATURA

[1] INDUSTRY 4.0

<https://www.hgk.hr/documents/hgk-industrija-4058d8c59722f1e.pdf>

[2] SIMATIC IOT2020, SIMATIC IOT2040

https://cache.industry.siemens.com/dl/files/658/109741658/att_899623/v1/IOT2000_operating_instructions_e_en-US.pdf

[3] Siemens IOT2000 / IOT2020 / IOT2040 Arduino IDE Integration

<https://www.14core.com/siemens-simatic-IOT2000-iot2020-IOT2040-arduino-ide-integration/>

[4] Node-RED

<https://nodered.org/>

[5] MySQL baza podataka

<https://www.mysql.com/>

SAŽETAK

Glavni zadatak ovog diplomskog rada bio je postojeći upravljački sustav nadograditi sa Siemens IOT2040 internet modulom te demonstrirati prednosti korištenja modula na različite načine: pohrana podataka, nadzor i vizualizacija procesa, postavljanje parametara upravljačkog sustava (npr. ovisno o dobu dana, tarifnom modelu i sl.), izračunavanje i osvježavanje parametara regulatora i slično. Svi navedeni zadaci su obavljeni pomoću Node-RED programa, JavaScript programskog jezika i MySQL baze podataka. Siemens IOT2040 mrežni modul je na sebi imao pokrenuti Node-RED s kojim je prosljeđivao podatke s PLC-a upravljanog procesa na bazu podataka na serveru i čitao nove referentne upravljane veličine te ih slao na PLC upravljanog procesa. Na serveru je instalirana MySQL bazu podataka iz koje je aplikacija čitala spremljene vrijednosti s PLC-a i prikazivala ih na nadzornoj ploči. Korisnik je na aplikaciji mogao promijeniti vrijednosti referentnih upravljanih veličina te ih je aplikacija spremala u bazu podataka.

Ključne riječi: automatika, PLC, internet objekata, servisi na oblaku, baze podataka,

ABSTRACT

The main task of this master's thesis was to upgrade the existing control system with Siemens IOT2040 internet module and to demonstrate the pros of using this module in different ways: data storage, supervision and visualization of process, setting control system parameters (e.g. depending on the time of day, tariff model, etc.), calculating and refreshing controller parameters and similar. All listed tasks were performed using Node-RED program, JavaScript programming language and the MySQL database. Siemens IOT2040 network module had to run Node-RED with which it forwarded data from PLC that manages the process to the database on the server and read new target values and forwarded them to PLC that manages the process. The MySQL database was installed on the server from which an application read stored data from PLC and showed it on the control panel. On the application, user could change target values and the application would store them in the database.

Key words: automation, PLC, internet of things, cloud services, databases

ŽIVOTOPIS

Davor Damjanović rođen je 17. rujna 1996. godine u Vinkovcima. U razdoblju od 2003. do 2011. pohađao je Osnovnu školu „Ivana Brlić-Mažuranić“ u Strizivojnoj. Nakon završene Osnovne škole upisuje Gimnaziju „Matija Mesić“ u Slavanskom te ju završava 2015. godine. Nakon srednje škole, 2015. Godine upisuje Fakultet elektrotehnike, računarstva i informacijskih tehnologija (tadašnji Elektrotehnički fakultet u Osijeku), smjer elektrotehnika. Nakon završene 1. godine prebacuje se na smjer računarstvo. Preddiplomski sveučilišni studij završava 2018. Godine i dobiva kvalifikaciju sveučilišnog prvostupnika inženjera računarstva (*univ.bacc.ing.comp.*). Zatim upisuje diplomski studij, izborni blok Robotika i umjetna inteligencija te ga pohađa u razdoblju od 2018. do danas. U razdoblju od 6. srpnja 2020. do 10. kolovoza 2020. obavlja praksu kao back-end developer u tvrtki „Atos“ u Osijeku.

PRILOZI

Prilog 1. Diplomski rad u *.docx* i *.pdf* formatu

Prilog 2. Programsko rješenje zadatka sa Siemens IOT2040 mrežnog modula

Prilog 3. Programsko rješenje zadatka sa servera