

Mladi za napredek Maribora 2013

30. srečanje

Inteligentne inštalacije

Raziskovalno področje: Elektrotehnika, elektronika

Inovacijski predlog

0e q | k0E ÖÜCZ Á ÖÜÖŠÖÜ

T ^} q | k0E ÖÜCZ Á ÖÜÖŠÖÜ

¥[| k0E ÖÜCZ Á ÖÜÖŠÖÜ W P ÖŠ P Ö ŠÖÜ U ŠÖÜ ÖÜÖŠÖÜ

Maribor, januar 2013

Vsebina

Povzetek	4
Zahvala	5
Uvod	6
Vsebinski del	7
Cilji	7
Topologija inštalacije	7
Napajanje enosmerne inštalacije	8
Elementi inštalacije	8
CAN komunikacija.....	8
MCP2551 CAN oddajnik (ang. Transciever).....	14
Protokol	16
Napajanje	17
Mikrokontroler	20
Serijska povezava z računalnikom	21
Izdelava vezij.....	22
Elementi.....	23
Tipkalo	23
Stikalni aktuator	25
Vmesnik za programiranje.....	28
Program	31
Cena inštalacije.....	32
Družbena odgovornost.....	32
Viri	34

Kazalo slik

Slika 1: topologija inštalacije (vir: avtor naloge).....	7
Slika 2: OSI model (1).....	9
Slika 3: logična stanja CAN vodila (1).....	10
Slika 4: Priključitev naprav na CAN komunikacijo (1).....	11
Slika 5: standardna terminacija (1).....	11
Slika 6: Split terminacija (1).....	12
Slika 7: Biased split terminacija.....	13
Slika 8: CAN oddajnik (avtor naloge).....	14
Slika 9: Blok shema oddajnika (1).....	14
Slika 10: Izklop oddajnika (1).....	15
Slika 11: Shema napajalnika (avtor naloge).....	18
Slika 12: Hlajenje regulatorja (5).....	19
Slika 13: Spektralna analiza (avtor naloge).....	20
Slika 14: mikrokontroler (6).....	21
Slika 15: Serijska komunikacija z računalnikom (avtor naloge).....	22
Slika 16: Izdelava vezja (avtor naloge).....	22
Slika 17: Delno izdelano vezje (avtor naloge).....	23
Slika 18: Tipkalo (avtor naloge).....	23
Slika 19: Simbol naprave razreda III.....	24
Slika 20: Shema stikalnega aktuatorja (avtor naloge).....	25
Slika 21: Izhodni element (avtor naloge).....	25
Slika 22: Shema izhodnega modula (avtor naloge).....	27
Slika 23: Simbol naprave razreda II.....	27
Slika 24: Shema USB vmesnika (avtor naloge).....	28
Slika 25: Tiskanina USB vmesnika pripravljena za izdelavo (avtor naloge).....	29
Slika 26: Program za nastavljanje USB vmesnika (avtor naloge).....	30
Slika 27: Del programa (avtor naloge).....	31

Povzetek

V želji po avtomatizaciji doma sem se odločil, da izdelam svojo inteligentno inštalacijo. Take inštalacije, kot so na primer KNX (EIB) sicer že obstajajo, vendar so zelo drage, dokumentacija pa zelo nedostopna. Zato so med drugim moji kriteriji tudi nizka cena, enostavno programiranje in dostopna dokumentacija. Krmiljenje poteka decentralizirano, kar omogoča hitro in enostavno dodajanje novih elementov z zelo malo programskimi spremembami, malo stroški in nepotrebnih nadgradenj, ki bi bile potrebne, če bi sistem krmilil centralni krmilnik. Programiranje elementov poteka preko USB vmesnika, ki nato pošilja podatke po vodilu, na katerega so priključeni.

Zahvala

Zahvaljujem se mentorju za podporo pri izdelavi naloge in, da mi je omogočil dostop do nove šolske pridobitve – rezkarja, ki mi je prihranil dosti dela in živcev pri izdelavi dvostranskih tiskanih vezjih.

Uvod

Lansko šolsko leto sem se ukvarjal z inteligentnimi inštalacijami KNX/EIB. V nasprotju s klasičnimi inštalacijami so te veliko bolj fleksibilne, večja je možnost spreminjanja in dodajanja novih funkcij ter elementov. Velika slabost teh inštalacij je njihova cena. Zato sem se odločil, da bom poskušal narediti podoben sistem, ki bo cenejši. Seveda moj izdelek ne bo primerljiv s tem, ampak bo samo predstavljal sistem, ki ga bo mogoče nadgrajevati in mogoče kdaj postane konkurenca tem inštalacijam.

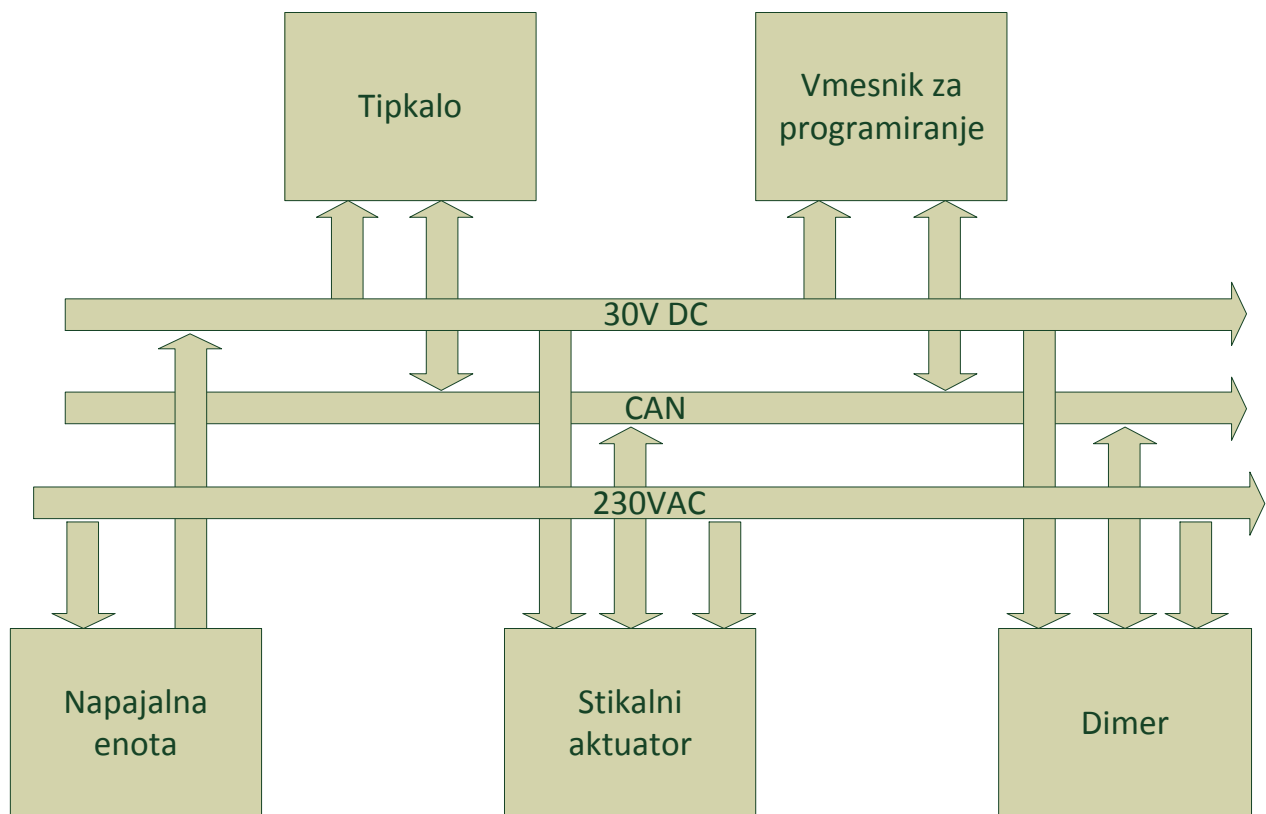
Vsebinski del

Cilji

Kot že povedano v uvodu je moj cilj narediti poceni sistem inteligentne inštalacije. Pri tem bom ceno moje inštalacije primerjal s ceno ostalih proizvajalcev. Moj cilj je tudi narediti program z grafičnim vmesnikom, s katerim bo možno programiranje inštalacij.

Topologija inštalacije

Celoten sistem inštalacije je decentraliziran, to nam omogoča hitro in enostavno dodajanje elementov v primerjavi s centraliziranim. Je sistem senzorjev in aktuatorjev, ki komunicirajo preko CAN vodila.



Slika 1: topologija inštalacije (vir: avtor naloge)

Elementi se napajajo s 30V enosmerne napetosti, nekateri pa potrebujejo še 230V za napajanje relejev in/ali porabnikov. Zato te inštalacije hkrati vsebujejo tudi elemente klasične inštalacije, kot so varovalke, inštalacijski odklopniki in RCD¹ stikalo.

¹ Tokovno zaščitno stikalo (ang. Residual Current Device)

Napajanje enosmerne inštalacije

Ker ima še vsak element napetostni stabilizator, napetostna stabilizacija za napajanje inštalacije ni potrebna. Dovolj je transformator, greatzov mostič in filter za glajenje napetosti. Ker dobimo pri usmerjeni napetosti maksimalno vrednost, nas zanima kakšen mora biti transformator.

$$U_{ef} = \frac{U_{max}}{2} = \frac{30 V}{2} = 21,21 V$$

Torej moramo izbrati transformator, ki ima izhodno napetost okoli 21,21 V.

Elementi inštalacije

Vsi elementi imajo enako napajanje, mikrokontroler in komunikacijo. Zato bom najprej opisal te stvari. Pri opisu vsakega modula pa samo tisto kar je za vsak modul posebnega. Elementi ,ki jih bom naredil so tipkalo, stikalni aktuator in vmesnik za programiranje.

CAN komunikacija

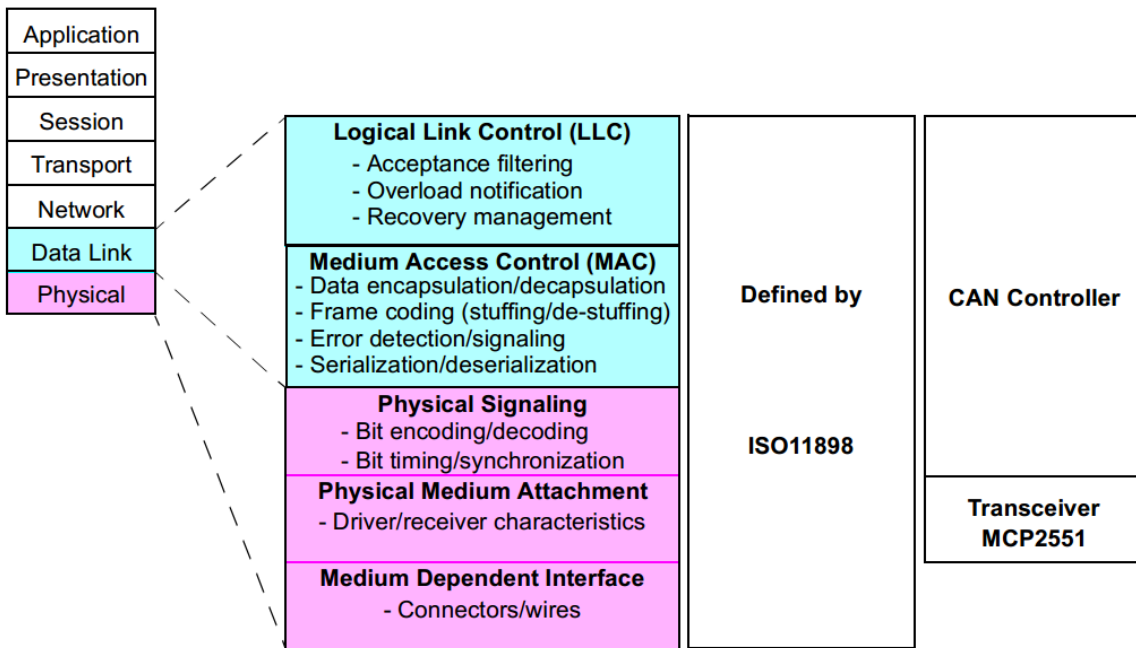
Prvotno je bila razvita za avtomobilsko industrijo, kasneje pa so jo začeli uporabljati tudi v avtomatiki in ostalih mobilnih napravah.

Je serijska povezava, ki podpira hitrosti do 1Mb/s, s 15 bitnim CRC in diferencialnim vodilom pa je odporna na elektromagnetne (EMI²) motnje iz okolice. Z manjšanjem hitrosti prenosa odpornost na elektromagnetne motnje še povečamo, povečamo pa lahko tudi dolžino vodila (do približno 500m pri 125kb/s). Priporočeno je uporabiti sukane vodnike (ang. twisted pair).

Večina komunikacij je predstavljenih z 7-slojnim OSI modelom.

² Electro Magnetic Interference

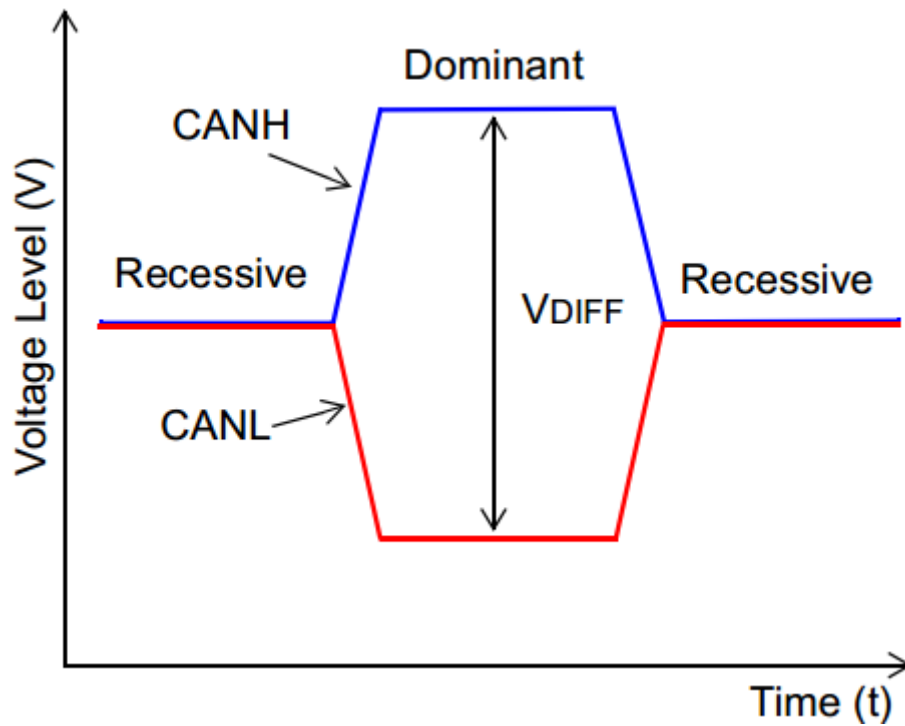
7- Layer OSI



Slika 2: OSI model (1)

Za CAN komunikacijo so definirani samo sloj podatkovne povezave (ang. Data link layer) in del fizičnega sloja (ang. Physical layer). Preostali fizični sloj in vse ostale višje mora definirati načrtovalec sistema. To pomeni, da tudi prenosni medij ni definiran, vendar mora izpolnjevati električno specifikacijo.

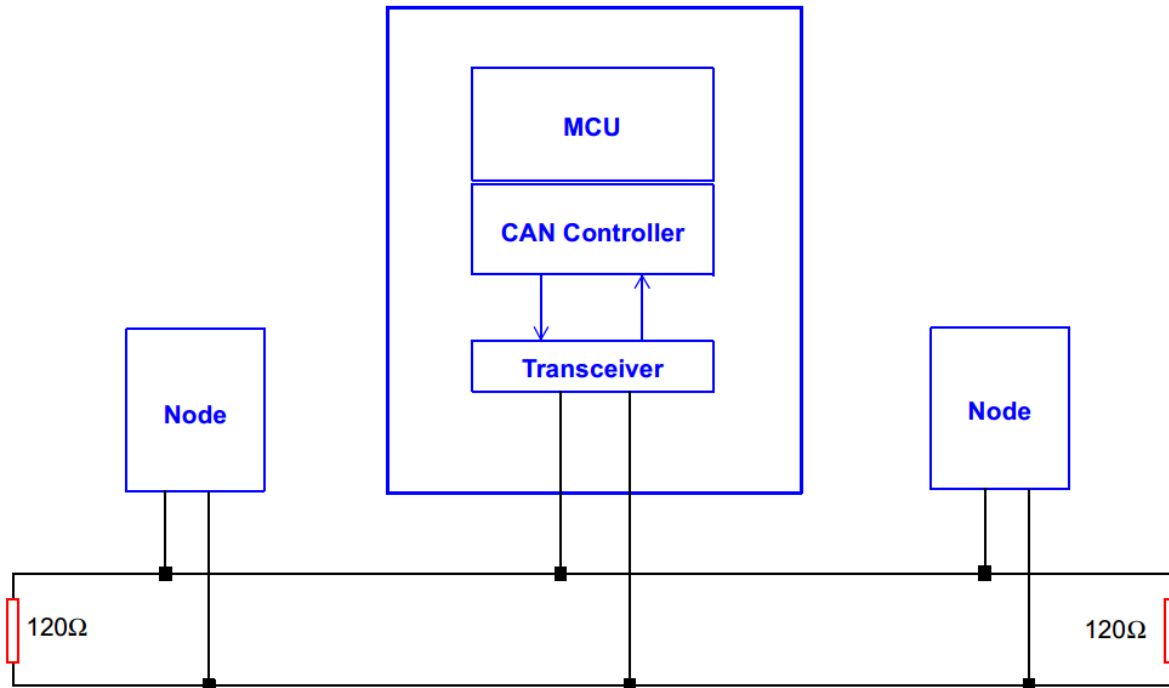
Definirana sta dve logični stanji: "recesivno" (logična '1') in "dominantno" (logična '0')



Slika 3: logična stanja CAN vodila (1)

Dominantni bit prevlada recesivnega v primeru ko začneta komunikacijo dve napravi hkrati.

Vodilo mora biti zaključeno (potrebuje t.i. terminacijo).

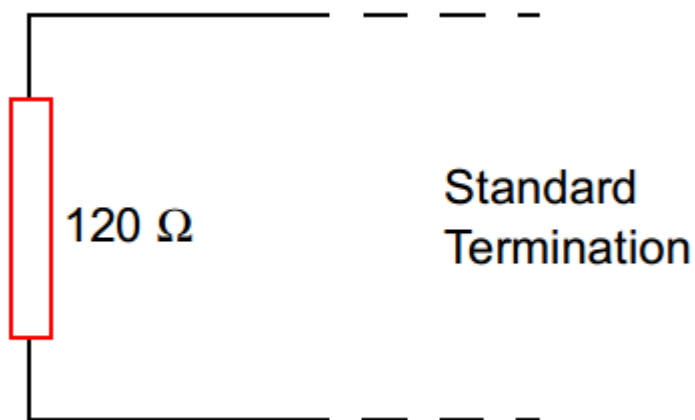


Slika 4: Priklučitev naprav na CAN komunikacijo (1)

Terminacija je potrebna, da zmanjšamo odbijanje signala. Standard predpisuje impedanco 120 Ω. Obstaja več metod terminacije, ki povišajo EMC zmožljivost.

1. Standardna terminacija

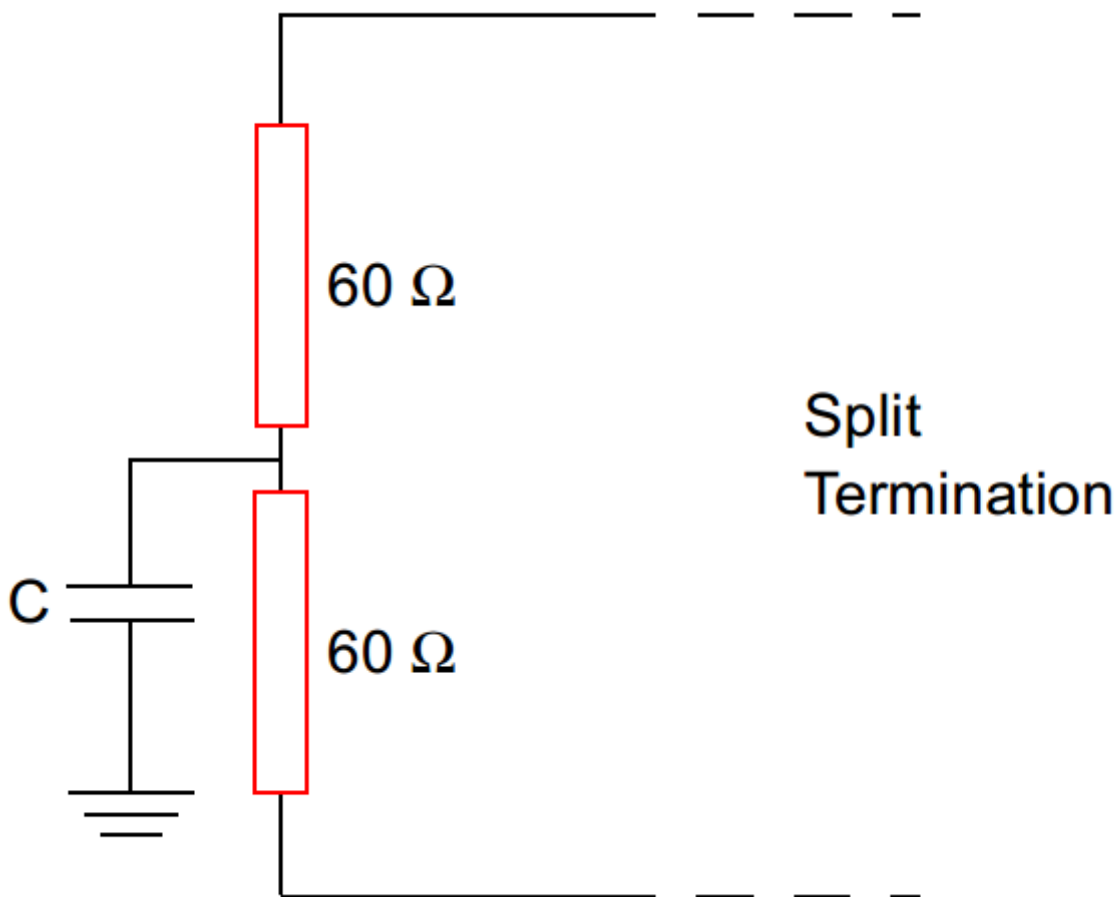
Uporablja 120 Ω upor na obeh koncih. Metoda je sprejemljiva za številne CAN sisteme.



Slika 5: standardna terminacija (1)

2. Split terminacija

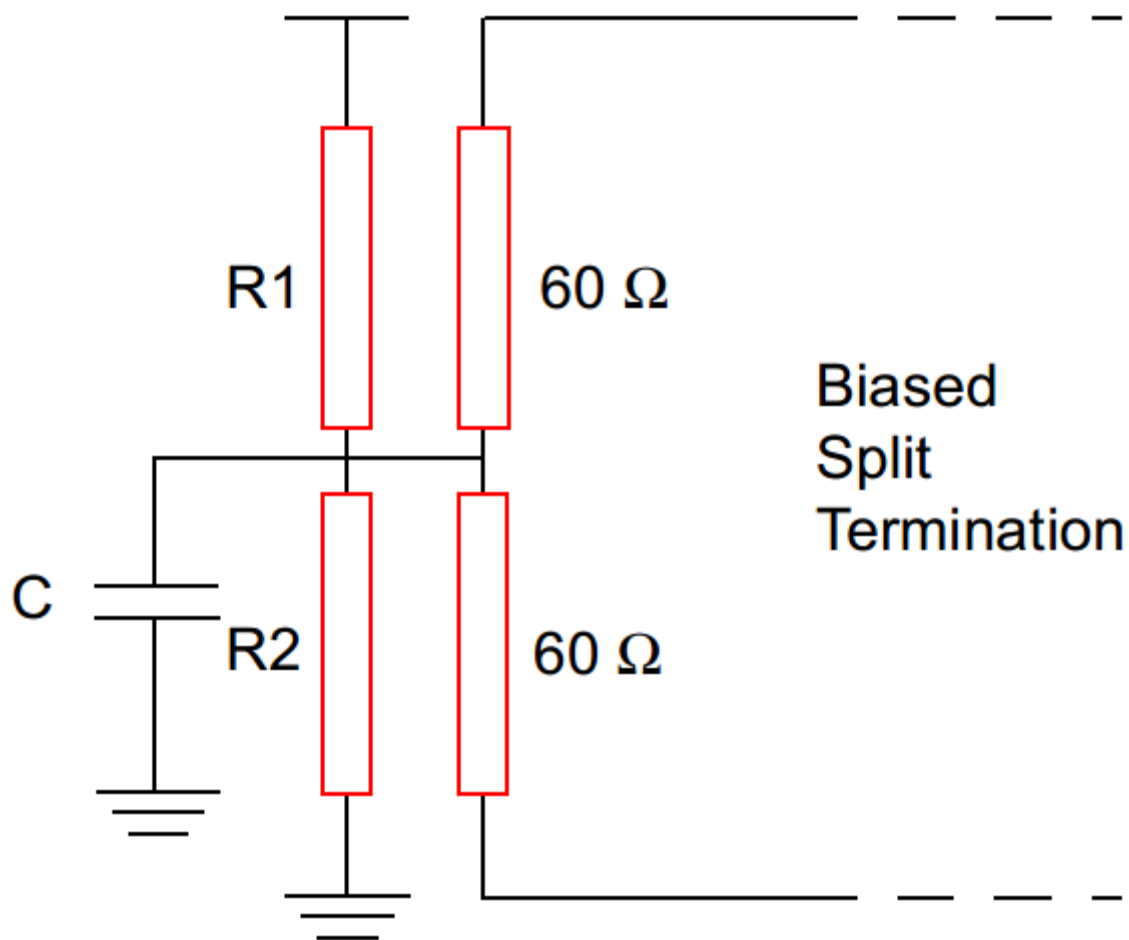
Je vse popularnejša metoda, ker na enostaven način zmanjšamo izsevan signal. Namesto enega upora uporablja dva 60Ω , ki imata na stičišču kondenzator povezan na maso.



Slika 6: Split terminacija (1)

3. Biased split terminacija

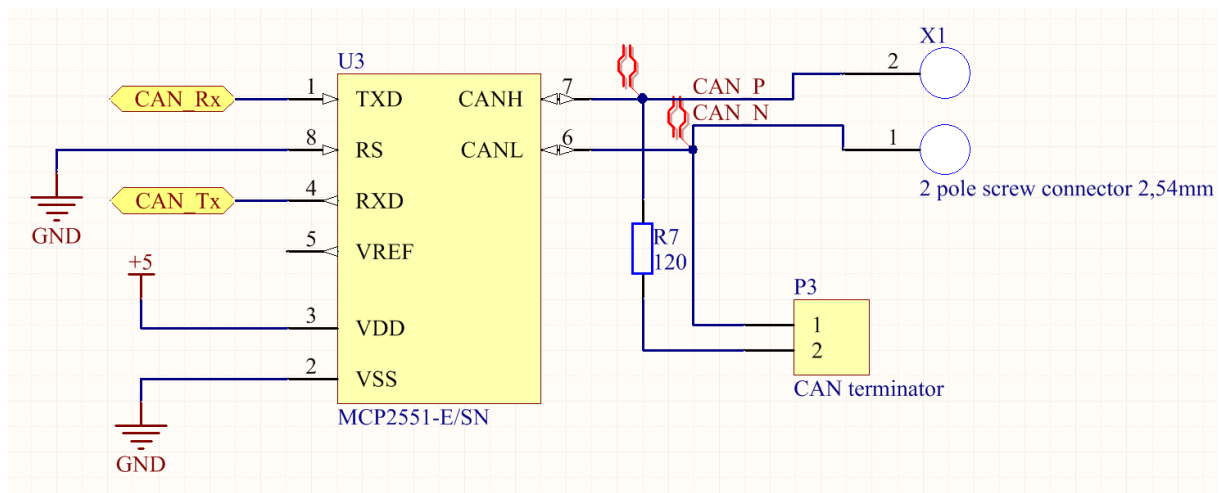
S to metodo vzdržujemo konstantno napetost logične 1, in tako povečamo EMC zmogljivost. Je podobna kot prejšnja metoda s to razliko, da je tukaj dodan napetostni delilnik, da dosežemo polovico napajalne napetosti med obema 60Ω uporoma.



Slika 7: Biased split terminacija

Ker gre za prototip, sem v izdelku uporabil standardno komunikacijo. Bi pa v primeru vgradnje sistema bilo potrebno uporabiti vsaj biased split terminacijo.

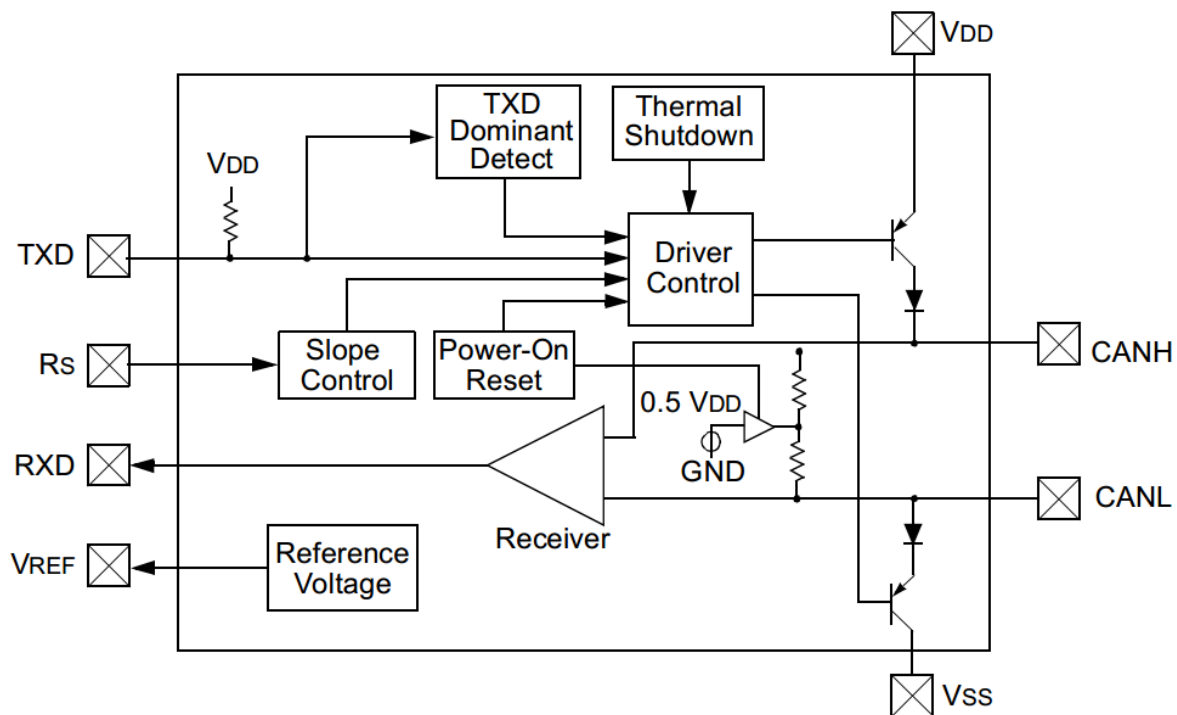
Ker ima sistem možnost dodajanja elementov je potrebno v primeru ko element dodamo na zadnjem elementu terminacijo odklopiti in jo vključiti na novem. V ta namen sem k uporabi vezal jumper s katerim odklopimo terminacijo.



Slika 8: CAN oddajnik (avtor naloge)

MCP2551 CAN oddajnik (ang. Transciever)

Za komunikacijo je poleg vgrajenega modula v mikrokontrolerju potreben tudi oddajnik, ki glede na podatke mikrokontrolerja generira diferencialni signal, ki je potreben za komunikacijo.



Slika 9: Blok shema oddajnika (1)

Implementira fizični sloj in ima tri načine delovanja:

1. High-speed

Ta način delovanja izberemo tako, da povežemo Rs pin na maso. V tem načinu imajo izhodi hitre čase preklopov. V tem načinu dosežemo hitrosti do 1 Mb/s.

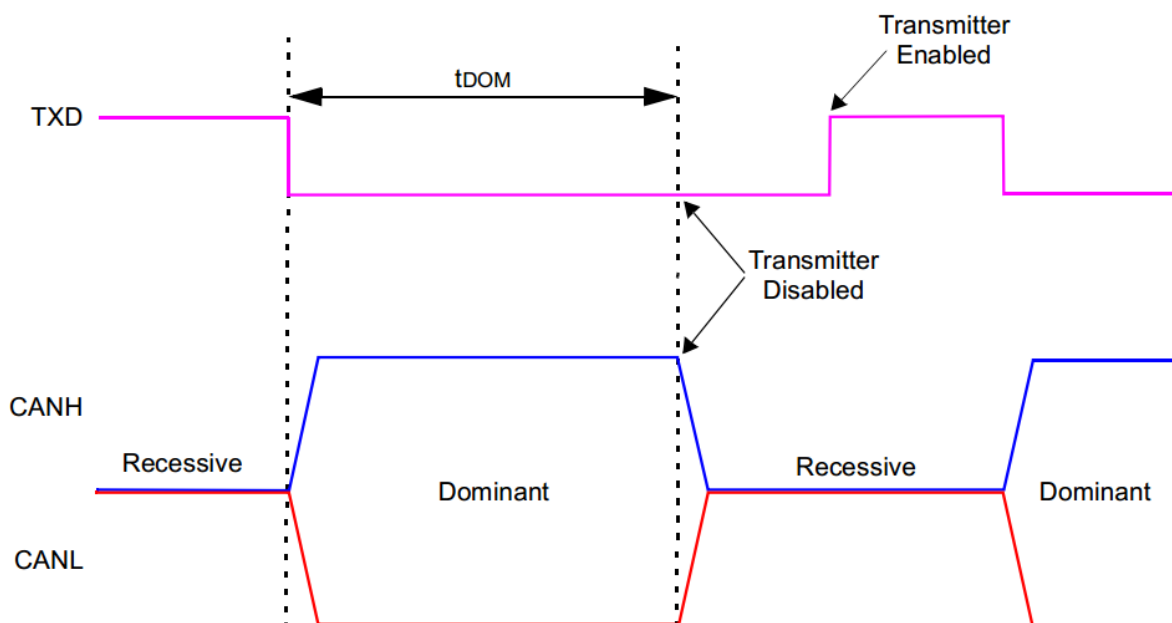
2. Slope control

Ta način delovanja je primeren, če je zahtevana manjša elektromagnetna interferenca (EMI). Izberemo ga tako, da povežemo upor med Rs pinom in maso, pri tem mora biti tok iz tega pina med 10 in 200 μ A. Časi preklopov izhodov so počasnejši, zato so v tem načinu delovanja hitrosti prenosa počasnejše.

3. Mirovanje (standby)

V tem načinu je oddajnik izklopljen sprejemnik deluje z manjšo porabo, komunikacija z mikrokontrolerjem pa je počasnejša. Izberemo ga tako, da Rs pin povežemo na napajalno napetost.

Ima tudi sistem za preprečevanje napak, če je na vhodu oddajnika dominantno stanje traja več kot 1,25 ms, se izhod izklopi. To preprečuje, da bi pokvarjena naprava preprečevala ostalim komunikacijo na vodilu.



Slika 10: Izklop oddajnika (1)

Protokol

Za komunikacijo uporablja CSMA/CD³ protokol. To pomeni, da mora vsaka naprava nekaj časa spremljati vodilo, in če je prosto, lahko začne prenos. Če je priključenih večje število naprav, se lahko zgodi, da začneta prenos dve napravi hkrati. V tem primeru bodo naprave zaznale "trk", prenos pa lahko nadaljuje naprava z višjo prioriteto. Da pa lahko "trke" zaznavajo, mora oddajnik pri oddajanju spremljati tudi dejansko stanje na vodilu. Če je oddano "recesivno" stanje, oddajnik pa zazna "dominantno" – kar pomeni, da oddaja še neka druga naprava z višjo prioriteto, se oddajanje sporočila takoj prekine. Oddajati pa lahko začne, ko zazna, da je vodilo spet prosto. Sporočila z višjo prioriteto ostanejo po "trku" nespremenjena. Prioriteta sporočil je določena z ID-jem sporočila.

Sporočilo vsebuje podatek o prioriteti in vsebini podatkov sporočila. Vse naprave sprejmejo vsa sporočila in se na njih odzovejo – pošljejo potrdilo o prejemu (ang. Acknowledge). Nato se vsaka naprava odloči ali ga zavrže ali ne. Sporočila se ne prenašajo od naprave do naprave glede na naslov. Prednost tega je, da z dodajanjem naprav ni nujno potrebno reprogramirati že obstoječih naprav. Nova naprava bo takoj začela sprejemati sporočila in se glede na njihovo vsebino odločila ali je sporočilo uporabno ali ne.

Še ena uporabna značilnost vgrajena v protokol je, da lahko naprava zahteva informacije od druge naprave – RTR⁴. Zato ne rabi čakati, da dobi potrebne podatke, ampak jih lahko kar zahteva.

V protokolu so definirani štirje tipi sporočil – okvirji (ang. Frame):

1. Podatki

To je najbolj pogost tip sporočila in se uporablja za prenos podatkov med napravami.

2. RTR

Je v bistvu podatkovno sporočilo z RTR bitom. S tem naprava zahteva podatke od druge naprave.

3. Napaka

Oddajo jih naprave, ki zaznajo kakršnokoli napako, ki je definirana v CAN protokolu.

4. Preobremenitev

³ Carrier Sense Multiple Access/Collision Detect

⁴ Remote Transmit Request

Oddajo naprave, ki potrebujejo več časa za obdelavo že sprejetih sporočil.

CAN naprave lahko zaznavajo napake na vodilu in lahko glede na napako menjajo način delovanja. V CAN protokolu je definiranih štiri vrste napak:

1. CRC napaka

Poleg podatkov se v CRC polju pošlje tudi 15-bitna CRC vrednost, ki jo izračuna naprava, ki sporočilo odda. Ostale naprave tudi izračunajo CRC vrednost in jo primerjajo s prejeto, če se vrednosti ne ujemata naprava pošlje CRC napako. Sporočilo je potrebno ponovno oddati.

2. Napaka pri potrjevanju sprejema

Ko naprava odda sporočilo, morajo ostale potrditi sprejem. Če naprava, ki sporočilo oddaja tega ne zazna – kar pomeni, da nobena naprava sporočila ni sprejela, ga je potrebno ponovno oddati.

3. Napaka v obliki okvira

Če naprava zazna dominantno stanje pri enem izmed štirih segmentov: konec okvira, prostor med okviri, ločitvijo med potrjevanjem sprejema ali prostor med CRC poljem, je to zaznano kot napačno oblikovan okvir. Sporočilo mora biti ponovno poslano.

4. Bitna napaka

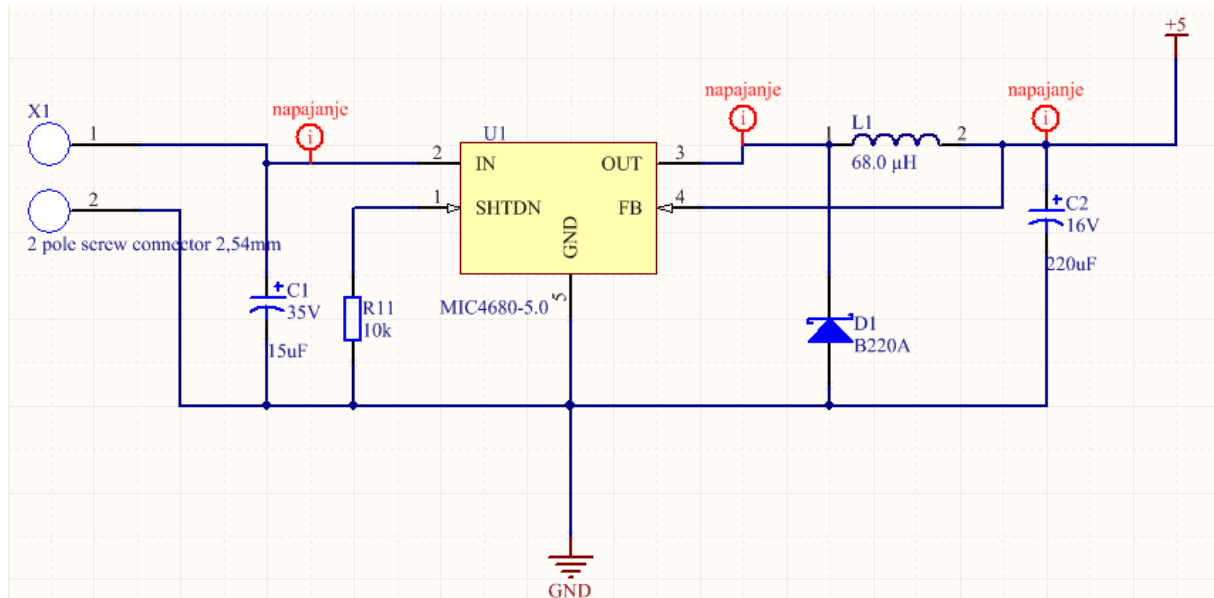
Ta napaka se generira, ko oddajnik pošlje dominantni bit, zazna pa recesivnega ali obratno. V primeru, ko oddajnik pošlje recesivni bit, zazna pa dominantnega, med "pogajanjem" za uporabo vodila ali prostorom za potrjevanje sprejema, se napaka ne zazna, ker oddaja neka druga naprava z višjo prioriteto, ali pa se odvija potrjevanje sprejema. V primeru zaznane napake mora biti sporočilo ponovno poslano.

Napajanje

Vsa elektronika v elementih deluje na napetost 5V, zato potrebuje napetostni regulator.

Ker je cilj čim večji izkoristek elementov, sem na obeh elementih uporabil stikalne regulatorje (MIC4680).

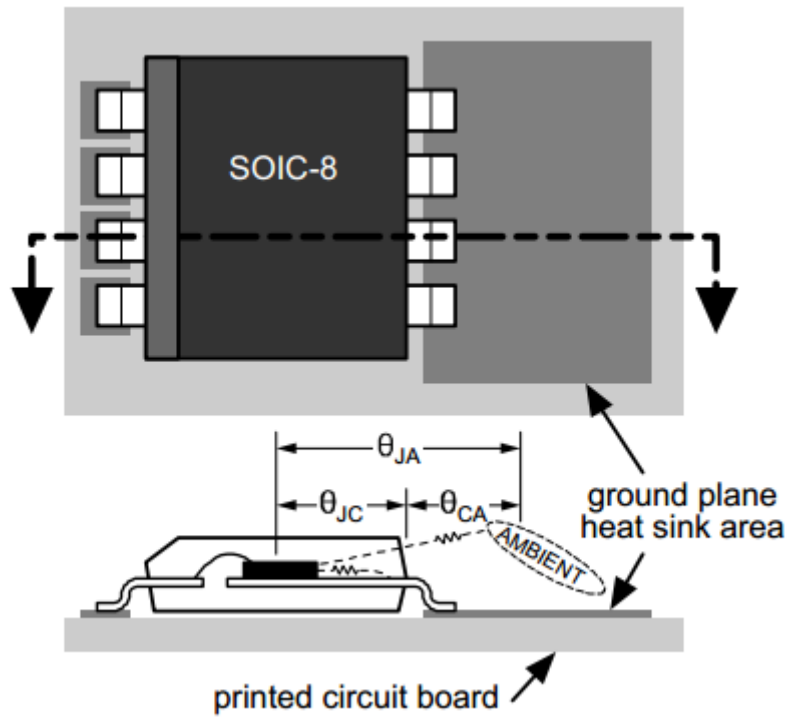
Regulator lahko obremenimo do 1,3A, ima tudi zaščito pred kratkim stikom in termično zaščito. Deluje pri frekvenci 200 kHz, zato ne potrebuje večjih zunanji komponent. Na vhod mu pa lahko priključimo napetost 4 do 34 V.



Slika 11: Shema napajalnika (avtor naloge)

Potrebno bi bilo dodati še zaščito pred napačno priključitvijo – dioda in kratkim stikom – varovalka.

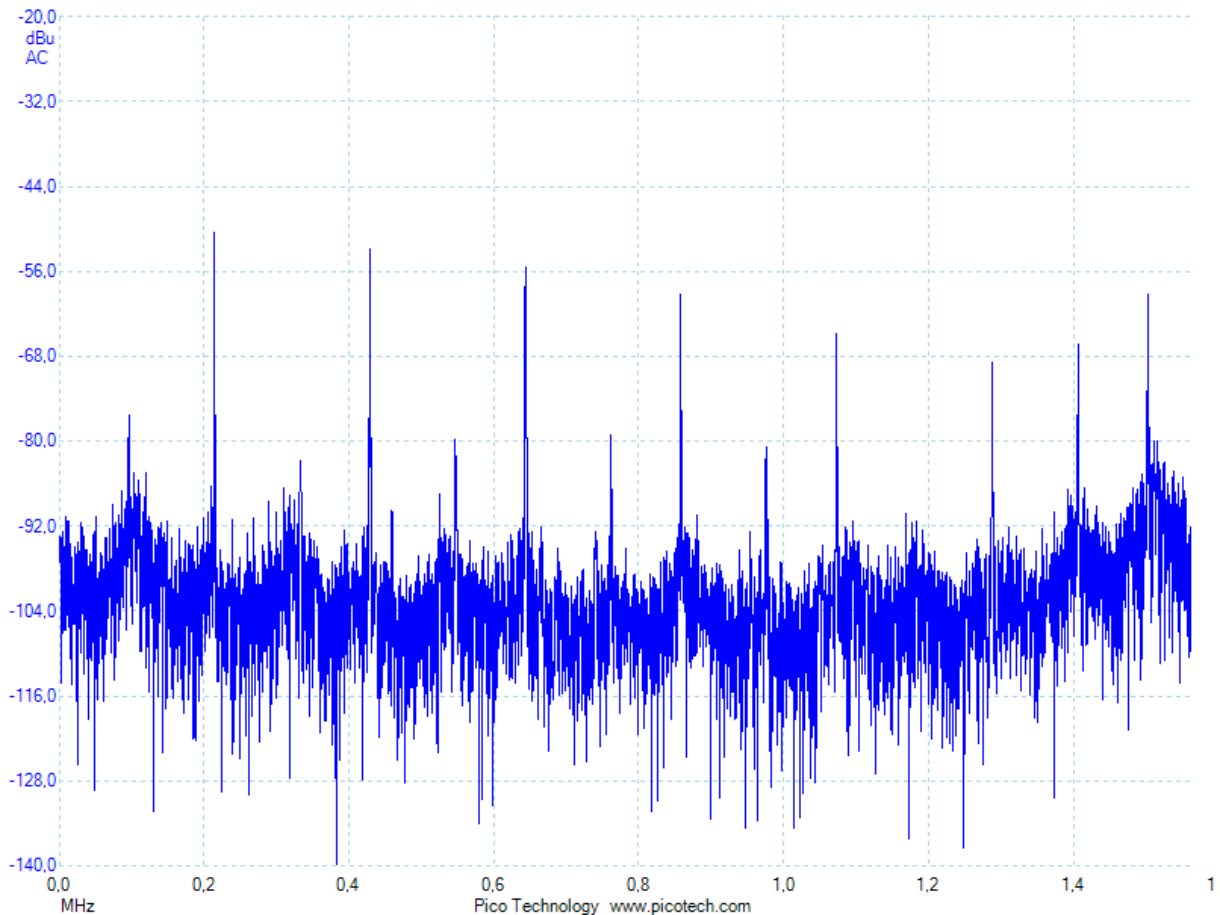
Za delovanje potrebuje samo 4 dodatne komponente, zaradi česar vezje ni veliko večje, kot, če bi uporabil linearni regulator. Ima do 90% izkoristek, ki se z večjo vhodno napetostjo manjša. Hladi pa se preko štirih GND pinov.



Slika 12: Hlajenje regulatorja (5)

Zaradi tega je lahko v majhnem SOIC⁵-8 ohišju.

⁵ Small-Outline Integrated Circuit



Slika 13: Spektralna analiza (avtor naloge)

Iz slike vidimo, najvišjo amplitudo pri 200 kHz, ki je frekvenca delovanja regulatorja, in njene višje harmonske komponente.

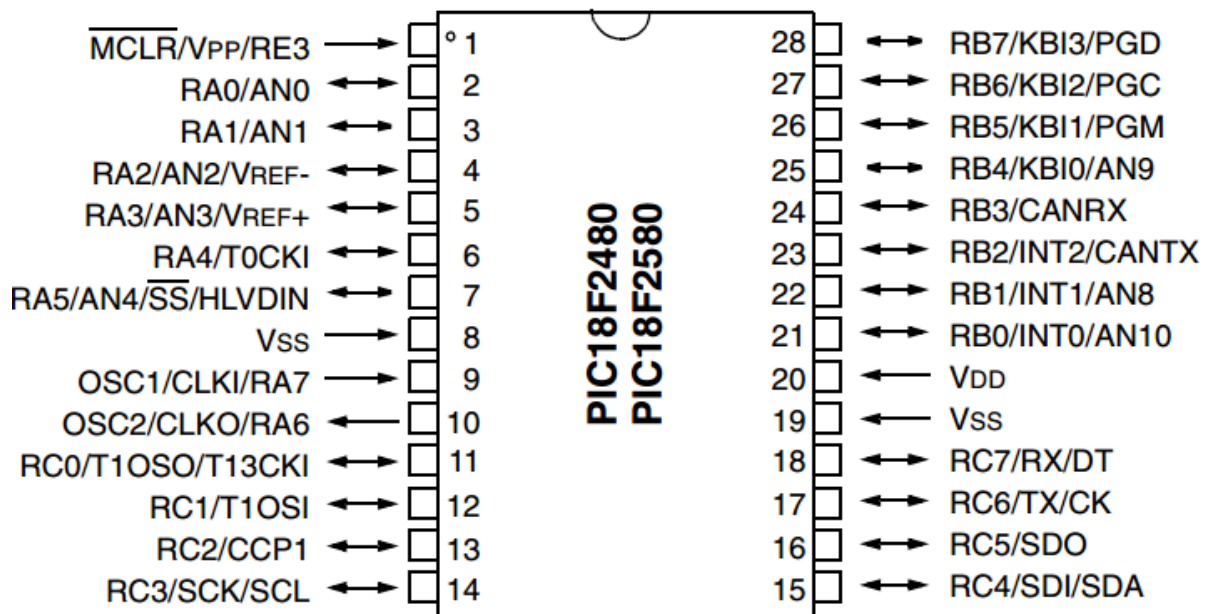
V nalogi sem uporabil za napajanje in komunikacijo med elementi UTP kabel, ki ima upornost 0,188 Ω /m. Ker je minimalna napetost regulatorjev 4 V, napajalna napetost pa kot že povedano 30 V, imamo lahko na enosmerni inštalaciji padec napetosti 26 V. Pri 500 m dolgem vodilu je lahko torej maksimalen tok:

$$I = \frac{U}{R_0 * l} = \frac{26 \text{ Vm}}{0,188 \Omega * 500 \text{ m}} = 276,6 \text{ mA}$$

Seveda bi za realizacijo bilo potrebno uporabiti vodnike z večjim presekom, po katerih bi lahko tekel večji tok. Maksimalni dopustni tok pa bi izračunali po isti enačbi.

Mikrokontroler

Za krmiljenje vseh elementov sem uporabil PIC18F2580.



Slika 14: mikrokontroler (6)

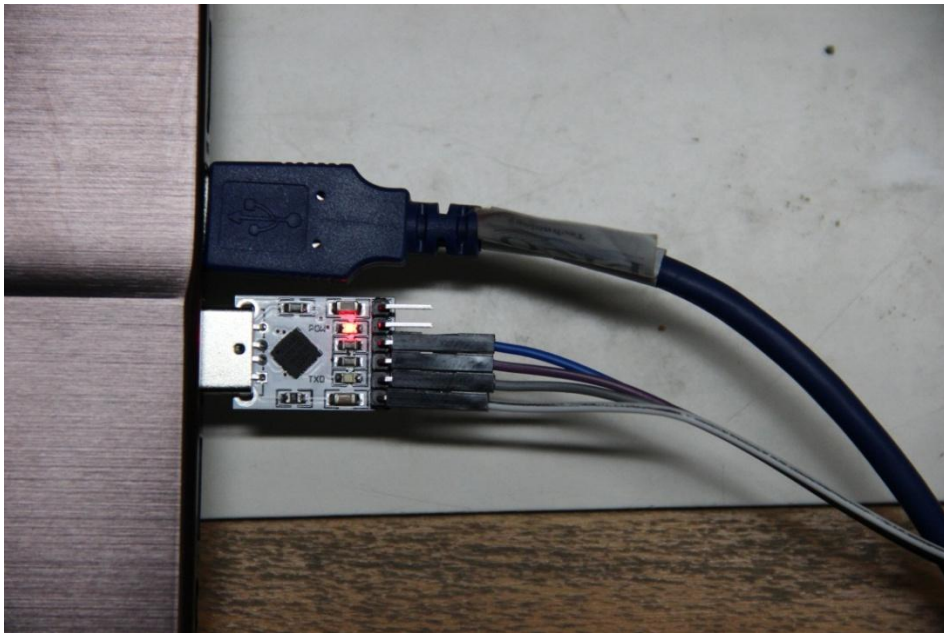
Je v majhnem SOIC-28 ohišju. Zaradi možnosti uporabe internega oscilatorja in PLL množilnika frekvence, lahko dobimo hitrost delovanja 32 MHz brez dodatnega zunanega oscilatorja, zaradi česar sem lahko še dodatno zmanjšal velikost vezja.

V končnem izdelku bi uporabil enak mikrokontroler v QFN⁶-28 ohišju, ki je še manjše s tem pa bi še dodatno zmanjšal velikost vezja.

Serijska povezava z računalnikom

Elementi imajo še dodano možnost povezave z računalnikom preko UART protokola. Uporabljam jo za pošiljanje informacij o stanju programa računalniku, s tem je odkrivanje in odpravljanje napak dosti lažje. Ker pa nobeden novejši računalnik nima več serijskega priključka, tisti, ki ga pa imajo, pa ne ustreza napetostnim nivojem, sem moral uporabiti pretvornik iz USB v UART protokol.

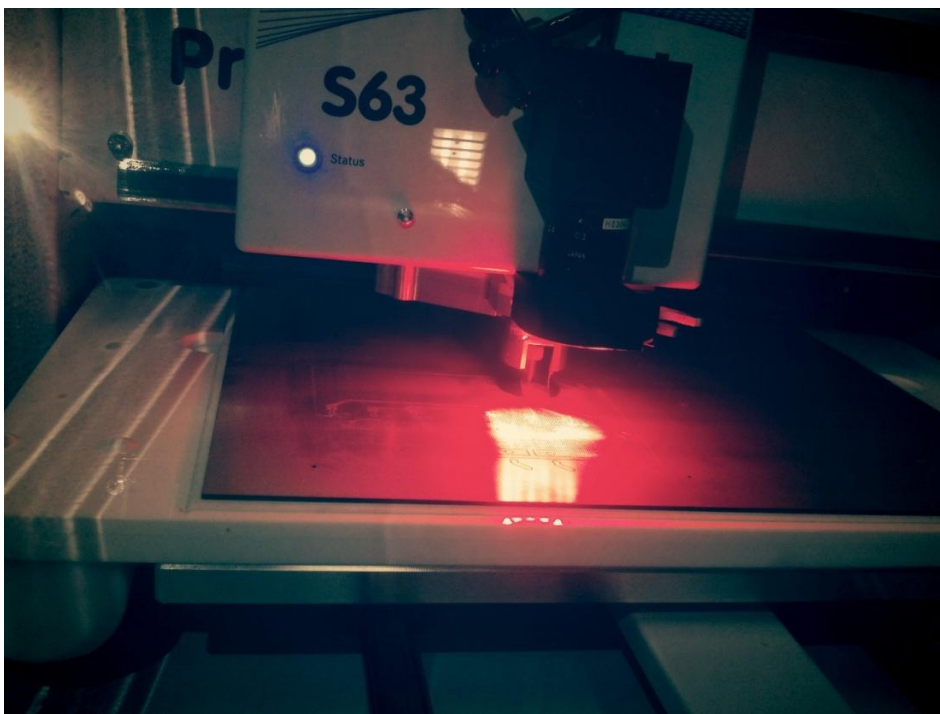
⁶ Quad Flat No-lead



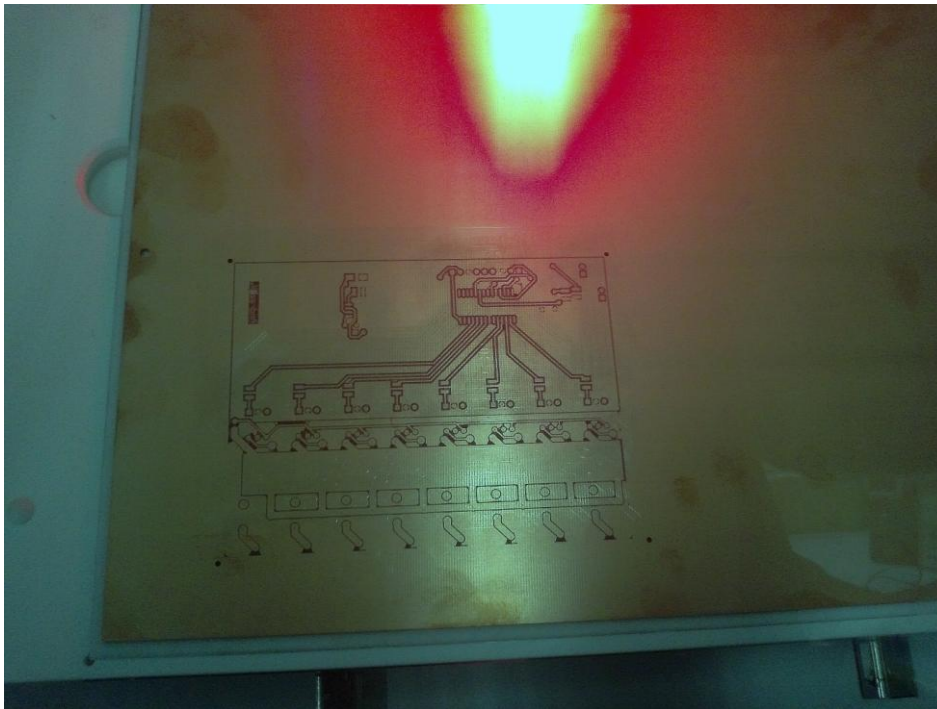
Slika 15: Serijska komunikacija z računalnikom (avtor naloge)

Izdelava vezij

Za izdelavo vezij sem uporabil novo šolsko pridobitev – rezkar.



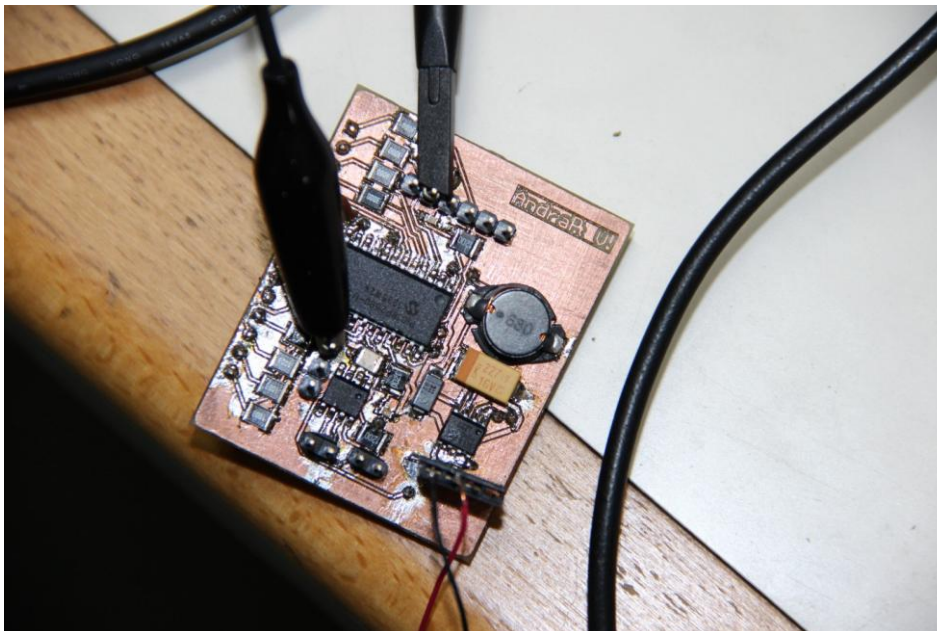
Slika 16: Izdelava vezja (avtor naloge)



Slika 17: Delno izdelano vezje (avtor naloge)

Elementi

Tipkalo



Slika 18: Tipkalo (avtor naloge)

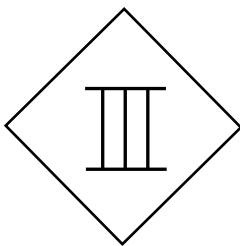
Je senzor. Zanj je uporabljena 3x4 matrična tipkovnica, ki ima tudi enako LED matriko. Meri lahko še temperaturo prostora – uporabljen je LM35 analogni termometer.

S tipkovnico lahko vklopimo ali izklopimo porabnik, razsvetljavo, dvignemo ali spustimo žaluzije, zatemnimo prostor (dimanje), odvisno od tega za kakšno funkcijo je določena tipka sprogramirana, če je.

Led diode, ki so poleg tipk lahko signalizirajo ali je določen porabnik ali razsvetljava vklopljena. Funkcije signalnih led lahko po želji sprogramiramo.

Podatke o temperaturi lahko pošilja npr. nekemu elementu z zaslonom, v bazo podatkov, lahko pa tudi glede na temperaturo vključi npr. ogrevanje.

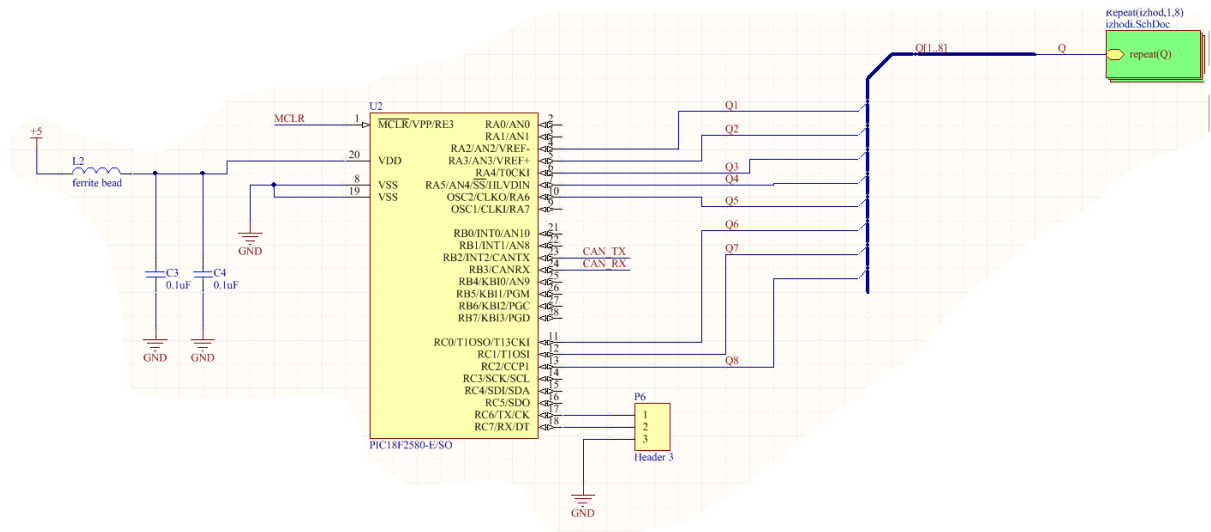
Ta element je naprava razreda III in se priključi na SELV⁷ sistem – enosmerna inštalacija.



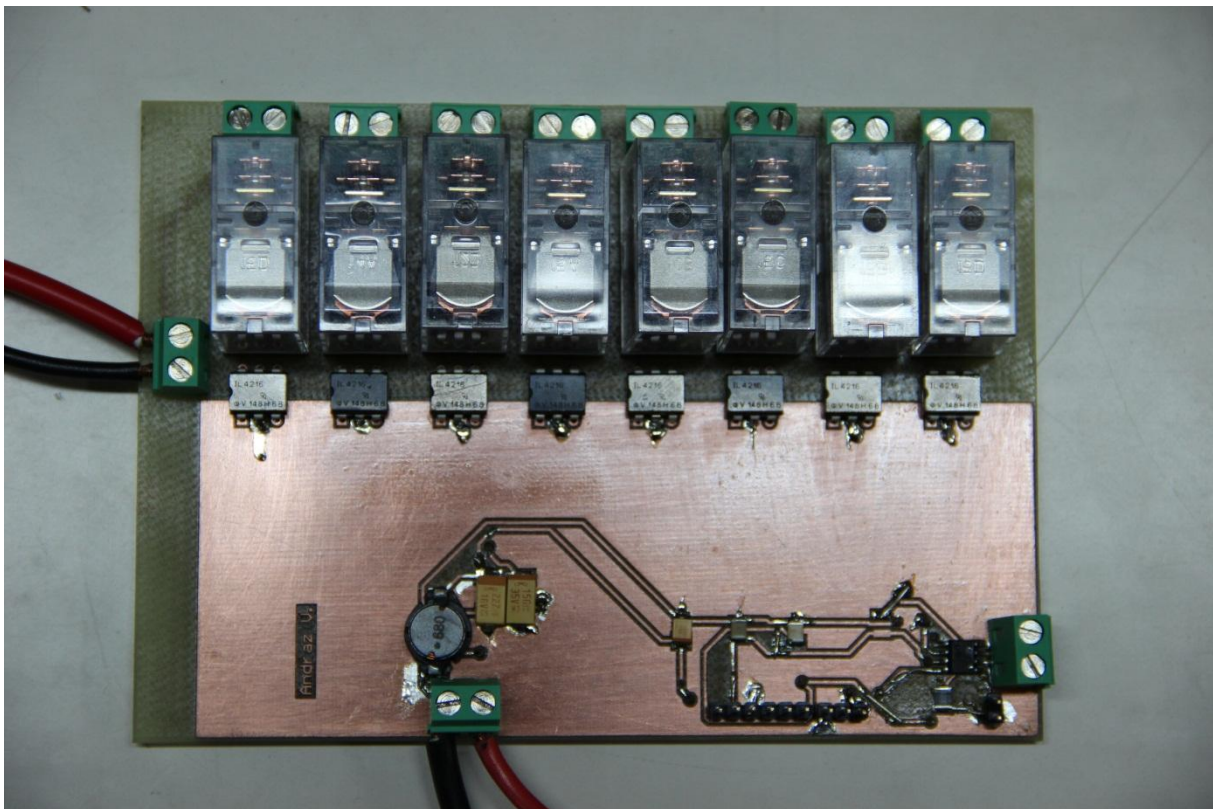
Slika 19: Simbol
naprave razreda III

⁷ Safety Extra Low Voltage

Stikalni aktuator



Slika 20: Shema stikalnega aktuatorja (avtor naloge)



Slika 21: Izhodni element (avtor naloge)

Izhodni modul

Izhod sestavlja 8 relejev (G2R-1), ki jih lahko obremenimo s tokom 10 A, s katerim lahko pri omrežni napetosti krmilimo porabnik z navidezno močjo 3200 VA. Največja dovoljena

delovna moč enofaznih porabnikov je 3 kW, zato nas zanima kolikšen je lahko najmanjši faktor delavnosti porabnika.

$$S = 3200 \text{ VA}$$

$$P = 3000 \text{ W}$$

Ker gre pri večini porabnikov za ohmsko – induktivno breme, računamo Q_L .

$$Q_L = \sqrt{S^2 - P^2} = \sqrt{(3200 \text{ VA})^2 - (3000 \text{ W})^2}$$

Kar znaša:

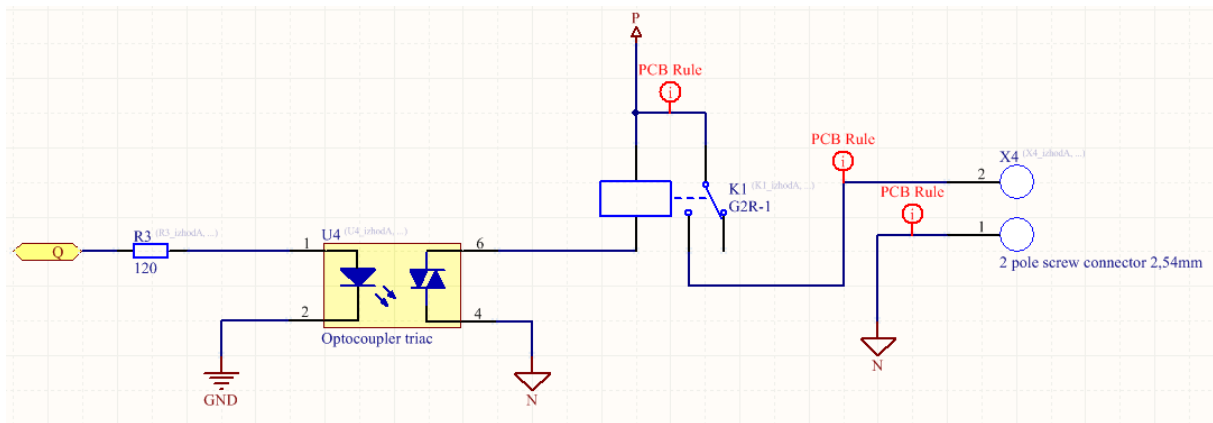
$$Q_L = 1113,55 \text{ VAR}$$

Faktor delavnosti pa je:

$$\cos \varphi = \frac{P}{S} = \frac{3000 \text{ W}}{3200 \text{ VA}} = 0,94$$

Vidimo, da je lahko porabnik, ki se približuje moči 3 kW lahko skoraj samo čisto ohmski. Če bi želeli priključiti porabnike z manjšim faktorjem delavnosti kot npr. motorje, bi bilo potrebno povečati moč izhodnih relejev ali pa poskrbeti za kompenzacijo induktivne jalove energije z vzporedno vezanim kondenzatorjem.

Rele pri napetosti 5V in s tuljavo moči 500mW potrebuje 100mA toka, kar je pri osmih relejih 800mA, za en aktuator. Pri večjem številu aktuatorjev predstavlja tak tok preveliko obremenitev za enosmerno inštalacijo. Zato sem tu uporabil releje s proženjem z omrežno napetostjo.

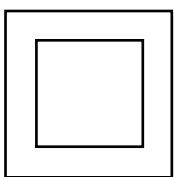


Slika 22: Shema izhodnega modula (avtor naloge)

Iz sheme vidimo, da faznega in nevtralnega vodnika ne smemo zamenjati, ker bi sicer pri večji obremenitvi izhoda uničili optosklopnik, ki vzdrži tok 300 mA. Da pa to preprečimo, bi bilo potrebno dodati varovalke.

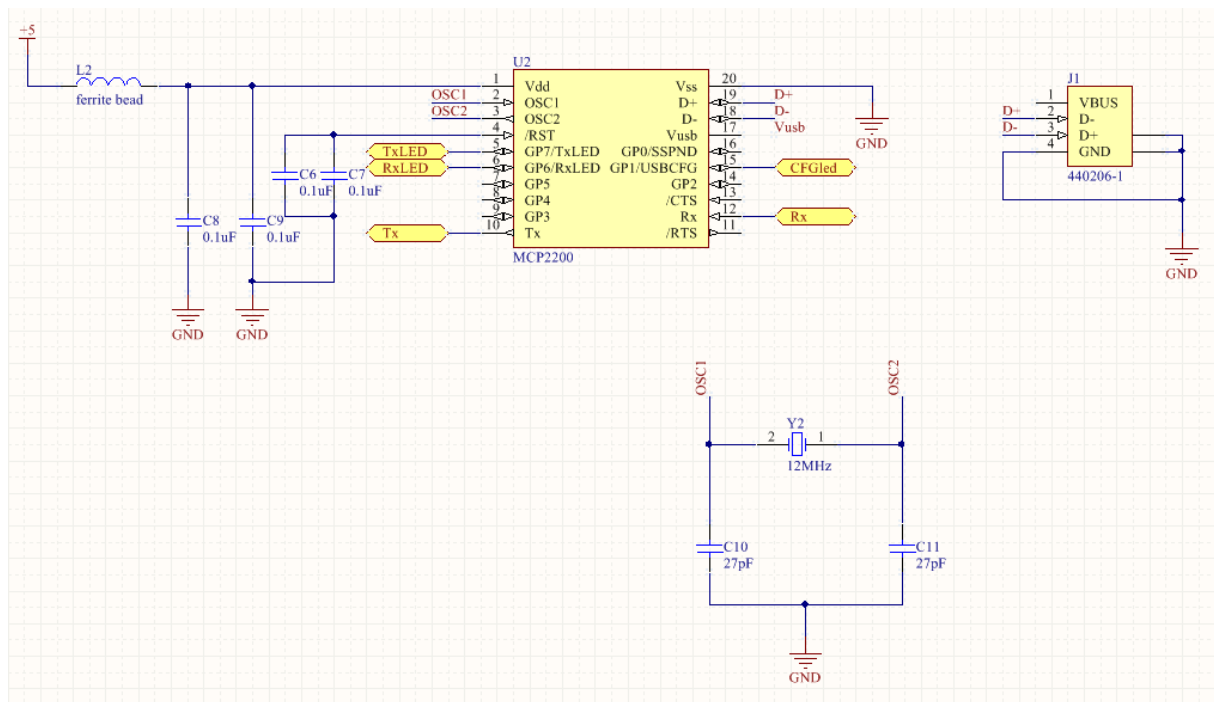
V tem elementu bi lahko obstoječi mikrokontroler zamenjal z MCP25020, ki implementira preprosto CAN napravo brez potrebe po mikrokontrolerju. S tem pa je tudi strošek materiala manjši.

Element se vgradi v plastično ohišje, ki se lahko pritrdi na DIN letev. Zaradi prisotne omrežne napetosti je ta element razreda II.

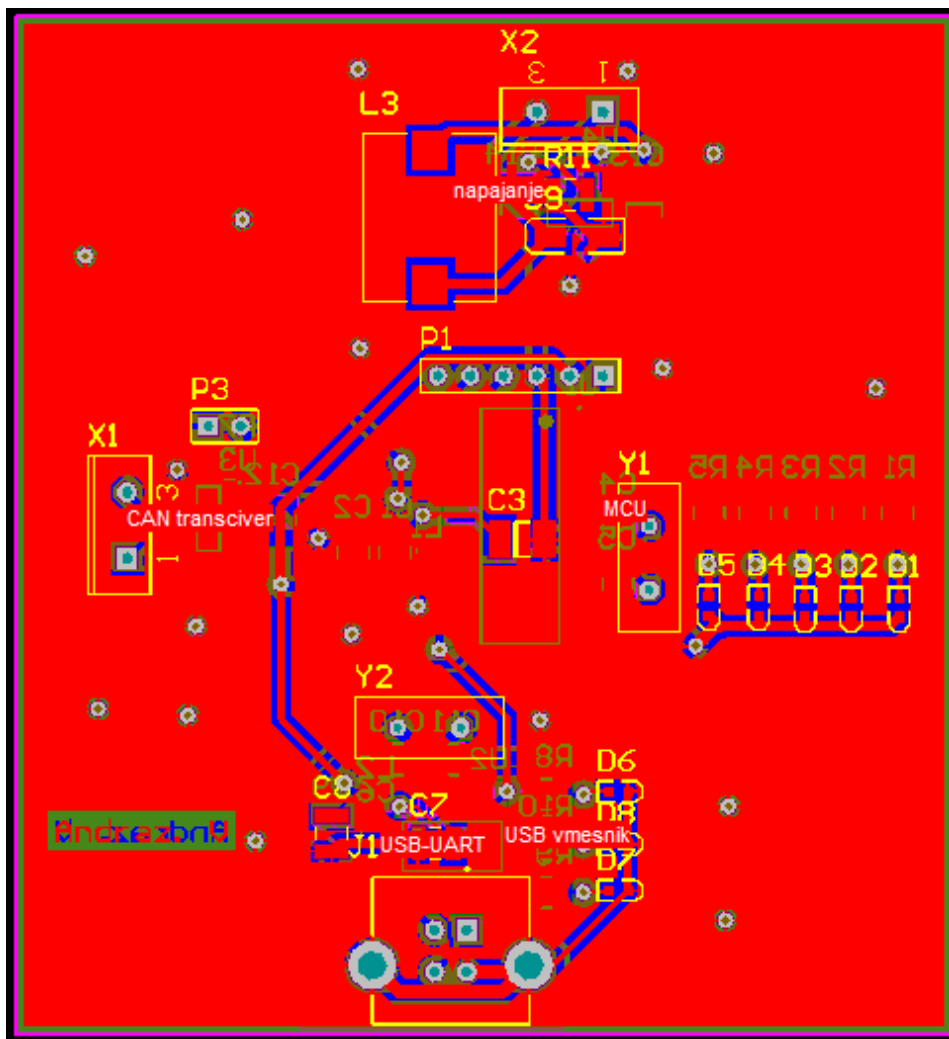


Slika 23: Simbol naprave razreda II

Vmesnik za programiranje



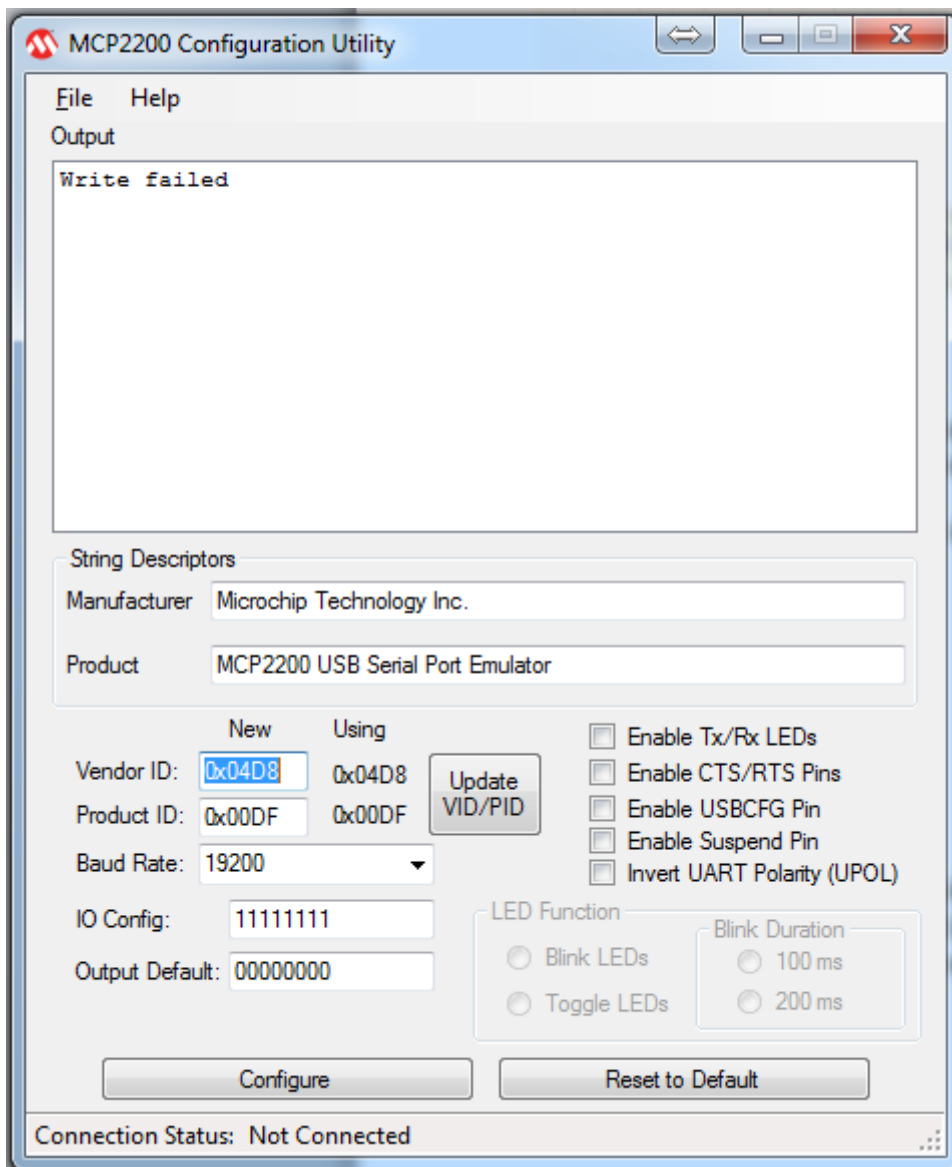
Slika 24: Shema USB vmesnika (avtor naloge)



Slika 25: Tiskanina USB vmesnika pripravljena za izdelavo (avtor naloge)

Na računalnik se priključi z USB priključkom. Ker uporabljen mikrokontroler nima vgrajene USB komunikacije, sem uporabil MCP2200, ki pretvori USB protokol v UART, ki ga mikrokontroler ima.

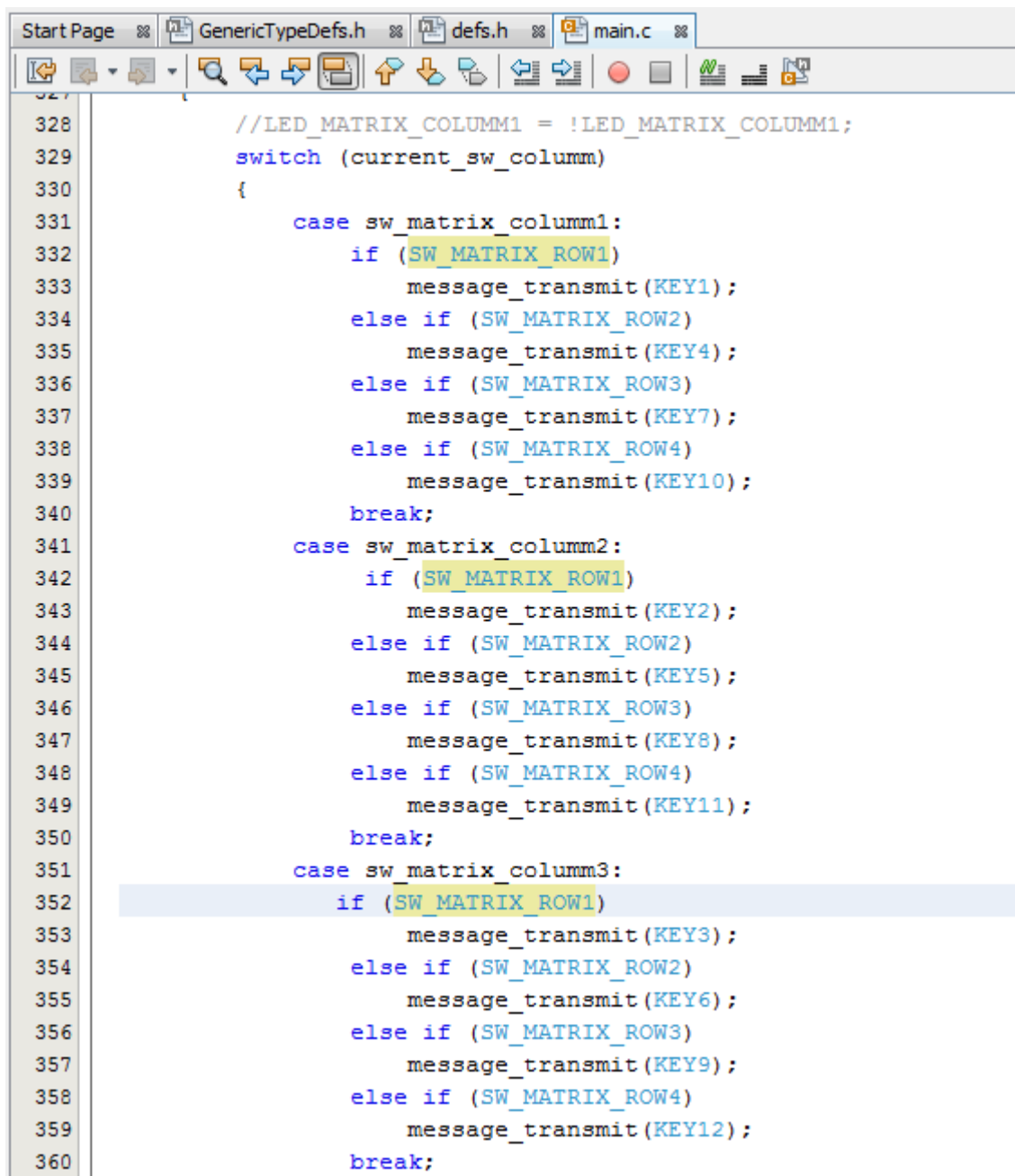
MCP2200 implementira HID napravo, ima še vhodno izhodne pine, katere lahko kontroliramo preko USB komunikacije. S programskim orodjem lahko nastavimo "Descriptorje", hitrost prenosa in vhodno-izhodne pine.



Slika 26: Program za nastavljanje USB vmesnika (avtor naloge)

Program

Programiral sem v MPLABX IDE v C jeziku. Za prevajanje sem uporabljal C18 prevejalnik.



```
327
328     //LED_MATRIX_COLUMN1 = !LED_MATRIX_COLUMN1;
329     switch (current_sw_column)
330     {
331         case sw_matrix_column1:
332             if (SW_MATRIX_ROW1)
333                 message_transmit(KEY1);
334             else if (SW_MATRIX_ROW2)
335                 message_transmit(KEY4);
336             else if (SW_MATRIX_ROW3)
337                 message_transmit(KEY7);
338             else if (SW_MATRIX_ROW4)
339                 message_transmit(KEY10);
340             break;
341         case sw_matrix_column2:
342             if (SW_MATRIX_ROW1)
343                 message_transmit(KEY2);
344             else if (SW_MATRIX_ROW2)
345                 message_transmit(KEY5);
346             else if (SW_MATRIX_ROW3)
347                 message_transmit(KEY8);
348             else if (SW_MATRIX_ROW4)
349                 message_transmit(KEY11);
350             break;
351         case sw_matrix_column3:
352             if (SW_MATRIX_ROW1)
353                 message_transmit(KEY3);
354             else if (SW_MATRIX_ROW2)
355                 message_transmit(KEY6);
356             else if (SW_MATRIX_ROW3)
357                 message_transmit(KEY9);
358             else if (SW_MATRIX_ROW4)
359                 message_transmit(KEY12);
360             break;
```

Slika 27: Del programa (avtor naloge)

Cena inštalacije

Cena je odvisna predvsem od velikosti inštalacije. Kot že povedano je vedno možno dodajanje novih elementov, s tem je pa cena seveda višja.

Cena materiala za tipkalo je okoli 50 €, stikalnega aktuatorja okoli 97 €, vmesnika za programiranje pa okoli 45 €. Seveda moramo pri tem upoštevati še stroške izdelave in vgradnjo elementov.

Družbena odgovornost

Po pregledu kriterijev za družbeno odgovornost sem ugotovil, da je naloga družbeno odgovorna.

Sklep

Uspel sem narediti elemente, ki jih še nisem uspel povezati med sabo, pa tudi program še ni dokončan. Največji problem mi je predstavljala CAN komunikacija saj sem jo v tej nalogi komaj spoznal. Cena je nižja kot pri ostalih proizvajalcih, me je pa vseeno presenetila njena višina, ki ni toliko nižja kot sem pričakoval. Ta cilj je zato samo delno dosežen. Programa s katerim bi lahko preko USB vmesnika programirali inštalacije, mi še zaradi ne dovolj dobrega znanja programiranja tudi ni uspelo izdelati.

Veliko dela imam še z razvojem programov za mikrokontrolerje, ker moje znanje programiranja še ni tako zelo dobro. Možnosti za nadaljnji razvoj je veliko, saj lahko dodamo veliko novih elementov.

Viri

1. <http://ww1.microchip.com/downloads/en/AppNotes/00713a.pdf>
1. http://en.wikipedia.org/wiki/CAN_bus
2. <http://esd.cs.ucr.edu/webres/can20.pdf>
3. <http://ww1.microchip.com/downloads/en/AppNotes/00228a.pdf>
4. http://www.produktinfo.conrad.com/datenblaetter/700000-724999/708988-da-01-en-TASTATUR_3X4_BELEUCHTET_AC3560ILL.pdf
5. <http://www.micrel.com/PDF/mic4680.pdf>
6. <http://ww1.microchip.com/downloads/en/DeviceDoc/39637b.pdf>