



Št. naloge: 01660/2010

Datum: 05.04.2010

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **TADEJ SLEMC**


Naslov: **VISOKOHITROSTNA FOTÓGRAFIJA  
HIGH SPEED PHOTOGRAPHY**

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

Pri fotografiranju enkratnih hitrih dogodkov, za katere ne vemo kdaj se bodo pričeli, človek ne more pravočasno prožiti fotoaparata. Potrebna je posebna naprava za visokohitrostno fotografijo, ki s pomočjo tipal za zaznavanje dogodkov izbere pravi trenutek proženja. Na osnovi pregleda obstoječih rešitev razvijte in izdelajte napravo, ki omogoča ločeno proženje fotoaparata in bliskavice. Omogoča naj uporabo različnih tipal in naj ima vgrajeno možnost nastavljanja zamika med zaznavo dogodka in proženjem. Uporabite standardne elektronske komponente in izberite ustrezní mikrokrmilnik, za katerega izdelajte potrebno programsko opremo. Napravo preizkusite, opišite morebitne pomanjkljivosti in podajte možne nadaljnje smeri razvoja.

Mentor:



prof. dr. Dušan Kodek



Dekan:



prof. dr. Franc Solina

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Tadej Slemc

## **Visokohitrostna fotografija**

DIPLOMSKO DELO  
NA UNIVERZITETNEM ŠTUDIJU

Mentor: prof. dr. Dušan Kodek

Ljubljana, 2010

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.

*Besedilo je oblikovano z urejevalnikom besedil  $\LaTeX$ .*

Namesto te strani **vstavite** original izdane teme diplomskega dela s podpisom mentorja in dekana ter žigom fakultete, ki ga diplomant dvigne v študentskem referatu, preden odda izdelek v vezavo!



# IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani/-a Tadej Slemc,

z vpisno številko 63000271,

sem avtor diplomskega dela z naslovom:

Visokohitrostna fotografija

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom prof. dr. Dušana Kodeka
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne 28.06.2010

Tadej Slemc \_\_\_\_\_



# Zahvala

Zahvaljujem se mentorju prof. dr. Dušanu Kodeku za nasvete in vodenje pri izdelavi diplomske naloge ter dr. Andreju Štrancarju za pomoč pri izdelavi tiskanega vezja. Zahvaljujem se tudi staršem za podporo v času študija.



# Kazalo

<b>Povzetek</b>	<b>1</b>
<b>Abstract</b>	<b>3</b>
<b>1 Uvod</b>	<b>5</b>
<b>2 Pristop k problemu</b>	<b>7</b>
2.1 Način fotografiranja . . . . .	7
2.1.1 Način s proženjem fotoaparata . . . . .	7
2.1.2 Način s proženjem bliskavice . . . . .	9
2.2 Proženje ob dogodku . . . . .	10
<b>3 Strojni del</b>	<b>13</b>
3.1 Osnovna plošča . . . . .	14
3.1.1 Napajalni del . . . . .	14
3.1.2 Predojačevalnik signala mikrofonskega vhoda . . . . .	16
3.1.3 Povezovalni del . . . . .	16
3.1.4 Izhodni del . . . . .	18
3.1.5 Kontrolni del . . . . .	20
3.2 Tipke za nadzor ter prikazovalnik LCD . . . . .	21
3.3 Zunanja tipala . . . . .	23
3.3.1 Analogno svetlobno tipalo . . . . .	23
<b>4 Programski del</b>	<b>25</b>
4.1 Programiranje mikrokrmilnika . . . . .	25
4.1.1 Bootloader . . . . .	26
4.1.2 Nalaganje programske kode . . . . .	27
4.2 Uporabljene knjižnice . . . . .	27
4.2.1 Knjižnica za delo z prikazovalnikom LCD . . . . .	27
4.2.2 Knjižnica za delo s pomnilnikom EEPROM . . . . .	28

## KAZALO

4.3	Opisi procedur . . . . .	28
4.3.1	Procedura read_serial . . . . .	28
4.3.2	Procedura lcd_print_credits . . . . .	28
4.3.3	Procedura spremeni_vrednost_gumb . . . . .	28
4.3.4	Procedura menu . . . . .	29
4.3.5	Procedura f_intervalometer . . . . .	29
4.3.6	Procedura f_foto_prozenje . . . . .	29
4.3.7	Procedura f_slash_svetloba . . . . .	29
4.3.8	Procedura hitri_adc . . . . .	30
4.3.9	Procedura setup . . . . .	30
4.3.10	Procedura loop . . . . .	31
4.4	Razlaga pomembnih delov programske kode . . . . .	32
4.4.1	Definicije dodatnih ukazov . . . . .	32
4.4.2	Manipulacija z vrati . . . . .	32
4.5	Serijski protokol . . . . .	33
4.5.1	Oblika ukazov . . . . .	33
4.5.2	Izbira računalniško vodenega načina delovanja . . . . .	34
4.5.3	Komunikacija med delovanjem v izbranem funkcijskem načinu . . . . .	34
<b>5</b>	<b>Program za delo z osebnim računalnikom</b>	<b>39</b>
5.1	Izgled uporabniškega vmesnika . . . . .	39
5.2	Komunikacija preko serijskega vmesnika . . . . .	40
<b>6</b>	<b>Uporaba naprave</b>	<b>41</b>
6.1	Priklop naprave . . . . .	41
6.1.1	Vir napajanja . . . . .	41
6.1.2	Priklop ohišja fotoaparata . . . . .	41
6.1.3	Priklop bliskavice . . . . .	42
6.1.4	Priklop tipal . . . . .	42
6.2	Upravljanje z napravo . . . . .	43
6.2.1	Samostojen način delovanja . . . . .	43
6.2.2	Računalniško vodeni način delovanja . . . . .	47
6.3	Primer uporabe . . . . .	48
<b>7</b>	<b>Sklepne ugotovitve</b>	<b>51</b>
<b>A</b>	<b>Priloge</b>	<b>53</b>
A.1	Slike naprave . . . . .	53
A.2	Slike narejene s pomočjo naprave . . . . .	59

Seznam slik	63
Seznam tabel	65
Literatura	66



# Seznam uporabljenih kratic in simbolov

ADC	( <i>angl.</i> ) <i>Analog to Digital Converter</i> ; Analogno digitalni pretvornik.
ANSI	( <i>angl.</i> ) <i>American National Standards Institute</i> .
DIP	( <i>angl.</i> ) <i>Dual In-line Package</i> ; način pakiranja elektronskih komponent.
DTR	( <i>angl.</i> ) <i>Data Terminal Ready</i> ; Kontrolni signal protokola RS-232.
EEPROM	( <i>angl.</i> ) <i>Electrically Erasable Programmable Read-Only Memory</i> . Električno izbrisljiv programirljiv bralni pomnilnik.
FLASH	( <i>angl.</i> ) <i>FLASH memory</i> ; Bliskovni pomnilnik.
FTDI	( <i>angl.</i> ) <i>Future Technology Devices International</i> .
GCC	( <i>angl.</i> ) <i>GNU Compiler Collection</i> ; Zbirka prevajalnikov GNU.
ISO	( <i>angl.</i> ) <i>International Organization for Standardization</i> .
ISP	( <i>angl.</i> ) <i>In System Programming</i> ; Programiranje naprav nameščenih v ciljni napravi.
LCD	( <i>angl.</i> ) <i>Liquid Crystal Display</i> ; Zaslon s tekočimi kristali.
PC-SYNC	( <i>angl.</i> ) <i>Prontor-Compur Synchronization</i> ; Standardni priključek za sinhronizacijo bliskavic.
PID	( <i>angl.</i> ) <i>Product ID</i> ; Identifikacijska številka proizvoda.
PIMA	( <i>angl.</i> ) <i>Photographic and Imaging Manufacturers Association</i> .
PTP	( <i>angl.</i> ) <i>Picture Transfer Protocol</i> ; Protokol za prenos slik.
RC	( <i>angl.</i> ) <i>Resistor-Capacitor circuit</i> ; Vezje ki vsebuje upor in kondenzator.
SRAM	( <i>angl.</i> ) <i>Static Random Access Memory</i> ; Statični pomnilnik.
SSOP	( <i>angl.</i> ) <i>Shrink Small-Outline Package</i> ; način pakiranja elektronskih komponent.
UART	( <i>angl.</i> ) <i>Universal asynchronous receiver/transmitter</i> ; Univerzalni asinhroni sprejemnik/oddajnik.
USB	( <i>angl.</i> ) <i>Universal Serial Bus</i> ; Univerzalno serijsko vodilo.
VID	( <i>angl.</i> ) <i>Vendor ID</i> ; Identifikacijska številka proizvajalca.



# Povzetek

Bistvo fotografije je zamrznitev nekega trenutka v času in prenos zajete informacije na fotografski papir. Visokohitrostna fotografija je zvrst fotografije, ki se ukvarja z zajemanjem delov hitro odvijajočih dogodkov. To so predvsem dogodki, katere zaradi hitrosti človeško oko ne zazna. Kot primer lahko navedemo dotik kapljice vode z neko površino. Človek dotik sicer zazna, a šele s pomočjo visokohitrostne fotografije je dejansko vidno, kaj se v določenem delu dotika kapljice s površino res dogodi. Problem nastane pri izbiri trenutka za proženje fotoaparata, ki mora biti ravno pravšnji, da bomo dogodek sploh ujeli. Ker ljudje ponavadi odreagiramo prepočasi in bi večino hitrih dogodkov zamudili, sem razvil napravo, ki odreagira na nek dražljaj in s pomočjo fotoaparata ter bliskavic pomaga ujeti željen del dogodka.

V uvodnem poglavju diplomske naloge so opisani možni pristopi k visokohitrostni fotografiji skupaj z omejitvami in prednostmi vsakega pristopa. Osrednja poglavja so namenjena razlagi strojnega in programskega dela naprave. Bistvo naprave so predvsem funkcije, s katerimi si lahko olajšamo fotografiranje v tehniki visokohitrostne fotografije, poleg tega pa vsebuje tudi funkcije za pomoč pri vsakodnevnem fotografiranju. Upravljanje je enostavno, s pomočjo kontrolnih tipk in informacij na prikazovalniku LCD, ali pa s pomočjo programa na osebnem računalniku. Upravljanje z napravo je opisano v zadnjem delu naloge. V zaključku je podanih nekaj predlogov za izboljšave ter priloženih nekaj fotografij dogodkov, narejenih s pomočjo te naprave.

## **Ključne besede:**

visokohitrostna fotografija, fotoaparat, bliskavica, intervalometer, tipalo



# Abstract

The main purpose of photography is to capture a moment in time and transfer the captured information onto the photographic paper. High speed photography is a photography technique in which parts of a very fast phenomenon are being captured. A phenomenon, which usually happens too fast to be captured by the human eye. An example of such is contact of a water drop with another surface. A person is capable to register the contact, but it is only by means of the high speed photography that one is able to see what happens at the exact moment of contact. The main problem here is the appropriate timing of the shutter release because the shutter must be released at exactly the right moment in order to capture the desired phenomena. Because the human reaction is too slow and therefore most of the phenomena would be lost. I developed a device which reacts to a stimulus and together with a camera and flash units helps capturing the desired moment.

The different approaches to the high speed photography, along with the positive and negative side of each approach, are described in the introductory chapter of this work. In the main chapters the software and hardware of the device is being explained. The main characteristics of the device are the functions which make the high speed photography technique easier to use. Also incorporated are the functions that come in handy in everyday photography. The device is easily managed with the use of control buttons and information displayed on a LCD screen or with the use of a special program on personal computer. The device management is described in the last part of this work. In the final part a few suggestions for improvements are given and a few photo examples made by the use of this device are presented.

## **Key words:**

high speed photography, camera, flash, intervalometer, sensor



# Poglavje 1

## Uvod

Smisel visokohitrostne fotografije je zajeti čim krajši čas trajanja nekega hitrega dogodka. Na prvi izziv naletimo že ob vprašanju, kako sploh ujeti nek zelo hiter dogodek in kako del le tega zabeležiti na fotografiji. Če slikamo neke dogodke, ki se ponavljajo v enakomernih periodah (npr. padanje kapljice vode iz cevi), lahko z več poskusi in nekaj sreče ujamemo željen del dogodka. Fotografijo sicer imamo, vendar je izkupiček uporabnih slik dokaj majhen, ponovitev posnetka pa je skoraj nemogoča. Problem nastane pri slikanju enkratnih dogodkov, za katere niti ne vemo koliko časa trajajo, niti kdaj se bodo pričeli izvajati (npr. pok balona). V tem primeru je potrebno uporabiti neko napravo, s pomočjo katere lahko zaznamo dogodek ter zajamemo sliko v točno določenem času izvajanja le tega. Ker v času, ko se mi je porodila ideja o tovrstnem fotografiranju, na tržišču ni bilo naprave, ki bi zadoščevala vsem mojim zahtevam, sem se odločil tako napravo narediti sam. Zahteve so bile sledeče:

- naprava omogoča ločeno proženje ohišja fotoaparata in bliskavice;
- naprava omogoča uporabo različnih tipal za zaznavanje dogodkov;
- naprava mora čim hitreje odreagirati na nek zunanji dražljaj in sicer mora biti hitrost reagiranja tolikšna, da ob poku balona, napolnjenega z vodo, omogoča zajem fotografije, na kateri voda še ni izgubila oblike balona;
- obstajati mora možnost nastavljanja zamika med zaznavo dogodka ter proženjem fotoaparata ali bliskavice;
- poleg svojih naprednejših funkcij za pomoč pri visokohitrostni fotografiji mora biti uporabna še kot žično prožilo in intervalometer;
- enostavna uporaba.



## Poglavje 2

# Pristop k problemu

### 2.1 Način fotografiranja

Pri visokohitrostni fotografiji obstaja več različnih pristopov, ki se razlikujejo v načinu zajema dogodka. Opredelil se bom zgolj na način zajema dogodka z uporabo fotoaparata. Slikanje hitrih dogodkov je sestavljeno iz dveh zaporednih faz - reakcije na nek dražljaj ter samega zajema slike. V nadaljevanju sta opisana dva možna načina zajema slike pri čemer predpostavljamo, da smo na dražljaj že odreagirali.

#### 2.1.1 Način s proženjem fotoaparata

Eden od načinov visokohitrostne fotografije je način s kontroliranim proženjem zaklopa ohišja fotoaparata. Ta način je v bistvu zelo podoben normalnemu slikanju s fotoaparatom, kjer pritisnemo gumb in počakamo da fotoaparat opravi svoje delo. Ta način je uporaben ob pogoju, da reakcijski čas ni bistvenega pomena. Zaradi fizične zgradbe ter delovanja fotoaparata se pojavi t.i. zamik proženja (lag) [1]. To je zamik med časom, ko fotograf do konca pritisne gumb za proženje in časom, v katerem je slika dejansko zajeta. Zamik proženja je sestavljen iz dveh zamikov - zamika potrebnega za samodejno ostrenje in izračun pravilne osvetlitve (autofocus lag) ter zamika proženja zaklopa (shutter release lag). Zamik proženja se bistveno razlikuje med različnimi tipi fotoaparatorov (kompaktni fotoaparati, ragefinderji, zrcalnorefleksni aparati) in tudi med različnimi razredi znotraj enega tipa (aparati namenjeni novinarskemu delu so praviloma precej hitrejši kot aparati namenjeni studijskemu delu). Največja razlika med aparati se pojavi pri zamiku potrebnem za samodejno ostrenje - le ta je zelo odvisen od same zgradbe aparata. Kompaktni fotoaparati praviloma

uporabljajo ostrenje na principu kontrasta na delu slike [2], kar je relativno počasen postopek. Po končanem samodejnem ostrenju (focus lock) se opravi še izračun pravilne osvetlitve (metering, exposure). Za vse to delo se uporabljajo različni algoritmi, kateri podatke dobivajo iz glavnega tipala aparata. Pri zrcalnorefleksnih fotoaparatih je proces bistveno hitrejši. Ti aparati imajo v svoji zgradbi polprosojna zrcala (od tukaj tudi ime), katera del zajete svetlobe odbijajo na posebno tipalo za avtomatsko ostrenje, kateri uporablja hitrejšo metodo z ugotavljanjem fazne razlike intenzitete svetlobe [2], del odbijajo na tipalo, ki sproti računa vrednosti za pravilno osvetlitev, del pa se odbija v optično kukalo. Glavno tipalo je uporabljeno zgolj za zajem slike. Ker se visokohitrostna fotografija ponavadi izvaja v kontroliranem okolju ta podatek niti ni tako pomemben, saj se ponavadi uporabi ročno ostrenje ter ročne nastavitve osvetlitve. Bolj pomemben je zamik proženja zaklopa. To je čas med polnim pritiskom gumba in dejanskim zajemom slike. Predvideva se, da sta samodejno ostrenje in izračun pravilne osvetljenosti že opravljena. Tu so razlike med fotoaparati precej manjše, a za visokohitrostno fotografijo bistvene. Pri kompaktnih fotoaparatih je to zgolj čas branja informacije iz glavnega tipala (scanning time), saj je tipalo vedno izpostavljen vpadni svetlobi. Pri zrcalnorefleksnih fotoaparatih pa je to čas potreben za umik polprosojnega ogledala izpred zaklopa, čas za odpiranje mehanskega zaklopa ter čas branja informacije na glavnem tipalu. Kljub temu so zrcalnorefleksni fotoaparati hitrejši, saj uporabljajo kvalitetnejša tipala ter hitrejša algoritme za zajem.

Za zajem hitrega dogodka je pomemben še čim krajši čas osvetljevanja tipala (čim krajša ekspozicija). Če je neko gibanje hitrejšo, kot ga lahko zajamemo v določenem trenutku časa, bo slika razmazana in kot taka neuporabna. Čas najkrajše ekspozicije pa je zopet odvisen od same zgradbe fotoaparata. Kompaktni fotoaparati imajo najkrajši čas zajema okrog 1/1500s, boljši kompaktni aparati okrog 1/2000s. Bistveno bolje se obnesejo zrcalnorefleksni aparati, pri katerih že aparati nižjega razreda dosežejo ekspozicijo dolžine 1/4000s, aparati višjega razreda pa 1/8000s. Kljub temu je lahko čas zajema 1/8000s prepočasen. Poleg tega za zajem slike ob tako kratkem času ekspozicije potrebujemo izpolnjeno vsaj še eno izmed naslednjih zahtev:

- zelo močan vir svetlobe (saj zaradi kratkega časa ekspozicije na tipalo pade zelo malo svetlobe);
- objektiv z veliko vrednostjo zaslonke in/ali;
- visoko občutljivost ISO tipala.

Žal smo s temi parametri velikokrat omejeni in je izvedba posnetka zaradi tega neizvedljiva. Alternativno lahko zato uporabimo način s proženjem bliskavice.

### 2.1.2 Način s proženjem bliskavice

Način s proženjem bliskavice zahteva čisto drugačen pristop. Ta način je zanimivejši, ker na hitrost zajema slike ne vpliva hitrost delovanja fotoaparata, temveč zgolj hitrost trajanja bliska bliskavice. Pomanjkljivost, ali pa tudi ne, pa je delo v še bolj kontroliranem okolju. Pri tovrstni fotografiji je zaradi pomanjkanja ročnih nastavitv onemogočeno delo s kompaktnimi fotoaparati, zato se ponavadi uporabljajo zrcalnorefleksni fotoaparati, pri katerih lahko ročno kontroliramo vse parametre. Ideja je v tem, da se vse delo opravlja v popolni temi. Fotoaparat postavimo na fotografsko stojalo, ročno nastavimo ostrino ter nastavitve same ekspozicije (dolžina proženja zaklopa, vrednost zaslonke, vrednost občutljivosti ISO tipala, nastavitve beline). Nastavitve so odvisne od same postavitve fotografiranega objekta, postavitve bliskavic, uporabljenega objektiva ter seveda željenega rezultata. Izkušeni fotograf nastavitve običajno nastavi po občutku, drugače pa se ob več poskusnih posnetkih prilagaja nastavitve, dokler ne dosežemo željenega rezultata. Na (iz fotografskega stališča) primerno mesto ob slikanem objektu postavimo še naš glavni vir svetlobe - bliskavico (oz. več bliskavic, če je potrebno). Tudi nastavitve na bliskavici, kot so moč bliska ter širina svetlobnega snopa, je potrebno ročno nastaviti. V tem načinu logični sistemi bliskavice ne poskušajo prilagajati nastavitv, ali celo sprožiti predbliskov, ter le sprožijo blisk, ko je potrebno. Čas potreben od ukaza za proženje bliska in samim bliskom je tako skoraj ničen in zato bliskavica v tem primeru ne predstavlja omembe vrednega zamika pri fotografiranju. Bistveni pomen pri tej vrsti fotografije ima torej prav uporabljena bliskavica ter izbrane nastavitve delovanja le te. Princip samega slikanja je sledeč: v popolni temi sprožimo zaklop ohišja fotoaparata in tako se začne zajemanje informacije iz vpadne svetlobe na svetlobno občutljivo tipalo. Ker delamo v popolni temi, pomeni da vpadne svetlobe ni, ter zaradi tega tudi ni nobene informacije na samem svetlobno občutljivem tipalu. Kljub temu poskušamo čas ekspozicije obdržati čim krajši, saj se ob dolgih ekspozicijah na digitalnih tipalih začne pojavljati t.i. šum - kar povzroči zrnato sliko (razmerje signal/šum je majhno, saj signala praktično ni, beležimo zgolj šum). V določenem trenutku za primerno dolžino časa sprožimo en blisk iz bliskavice. Svetloba se od fotografiranega objekta odbije na svetlobno občutljivo tipalo in se tam zajame. Ker je po blisku zopet tema, pomeni, da ne zajemamo nobene

informacije več. Po blisku seveda čim hitreje zaustavimo samo ekspozicijo na fotoaparatu. Tako smo zajeli sliko, ki je bila razkrita le za čas trajanja bliska iz bliskavice ter morebitne odbite svetlobe iz okoliških objektov, vendar je ob hitrosti potovanja svetlobe v zraku, faktorju padanja moči svetlobe ter zelo kratkih razdaljah to zanemarljivo. Pri zajemanju hitro izvajajočih dogodkov je torej bistvenega pomena, da je zajem slike čim krajši, kar pri načinu slikanja z bliskavico posledično pomeni, da je trajanje bliska čim krajše. Trajanje bliska je odvisno od samega tipa bliskavice ter moči proženja ksenonske bliskovne cevi. Praviloma močnejši bliski trajajo dlje časa. Za slikanje kratkih dogodkov se zaradi tega uporabljajo navadne, natične bliskavice, ki ob zadovoljivi moči dosežejo presenetljivo kratke čase trajanja bliska - ob manjših močeh tja do  $1/40000$ s. Studijske bliskavice so za to delo manj primerne, saj zaradi drugačnega prožilnega vezja dosežejo zgolj čase okrog  $1/2000$ s, so pa zato bistveno močnejše. Primer dolžine trajanja bliska, v odvisnosti od moči, za bliskavico Canon Speedlite 580EX je podana v tabeli 2.1 [3]. Izmerjene so povprečne vrednosti dolžine bliskov, saj je pri majhnih močeh odstopanje tudi do  $\pm 25\%$ .

Moč	Trajanje bliska [s]
$\frac{1}{1}$	$\frac{1}{1000}$
$\frac{1}{2}$	$\frac{1}{2000}$
$\frac{1}{4}$	$\frac{1}{4000}$
$\frac{1}{8}$	$\frac{1}{9000}$
$\frac{1}{16}$	$\frac{1}{15000}$
$\frac{1}{32}$	$\frac{1}{21000}$
$\frac{1}{64}$	$\frac{1}{30000}$
$\frac{1}{128}$	$\frac{1}{35000}$

Tabela 2.1: Dolžina trajanja bliska, v odvisnosti od moči, za bliskavico Canon Speedlite 580EX.

## 2.2 Proženje ob dogodku

Problem, ki se pojavi pri visokohitrostni fotografiji je, kako ujeti pravi trenutek za zajem slike. Če dogodek traja dlje časa in ne potrebujemo točno določenega trenutka trajanja tega dogodka, načeloma ni problema. Preprosto vzamemo fotoaparatus in zamrznemo trenutek. Problemi se pojavijo pri zajemu dogod-

kov, ki trajajo zelo kratek čas ali pa želimo zajeti točno določen del nekega dogodka. Pri nekaterih periodično ponavljajočih dogodkih (npr. pri enakomernem padanju kapljic vode na neko površino) se lahko slikanja lotimo po občutku, vendar je izplen uporabnih slik zelo majhen. Pri enkratnih hitrih dogodkih (npr. pri puku balona) pa take metode niso uporabne, saj niti ne vemo, kdaj točno se bo dogodek zgodil, niti koliko časa bo trajal (ponavadi premalo časa, da lahko ustrezno odreagiramo). V takih primerih nam ne preostane drugega, kot da uporabimo neko napravo, ki na določen dražljaj odreagira bistveno hitreje kot človek. V ta namen se uporabljajo razna tipala dotikov, zvoka in količine svetlobe, pa tudi druga tipala - odvisno od tipa dogodka, ki ga hočemo zabeležiti.



## Poglavje 3

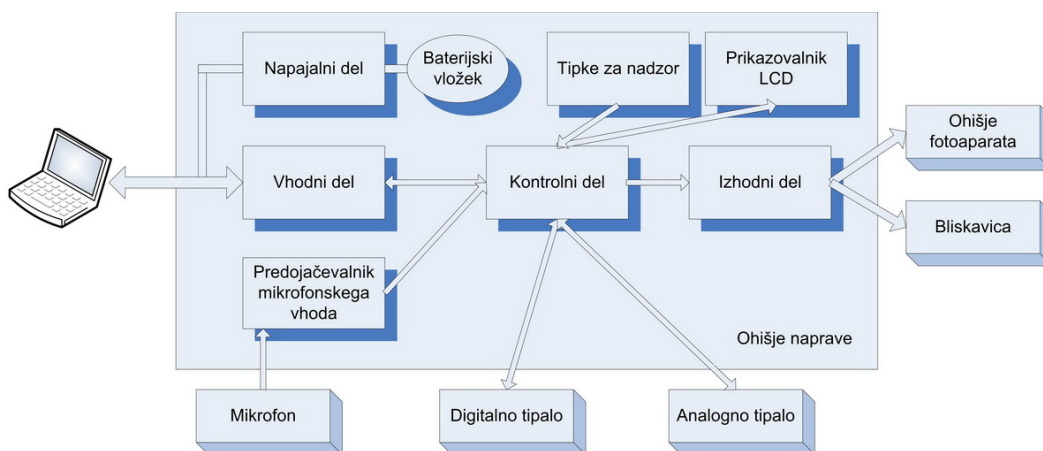
# Strojni del

Namen je bil izdelati napravo, sestavljeno iz standardnih elektronskih komponent, ki so cenovno ugodne in lokalno dobavljive. Izdelava samega tiskanega vezja naj bi bila toliko preprosta, da jo lahko vsak z malce elektrotehniške žilice reproducira. Vezje je tako v celoti sestavljeno iz petih integriranih vezij, nekaj uporov, kondenzatorjev, tipk ter prikazovalnika LCD.

Začetni razvoj je potekal na razvojni plošči Arduino Diecimila z 8-bitnim mikrokontrolerom Atmel ATmega 168-20PU. Arduino je odprtokodna razvojna plošča sestavljena iz nezahtevnih elektronskih komponent, sama uporaba pa je zelo enostavna. Mikrokontroler lahko programiramo v lastnem razvojnem okolju bodisi z uporabo zbirnika ali pa programskega jezika Wiring. Programski jezik Wiring je v bistvu le knjižnica za programski jezik ANSI C, ki pa je namenjena lahkemu programiranju mikrokontrolerov. Preko priključkov na razvojni plošči lahko dostopamo do 14 digitalnih V/I nožic ter 6 analognih vhodov (kateri se lahko uporabijo tudi kot digitalni V/I). Začetni razvoj je torej potekal v kombinaciji razvojne plošče Arduino in elektronskih elementov nameščenih na preizkusni plošči (protoboardu). Ko je bila zadeva uporabna, sem izdelal še enostransko tiskano vezje, ki je primerno za domačo izdelavo.

Naprava je sestavljena iz več delov in sicer iz osnovne plošče, prikazovalnika LCD in kontrolnih tipk ter zunanjih tipal, kot je vidno na shemi 3.1.

Naprava lahko deluje v dveh načinih in sicer v *samostojnem* načinu delovanja, kjer celotna naprava deluje samozadostno, upravljamo pa jo preko kontrolnih tipk in prikazovalnika LCD, ter v *računalniško vodenem* načinu delovanja, v katerem napravo upravljamo s pomočjo programa na osebni računalniku. Razlika je samo v načinu upravljanja z napravo, delovanje same naprave pa je identično. Omogoča priklop in uporabo različnih analognih in digitalnih tipal. Z uporabo digitalnih tipal lahko hitreje odreagiramo na nek zunanji dražljaj,



Slika 3.1: Logična shema vezja.

kot z uporabo analognih tipal. Prednost analognih tipal je možnost reagiranja na veličino nekega dražljaja (npr. jakost zvoka), z digitalnimi tipali pa lahko odreagiramo le na prisotnost oz. odsotnost nekega dražljaja. Kot analogni tipalo lahko uporabimo npr. mikrofona ali fotoupor, kot digitalno tipalo pa npr. fototranzistor ali tipko. Kakšno tipalo bomo uporabili je seveda odvisno od tipa dogodka, katerega hočemo fotografirati. Vsa tipala, z izjemo mikrofona, se priključijo na splošni priključek tipal na ohišju naprave (priključek oblike USB tipa A, viden v prilogi A.10). Mikrofona priključimo na poseben vhod, namenjen samo priključu mikrofona, saj je potrebno pridobljeni signal predhodno ojačati, kot je opisano v poglavju 3.1.2.

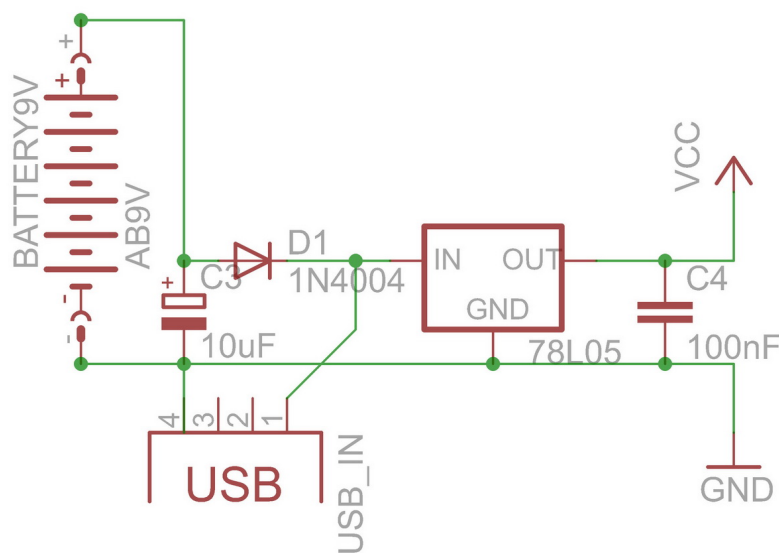
## 3.1 Osnovna plošča

Na osnovni plošči se nahaja pet sklopov in sicer napajalni del, predojačevalnik signala mikrofonskega vhoda, kontrolni del, povezovalni del ter izhodni del. Razporeditev je prikazana v prilogi A.1.

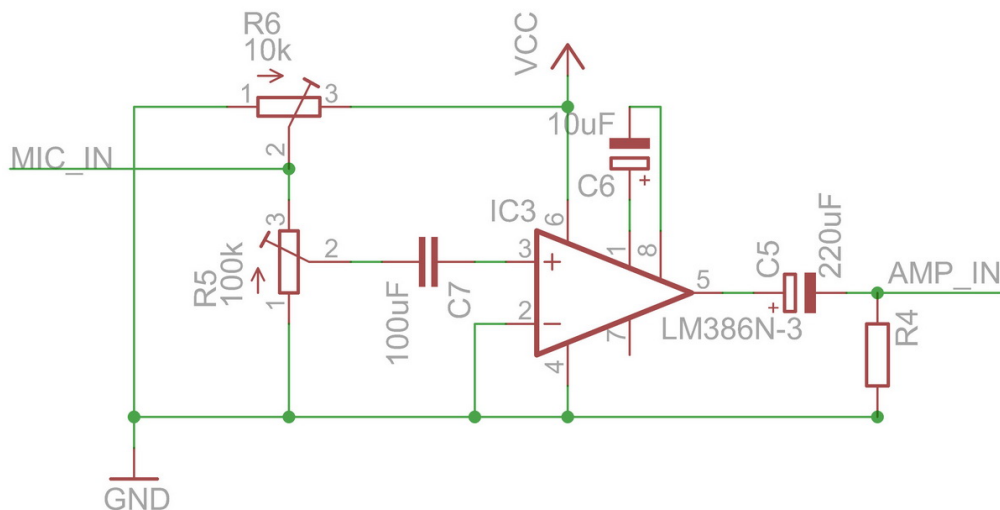
### 3.1.1 Napajalni del

Napajalni del skrbi, da vezje dobiva konstantno napetost 5V. To je zelo pomembno, saj le tako pri digitalizaciji analognih signalov za določen analogen signal vedno dobimo isto digitalno vrednost. Za omejevanje napetosti skrbi napetostni regulator LM78L05, ki omejuje napetost na 5V, dovoljuje pa obremenitve do 100mA, kar je za to vezje dovolj. Shema napajalnega dela vezja

je vidna na sliki 3.2, podrobnosti vezja pa v prilogi A.2. Naprava omogoča izbiro vira napajanja in sicer lahko uporabimo notranji ali zunanji vir napajanja. Najpogosteje se uporablja notranji vir napajanja, ki je 9V baterijski vložek, zaprt v notranjosti ohišja. Ta je primeren zaradi svoje velikosti, moč pa je zadovoljiva, da vezje deluje dovolj dolgo. Kot vir zunanje napajanja lahko uporabimo praktično katerikoli vir napetosti, ki ima enosmerni tok, ter napetost med 5 in 30V. Ob izbiri zunanje vira napajanja se notranji vir napajanja ne uporablja. Zunanji vir priključimo na vhodni priključek USB naprave (priključek USB tipa B, viden v prilogi A.10), pri čemer mora biti pozitivni pol dostopen preko nožice 1, ozemljitev pa preko nožice 4 priključka USB. Preko tega priključka naprava lahko le pridobi vir napajanja za svoje delovanje, ne more pa služiti kot vir napajanja drugim napravam. Isti priključek USB se uporablja tudi za komunikacijo z osebnim računalnikom, kot je opisano v poglavju 3.1.3. Možnost uporabe zunanje vira napajanja je precej uporabna, saj imamo fotografi ob sebi vedno vsaj nekaj kompletov 1.5V baterijskih vložkov, katere lahko z zaporedno vezavo uporabimo kot zunanji vir napajanja. Ena od možnosti zunanje napajanja je tudi napajanje preko vrat USB osebnega računalnika.



Slika 3.2: Električna shema napajalnega dela vezja.



Slika 3.3: Električna shema predajačevalnika signala mikrofonkega vhoda.

### 3.1.2 Predajačevalnik signala mikrofonkega vhoda

Kot vhodno zvočno tipalo se uporablja kondenzatorski mikrofonski. Ker le ta sam po sebi generira zelo šibak signal, ga je potrebno za uspešno digitalizacijo z analogno-digitalnim pretvornikom ustrezno ojačati. Za ojačitev zvočnega signala se uporablja nizkonapetostni zvočni ojačevalnik LM386N-3 [4]. Vezje je pakirano v DIP ohišje z šestimi nožicami in za svoje delovanje potrebuje zelo malo dodatnih zunanjih elementov. Deluje na napetostih med 4 in 18V, v mirovanju porabi zelo malo toka, ter deluje v dveh režimih - v režimu z 20-kratno ali v režimu z 200-kratno ojačitvijo. Zaradi šibkega vhodnega signala je potrebno uporabiti način z 200-kratno ojačitvijo - le to pa dosežemo z vezavo kondenzatorja velikosti  $10\mu\text{F}$  med nožici 1 in 8 na integriranem vezju. Shema predajačevalnika signala mikrofonkega vhoda je vidna na sliki 3.3, podrobnosti pa v prilogi A.3.

### 3.1.3 Povezovalni del

Povezovalni del skrbi za povezavo naprave z osebnim računalnikom preko vrat USB. Jedro povezovalnega dela predstavlja integrirano vezje FTDI FT232RL [5]. To vezje je pretvornik med vmesnikom USB in serijskim vmesnikom UART. Nekaj pomembnih značilnosti integriranega vezja:

- celoten protokol USB je realiziran v integriranem vezju, zato nam za sam



poteka preko standardnega serijskega protokola, katerega parametre lahko poljubno nastavljamo. Izhodna nožica DTR je preko kondenzatorja priklopljena na nožico RST mikrokrmilnika in omogoča avtomatski reset naprave vsakič, ko se vzpostavi nov serijski prenos, torej ob priključitvi naprave na osebni računalnik in inicializaciji naprave na strani le tega. Avtomatski reset omogoča tudi avtomatsko programiranje mikrokrmilnika, brez dodatnih naprav, saj tako ujamemo pravi trenutek, ko bootloader čaka na ukaz navideznega programatorja. Vezava je vidna na sliki 3.4. Fizično se integrirano vezje nahaja na spodnji strani tiskanega vezja (in ne na zgornji, kot ostale komponente), saj je zaradi tipa ohišja (SSOP ohišje z 28 nožicami) možna le površinska namestitvev. Zaradi majhnosti vezja in majhnih razmikov med nožicami je prav ta del vezja najbolj zahteven za izdelavo. Podrobnosti tega dela vezja so videne v prilogi A.4.

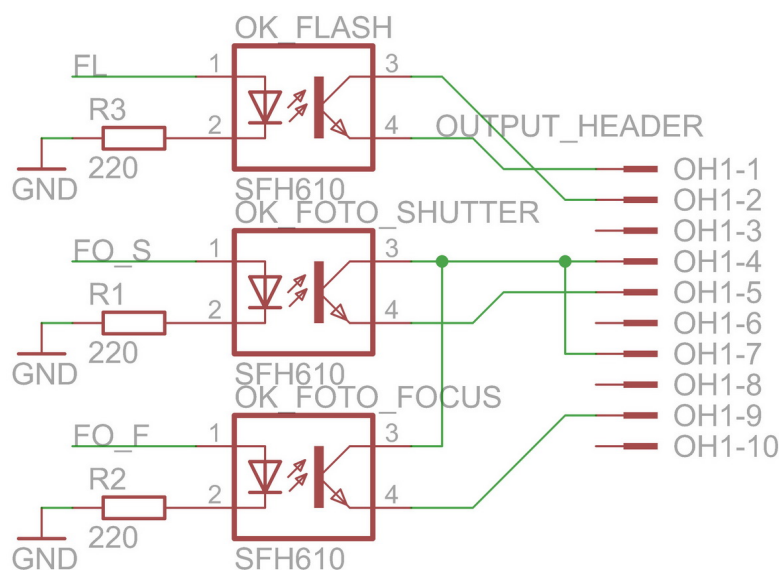
### 3.1.4 Izhodni del

Izhodni del vezja skrbi za povezavo kontrolnega dela vezja z zunanjimi napravami, ki jih hočemo upravljati, t.j. za povezavo z bliskavico ter ohišjem fotoaparata. Upravljanje z obema tipoma naprav poteka preko sklapljanja določenih nožic na priključku. Prvotno je bilo mišljeno, da bodo za to uporabljeni klasični tranzistorji, a se zaradi zaščite vezja pred nepredvidenimi zunanjimi dogodki uporabljajo optospojniki. Z njimi električno ločimo kontrolni del naprave od zunanjih naprav. Nekatere starejše, t.i. *high voltage* bliskavice, lahko ob proženju na priključne nožice vrnejo napetost nekaj sto voltov, kar pa bi zagotovo uničilo vezje. Zaradi primernih značilnosti sem izbral optospojnik SFH610A-2. Nekaj odločilnih značilnosti:

- varuje pred prebojem do napetosti 5300V;
- omogoča visoko emitorsko-kolektorsko napetost, 70V;
- primerna hitrost preklopa (pri napetosti 5V in bremenu  $75\Omega$  je čas preklopa  $2\mu s$ ).

Shema izhodnega dela vezja je vidna na sliki 3.5, podrobnosti vezja pa v prilogi A.5.

Izhodni del se po namenu loči se na izhodni del za nadzor ohišja fotoaparata ter na izhodni del za nadzor bliskavice.



Slika 3.5: Električna shema izhodnega dela vezja.

### Izhodni del za nadzor ohišja fotoaparata

Izhodni del za nadzor ohišja fotoaparata vsebuje dva optospojnika, saj je proženje proces z dvema korakoma:

1. izostritev objekta ter prilagoditev parametrov za pravilno osvetlitev (focusing, metering, exposure correcting). Tu se uporablja optospojnik OK\_FOTO\_FOCUS;
2. zajem slike s proženjem zaklopa. Tu se uporablja optospojnik OK\_FOTO\_SHUTTER.

Naprava ima različne funkcijske načine delovanja in v nekaterih načinih lahko uporabljamo oba koraka, v nekaterih načinih pa uporabljamo le proženje zaklopa. Pri tem je predpogoj uporaba ročnih nastavitvev ter zato koraka 1 ne potrebujemo. Pri načinu za visokohitrostno fotografiranje se vedno uporabi le proženje zaklopa, saj mora fotograf ročno poskrbeti za vse nastavitve. Pri načinu navadnega žičnega sprožila lahko uporabimo oba koraka - najprej izostrimo slikani predmet in nato sprožimo zaklop. V primeru ročnih nastavitvev seveda lahko uporabimo samo proženje zaklopa. Pri načinu *intervalometer* pa prvi korak lahko uporabimo za morebitno prebuditev fotoaparata iz stanja spanja v stanje pripravljenosti, kjer lahko šele sprožimo zaklop. To pride prav

pri daljših razmikih med intervali, saj se lahko med intervali ohišje fotoaparata izklopi ter tako prihrani energijo.

Tak način proženja s spajanjem nožic uporabljajo ohišja fotoaparatorov znamk Canon in Nikon. Nekateri proizvajalci imajo malce drugačen sistem, a se z uporabo minimalnega dodatnega vezja lahko proži tudi ohišja teh proizvajalcev. Ohišje fotoaparata na ta del vezja priklopimo s trižilnim vodnikom, ki ima na strani ohišja fotoaparata primeren priključek, na strani naprave pa je standardni 3.5mm stereo vtič.

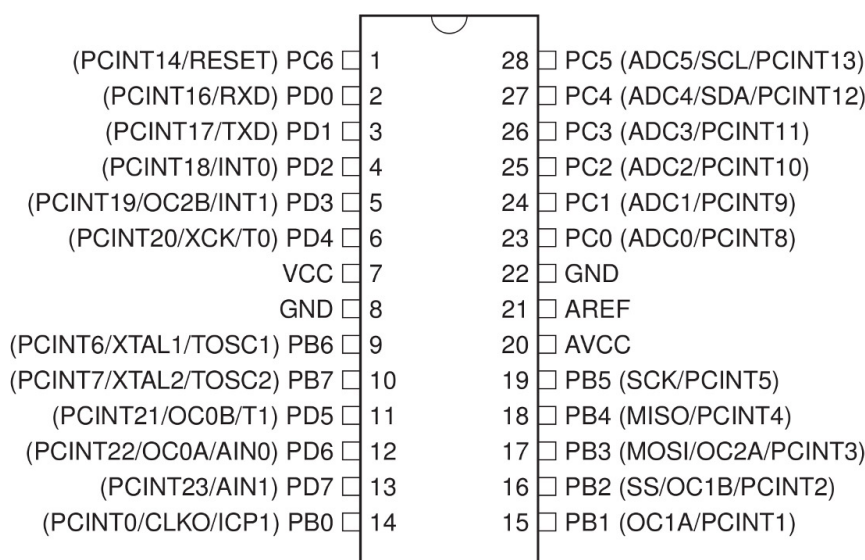
### Izhodni del za nadzor bliskavice

Proženje bliskavice dosežemo s spajanjem dveh nožic na bliskavici, zato ta del vezja vsebuje en optospojnik (OK\_FLASH). Tak način proženja je uporaben prav za vsak tip bliskavice, saj so način proženja, povezave in priključki standardizirani tako za natične kot tudi za studijske bliskavice. Generalno gledana sicer obstajata dva priključka. Prvi je t.i. PC-SYNC priključek ki ga zaenkrat najdemo na večini ohišij fotoaparatorov ter bliskavic srednjega in višjega razreda. Ta priključek je v bistvu 3.5mm koaksialni priključek, a ga žal nisem našel v nobeni trgovini z elektronskim materialom, zato sem uporabil drug standardni priključek, ki je preprost 2.5mm mono vtič. V fotografskih krogih sicer obstajajo domneve, da bodo proizvajalci kmalu nadomestili PC-SYNC priključek s klasičnim 2.5mm vtičem - to je namreč že naredilo nekaj proizvajalcev studijskih bliskavic. Natične bliskavice imajo za možnost zunanega proženja zaenkrat na uporabo le PC-SYNC priključek ali pa ga sploh nimajo. V takem primeru moramo uporabiti pretvornik, preko katerega lahko priklopimo napravo direktno na "hot-shoe" natično podnožje bliskavic, npr. kocko *Kaiser 1301*.

### 3.1.5 Kontrolni del

Celotno vezje je sestavljeno okrog 8-bitnega mikrokrmilnika Atmel ATmega168-20PU. Mikrokrmilnik obstaja v različnih oblikah pakiranja, a sem zaradi enostavnosti uporabe izbral različico v ohišju DIP z 28 nožicami. Razporeditev nožic je vidna na sliki 3.6. Ta tip mikrokrmilnika ima vse potrebne vhode in izhode, notranje elemente ter primerno hitrost in porabo. Nekaj značilnosti mikrokrmilnika [6]:

- 16KB programabilnega pomnilnika FLASH (od katerih jih zgornja 2KB porabimo za Arduino bootloader);



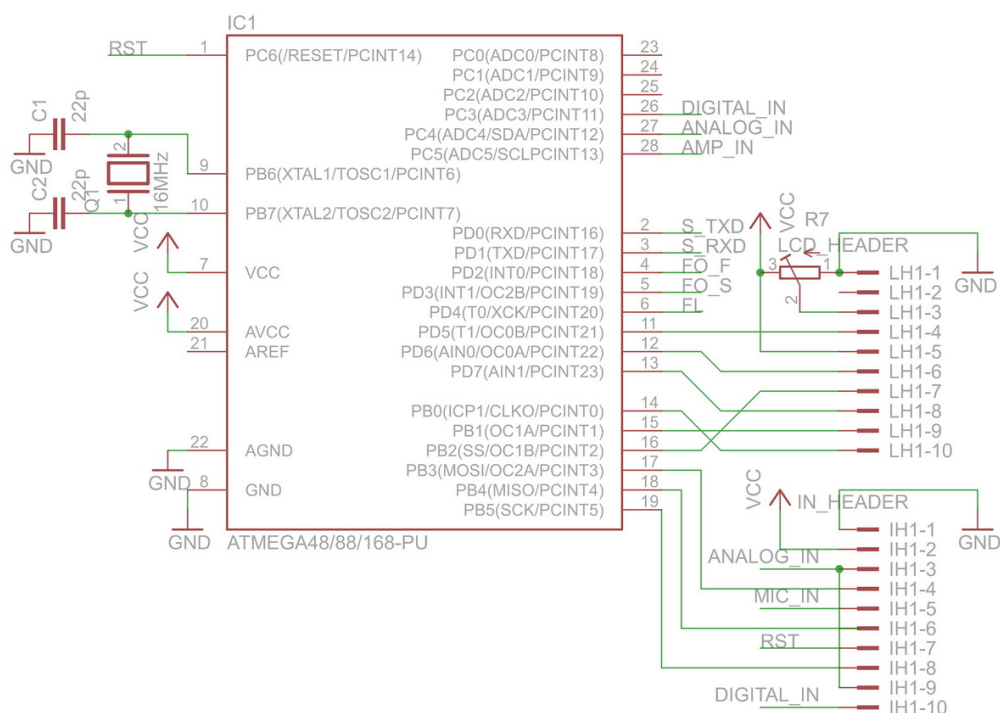
Slika 3.6: Razpored nožic mikrokrmilnika Atmel ATmega186 v ohišju DIP.

- 512B programabilnega pomnilnika EEPROM;
- 1KB pomnilnika SRAM;
- 6-kanalni, 10-bitni pretvornik ADC;
- programabilni serijski vmesnik UART;

Čeprav že vsebuje vgrajen 8MHz oscilator RC kot vir urinega signala, sem zaradi hitrejšega in stabilnejšega signala uporabil zunanji kvarčni signal s frekvenco osciliranja 16.00MHz. Fizično je procesor vstavljen v podnožje DIP z 28 nožicami. Shema kontrolnega dela vezja je prikazana na sliki 3.7, podrobnosti vezja pa v prilogi A.6.

## 3.2 Tipke za nadzor ter prikazovalnik LCD

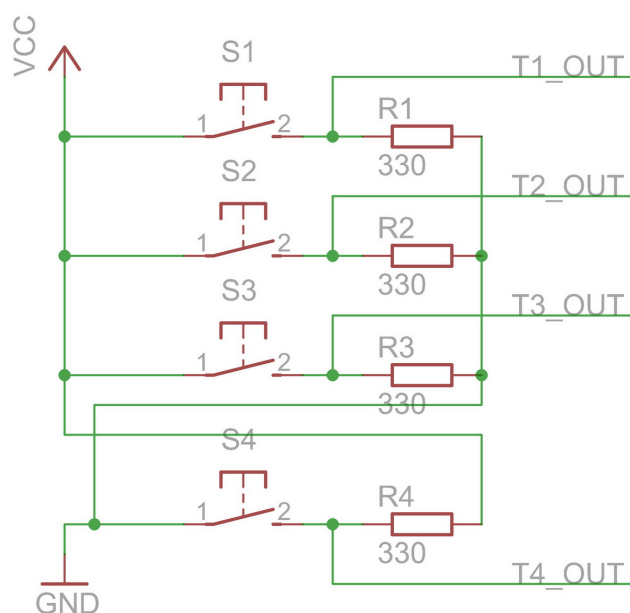
Za nadzor naprave preko menijskega vmesnika se uporabljajo štiri tipke, ki so zaradi lažje namestitve v ohišje nameščene na ločenem vezju. Na tem delu vezja se nahajajo 3 tipke (tipka plus, tipka minus, tipka OK) s katerimi se sprehajamo po menijski strukturi ter tipka reset, s katero napravo ponovno zaženemo. Na samem vezju se nahajata tudi dva priključka, kamor priklo-



Slika 3.7: Električna shema kontrolnega dela vezja.

pimo zunanja priključka na ohišju. Na štiripolni priključek se priključi splošni priključek tipk (priključek oblike USB), preko katerega zunanja tipala dobijo napajanje, ter pošiljajo svoje signale v vezje, bodisi so to analogni ali digitalni signali. Na tripolni priključek se prikljopi 3.5mm mono vtič, kamor pa priključimo kondenzatorski mikrofoni. Ta del vezja je priključen na 10 polni priključek z oznako IN na osnovni plošči. Shema vezja je vidna na sliki 3.8, podrobnosti vezja pa v prilogi A.7.

Za spremljanje nadzora na vezju se uporablja prikazovalnik LCD, velikosti 4x20 znakov, ki je nameščen na zgornjo stran ohišja. Uporabil sem prikazovalnik s kontrolnim vezjem DEM20485, kateri je kompatibilen z malce bolj standardnim vezjem HD74800. Za komunikacijo sem uporabil 4-bitni podatkovni način (namesto 8-bitni), s čemer sem privarčeval s štirimi nožicami na mikrokrmilniku ter tako malce poenostavil vezje. Prav tako sem enega od treh kontrolnih signalov (signal  $R\bar{W}$ ) povezal direktno na nizko logično stanje (saj v prikazovalnik vedno le vpisujemo podatke in iz njega nikoli beremo) in tako privarčeval še dodatno nožico. Prikazovalnik je priključen na 10-polni



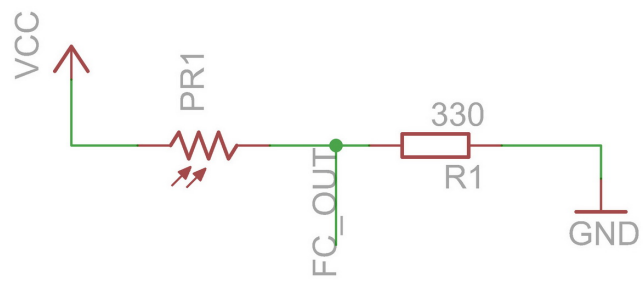
Slika 3.8: Električna shema vezja s kontrolnimi tipkami.

priključek z oznako LCD na osnovni plošči. Za uravnavanje kontrasta na prikazovalniku se uporablja  $10k\Omega$  potenciometer, ki se nahaja na osnovni plošči, v neposredni bližini priključka LCD.

### 3.3 Zunanja tipala

#### 3.3.1 Analogno svetlobno tipalo

Glavni del analognega svetlobnega tipala je fotoupor. Vezje deluje kot tokovni delilnik, pri čemer je en upor fiksni, upornost fotoupora pa se spreminja z jakostjo vpadne svetlobe - višja je jakost svetlobe, manjša je upornost fotoupora. Analogno svetlobno tipalo priklopimo na napravo preko splošnega priključka tipal, kjer je ena od nožic namenjena analognim vodom. Vezje dobi napajanje neposredno iz naprave preko istega priključka. Shema vezave analognega svetlobnega tipala je vidna na sliki 3.9, podrobnosti vezja pa so vidna v prilogi A.11.



Slika 3.9: Električna shema zunanjega analognega svetlobnega tipala.

# Poglavje 4

## Programski del

Naprava je neuporabna brez ustreznega programa mikrokrmilnika (firmware). Za razvoj sem uporabljal prostodostopno razvojno okolje Arduino development environment. Okolje omogoča pisanje kode in neposredno nalaganje programa v pomnilnik FLASH mikrokrmilnika preko bootloaderja Arduino. Za to se uporabljajo vgrajena orodja iz prostodostopne zbirke AVR-GCC. Program je napisan v programskem jeziku Wiring, ki je v bistvu programski jezik ANSI C s priloženo knjižnico Wiring ter definicijami za projekt Arduino. Za delo s prikazovalnikom LCD sem uporabil prostodostopno knjižnico LCD4Bit, katero sem moral zaradi značilnosti prikazovalnika malce prirediti.

### 4.1 Programiranje mikrokrmilnika

Mikrokrmilnik ATmega168 lahko programiramo na več načinov. Najbolj znana je uporaba klasičnega programatorja, prirejenega za programiranje serije AT-Mega Atmelovih mikrokrmilnikov. Slabost tega načina programiranja je v tem, da moramo ob vsakem programiranju odstraniti mikrokrmilnik iz ciljnega vezja, ga premakniti v programator, kjer ga programiramo, in nato vstaviti nazaj v ciljno vezje. Posledično je tak način dokaj zamuden pri samem razvoju, kjer se koda nenehno spreminja, je pa zelo uporaben in hiter za izdelavo manjših serij. Drug način je programiranje z uporabo t.i. načina ISP (In-System Programming), kjer lahko mikrokrmilnik s pomočjo programatorja ISP programiramo kar v ciljnem vezju. Ponavadi se vezje priredi tako, da so nožice mikrokrmilnika, potrebne za programiranje, dostopne preko posebnega priključka ISP. Tretji način, ki ga omogoča serija mikrokrmilnikov ATmega, in katerega sem tudi sam uporabljal, pa je programiranje preko bootloaderja nameščenega v mikrokrmilniku. Mikrokrmilnik omogoča, da v zgornjih 2KB

pomnilnika FLASH posnamemo poljuben bootloader, ki omogoča programiranje preostalih 14KB pomnilnika FLASH, ob tem pa s pomočjo varovalk skrbi, da s programiranjem ne moremo prepisati samega bootloadera. Ta način je uporaben, ker nam za programiranje ni treba odstranjevati mikrokontrolnika. Potrebno je le, da je mikrokontrolnik povezan z osebnim računalnikom tako, kot le ta pač zahteva - odvisno od bootloadera samega.

### 4.1.1 Bootloader

Za serijo mikrokontrolnikov ATmega168 je na voljo kar nekaj bootloaderjev. Zaradi enostavnosti uporabe, ter poznanega razvojnega okolja sem se odločil za bootloader Arduino [7]. Le ta deluje tako, da približno 1 sekundo po resetu mikrokontrolnika spremlja dogajanje na serijskem vmesniku UART, ki se nahaja na nožicah TXD in RXD. V primeru da se pojavi prava sekvenca ukazov, sproži programiranje spodnjih 14KB pomnilnika FLASH. Če po pretečenem času ni primerne zahteve po programiranju, bootloader prepusti izvajanje programu, ki je v drugem delu pomnilnika. Deluje na hitrosti 19200bps ter zavzame 2KB pomnilnika FLASH. Za svoje delovanje torej potrebuje serijsko komunikacijo, katero v vezju zagotavlja integrirano vezje FD232RL na vhodnem delu osnovne plošče. Bootloader se predstavi kot programator tipa *avrISPmkii*, tako da ga lahko programiramo z orodji, ki omogočajo programiranje s tem programatorjem, npr. z orodjem *avrdude*.

### Nalaganje bootloadera

Ko kupimo mikrokontrolnik, je njegov pomnilnik praviloma popolnoma prazen. V primeru da hočemo tak mikrokontrolnik programirati s pomočjo bootloadera, moramo bootloader najprej nekako naložiti v pomnilnik mikrokontrolnika. Za to potrebujemo klasični ali programator ISP. Kot programator ISP sem uporabil kar razvojno ploščo Arduino. Na mikrokontrolnik naložimo program *MegaISP*, ki je emulator programatorja ISP tipa *AVRISP*. Tako se razvojna plošča Arduino osebnemu računalniku predstavi kot programator. Preostane nam le, da pravilno povežemo nožice med mikrokontrolnik in z ustreznimi orodji posnamemo prostodostopni bootloader na ciljni mikrokontrolnik [8]. Na koncu še pravilno postavimo varovalke (fuses) na mikrokontrolniku, s čemer onemogočimo izbris ali prepisovanje dela pomnilnika, kjer se nahaja bootloader.

### 4.1.2 Nalaganje programske kode

Vezje je zasnovano tako, da za programiranje lahko uporabimo okolje Arduino development environment, iz katerega lahko direktno prevajamo programe in jih posnamemo v pomnilnik mikrokrmilnika. Povezati je potrebno le izhodni priključek USB na napravi z osebnim računalnikom, namestiti omenjeno okolje ter sprožiti programiranje. To omogoča programiranje brez kakršnegakoli razstavljanja naprave.

## 4.2 Uporabljene knjižnice

### 4.2.1 Knjižnica za delo z prikazovalnikom LCD

Za delo z prikazovalnikom LCD sem uporabil prostodostopno knjižnico *LCD4Bit*, ki vsebuje vse potrebne procedure za inicializacijo ter samo delo s prikazovalnikom. Ker knjižnica podpira le prikazovalnike velikosti do 2x20 znakov (2 vrstici po 20 znakov), sem jo moral prirediti, da podpira delo z 4x20 znakovnim prikazovalnikom. Procedure za delo s prikazovalnikom žal niso najhitrejše in poberejo kar nekaj procesorskega časa. Prav tako knjižnica ne deluje dobro, če v programu uporabljamo prekinitve. V takem primeru se lahko zgodi, da so izpisi na prikazovalniku naključni, v najslabšem primeru pa se lahko delovanje programa popolnoma zaustavi. Pred uporabo procedur je treba v datoteki *LCD4BitT.cpp* definirati, na katere nožice mikrokrmilnika je priklopljen prikazovalnik, torej v našem primeru 2 kontrolni ter 4 podatkovne linije. V datoteki glavnega programa nato le določimo število vrstic prikazovalnika, ga inicializiramo ter uporabimo ustrezne procedure. Inicializacija poteka na sledeč način:

```
#include <LCD4BitT.h>
LCD4Bit lcd = LCD4Bit(4);  \\4-vrstični LCD prikazovalnik
lcd.init();  \\inicializacija prikazovalnika
```

Uporabljene procedure:

```
lcd.clear();  \\izbris vsebine prikazovalnika
lcd.println(char msg[]);  \\izpis niza
lcd.cursorTo(int line_num, int x);  \\pomik kurzorja na lokacijo
```

## 4.2.2 Knjižnica za delo s pomnilnikom EEPROM

Vse potrebne nastavitve naprave se zapisujejo v pomnilnik EEPROM, ki je vgrajen v mikrokrmilnik. Za delo s tem pomnilnikom je treba uporabiti priloženo knjižnico EEPROM.h. Na voljo je 512B pomnilnika. Uporabljene procedure:

```
#include <EEPROM.h>
byte EEPROM.write(int address); \\pisanje v pomnilnik
byte EEPROM.read(int address); \\branje iz pomnilnika
```

## 4.3 Opisi procedur

### 4.3.1 Procedura `char* read_serial()`

Procedura `read_serial()` prebere vrednost iz vhodnega izravnalnika serijskega vmesnika UART ter vrne tekst prejetega ukaza. V izravnalniku so shranjeni prejeti podatki velikosti 1 bajt in če jih hočemo sestaviti v neko smiselno celoto, je potrebno prebirati bajt za bajtom, ter ga sestaviti v tekst. Ko se branje začne, poteka dokler so v izravnalniku še podatki. Ob branju se izloči še znak za začetek (<) in znak za konec (>) prenosa, kakor je definirano v protokolu prenosa.

### 4.3.2 Procedura `void lcd_print_credits()`

Procedura izpiše ime naprave, različico naložene programske kode, avtorja, ter vnese zamik, potreben za inicializacijo ob morebitni uporabi serijskega vmesnika.

### 4.3.3 Procedura `int spremeni_vrednost_gumb(int GUMB_CUR, int GUMB_PRE, int SPR, int SMER)`

Procedura za nastavljanje vrednosti parametrov preko kontrolnih tipk na ohišju. Kot argumente sprejme trenutno stanje tipke (`GUMB_CUR`), prejšnje stanje tipke (`GUMB_PRE`), začetno vrednost spremenljivke, kateri se bo spreminjala vrednost (`SPR`) ter smer spreminjanja vrednosti (`SMER`) v pozitivno ali negativno. Kot rezultat vrne novo, spremenjeno vrednost spremenljivke `SPR`. Primer klica procedure iz programa:

```
if (GUMB_M_CUR == HIGH)
{
    VREDNOST=spremeni_vrednost_gumb(GUMB_M_CUR,GUMB_M_PRE,VREDNOST,1);
}
```

Ob pritisku na tipko se vrednost spremenljivke spreminja v koraku po 1 v določeno smer ob vsakem pritisku. Če je tipka pritisnjen več kot 1 sekundo, potem se vrednost spreminja s koraku po 10 v določeno smer, dokler držimo tipko. Procedura dovoli zalogo vrednost spremenljivke od vrednosti 0 do vrednosti 9999. Prav tako ob spreminjanju poskrbi za ustrezen prikaz na prikazovalniku LCD.

#### 4.3.4 Procedura `int menu()`

Procedura skrbi za delo z menijsko strukturo ter prikaz le te na prikazovalniku LCD. Po menijski strukturi se sprehajamo s kontrolnimi tipkami, ko pa pridemo do končnega menija, procedura vrne številko izbranega načina delovanja, katera se kasneje interpretira v glavni zanki programa.

#### 4.3.5 Procedura `void f_intervalometer(int ZAMIK, int ST_INTERVALOV)`

Procedura skrbi za izvajanje načina delovanja *intervalometer*. Kot argument sprejme število intervalov ter zamik med njimi, ter v zanki poskrbi da se intervali izvedejo do konca, ter da je zamik med intervali tolikšen, kot je bil zahtevan iz strani uporabnika. Hkrati poskrbi še za prikaz stanja intervalometra (število preostalih intervalov ter določeni zamik) na prikazovalniku LCD.

#### 4.3.6 Procedura `f_fotoprozenje_bulb()`

Procedura poskrbi za proženje zaklopa v ohišju fotoaparata v t.i. načinu BULB. Ob pritisku na tipko OK sproži zaklop, ter zadrži proženje do naslednjega pritiska tipke.

#### 4.3.7 Procedura `f_flash_svetloba_slave(int i_AMBIENT)`

Procedura kot argument vzame odčitek vrednosti moči ambientalne svetlobe iz analognega tipala s fotocelico. Ob povečanju moči svetlobe nad določeno

vrednostjo preko ambientalne svetlobe, sproži bliskavico. Tako bliskavica deluje v načinu *optični suženj*, saj se npr. ob proženju bliskavice gospodarja, bliskavica suženj skoraj sočasno sproži.

### 4.3.8 Procedura void hitri\_adc()

S proceduro omogočimo hitrejšo delovanje analogno/digitalnega pretvornika (ADC). Vgrajeni ADC deluje na lastni frekvenci, ki je z nekim delilnikom (prescale) deljena frekvenca delovanja jedra mikrokontrolerja. Privzeta vrednost delilnika v jeziku Wiring je 128, kar pomeni da ADC deluje na frekvenci 125KHz ( $16\text{MHz}/128=125\text{KHz}$ ). Vsaka analogno/digitalna pretvorba potrebuje 13 ciklov ADC, kar pomeni da je frekvenca vzorčenja okrog 9600Hz ( $125\text{KHz}/13=9615\text{Hz}$ ), torej ena pretvorba traja okrog  $100\mu\text{s}$ . Glede na dokument [6] je priporočena frekvenca delovanja ADC do 200KHz, vendar tudi frekvence do velikosti 1MHz bistveno ne vplivajo na točnost pretvorbe, pri višjih frekvencah pa je lahko verodostojnost pretvorbe močno vprašljiva. Z uporabo procedure *hitri\_adc()* nastavimo delilnik na vrednost 16, kar pomeni da ADC deluje na frekvenci 1MHz ( $16\text{MHz}/16=1\text{MHz}$ ). To frekvenco delimo s 13 cikli, ki so potrebni za eno analogno/digitalno pretvorbo in dobimo frekvenco vzorčenja okrog 77KHz ( $1\text{MHz}/13=76923\text{Hz}$ ), torej ena pretvorba traja okrog  $13\mu\text{s}$ . S tem smo dosegli 8-kratno pohiترitev analogno/digitalne pretvorbe, pri čemer je pretvorba še vedno verodostojna. Ukaz za nastavljanje delilnika v jeziku Wiring ne obstaja, zato je potrebno direktno nastaviti vrednost bitov v registru ADCSRA. Koda za nastavljanje delilnika na vrednost 16 je sledeča (za pravilno nastavitvev je potrebno prepisati vrednost iz navodil za uporabo mikrokontrolerja):

```
sbi(ADCSRA,ADPS2);
cbi(ADCSRA,ADPS1);
cbi(ADCSRA,ADPS0);
```

### 4.3.9 Procedura void setup()

Procedura *setup()* je standardna, rezervirana procedura jezika Wiring. Ob začetku izvajanja programske kode, se najprej enkrat izvede koda, ki je napisana v tem delu procedure, nato pa se izvajanje prepusti proceduri *loop()*. V tej proceduri poteka predvsem začetna nastavitvev mikrokontrolerja. Nastavi se, katere nožice mikrokontrolerja so definirane kot vhodi, katere kot izhodi, sprožita se inicializacija prikazovalnika LCD ter inicializacija serijskega vmesnika. Ob klicu procedure *lcd\_print\_credits()* se čaka tudi na morebitni ukaz

za prehod v serijski način delovanja. Preberejo se nastavitve naprave iz pomnilnika EEPROM ter se preda nadzor nad napravo proceduri `loop()`.

#### 4.3.10 Procedura `void loop()`

Tudi procedura `loop()` je standardna procedura jezika Wiring. Začne se izvajati po tem, ko izvajanje zaključi procedura `setup()` ter se nato neskončno izvaja v zanki. Procedura je načeloma razdeljena na dva dela glede na način delovanja naprave ki je lahko samostojen (naprava deluje v samostojnem načinu delovanja in je samozadostna) ali računalniško vodeni način (napravo vodimo preko programa na osebнем računalniku). Način delovanja se določi v proceduri `setup()`.

##### **Računalniško vodeni način delovanja**

Ko je izbran računalniško vodeni način delovanja se znotraj procedure `loop()` v zanki izvaja le del kode, ki je namenjen temu načinu. V vsaki ponovitvi zanke se izvaja procedura `read_serial()`, ki preveri če v serijskem izravnalniku čaka kakšen podatek, ga prebere, ter vrne njegovo vrednost. Ta podatek je ukaz, ki smo ga napravi preko serijske komunikacije poslali iz programa, ki teče na osebнем računalniku. Ko funkcija vrne prebrani ukaz se najprej dešifrira ter loči na 4 osnovne parametre, s katerimi lahko upravljamo z napravo. Način dešifriranja ukaza je opisan v protokolu prenosa v poglavju 4.5. Eden izmed prebranih parametrov je funkcijski način delovanja naprave. Glede na ta način se potem izvajajo posamezne procedure, v katerih poteka dvosmerna komunikacija med napravo ter osebnim računalnikom, odvisno seveda od funkcijskega načina samega. Glede na to, da za nadzor uporabljamo osebni računalnik, za izpis podatkov na prikazovalniku LCD ni potrebno skrbeti, zato ob prehodu v ta način delovanja na prikazovalnik izpišemo le sporočilo, da smo v tem načinu, ter ga kasneje ne spreminjamo.

##### **Samostojni način delovanja**

Samostojni način delovanja pomeni, da naprava deluje samozadostno. Vse upravljanje poteka preko kontrolnih tipk ter prikazovalnika LCD. Glede na izbran funkcijski način delovanja, katerega dobimo po klicu procedure `menu()`, se potem izvajajo pripadajoče procedure. V tem primeru moramo venomer skrbeti za izpis podatkov na prikazovalniku LCD ter pravilno upravljanje s pomočjo tipk.

## 4.4 Razlaga pomembnih delov programske kode

### 4.4.1 Definicije dodatnih ukazov

Definirana sta dva nova ukaza, ki se uporabljata za manipulacijo bitov v spremenljivki. Kot parameter pri obeh ukazih podamo ime spremenljivke ter mesto bita, katerega želimo spremeniti. Definicija ukazov je sledeča:

```
#ifndef cbi
#define cbi(sfr, bit) (_SFR_BYTE(sfr) &= ~_BV(bit))
#endif
#ifndef sbi
#define sbi(sfr, bit) (_SFR_BYTE(sfr) |= _BV(bit))
#endif
```

Ukaz *cbi* (clear bit) se uporablja za brisanje oz. postavitv določenega bita na vrednost 0. Ukaz *sbi* (set bit) se uporablja za nastavljanje oz. postavitv določenega bita na vrednost 1. Primer uporabe: v spremenljivki SPR naj se postavi vrednost bita na mestu 0 na 1.

```
sbi(SPR,0);
```

### 4.4.2 Manipulacija z vrati

V nekaterih delih kode je bistvenega pomena hitrost delovanja. Npr. pri proženju bliskavice z nekim zvočnim dogodkom je pomembno da najprej čim hitreje naredimo analogno/digitalno pretvorbo signala dobljenega iz tipala, nato čim hitreje primerjamo dobljeno vrednost z zastavljenimi pogoji ter na koncu čim hitreje sprožimo bliskavico. Hitrost analogno/digitalne pretvorbe se da povečati s klicom procedure `hitri_adc()`, na hitrost primerjanja dobljene vrednosti z nekim pogojem nimamo bistvenega vpliva, lahko pa pohitrilo sam čas, potreben za proženje bliskavice. V bistvu pohitrilo postavitv neke nožice mikrokrmilnika v željeno stanje. Za to uporabimo metodo, ki jo v jeziku Wiring imenujejo *manipulacija z biti* [9]. S pomočjo te metode direktno vpisujemo vrednosti bitov v registre, kar se pri vpisu v podatkovne registre PORTx neposredno izraža v stanju na nožicah mikrokrmilnika. Primer: postavitv nožic 7,6, in 5 vrat D v visoko logično stanje z uporabo funkcije `digitalWrite()`;

```
digitalWrite(7, HIGH);
digitalWrite(6, HIGH);
digitalWrite(5, HIGH);
```

V tem primeru se najprej postavi vrednost nožice 7, nato vrednost nožice 6 in nazadnje vrednost nožice 5. Isti proces z uporabo manipulacije bitov:

```
PORTD |= 11100000 ;
```

V tem primeru vrednost vseh nožic postavimo v enem koraku.

## 4.5 Serijski protokol

V računalniško vodenem načinu delovanja vsa komunikacija med napravo ter programom na osebem računalniku poteka po določenem protokolu, ki je razložen v nadaljevanju. Za pravilno komunikacijo je najprej potrebno nastaviti parametre prenosa in sicer:

- hitrost prenosa: 19200bps;
- količina prenesenih podatkov v enem bloku: 8 bitov;
- število stop bitov: 1;
- brez paritetnih bitov;
- uporaba DTR mora biti omogočena.

Smer pošiljanja podatkov je označena na sledeč način: če naprava pošilja podatek proti računalniškemu programu je uporabljena puščica  $\rightarrow$ , če pa podatke proti napravi pošilja program na osebem računalniku, pa je uporabljena puščica  $\leftarrow$ . Podatki, ki se pošiljajo proti napravi, morajo vsebovati znak za začetek ukaza ( $<$ ), znak za konec ukaza ( $>$ ), prenos pa se mora zaključiti z znakom za novo vrstico.

### 4.5.1 Oblika ukazov

Ukaz je sestavljen iz štirih parametrov, ki so med seboj ločeni z znakom - .

```
<AA-BBBB-CCCC-DDDD>
```

Posamezna polja predstavljajo sledeče:

- AA: Funkcijski način delovanja naprave. Možne vrednosti so:
  - 1: intervalometer;
  - 2: proženje zaklopa fotoaparata - navadno ali BULB proženje;

- 4: proženje bliskavice z zvokom;
  - 5: proženje bliskavice z analognim svetlobnim tipalom;
  - 6: način bliskavice optični suženj - z uporabo analognega svetlobnega tipala;
  - 44: branje vrednosti ambientalnega zvoka;
  - 55: branje vrednosti moči ambientalne svetlobe z uporabo analognega svetlobnega tipala.
- BBBB: parameter 1 - zamik proženja;
  - CCCC: parameter 2 - meja proženja;
  - DDDD: parameter 3 - število intervalov.

V primeru, da kakšen funkcijski način ne potrebuje podanega katerega od parametrov, je lahko vrednost neuporabljenega parametra načeloma poljubna. V izogib problemov, se pri neuporabljenih parametrih vedno uporabi vrednost nič.

### 4.5.2 Izbira računalniško vodenega načina delovanja

Ob sami inicializaciji naprave se na serijski vmesnik pošlje podatek o različici programske kode mikrokontrolerja.

→ `fotofriend-20100627`

Ta podatek prestreže računalniški program in če se različici skladata, to pomeni da lahko napravi pošlje zahtevo po računalniško vodenem načinu delovanja.

← `<startserial>`

Naprava potrdi izbran serijski način komunikacije.

→ `serial_started`

Naprava je sedaj pripravljena za sprejem ukazov, kateri morajo biti v že določeni obliki `<AA-BBBB-CCCC-DDDD>`.

### 4.5.3 Komunikacija med delovanjem v izbranem funkcijskem načinu

#### Funcijski način intervalometer

Izbira funkcijskega načina intervalometer poteka na sledeč način (v primeru sprožimo intervalometer z 8 intervali ter razmikom med intervali 3 sekunde).

← <1-3-0-8>

Naprava potrди sprejem ukaza ter začetek izvajanja funkcijskega načina intervalometer:

→ 1-start

Po vsakem izvedenem intervalu potrди izvedbo ter doda podatek o številu preostalih intervalov (v primeru preostane še 5 intervalov):

→ se\_intervalov-5

Ko se izvajanje zaključi, se potrди zaključek funkcijskega načina intervalometer:

→ 1-done

### Funkcijski način proženje zaklopa fotoaparata

Pri prehodu v funkcijski način *proženje zaklopa fotoaparata* ni potrebno podajati nobenih posebnih parametrov, zato so vrednosti vseh parametrov po dogovoru nič. Napravi sporočimo naj se pomakne v izbrani način delovanja:

← <2-0-0-0>

Naprava potrди sprejem ukaza ter začetek izvajanja funkcijskega načina *proženje zaklopa fotoparata*.

→ 2-start

Če hočemo, da se zaklop na aparatu odpre, napravi pošljemo sledeč ukaz:

← <shutter-1>

Po določenem času (odvisno od izbranega načina v računalniškem programu) napravi pošljemo ukaz za prenehanje proženja:

← <shutter-0>

Program se izvaja v zanki, in čaka na enega izmed ukazov za nadzor proženja, dokler ne pošljemo ukaza za izhod iz tega funkcijskega načina delovanja:

← <end-2>

Naprava potrди sprejem ukaza ter zaključi z izvajanjem izbranega načina:

→ 2-done

### Funkcijski način proženje bliskavice z zvokom

Pri funkcijskem načinu proženja bliskavice z zvokom vse potrebne parametre podamo vsakič ko zahtevamo ta način delovanja. Po izvedenem proženju se način avtomatsko konča. Primer: proženje bliskavice z zvokom, zamik proženja 243ms, meja proženja pri vrednosti 370.

← <4-243-370-0>

Naprava potrди sprejem ukaza ter začetek izvajanja zanke, kjer čaka na zadostitev pogojev za proženje:

→ 4-started

Ko naprava sproži bliskavico in se izvajanje zanke zaključí, se avtomatsko zaključí tudi izvajanje tega funkcijskega načina delovanja:

→ 4-done

### **Funkcijski način proženje bliskavice s svetlobo**

Tudi pri tem načinu delovanja vse potrebne parametre podamo vsakič sproti. Po izvedenem proženju se način delovanja avtomatsko konča. Primer: proženje bliskavice s svetlobo, zamik proženja 10ms. Meje proženja ni potrebno podajati, saj je mej pri tem načinu shranjena kot nastavitev v pomnilniku EEPROM.

← <5-10-0-0>

Naprava potrди sprejem ukaza ter začetek izvajanja zanke, kjer čaka na zadostitev pogoja za proženje:

→ 5-started

Po proženju bliskavice se izvajanje izbranega funkcijskega načina delovanja zaključí:

→ 5-done

### **Funkcijski način *bliskavica kot optični suženj***

Način delovanja bliskavice kot optični suženj izberemo s sledečim ukazom:

← <6-0-0-0>

Posebnost tega funkcijskega načina delovanja je, da se nikoli ne zaključí in zaradi tega ne obstajajo dodatni ukazi za krmiljenje. Razlog je v tem, ker je pri tem načinu potrebno čim hitreje odreagirati na dražljaj, in v primeru da bi poleg čakanja na potreben pogoj za proženje zanka opravljala še dodatne naloge, kot npr. serijska komunikacija, bi bilo lahko delovanje prepočasno.

### **Meritev ambientalne vrednosti zvoka**

Za meritev vrednosti ambientalne vrednosti zvoka napravi pošljemo sledeč ukaz:

← <44-0-0-0>

Naprava izvede meritev ter vrne vrednost kot parameter (primer: izmerjena vrednost je 371):

→ sound\_ambient-371

**Meritev ambientalne vrednosti svetlobe**

Za meritev vrednosti ambientalne vrednosti svetlobe napravi pošljemo sledeč ukaz:

← <55-0-0-0>

Naprava izvede meritev ter vrne vrednost kot parameter (primer: izmerjena vrednost je 96):

→ `light_ambient-96`



## Poglavje 5

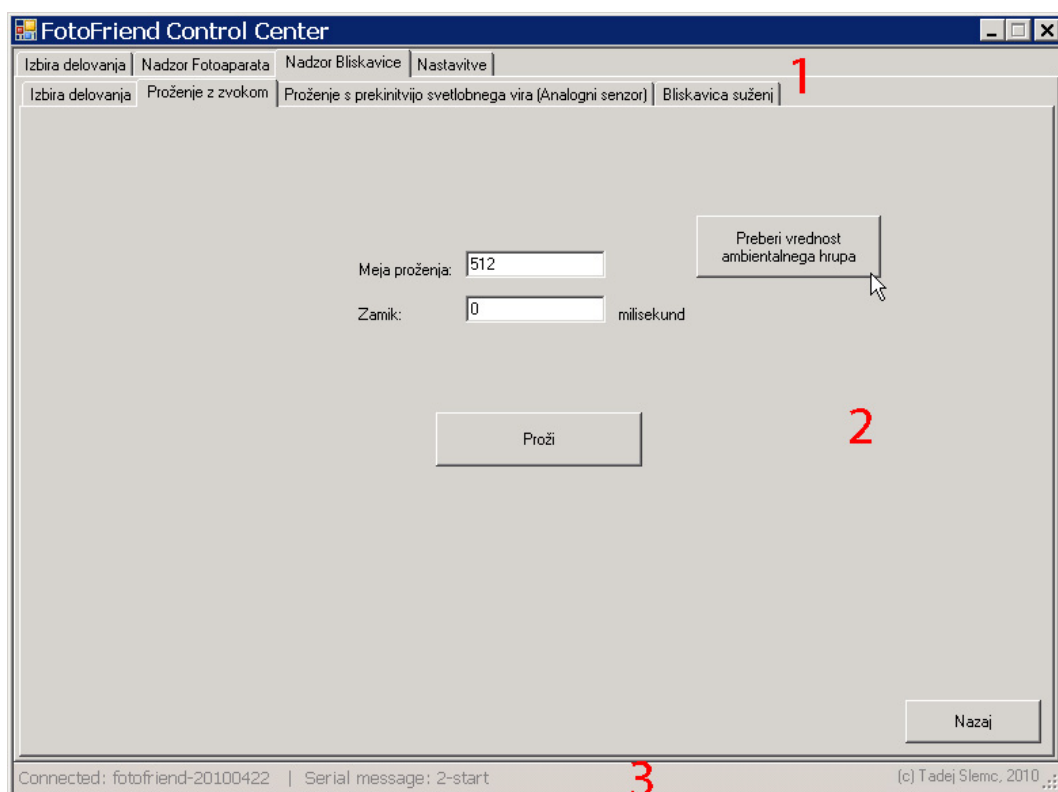
# Program za delo z osebnim računalnikom

V primeru da naprava deluje v računalniško vodenem načinu, celotno napravo upravljamo preko programa, ki teče na osebnem računalniku. Za delovanje programa potrebujemo osebni računalnik z nameščenim operacijskim sistemom Microsoft Windows, nameščenimi gonilniki za upravljanje z navideznimi serijskimi vrati ter ena prosta vrata USB. Celoten program je razvit v okolju Microsoft Visual Studio, napisan pa je v jeziku C#. Za ta jezik sem se odločil zaradi izvrstnih knjižnic za delo s serijskim vmesnikom.

Program teče v dveh nitih. Del programa, ki skrbi za serijsko komunikacijo z napravo preko serijskega vmesnika, teče v ločeni niti kot pa del programa, ki skrbi za izris uporabniškega vmesnika. S tem je zagotovljeno, da je uporabniški vmesnik odziven kljub temu, da program čaka na sporočilo iz naprave. Za delo z nitmi se uporablja sistemska knjižnica `System.Threading`.

### 5.1 Izgled uporabniškega vmesnika

Bistvo uporabniškega vmesnika je nedvoumnost in enostavnost uporabe. Vsebuje le eno okno v katerem so postavljeni zgolj najnujnejši elementi. Okno je razdeljeno na tri dele, kot je vidno na sliki 5.1. V zgornjem delu (1) se nahajajo zavihki za izbiro funkcijskega načina delovanja, v srednjem delu (2) je nastavitveno/statusno okno izbranega funkcijskega načina delovanja, v spodnjem delu (3) pa je statusna vrstica s podatki o serijski komunikaciji. Delo z uporabniškim vmesnikom je opisano v poglavju 6.2.2.



Slika 5.1: Razdelitev uporabniškega vmesnika

## 5.2 Komunikacija preko serijskega vmesnika

Za delo s serijskim vmesnikom se uporablja knjižnica `System.IO.Ports`. Ob zagonu programa se vzpostavi serijska komunikacija z napravo, pri čemer morajo biti nastavitve serijskega vmesnika takšne, kot so opisane v poglavju 4.5. Komunikacija poteka po protokolu, opisanem v poglavju 4.5.1. Takoj po vzpostavitvi serijske komunikacije program najprej preveri različico programske opreme naprave. Če se le ta sklada z različico programa na osebnem računalniku, se vzpostavi računalniško vodeni način delovanja, v nasprotnem primeru se naprava zažene v samostojnem načinu delovanja. S tem onemogočimo, da bi med komunikacijo prišlo do pošiljanja ukazov, ki niso definirani v določeni različici protokola.

# Poglavje 6

## Uporaba naprave

Naprava lahko deluje v dveh načinih in sicer v samostojnem načinu delovanja, kjer celotna naprava deluje samozadostno, upravljamo pa jo preko kontrolnih tipk in prikazovalnika LCD, ter v računalniško vodenem načinu delovanja, v katerem napravo upravljamo s pomočjo programa na osebnem računalniku.

### 6.1 Priklop naprave

#### 6.1.1 Vir napajanja

Naprava omogoča izbiro vira napajanja. Najpogostejša je uporaba naprave z uporabo notranjega vira napajanja, ki je 9V baterijski vložek, zaprt v notranjosti ohišja. Notranji vir napajanja izberemo tako, da s stikalom za vklop preprosto vklopimo napravo. Obstaja tudi možnost delovanja z zunanjim virom napajanja. Zunanji vir enosmernega toka priklopimo na vhodni USB priključek. Pozitivni priključek mora biti priklopljen na nožico 1, ozemljitev pa na nožico 4 priključka USB. Dovoljena napetost je med 5V in 30V. Kot zunanji vir napajanja tako lahko uporabimo npr. 4 zaporedno vezane 1.5V baterijske vložke ali pa napajanje preko priključka USB pridobimo iz osebnega računalnika. Pri tem načinu se naprava prižge takoj, ko je prisoten zunanji vir napajanja in ni potrebno prižgati naprave z uporabo stikala za vklop.

#### 6.1.2 Priklop ohišja fotoaparata

Ohišje fotoaparata je potrebno priklopiti v določenih funkcijskih načinih delovanja. Na strani naprave priklopimo povezovalno žico v označen izhodni priključek, ki je standardna 3.5mm stereo vtičnica. Na strani aparata pa je

potrebna priključitev v vhod za žično prožilo, tip priključka pa je različen glede na serijo ohišja ali znamko proizvajalca (lahko so uporabljeni klasični 2.5mm stereo vtiči ali pa posebni priključki, ponavadi patentirani s strani proizvajalca). Ko je aparat priklopljen, moramo ročno nastaviti potrebne nastavitve za slikanje, odvisno od uporabljenega funkcijskega načina naprave ter željenega rezultata.

### 6.1.3 Priklop bliskavice

Bliskavico na strani naprave priklopimo na izhodni priključek za bliskavico, ki je 2.5mm mono vtičnica. Na strani bliskavice imamo več načinov priklopa, odvisno od samega tipa bliskavice. Nekatere natične bliskavice imajo že vgrajeno 2.5mm mono vtičnico ali pa vtičnico PC-SYNC (3.5mm koaksialna vtičnica), kamor lahko direktno priklopimo žico. Če bliskavica teh priključkov nima, moramo z uporabo ustreznega pretvornika (npr. kocke *Kaiser 1301*) priklopiti bliskavico preko hot-shoe priključka. Po priklopu bliskavice moramo ročno nastaviti vse nastavitve bliskavice.

### 6.1.4 Priklop tipal

Naprava je narejena tako, da omogoča direkten priklop kondenzatorskega mikrofona, ter priklop poljubnega analognega ali digitalnega tipala. V tem primeru vezja s tipali napajanje pridobijo direktno iz naprave same.

#### Priklop mikrofona

Mikrofon potrebujemo pri funkcijskem delovanju proženja bliskavice s pomočjo zvoka. Na napravo priključimo kondenzatorski mikrofon preko mikrofonskega vhoda, ki je standardni 3.5mm mono vhod.

#### Priklop poljubnega analognega ali digitalnega tipala

Na napravo lahko priključimo poljubno analogno ali digitalno tipalo preko splošnega priključka tipal. Vezje s tipalom lahko pridobi napajanje direktno iz naprave in sicer je na nožici 1 vtiča USB prisotnih 5V, na nožici 4 pa ozemljitev. Žici 2 in 3 sta vhodni in sicer se nožica 2 uporablja za vhod analognih signalov, nožica 3 pa za vhod digitalnih signalov.

## 6.2 Upravljanje z napravo

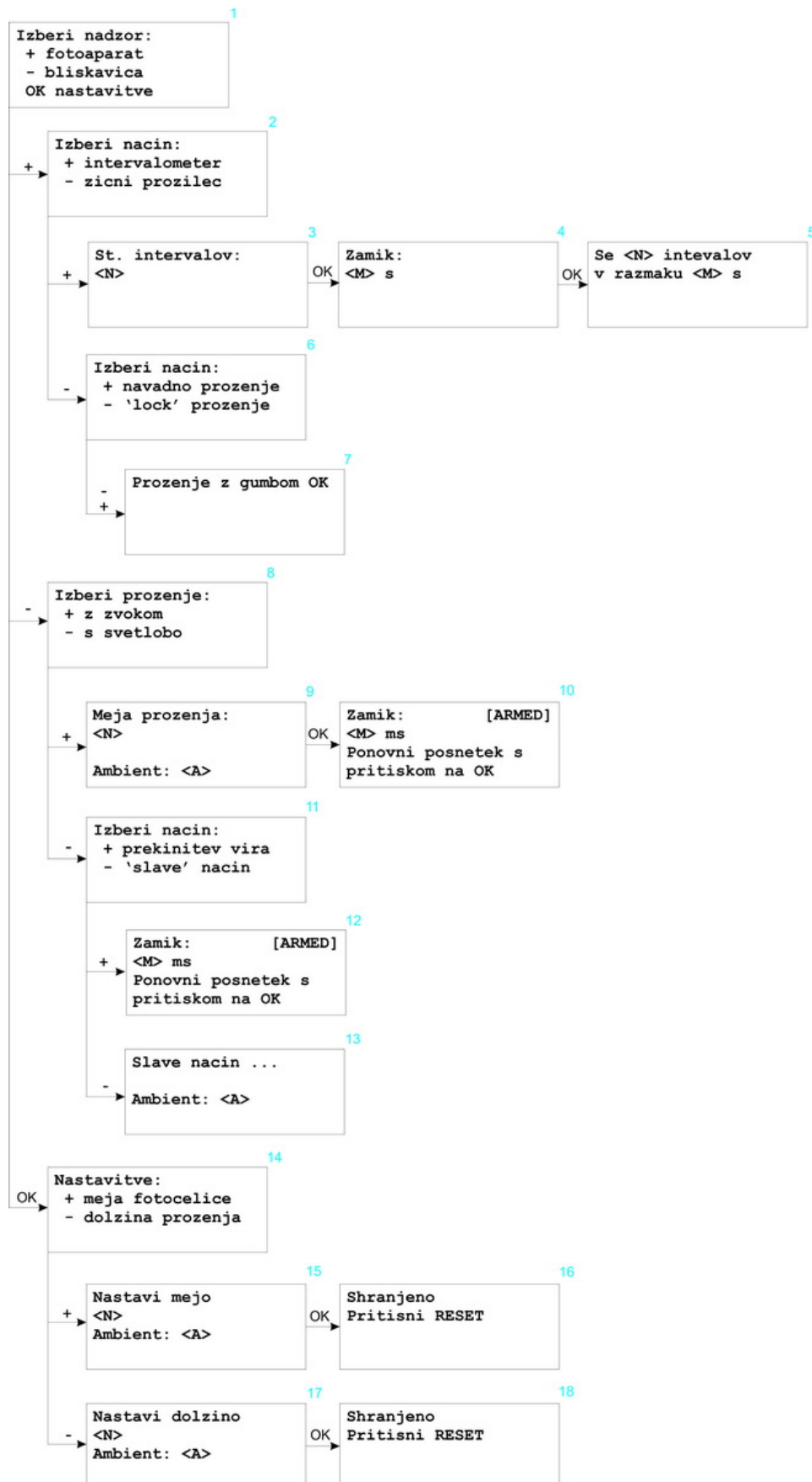
Upravljanje z napravo se razlikuje glede na to, v kakšnem načinu uporabljamo napravo. Način delovanja se izbere ob zagonu naprave. V primeru, da je naprava priključena na osebni računalnik in smo v uporabniškem vmesniku za upravljanje z napravo podali zahtevo za vzpostavitev računalniško vodenega načina delovanja, se izbere ta način delovanja, v nasprotnem primeru se privzame samostojen način.

### 6.2.1 Samostojen način delovanja

Napravo upravljamo preko menijskega uporabniškega vmesnika. Vmesnik se prikazuje na prikazovalniku LCD, za sprehajanje po menijski strukturi ter nastavitve parametrov pa uporabljamo kontrolne tipke, ki se nahajajo na ohišju naprave. Uporaba vmesnika je pregledna in enostavna. Na prikazovalniku LCD sta vedno izpisani le dve možnosti izbire podmenija, v končnih nastavitvenih oknih, pa so izpisani le podatki pomembni za delovanje naprave. Menijska struktura je prikazana na sliki 6.1.

Ko napravo prižgemo na prikazovalniku izpišejo podatki o različici programske kode, ime ter avtor naprave. Po tem delu se pokaže prvi nivo menijske strukture. Opis menijev in delovanja pa je sledeč:

1. Izbira nadzora zunanje naprave:
  - s tipko + izberemo način nadzora ohišja fotoaparata;
  - s tipko - izberemo načina nadzora bliskavice;
  - s tipko OK se premaknemo v meni z nastavitvami naprave.
2. Izbira načina proženja ohišja fotoaparata. V koraku 1. je bila pritisnjena tipka + :
  - s tipko + izberemo način *intervalometer*;
  - s tipko - izberemo način *žično prožilo*;
  - s tipko OK se vrnemo nivo višje v menijski strukturi.
3. V koraku 2 je bil izbran način *intervalometer*. V tem nastavitvenem oknu s tipkami + in - izberemo število intervalov oz. število proženj zaklopa ohišja fotoaparata. Možne so vrednosti med 1 in 9999. S tipko OK potrdimo izbiro in se pomaknemo v naslednje nastavitveno okno (4).



Slika 6.1: Menijska struktura.

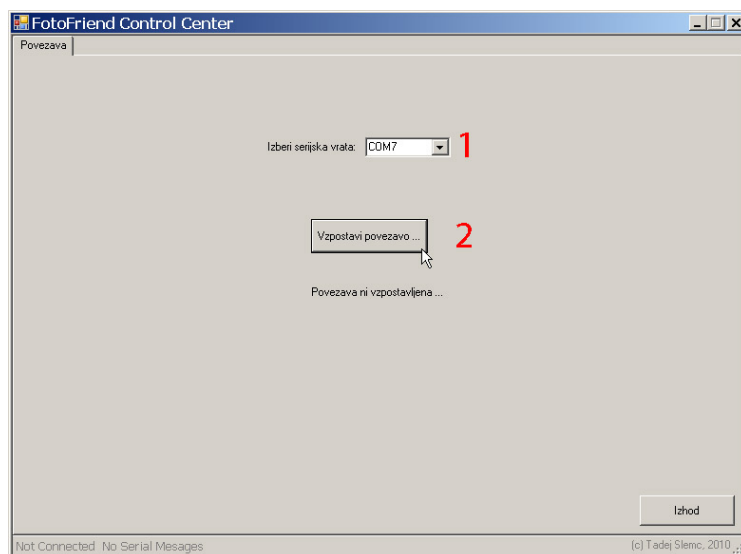
4. V nastavitvenem oknu izberemo zamik med intervali oz. zaporednimi proženji zaklopa ohišja fotoaparata. Zamik se izraža v sekundah. Možne so dolžine zamika od 1 do 9999 sekund. Izbiro potrdimo s tipko OK in se pomaknemo v statusno okno intervalometra (5). Ob tej izbiri se delovanje intervalometra zažene.
5. Statusno okno delovanja intervalometra. Na statusnem oknu je izpisan podatek o izbrani dolžini zamika med intervali ter preostalim številom intervalov. Ob vsakem proženju zaklopa ohišja fotoaparata se stanje spremeni. Ko se program izteče, se na tej točki zaustavi. Možen je ponoven zagon naprave s tipko RESET.
6. Izbira načina delovanja žičnega prožila. V koraku 2 je bil izbran način *žično prožilo*.
  - s tipko + izberemo način *ročno proženje*, kar pomeni da s pritiskom na tipko žičnega prožila sprožimo zaklop ohišja fotoaparata;
  - s tipko - izberemo način *BULB proženje*, kar pomeni da z enim pritiskom na tiko žičnega prožila sprožimo zaklop, proženje se zaklene in se konča šele ob naslednjem pritisku na tipko prožila;
  - s tipko OK se vrnemo en nivo nazaj po menijski strukturi.
7. Statusno okno žičnega prožila. Proženje zaklopa ohišja fotoaparata se izvede s pritiskom na tipko OK.
8. V koraku 1 je bila pritisnjena tipka -, torej je bil izbran način nadzora bliskavice. V tem koraku izberemo tip dogodka, ki bo vplival na proženje bliskavice.
  - s tipko + izberemo proženje bliskavice z zvokom. Kot tipalo moramo uporabiti mikrofona;
  - s tipko - izberemo proženje bliskavice s svetlobo. Kot tipalo moramo uporabiti analogno tipalo s fotouporom;
  - s tipko OK se vrnemo nivo višje v menijski strukturi.
9. V nastavitvenem oknu nastavimo mejno vrednost zvoka, ob presežku katere bo naprava prožila bliskavico. Za orientacijo je podan odčitek vrednosti ambientalnega zvoka. Možne so vrednosti med 0 in 1023. Nastavitev potrdimo s tipko OK.

10. V nastavitveno/statusnem oknu nastavimo zamik proženja bliskavice od časa zaznave prekoračitve nivoja zvoka, nastavljenega v prejšnjem koraku. Možne vrednosti zamika so od 0 do 9999 milisekund. Nastavitev potrdimo s tipko OK. S tem je naprava pripravljena na proženje. V zgornjem desnem kotu prikazovalnika se izpiše status ARMED in naprava čaka na dražljaj za proženje. Ko se proženje izvede, se status ARMED pobriše. Med proženji so možni popravki zamika proženja.
11. V koraku 8 je bil izbran način proženja s svetlobo. V tem koraku izberemo način delovanja
  - s tipko + izberemo način proženja bliskavico s prekinitvijo vira svetlobe. To pomeni, da bo naprava sprožila bliskavico v primeru, ko bo količina svetlobe na tipalu padla pod določeno mejo (npr. ob prekinitvi svetlobe laserskega žarka);
  - s tipko - izberemo način bliskavice *optični suženj*. To pomeni, da bo naprava sprožila bliskavico v primeru, ko količina svetlobe na tipalu preseže določeno mejo;
  - s tipko OK se vrnemo nivo višje v menijski strukturi.
12. V nastavitveno/statusnem oknu nastavimo zamik proženja bliskavice od časa zaznave prenizkega nivoja vpadne svetlobe na tipalo. Mejna vrednost se nastavi v nastavitvah (menijska okna 14 in 15). Možne vrednosti zamika so od 0 do 9999 milisekund. Nastavitev potrdimo s tipko OK. S tem je naprava pripravljena na proženje. V gornjem desnem kotu prikazovalnika se izpiše status ARMED in naprava čaka na dražljaj za proženje. Ko se proženje izvede se status ARMED pobriše. Med proženji so možni popravki zamika proženja.
13. Statusno okno načina *optični suženj*. Na prikazovalniku je izpisana izmerjena vrednost ambientalne svetlobe. V tem načinu naprava proži bliskavico ob povečanju nivoja vpadne svetlobe čez izmerjeno ambientalno vrednost. Ker je odziv zelo hiter, se ta način uporablja za proženje večih bliskavic naenkrat. Blisk bliskavice *gospodar* tako sproži še bliskavico *suženj* in rezultat je tak, kot da smo obe bliskavici sprožili istočasno.
14. V koraku 1 je bila pritisnjena tipka OK in izbrano je okno z izbiro nastavitvev:
  - s tipko + nastavimo spodnjo mejno vrednost za proženje bliskavice v načinu *prekinitve vira*;

- s tipko - nastavimo dolžno trajanja impulza za proženje zaklopa ohišja fotoaparata ter bliskavice.
15. Nastavitveno okno za nastavev meje, ki se uporablja pri načinu proženja bliskavice s prekinitvijo vira. Informativno je podana vrednost odčitka ambientalne svetlobe. Možne so vrednosti med 0 in 1023. Nastavev potrdimo s pritiskom na tipko OK.
  16. Nastavev je shranjena v pomnilnik EEPROM. Za nadaljevanje je potrebno pritisniti tipko RESET.
  17. Nastavitveno okno za nastavev dolžine trajanja impulza za proženje zaklopa ohišja fotoaparata ter bliskavice. Nastavev je pomembna, saj različni tipi ohišij fotoaparatorov različno odreagirajo na dolžino pulza. Primer: Canon EOS 350D pri trajanju impulza pod 100ms ne bo sprožil zaklopa, medtem ko ga recimo pri trajanju 150ms bo. Canon 1D MkIV pa ob trajanju impulza 150ms naredi vsaj dve proženji.
  18. Nastavev je shranjena v pomnilnik EEPROM. Za nadaljevanje je potrebno pritisniti tipko RESET.

### 6.2.2 Računalniško vodeni način delovanja

Napravo upravljamo s pomočjo programa, ki teče na osebnem računalniku. Upravljanje preko kontrolnih tipk na ohišju naprave je onemogočeno. Na prikazovalniku LCD je izpisano le sporočilo, ki nam pove, da uporabljamo ta način delovanja. Ob zagonu programa se pojavi okno, ki omogoča vzpostavitev povezave z napravo. Kot je vidno na sliki 6.2, to okno vsebuje dva pomembna elementa in sicer spustni meni (1), s pomočjo katerega izberemo serijska vrata, na katera je priključena naprava ter gumb za vzpostavitev povezave (2). Povezavo med napravo in uporabniškim vmesnikom lahko vzpostavimo takoj po zagonu naprave, ko je na prikazovalniku LCD izpisano pozdravno sporočilo. Če v tem času pritisnemo gumb za vzpostavitev povezave v uporabniškem vmesniku, bo povezava vzpostavljena in vsebina uporabniškega vmesnika se bo spremenila. Uporabniški vmesnik poskuša voditi uporabnika do končne izbire funkcijskega načina delovanja preko izbirnih menijev na način čarovnika za izbiro. V vsakem koraku se odločimo za eno od ponujenih možnosti, dokler ne pridemo do končne izbire funkcijskega načina delovanja ter statusno/nastavitvenega okna za izbrani način. Primer izbire: Nadzor bliskavice → Proženje z zvokom. Druga možnost je direktna izbira funkcijskega načina delovanja preko zavihkov v zgornjem delu uporabniškega vmesnika. Ker s pomočjo uporabniškega



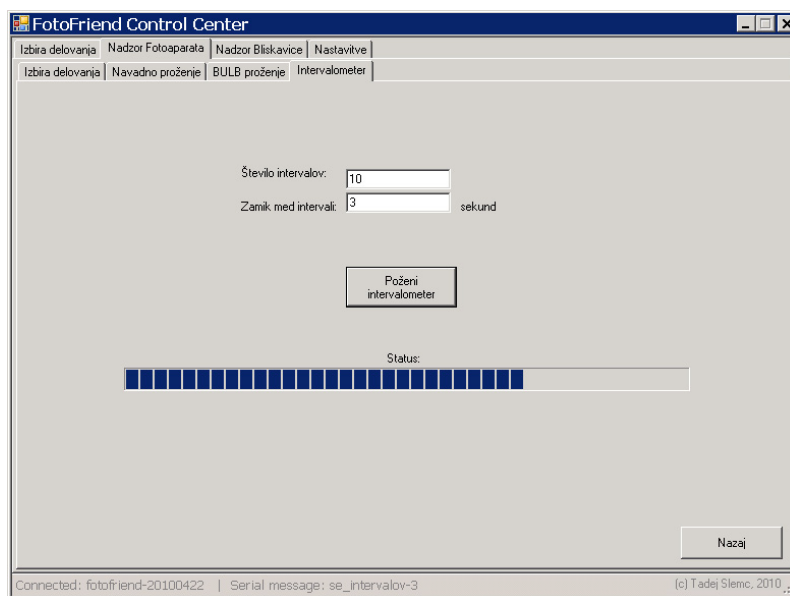
Slika 6.2: Okno uporabniškega vmesnika z nastavitvami za vzpostavitev računalniško vodenega načina delovanja.

vmesnika lahko nadzorujemo vse možne funkcijske načine delovanja, ki so že opisani v poglavju 6.2.1, je ponoven opis nepotreben. Kot primer je na sliki 6.3 prikazan uporabniški vmesnik z izbranim funkcijskim načinom delovanja *Intervalometer*.

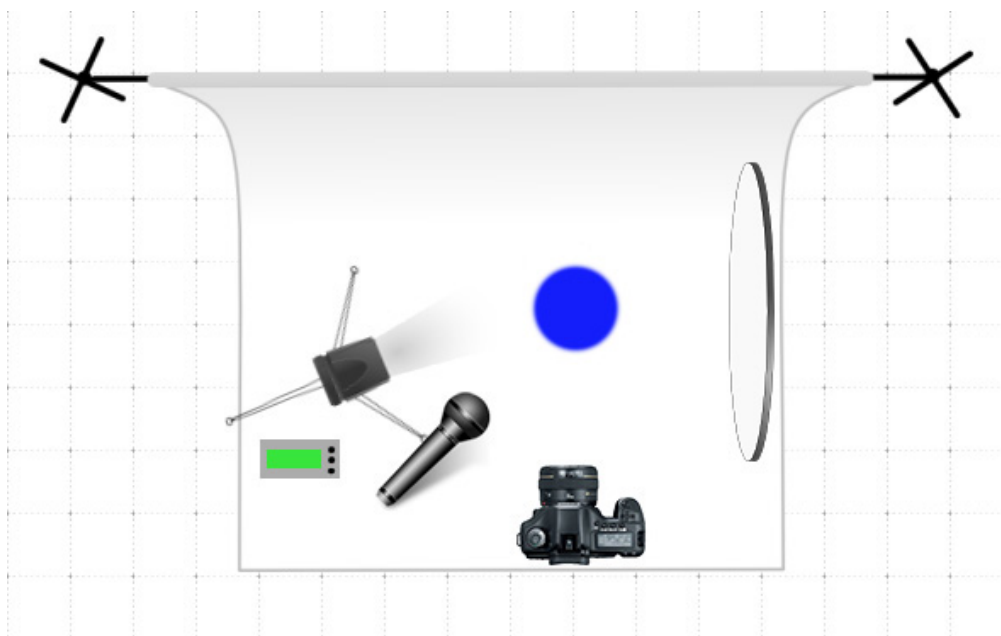
### 6.3 Primer uporabe

S pomočjo naprave lahko odreagiramo na različne dražljaje, odvisno od tipa dogodka, katerega želimo fotografirati in uporabljenega tipala. Kot primer je podano fotografiranje poka balona, napolnjenega z vodo.

Na sliki 6.4 je vidna postavitve elementov, potrebnih za nastanek fotografije. Pred poljubnim ozadjem je postavljen balon, ki je naš predmet fotografiranja. Na levi strani balona je postavljena bliskavica, ki je glavni in edini vir svetlobe. Ob bliskavici je postavljen mikrofonski senzor, ki služi kot tipalo za zaznavanje poka balona. Na desni strani balona je postavljen odbojnik svetlobe, ki poskrbi za osvetlitev balona še z desne strani. Bliskavico, ohišje fotoaparata in mikrofonski senzor priključimo na napravo. V napravi izberemo funkcijski način "proženje bliskavice s pomočjo zvoka". Izberemo ustrezno mejno vrednost, ob kateri bo naprava odreagirala in ustrezen zamik proženja. Na ohišju fotoapa-



Slika 6.3: Uporabniški vmesnik z izbranim funkcijskim načinom delovanja *Intervalometer*.



Slika 6.4: Diagram postavitve elementov za fotografiranje poka balona napolnjenega z vodo.

rata izberemo programski način M (Manual). Za nastavitve beline uporabimo toplotno vrednost svetlobe bliskavice. Izberemo ustrezno vrednost ISO ter vrednost zaslonke. Hitrost zaklopa nastavimo na vrednost BULB (za dolžino proženja bo v celoti skrbela naprava). Na bliskavici izberemo programski način M ter nastavimo moč ter širino snopa bliska. Poskrbimo, da je v prostoru, kjer slikamo, popolna tema. V nastavitvah naprave potrdimo vse izbrane možnosti in pričnemo s slikanjem. Naprava sproži zaklop ohišja fotoaparata ter čaka na zvočni dražljaj, ki bo presegel nastavljeno vrednost. Ko se dražljaj pojavi, ob puku balona, naprava sproži bliskavico, počaka 100ms ter prekine proženje ohišja fotoaparata. Zaklop se zapre in slika je zajeta. Rezultat slikanja balona je viden na sliki A.14.

# Poglavje 7

## Sklepne ugotovitve

Izdelana naprava zadostuje vsem uvodoma podanim zahtevam. Hitrost reagiranja na zunanji dražljaj je zadovoljiva, saj lahko fotografiramo različne stopnje poka balona napolnjenega z vodo, kot je vidno v prilogi A.14. Omogoča nastavljanje zamika od zaznave dražljaja do proženja, od 0 do 9999 milisekund, v korakih po 1 milisekundo. Za zaznavo dražljajev bi lahko uporabili praktično kakršnokoli analogno ali digitalno tipalo pri pogoju, da bi tipalu priredili programski del naprave. Zaenkrat naprava podpira dve vrsti tipal. Prvo je analogno tipalo svetlobne jakosti, ki se priklopi na splošni vhod. Kot posebno tipalo lahko uporabimo tipalo jakosti zvoka oz. mikrofona, saj je v napravo vgrajen ojačevalnik signala mikrofonskega vhoda, priklopi pa se na poseben vhod. Upravljanje naprave je enostavno in nedvoumno bodisi v samostojnem načinu, bodisi v računalniško vodenem načinu delovanja. Poleg uporabe v visokohitrostni fotografiji, je naprava lahko uporabna tudi pri vsakodnevnem fotografiranju, saj se lahko uporabi kot intervalometer, žično prožilo ali pa kot tipalo za sinhronizacijo bliskavic v načinu optični suženj.

Kljub temu da naprava zadostuje vsem zahtevam podajam še nekaj idej za morebitni nadaljnji razvoj:

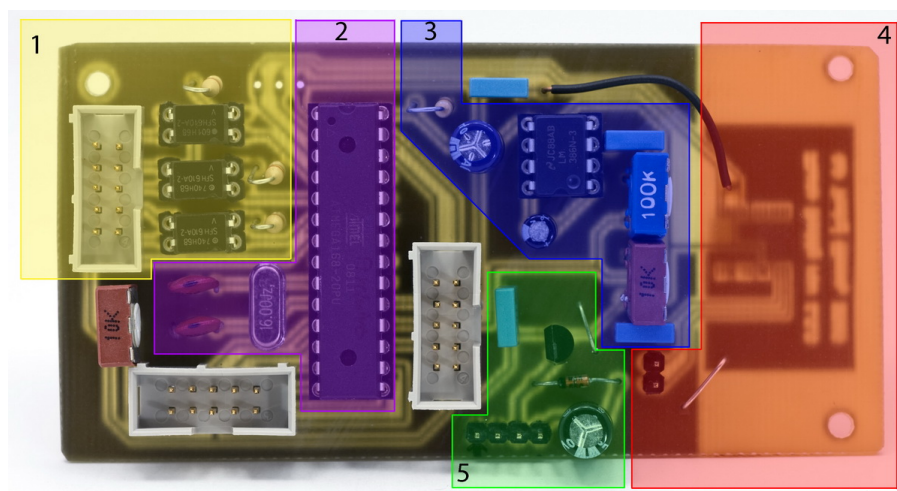
- splošna podpora različnim tipalom brez potrebe po spreminjanju programske kode naprave;
- uporaba več tipal naenkrat;
- podpora upravljanju dodatnih zunanjih naprav kot so npr. razni elektromagnetni ventili ali koračni motorji;
- izpuščanje dogodkov;

- podpora manjšim grafičnim prikazovalnikom, ki bi nadomestili 4x20 znakovni prikazovalnik. S tem bi se velikost naprave bistveno zmanjšala;
- zamenjava uporabljenega mikrokrmilnika za mikrokrmilnik Atmel serije AT32UC3. S to zamenjavo bi se zaradi tipa pakiranja mikrokrmilnika sicer zapletla izdelava tiskanega vezja, bi pa precej pridobili pri hitrosti delovanja (namesto 16MHz bi vezje lahko delovalo na hitrostih do 66MHz). Ta mikrokrmilnik ima že vgrajeno podporo za protokol USB, kar pomeni da ne bi bilo potrebno uporabljati vezja FTDI FT232R ter bi lahko vsa komunikacija potekala preko protokola USB in ne preko navideznega serijskega vmesnika. Možna bi bila tudi implementacija protokola PTP (Picture Transfer Protocol) [10] z uporabo katerega bi lahko preko vmesnika USB upravljali z vsemi nastavitvami fotoaparata, vključno s proženjem samega zaklopa;
- razvoj novih tipal.

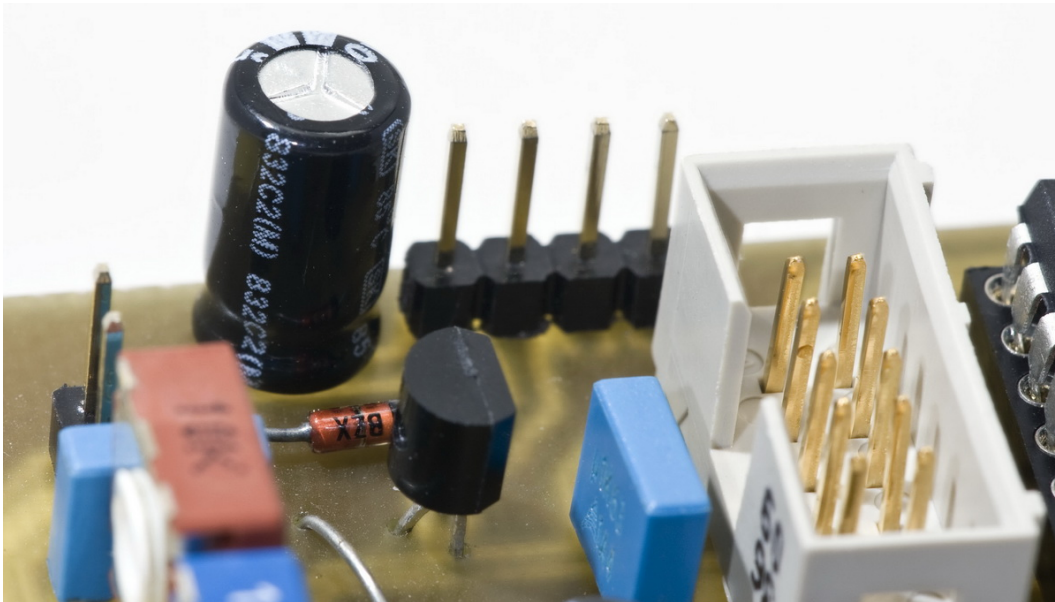
# Dodatek A

## Priloge

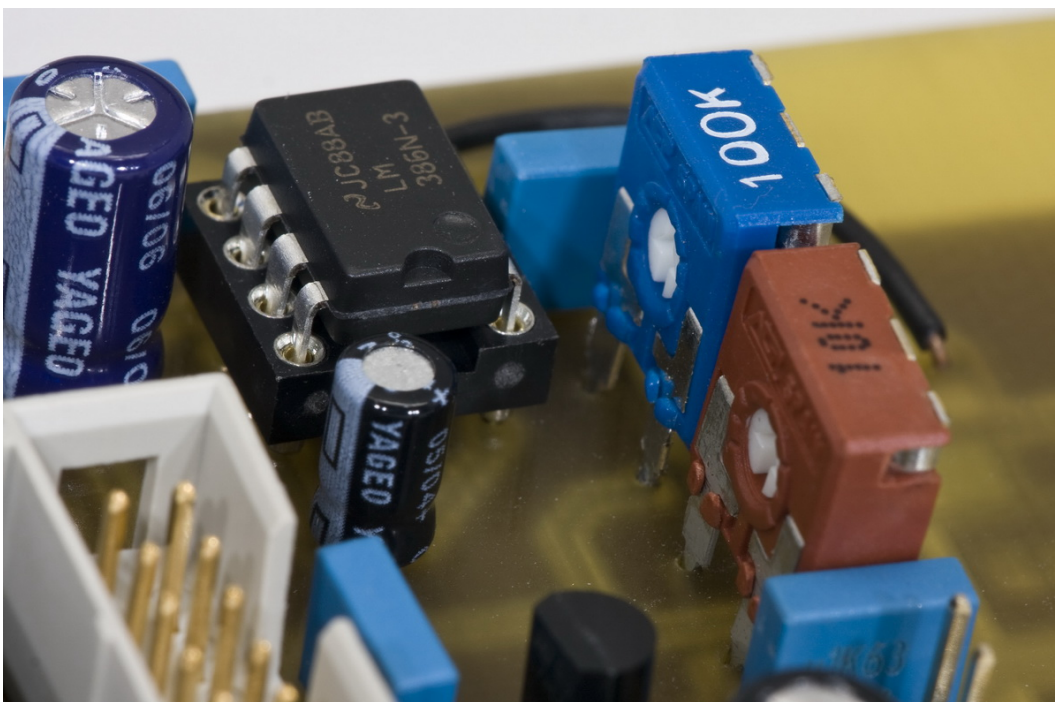
### A.1 Slike naprave



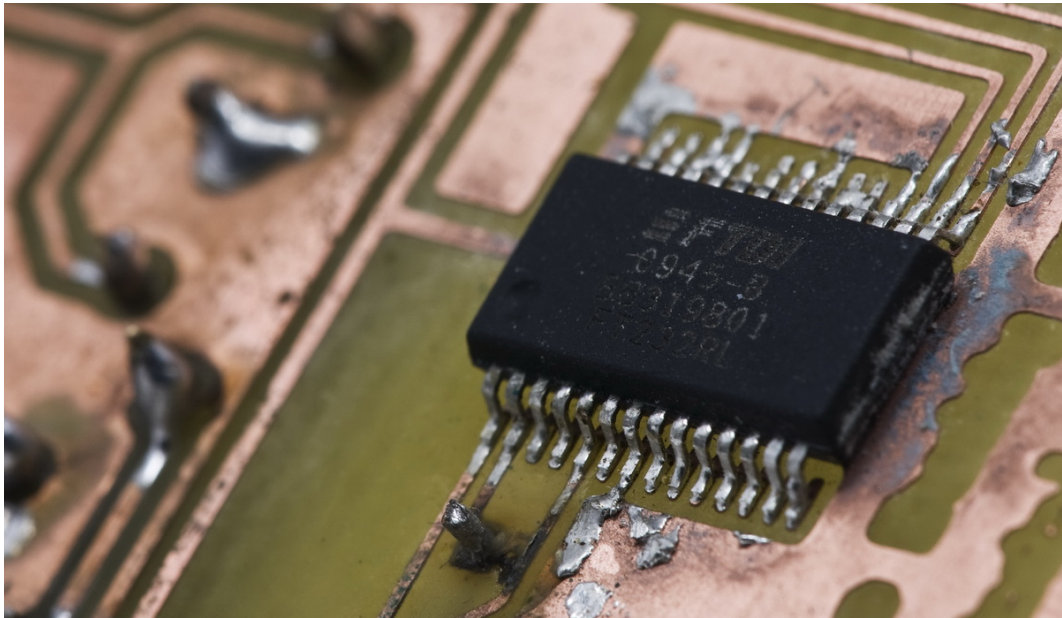
Slika A.1: Osnovna plošča z označenimi sklopi. Rumena (1): izhodni del, vijolična (2): kontrolni del, modra (3): predajačevalnik mikrofonkega vhoda, rdeča (4): povezovalni del, zelena (5): napajalni del.



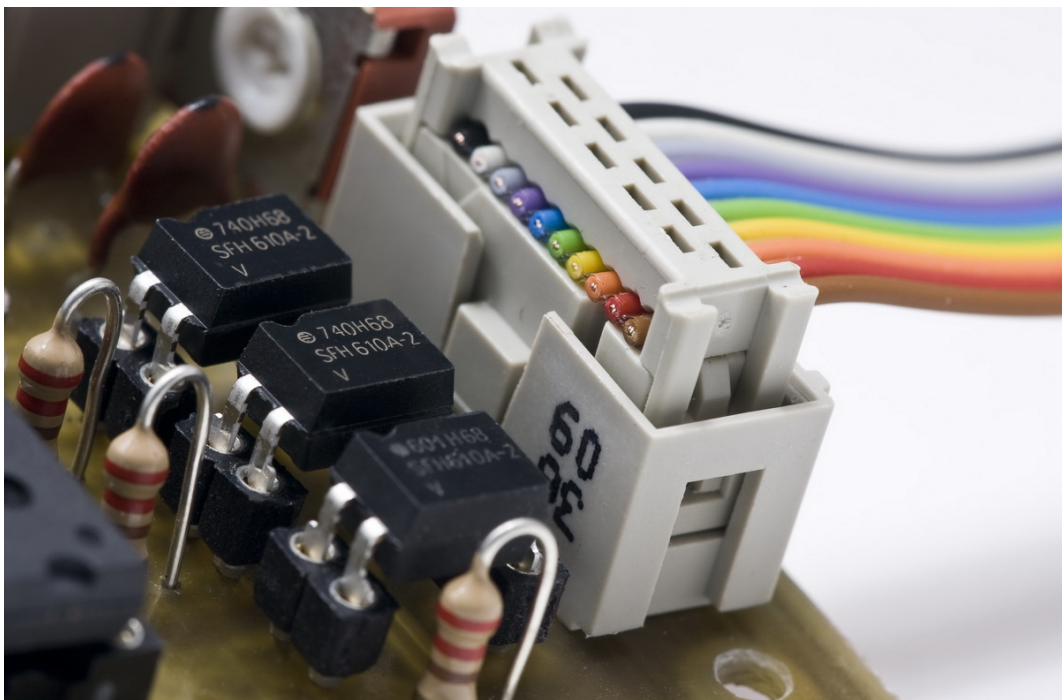
Slika A.2: Podrobnosti napajalnega dela vezja.



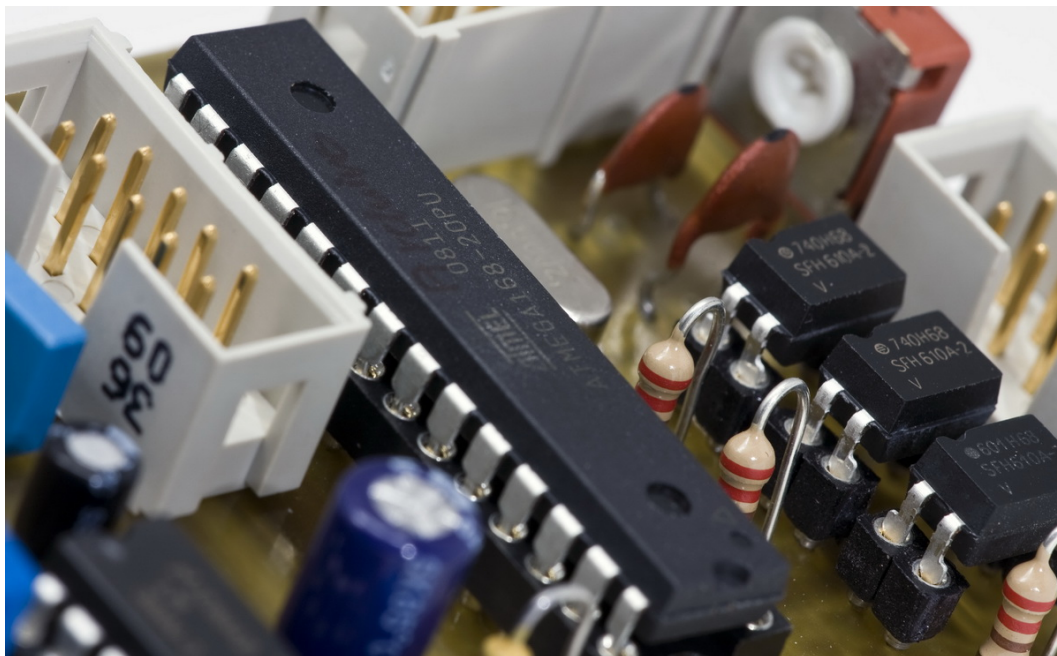
Slika A.3: Podrobnosti predojačevalnika signala mikrofonskega vhoda.



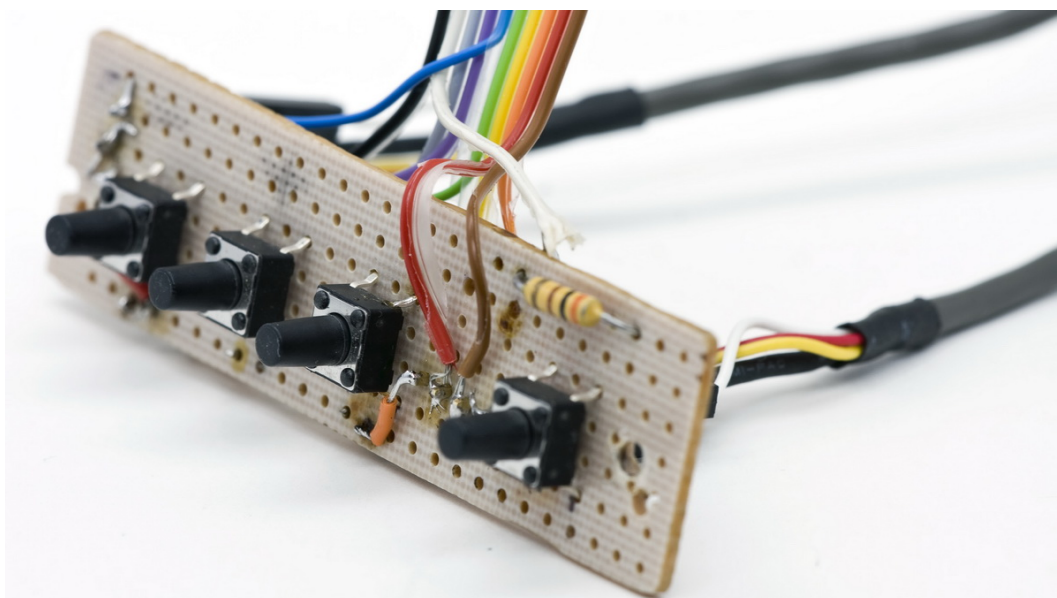
Slika A.4: Podrobnosti povezovalnega dela vezja.



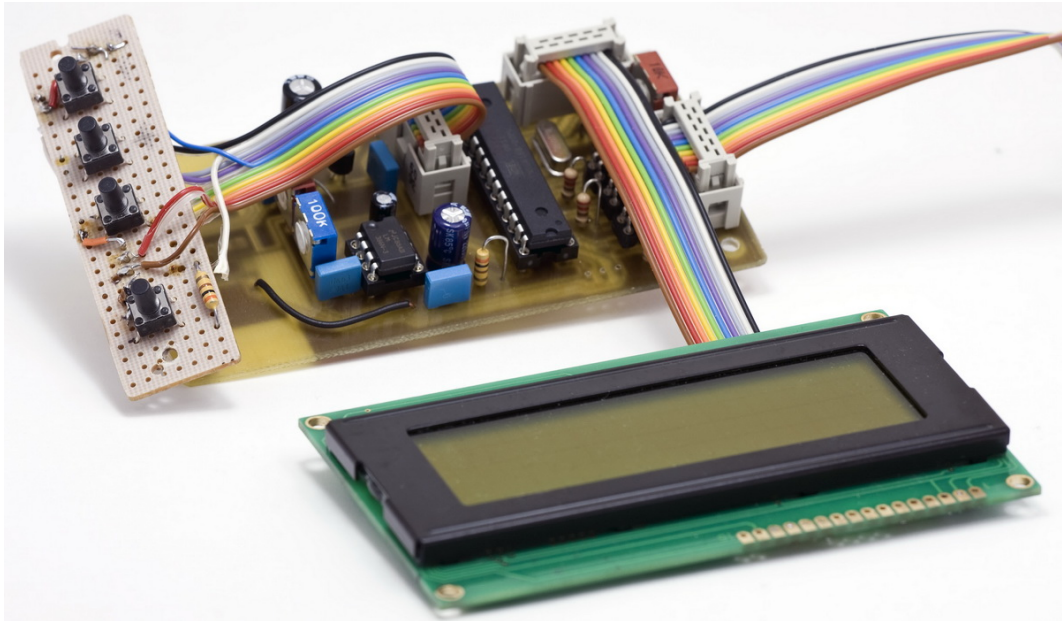
Slika A.5: Podrobnosti izhodnega dela vezja.



Slika A.6: Podrobnosti kontrolnega dela vezja z mikrokontrolnikom Atmel AT-Mega168.



Slika A.7: Podrobnosti vezja s kontrolnimi tipkami.



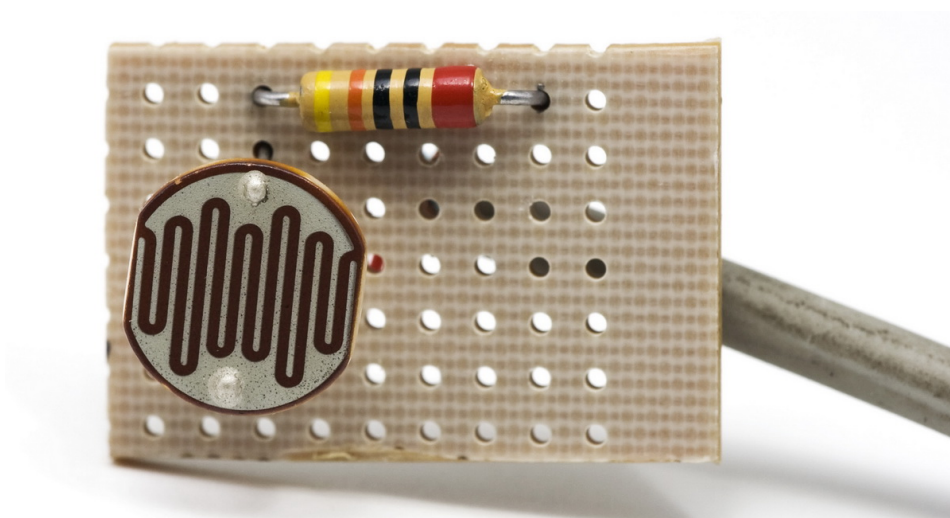
Slika A.8: Slika celotnega vezja.



Slika A.9: Slika naprave.



Slika A.10: Slika priključkov na napravi. Od leve proti desni si v zgornji vrsti sledijo: izhod za proženje fotoaparata, vhod USB za povezovanje z osebnim računalnikom ter priključek zunanje napajanja, splošni priključek tipal. V spodnji vrsti: izhod za proženje bliskavice, mikrofonski vhod.



Slika A.11: Podrobnosti vezja zunanega analognega svetlobnega tipala.

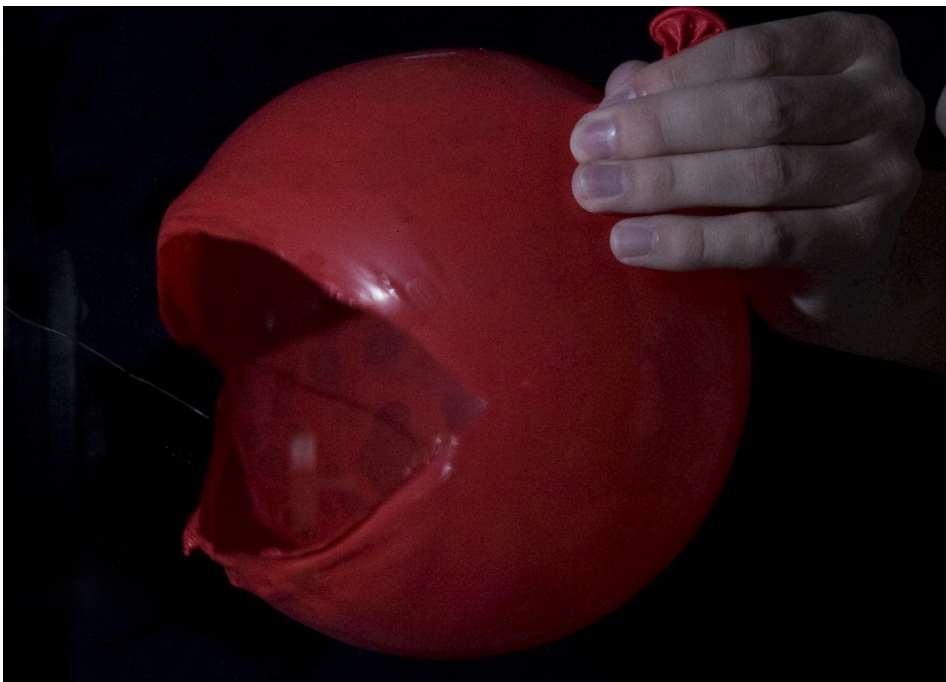
## A.2 Slike narejene s pomočjo naprave



Slika A.12: Slika poka balona napolnjenega z vodo. V nastavitvah naprave ni bilo uporabljenega nobenega zamika.



Slika A.13: Slika poka balona napolnjenega z vodo. Zamik med zaznavo dogodka ter proženjem je 2ms.



Slika A.14: Slika poka balona napolnjenega z zrakom. V nastavitvah naprave ni bilo uporabljenega nobenega zamika.



Slika A.15: Slika sekvence padanja vodne kapljice v kozarec napolnjen z vodo. Ob vsaki sličici je zapisan uporabljen zamik nastavljen v napravi.



# Slike

3.1	Logična shema vezja. . . . .	14
3.2	Električna shema napajalnega dela vezja. . . . .	15
3.3	Električna shema predojačevalnika signala mikrofonskega vhoda. . . . .	16
3.4	Električna shema povezovalnega dela vezja. . . . .	17
3.5	Električna shema izhodnega dela vezja. . . . .	19
3.6	Razpored nožic mikrokrmilnika Atmel ATMega186 v ohišju DIP. . . . .	21
3.7	Električna shema kontrolnega dela vezja. . . . .	22
3.8	Električna shema vezja s kontrolnimi tipkami. . . . .	23
3.9	Električna shema zunanlega analognega svetlobnega tipala. . . . .	24
5.1	Razdelitev uporabniškega vmesnika . . . . .	40
6.1	Menijska struktura. . . . .	44
6.2	Uporabniški vmesnik z nastavitvami komunikacije. . . . .	48
6.3	Uporabniški vmesnik načina Intervalometer. . . . .	49
6.4	Diagram postavitve elementov za fotografiranje. . . . .	49
A.1	Osnovna plošča z označenimi sklopi. . . . .	53
A.2	Podrobnosti napajalnega dela vezja. . . . .	54
A.3	Podrobnosti predojačevalnika signala mikrofonskega vhoda. . . . .	54
A.4	Podrobnosti povezovalnega dela vezja. . . . .	55
A.5	Podrobnosti izhodnega dela vezja. . . . .	55
A.6	Podrobnosti kontrolnega dela vezja. . . . .	56
A.7	Podrobnosti vezja s kontrolnimi tipkami. . . . .	56
A.8	Slika celotnega vezja. . . . .	57
A.9	Slika naprave. . . . .	57
A.10	Slika priključkov na napravi. . . . .	58
A.11	Podrobnosti vezja zunanlega analognega svetlobnega tipala. . . . .	58
A.12	Slika poka balona napolnjenega z vodo. . . . .	59
A.13	Slika poka balona napolnjenega z vodo. . . . .	60

A.14 Slika poka balona napolnjenega z zrakom. . . . .	60
A.15 Slika sekvence padanja vodne kapljice. . . . .	61

# Tabele

2.1 Dolžina trajanja bliska . . . . .	10
---------------------------------------	----



# Literatura

- [1] (2010) Shutter lag. Dostopno na:  
[http://en.wikipedia.org/wiki/Shutter\\_lag](http://en.wikipedia.org/wiki/Shutter_lag)
- [2] (2010) Autofocus. Dostopno na:  
<http://en.wikipedia.org/wiki/Autofocus>
- [3] (2010) Canon Speedlite 580EX Flash Durations. Dostopno na:  
<http://www.waynesthisandthat.com/flashdurations.html>
- [4] (2010) LM386 Low Voltage Audio Power Amplifier datasheet Dostopno na:  
<http://www.national.com/ds/LM/LM386.pdf>
- [5] (2010) FTDI FT232R USB UART IC datasheet Dostopno na:  
[http://www.ftdichip.com/Documents/DataSheets/DS\\_FT232R.pdf](http://www.ftdichip.com/Documents/DataSheets/DS_FT232R.pdf)
- [6] (2010) Atmel ATmega168 Datasheet. Dostopno na:  
[http://www.atmel.com/dyn/resources/prod\\_documents/doc2545.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2545.pdf)
- [7] (2010) Arduino bootloader Dostopno na:  
<http://www.arduino.cc/en/Main/Bootloader>
- [8] (2010) Programming an atmega using Arduino Dostopno na:  
<http://ntsdt.net/2009/08/07/programming-an-atmega8-using-arduino/>
- [9] (2010) Wiring Port Manipulation. Dostopno na:  
<http://www.arduino.cc/en/Reference/PortManipulation>
- [10] E. Steinberg, T. Looney, T. Whitcher, idr., *PIMA 15740:2000: Photography – Electronic still picture imaging - Picture Transfer Protocol (PTP) for Digital Still Photography Devices*. Dostopno na:  
<http://www.brooms closet.com/closet/photo/exif/PIMA15740-2000.PDF>