Upravljanje radom vodenog tuša na solarnu energiju

Žagar, Luka

Undergraduate thesis / Završni rad

2017

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

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:881557

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

zir.nsk.hr

Image not found or type unknown

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

Sveučilišni studij računarstva

UPRAVLJANJE RADOM VODENOG TUŠA NA SOLARNU ENERGIJU

Završni rad

Luka Žagar

Osijek, 2017.

SADRŽAJ

1. UVOD 1

1.1. Zadatak završnog rada 1

2. VODENI TUŠ NA SOLARNU ENERGIJU 2

3. UPRAVLJANJEM RADOM VODENOG TUŠEM NA SOLARNU ENERGIJU 5

3.1. Sklopovlje 5

3.1.1. Croduino Basic 2 6

3.1.2. Dioda za zaštitu od inducirane struje 8

3.1.3. Djelitelj napona 9

3.1.4. MOSFET traznistor kao sklopka 9

3.2. Programska podrška 10

3.2.1. Globalne varijable i Setup() 11

3.2.2. Loop() 12

3.2.3. Prekidna rutina 13

3.2.4. Stanje mirovanja 14

4. TESTIRANJE 15

5. ZAKLJUČAK 17

LITERATURA 18

SAŽETAK 19

ABSTRACT 20

ŽIVOTOPIS 21

PRILOZI 22

1. UVOD

Cilj ovog završnog rada je analizirati i opisati probleme trenutnog sklopovlja vodenog tuša na solarnu energiju. Prema zahtjevima potrebno je izraditi sklopovlje i programsku podršku koja će upravljati radom vodenog tuša na solarnu energiju. Upravljanje podrazumijeva upravljanje električnim ventilom, ekranom i svjetlima uz što manju potrošnju električne energije. U radu su opisani svi dijelovi sklopovlja, te je opisana programska podrška za upravljanje radom vodenog tuša. Umjesto Arduino razvojne platforme koju koristi trenutno sklopovlje za „mozak“ sklopovlja koristiti će se Croduino Basic 2 razvojna platforma. Ona je hrvatska verzija Arduino razvojne platforme i ima sve mogućnosti kao i Arduino razvojna platforma. U programskoj podršci koristiti će opcija stanja mirovanja Croduino Basic 2 razvojne platforme kojom se smanjuje potrošnja električne energije sklopovlja. Svako stanje mirovanja mora imati neke prekidnu rutinu koja budi razvojnu platformu iz stanja mirovanja. Za potrebe upravljanja radom koristiti će se dvije prekidne rutine

1.1. Zadatak završnog rada

U ovom završnom radu razmatra se izrada arduino sustava za upravljanje radom vodenog tuša na solarni pogon. Potrebno je razmotriti potrošnju vode iz spremnika te električne energije iz baterije.

2. VODENI TUŠ NA SOLARNU ENERGIJU

Tvrtka Pitaya Solutions je 2015. godine započela razvoj vodenog tuša na solarnu energiju UNNI. Primarna djelatnost tvrtke je proizvodnja vodenih tuševa na solarnu energiju, dok sekundarne djelatnosti uključuju 3D modeliranje, izradom 3D kataloga i izradom web-stranica. Ideja vodenog tuša na solarnu energiju nastala je istraživanjem tržišta urbanističke opreme za plaže i kupališta na Jadranu. Vodeni tuš je morao ispunjavati određene uvjete kao što su moderan dizajn, jednostavna ugradnja, energetsku neovisnost, ekološku prihvatljivost i ekonomsku isplativost. [1]

Vodeni tuš na solarnu energiju UNNI („pametni tuš“) prikazan na slici 2.1 je prvi komercijalni vodeni tuš tvrtke Pitaya Solutions. Za svoj rad koristi energiju Sunčevog zračenja čime je osigurana energetska neovisnost i ekološka prihvatljivost. Ima ugrađeni ekran na kojem se prikazuju razne informacije vezane za mjesto u kojem se nalazi, vremensku prognozu, temperaturu zraka itd. UNNI ima opciju za spajanje s internetom, odnosno u sebi sadrži kontrolnu točku za povezivanje s internetom. Specifikacije prikazane u tablici 2.1.



Sl. 2.1.: Vodeni tuš na solarnu energiju UNNI.[1]

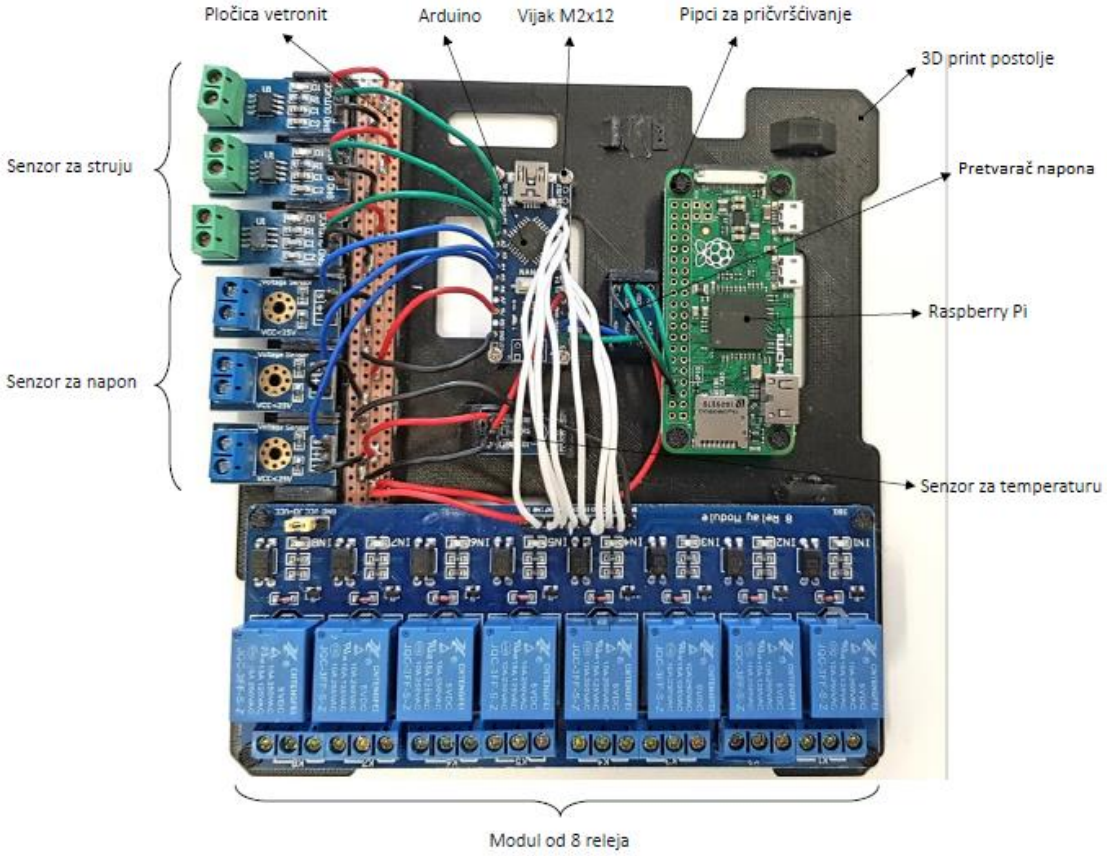
[1]https://i.imgur.com/sJs3EmZ.png

Naziv	Model	Količina	Opis
Solarni panel	Rađen po narudžbi	1	Monokristal, maksimalni napon 21 V, snaga 55 W
Regulator punjenja baterije	LS1024E	1	Smanjuje napon solarnog panela na 12 V, maksimalna struja 10 A
Baterija	LPCG12-50	2	Kapacitet 50 Ah,
Senzor napona	Djelitelj napona	3	Smanjuje napon za 3.221 ulazne vrijednosti
Senzor struje	*ACS712ELC-05B*	3	Mjeri struju do 5 A, osjetljivost mu je 185 mA
Infracrveni senzor	12V DC Human Body Motion Sensor Module Switch Board Infrared Sensor	1	Detektira presijecanje infracrvene zrake
Senzor temperature	DS18B20 temperature sensor	1	Mjeri temerature od -55 °C do +125 °C, preciznost ± 0.5 °C od -10 °C do +85 °C

Tab. 2.1.: Specifikacije vodenog tuša na solarnu energiju UNNI.[1]

Trenutnim radom vodenog tuša na solarnu energiju upravlja Arduino Nano razvojna platforma. Arduino Nano razvojna platforma prikuplja podatke od senzora napona, senzora struje, senzora temperature i infracrvenog senzora. Na osnovu tih podataka Arduino Nano razvojna platforma uključuje određena trošila vodenog tuša, pušta vodu za tuširanje itd. Senzor temperature očitava temperaturu unutra kućišta te ukoliko je temperatura prevelika uključuje se ventilator.

Trenutno sklopovlje prikazano na slici *2.2* koristi previše senzora koji očitavaju podatke koji se ne koriste za rad vodenog tuša. Također, nastaje problem kada se osoba prestane tuširati te voda koja teče prekida infracrvenu zraku i infracrveni senzor šalje podatak Arduino Nanu da relej koji upravlja električnim ventilom treba ostati zatvoren.



Sl. 2.2: UNNI sklopovlje[2].

[2]https://imgur.com/IFN4ZPa

3. UPRAVLJANJEM RADOM VODENOG TUŠEM NA SOLARNU ENERGIJU

U ovom poglavlju se razmatra cjelokupno upravljanje vodenog tuša na solarnu energiju. Upravljanje vodenog tuša na solarnu energiju sastoji se od detektiranje osobe ispod vodenog tuša, rješavanje problema prijašnjeg sklopa koji je imao problema s infracrvenim senzorom, efikasno iskorištavanje baterije, reagiranje izmjene dnevnih ciklusa tako što se pale svjetla vodenog tuša, a gasi ekran i obratno ovisno da li je dan ili noć.

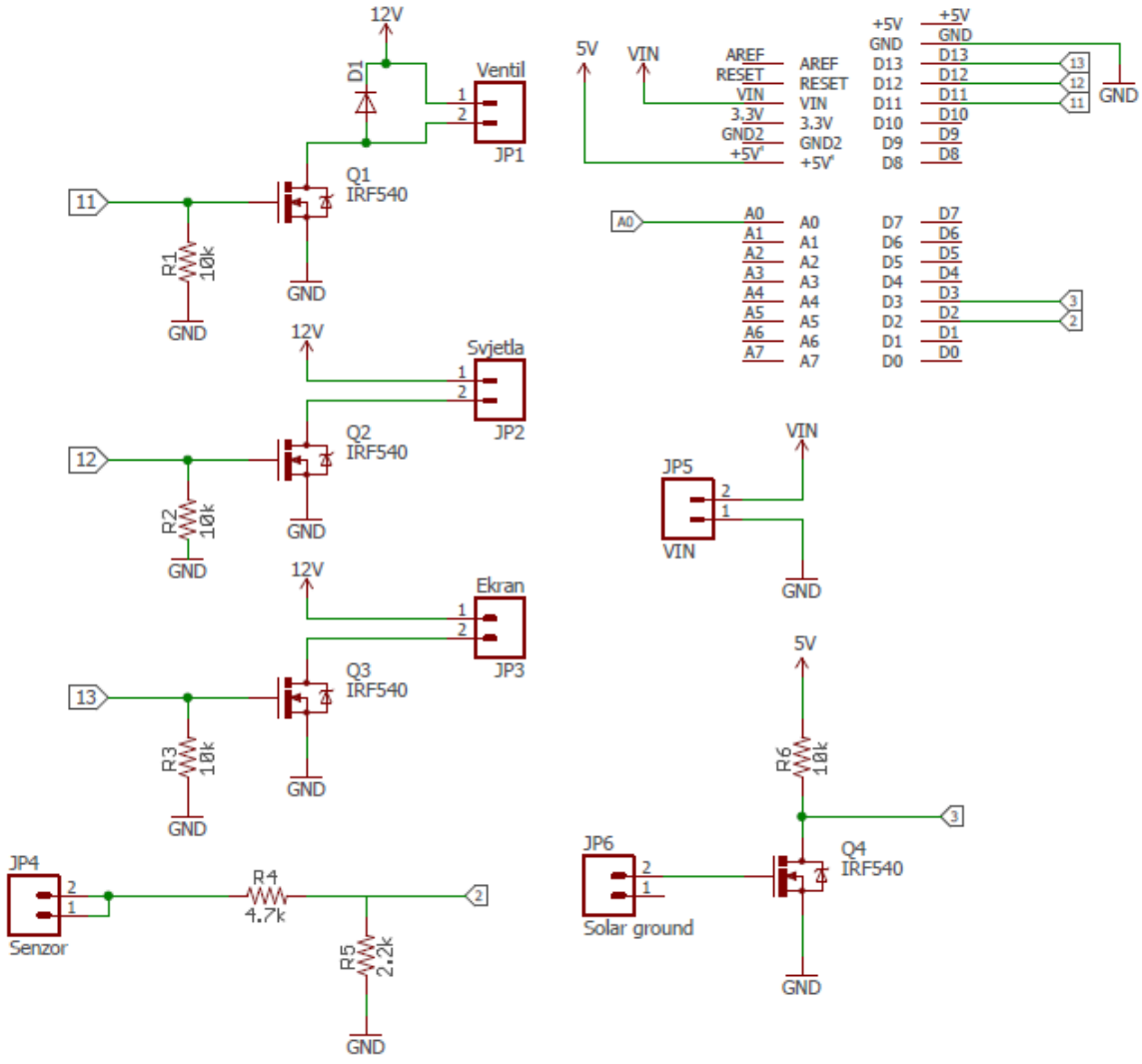
3.1. Sklopovlje

Sklopovlje se sastoji od nekoliko dijelova povezanih u cjelinu koji će putem Croduino Basic 2 razvojne platforme upravljati radom vodenog tuša. Sklopovlje upravlja radom električnog ventila, ekrana i svjetla. Zahtjev na sklopovlje je što manja potrošnja električne energije i veća autonomija vodenog tuša na solarnu energiju.

Sklopovlje se sastoji od:

1. Croduino Basic 2 prototipne pločice
2. infracrvenog senzora pokreta
3. djelitelja napona
4. MOSFET tranzistora kao sklopki
5. diode za negiranje povratnog napona

prikazanih na slici 3.1.



Sl. 3.1.: Shema sklopovlja

3.1.1. Croduino Basic 2

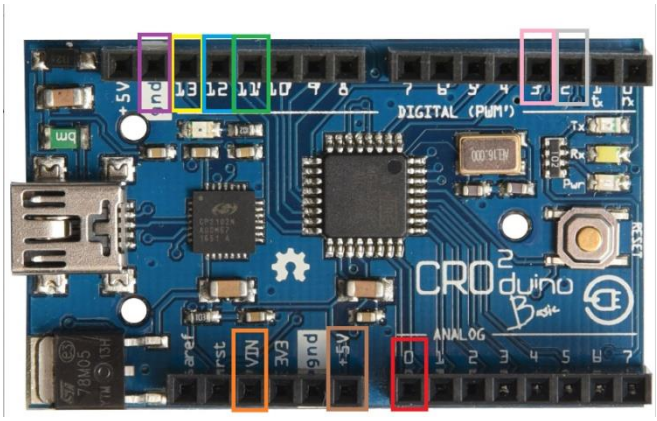
Croduino je hrvatska verzija Arduina koji je razvojna platforma namijenjena za kreiranje i razvoj elektroničkih projekata. Arduino je tip tehnologije koju može koristiti svatko što znači da korisnici imaju pristup originalnim nacrtima sklopa te ga mogu izmjenjivati i koristiti prema vlastitim potrebama. Sastoji se od računalnog sklopovlja odnosno fizičkog elektronički programibilnog strujnog kruga ili mikroupravljača i programskog dijela koji se pokreće na računalu. Namijenjen je korisnicima koji nikada nisu imali doticaja s elektroničkim komponentama i programiranje, prvenstveno zbog svoje lake krivulje učenja, mogućnosti programiranja na različitim platformama (Windows, Linux, Mac...), te pristupačnosti (cijena, zajednica...). Croduino se može koristiti u jednostavnim i kompliciranim projektima ovisno o znanju programera. Za upravljanje radom

vodenog tuša na solarnu energiju korišten je Croduino Basic 2 koji je identičan s Arduino Nano koji koriste ATmega328P procesor. [2]

Croduino Basic 2 sastoji se od mikro upravljača ATmega328P koji kontrolira što se događa na pinovima, očitava signale, na njega se sprema kod s računala itd, USB pretvornika koji je zadužen za prevođenje računalnog koda u strojni jezik, USB poveznik s kojim se povezuju računalo i mikro upravljača. Također, sastoji se od analognih pinova, digitalnih pinova, +5 V istosmjernog napona koji se koristi za spajanje senzora, komponenti i pokretača, te proizvodi struju od 400 mA. Gnd pin je negativni pol (*eng.ground*) istosmjernog napona, +3.3 V istosmjernog napajanja, V_{in} je ulaz koji služi za napajanje s baterije napona između 7-24 V. Za ovaj sklop su korišteni:

- digitalni pin 2 – spojen s infracrvenim senzorom (označen sivom bojom)
- digitalni pin 3 – spojen na odvod MOSFET tranzistora (označen rozom bojom)
- digitalni pin 11 - spojen na upravljačku elektrodu MOSFET tranzistora koji blokira ili propušta struju prema električnom ventilu (označen zelenom bojom)
- digitalni pin 12 – spojen na upravljačku elektrodu MOSFET tranzistora koji blokira ili propušta struju prema svjetla(označen svijetlo plavo bojom)
- digitalni pin 13 – spojen na upravljačku elektrodu MOSFET tranzistora koji blokira ili propušta struju prema ekranu(označen žutom bojom)
- pin V_{in}. – spojen s baterijom koja služi kao izvor napajanja(označen narančastom bojom)
- pin gnd – spojen sa svim negativnim polovima(označen ljubičastom bojom)
- pin +5V – spojen s odvodom MOSFET tranzistora (označen smeđom bojom)

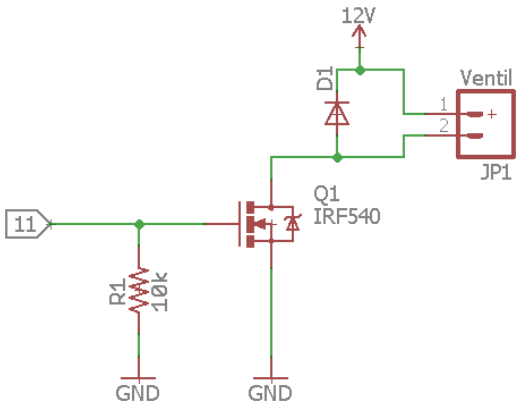
prikazani na slici 3.2.



Sl. 3.2.: Croduino Basic 2 razvojna pločica.[3]

3.1.2. Dioda za zaštitu od inducirane struje

Kada se u električnom krugu s induktivnim elementom naglo prekine dovod električne energije u električnom krugu se pojavljuju nagli porasti napona. To se događa jer induktivni element djeluje kao spremnik magnetske energije jer se struja u induktivnom elementu ne može trenutno promijeniti. Potrebno je neko vrijeme da struja prestane teći, preciznije potrebno je 5 vremenskih konstanti τ[3].Kao sigurnosna mjera koristi se dioda za zaštitu od inducirane struje koja je u paralelnom spoju s induktivnom elementom u ovom slučaju električnim ventilom.(slika *3.3.*).



Sl. 3.3.: Dioda za zaštitu od inducirane struje

[3]https://e-radionica.com/hr/croduino-basic2.html

3.1.3. Djelitelj napona

U elektronici, djelitelj napona je pasivni linearni krug koji proizvodi izlazni napon(V_{out}) koji je dio njegovog ulaznog napona (V_{in}). Podjela napona je rezultat raspodjele ulaznog napona među otpornicima i računa se prema formuli (3-1). Jednostavan primjer djelitelja napona su dva otpornika spojena u seriju. Djelitelj napona otpornika najčešće se upotrebljavaju za stvaranje referentnih napona, ili za smanjenje veličine napona tako da se može izmjeriti.[4]

$$V_{out} = V_{in} * \frac{R_2}{R_1 + R_2} \tag{3-1}$$

gdje je:

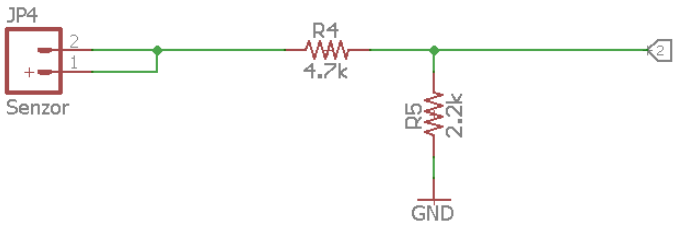
V_{out} – smanjeni napon odnosno napon iz djelitelja napona,

V_{in} – ulazni napon ili napon baterije

R_1 – vrijednost prvog otpornika

R_2 – vrijednost drugog otpornika

Djelitelj napon se u sklopovlju koristi za detektiranje napona sa infracrvenog senzora . Slika 3.4. prikazuje shemu djelitelja napona.

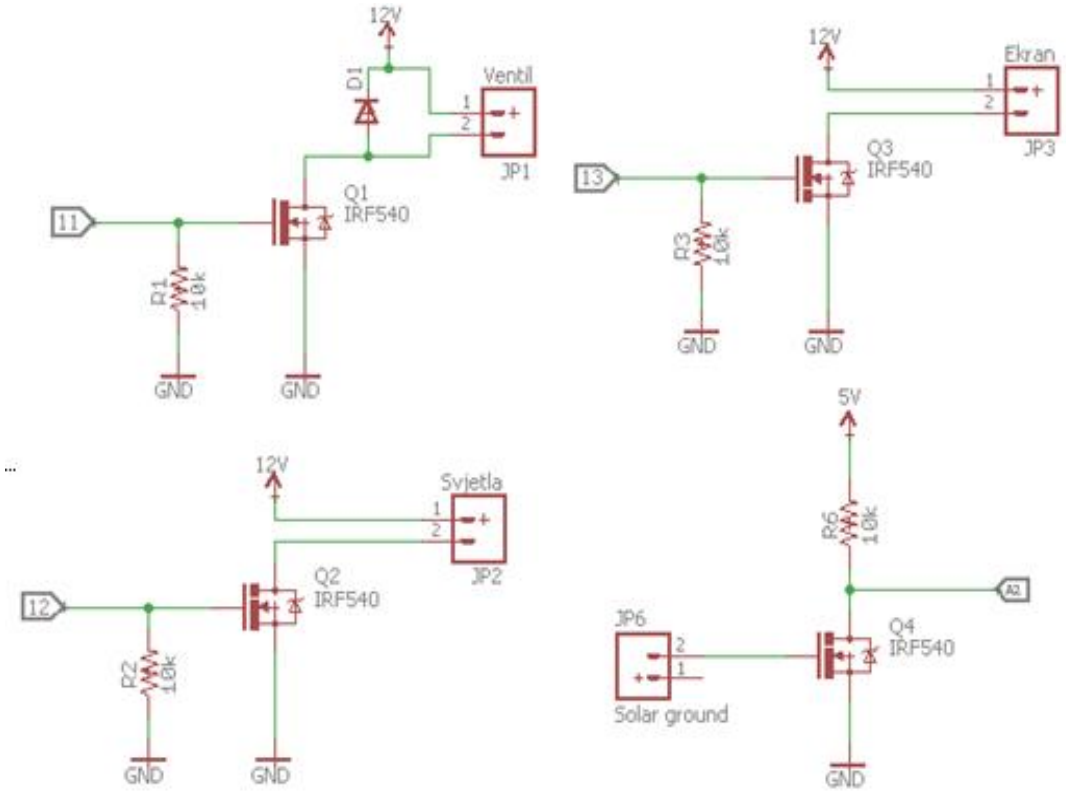


Sl. 3.4.: Shema djelitelja napona.

3.1.4. MOSFET tranzistor kao sklopka

Tranzistor s učinkom polja i izoliranom upravljačkom elektrodom ili MOSFET tranzistor upravlja se naponom upravljačke elektrode. Tranzistor ima tri elektrode uvod S, odvod D i upravljačku elektrodu G. MOSFET tranzistor koji se koristi kao sklopka najčešće bude izvedene u spoju zajedničkog uvoda.[5]

Za upravljanje elementima vodenog tuša na solarnu energiju koristi se n-kanalni MOSFET tranzistor IRF540. U električnom krugu postoje četiri IRF540 MOSFET tranzistora od koji svaki upravlja pojedinim dijelom vodenog tuša. Slika *3.5.* prikazuje kako se spajaju MOSFET tranzistori.



Sl. 3.5.: MOSFET kao sklopka.

3.2. Programska podrška

Programska podrška je pisana u programu Arduino IDE (*eng.Integrated Development Environment*). Arduino zajednica je programa otvorenog koda što znači da su sva moguća rješenja dostupna na internetu te ih treba preoblikovati za projekat koji je potreban korisniku.

Zahtijeva se što učinkovitije iskorištavanje baterije, da ekran preko dana bude uključen, a svjetla ugašena. Tijekom noći je potrebno uključiti svjetla, a isključiti ekran. Potrebno je riješiti problem neprestanog toka vode uzrokovanog infracrvenim senzorom.

3.2.1. Globalne varijable i Setup()

Globalne varijable su varijable koje se deklariraju izvan glavnog dijela programa i mogu koristiti u svakom dijelu programa. U programu postoje tri globalne varijable od kojih su dvije varijable tipa *volatile*. Tip varijable *volatile* se koristi kada se varijabla ne koristi u glavnom dijelu programa *setup()* nego mijenja svoje vrijednosti u nekom drugom dijelu programa. Varijable *interruptTime* označava kada se dogodila prekidna rutina koja će se kasnije koristiti u glavnoj dijelu programa *setup()* te služi za određivanje vremena trajanja rada električnog ventila. Varijabla tipa *boolean* naziva *ventilActivated* koristi se kao pokazivač da se dogodila prekidna rutina, gdje se mijenja iz stanja *false* u stanje *true*. Kasnije u glavnom programu se s obzirom na varijablu *ventilActivated* uključuje električni ventil.

U *setup()* dijelu programa gdje se postavljaju varijable, postavljanje pinova, počinju koristiti biblioteke itd. U ovom dijelu programa postavlja se upravitelj prekidom rutinom koja reagira na promjenu digitalnog signala s infracrvenog senzora i kad se dogodi promjena izvršava se prekidna rutina *funkc(). PinMode()* funkcija određuje koji pinovi će biti ulazni signali, a koji izlazni signali. Na slici 3.6. prikazan je kod globalnih varijabli i *setup()* dijela programa.

Linija	***Kod***
1:	`unsigned long sleepTime = 60000;`
2:	`volatile long interruptTime = 0;`
3:	`volatile bool ventilActivated = false;`
4:	`void setup() {`
5:	`Serial.begin(115200);`
6:	`pinMode(IRsenzor, INPUT);`
7:	`pinMode(svjetla, OUTPUT);`
8:	`pinMode(ekran, OUTPUT);`
9:	`pinMode(ventil, OUTPUT);`
10:	`pinMode(senzorSvjetla, INPUT);`
11:	`attachInterrupt(digitalPinInterrupt(IRsenzor), funkc, CHANGE);`

Sl. 3.6.: Kod globalnih varijabli i setup().

3.2.2. Loop()

U glavnom programu, koji se naziva *loop(),* provjerava se je da li infracrveni senzor prešao iz stanja digitalne nule u stanje digitalne jedinice. Ako je, prekidna rutina u kojoj se nalazi varijabla *ventilActivated* postala je *true* te će se u *loop*() dijelu programa pod tim uvjetom obaviti određeni blok koda (slika 3.7.). Taj blok prvo sadržava provjeru protekloga vremena od trenutka kada se dogodila promjena stanja. Za tu provjeru koristi se ugrađena funkcija *millis()* koja vraća broj proteklih milisekunda od trenutka uključivanja pločice. U *while()* petlji program ostaje sve dok ne prođe zadan broj vremena od 15000 ms ili 15 s. Nakon što je vrijeme isteklo, električni ventil se gasi te se događa pauza od 2000 ms ili 2 s. Nakon toga *boolean* varijabla *ventilActivated* se vraća u stanje *false* kako prekidna rutina mogla ponovo dogoditi.

Kada je proizvodnja solarnog panela manja od 10.5 V MOSFET tranzistor kojem je odvod spojen na digitalni pin 3 je u stanju zapiranja tada varijabla *senzorSvjetla = HIGH.* Na digitalni pin 3 je tada spojen pozitivan napon od 5 V koji se očitava kao digitalna jedinica. Tada se uključuju svjetla, a ekran gasi. Nakon što se proizvodnja napona solarnog panela poveća iznad granice od 10.5 V MOSFET tranzistor je u stanju vođenja i digitalni pin 3 je spojen na uzemljenje. U tom trenutku varijabla *senzorSvjetla* prelazi iz stanja *HIGH* u stanje *LOW* i gase se svjetla i pali se ekran.

Linija ***Kod***

```
1:      void loop() {
2:      if (ventilActivated) {
3:      while (millis() - interruptTime < 15000) {
4:      delay(100);}
5:      digitalWrite(ventil, LOW);
6:      delay(2000);
7:      ventilActivated = false;}
8:      if (digitalRead(senzorSvjetla)) {
9:      digitalWrite(svjetla, LOW);
10:     digitalWrite(ekran, HIGH);}
11:     else {
12:     digitalWrite(svjetla, HIGH);
13:     digitalWrite(ekran, LOW);}
14:     sleep.pwrDownMode();
15:     sleep.sleepDelay(sleepTime);
```

Sl. 3.7.: Kod glavnog dijela programa loop().

3.2.3. Prekidna rutina

Prekidna rutina(*eng. interrupts*) je rutina koja naređuje procesoru da prestane s trenutnim zadatkom te da prihvati zadatak većeg prioriteta. Proces koji se tada obavlja zove se upravitelj prekidnom rutinom (*eng. Interrupt Handler*). Svaki upravitelj prekidne rutine ima svoju prekidnu rutinu koja ga aktivira. Najčešće se koriste da bi se procesor probudilo iz stanja mirovanja.

Prekidne rutine se dijele na:

- vremenski prekidne rutine – nakon određenog vremena pojavljuje se prekidna rutina se pokreće upravitelja prekidnim rutinama
- vanjske prekidne rutine – kada se pojavi dogodi neki određeni događaj [6]

Programska podrška sastoji se od vremenski prekidne rutine nakon određenog vremena provjerava proizvodnju napona solarnog panela. Kada napon proizvodnje solarnog panela padne ispod 10.5 V upaliti će se svjetla a ugasiti ekran. Sklop se budi iz stanja mirovanja svakih 60 sekundi.

Druga prekidna rutina koja se koristi je vanjska prekidna rutina infracrvenog senzora prikazana na slici 3.8. Kada se infracrveni senzor aktivira on šalje signal procesoru da se dogodila promjena te da je potrebno pozvati upravitelja prekidnom rutinom. Upravitelj prekidnom rutinom budi procesor iz stanja mirovanja koji šalje digitalnu jedinicu na upravljačku elektrodu MOSFET tranzistora koji je povezan s električnim ventilom. Digitalna jedinica će dovesti MOSFET tranzistor u stanje vođenja. Digitalna jedinica traje četrdeseti sekundi i nakon isteka vremena procesor šalje digitalnu nulu koja će dovesti MOSFET tranzistor u stanje zapiranja. Stanje digitalne nule traje minimalno dvije sekunde, jer je potrebno osigurati da voda ponovno ne aktivira infracrveni senzor pokreta. Nakon isteka vremena od dvije sekunde procesor se vraća u stanje mirovanja te ostaje u stanju mirovanje do pojave novog prekidnog signala sa infracrvenog senzora.

Linija	***Kod***
1:	`void funkc(){`
2:	`noInterrupts();`
3:	`If(digitalRead(IRsenzor) && !ventilActivated){`
4:	`digitalWrite(ventil, HIGH);`
5:	`ventilActivated = true;`
6:	`interruptTime = millis();}`
7:	`interrupts();}`

Sl. 3.8.: Kod prekidne rutine.

3.2.4. Stanje mirovanja

Kako bi se osigurala što manja potrošnja ionako ograničenog izvora napajanja – baterije, na mikro upravljaču je postavljeno stanje mirovanja(*eng. sleep mode*). Ona omogućuje poseban način rada u kojem mikro upravljač troši manje električne energije nego u normalnom načinu rada. To se postiže isključivanjem određenih dijelova mikro upravljača, kao što je analogni-digitalni pretvarač.

Postoji više načina stanja mirovanja sklopa. Svaki od njih isključuje određene dijelove mikro upravljača te što je više dijelova isključeno veća je i ušteda energije. Ovo omogućuje da se odabere najoptimalniji način rada tako da se ostavi upaljeno ono što je potrebno, a ugasi ostatak uređaja. Stanje mirovanja *sleep_mode_pwr_down* je stanje u kojem Croduino Basic 2 razvojna platforma troši 15 mA električne struje. Postoji mogućnost za još manjom potrošnjom električne energije tako da se smanji brzina mikro upravljača i da se ukloni dio komponenti s Croduino Basic 2 razvojne platforme. U ovom slučaju uklanjanje komponenti nije moguće jer je radni napon Croduino Basic 2 razvojne platforme 5 V dok je napon baterije 12 V. Potreban je regulator koji će smanjivati napon sa 12 V na 5 V. On je spojen na pin V_{in} Stanje mirovanja se završava prekidnom rutinom ili nakon isteka određenog vremena.[7]

U kodu korištena je *Sleep_n0m1* biblioteka napisan za Arduino razvojnu platformu koji omogućuje vrlo jednostavno korištenje različitih način stanja mirovanja, trajanja stanja mirovanja ili načina buđenja iz stanja mirovanja[8]. Da bi se mogle koristiti funkcije iz *Sleep_n0m1* potrebno je napraviti objekt klase *Sleep*. Pomoću tog objekta stavljamo Croduino Basic 2 razvojnu platformu u stanje mirovanja pomoću nekoliko naredbi. Prva naredba je *sleep.pwrDownMode()* u kojoj se odabire način stanja mirovanja sklopa. Nakon odabira načina stanja mirovanja *sleep.sleepDelay(sleepTime)* naredba implementira odabrani način stanja mirovanja i varijabla *sleepTime* određuje koliko vremena će sklop provesti u stanju mirovanja prije sljedećeg buđenja. Kod se implementira u glavnoj petlji l*oop()* (slika 3.7.).

4. TESTIRANJE

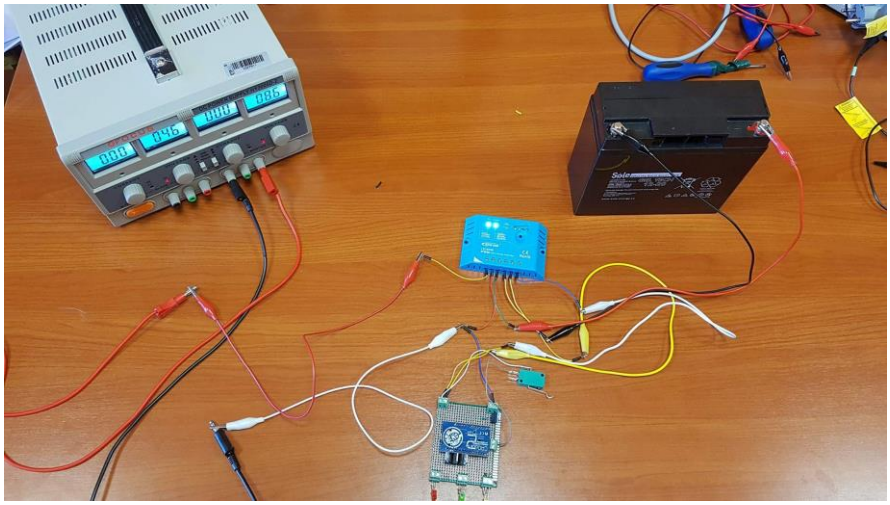
Testiranje sklopovlja se provodilo u simuliranim uvjetima u laboratoriju. Solarni panel simulirana je istosmjerni izvorom napajanja. Trošila su simulirana sa svjetlećim diodama gdje žuta dioda simulira električni ventil, zelena dioda simulira svjetla, crvena dioda simulira ekran. Infracrveni senzor je zamijenjen tipkalom. Testiranje se odvijalo u dvije faze.

U prvoj fazi testiranja se testiralo ispravnost sklopovlja, odnosno da li sklop obavlja funkcije za koje je predviđen. Sklop se spaja prema shemi prikazanoj na slici 3.1. Spojeni sklop prikazan je na slici 3.9.

Napon istosmjernog izvora napajanja mijenjan je između 8 V – 16 V. Kada istosmjerni izvor daje napon od 10.5 V MOSFET tranzistor se dovodi u stanje zapiranja jer je razlika između napona baterije i napona regulatora punjenja baterije manja od 2 V koliko je potrebno MOSFET tranzistoru da provede. Tada zasvjetli crvena dioda koja simulira ekran, a gasi zelena koja simulira svjetla. Napon manji od 10.5 V dovodi MOSFET tranzistor u stanje vođenja te se tada crvena dioda gasi, a zasvijetli zelena. Pritiskom na tipkalo dovodi simulira se signal dolaska osobe pod vodeni tuš i nakon pritiska tipkala zasvijetli žuta dioda koja svijetli 40 sekundi na pojave signala. Nakon isteka 40 sekundi žuta dioda se neće zasvijetliti sljedeće dvije sekunde neovisno o pritisku tipkala.

Druga faza testiranja sklopa je da se prikaže ušteda električne energije pomoću programske podrške i implementiranja stanja mirovanja i prekidne rutine. Na pozitivan pol istosmjernog izvora napajanja spaja se pozitivan pol ampermetra. Na negativan pol ampermetra se spaja V_{in} pin Croduino Basic 2 razvojne pločice. Na gnd pin Croudino Basic 2 razvojne pločice se spaja negativan pol istosmjernog izvora. Istosmjerni izvor daje napon od 12 V.

Sklop u stanju mirovanja troši 15 mA električne struje. Na pojavu prekidne rutine sklop se budi iz stanja mirovanja i troši 27 mA. Stanjem mirovanja postignuta je ušteda od 12 mA.



Sl. 3.9.: Prikaz spojenog sklopa.

5. ZAKLJUČAK

Kao rezultat završnog rada napravljeno je sklopovlje i programska podrška za vodeni tuš na solarnu energiju pomoću Croduino Basic 2 razvojne platforme. Sklopovlje koje je napravljeno ima manje dimenzije od prijašnjeg sklopovlja, sadrži manje komponenti. Programska podrška ispunjava predodređene zahtjeve koji su puštanje vode na pojavu osobe ispred infracrvenog senzora, paljenje ili gašenje ekrana ili svjetla ovisno o dobu dana i smanjena potrošnja energije. Granica iznad koje se pali ekran a gase svjetla je 10.5.V (za slučaj kada je baterija na 100% kapaciteta). MOSFET tranzistoru je potrebno 3 V da pređe iz stanja zapiranja u stanje vođenja. Granica ovisi o naponu baterije. Razlika između napona baterije i napona koji proizvodi solarni panel je napon kojim se upravlja MOSFET tranzistorom. Kada ta razlika bude manja od 3 V MOSFET tranzistor prelazi u stanje zapiranja. Croduino Basic 2 razvojna platforma u stanju mirovanja troši 12 mA manje električne energije.

LITERATURA

[1] http://pitaya-solutions.hr/hr/proizvodi/ - pristupljeno 10. ruajn 2017

[2] Što je Arduino, a što Croduino?, https://e-radionica.com/hr/blog/2015/10/08/sto-je-arduino-i-croduino/, - pristupljeno 18. srpnja 2017.

[3] Inductive Flyback and Flyback Diodes, https://learn.digilentinc.com/Documents/390 - pristupljeno 9. rujna 2017

[4] Voltage divider, https://en.wikipedia.org/wiki/Voltage_divider - pristupljeno 26. lipanj 2017.

[5] M.C. Schneider,MOSFET modeling for circuit analysis and design, World Scientific, 2007

[6] What is an interrupt?, https://learn.adafruit.com/multi-tasking-the-arduino-part-2/what-is-an-interrupt - pristupio 7. rujan 2017

[7] Reducing Arduino Power Consumption, https://learn.sparkfun.com/tutorials/reducing-arduino-power-consumption - pristupljeno 7.rujan 2017.

[8] https://github.com/n0m1/Sleep_n0m1 - pristupljeno 8. Rujan 2017.

[9] Interrupts, https://playground.arduino.cc/Code/Interrupts - pristupljeno 7.rujan 2017.

[10] Using interrupts on Arduino, https://www.allaboutcircuits.com/technical-articles/using-interrupts-on-arduino/ - pristupljeno 7.rujan 2017.

[11] http://www.engblaze.com/hush-little-microprocessor-avr-and-arduino-sleep-mode-basics/ - pristupljeno 8. rujan 2017.

[12] https://playground.arduino.cc/Learning/ArduinoSleepCode - pristupljeno 10.rujan 2017.

[13] https://www.arduino.cc/en/Reference/Interrupts - pristupljeno 7.rujan 2017.

SAŽETAK

U ovom završnom radu opisano je upravljanje radom vodenog tuša na solarnu energiju pomoću Croduino Basic 2 razvojne platforme. Opisano je kako radi vodeni tuš na solarnu energiju UNNI i analizirani problemi trenutnog sklopovlja. Kao rezultat završnog rada napravljeno je sklopovlje i programska podrška koja upravlja radom vodenog tuša na solarnu energiju.

Ključne riječi: Croduino Basic 2 razvojna platforma, MOSFET tranzistor, prekidna rutina, stanje mirovanja.

ABSTRACT

Title: Managing water shower on solar energy

This final paper describes and analyzes the operation of a water shower on solar energy and is based on the Croduino Basic 2 development platform. It describes how the UNNI water shower on solar energy works and analyses the problems of the current hardware. As a result of the final paper, a hardware and software program that manages the operation of water shower on solar energy has been made.

Key words: Croduino Basic 2 development platform, MOSFET tranzistor, interrupts, sleep mode.

ŽIVOTOPIS

Luka Žagar rođen je 17.10.1995. godine u Osijeku. Živi u Osijeku. Završio Osnovnu školu Višnjevac u Višnjevcu, te po završetku upisao Elektrotehničku i prometnu školu Osijek. U srednjoj školi je sudjelovao u projektu SOELA (Solarni električni automobil) gdje je bio dio tima koji je napravio solarni električni automobil. Solarni električni automobil koji je napravio zajedno sa svojim timom osvojio je 3. mjesto u utrci solarnih električnih automobila u Sisku. Sudjelovao je na županijskom natjecanju iz matematike gdje je osvojio 7. mjesto. Nakon srednje škole upisuje Elektrotehnički fakultet Osijek (sada Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek) gdje je imao direktan upis zbog odličnih ocjena u srednjoj školi. Prve godine svog studija učlanjuje se u studentski zbor fakulteta gdje zastupa interese studenata, te aktivno sudjelovao u organiziranju Elektrijade. 2017. godine ponovno izabran kao član studenskog zbora.

PRILOZI

CD s .pdf i .docx verzijom završnog rada.