

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Rok Stražišar

Sistem za vodenje in nadzor pametne hiše

diplomsko delo
visokošolskega študija

Mentor: doc. dr. Uroš Lotrič

Ljubljana, 2008

Zahvala

Najprej bi se želel zahvaliti mentorju, doc. dr. Urošu Lotriču za vsestransko podporo, strokovno pomoč ter nasvete pri izdelavi diplomske naloge.

Zahvalil bi se tudi sodelavcu Branku Gruden in zaposlenim v podjetju Smarteh d.o.o., ki so mi s strokovnimi nasveti stali ob strani pri izdelovanju praktičnega dela diplomske naloge.

Na koncu bi se rad zahvalil še moji puncu Poloni Kohne, za vsestransko podporo in lektoriranje moje diplomske naloge.

Kazalo

| | | |
|-------|------------------------------------------------------|----|
| 1. | Uvod..... | 5 |
| 2. | Pregled in ponudba sistemov na slovenskem trgu | 11 |
| 3. | Opis krmilnika LPC-2 MC7 | 13 |
| 3.1 | Povezovanje krmilnikov - protokol LonTalk | 14 |
| 3.1.1 | Izmenjava podatkov med krmilniki po mreži LON | 17 |
| 4. | Opis sistema | 21 |
| 4.1 | Delovanje sistema | 21 |
| 4.2 | Shema sistema | 23 |
| 4.3 | Programska koda v krmilnikih | 27 |
| 4.3.1 | Splošni signali in nastavitve | 27 |
| 4.4 | Luči in vtičnice..... | 31 |
| 4.4.1 | Izklop vseh luči | 31 |
| 4.4.2 | Vodenje luči | 31 |
| 4.4.3 | Vodenje žaluzij..... | 37 |
| 4.4.4 | Nastavitve urnikov za žaluzije | 39 |
| 4.5 | Upravljanje z mobilnim telefonom | 43 |
| 4.6 | Upravljanje z daljinskim upravljalnikom | 43 |
| 4.7 | Ogrevanje | 44 |
| 4.8 | Uporabniški vmesnik - zaslon občutljiv na dotik..... | 47 |
| 4.8.1 | Osnovna stran | 47 |
| 4.8.2 | Nadstropje in mansarda..... | 47 |
| 4.8.3 | Ambienti..... | 49 |
| 4.8.4 | Nastavitve..... | 50 |
| 4.8.5 | Nastavitve – hiša | 50 |
| 4.8.6 | Nastavitve – ura/datum..... | 51 |
| 4.8.7 | Nastavitve – ogrevanje | 52 |
| 4.8.8 | Nastavitve – urnik žaluzije | 52 |
| 5. | Sklepne ugotovitve..... | 55 |
| 6. | Viri | 57 |

Povzetek

Diplomska naloga vsebuje opis pametne hiše in ustreznega sistema za njeno vodenje in nadzor. Namen naloge je predstaviti strojno in programsko opremo, razmišljati o njeni uporabnosti ter o prednostih in slabostih takega sistema. V delu je predstavljeno razmišljanje o pametni hiši z vidika varnosti, udobja, varčevanja, uporabnosti za osebe s posebnimi potrebami ter cenovni dostopnosti. Sistem pametne hiše prinaša v vsakodnevno življenje večjo varnost, več udobja, osebam s posebnimi potrebami pa omogoča kvalitetnejše in bolj samostojno življenje.

Z vidika varčevanja sistemi ponujeni na slovenskem trgu zaenkrat ne prinašajo večjih prihrankov. Trenutno je pametna hiša cenovno dostopna le premožnejšim in tistim navdušencem, ki sami sprogramirajo krmilje in s tem bistveno zmanjšajo stroške investicije. Pregledana je ponudba na slovenskem trgu, na kratko pa so opisane rešitve štirih podjetij.

Jedro dela je namenjeno predstavitvi razvoja sistema za vodenje in nadzor pametne hiše. Sledi opis programske kode, delovanje in opis krmilne opreme, s katero je bil realiziran projekt pametne hiše ter protokola LON, ki se uporablja za prenos podatkov med krmilniki. Uporabniki sistem uporabljajo dva meseca in so z njim zadovoljni. Večjih težav s prilagajanjem na sistem niso imeli, saj je uporabniku prijazen in enostaven za uporabo.

Ključne besede:

- pametna hiša
- avtomatizacija
- krmilnik
- vodenje
- protokol LON

Abstract

The thesis describes a smart house and the system that empowers it with intelligence. The goal of the thesis is to present the hardware and software involved and debate about the usability, pros and cons of such a system. The work addresses the smart house from several viewpoints: safety, comfort, economy, accessibility for people with special needs and affordability. Using smart house makes every day's life more secure and comfortable, while it also enables independency to the people with special needs.

After reviewing the offer on the Slovenian smart house market, the solutions offered by four different companies are summarized. For the present, market prices of smart houses are prohibitive to people with average incomes, except for the enthusiasts that have knowledge of programming the controller hardware.

The main part of the thesis is devoted to the development of a smart house control system. Included is the description of the controller hardware, LON protocol and the source code with which the smart house project is realized. Customers that use the smart house for two months now are content with it. While the system is user friendly, there were no major problems in adaption to it.

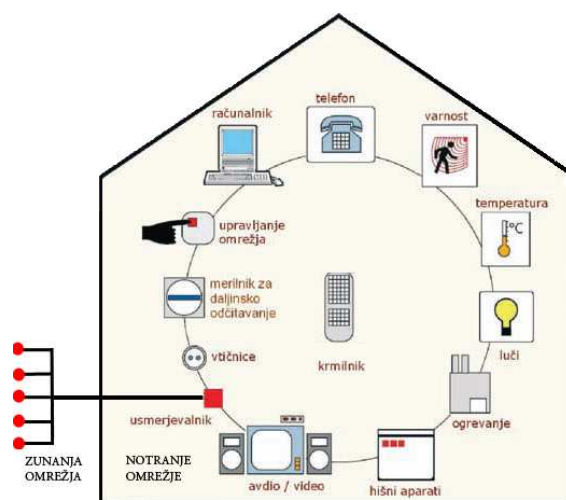
Keywords:

- smart house
- automation
- controller
- management
- LON protocol

1. Uvod

Pametna hiša je hiša, ki povezuje manjše podsisteme za upravljanje v celovit in enostaven upravljalni in nadzorni sistem. Hiša omogoča dodatne funkcije za izboljšanje kakovosti bivanja, povečanje varnosti in varčevanje z energijo. Začetki pametne hiše segajo v leto 1989, ko so na Japonskem avtomatizirali hišo imenovano pametna hiša Tron [1]. Hišo so avtomatizirali s pomočjo kar 380 računalnikov, ki so skrbeli za upravljanje in nadzor nad razsvetljavo, temperaturno regulacijo, odpiranjem vrat in podobno. Računalniki so bili povezani v arhitekturo Tron, ki je odprt sistem v realnem času. V tem času je bila tehnologija predraga za komercialno uporabo, saj je pametna hiša Tron stala kar 8 milijonov evrov. Z napredkom tehnologije so se odpirale nove možnosti, ki so bile cenovno veliko bolj dostopne. Možnosti za avtomatizacijo so skoraj neomejene in jih lahko prepuščamo svoji domišljiji, potrebam in zmožnostim. Cene sistemov, ki dajo hiši inteligenco, se začnejo pri nekaj tisoč evrih, navzgor pa skoraj ni omejitev. Ustanovitelj Microsofta Bill Gates je porabil več kot 100 milijonov evrov, da je zgradil in opremil svojo pametno hišo [2]. Skoraj polovico denarja je namenil za dodatno opremo in tehnologijo nameščeno v hiši. Med drugim računalniški sistem neprekinjeno nadzira in spremlja vlažnost zemlje okoli drevesa in ga zalije po potrebi. Klima se nastavlja samodejno, tako da so nastavitve najboljše za zdravje in dajejo občutek prijetnosti. Med sprehajanjem po hiši nas spremlja glasba, TV slika pa se prenaša na najbližji zaslon. Preko mobilnega telefona se lahko napolni kopalno kad z vodo, pri čemer se poljubno izbere količino in temperaturo vode. Poleg tega hiša ponuja še nešteto možnosti; nekatere so javnosti skrite in nedostopne.

Večina slovenskih podjetij, ki delujejo na tem področju, ponuja rešitve za razsvetljavo, žaluzije, urnike, ambiente, ogrevanje, zalivanje vrta, daljinsko ugašanje ali prižiganje naprav, protivlomno in požarno varnost, avdio in video distribucijo, hišni kino, prezračevanje hiše in še bi lahko naštevali. Uporabniki pričakujejo od svoje pametne hiše komunikacijo z zunanjim svetom, udobje, kakovostne bivalne razmere, varnost in racionalno porabo energije. Slika 1.1 prikazuje pametno hišo, simboli v njej pa prikazujejo, s čim vse pametna hiša lahko upravlja.

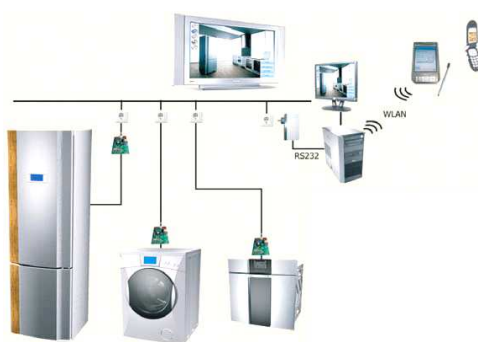


Slika 1.1: Pametna hiša [4]

Poleg že naštetih lastnosti, pa lahko tak sistem olajša življenje starejšim ljudem in ljudem s posebnimi potrebami. Prvi sistem namenjen ljudem s posebnimi potrebami, ki sem ga zasledil, je bil predstavljen leta 2000 v Ameriki [3]. Sistem je bil plod sodelovanja dobrodelne organizacije Dementia Voice, združenja Housing 21 in Bath Institute of Medical Engineering. Poudarek je bil na varnosti hiše in njeni pomoči pri iskanju različnih predmetov (torbice, očal, ključev ...). Predvidene pa so bile tudi razširitve kot so izklop kuhalnika, ko izhlapi vsa voda, preprečevanje puščanja odprtih pip, regulacija temperature v kopalni kadi in sledenje osvetlitve pri gibanju po hiši. Poleg omenjenega projekta pa sem našel tudi predstavitveni model pametne hiše Inštituta Republike Slovenije za rehabilitacijo, ki so ga poimenovali Dom iRiS (Inteligentne rešitve in inovacije za samostojno življenje). Namen takega doma je olajšati vsakodnevna opravila ljudem s posebnimi potrebami [4]. Dom iRiS omogoča tudi večjo samostojnost, večjo varnost in lažjo pomoč tem ljudem na daljavo. S takim domom, lahko zdravstveni delavci spremljajo stanje pacienta na domu s kamerami, senzorji in klici na pomoč, ki jih lahko pošlje hiša avtomatsko ali pa jih pošlje uporabnik. Predstavitvena hiša je bila širši javnosti predstavljena konec leta 2007.

Z razvojem brezžičnih tehnologij, ki počasi prodirajo tudi v sisteme pametnih hiš, se odpira nova opcija za lastnike starejših objektov, ki bi želeli spremeniti svojo hišo v pametno hišo. Z brezžično tehnologijo se lahko izognemo napeljevanju dodatne napeljave in s tem nepotrebnim in neprijetnim gradbenim posegom. Eno od slovenskih podjetji, ki ponuja sistem z brezžičnimi rešitvami, je podjetje KomfortKlik [5].

Razvoj opreme gre v smeri, da bomo lahko nekoč imeli vse naprave v hiši povezane z enim sistemom, s katerim bo mogoče spremljanje in spreminjanje njihovega stanja in podobno. To bi napravam povečalo njihovo uporabno vrednost in poenostavilo njihovo upravljanje, saj bi imeli en preprost uporabniški vmesnik. Povezanost naprav prikazuje slika 1.2. Slovenska podjetja Telekom, Goap in Gorenje sodelujejo na tem področju [6]. Podjetje Gorenje ima že od leta 2004 v svojem prodajnem programu tudi aparate, ki sledijo tehnološkim rešitvam s področja okoljske inteligence in povsod prisotnih osebnih računalnikov. Gre za vodenje procesov, lokalno in daljinsko upravljanje naprav ter inteligentne senzorje. Gorenjeva partnerja Telekom in Goap razvijata svoje rešitve v podobni smeri. Goap razvija uporabniške vmesnike in module ter druge rešitve, ki omogočajo povezljivost s telekomunikacijskim in internetnim omrežjem. Telekom pa poleg omrežnih storitev potrebnih za te namene, omogoča tudi druge storitve pametne hiše.



Slika 1.2: Povezljivost naprav [2]

Tu velja omeniti, da je francosko podjetje Honeywell že leta 1969 naredilo prvi računalnik namenjen kuhinji, veliko pred pojavom prvega osebnega računalnika IMB leta 1981[7]. Honeywell-ov računalnik H316, ki ga prikazuje slika 1.3, je bil zmožen shranjevati recepte in imel vgrajeno desko za rezanje. Za pravilno uporabo je bilo potrebno dvotedensko uvajanje. Glavni razlog za uvajanje je bilo to, da računalnik ni imel ekrana, ampak samo vrsto utripajočih LED diod, ki so predstavljale binarni zapis (00010011 pomeni 2 žlici sladkorja, 11010101 pa pol kilograma moka). Zaradi zahtevnega uporabniškega vmesnika niso prodali nobenega kosa. To pa ni bila edina težava, saj je bil to ogromen računalnik, ki je tehtal 50 kilogramov, stal pa je takratnih 10.500 ameriških dolarjev.



Slika 1.3: Računalnik Honeywell 316

Pametna hiša nam poleg večjega udobja omogoča tudi varčevanje z energijo in nudi večjo varnost. Mnoge njene funkcionalnosti pa lahko s pridom izkoristijo tudi ljudje s posebnimi potrebami. V nadaljevanju je pri pregledu omenjenih tem navedenih nekaj najbolj običajnih rešitev.

○ **UDOBJE:**

- Za udobje skrbijo ambienti, ki s spremembo jakosti razsvetljave, ugašanjem oziroma s prižiganjem določenih luči, dviganjem in spuščanjem žaluzij ustvarijo posebno atmosfero v nekem prostoru ter tako enostavno pričarajo okolje za različne priložnosti.
- Prižiganje in ugašanje luči, ambientov in drugih stvari s ploskanjem.
- Tipka ob postelji, ki spusti vse žaluzije, ugasne vse luči in aktivira alarme.
- Tipka ob izhodu iz hiše, ki izklopi hišo in s tem vse luči, določenemu sklopu vtičnic odvzame napajanje in prižge alarm.
- Upravljanje celotne hiše z daljinskim upravljalnikom, stenskim ali prenosnim zaslonom ter računalnikom.

- Slika, ki pride iz kamere na vhodnih vratih nam pove, kdo je naš obiskovalec. Tako lahko odpremo vhodna vrata kar iz naslonjača.
 - Ko se vračamo pozimi iz daljšega dopusta, lahko preko interneta ali mobilnega telefona prižgemo ogrevanje hiše, tako da se vrnemo v prijetno ogreto hišo.
 - Samodejno zalivanje vrta ob določeni uri, če je to potrebno glede na vlažnost zemlje.
 - Avtomatsko prižiganje luči glede na zunanjo svetlobo ali senzorje gibanja.
 - Nastavljanje glasnosti glasbe po posameznih prostorih, spremljanje glasbe in TV slike med gibanjem po hiši.
 - Odklepanje vhodnih vrat z mobilnim telefonom (klicem, modrim zobom ali sporočilom SMS), s prsnim odtisom, z brezkontaktnimi karticami ...
- **VARČEVANJE:**
- Ogrevamo lahko vsako sobo posebej glede na želje in potrebe. Na podlagi zelene in izmerjene temperature v sobi pametna hiša ustrezno uravnava ventile ogrevalnega sistema.
 - Drugi način varčevanja z ogrevanjem je ekonomično ogrevanje, ki je uporabno takrat, ko nas čez dan ni v hiši, zvečer pa bi se želeli vrniti v prijetno ogreto stanovanje. V tem primeru ogrevanje stanovanja poteka z manj zahtevnimi parametri.
 - Privarčujemo tudi z urniki za žaluzije. Poleti spustimo žaluzije oken, ki jih obsije močno popoldansko sonce in s tem preprečimo segrevanje stanovanja. Pozimi pa se vse žaluzije avtomatsko spustijo ob določeni uri ali ob mraku. S tem zmanjšamo oddajanje temperature v ozračje.
 - Izklop klime, ko so odprta balkonska vrata ali okno. Opozarjanje na odprta vrata in okna med ogrevanjem stanovanja.
- **VARNOST:**
- Nadzor nad varnostnimi kamerami ter detektorji gibanja okoli nepremičnine.
 - Varnostni sistem lahko poroča o slabo zaprtih oknih in vratih, kot tudi o nepooblaščenih uporabi določenih funkcij.
 - Senzorji za ogenj in dim ter protipoplavni senzorji v kletih.

- Nadzor je mogoč lokalno in daljinsko preko interneta ali mobilnega telefona.
 - Senzorji izlitja vode v kopalnici in sobi, kjer je pralni stroj.
 - Ko gremo na dopust in pustimo hišo samo, je ta zelo mamljiva za tatove. Nepriprava lahko pretentamo s spuščanjem in dviganjem žaluzij, prižiganjem in ugašanjem luči in tako dajemo občutek, da je nekdo v hiši.
 - Ob vedno bolj prisotnih vremenskih nevšečnostih (nevihtah s točo) lahko poskrbimo, da se okna zaprejo in žaluzije samodejno spustijo.
 - Ko zapustimo hišo gre ta v stanje mirovanja, ugasnejo se vse luči, odvzame se električno napajanje določenim vtičnicam in zapre se ventil za vodo.
 - Ob alarmih nas hiša preko sporočila SMS obvesti na mobilni telefon o aktivnih alarmih. Če so aktivni senzorji gibanja si lahko preko interneta pogledamo, kaj so posnele naše kamere.
- **LJUDJE S POSEBNIMI POTREBAMI:**
- Ljudem s posebnimi potrebami bi lahko tak sistem prinesel večjo samostojnost, olajšal vsakodnevna opravila, zmanjšal potrebo po odhodih v domove upokojencev, zmanjšal stroške zdravstvene oskrbe in jim zagotovil večjo varnost.
 - Urniki, ki se jih predhodno nastavi, da ljudi opomnijo na določene stvari (vzemi tablete, pojdi na pregled ...) z različnimi oblikami javljanja (preko sporočila SMS, govornega sporočila, spletne pošte).
 - Klic na pomoč, ki ga uporabnik sproži s pritiskom tipke na obesku zraven ključev, s tipko na steni, s klicem preko mobilnega telefona ...
 - S pomočjo interneta omogočimo spletno izobraževanje, spremljanje konferenc, dostop do bančnih podatkov, možnost nakupa različnih izdelkov, tudi živil in še bi lahko naštevali.
 - Samodejna identifikacija osebe z gibalnimi omejitvami, ostalih prebivalcev in obiskovalcev pametnega doma. Samodejno, uporabniku prilagojeno nastavljanje višine elementov prostora (računalniškega kotička, kuhinjskih elementov, kopalniških elementov).

Vsekakor se zdi, da pametna hiša lahko ponudi veliko več udobja in varnosti, kot pa prihrankov energije. Včasih celo težko govorimo o prihrankih, če želimo pridobiti na udobju in varnosti. A vendar pametna hiša lahko pripomore tudi k varčevanju z energijo. V manjši meri je varnost povezana z varčevanjem. Lahko bi rekli, da večja varnost lahko pomeni nižje

zavarovalne premije. Senzor izlitja vode takoj, ko zazna izlitje, vodo zapre, da ne pride do nepotrebnega trošenja vode. Ob nevihtah sistem zapre okna, spusti žaluzije in ob detekciji vode v kleti sam začne črpati vodo. Tako se omeji škoda na objektu in zniža stroške popravila.

Poseben poudarek na varčevanju z energijo pa bo pri novih gradnjah tudi zaradi zakona, ki ga je izdalo Ministrstvo za okolje in prostor. Pravilnik določa, da bo dovoljena le gradnja stanovanjskih in poslovnih zgradb, ki bodo porabile dvakrat manj energije kot sedanje zgradbe[8]. S pametno hišo bomo lažje dosegli cilje, ki nam jih narekuje zakon.

○ **OMEJITVE:**

Sistemi za avtomatizacijo hiše so še vedno precej dragi, saj se cene gibljejo od 6.000 EUR naprej. Za tako ceno se ponavadi dobi osnovni sistem, ki omogoča vodenje žaluzij in luči, z dodanimi ambientni in urniki. Če želimo sistemu dodati posebne funkcije, se cena hitro dviga. Na trgu pa lahko kupimo samo krmilje brez logike, kar lahko ceno zelo zniža. Če omejimo zahteve lahko dobimo za nekaj 100 evrov krmilnik, z vhodi, izhodi in zaslonom. Eden takih krmilnikov je VISION 350 podjetja Unitronics, ki pri nas stane okrog 700 evrov. To je priložnost za nekoga, ki zna programirati ali pa ima voljo in željo naučiti se in s tem dodati svojemu domu inteligenco. V primeru, da se bomo sami lotili programiranja, se moramo zavedati, da bomo porabili kar nekaj časa, preden bo sistem funkcionalen. Najprej je treba spoznati programsko okolje in ugotoviti kako se posamezni elementi obnašajo, kar pa včasih vzame veliko časa. Samostojno programiranje ima poleg varčevanja še eno dobro lastnost: lahko sprogramiramo kot smo si želeli in zamislili.

2. Pregled in ponudba sistemov na slovenskem trgu

Pregledal sem ponudbo na slovenskem trgu in na kratko opisal podjetja, ki se ukvarjajo s hišno avtomatizacijo. Pozanimal sem se, koliko stane podoben sistem, kot je opisan v diplomski nalogi z uporabo njihove opreme skupaj s programsko opremo.

Zahteve sistema:

- vodenje 21 navadnih luč,
- vodenje 7 luči s funkcijo zatemnitve,
- vodenje 14 žaluzij ročno ali po urniku,
- upravljanje in spremljanje hiše daljinsko preko mobilnega telefona po protokolu GSM,
- upravljanje hiše z daljinskim upravljalnikom,
- upravljanje z zaslonom občutljivim na dotik,
- vpisovanje urnikov,
- nastavljanje ambientov,
- nadzor nad ogrevalnim sistemom.

Podjetje Smarteh d.o.o. razvija lastne prosto programirne krmilnike LPC-2 MC3 in LPC-2 MC7. Krmilnik LPC-2 MC3 ima vgrajen mrežni vmesnik LON in nanj ni mogoče priključiti modula Ethernet. Krmilnik LPC-2 MC7 nima vgrajenega mrežnega vmesnika, ima pa možnost priključitve modula Ethernet in modula LON. Krmilnik LPC-2 MC7 je procesorsko zmogljivejši od svojega predhodnika.

Oba krmilnika sta grajena na modularen način [9]. To pomeni, da imamo eno osnovno enoto, na katero priključujemo razširitvene module; tiste vrste in toliko, kot jih potrebujemo. Osnova sistema LPC-2 je glavni kontrolni modul, kateremu se doda modul LON za komunikacijo z ostalimi krmilniki v sistemu (krmilnik MC3 ima že vgrajen vmesnik LON). Če želimo komunicirati z računalnikom, zaslonom ali dlančnikom, je potrebno krmilniku dodati modul Ethernet (omogoča samo krmilnik MC7). Programski jezik, v katerem pišemo aplikacijo, je kombinacija lestvičnega in blokovno orientiranega programiranja. Trenutno razvijajo nov program za pisanje aplikacije, ki naj bi prinesel veliko inovacij.

Podjetje Liko Pris d.o.o. se ukvarja s hišno avtomatizacijo že dobro desetletje. Uporabljamo krmilnike domačih in tujih proizvajalcev. Med drugim od domačih proizvajalcev uporabljamo krmilnike podjetja Smarteh d.o.o. Opisani sistem sem naredil s krmilniško opremo tega podjetja.

Cena zgoraj opisanega sistema s krmilniškim sistemom podjetja Smarteh d.o.o. in programsko rešitvijo podjetja Liko Pris d.o.o. stane približno 7.000 evrov.

Podjetje Robotina d.o.o. je razvilo lasten prosto programirni krmilnik CyBro [10]. Tudi krmilni sistem Cybro je grajen na modularen način. Ker so bile aplikacije vedno zahtevnejše, prva generacija krmilnikov ni več zadoščala, zato so razvili nov krmilnik imenovan CyBro 2.

Poleg svojih krmilnikov pa imajo v ponudbi tudi krmilnike tujih proizvajalcev, kot sta na primer podjetji Hima in Hitachi. Komunikacija med krmilnikom CyBro in ostalimi moduli vezanimi na ta krmilnik poteka po protokolu CAN, komunikacija med različnimi krmilniki pa poteka preko protokola Ethernet TCP/IP. Druga generacija krmilnikov ima vgrajen vmesnik Ethernet. Vmesnik omogoča priključitev drugih naprav, ki se pogovarjajo preko tega protokola (SCADA, Internet).

Razvili so lasten koncept pod imenom Integra BM (Integra building management system), ki temelji na najsodobnejši tehnologiji. S sistemom Integra BM je mogoče upravljati z vsemi napravami in aparati, ki so priključeni na električno napeljavo. Poleg tega poveča varnost in zmanjša energetske porabe. Podjetje je razvilo tudi programski paket SCADA imenovan Integra View. Paket omogoča spremljanje in upravljanje s hišo v realnem času in pregledovanje zgodovine podatkov. Programsko orodje omogoča tudi enostavno izdelavo spletno podprtih SCAD brez potrebe po posebnih znanjih HTML-ja ali Jave (za nadzor in upravljanje preko spleta).

Cena zgoraj opisanega sistema s programsko rešitvijo stane približno 10.000 evrov.

Podjetje KomfortKlik d.o.o. uporablja strojno opremo drugih proizvajalcev. Sami opremo povežejo in jo sprogramirajo. Sistem KomfortKlik vključuje odprte standarde EIB/KNX in EnOcean. Prvi standard se uporablja pri žičnih sistemih, drugi pa pri brezžičnih sistemih. Oba sistema sta med seboj povezljiva. Prvi sistem se uporablja predvsem v novogradnjah, drugi sistem pa se vgrajuje v že obstoječe hiše. V njihovih poslovnih prostorih si lahko pogledamo predstavitveni model pametne hiše.

Cena zgoraj opisanega sistema s programsko rešitvijo stane približno 6.000 evrov.

3. Opis krmilnika LPC-2 MC7

Krmilniki so srce sistema, saj skrbijo za izvajanje programa in za izmenjavo podatkov z vhodno-izhodnimi moduli. Krmilniki se napajajo iz električnega omrežja, vhodno-izhodni moduli pa iz krmilnika, na katerega so priključeni. Za komunikacijo med krmilnikom in drugimi moduli vezanimi na ta krmilnik skrbi interno vodilo. Komunikacija med krmilniki poteka preko modulov LON, ki so povezani v omrežje LON. Na vsak krmilnik lahko priključimo do dva modula LON. Če želimo, da modul LON komunicira z več moduli se uporabi razdelilnik SPL-2, na katerega lahko priklopimo do osem naprav, ki komunicirajo preko omrežja LON. Vsak modul lahko komunicira z vsakim. Krmilniki imajo lahko poleg modula LON tudi modul Ethernet, ki skrbi za komunikacijo preko protokola Ethernet TCP/IP. Na krmilnik sta lahko priključena do dva modula Ethernet. Za razširitev mreže lahko uporabimo stikalo.

Krmilniki imajo troje vrat za povezavo z drugimi napravami. Prva vrata, imenovana COM1, so namenjena komunikaciji po protokolu RS-232. Na vrata COM1 lahko povežemo le eno napravo. Običajno je to računalnik, ki se obnaša kot gospodar, krmilnik pa kot suženj. Kanal je izmenično dvosmerni (angl. half duplex), kar pomeni, da sistem zagotavlja komunikacijo v obe smeri, vendar ne hkrati. Kadar ena stran sprejema signal, mora počakati, da oddajnik konča z oddajanjem, šele nato lahko odgovori. Hitrost komunikacije je 19200 bitov/sekundo. Pošilja se osem podatkovnih bitov, ki se zaključijo z enim stop bitom. Komunikacija nima ne preverjanja paritete, ne kontrole pretoka, ima pa dvakratno preverjanje poslanega oziroma sprejetega okvirja. Vrata COM1 se uporablja za spremljanje in nastavljanje lokalnih spremenljivk na krmilniku. Med drugim za nastavitev naslova IP in naslova MAC. Druga vrata so imenovana COM2. Na vrata COM2 lahko priklopimo do štiri specialne module iz programa LPC-2. Naprave, ki so povezane na ta vrata, se pogovarjajo preko protokola RS-485. Ker ima vsak krmilnik samo en vhod, uporabimo razdelilnik SPL-2, na katerega sicer lahko priklopimo do osem naprav. Da lahko procesor razlikuje podatke iz modulov, je potrebno vsakemu modulu nastaviti naslov z dvema preklopnima stikaloma na modulu. Dve preklopni stikali omogočata nastavitev štirih različnih naslovov specialnim modulom. Nekateri moduli ne omogočajo nastavljanja naslova, zato se lahko na krmilnik priklopi le en tak modul. Tretja vrata se imenujejo PROG in so namenjena izključno nalaganju programa na krmilnik.

Krmilniki imajo vgrajen pomnilnik, ki ga ločimo na tri dele. Prvi del se nahaja od naslova $0_{[16]}$ do naslova $2000_{[16]}$. To so sistemski naslovi in so nespremenljivi. Te naslove spremenljivkam dodeli programsko okolje LPC Composer ob konfiguraciji sistema. Krmilnik jih uporablja za svoje potrebe in za komunikacijo z napravami, ki so povezane nanj. V tem območju se nahajajo spremenljivke krmilnika, ki jih delimo na dva dela. Prve so v pomnilniku RAM, v katerem se informacija po izklopu napajanja izbriše. Druge pa so v pomnilniku remanentnega RAM-a, v katerem informacija ostane tudi po izklopu napajanja. Pomnilnik RAM je za potrebe vrat COM1 je razdeljen na štiri sklope spremenljivk pomnilnik EEPROM pa je razdeljen na dva sklopa spremenljivk. Imamo vhodni in izhodni RAM1, vhodni in

izhodni RAM2 ter vhodna EEPROM1 in EEPROM2. V vsakem RAM-u imamo šestnajst digitalnih spremenljivk ter deset analognih spremenljivk.

Drugi del se nahaja od naslova 2001_[16] do naslova 7FFF_[16] in se uporabljajo za spremenljivke, ki jih dodamo sami. Spremenljivkam lahko naslove določimo sami. Podobno je s tretjim delom pomnilnika, ki ima naslovno območje od naslova 8000_[16] do naslova EFFF_[16]. Namenjen je za spremenljivke, ki jih dodamo sami. Razlika med drugim in tretjim delom je v tem, da spremenljivke shranjene v tretjem delu ostanejo shranjene tudi po izpadu napajanja. V to območje se shranijo tudi spremenljivke modulov Ethernet. Spremenljivke dodajamo v programskih blokih, po več spremenljivk hkrati. Sistem omogoča do največ 64 pomnilniških blokov na enem krmilniku. Bloki so različno dolgi in lahko vsebujejo 8, 16, 32 ali 64 spremenljivk. Spremenljivke so lahko tipa bit, beseda ali dvojna beseda. Obstajajo tudi trije tipi blokov. Prvi tip bloka se začne s črko M (MWORD8) in se uporablja za dodatne spremenljivke, ki so nam v pomoč pri programiranju (območje od naslova 0100_[16] do naslova 0400_[16]). Tem spremenljivkam se samodejno določi naslov in ga ni mogoče spreminjati. Drugi tip bloka se začne s črko V (VWORD8) in se uporablja za spremenljivke vhodno-izhodnih in mrežnih modulov. Tretji tip bloka se začne s črko F (FWORD8) in se uporablja za shranjevanje spremenljivk tudi po izklopu napajanja (remanentni RAM, območje od naslova 8000_[16] do naslova EFFF_[16]).

Na krmilnike se modularno dodaja vhodno-izhodne module. Ker je število modulov na krmilniku in s tem število vhodov in izhodov omejeno, je potrebno imeti večje število krmilnikov, da imamo na voljo želeno število vhodno-izhodnih modulov. Na krmilnike so poleg vhodno-izhodnih modulov povezani še moduli za zatemnjevanje, modul GSM, moduli LON in moduli Ethernet. Dodajanje modulov na krmilnike ima določena pravila. Konfiguracijo preverimo s programskim orodjem LPC Composer. Če želimo dodati nov modul, nam program sam predlaga mesto, kamor ga lahko dodamo. Dodajanje modulov s pomočjo orodja LPC Composer je enostavno, saj program sam določi naslove spremenljivk na modulu. Tako ne pride do zmede v naslovnem prostoru spremenljivk. Spremenljivke se poimenujejo logično glede na modul. Na primer, če gre za spremenljivko modula Ethernet, se spremenljivka imenuje NE1_2_BRx101. Prvi del imena (NE1) nam pove, kateremu modulu pripada spremenljivka. Drugi del imena (2) nam pove, na katerem mestu je modul glede na krmilnik. V našem primeru je na drugem mestu, torej je med modulom NE1 in krmilnikom še en modul. Tretji del imena (BRx101) nam pove, da gre za prvi sprejemni bit (B - bit, R - sprejem (ang. receive), 101 - prvi bit).

3.1 Povezovanje krmilnikov - protokol LonTalk

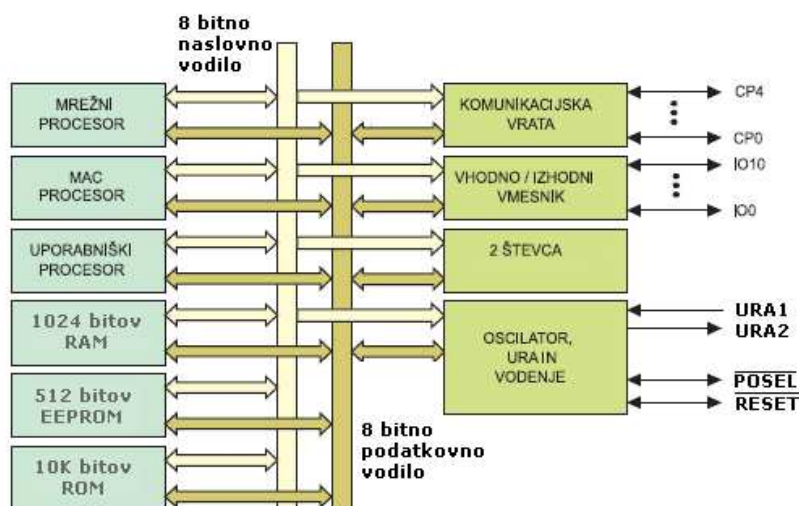
Sistem LonWorks je tehnološka rešitev korporacije Echelon, ki bazira na decentraliziranem principu vodenja ter učinkovito in univerzalno združuje kontrolno-nadzorne funkcije različnih mrežnih protokolov [11,12]. Kratico LON (ang. Local Operating Networks) bi lahko slovensko prevedli kot lokalno delujoča mreža, nad katero se izvajajo operacije. LON poveže procese med seboj tako, da ga uporabnik vidi kot eno logično celoto. Topologija omrežja LON ni določena v naprej. Omrežje LON je sestavljeno iz inteligentnih elementov, vozlov, ki

komunicirajo med seboj preko skupnega protokola, ki ga imenujemo protokol LonTalk. Protokol v vseh elementih ustreza 7-nivojskemu OSI referenčnemu modelu, podrobnosti so predstavljene v tabeli 1. Protokol LonTalk je vgrajen v pomnilnik vsakega od vozlov, ki je poleg komunikacije sposoben izvrševati tudi krmilne funkcije na ravni uporabnika.

Tabela 1: Protokol LonTalk in plasti modela OSI

| OSI Nivo | Pomen | servis LonTalk |
|----------------------|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| aplikacijska plast | združljivost na nivoju aplikacije | Določitev objektov, uporaba standardnih mrežnih spremenljivk, poenoteno programiranje in spuščanje v pogon. |
| predstavitvena plast | prepoznavanje | Prenos sporočil v standardnih okvirih. |
| sejna plast | operacije | Mehanizem podpira tudi delovanje: vprašanje - odgovor. |
| prenosna plast | zanesljivost | Prenos s prekinitvami ali brez. Naslavljanje posameznega vozla ali skupine. Označena sporočila. |
| mrežna plast | mrežno naslavljanje | Razširjanje podatkov po mreži, transparentni, nastavljivi in samoučni usmerjevalniki. 32385 vozlov po domeni, 2^{48} domen, 48 bitna enolična koda v vsakem vozlu |
| povezavna plast | prenos sporočil | Preverjanje okvirov. Kodiranje sporočil. 16 bitni CRC kot zaščita podatkov. Preprečevanje trkov sporočil na mreži z uporabo časovnih algoritmov. Uporaba sporočil, ki imajo prednost na mreži. |
| fizična plast | električne prenosne poti | Podpira vse komunikacijske medije. RS-485, vpleteni par, 230V mrežno napeljavo, infrardečo radio zvezo, koaksialni kabel, optične povezave. Hitrost prenosa je od 300 bit/s do 1.25 Mbit/s. |

Vozel v mreži LonWorks lahko izvršuje preproste ali pa tudi zapletene naloge. Tako so elementi, kot so različni senzorji, stikala, motorni pogoni in inštrumenti, lahko posamezni vozli v mreži, ki lahko vršijo kompleksne nadzorne in izvršilne funkcije. Osnova vsakega vozlišča LonWorks je poseben mikroprocesor imenovan Neuron, katerega blok shemo prikazuje slika 3.1. Zgrajen je podobno kot ostali mikrokontrolerji, le da vsebuje v enem vgrajenem vezju tri mikroprocesorje. Prva dva mikroprocesorja (mrežni mikroprocesor in mikroprocesor MAC) skrbita za komunikacijo med vozlišči. V teh dveh mikroprocesorjih poteka procesiranje komunikacije do šestega OSI nivoja. Do teh dveh mikroprocesorjev uporabnik nima dostopa. Iz aplikacije so nastavljivi samo določeni parametri, ki so potrebni za komunikacijo. Tretji mikroprocesor je uporabniški procesor. V njem se izvaja uporabniška aplikacija. Na podlagi vezja Neuron so različni proizvajalci iz celega sveta razvili veliko število LonWorks tehnoloških izdelkov, ki jih sistemski inženirji lahko uporabljajo kot gradnike.



Slika 3.1 Blok shema Neuron čipa [13]

Glede na razmerja med posameznimi deli mreže, podpira LonWorks komunikacijo točka-točka, komunikacijo gospodar suženj ter deterministični in mešani tip komunikacije. Računalnik v sistemu LonWorks lahko uporabimo za nadzor, opazovanje ali pa za upravljanje v realnem času. Računalnik deluje kot eden od vozlov na mreži.

Medsebojno komunikacijo vršijo vozli preko enega ali več medijev, pri čemer uporabljajo standardni protokol. Kontrolna mreža v LonWorks tehnologiji ima lahko največ 32385 vozlov znotraj enega medija, prenosnih medijev pa je lahko 2^{48} . Medije lahko med seboj enostavno povezujemo s prehodi. Ker imajo vozli vgrajeno določeno stopnjo inteligence, ima mreža boljšo porazdelitev procesne zmogljivosti, saj lahko podatke do neke mere obdelajo kar vozli sami in pošljejo v mrežo le rezultate svojih analiz. Tako porazdeljene kontrolne funkcije tudi izboljšajo zmogljivost in zanesljivost celotnega sistema. Prav v teh lastnostih pa se kontrolne mreže bistveno razlikujejo od podatkovnih, pri katerih nas najbolj zanima količina izmenjanih podatkov na mreži. Pri kontrolni mreži pa višje vrednotimo lastnosti kot so: zanesljivost, odzivnost in predvidljivost.

Prednosti protokola LonTalk:

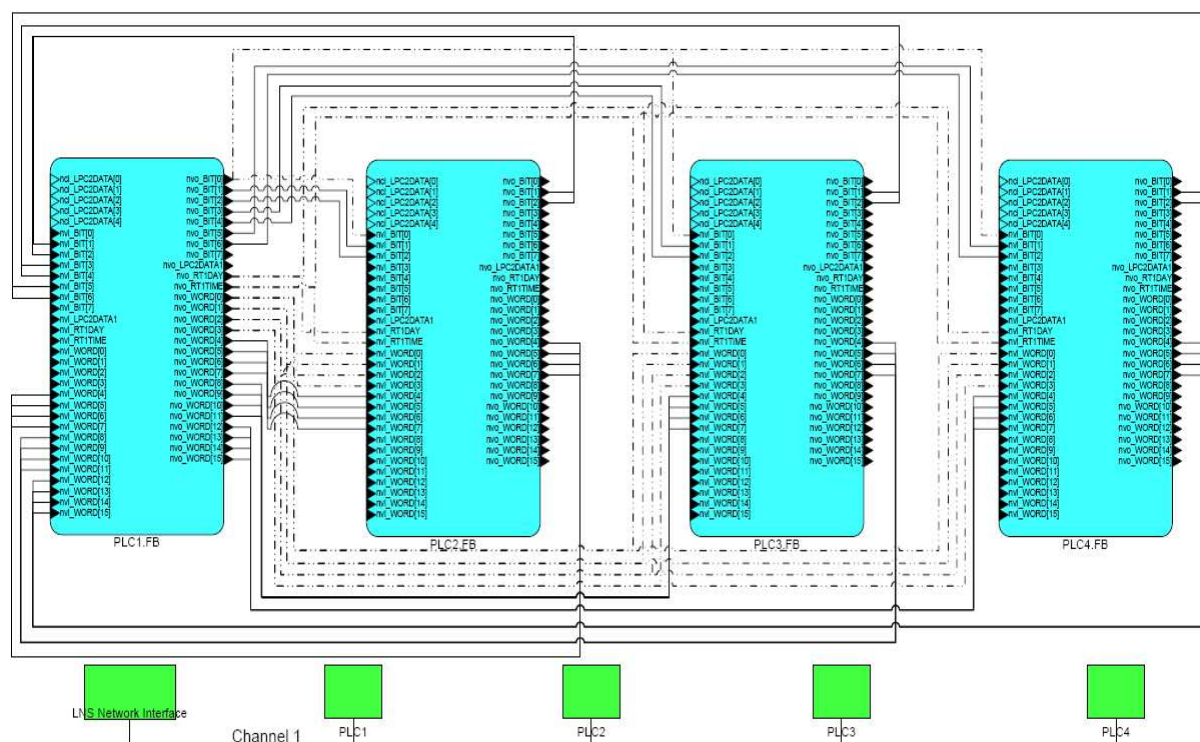
- Ob okvari ene enote ostale delujejo nemoteno, okrnjena je samo ena izmed funkcij celotnega sistema. Ostali deli sistema delujejo nemoteno naprej.
- Lažje nadgrajevanje in spreminjanje celotnega sistema. Obstoječemu sistemu dodajamo vozle, ne da bi s tem porušili že obstoječo strukturo.
- Prilagodljivost ter zmanjšanje stroškov inštalacij in vzdrževanja.
- Decentralizirana komunikacija - vsak z vsakim.
- Pogosto, zanesljivo in varno komuniciranje med napravami.
- Kratka sporočila, ki so poslana samo vozliščem, ki jih potrebujejo.

- Nizka cena, ki omogoča vgradnjo v cenene instrumente.
- Naprave se pogovarjajo med seboj po protokolu LonTalk ne glede na fizični komunikacijski medij (žica, 220V, infrardeča povezava, optika, radio zveza).
- Povezovanje mrež LON preko internetnih povezav omogoča mrežni strežnik iLON 1000.
- Veliko različnih elementov LonWorks, različnih dobaviteljev in veliko uporabniških vmesnikov ter orodij za integracijo in vzdrževanje, ki se lahko med seboj pogovarjajo. To nam omogoča izbor elementov po naših potrebah, cenovnih zmožnostih in željah.

3.1.1 Izmenjava podatkov med krmilniki po mreži LON

Vozlišča si med seboj pošiljajo vhodno-izhodne spremenljivke, ki jih definira programer. Določi katere izhodne spremenljivke se kam pošlje in katere vhodne spremenljivke se iz kje dobi. Na ta način vsak vozlel točno ve, kaj pomeni določena spremenljivka.

Vsa komunikacija po omrežju LON poteka preko prvega krmilnika (glavni krmilnik). Ta krmilnik pošilja in sprejema ukaze ostalih treh krmilnikov. Če drugi krmilnik zahteva neko akcijo od tretjega krmilnika, potem mora drugi krmilnik zahtevo najprej poslati prvemu krmilniku, ta pa jo posreduje tretjemu krmilniku. Odgovor poteka v obratni smeri preko prvega krmilnika. Vsak modul LON ima osem bitnih vhodnih in izhodnih telegramov ter šestnajst besednih vhodnih in izhodnih telegramov. Da se lahko pošilja vse potrebne informacije med krmilniki, je potrebo pametno razdeliti vhode in izhode. Slika 3.2 prikazuje kako so definirani telegrami za komunikacijo med krmilniki. Telegrami definiramo s pomočjo programskega orodja LONMaker, s katerim določimo, kateri sklop bitov oziroma besed se bo kam pošiljal.



Slika 3.2: Pošiljanje besed in bitov med krmilniki po mreži LON

V nadaljevanju pomeni $nvO_bit_PLC^*$ izhodni bit določenega krmilnika, $nvI_bit_PLC^*$ vhodni bit določenega krmilnika, nvO_word_PLC izhodno besedo določenega krmilnika in nvI_word_PLC vhodno besedo določenega krmilnika (* nadomestimo s številko krmilnika). V oglatih oklepajih so napisani bitni in besedni telegrami, ki se pošiljajo. Vsak bitni telegram vsebuje 16 bitov. Besedni in bitni telegrami se začnejo šteti z 0.

Komunikacija preko mreže LON med prvim in drugim krmilnikom:

- $nvO_bit_PLC1 [0] \rightarrow nvI_bit_PLC2 [0]$

Prvi krmilnik pošilja nulti izhodni bitni telegram drugemu krmilniku na nulti vhodni bitni telegram. Prvi krmilnik pošilja nulti izhodni bitni telegram še tretjemu in četrtemu krmilniku.

- $nvO_bit_PLC1 [1, 2] \rightarrow nvI_bit_PLC2 [1, 2]$

Prvi krmilnik pošilja prvi in drugi izhodni bitni telegram drugemu krmilniku na prvi in drugi vhodni bitni telegram.

- $nvO_bit_PLC2 [1, 2] \rightarrow nvI_bit_PLC1 [1, 2]$

Drugi krmilnik pošilja prvi in drugi izhodni bitni telegram prvemu krmilniku na prvi in drugi vhodni bitni telegram.

- $nvO_word_PLC1 [0, 1, 2, 3] \rightarrow nvI_word_PLC2 [0, 1, 2, 3]$

Prvi krmilnik pošilja nulti, prvi, drugi in tretji izhodni besedni telegram drugemu krmilniku na nulti, prvi, drugi in tretji vhodni besedni telegram. Prvi krmilnik pošilja nulti, prvi, drugi in tretji izhodni besedni telegram še tretjemu in četrtemu krmilniku.

- $nvO_word_PLC1 [4, 5, 6, 7] \rightarrow nvI_word_PLC2 [4, 5, 6, 7]$

Prvi krmilnik pošilja četrti, peti, šesti in sedmi izhodni besedni telegram drugemu krmilniku na četrti, peti, šesti in sedmi vhodni besedni telegram.

- $nvO_word_PLC2 [4, 5, 6, 7] \rightarrow nvI_word_PLC1 [4, 5, 6, 7]$

Drugi krmilnik pošilja četrti, peti, šesti in sedmi izhodni besedni telegram prvemu krmilniku na četrti, peti, šesti in sedmi vhodni besedni telegram.

Komunikacija preko mreže LON med prvim in tretjim krmilnikom:

- $nvO_bit_PLC1 [0] \rightarrow nvI_bit_PLC3 [0]$

Prvi krmilnik pošilja nulti izhodni bitni telegram tretjemu krmilniku na nulti vhodni bitni telegram.

- $nvO_bit_PLC1 [3, 4] \rightarrow nvI_bit_PLC3 [1, 2]$

Prvi krmilnik pošilja tretji in četrti izhodni bitni telegram tretjemu krmilniku na prvi in drugi vhodni bitni telegram.

- $nvO_bit_PLC3 [1, 2] \rightarrow nvI_bit_PLC1 [3, 4]$

Tretji krmilnik pošilja prvi in drugi izhodni bitni telegram prvemu krmilniku na tretji in četrti vhodni bitni telegram.

- $nvO_word_PLC1 [0, 1, 2, 3] \rightarrow nvI_word_PLC3 [0, 1, 2, 3]$

Prvi krmilnik pošilja nulti, prvi, drugi in tretji izhodni besedni telegram tretjemu krmilniku na nulti, prvi, drugi in tretji vhodni besedni telegram.

- $nvO_word_PLC1 [8, 9, 10, 11] \rightarrow nvI_word_PLC3 [4, 5, 6, 7]$

Prvi krmilnik pošilja osmi, deveti, deseti in enajsti izhodni besedni telegram tretjemu krmilniku na četrti, peti, šesti in sedmi vhodni besedni telegram.

- $nvO_word_PLC3 [4, 5, 6, 7] \rightarrow nvI_word_PLC1 [8, 9, 10, 11]$

Tretji krmilnik pošilja četrti, peti, šesti in sedmi izhodni besedni telegram prvemu krmilniku na osmi, deveti, deseti in enajsti vhodni besedni telegram.

Komunikacija preko mreže LON med prvim in četrtim krmilnikom:

- $nvO_bit_PLC1 [0] \rightarrow nvI_bit_PLC4 [0]$

Prvi krmilnik pošilja nulti izhodni bitni telegram četrtemu krmilniku na nulti vhodni bitni telegram.

- $nvO_bit_PLC1 [5, 6] \rightarrow nvI_bit_PLC4 [1, 2]$

Prvi krmilnik pošilja peti in šesti izhodni bitni telegram četrtemu krmilniku na prvi in drugi vhodni bitni telegram.

- *nvO_bit_PLC4 [1, 2] → nvI_bit_PLC1 [5, 6]*

Četrty krmilnik pošilja prvi in drugi izhodni bitni telegram prvemu krmilniku na peti in šesti vhodni bitni telegram.

- *nvO_word_PLC1 [0, 1, 2, 3] → nvI_word_PLC4 [0, 1, 2, 3]*

Prvi krmilnik pošilja nulti, prvi, drugi in tretji izhodni besedni telegram četrtemu krmilniku na nulti, prvi, drugi in tretji vhodni besedni telegram.

- *nvO_word_PLC1 [12, 13, 14, 15] → nvI_word_PLC4 [4, 5, 6, 7]*

Prvi krmilnik pošilja dvanajsti, trinajsti, štirinajsti in petnajsti izhodni besedni telegram prvemu krmilniku na četrty, peti, šesti in sedmi vhodni besedni telegram.

- *nvO_word_PLC4 [4, 5, 6, 7] → nvI_word_PLC1 [12, 13, 14, 15]*

Četrty krmilnik pošilja četrty, peti, šesti in sedmi izhodni besedni telegram prvemu krmilniku na dvanajsti, trinajsti, štirinajsti in petnajsti vhodni besedni telegram.

4. Opis sistema

Sistem za upravljanje in nadzor hiše, ki sem ga naredil v okviru diplomskega dela, omogoča ugašanje in prižiganje vseh luči v hiši s tipkami, daljinskim upravljalnikom in preko zaslona občutljivega na dotik. Določenim lučem v hiši se lahko nastavlja tudi intenzivnost osvetlitve. Žaluzije lahko spuščamo in dvigamo preko tipk, zaslona ali daljinskega upravljalnika. Poleg ročnega upravljanja sistem omogoča tudi avtomatsko spuščanje in dviganje žaluzij po urnikih. Del žaluzij in luči lahko upravljamo tudi z izbirnimi ambientmi, ki pričarajo različna vzdušja. Sistem skrbi še za ogrevanje bivalnih prostorov glede na dejanske potrebe po sobah in za varovanje pred zmrzovanjem ob daljši odsotnosti lastnikov. Ko je hiša v ekonomičnem režimu, sistem vzdržuje temperaturo v stanovanju v predhodno določenih mejah, ki niso tako stroge, kot če bi bili v hiši. S hišo lahko upravljamo in spremljamo njeno stanje tudi preko mobilnega telefona.

4.1 Delovanje sistema

Izklop hiše

Ob izklopu hiše sistem skrbi samo še za ogrevanje (dve funkciji) in za vklop luči parkirišča in hišne številke. Vse ostale funkcije se onemogočijo. Ko gre hiša v stanje izključeno, se po določenem času ugasnejo vse luči v stanovanju. Na hodniku se ugasnejo po minuti in pol, na stopnicah pa po dveh minutah. Vse ostale luči se ugasnejo takoj po izklopu hiše. Luči na hodniku in stopnicah se ugasnejo z zakasnitvijo, da lahko varno zapustimo hišo. Ob izklopu hiše ostanejo porabniki na vtičnicah brez napajanja (razen hladilnika, videorekorderja, akvarija). Če želimo spustiti ali dvigniti vse žaluzije, se to izvede s posebnim ukazom pred izklopom hiše. Ko je hiša izključena, ne upošteva nobenega ukaza, razen ukaza za vklop hiše in vklop ali izklop klime. Hišo lahko vklopimo preko mobilnega telefona, zaslona in daljinskega upravljalnika. Klimo lahko vklopimo preko mobilnega telefona in daljinskega upravljalnika. Luči na hodniku in stopnicah lahko prižgemo tudi takrat, ko je hiša izklopljena. S tem si osvetlimo pot do zaslona ali daljinskega upravljalnika ter vključimo hišo.

Navadne luči, programirani ambient

Luči upravljamo s tipkami, daljinskim upravljalnikom in zaslonom. Za vklop ali izklop luči s tipko je dovolj kratek pritisk, za vklop preko zaslona pa mora pritisk na gumb trajati približno eno sekundo. Določene luči se lahko prižiga tudi preko daljinskega upravljalnika. Pritisk na tipko, ki spremeni stanje luči, je lahko kratek, luč pa se prižge z majhno zakasnitvijo. Vse luči v hiši lahko izključimo z dvema ukazoma. Prvi ukaz – izklop hiše je že opisan zgoraj, drugi ukaz pa je izklop vseh luči. Ta ukaz v trenutku ugasne vse luči v stanovanju (razen luči parkirišča in osvetlitev hišne številke).

Ambienti so prednastavljene kombinacije razsvetljave in žaluzij. Ambienti vplivajo na luči in žaluzije v dnevni sobi, kuhinji in hodniku. Vsak ambient vpliva na vse luči in žaluzije, ki jih zajemajo ambient. Nek ambient torej prižge luč, jo ugasne ali pa ji nastavi določeno intenzivnost osvetlitve. Enako velja za žaluzije. Žaluzije dvigne ali pa jih spusti.

Sistem meri zunanjo osvetljenost. Na osnovi zunanje svetlobe se prižiga razsvetljava parkirišča in hišne številke.

Luči z možnostjo zatemnjevanja

Določenim lučem v hiši lahko nastavljamo intenzivnost osvetlitve: v spalnici, v zgornji kopalnici, na dveh lučeh v dnevni sobi, na terasi, v otroški in delovni sobi. Intenzivnost osvetlitve upravljamo s tipkami, s katerimi večamo ali manjšamo intenzivnost osvetlitve, jih vključimo in tudi izključimo (velja tudi za zaslon in daljinski upravljalnik). Luč z možnostjo zatemnjevanja prižgemo in ugasnemo s kratkim pritiskom na tipko ali kratkim pritiskom na tipko na daljinskem upravljalniku. Če je pritisk daljši, se začne spreminjati intenzivnost osvetlitve luči. Prvi daljši pritisk pomeni višanje intenzivnosti osvetlitve, drugi daljši pritisk pa pomeni nižanje intenzivnosti. Na zaslonu se luč prižge ali ugasne z daljšim pritiskom na gumb (približno 1 sekundo). Intenzivnost osvetlitve luči nastavljamo s skalo (višanje in nižanje) in na koncu pritisnemo gumb za potrditev.

Ogrevanje

Za regulacijo ogrevanja skrbijo termostati po sobah. Temperaturo se nastavlja lokalno po posameznih prostorih, krmilijo pa se elektromagnetni ventili posameznih ogrevalnih vej. Izjema je primer, ko je hiša v stanju izključeno in je aktiven ekonomični režim. Ekonomični režim deluje centralno, glede na temperaturo v dnevni sobi in ne glede na nastavitve posameznih termostatov po prostorih.

Ko je hiša prazna, jo lahko izklopimo. Tedaj lahko vklopimo ekonomični način ogrevanja, ki ogreva hišo tedaj, ko temperatura v dnevni sobi pade pod 15 °C. Ta način nam ohranja temperaturo v hiši v naprej predvidenih temperaturnih okvirih, ki niso tako strogi, kot če bi bili v hiši. Enako kot ekonomični režim deluje tudi zaščita proti zmrzovanju. Razlika je v tem, da je ekonomični režim potrebno vklopiti, zaščita proti zmrzovanju pa je vedno aktivna. Tudi zaščita proti zmrzovanju deluje centralno in spremlja temperaturo v dnevni sobi. Ta način skrbi, da temperatura ne pade pod 5 °C. Ko temperatura pade pod 5 °C se vklopi ogrevanje celotne hiše, na mobilni telefon pa prejmemo obvestilo o alarmu.

Vodenje žaluzij

Žaluzije lahko vodimo s tipkami, s pomočjo daljinskega upravljalnika ali zaslona. Žaluzije se krmilijo tudi avtomatsko v naprej predvidenih urnikih. Urniki omogočajo avtomatsko dviganje in spuščanje žaluzij: poleti popoldan, da preprečimo segrevanje stanovanja in pozimi v nočnem času, da se zmanjša oddajanje toplote v okolico. Pozimi se spuščajo vse žaluzije, razen žaluzij na stopnišču. Poleti pa se spuščajo samo žaluzije v dnevni sobi, otroški sobi in del žaluzij v kuhinji. Ko je hiša izključena, se žaluzije ne premikajo po urniku, ampak ostanejo v trenutnem položaju. Na zaslonu je posebna stran za nastavljanje parametrov za avtomatsko spuščanje in dviganje žaluzij. Urnike lahko omogočimo ali onemogočimo, vklopimo poletni ali zimski režim ter nastavimo čas spuščanja in dviganja.

Zaslon za upravljanje

Sistem lahko upravljamo in spremljamo preko zaslona občutljivega na dotik. Na zaslonu se prikazujejo vsi alarmi, vsa obvestila in delovanje naprav. Zaslon se nahaja v prvem nadstropju, kjer se največ zadržujemo. Na zaslonu se nastavlja ura, datum, nastavitve za urnike in ogrevanje. Preko zaslona lahko upravljamo z vsemi lučmi in žaluzijami.

Upravljanje z mobilnim telefonom

Modul GSM C05 je namenjen upravljanju naprav z razdalje preko mobilnega telefona. S pomočjo sporočil SMS lahko spremljamo stanje naprav in jim posredujemo ukaze. Z modulom CO5 lahko daljinsko vklopimo ali izklopimo ogrevanje (vklop ali izklop hiše) in klimatsko napravo. Zahtevo posredujemo s poslanim sporočilom SMS na telefonsko številko kartice SIM, ki je v modulu GSM. Za vklop hiše je potrebno poslati sporočilo SMS z vsebino #o1, za izklop hiše pa potrebno poslati #f1. Za vklop klime sporočilo SMS z vsebino #o2, za izklop pa #f2. S pomočjo sporočil SMS lahko spreminjamo določene nastavitve modula GSM. Na primer ukaz #r0 vklopi potrjevanje sprejema sporočila, #r1 pa izklopi. Ob prejemu sporočila #s modul GSM pošlje kot odgovor vsa digitalna in vsa analogna stanja. Ob prejemu sporočila #h pa modul GSM pošlje kot odgovor vse ukaze, ki jih modul sprejme (pozna približno 30 ukazov). Modul nas samodejno obvešča o aktivnih alarmih. Sistem ima definiran samo alarm proti zmrzovanju. Če temperatura v stanovanju pade pod 5 °C, sistem samodejno pošlje sporočilo SMS z vsebino Alarm zmrzovanje. Sporočila se sprejema in oddaja samo številkam, ki so shranjene na kartici SIM.

Delo z daljinskim upravljalnikom

V dnevni sobi se nahaja infrardeči sprejemnik, ki spremlja ukaze univerzalnega daljinskega upravljalnika. Z daljinskim upravljalnikom se upravlja z razsvetljavo in žaluzijami dnevne sobe, kuhinje in hodnika. Izbira se ambiente ter vklopi ali izklopi pametno hišo. Z istim daljinskim upravljalnikom lahko upravljamo še druge naprave kot so televizor, videorekorder in glasbeni stolp.

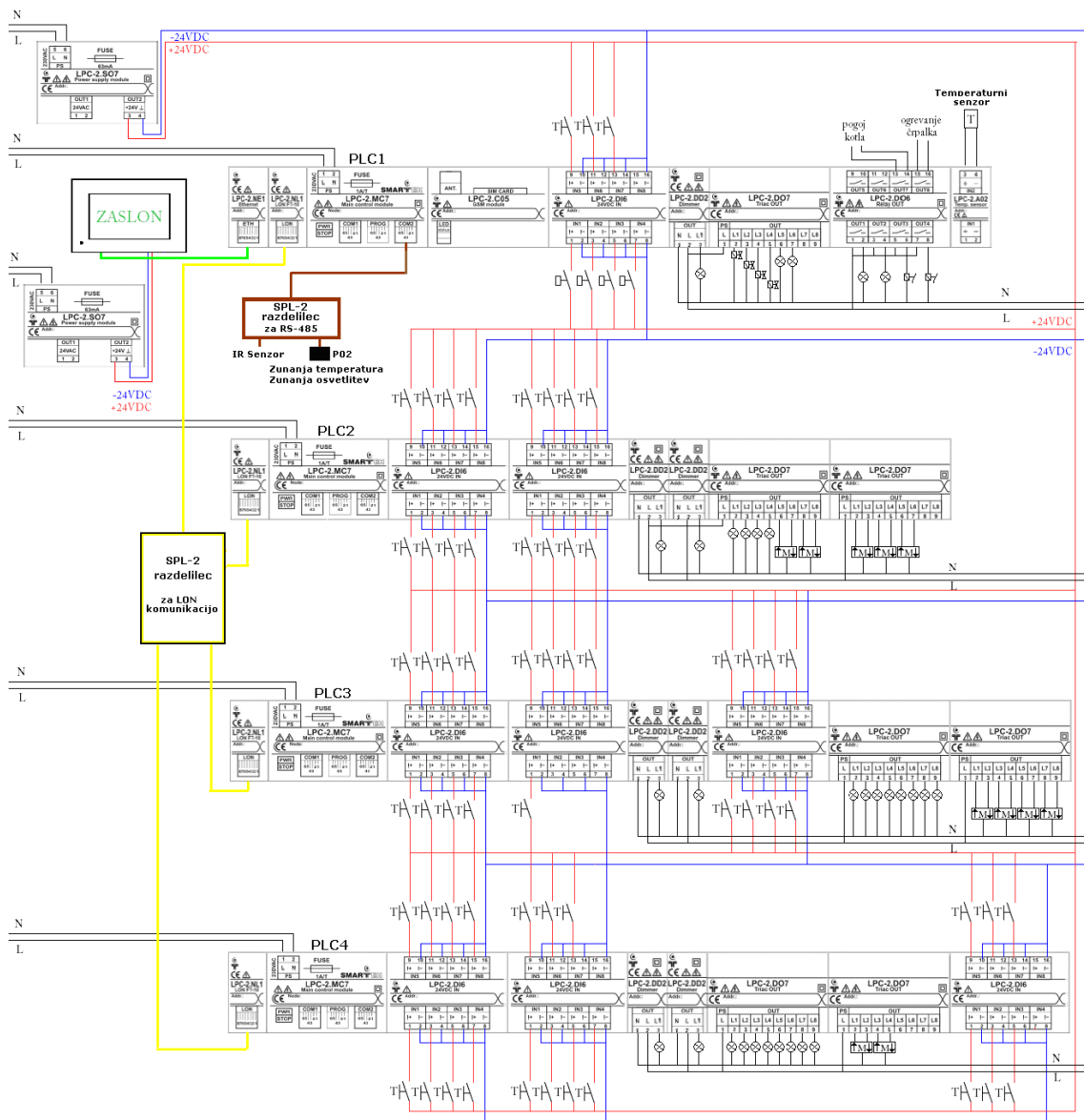
4.2 Shema sistema

Krmilni sistem je sestavljen iz naslednjih komponent:

- štirih krmilnikov, LPC-2 MC7,
- štirih modulov za komunikacijo v omrežju LON, LPC-2 NL1,
- modula za komunikacijo v omrežju Ethernet, LPC-2 NE1,
- devetih modulov s po 8 digitalnimi vhodi 24VDC, LPC-2 DI6, za zbiranje podatkov iz:
 - tipk za vodenje žaluzij (28),
 - tipk za prižiganje in ugašanje luči (26),
 - petih termostatov (5),
- dveh napajalnikov 24VDC, LPC-2 SO7, za napajanje:
 - zaslona (1),
 - tipk (1),
- sedmih modulov s po 8 triak izhodi 230VAC, LPC-2 DO7, za upravljanje:
 - izhodov za vodenje navadnih luči (19),

- izhodov za vodenje žaluzij (28 - žaluzija porabi dva izhoda; enega za gor in drugega za dol),
- modula z 8 relejskimi izhodi 230VAC, LPC-2 DO6, za upravljanje:
 - izhoda za pogoj kotla (1),
 - izhoda za obtočno črpalko (1),
 - izhodov za luči (2),
 - izhoda za vtičnice (1),
 - izhoda za ogrevanje (1),
- sedmih modulov za zatemnjevanje luči, LPC-2 DD2, za upravljanje:
 - izhodov za luči z možnostjo zatemnjevanja (7),
- modula z 2 analognima vhodoma, LPC-2 AO2, za zbiranje podatkov iz:
 - temperaturnega senzorja (1),
- modula za upravljanje preko telefona, GSM LPC-2 C05,
- temperaturnega upora NTC, LPC-2 NTC-4,
- modula za merjenje zunanje temperature in svetlobe LPC-2 PO2,
- infrardečega sprejemnika,
- dveh univerzalnih daljinskih upravljalnikov,
- zaslona občutljivega na dotik, Magelis XBT GT 1335.

Slika 4.1 prikazuje električno shemo sistema. Prikazano je, kako so različni moduli dodani krmilnikom. Omrežje LON je na shemi sistema označeno z rumeno povezavo. Prvi krmilnik poleg komunikacije z ostalimi krmilniki skrbi še za komunikacijo z zaslonom, ki poteka preko modula Ethernet. Povezava je označena z zeleno barvo. Vse zahteve iz zaslona se pošljejo na prvi krmilnik, ta pa preko omrežja LON pošlje ukaz ustreznemu krmilniku. Komunikacija med zaslonom, občutljivim na dotik, in prvim krmilnikom poteka po protokolu Ethernet TCP/IP. Poleg omenjenih dveh protokolov je uporabljen še protokol RS-485. Ta protokol se uporablja za komunikacijo med infrardečim sprejemnikom za daljinski upravljalnik, enoto P02 za merjenje zunanje temperature in osvetljenosti ter krmilnikom. V primeru, da je na vrata COM2 vezan razdelilnik SPL-2 in sta na njem več kot dve napravi, lahko govorimo o omrežju RS-485, ki ima topologijo zvezde. Komponente, ki komunicirajo preko protokola RS-485, so na sliki povezane z rjavo barvo.



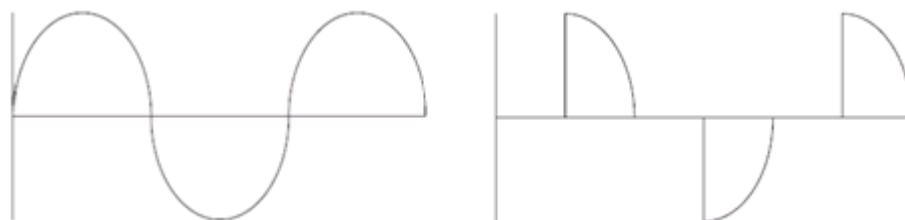
Slika 4.1: Povezovalna shema sistema

Tipke so vezane na vhodne module. Ob pritisku na tipko se vklopi ustrezen bitni vhod na vhodnem modulu. Ta bit sproži izvajanje določene logike, ki vklopi izhodni bit na izhodnem modulu. Navadne luči in vse žaluzije so vezane na triak izhodne module. Triak deluje kot izredno hitro stikalo, ki ob pozitivnem prožilnem impulzu priklopi breme (navadne luči in žaluzije) na 230V. Na prvi krmilnik je vezan poleg izhodnega modula s triak izhodi tudi izhodni modul z relejskimi izhodi. Na relejske izhode so vezane luči za razsvetljavo parkirišča in hišne številke ter pogoj za vklop klime, vtičnic, kotla in črpalke. Relejski izhodi so uporabljeni, ker so stvari, ki so priključene nanj, vezane na druge varovalke kot krmilnik.

Na modul AO2 je vezan upor NTC (ang. negative temperature coefficient). Uporu s segrevanjem upornost pada in s tem vhodna vrednost na modulu AO2. S tem uporom merimo temperaturo v dnevni sobi.

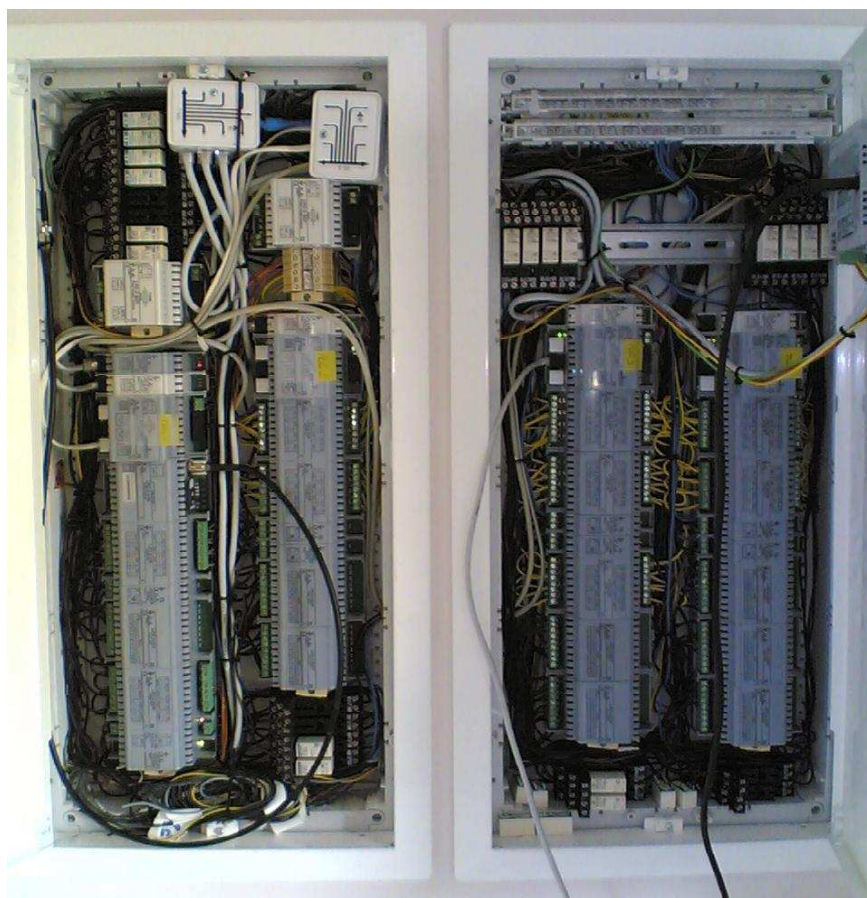
Sistem lahko nadgradimo z vgradnjo stikala med modul Ethernet NE1 in zaslon. Če bi imeli stikalo, bi nato lahko povezali še druge naprave, ki komunicirajo preko protokola Ethernet TCP/IP. Dodali bi lahko namizni računalnik, na katerem bi lahko teklen nadzorni sistem SCADA (ang. supervisory control and data acquisition), ki omogoča nadzor in upravljanje s sistemom. Poleg tega bi lahko dodali še dodatne zaslone ali kakšno drugo napravo.

Luči z možnostjo zatemnjevanja so vezane na module za zatemnjevanje. Moduli za zatemnjevanje režejo vhodni sinusni signal in dajo na izhod rezan sinusni signal, kot je prikazano na sliki 4.2. Več sinusa ostane, bolj sveti luč z možnostjo zatemnjevanja.



Slika 4.2: Vhod modula za zatemnjevanje (levo) in izhod modula za zatemnjevanje (desno)

Slika 4.3 prikazuje krmilnike, na katerih teče sistem. Do teh krmilnikov so pripeljani kabli tipk, luči, žaluzij, termostatov in senzorjev. Zaslon občutljiv na dotik se nahaja na vratih desne omarice.



Slika 4.3: Krmilni omari v pametni hiši

4.3 Programska koda v krmilnikih

Ker je programska koda enaka za vse luči, žaluzije in ostale dele programa sem opisal vsak del samo enkrat. Deli programske kode se razlikujejo samo v spremenljivkah.

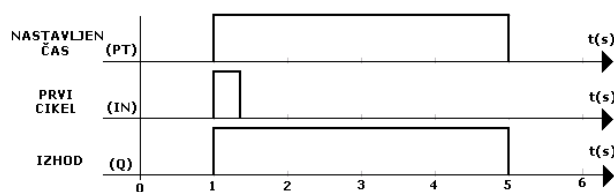
4.3.1 Splošni signali in nastavitve

Slika 4.4 prikazuje kako se signal *PRVICIKEL*, ki je aktiven eno časovno enoto po zagonu krmilnika, obdrži aktivnega za štiri sekunde. Logika se izvede na prvem krmilniku, ta pa pošlje signal še vsem ostalim trem krmilnikom.



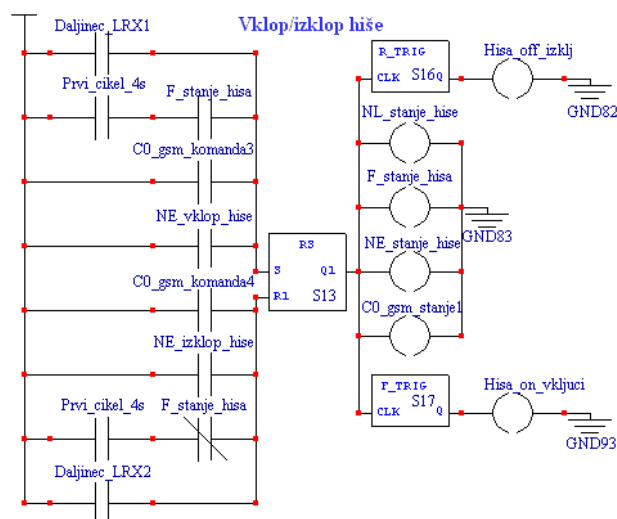
Slika 4.4 :Signal PRVICIKEL

Delovanje modula TP prikazuje slika 4.5. Ko je vhod $IN = 1$, začne modul odšteti čas, ki je določen na vhodu PT (ang. programmed time). Dokler se čas ne izteče, je izhod $Q = 1$ in s tem bit *Prvi_cikel_4s*. Ko se čas izteče, je izhod $Q = 0$. Izhod ET (ang. elapsed time) poroča, koliko je še do izteka nastavljenega časa PT.



Slika 4.5 :Delovanje časovnika TP

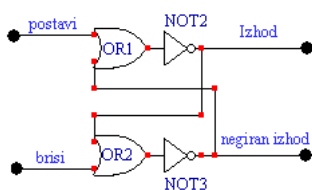
Del programa, ki vklopi ali izklopi hišo in s tem funkcije hiše prikazuje slika 4.6. Logika se izvede na prvem krmilniku, signal pa se pošlje še vsem ostalim trem krmilnikom. V tem delu programa se prvič pojavijo programski elementi: proženje na prednjo fronto *R_TRIG*, proženje na zadnjo fronto *F_TRIGGER* in pomnilna celica *RS*.



Slika 4.6:Stanje hiše

Elementa R_TRIG in F_TRIGGER pošljeta na izhod impulz v trenutku, ko gre signal na vhodu iz 1 → 0 za R_TRIG oziroma iz 1 → 0 za F_TRIGGER.

Pomnilna celica RS, katere vezje je prikazano na sliki 4.7 si zapomni stanje. To pomeni, da se signal na vhodu pojavi samo za kratek čas (prehod signala iz 0 → 1), na izhodu pa bo stanje mirovalo do naslednje spremembe signala na vhodu. Uporabljena pomnilna celica RS ima vhod R1 dominantnejši. Če pride hkrati aktiven signal na vhoda S in R1, potem prevlada vhod R1 in izhod pomnilne celice je neaktiven. Stanje izhodov ob določenih dogodkih je razvidno iz tabele 2.



Slika 4.7: Vezje pomnilne celice RS

Tabela 2: Delovanje pomnilne celice RS

| Postavi | Briši | Izhod $_{(n+1)}$ | Neg. Izhod $_{(n+1)}$ |
|---------|-------|------------------|-----------------------|
| 0 | 0 | Izhod $_{(n)}$ | Neg. Izhod $_{(n)}$ |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

n – trenutno stanje

n+1 – naslednje stanje

Vhod S v pomnilno celico RS (izhod Q1 postane aktiven – hiša je vklopljena):

- če je *Daljinec_LRX1* = 1; zahteva za vklop hiše iz daljinskega upravljalnika
- če je *Prvi_cikel_4s* = 1 in je *F_stanje_hisa* = 1; pred izpadom napajanja je bila hiša vklopljena
F_stanje_hisa je spremenljivka, ki je shranjena v remanentnem RAM-u. Informacija o stanju v remanentnem RAM-u ostane tudi po izpadu napajanja. Ko napajanje pride nazaj, je aktiven *PRVICIKEL* in s tem *Prvi_cikel_4s*. Če je spremenljivka *F_stanje_hisa* = 1 to pomeni, da je bil pred izpadom napajanja izhod pomnilne celice RS aktiven. S tem si krmilnik zapomni stanje hiše pred izpadom napajanja. Podobno je pri vseh ostalih delih programa, kjer shranjujem izhod v remanentni RAM.
- če je *C0_gsm_komanda3* = 1; zahteva za vklop hiše iz mobilnega telefona, ki jo krmilniku posreduje modul GSM
- če je *NE_vklop_hise* = 1; zahteva za vklop hiše iz zaslona, ki jo krmilniku posreduje modul Ethernet

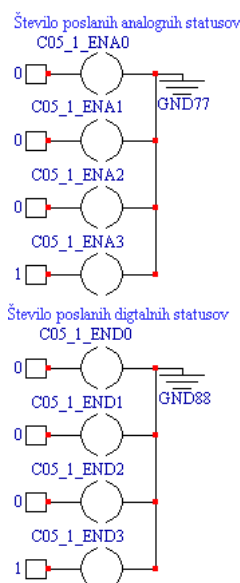
Vhod R v pomnilno celico RS (izhod $Q1$ postane neaktiven – hiša je izklopljena):

- če je $CO_gsm_komanda4 = 1$; zahteva za izklop hiše iz mobilnega telefona, ki jo krmilniku posreduje modul GSM
- če je $NE_izklop_hise = 1$; zahteva za izklop hiše iz zaslona, ki jo krmilniku posreduje modul Ethernet
- če je $Prvi_cikel_4s = 1$ in $F_stanje_hisa = 0$; pred izpadom napajanja je bila hiša izklopljena
- če je $Daljinec_LRX2 = 1$; zahteva za izklop hiše iz daljinskega upravljalnika

Izhod $Q1$ v pomnilni celici RS :

- NL_stanje_hise ; stanje, ki je poslano preko mreže LON vsem ostalim krmilnikom
- F_stanje_hisa ; stanje hiše, ki se shrani v remanentnem RAM
- NE_stanje_hise ; stanje, ki je preko Etherneta poslano zaslonu za prikaz stanja
- $CO_gsm_stanje1$; stanje, ki je se preko sporočila SMS pošlje na mobilni telefon

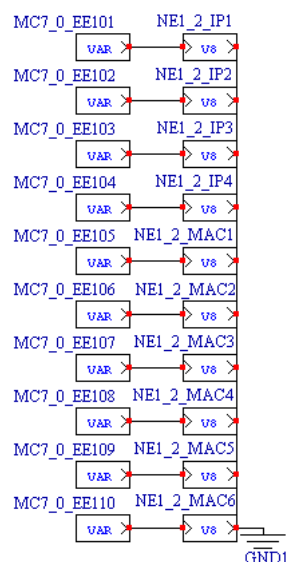
Nastavitve modula GSM se izvedejo z delom programa, ki je prikazan na sliki 4.8. Nastavitev pove število poslanih analognih in digitalnih statusov v sporočilu SMS. Številke desno (pred kvadratici) so zapisane v dvojiškem sistemu ($1000_{[2]} = 8_{[10]}$). V tem primeru bi ob poizvedovalnem sporočilu SMS prejeli kot odgovor osem analognih in osem digitalnih statusov.



Slika 4.8: Nastavitve modula GSM

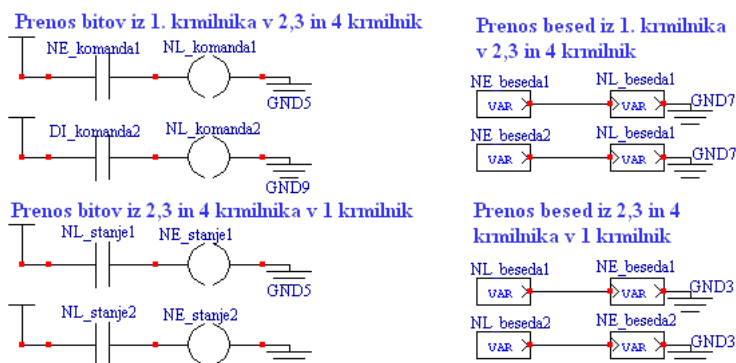
Slika 4.9 prikazuje nastavitve naslova IP (ang. internet protocol address) in naslova MAC (ang. media access control address) modula Ethernet NE1. Naslov nastavimo s pomočjo programa LPC Tester. Naslov shranimo v EEPROM, ki se nahaja v prvem krmilniku, iz tam pa ga prekopiramo v modul Ethernet NE1.

Nastavljanje IP in MAC naslova na modulu ethernet



Slika 4.9: Nastavitve naslova IP in naslova MAC za modul Ethernet.

Programsko rešitev za pošiljanje in sprejemanje bitov in besed preko omrežja LON v prvem krmilniku prikazuje slika 4.10. Pošilja se vsak bit posebej in vsaka beseda posebej. Spremenljivko, ki jo želimo poslati, prekopiramo v izhodno spremenljivko LON-a. Pri tem moramo vedeti kam izhodna spremenljivka LON-a potuje. Kam katera spremenljivka potuje definiramo z orodjem LONMaker. S tem orodjem definiramo telegrame za komunikacijo LON med krmilniki. S telegrami določimo kateri paket bitov se bo poslal kateremu krmilniku.



Slika 4.10: Pošiljanje in sprejemanje besed in bitov

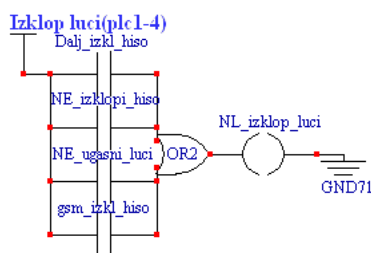
4.4 Luči in vtičnice

V hiši sta dve vrsti luči. Navadne, ki jih lahko samo prižgemo ali ugasnemo ter luči z možnostjo zatemnjevanja, ki poleg vklopa ali izklopa omogočajo še nastavljanje intenzivnosti osvetlitve. Z lučmi lahko operiramo s pomočjo tipk, daljinskim upravljalnikom in zaslonom občutljivim na dotik. Sistem ima vgrajeno tudi meritev zunanje osvetljenosti, na osnovi katere se prižiga razsvetljava parkirišča in hišne številke. Luči se lahko upravljajo tudi z ambient, ki so predhodno definirani.

Ko gre hiša v stanje izključeno, se ugasnejo vse luči v stanovanju. Luči na hodniku in stopnicah se ugasnejo z zakasnitvijo. Poleg tega ostanejo brez napajanja tudi porabniki priključeni na vtičnice, razen hladilnika, videorekorderja in akvarija.

4.4.1 Izklop vseh luči

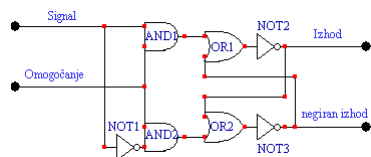
Pogoj za izklop vseh luči je prikazan na sliki 4.11 . Izklop vseh luči se uporablja takrat, ko zapuščamo hišo ali pa pred spanjem. Če je aktiven eden od pogojev: ukaz iz zaslona (*NE_izklopi_hiso*, *NE_ugasni_luci*), ukaz iz daljinskega upravljalnika (*Dalj_izkl_hiso*) ali ukaz iz mobilnega telefona preko sporočila SMS (*gsm_izkl_hiso*), pošlje prvi krmilnik ukaz za izklop luči (*NL_izklop_luci*) vsem krmilnikom.



Slika 4.11: Izklop vseh luči v hiši

4.4.2 Vodenje luči

Vsi ukazi za navadno luč so vezani na vhod pomnilne celice D. Pomnilno celico D prikazuje slika 4.12, njeno delovanje pa prikazuje tabela 3. Pomnilna celica D ob signalu na vhodu, spremeni stanje na izhodu, ki se ohrani do naslednje spremembe na vhodu. To pomeni, da se signal na vhodu pojavi samo za kratek čas (prehod signala iz 0 → 1), na izhodu pa bo ostalo stanje do naslednje spremembe signala na vhodu.



Slika 4.12: Shema pomnilne celice D

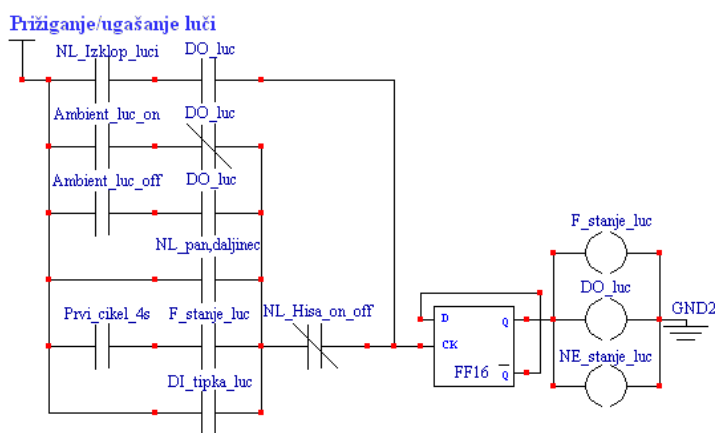
Tabela 3: Delovanje pomnilne celice D

| Izhod (n) | Signal | Izhod (n+1) |
|-----------|--------|-------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

n – trenutno stanje

n+1 – naslednje stanje

Del programa, ki upravlja z navadno lučjo je prikazan na sliki 4.13.



Slika 4.13: Vodenje navadnih luči

VHOD V pomnilno celico D:

Predpogoj, da se upošteva spodnje signale, je $NL_Hisa_on_off = 0$.

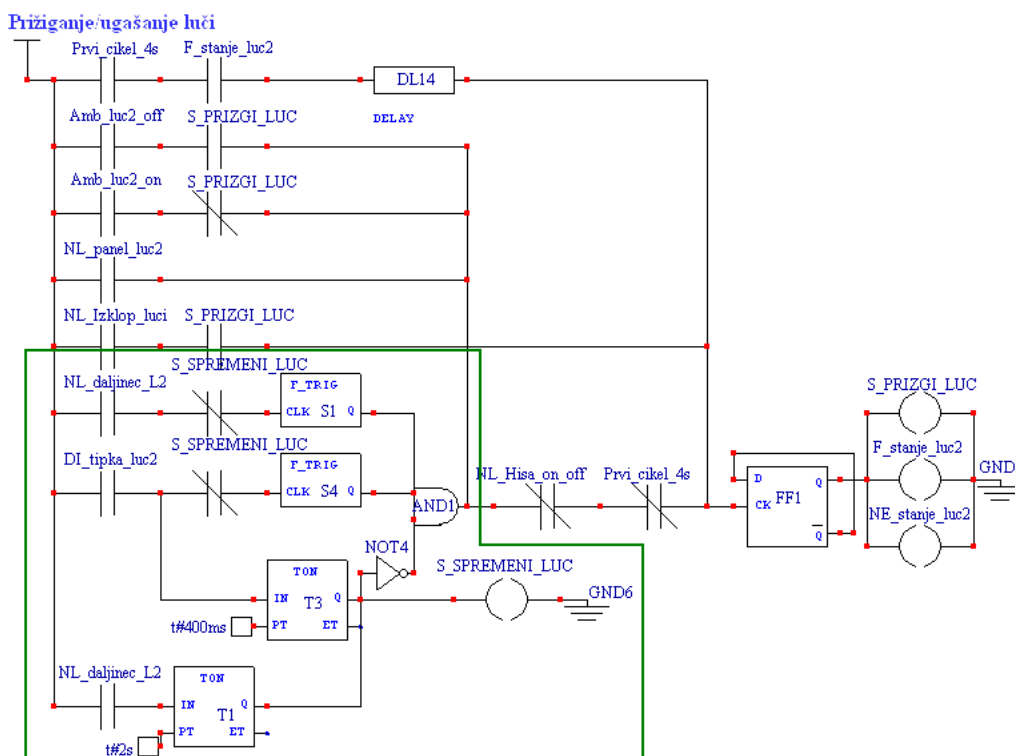
Če je signal $NL_Hisa_on_off = 1$, potem so onemogočeni vsi signali, razen NL_Izklop_luci , ki gre na vhod pomnilne celice D mimo pogoja $NL_Hisa_on_off$.

- če je signal $NL_Izklop_luci = 1$ in je izhod $DO_luc = 1$, potem se na CK (vhod pomnilne celice D) pošlje signal in izhod pomnilne celice Q = 0 (luč izklopljena); zahteva za izklop hiše ali izklop luči iz daljinskega upravljalnika ali zaslona
- če je signal $Ambient_luc_on = 1$ in je izhod $DO_luc = 0$, potem se na CK pošlje signal in izhod pomnilne celice Q = 1 (luč vklopljena); izbran je ambient, ki vklopi luč
- če je signal $Ambient_luc_off = 1$ in je izhod $DO_luc = 1$, potem se na CK pošlje signal in izhod pomnilne celice Q = 0 (luč izklopljena); izbran je ambient, ki izklopi luč
- če je signal $NL_pan,daljinec = 1$, potem se na CK pošlje signal in izhod pomnilne celice se obrne; zahteva iz daljinca ali panela, ki se prenaša preko mreže LON

če je $Prvi_cikel_4s = 1$ in je $F_stanje_luc = 1$, potem se na CK pošlje signal in izhod pomnilne celice $Q = 1$ (luč vklopljena); pred izpadom napajanja je bila luč prižgana
IZHOD pomnilne celice D:

Izhod pomnilne celice D je vezan na izhod luči (DO_luc – spremenljivka izhodnega modula), stanje luči, ki se pošlje zaslonu (NE_stanje_luc) in na spremenljivko (F_stanje_luc), ki je shranjena v remanentnem RAM-u.

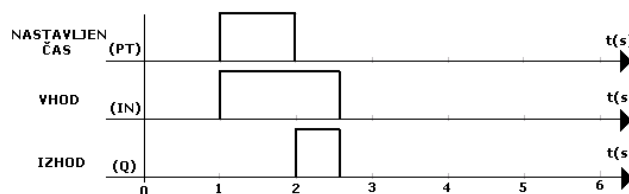
Vodenje luči z moduli za zatemnjevanje je zaradi kompleksnosti programa za prižiganje ali ugašanje in nastavljanje intenzivnosti osvetlitve razdeljen na štiri dele. Prvi del programa, ki je podoben kot pri navadni luči prikazuje slika 4.14. Razlikuje se le v spodnjem delu, ki je obrobjen (vklop ali izklop in omogočanje spremembe intenzivnosti osvetlitve s tipko ali daljinskim upravljalnikom).



Slika 4.14: Prvi del programa za luči z možnostjo zatemnjevanja

Ko je $DI_tipka_luc2 = 1$ (pritisnjena tipka) in $S_SPREMENI_LUC = 0$ (ni aktivno nastavljanje intenzivnosti osvetlitve) in hkrati je izhod Q časovnika T3 neaktiven, potem se pošlje signal na vhod pomnilne celice D (sprememba izhodnega stanja). Časovnik deluje tako kot prikazuje slika 4.15. Ko vhod časovnika IN postane aktiven se začne odšteti nastavljen čas PT. Potreben pogoj za aktiven izhod Q je iztečen nastavljen čas in še vedno aktiven vhod IN v časovnik. Ko gre vhod na nič, gre tudi izhod na nič. Če vhod pred iztekom nastavljenega časa pade na nič, potem se časovnik ponastavi, izhod časovnika pa je ves čas nič. Časovnik torej preverja ali smo tipko držali manj ali več kot 400 ms. Če smo držali manj kot 400 ms, sta izpolnjena oba pogoja za vrata AND1 in to pomeni samo spremembo stanja. Sprememba stanja se onemogoči, če je eden izmed bitov $NL_Hisa_on_off$ ali $Prvi_cikel_4s$ enak 1 (hiša

izključena ali ponoven zagon krmilnikov). Če pa smo držali tipko dlje kot 400 ms, potem se luč ne ugasne, ampak se postavi bit $S_SPREMENI_LUC$, ki omogoča nastavitvev intenzivnosti osvetlitve. Enaka logika velja za $NL_daljinec_L2$ (vklop, izklop in nastavitvev intenzivnosti osvetlitve iz daljinskega upravljalnika), le da je nastavljen čas na časovniku T1 2 s (zaradi počasnejšega odziva daljinskega upravljalnika).



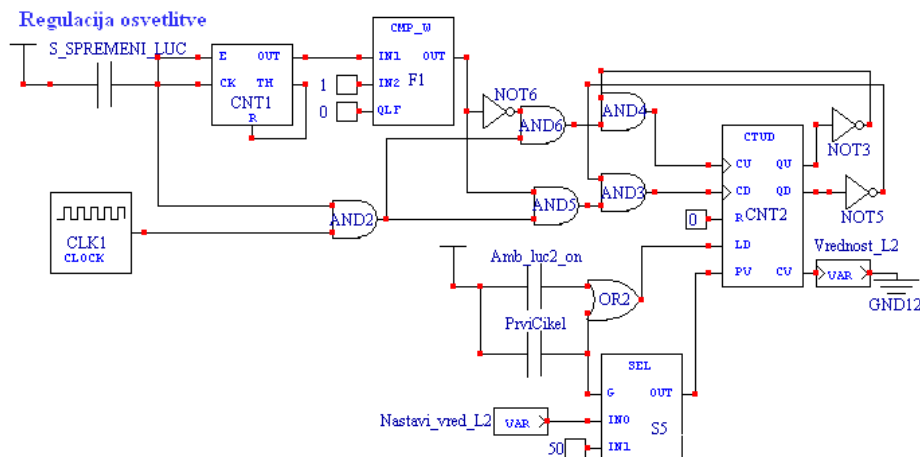
Slika 4.15: Delovanje časovnika TON

Drugi del programa, ki ga prikazuje slika 4.16 je namenjen nastavljanju intenzivnosti osvetlitve. $S_SPREMENI_LUC$ je bit, ki ga postavlja prvi del programa. Ko je $S_SPREMENI_LUC = 1$, se aktivira števec CNT1 (za aktivacijo morata biti vhoda CK = 1 (ang. clock) in E = 1 (ang. enable)). Časovnik ob impulzih šteje od 0 do 2. Trenutna cifra časovnika gre na izhod OUT. Ko je vrednost enaka 2, postane izhod TH = 1 (ang. threshold), ta postavi vhod R = 1 (ang. reset) in izhod časovnika OUT postane 0. Cifro časovnik pošlje na prvi vhod IN1 primerjalnika F1. Vrsto operacije primerjalnika določimo z vhomom QLF (0 pomeni enakost). Primerjalnik torej preveri ali je prvi vhod enak drugemu vhodu IN2 (ta je vedno enak 1). Če sta vhoda enaka, potem je izhod primerjalnika OUT = 1. CNT1 omogoča, da se enkrat intenzivnost osvetlitve dviga, drugič pa se intenzivnost osvetlitve spušča. Če je izhod primerjalnika OUT = 1, potem je izpolnjen prvi pogoj za vrata AND5 (spuščanje). Če je izhod primerjalnika OUT = 0, potem je izpolnjen prvi pogoj za vrata AND6 (dviganje). Če je $S_SPREMENI_LUC = 1$ in je pozitivna urina fronta ure CLK1, potem je izpolnjen pogoj za vrata AND2 in s tem drugi pogoj za vrata AND5 ali AND6. Za tema dvema pogojema preverim, ali je trenutna vrednost števca, maksimalna oziroma minimalna, mogoča. Če je števec na zgornji meji, je aktiven izhod števca (CNT2) QU, če pa je števec na spodnji meji, je aktiven izhod števca QD. Izhoda sta negirana in povezana na vrata AND4 in AND3. To je potrebno zato, da se štetje omeji. Navzgor se šteje le, če vrednost števca ni maksimalno mogoča in obratno za minimalno vrednost. Ko je signal na vhodu CU (ang. counter up), šteje števec navzgor, ko pa je signal na vhodu CD (ang. counter down), šteje števec navzdol.

Vhod R (ang. reset) v števec je vedno neaktiven, vhoda LD (ang. load) in PV (ang. programmed value) pa se uporabljata v paru. Ko je vhod LD aktiven, se v števec prenese vrednost iz vhoda PV. Ko je aktiven *PrviCikel*, je vhod LD = 1, na selektorju S5 pa je izbran drugi vhod IN1, ki ima nastavljeno vrednost na 50. Ta vrednost se prenese na izhod selektorja in s tem na vhod PV v števec. S tem je omejeno štetje od 0 - 50. Blok SEL izbira med dvema vhomoma (IN0 in IN1) glede na vhod G. Če je vhod G = 0, potem je izbran vhod IN0, če je vhod G = 1, potem je izbran vhod IN1.

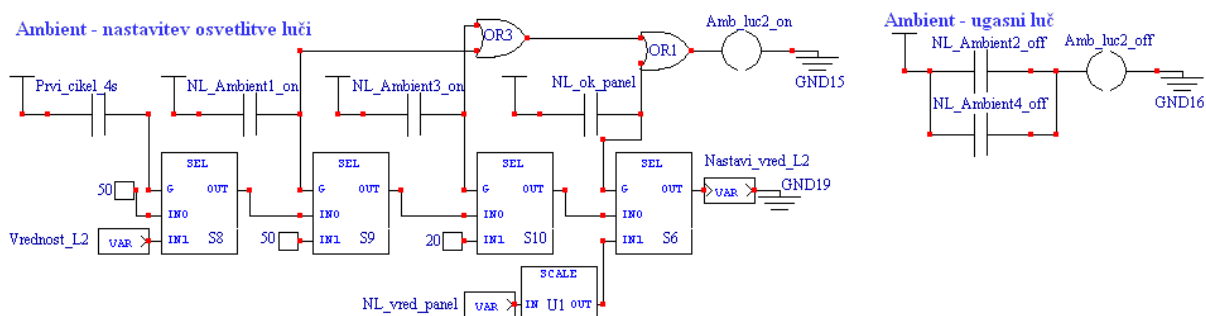
Cifra, ki pride na izhod CV (ang. counting value) in se prepíše v besedo $Vrednost_L2$, se v četrtem delu programa množi z 2. Tako dobimo razpon štetja 0 - 100, ki predstavlja procent

osvetlitve luči. Množenje z 2 se uporablja zaradi hitrejšega spreminjanja osvetlitve. Če je aktiven *Amb_luc2_on*, je vhod LD = 1, na selektorju S5 pa je izbran prvi vhod IN0 (beseda *Nastavi_vred_L2*). Beseda se prenese na izhod selektorja OUT in s tem na vhod števec PV. Besedo *Nastavi_vred_L2* dobimo iz tretjega dela programa in pomeni vrednost, ki jo nastavimo ob izbranem ambientu ali pa ob nastavljanju intenzivnosti osvetlitve iz zaslona.



Slika 4.16: Drugi del programa za luči z možnostjo zatemnjenja

Tretji del programa, ki ga prikazuje slika 4.17 je namenjen nastavljanju intenzivnosti osvetlitve ob vklopu ambientov, ponovnem zagonu krmilnikov in nastavljanju intenzivnosti osvetlitve iz zaslona. Luč z možnostjo zatemnjenja ob izberi ambienta 1 ali 3 sveti z določeno osvetlitvijo, ob izbiri ambienta 2 ali 4, pa je luč ugasnjena. Drugi in tretji selektor se uporabljata za nastavev intenzivnosti osvetlitve ob izberi ambienta 1 ali 3. Prvi selektor je namenjen kopiranju besede *Vrednost_L2*, ki je dejanska vrednost luči z možnostjo zatemnjenja, v besedo *Nastavi_vred_L2* (ob ponovnem zagonu krmilnika). Četrti selektor je namenjen kopiranju besede *NL_vred_panel* (nastavev iz zaslona) v *Nastavi_vred_L2*.



Slika 4.17: Tretji del programa za luči z možnostjo zatemnjenja

Prvi selektor S8 ima na izbirnem vhodu G *Prvi_cikel_4s*. Ko je ta signal aktiven, je izbran drugi vhod IN1 in vrednost tega vhoda (beseda *Vrednost_L2*, ki se shranjuje v remanentni RAM) se prepíše na izhod OUT selektorja. Če je *Prvi_cikel_4s* neaktiven, se na izhod prepisuje vrednost prvega vhoda IN0 (50).

Drugi selektor S9 ima na izbirnem vhodu G *NL_Ambient1_on*. Ko je ta signal aktiven, je izbran drugi vhod IN1 in vrednost tega vhoda (50) se prepiše na izhod OUT selektorja. Če je *NL_Ambient1_on* neaktiven se na izhod prepisuje vrednost prvega vhoda IN0 (vrednost, ki pride iz prvega selektorja).

Tretji selektor S10 ima na izbirnem vhodu G *NL_Ambient3_on*. Ko je ta signal aktiven, je izbran drugi vhod IN1 in vrednost tega vhoda (20) se prepiše na izhod OUT selektorja. Če je *NL_Ambient3_on* neaktiven, se na izhod prepisuje vrednost prvega vhoda IN0 (vrednost, ki pride iz drugega selektorja).

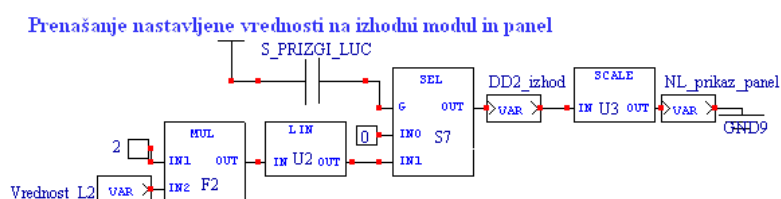
Četrti selektor S6 ima na izbirnem vhodu G *NL_ok_panel*. Ko je ta signal aktiven, je izbran drugi vhod IN1 in vrednost tega vhoda (*NL_vrednost_panel*, ki je skalirana vrednost poslana iz zaslona) se prepiše na izhod OUT selektorja. Če je *NL_ok_panel* neaktiven, se na izhod prepisuje vrednost prvega vhoda IN0 (vrednost, ki pride iz tretjega selektorja).

Če je izbirni vhod drugega, tretjega ali četrtega selektorja aktiven se postavi bit *Amb_luc2_on*. Ta bit in vrednost besede *Nastavi_vred_L2* se uporablja v drugem delu programa (opisano zgoraj).

Ker je izhod prvega selektorja vezan na vhod drugega selektorja itd., se vrednost, ki jo želimo vpisati v števec prenaša od prvega selektorja do zadnjega. Zadnji selektor na izhodu zapiše v besedo *Nastavi_vred_L2*, ki se v drugem delu programa vpiše v števec, ki vrednost prepiše v besedo *Vrednost_L2*, ta pa se v četrtem delu programa prepiše na izhod luči z možnostjo zatemnjenja (*DD2_izhod*).

Ob izberi drugega ali četrtega ambienta se postavi bit *Amb_luc2_off*, ki ugasne luč z možnostjo zatemnjenja. Logika za ugašanje je narejena v prvem delu programa.

Četrti del programa, ki ga prikazuje slika 4.18 je del, ki vklopi ali izklopi luč z možnostjo zatemnjenja in prekopira želeno vrednost na izhod modula za zatemnjenje. Luč se prižiga in ugaša z bitom *S_PRIZGI_LUC*, ki ga postavlja prvi del programa.



Slika 4.18: Četrti del programa za luči z možnostjo zatemnjenja

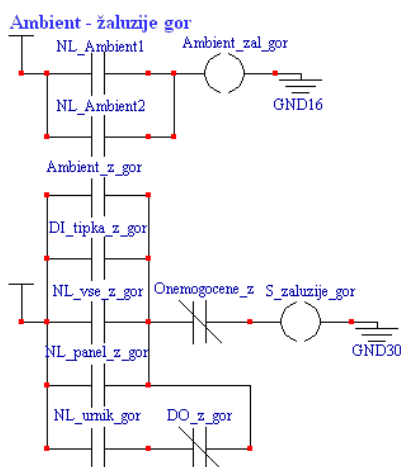
Beseda *Vrednost_L2* pride iz drugega dela programa in se množi z 2 zaradi hitrejšega zatemnjenja. Za pospešitev zatemnjenja se uporablja se tudi linearno skaliranje. Linearno skaliranje je uporabljeno zato, da na začetku ob rastočem vhodu izhod raste hitreje kot na koncu. To pomeni, da luč potrebuje iz osvetlitve 5 % (najmanj) do 70 % približno 3 sekunde.

Od 70 % do 100 % pa skoči praktično v trenutku. Razlika med 70 % in 100 % osvetlitve je zelo majhna, skoraj neopazna.

Selektor S7 se uporablja za prižiganje in ugašanje luči. Če je signal $S_PRIZGI_LUC = 1$, je izbran drugi vhod IN1 in s tem zelena vrednost luči z možnostjo zatemnjevanja. Če je $S_PRIZGI_LUC = 0$, potem je izbran prvi vhod IN0, ki ima vrednost 0. Če je izhod selektorja $OUT = 0$, potem se v $DD2_izhod$ vpiše 0 in luč z možnostjo zatemnjevanja ne sveti. Sicer sveti z določeno osvetlitvijo. Skaliranje $DD2_izhoda$ se uporablja za prikaz na zaslonu (NL_prikaz_panel). Na zaslonu se prikazuje v procentih (0 - 100 %), $DD2_izhod$ pa ima območje 0 - 255.

4.4.3 Vodenje žaluzij

Žaluzije lahko vodimo s tipkami, daljinskim upravljalnikom ter z ročnimi in avtomatskimi ukazi iz zaslona. Vodenje žaluzij po urniku lahko omogočimo ali onemogočimo. Imamo tudi možnost izbire med zimskim in letnim režimom. V zimskem režimu velja avtomatski ukaz določen po urniku za vse žaluzije v hiši (razen stopnišča), v letnem režimu pa samo za žaluzije, ki ščitijo pred segrevanjem prostora s sončnimi žarki. Vodenje žaluzij se izvaja na krmilnikih, urniki pa se izvajajo na zaslonu. Del programa za vodenje žaluzij je zaradi kompleksnosti razdeljen na tri dele. Prvi del, ki zbere vse vhodne bite za dviganje žaluzij v en sam bit, ki se nato pošlje v drugi del programa, kjer sproži dviganje žaluzij, prikazuje slika 4.19.



Slika 4.19: Prvi del programa za dviganje žaluzij

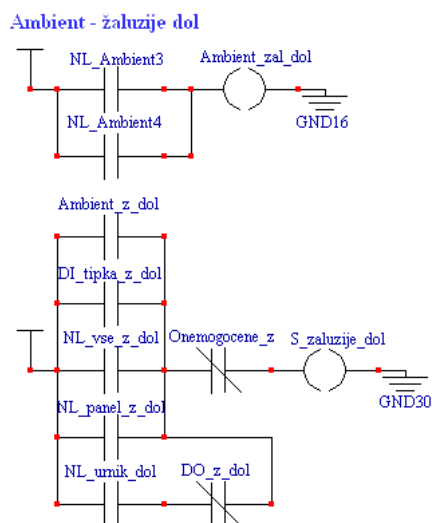
Če je $NL_Ambient1 = 1$ ali $NL_Ambient2 = 1$ se postavi bit $Ambient_zal_gor$ (eden izmed vhodov za dviganje žaluzij).

Bit $S_zaluzije_gor = 1$, če je izpolnjen eden izmed pogojev:

- Če je $Ambient_z_gor = 1$ in $Onemogocene_z = 0$; ambient 1 in 2 dvigneta žaluzijo
- Če je $DI_tipka_z_gor = 1$ in $Onemogocene_z = 0$; ukaz iz tipke

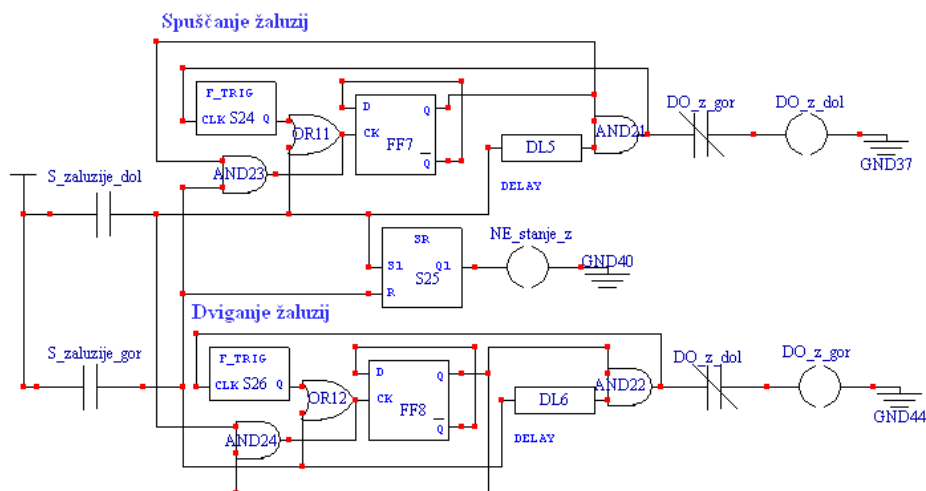
- Če je $NL_vse_z_gor = 1$ in $Onemogocene_z = 0$; ukaz iz zaslona ali daljinca, vse žaluzije gor
- Če je $NL_panel_z_gor = 1$ in $Onemogocene_z = 0$; ukaz iz zaslona, za določeno žaluzijo
- Če je $NL_urnik_gor = 1$, $DO_z_gor = 0$ in $Onemogocene_z = 0$; ukaz iz zaslona, žaluzija po urniku gor

Drugi del programa zbere vse vhodne bite za spuščanje žaluzij v en sam bit, ki se nato pošlje v drugi del programa kjer sproži spuščanje žaluzij, prikazuje slika 4.20.



Slika 4.20: Drugi del programa za spuščanje žaluzij

Tretji del programa je logika, ki drži izhod žaluzije postavljen toliko časa, kot ga žaluzija potrebuje za premik od najnižje točke do najvišje točke. Ta del programa je prikazan na sliki 4.21. $S_zaluzije_gor$ in $S_zaluzije_dol$ sta bita, ki se postavljata v prvem in drugem delu programa delu programa.



Slika 4.21: Tretji del programa za dviganje ali spuščanje žaluzij

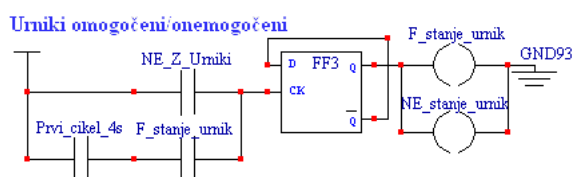
Če je signal $S_{zaluzije_gor} = 1$, se pomnilna celica RS (S25) resetira in posledično je $NE_stanje_z = 0$ (stanje, ki se pošlje zaslonu za prikaz zadnjega gibanja žaluzij). Če je spremenljivka $NE_stanje_z = 0$, potem smo nazadnje žaluzijo dvigali, sicer smo jo spuščali.

Signal $S_{zaluzije_gor}$ postavi izhod pomnilne celice (FF8) $Q = 1$, če ta še ni bil aktiven. Če je bil ta že aktiven, potem postane izhod $Q = 0$. Tako lahko vklopimo ali izklopimo dviganje žaluzije za določen čas. Če se je žaluzija spuščala ($DO_z_dol = 1$) pred signalom $S_{zaluzije_gor}$, potem sta izpolnjena oba pogoja za vrata AND23 in zato pošljemo signal na vhod v pomnilno celico D (FF7). Tako prekinemo spuščanje žaluzij in začnemo z dviganjem žaluzij. Ko je $S_{zaluzije_gor} = 1$, se aktivira tudi števec DL6, ki drži aktiven signal 70 sekund - toliko je potrebno za premik žaluzije od najnižje točke do najvišje točke. S tem se izpolnita oba pogoja za vrata AND22. Če izhod za spuščanje žaluzije DO_z_dol ni aktiven (zaščita motorja zaradi počasnega preklopa modula), potem se aktivira izhod za dviganje žaluzije $DO_z_gor = 1$. Ko izhod vrat AND22 pade iz $0 \rightarrow 1$, se aktivira izhod F_TRIG (S26). Signal je poslan na vhod pomnilne celice D (FF8) in s tem postane izhod $Q = 0$.

Enaka logika velja za spuščanje žaluzij.

4.4.4 Nastavitve urnikov za žaluzije

Urniki omogočajo avtomatsko spuščanje žaluzij v letnem ali zimskem režimu. Urnike lahko tudi omogočimo ali onemogočimo z delom programa, ki je prikazan na sliki 4.22. Pri izključenih hiši je premikanje žaluzij po urniku onemogočeno. Vse nastavitve urnikov se izvajajo na prvem krmilniku.



Slika 4.22: Omogočanje in onemogočanje urnikov

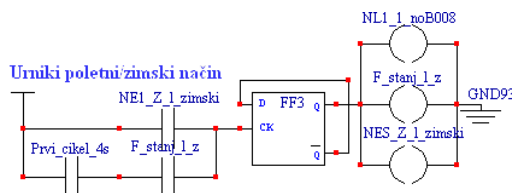
VHOD v pomnilno celico D:

- če je signal $NE_Z_Urniki = 1$ se stanje pomnilne celice na izhodu obrne
- če je signal $Prvi_cikel_4s = 1$ in $F_stanje_urnik = 1$ je izhod pomnilne celice $Q = 1$

IZHOD pomnilne celice D:

Izhod pomnilne celice D je vezan na NE_stanje_urnik in F_stanje_urnik . Prvi bit je poslan zaslonu za prikaz stanja, drugi bit pa je shranjen v remanentni RAM. Bit F_stanje_urnik se uporablja tudi pri omogočanju pošiljanja signala za avtomatsko spuščanje oziroma dviganje žaluzij (kot prikazuje slika 4.24).

Slika 4.23 prikazuje del programa, kjer se izbira med letnim in zimskim režimom delovanja žaluzij.



Slika 4.23: Letni/zimski režim urnikov

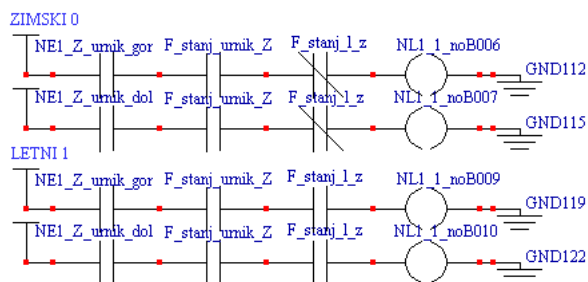
VHOD v pomnilno celico D:

- če je signal $NE_Z_l_zimski = 1$ se stanje pomnilne celice na izhodu obrne
- če je signal $Prvi_cikel_4s = 1$ in $F_stanje_l_z = 1$ je izhod pomnilne celice $Q = 1$

IZHOD pomnilne celice D:

Izhod pomnilne celice D je vezan na $NES_Z_l_zimski$, $F_stanje_l_z$ in $NL1_1_noB008$. Prvi bit je poslan zaslonu za prikaz stanja, drugi bit je shranjen v remanentni RAM, tretji pa se pošlje drugemu, tretjemu in četrtemu krmilniku. Bit F_stanje_urnik se uporablja tudi pri omogočanju pošiljanja signala za letni oziroma zimski režim (kot prikazuje slika 4.24).

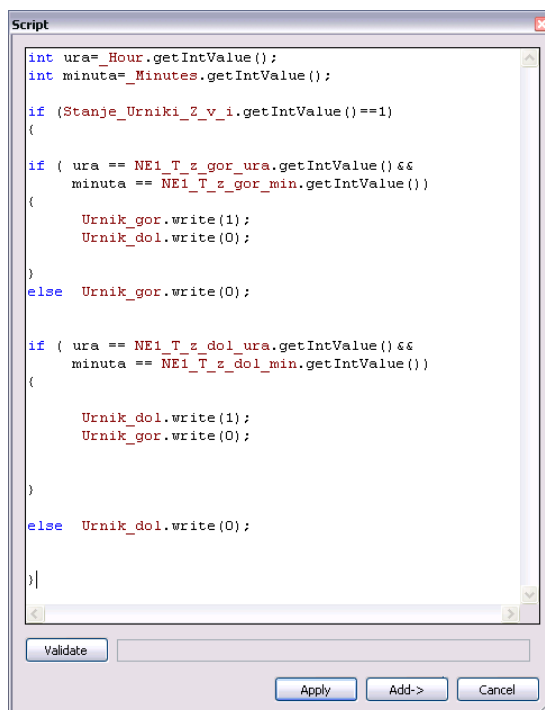
Slika 4.24 prikazuje pošiljanje bitov za dviganje ali spuščanje žaluzij po urnikih. Izhodni biti se pošiljajo vsem trem krmilnikom in gredo na vhod logike za vodenje žaluzij. Signal pošljemo žaluzijam na katere želimo, da bi imel urnik vpliv.



Slika 4.24: Pošiljanje bitov za spuščanje in dviganje žaluzij krmilnikom

- Če je bit $F_stanje_urnik_Z = 1$ (urniki omogočeni) in bit $F_stanj_l_z = 0$ (zimski režim), potem sta lahko aktivna bita $NL1_1_noB006$ in $NL1_1_noB007$ (v zimskem režimu urnik vpliva na vse žaluzije razen na žaluzije na stopnišču). Bit $NL1_1_noB006$ je aktiven, če je bit $NE1_Z_urnik_gor = 1$, bit $NL1_1_noB007$ pa je aktiven, če je bit $NE1_Z_urnik_dol = 1$.
- Če je bit $F_stanje_urnik_Z = 1$ (urniki omogočeni) in bit $F_stanj_l_z = 1$ (poletni režim), potem sta lahko aktivna bita $NL1_1_noB009$ in $NL1_1_noB010$ (v poletnem režimu urnik vpliva na žaluzije, ki jih obsije popoldansko sonce). Bit $NL1_1_noB009$ je aktiven, če je bit $NE1_Z_urnik_gor = 1$, bit $NL1_1_noB010$ pa je aktiven, če je bit $NE1_Z_urnik_dol = 1$.

Del programa, ki preverja, ali je dejanski čas (čas na zaslonu) enak nastavljenemu času za spuščanje ali dviganje žaluzij, je realiziran na zaslonu in je prikazan na sliki 4.25. Če so urniki omogočeni, se program izvaja na vsakem celozaslonskem oknu in sicer periodično vsakih 15 sekund.



```

Script
int ura=_Hour.getIntValue();
int minuta=_Minutes.getIntValue();

if (Stanje_Urniki_Z_v_i.getIntValue()==1)
{
if ( ura == NE1_T_z_gor_ura.getIntValue() &&
minuta == NE1_T_z_gor_min.getIntValue() )
{
    Urnik_gor.write(1);
    Urnik_dol.write(0);
}
else Urnik_gor.write(0);

if ( ura == NE1_T_z_dol_ura.getIntValue() &&
minuta == NE1_T_z_dol_min.getIntValue() )
{
    Urnik_dol.write(1);
    Urnik_gor.write(0);
}
else Urnik_dol.write(0);
}
}
Validate
Apply Add-> Cancel

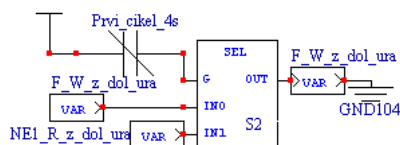
```

Slika 4.25: Program za urnike žaluzij

Spremenljivki $_Hour$ in $_Minutes$ sta sistemski in imata informacijo o trenutni uri in trenutnih minutah.

Najprej se preveri, če so urniki omogočeni. Če so omogočeni, se preveri, če je nastavljena ura ($NE1_T_z_gor_ura$) in če je nastavljena minuta ($NE1_T_z_gor_min$) enaka trenutnemu času. Če sta časa enaka, se postavi spremenljivko $Urnik_gor = 1$. Spremenljivka $Urnik_gor$ je izhodna spremenljivka in se pošlje na prvi krmilnik. Prvi krmilnik pošlje ukaz še vsem

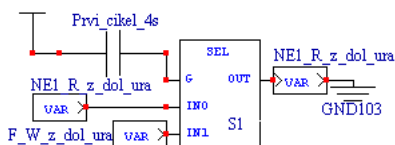
ostalim krmilnikom. Enako velja za spodnji del programa, ki je namenjen spuščanju žaluzij. Slika 4.26 prikazuje, kako se informacija o nastavljenih časih za spuščanje in dviganje žaluzij po urniku shrani v remanentni RAM. Če *Prvi_cikel_4s* ni aktiven, torej krmilnik deluje dlje kot 4 sekunde, potem je izbran vhod IN1. V tem primeru se *NE1_R_z_dol_ura* (beseda, ki jo krmilnik prejme iz zaslona) kopira v *F_W_z_dol_ura* (remanentni RAM v krmilniku). V primeru da je aktiven *Prvi_cikel_4s* (pomeni, da krmilnik deluje manj kot 4 s), se *F_W_z_dol_ura* kopira sama vase. Bit *Prvi_cikel_4s* je negiran.



Slika 4.26: Shranjevanje vrednosti v remanentni RAM

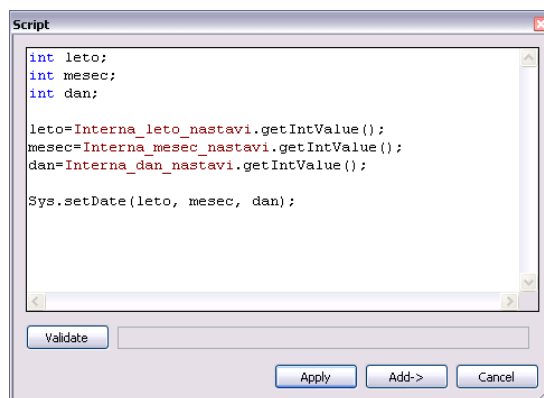
Slika 4.27 prikazuje, kako se informacija o nastavljenih časih za spuščanje in dviganje žaluzij po urniku bere iz remanentnega RAM-a (po ponovnem zagonu krmilnikov) za prikaz na zaslonu. Če je *Prvi_cikel_4s* = 1 (krmilnik deluje manj kot 4 s), potem je izbran vhod IN1. V tem primeru se *F_W_z_dol_ura* (beseda, ki je shranjena v remanentnem RAM-u) kopira v *NE1_R_z_dol_ura* (beseda, ki jo krmilnik prejme iz zaslona). V primeru da je bit *Prvi_cikel_4s* = 0 (pomeni, da krmilnik deluje več kot 4 s), se *NE1_R_z_dol_ura* kopira sama vase.

Na ta način si prvi krmilnik zapomni vse nastavljene in prikazane parametre na zaslonu.



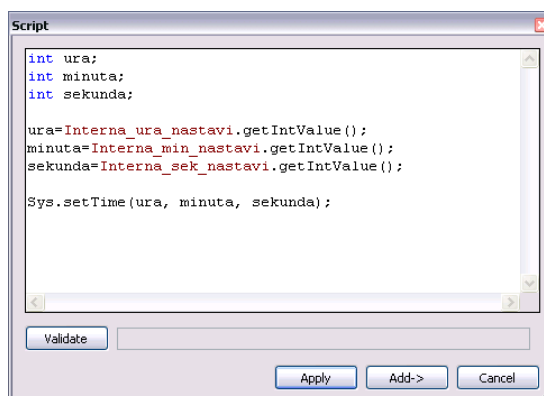
Slika 4.27: Kopiranje vrednosti iz remanentnega RAM-a ob ponovnem zagonu

Slika 4.28 prikazuje del programa, ki se izvede s klikom na gumb za nastavitev datuma. Predhodno vnesemo želeni datum. Želeni datum se shrani v spremenljivkah *Interna_letu_nastavi*, *Interna_mesec_nastavi* in *Interna_dan_nastavi*. Te spremenljivke se kopirajo v lokalne spremenljivke *letu*, *mesec* in *dan*, nato pa se s sistemskim ukazom `Sys.setDate` nastavi želeni datum.



Slika 4.28: Nastavljanje datuma na zaslonu

Slika 4.29 prikazuje del programa, ki se izvede s klikom na gumb za nastavitev ure. Logika je enaka kot pri nastavljanju želenega datuma.



Slika 4.29: Nastavljanje ure na zaslonu

4.5 Upravljanje z mobilnim telefonom

Modul GSM Komander GM05 je namenjen upravljanju naprav z razdalje preko mobilnega telefona. Ukazi iz mobilnega telefona so enakovredni ukazom tipk, zaslona ali daljinskega upravljalnika. Vsi ukazi so samo kratki impulzi. GSM Komander GM05 ob ukazu pulzno postavi določeno spremenljivko (spremenljivka je kratek čas postavljena na 1). Poleg pulznega postavljanja spremenljivk modul omogoča, da ob določenem ukazu spremenljivko obdrži na 1, dokler ne pride nov ukaz, ki spremenljivko postavi na 0.

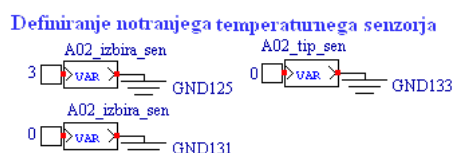
4.6 Upravljanje z daljinskim upravljalnikom

V dnevni sobi je infrardeči sprejemnik, s katerim se s pomočjo univerzalnega infrardečega daljinskega upravljalnika upravlja z razsvetljavo luči, izbira ambiente, upravlja z žaluzijami ter vklopi ali izklopi pametno hišo. Univerzalnemu daljinskemu upravljalniku je potrebno nastaviti kodo 1238, da daljinski upravljalnik lahko komunicira z infrardečim sprejemnikom. Ob pritisku na določeno tipko daljinskega upravljalnika se infrardečemu sprejemniku pošlje določeno kodo. Katera koda postavi kateri bit lahko določimo sami in to nastavitev shranimo v prvi krmilnik.

primerjata ali je vhod (IN1) enak ali manjši kot vhod (IN2). Primerjavo enak ali manjši določa bit $QLF = 3$. Prvi primerjalni blok preprečuje zmrzovanje. Če je temperatura enaka ali manjša od $5\text{ }^{\circ}\text{C}$, potem se postavi bit *zmrzovanje* (ki vklopi gretje v vseh prostorih) in bit *G05_gsm_stanje*. Ko je bit *G05_gsm_stanje* = 1, se alarm samodejno pošlje na mobilni telefon.

Drugi primerjalni blok obdrži temperaturo v določenih mejah v ekonomičnem režimu. Če je temperatura enaka ali manjša od $15\text{ }^{\circ}\text{C}$ in je *NE_ekon_on_off* = 1 (vklopljen ekonomični režim), potem se postavi bit *Prizgi_ekonom*, ki vklopi gretje v vseh prostorih.

Temperaturni senzor je pripeljan na modul AO2 in je definiran kot kaže slika 4.32. Na modul AO2 lahko vežemo več različnih temperaturnih senzorjev, ki imajo različne karakteristike, zato je potrebno določiti, kateri senzor je vezan na vhod modula AO2.



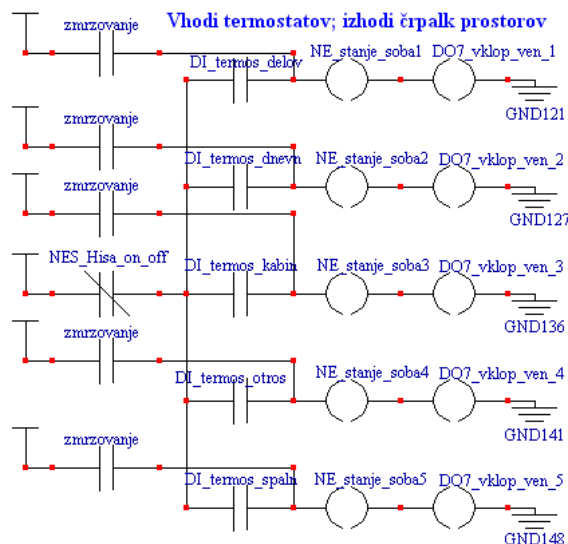
Slika 4.32:Definicija notranjega temperaturnega senzorja

Ker na vhodu $IN1 = 215$ pomeni $21,5\text{ }^{\circ}\text{C}$, je potrebno besedo *P01_zun_temp* deliti z 10, da dobimo pravo vrednost. Vrednost se potem prepiše v besedo *NE_panel_z_temp*, ki je poslana zaslonu. Del programa, ki vhodno temperaturo deli in jo pošlje na zaslon je prikazan na sliki 4.33.



Slika 4.33:Zunanji temperaturni Senzor

Za vklop posameznih elektromagnetnih ventilov za ogrevanje posameznih ogrevalnih vej skrbi program prikazan na sliki 4.34.

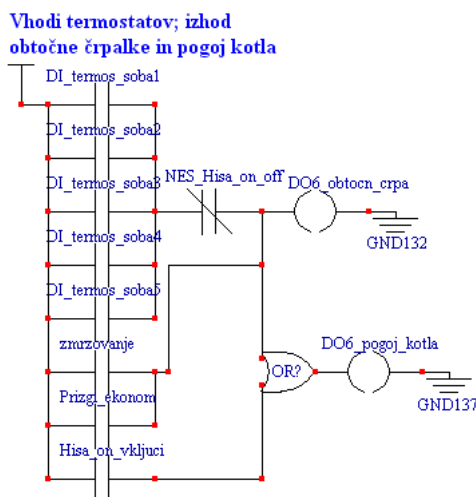


Slika 4.34: Vključitev črpalke po prostorih

- če je signal $NES_Hisa_on_off = 1$, potem so onemogočeni vsi biti, razen *zmrzovanje*, ki gre mimo pogoja $NES_Hisa_on_off$. Ko je bit $zmrzovanje = 1$ se vklopi vse ogrevalne veje.
- če je $NES_Hisa_on_off = 0$ in $DI_termos_soba1 = 1$, potem se postavi bita NE_stanje_soba1 (stanje poslano na zaslon) in $DO7_vklop_ven_1$ (izhodni modul, ki odpre ventil za ogrevanje sobe 1)

Enako se obnašajo tudi vse ostale štiri veje.

Za vklop obtočne črpalke in kotla za sanitarno vodo skrbi del programa na sliki 4.35.



Slika 4.35: Prižiganje obtočne črpalke in pogoja kotla

Če $DI_termos_soba1 - DI_termos_soba5$ (aktivna je vsaj ena zahteva termostata) in je $NES_Hisa_on_off = 0$ (hiša je vklopljena), potem se postavi bita $DO6_obtocn_crpa$ (vklopi se črpalka) in $DO6_pogoj_kotla$ (omogočanje ogrevanja sanitarne vode).

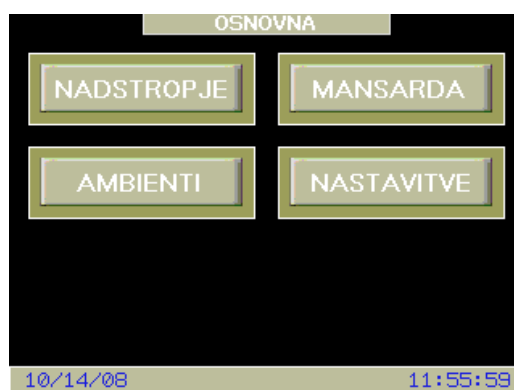
V primeru, da je postavljen bit *zmrzovanje* = 1 ali bit *Prizgi_ekonom* = 1, se bita *DO6_obtocn_crpa* in *DO6_pogoj_kotla* vklopita tudi takrat, ko je *NES_Hisa_on_off* = 1 (hiša izklopljena). Ko je bit *Hisa_on_vkljuci* = 1, se postavi bit *DO6_pogoj_kotla*, ki omogoči toplo sanitarno vodo.

4.8 Uporabniški vmesnik - zaslon občutljiv na dotik

Sistem se lahko v celoti upravlja in spremlja preko zaslona. Na zaslonu se prikazujejo vsa obvestila in delovanje naprav. Nastavljajo se tudi urniki, nastavitve za ogrevanje ter nastavitve za razsvetljavo - ambiente. Zaslon je občutljiv na dotik. S pritiskom prožimo dogodke. S pritiskom na gumb (ki je vezan na bit), se krmilniku pošlje kot kratek impulz. Vse gumbе, ki pomenijo neko zahtevo od krmilnika, je potrebno držati približno sekundo, da se zahteva zagotovo izvede.

4.8.1 Osnovna stran

Osnovna stran, prikazana na sliki 4.36 se pojavi vsakič, ko se zaslon ponovno zažene (v primeru izpada elektrike). Na osnovni strani sta prikazana trenutna ura in datum. S pritiskom na gumbe prehajamo na ostale strani (nadstropje, mansarda, ambiente, nastavitve).



Slika 4.36: Panel osnovna stran

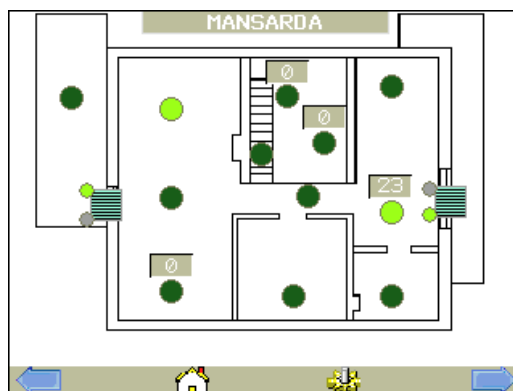
4.8.2 Nadstropje in mansarda

Strani na slikah 4.37 in 4.38 prikazujeta stanje luči in žaluzij. Stanje luči je predstavljeno s krogcem, ki se obarva glede na stanje spremenljivke *NE_stanje_luc*. Če je krogec obarvan temno zeleno, potem je luč ugasnjena. Če je krogec obarvan svetlo zeleno je luč prižgana. Če gre za luč, vezano na modul za zatemnjevanje, potem je zraven prikazan še nivo osvetlitve luči v procentih (*NE_panel_prikaz*).

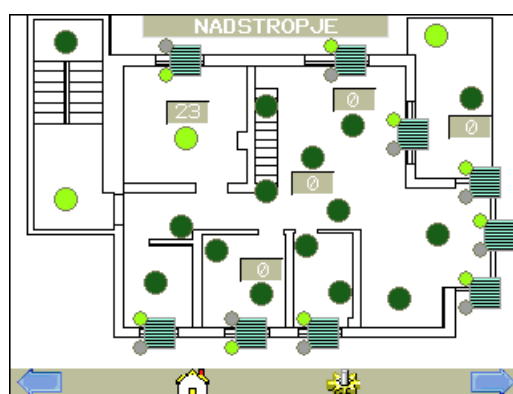
Pri žaluzijah je prikazano, kam se je zadnjič premikala žaluzija. Če sveti zgornji krogec (je svetlo zelen) in je spodnji krogec siv, potem smo nazadnje žaluzijo dvigali. Obratno velja, če smo nazadnje žaluzijo spuščali. Za prikaz se uporablja samo en bit *NE_stanje_z*.

Na spodnjem sivem pasu so gumbi za navigacijo. S pritiskom na puščico, obrnjeno v levo, se vrnemo na predhodno okno, s pritiskom na puščico, obrnjeno v desno, se premaknemo na

naslednje okno, s pritiskom na hišo se vrnemo na osnovno okno, s pritiskom na zobnik pa se premaknemo na stran z nastavitvami. Navigacija je na vseh straneh enaka.



Slika 4.37: Panel mansarda



Slika 4.38: Panel nadstropje

S pritiskom na običajno luč se nam odpre pojavno okno (ang. pop-up) , ki je prikazano na sliki 4.39, preko katerega prižigamo in ugašamo luči.



Slika 4.39: Pojavno okno navadna luč a) prižgana, b) ugasnjena

S pritiskom na tipko on/off spremenimo stanje luči. Preko Etherneta se pošlje bit *NE_luc_kopalnica* prvemu krmilniku, ta pa jo pošlje krmilniku, ki vodi določeno luč. Če je luč prižgana, se ob pritisku luč ugasne in obratno. Ko končamo nastavljanje luči, za izhod iz pojavnega okna pritisnemo križec v desnem zgornjem kotu. Povezava na krogec, ki predstavlja stanje luči na pojavnem oknu, je enaka, kot na sliki mansarda oziroma nadstropje.

S pritiskom na luč, vezano na modul za zatemnjevanje (zraven krogeca je še vrednost, ki v procentih pove s kakšno intenzivnostjo sveti luč), se nam odpre pojavno okno na sliki 4.40, s katerim prižigamo ali ugašamo luči ter nastavljamo zeleno intenzivnost.



Slika 4.40: Pojavno okno luči z možnostjo zatemnjevanja a) prižgana, b) ugasnjena

S pritiskom na tipko on/off spremenimo stanje luči (velja kot pri navadni luči). Če je luč prižgana, se ob pritisku luč ugasne in obratno. Zraven luči je številka, ki v procentih pove, s kakšno intenzivnostjo luč sveti (beseda *NL_prikaz_panel*). S pritiskom na skalo se odpre še eno pojavno okno, ki ga prikazuje slika 4.41. Želena intenzivnost svetlobe nastavimo s tipkama gor in dol. Ko dosežemo želeno intenzivnost svetlobe, pritisnemo tipko close in pritisnemo tipko OK, da potrdimo želeno intenzivnost svetlobe. S tem pošljemo bit *NE_ok_panel* prvemu krmilniku, ta pošlje zahtevo ustreznemu krmilniku, ki prepíše vrednost besede *NE_vred_panel* v besedo *Nastavi_vred_L2*. Beseda *Nastavi_vred_L2* se nato vpiše v števec in števec prepíše vrednost na izhod. Ko končamo nastavljanje luči z možnostjo zatemnjevanja, za izhod iz pojavnega okna pritisnemo križec v desnem zgornjem kotu.



Slika 4.41: Nastavljanje intenzivnosti svetlobe luči z možnostjo zatemnjevanja

S pritiskom na žaluzijo se nam odpre pojavno okno na sliki 4.42, s katerim dvigamo ali spuščamo žaluzije.



Slika 4.42: Dviganje in spuščanje žaluzij

S pritiskom na tipko gor se pošlje prvemu krmilniku ukaz (*NE_panel_z_gor*) za dviganje žaluzije, ta pa jo pošlje krmilniku, ki krmili žaluzijo. Nato se začne izvajati logika, ki je opisana v poglavju 4.4.3. Enako velja za spuščanje žaluzij. Na desni strani je stanje zadnjega ukaza, npr.: če smo zadnjič spuščali žaluzijo, je obarvana spodnja puščica in obratno. Povezava je enaka kot na oknih mansarda in nadstropje.

4.8.3 Ambienti

Slika 4.43 prikazuje stran, kjer na zaslonu zbiramo ambiente. S pritiskom na gumb (Zabava, Večerja, Gledanje TV, Dnevni) preklaplamo med ambiente. Ko je pritisnjena tipka za izbiro ambienta, se pošlje prvemu krmilniku samo en bit za izbrani ambient (gumbi za ambiente so vezani na različne bite). Prvi krmilnik pošlje ta bit vsem ostalim trem krmilnikom. Ko

krmilniki prejmejo bit, z lokalno logiko izvedejo akcije (prižiganje, ugašanje in nastavljanje intenzivnosti osvetlitve luči ter dviganje in spuščanje žaluzij). V programu so nastavljeni štiri ambient, ki so fiksni. Z manjšimi spremembami v programu, bi se lahko parametre ambientov nastavljalo, oziroma bi se kakšnega še dodalo. Ko izberemo določen ambient, ta prižge luči, ki so vključene v izbrani ambient, ostale luči, ki se upravljajo z drugimi ambient, pa se izklopijo.



Slika 4.43: Izbira ambientov

4.8.4 Nastavitve

Slika 4.44 prikazuje osnovno stran nastavitvev. S pritiskom na gumb prehajamo na ostale nastavitvene strani (urnik, ogrevanje, ura/datum, hiša). S pritiskom za zobnik v sivi navigacijski vrstici se nam odpre osnovna stran z nastavitvami.



Slika 4.44: Osnovna stran z nastavitvami

4.8.5 Nastavitve – hiša

Slika 4.45 prikazuje stran na zaslonu, kjer lahko vklopimo ali izklopimo hišo, dvigujemo ali spuščamo vse žaluzije ter izklopimo vse luči v hiši. S pritiskom na gumb Vse žaluzije dol bomo spustili vse žaluzije v hiši. Prvemu krmilniku se pošlje samo en bit *NE1_Z_vseDOL*, ta pa pošlje vsem ostalim trem krmilnikom bit *NLI_1_noB004*. Ko krmilniki prejmejo bit, z lokalno logiko izvedejo spuščanje vseh žaluzij. Ko krmilniki prejmejo bit *NLI_1_noB004*, se

začne izvajati program kot pri pritisku na tipko. S pritiskom na gumb Vse žaluzije gor bomo dvignili vse žaluzije v hiši. Velja enaka logika kot pri spuščanju žaluzij.

S pritiskom na gumb Vklop hiše ali na gumb Izklop hiše vklopimo ali izklopimo hišo. Ko je hiša vklopljena, je ozadje gumba Vklop hiše obarvano zeleno (kot na sliki 4.45), ko pa je izklopljena, pa je ozadje za gumbom Izklop hiše obarvano rdeče. Animacija za gumboma je narejena z enim bitom (*Stanje_Hisa_on_off*). Če je bit postavljen, potem je ozadje gumba Vklop hiše zeleno, ozadje gumba Izklop hiše pa rjavo. Če bit ni postavljen, je ozadje gumba Izklop hiše obarvano rdeče, ozadje gumba Vklop hiše pa rjavo. Akcije, ki se zgodijo ob vklopu oziroma izklopu hiše, so opisane v poglavju 4.3. S pritiskom na gumb Izklop vseh luči izklopimo vse luči v hiši (razen osvetlitev hišne številke in talnih lučk zunaj – te se samodejno prižigajo glede na zunanjo svetlobo). Prvemu krmilniku se pošlje bit *NE_ugasni_luci*, ta pa pošlje vsem ostalim trem krmilnikom bit *NL_ugasni_luci*.



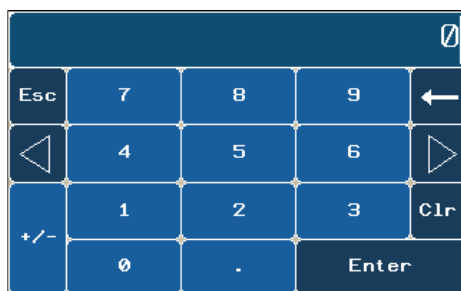
Slika 4.45: Nastavitve hiše

4.8.6 Nastavitve – ura/datum

Slika 4.46 prikazuje stran na zaslonu, kjer lahko nastavimo želeno uro in datum. Datum lahko popravimo s pritiskom na okno dd in nastavimo točen dan, na okno mm in nastavimo mesec ter na okno ll in nastavimo leto. S pritiskom na okna se nam odpre številčnica prikazana na sliki 4.47. Vnesemo želeno vrednost in pritisnimo enter. Ko nastavimo želeni datum (dan, mesec, leto), pritisnemo tipko potrdi datum. Program, ki se izvede ob pritisku na gumb potrdi datum, je opisan v poglavju 4.1.5. Enako velja za nastavitve ure.



Slika 4.46: Nastavitve datuma in ure

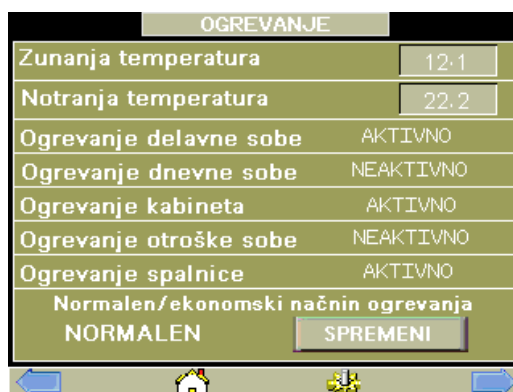


Slika 4.47: Številčnica

4.8.7 Nastavitve – ogrevanje

Na sliki 4.48 je stran na zaslonu, kjer sta prikazani zunanja (*PO1_zun_temp*) in notranja temperatura (*AO2_vhod*) ter trenutno ogrevani prostor (*NE_stanje_soba1* - *NE_stanje_soba5*). Za prikaz notranje temperature skrbi temperaturni senzor v dnevni sobi, za prikazovanje zunanje temperature pa skrbi zunanja enota PO2. Del programa, ki se nanaša na ogrevanje, je opisan v poglavju 4.7.

Če je zraven prostora napis AKTIVNO, se prostor ogreva (*DO7_vklop_ven_1* = 1), sicer se prostor ne ogreva in je zraven prostora napis NEAKTIVNO (*DO7_vklop_ven_1* = 0). S pritiskom na gumb SPREMENI izbiramo med normalnim in ekonomičnim načinom (prvemu krmilniku se pošlje bit *NE1_ekon_on_off*). V katerem načinu je ogrevanje, prikazuje napis levo od gumba (krmilnik pošlje zaslonu bit *NES_ekonomski*).



Slika 4.48: Nastavitve ogrevanje

4.8.8 Nastavitve – urnik žaluzije

Slika 4.49 prikazuje stran na zaslonu, kjer lahko nastavimo želeni čas avtomatskega spuščanja in dviganja žaluzij. Čas nastavimo enako, kot je zgoraj opisano za uro in datum.



Slika 4.49: Nastavi urnik za žaluzije

S pritiskom na tipko SPREMENI 1 preklapljamemo med urnik omogočen in urnik onemogočen (krmilniku se pošlje bit NE_Z_Urniki). Stanje prikazuje napis levo od gumba (krmilnik pošlje NE_stanje_urnik). Če je $NE_stanje_urnik = 1$ so urniki omogočeni, sicer so onemogočeni.

S pritiskom na tipko SPREMENI 2 preklapljamemo med poletnim in zimskim režimom. Stanje prikazuje napis levo od gumba (krmilnik pošlje $NES_Z_l_zimski$). Če je $NES_Z_l_zimski = 1$, so urniki omogočeni, sicer so onemogočeni.

5. Sklepne ugotovitve

V diplomskem delu je opisan sistem, ki sem ga naredil v podjetju Liko Pris d.o.o. za mlado družino v Ljubljani. Sistem deluje v celoti od oktobra 2008. Zaenkrat deluje brez težav in je v zadovoljstvo uporabnikom. Vendar je bilo potrebno veliko dela, da je sistem v celoti zaživel. Problemi so se začeli, ko napisani program ni deloval tako, kot bi bilo smiselno. Odkril smo več napak na krmilni opremi sistema. Pomanjkljivosti sem posredoval podjetju Smarteh d.o.o., ki jih je uspešno odpravilo pred zagonom sistema. Ko je bil programski del zaključen, je sledil zagon in z njim nove težave. Pri testiranju tipk, luči in žaluzij se je pokazalo, da nekaterih luči (ki so bile v signalni listi) ni, nekatere so nove, spet druge so povezane na drugo tipko. Ko je bila vsaka žica, vsak bit na svojem mestu, je sledilo testiranje funkcij. Pri testiranju se je kdaj tudi pokadilo. Uničili smo kar nekaj modulov, preden smo ugotovili, da je napaka na strojni opremi krmilja. Problemi so nastajali tudi zato, ker na začetku nisem razumel, kako bo sistem v celoti deloval. Vsak dan sem se spomnil in dodal nekaj novih funkcij, ki so vplivale na druge dele programa in mi tako podrle ostalo logiko. Nekaj težav je bilo tudi s komunikacijo LON. Na mreži LON so se ob ponovnem zagonu krmilnikov pojavljali naključni biti, ki so prožili izvajanje določenega dela kode. Ob vsakem problemu sem pomislil, da vsaka šola nekaj stane in da je to investicija za naprej. Ta projekt je terjal veliko časa in potrpljenja, vendar sem si nabral veliko novih izkušenj in se ob tem naučil veliko novih stvari, ki mi bodo prihranile čas v prihodnje.

Zdi se, da ljudje še vedno ne zaupajo tehnologiji. Ko sem na spletnih klepetalnicah prebiral teme o pametnih hišah, sem v vsakem pogovoru našel skeptika, ki je bil mnenja, da več ko je elektronike v hiši, več svari se lahko pokvari. V osnovi bi se strinjal s to trditvijo, vendar se nenehnim napakam izognemo, če izberemo preverjeno strojno opremo. Tako imamo veliko manj možnosti, da se bo sistem vseskozi kvaril. Po drugi strani me je presenetilo veliko zanimanje ljudi za sisteme pametnih hiš in veliko število slovenskih podjetji, ki se ukvarjajo s hišno avtomatizacijo. V diplomskem delu sem opisal tri podjetja, našel pa sem jih vsaj deset. Čeprav se v Sloveniji že kar nekaj časa govori o pametnih hišah, so še vedno zelo redke. Menim, da je vzrok za to cena takega sistema. Predvidevam, da bodo pametne hiše v prihodnosti vedno bolj prisotne, saj bo na trgu prisotnih vedno več ponudnikov sistemov in opreme za hišno avtomatizacijo. Več ponudnikov pomeni večjo konkurenco, kar lahko posledično privede do nižjih cen takih sistemov. Z nižjimi cenami pa bo pametna hiša postala dosegljiva širši množici. Verjetno se graditelji novih hiš v prihodnosti ne bodo odločali ali dati hiši inteligenco ali ne, ampak se bodo odločali o tem, kaj vse naj bi njihova hiša znala. Nekoč, ko se bom sam znašel v vlogi graditelja, bom vsekakor pregledal in pretehtal vse možnosti za avtomatizacijo svoje hiše. V prvi vrsti me bodo zanimali prihranki in varnost, šele nato udobje in prestiž.

6. Viri

- [1] TRON Intelligent House, dostopno na: <http://tronweb.super-nova.co.jp/tronintlhouse.html>
- [2] E. Jakupović, Digitalni dom med domišljijo in resničnostjo; Naravni dialog med človekom in strojem, Časnik Finance, 1.8.2006, številka 146, str. 17 - 19
- [3] U. Vovk, seminar : Pametna hiša, 2001. dostopno na: <http://luks.fe.uni-lj.si/sl/studij/SUIS/seminarji/vovku/>
- [4] Inštitut Republike Slovenije za rehabilitacijo, Dom IRIS. dostopno na: <http://www.ir-rs.si/filelib/dom-iris/dom-iris.htm>
- [5] Podjetje KomfortKlik d.o.o. , dostopno na: <http://www.comfortclick.com/resitve.html>
- [6] Podjetje Gorenje, Novice 18.6.2007, dostopno na: <http://www.gorenje.si/novice?nid=468>
- [7] Sun Microsystems, Novice 12.3 2008 dostopno na: <http://zda2008.fri.uni-lj.si/?q=sl/prva&page=4>
- [8] 93. Uradni list RS, št. 93/2008 z dne 30.09.2008, dostopno na: <http://www.uradni-list.si/1/objava.jsp?urlid=200893&stevilka=3939>
- [9] Smarteh d.o.o. , dostopno na: <http://www.smarteh.si>
- [10] Robotina d.o.o., dostopno na: <http://www.robotina.si>
- [11] S. Peršin, Sodobni sistemi hišne avtomatizacije , Avtomatika, 42 (2005), str. 7 – 12, dostopno na: http://89.142.199.46/ARHIV_AVTOMATIKA/A42.pdf
- [12] M. Robič, J. Torkar, M. Erklavec, Strokovno posvetovanje Slovenskega društva za daljinsko energetiko, Energetika v našem okolju, Portorož, 12. – 13. marec 2001, dostopno na: <http://www.el-tec-mulej.si/data/upload/sdde01-LonWorks.pdf>

Izjava o avtorstvu

Izjavljam, da sem diplomsko delo z naslovom Sistem za vodenje in nadzor pametne hiše izdelal samostojno pod mentorstvom doc. dr. Uroša Lotriča. Hkrati soglašam z javno objavo diplomskega dela v elektronski zbirki Dela FRI.

Rok Stražišar

