

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**KATEDRA ELEKTROMECHANIKY A VÝKONOVÉ  
ELEKTRONIKY**

# **BAKALÁŘSKÁ PRÁCE**

**Zařízení pro měření reálného času s mikrokontrolérem**

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta elektrotechnická

Akademický rok: 2014/2015

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Tomáš KUDERA**  
Osobní číslo: **E12B0032P**  
Studijní program: **B2612 Elektrotechnika a informatika**  
Studijní obor: **Elektrotechnika a energetika**  
Název tématu: **Zařízení pro měření reálného času s mikrokontrolérem**  
Zadávací katedra: **Katedra elektromechaniky a výkonové elektroniky**

### Z á s a d y p r o v y p r a c o v á n í :

1. Zvolte vhodný typ mikrokontroléru pro aplikaci časomíry, tj. zařízení pro měření času průchodu světelnou závorou (fotobuňkou).
2. Navrhněte a sestrojte hardwarovou část zařízení pro snímání času průchodu předmětu nebo osoby světelnou závorou.
3. Navrhněte software, který umožní zápis změřených časů do PC pro případné další zpracování.
4. Navržené a sestrojené zařízení vč. softwarového vybavení otestujte na správnou funkci.

Rozsah grafických prací: podle doporučení vedoucího

Rozsah pracovní zprávy: 20 - 30 stran

Forma zpracování bakalářské práce: tištěná/elektronická

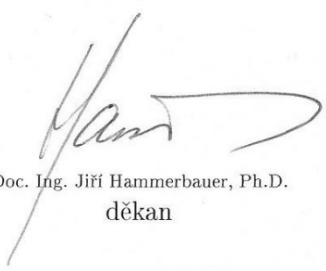
Seznam odborné literatury:

Student si vhodnou literaturu vyhledá v dostupných pramenech podle doporučení vedoucího práce.

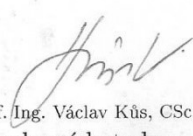
Vedoucí bakalářské práce: Ing. Tomáš Glasberger, Ph.D.  
Regionální inovační centrum elektrotechniky

Datum zadání bakalářské práce: 15. října 2014

Termín odevzdání bakalářské práce: 8. června 2015

  
Doc. Ing. Jiří Hammerbauer, Ph.D.  
děkan



  
Prof. Ing. Václav Kůs, CSc.  
vedoucí katedry

V Plzni dne 15. října 2014

## **Abstrakt**

Předkládaná bakalářská práce se zabývá návrhem softwarové i hardwarové části zařízení pro snímání času průchodu předmětu nebo osoby světelnou závorou. V první části práce je věnována pozornost základním funkčním blokům zařízení. Druhá část práce se zabývá návrhem elektronické, optické a mechanické části zařízení. Ve třetí části práce je pozornost zaměřena na analýzu a popis vytvořeného software pro mikroprocesor umožňujícího odesílání časů do PC přes USB a také na popis vytvořeného software pro PC sloužícího ke zpracování přijatých časů.

## **Klíčová slova**

Světelná závora, měření času, časomíra, laser, mikroprocesor, UART, USB, Texas Instruments, MSP430, Launchpad, Python

## **Abstract**

The bachelor thesis deals with the design of both software and hardware part of a device which is intended for capturing the time of human or item gone through a light barrier. In the first part of the thesis the attention is paid to elementary blocks of the device. The second part of the thesis deals with the design of electronic, optical and mechanical parts of the device. In the third part of the thesis the attention is focused on the analysis and description of created software for the microprocessor which allows transmitting times to PC over USB and also on the description of the software designed for PC which is used for processing of received times.

## **Key words**

Light barrier, time measurement, timekeeping, laser, microprocessor, UART, USB, Texas Instruments, MSP430, Launchpad, Python

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské práce, je legální.

.....  
podpis

V Plzni dne 1.6.2015

Tomáš Kudera

## **Poděkování**

Tímto bych rád poděkoval vedoucímu mé bakalářské práce Ing. Tomáši Glasbergerovi, Ph.D., za ochotně poskytnuté cenné rady a připomínky, které mi pomohly s tvorbou práce. Děkuji také za poskytnutí stipendia za aktivní studium na KEV FEL ZČU, v rámci kterého jsem tvořil tuto práci.

# Obsah

ÚVOD .....	9
SEZNAM SYMBOLŮ A ZKRATEK .....	10
<b>1 BLOKOVÉ SCHÉMA .....</b>	<b>12</b>
1.1 POPIS JEDNOTLIVÝCH BLOKŮ .....	12
1.1.1 Světelná závora .....	12
1.1.2 Napájení .....	14
1.1.3 Detektor světla .....	15
1.1.4 Komparátor .....	16
1.1.5 Mikroprocesor .....	18
1.1.6 UART/USB převodník .....	18
<b>2 NÁVRH HARDWARE .....</b>	<b>20</b>
2.1 VYSÍLAČ .....	20
2.2 PŘIJÍMAČ .....	22
2.2.1 Optika .....	22
2.2.2 Mikroprocesor .....	24
2.2.3 Periferie .....	27
2.3 REALIZACE .....	32
2.3.1 Návrh DPS .....	32
2.3.2 Konstrukce .....	34
<b>3 NÁVRH SOFTWARE .....</b>	<b>36</b>
3.1 PROGRAM MIKROPROCESORU .....	36
3.1.1 Náhled na funkci .....	36
3.1.2 Nastavení registrů .....	38
3.1.3 Odesílání časů .....	40
3.1.4 Využití programu Putty pro komunikaci .....	41
3.2 PC PROGRAM (TIME TERMINAL) .....	43
3.2.1 Náhled na funkci .....	43
3.2.2 Vývojový diagram .....	44
3.2.3 Ukázka použití aplikace Time Terminal .....	49
<b>ZÁVĚR .....</b>	<b>51</b>
<b>SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ .....</b>	<b>53</b>
<b>PŘÍLOHY .....</b>	<b>1</b>



## Úvod

Tato bakalářská práce je zaměřena na realizaci zařízení, které umožňuje zaznamenávat časy průchodu osob, případně předmětů světelnou závorou a tyto časy odesílat do počítače. Průchod osoby nebo předmětu se zaznamenává pomocí světelné závory. Světelnou závoru zde tvoří dvě optoelektronické jednotky: vysílač a přijímač. Vysílačem je zdroj světla ve formě úzkého svazku (paprsku). Úkolem přijímače je bezprostředně po přerušení světelného paprsku odeslat časový údaj do počítače, ve kterém se následně zaznamená, případně dále zpracuje. Zařízení by mělo být schopno provozu v různorodém průmyslovém prostředí nebo v terénu, např. při běžeckých či jiných závodech. Z toho plynou nároky na kompaktnost, odolnost a schopnost provozu na baterie. Zařízení má být koncipováno primárně pro účely běžeckých závodů, čemuž odpovídají i určité speciální funkce, např. intervalové starty, nabízí se však i jiná možná užití, např. zaznamenávání počtu a časů prošlých výrobků. Měřený čas je pro účely běžeckých a jiných závodů třeba zaznamenávat s přesností na setiny sekundy.

Nejdůležitějším prvkem přijímače je mikroprocesor, který musí být vybaven rozhraním UART pro obousměrnou komunikaci s počítačem. Počítač odesílá povely ke startu, případně k vynulování času a mikroprocesor odesílá časové údaje kdykoli při přerušení paprsku. Časovač mikroprocesoru by měl být pro tyto účely řízen přesným krystalovým oscilátorem. Mikroprocesor by měl mít dostatek vstupně/výstupních pinů pro další periferie, jako jsou signalizační diody, LCD modul pro zobrazování stavu a aktuálního času, nebo piezoreproduktor pro zvukovou signalizaci při intervalových startech. Zařízení může být ovládáno pomocí jakéhokoli terminálového programu, který podporuje sériovou komunikaci. Pro náročnější užití byla navržena a vyvinuta uživatelská aplikace Time Terminal, která přináší další možnosti pro práci s přijatými časy.

Hlavní cíle práce:

- 1) Návrh celkové koncepce a elektronické části zařízení
- 2) Návrh optické a mechanické části zařízení
- 3) Návrh uživatelského software
- 4) Realizace a otestování zařízení

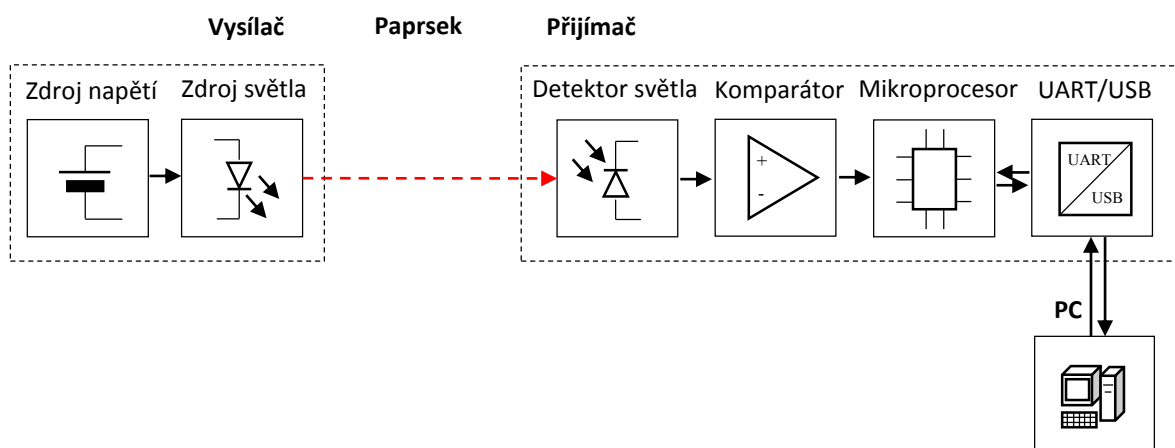
## Seznam symbolů a zkratk

ADC		Analog-to-digital converter
ASCII		American Standard Code for Information Interchange
COM		Communication port
CCS		Code composer studio
DIP		Dual in-line package
DPS		Deska plošných spojů
E	(mW/cm <sup>2</sup> )	Intenzita osvětlení
GPIO		General-purpose input/output
I <sub>2</sub> C		Inter-Integrated Circuit
I <sub>F</sub>	(mA)	Maximální proud v propustném směru
I <sub>R0</sub>	(nA)	Proud v závěrném směru při E = 0 mW/cm <sup>2</sup>
I <sub>RA</sub>	(μA)	Proud v záv. směru při E = 1 mW/cm <sup>2</sup> , λ = 950 nm, U <sub>R</sub> = 5 V
IrDA		Infrared Data Association
LCD		Liquid-crystal display
LED		Light-emitting diode
MCU		Microcontroller unit
OS		Operating system
P <sub>O</sub>	(mW)	Maximální optický výkon
PIN		Positive-intrinsic-negative
RAM		Random-access memory
RISC		Reduced instruction set computing
RXD		Receive data
SMD		Surface mount device
SPI		Serial Peripheral Interface
THT		Through-hole technology
TI		Texas instruments
T <sub>R</sub>	(ns)	Doba reakce na změnu osvětlení
T <sub>SL</sub>	(h)	Minimální doba životnosti
TXD		Transmit data
UART		Universal asynchronous receiver/transmitter

$U_0$	(V)	Napětí naprázdno při $E = 1 \text{ mW/cm}^2$ , $\lambda = 950 \text{ nm}$ ,
$U_{CC}$	(V)	Napětí zdroje
$U_F$	(V)	Napětí na diodě v propustném směru
USB		Universal serial bus
USCI		Universal serial communication interface
$\lambda$	(nm)	Vlnová délka vyzářeného světla

# 1 Blokové schéma

V této části jsou stručně popsány části (bloky), ze kterých by se navrhované zařízení mělo skládat. Pozornost je zde věnována zásadním částem, které jsou nutné pro chod celého zařízení. Blokové schéma zařízení je patrné na Obr. 1.

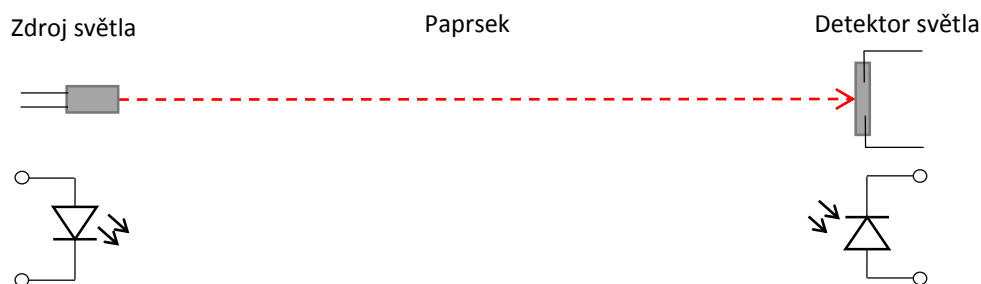


Obr. 1: Blokové schéma zařízení

## 1.1 Popis jednotlivých bloků

### 1.1.1 Světelná závora

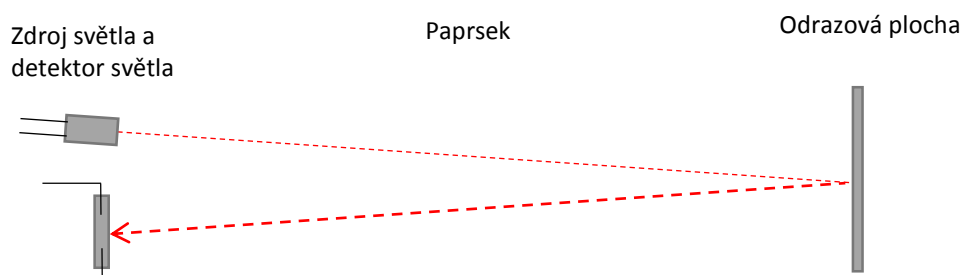
Základním prvkem celého zařízení je světelná závora, optoelektronický prvek, který se skládá ze zdroje světla a z detektoru světla. Zdrojem světla bývá nejčastěji LED dioda, případně laserová dioda. Jako detektoru světla se nejčastěji používá fotodiody, případně fototranzistoru. Světelné závory je možné provozovat ve viditelném i v infračerveném spektru, v závislosti na použitých optoelektronických součástkách. Vždy ale platí, že zdroj světla i detektor světla musí pracovat na stejné vlnové délce a detektor musí být dostatečně citlivý na vlnovou délku světla zdroje. Jednoduchou světelnou závora je možné vidět na Obr. 2.



Obr. 2: Prvky optické jednocestné závory a jejich obvyklé schématické značky

Dopadá-li světlo na fotocitlivou plochu detektoru, prochází jím proud. Pokud dojde k přerušení paprsku osobou nebo předmětem, detektor světla je neosvětlen a proud jím v ideálním případě neprochází. Tímto způsobem dochází k detekci průchodu osob nebo předmětů.

Světelné závory se nejčastěji používají jako bezpečnostní prvky v průmyslu, případně pro zaznamenávání počtu prošlých výrobků na výrobním páse. Lze je použít i pro zabezpečení nejrůznějších objektů, např. nemovitostí. Používají se i takzvané reflexní závory, které sdružují zdroj světla i detektor v jednom modulu a využívají odrazu paprsku na protější straně. Princip reflexní závory je naznačen na Obr. 3.



Obr. 3: Prvky reflexní závory a její princip

### 1.1.2 Napájení

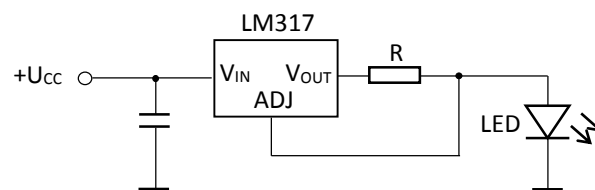
Vysílač světelné závory je možné napájet z baterií, případně ze síťového napětí. V prvním případě není třeba použít usměrňovač a je možné LED diodu nebo laserovou diodu připojit k baterii pouze s předřadným odporem. Velikost předřadného odporu je možné určit ze vztahu:

$$R = \frac{U_{cc} - U_F}{I_F}, \quad (\text{rov.1})$$

kde  $U_{cc}$  je napětí stejnosměrného zdroje (baterie)  $U_F$  a  $I_F$  je napětí a proud diodou v propustném směru.

Úbytek napětí na diodě je závislý především na vlnové délce emitovaného světla. Maximální hodnota  $I_F$  závisí na příkonu diody. Proud diodou se zpravidla volí menší, než je maximální dovolená katalogová hodnota.

Pokud je zdroj světla napájen z baterie, může docházet vlivem jejího vybíjení ke snižování napětí baterie. Tím podle Ohmova zákona klesá i proud diodou. Při požadavku na konstantní proud diodou lze použít například obvodu LM317 v zapojení proudového zdroje.



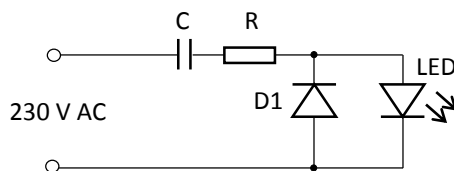
Obr. 4: Stabilizátor LM317 jako zdroj proudu [1]

Proud dodávaný do zátěže je v tomto zapojení roven:

$$I = \frac{1,25}{R}, \quad (\text{rov.2})$$

kde  $R$  je velikost odporu mezi vývody OUT a ADJ .

Pro napájení zdroje světla ze sítě je možné použít klasický transformátor s můstkovým usměrňovačem, případně spínaný zdroj. Existují také mnohá úsporná řešení, za zmínku stojí například řešení naznačené na Obr. 5.



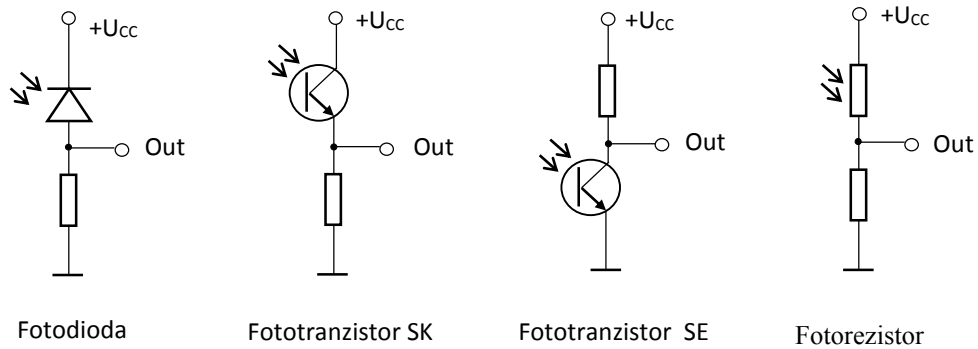
Obr. 5: Úsporné řešení pro připojení LED diody k síti 230 V

Proud procházející LED diodou v kladné půlperiodě je omezen kapacitní reaktancí kondenzátoru. Rezistor R slouží k omezení proudové špičky při připojení vybitého kondenzátoru k síti. Protože proud kondenzátorem musí procházet obousměrně, je nutné zařadit antiparalelně k LED diodě diodu D1. Ta také slouží k omezení napěťového namáhání LED diody při záporné půlperiodě. [2]

### 1.1.3 Detektor světla

Na místo detektoru přerušení světelné závory je možné použít fotodiodu nebo fototranzistor, v některých případech také fotorezistor. Fotodioda se v těchto aplikacích zpravidla zapojuje v závěrném směru, přičemž velikost závěrného proudu je úměrná intenzitě osvětlení fotodiody. K fotodiodě je sériově připojen rezistor, na němž vlivem procházejícího proudu vzniká úbytek napětí, jehož velikost je snímána a dále vyhodnocována. Fototranzistor je možné použít v zapojení se společným kolektorem, případně se společným emitorem. Při nulové intenzitě osvětlení je u zapojení se společným kolektorem na výstupu ideálně nulové napětí, což odpovídá pozitivní logice. Naproti tomu u zapojení se společným emitorem je na výstupu napětí  $+U_{cc}$ , což odpovídá negativní logice. Při maximální intenzitě osvětlení je situace inverzní.

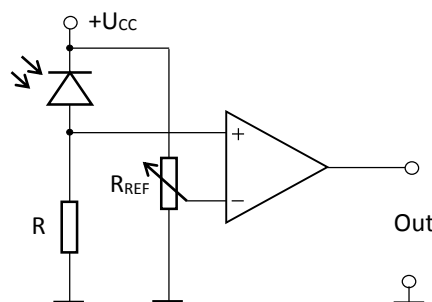
Teoreticky by také bylo možné pro tuto aplikaci použít fotorezistor, který s rostoucí intenzitou osvětlení snižuje svůj odpor. Fotorezistor má ale v porovnání s fotodiodou a fototranzistorem obvykle menší citlivost na světlo a pomaleji reaguje na změny intenzity osvětlení [3]. Z toho důvodu se hodí méně pro použití ve světelné závoře.



Obr. 6: Obvyklá zapojení fotocitlivých polovodičových součástek

### 1.1.4 Komparátor

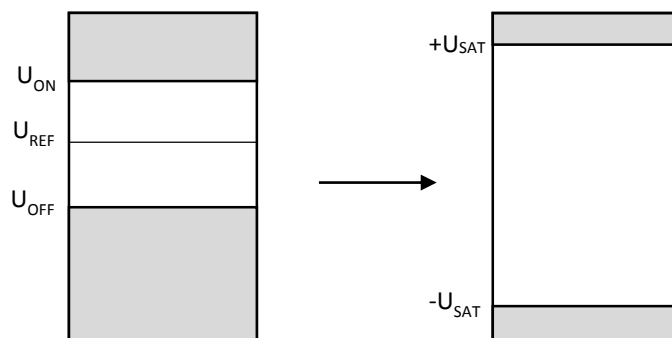
Světelné závory se často provozují v prostředích osvětlených od jiných zdrojů světla, např. Slunce. V tomto případě jsou další zdroje světla považovány za zdroje rušení a je třeba počítat s jejich vlivem, díky němuž nikdy nebude napětí na rezistoru za fotodiodou nulové. Mohlo by totiž dojít k tomu, že napětí na vstupu logického obvodu, který zpracovává vstupní signál, překročí pro neosvětlený stav rozhodovací úroveň a nebude tak možné rozpoznat osvětlenou nebo neosvětlenou fotodiodu. Jedním ze způsobů, jak rozpoznat rušení od světla ze zdroje je použití komparátoru, například s operačním zesilovačem. Výstupní napětí díky tomu bude ideálně nabývat dvou hodnot reprezentujících logickou nulu a jedničku. Při nesymetrickém napájení operačního zesilovače bude logickou nulu reprezentovat hodnota 0 V a logickou jedničku kladné saturační napětí operačního zesilovače. Výstup z operačního zesilovače pak lze přivést k dalším logickým obvodům, případně k mikroprocesoru pro další zpracování.



Obr. 7: Zapojení komparátoru s fotodiodou



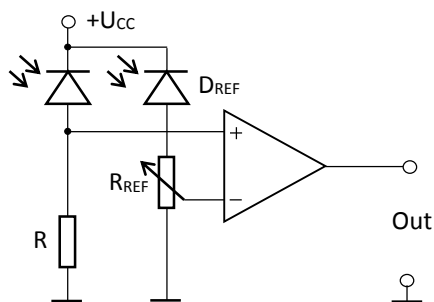
Komparátor je třeba pomocí potenciometru nastavit tak, aby na invertujícím (referenčním) vstupu bylo napětí vyšší, než je napětí na neinvertujícím vstupu při vypnutém zdroji světla. Takové napětí je úměrné rušení od okolních zdrojů světla. Na Obr. 8 lze vidět možné úrovně vstupních a výstupních napětí.



Obr. 8: Úrovně napětí před a po komparaci

Napětí  $U_{ON}$  odpovídá napětí na rezistoru  $R$  při plně osvětlené fotodiodě pomocí zdroje světla v optické závoře. Napětí  $U_{OFF}$  odpovídá napětí na rezistoru  $R$  při vypnutém zdroji světla nebo přerušném paprsku. Toto napětí může vlivem okolních světelných zdrojů nabývat nezanedbatelných hodnot. Je možné ho snížit například odfiltrováním části slunečního spektra, které dopadá na fotodiodu, případně také kvalitním zatemněním optiky, ve které je umístěna fotodioda. Úroveň napětí  $U_{REF}$  je možné libovolně měnit pomocí potenciometru. Pro správnou funkci je třeba velikost  $U_{REF}$  nastavit tak, aby byla vždy vyšší než maximální hodnota  $U_{OFF}$ .

Jedno z dalších možných řešení je místo potenciometru použít druhou fotodiodu se sériovým rezistorem. Do této diody se paprsek nesměruje, neboť slouží jako referenční a zaznamenává intenzitu osvětlení od dalších rušivých zdrojů světla. Tím se dosáhne proměnné hodnoty  $U_{REF}$  v závislosti na okolním osvětlení. Napětí na rezistoru za referenční diodou se pak porovnává s napětím na rezistoru za hlavní diodou. V neosvětleném stavu by byla obě napětí stejná a z toho důvodu je třeba zvolit velikost referenčního rezistoru tak, aby na něm bylo v neosvětleném stavu vyšší napětí.



Obr. 9: Zapojení komparátoru s referenční diodou

### 1.1.5 Mikroprocesor

Úkolem mikroprocesoru v tomto zařízení je především záznam časů a jejich odesílání pomocí sériové jednotky UART, dále pak obsluha případných dalších periférií. Požadavky na výpočetní výkon tedy nejsou velké. Je ovšem potřeba značné přesnosti časovače mikroprocesoru. Časovač je možné řídit hodinovým signálem vnitřního oscilátoru mikroprocesoru. Tyto oscilátory ale obvykle nevykazují dostatečnou přesnost, problémem bývá také značná teplotní závislost. Lepších vlastností lze dosáhnout použitím externího hodinového krystalu. Frekvence těchto krystalů je obvykle 32768 Hz, což je  $2^{15}$  Hz. Jedná se o mocninu dvou, se kterou mohou číslicové systémy snadněji pracovat než s mocninou 10. Dělením lze snadno získat frekvenci 1 Hz, která odpovídá periodě jedné sekundy. Problematické je ovšem dělení tohoto čísla na setiny sekundy, viz dále.

### 1.1.6 UART/USB převodník

Současné mikroprocesory jsou v drtivé většině vybaveny obvodem, který má na starosti sériovou komunikaci. Často se setkáváme s podporou protokolů UART, SPI nebo I<sub>2</sub>C. Pro komunikaci přes UART postačí 3 konektory obvykle značené Rx (příjem), Tx (vysílání) a GND (společná země). Komunikace zpravidla probíhá po bajtech. Nejdříve je odeslán start bit, za ním následuje nejčastěji 8 datových bitů, případný paritní bit a končí se jedním, případně dvěma stop bity [4]. Obě komunikující zařízení musí pracovat na přibližně stejné přenosové rychlosti, tzv. baud rate. K mikroprocesoru je takto možné přímo připojit další mikroprocesor nebo jiné zařízení. Pro komunikaci mikroprocesoru s PC je ovšem potřeba použít dalšího přídavného obvodu. Pokud je na PC k dispozici konektor RS-232, stačí použít převodník napětových úrovní a vybavit zařízení portem RS-

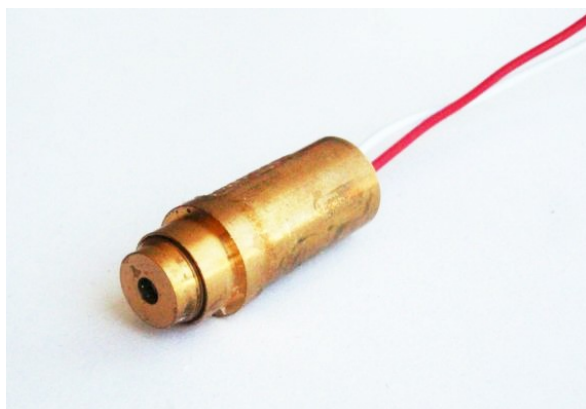
232. V opačném případě je třeba použít složitějších UART/USB převodníků. Někteří výrobci nabízí k mikroprocesorům vývojové desky, které již tento převodník obsahují.

## 2 Návrh hardware

V této části je již věnována pozornost samotnému návrhu hardwarové části zařízení. Zařízení je rozděleno na modul vysílače a přijímače. Hardware obou modulů je možné dále rozdělit na optickou, elektronickou a konstrukční část.

### 2.1 Vysílač

Modul vysílače je v našem případě pouze zdrojem světla, nemá žádnou jinou funkci. Ideální pro světelnou závoru tohoto typu by byl úzký paprsek monochromatického světla. Jako vhodné řešení se nabízelo použití laserové diody vyzařující světlo o viditelné vlnové délce. Infračervený laser byl zavrhnut z důvodu, že při nastavování paprsku na přijímač by světelný bod nebyl okem viditelný, čímž by se nastavování znesnadňovalo. Nakonec byla zvolena laserová dioda o vlnové délce 650 nm s optikou o maximálním výstupním výkonu 4mW. Tento výkon se později ukázal jako naprosto dostačující pro několikametrovou vzdálenost mezi vysílačem a přijímačem.



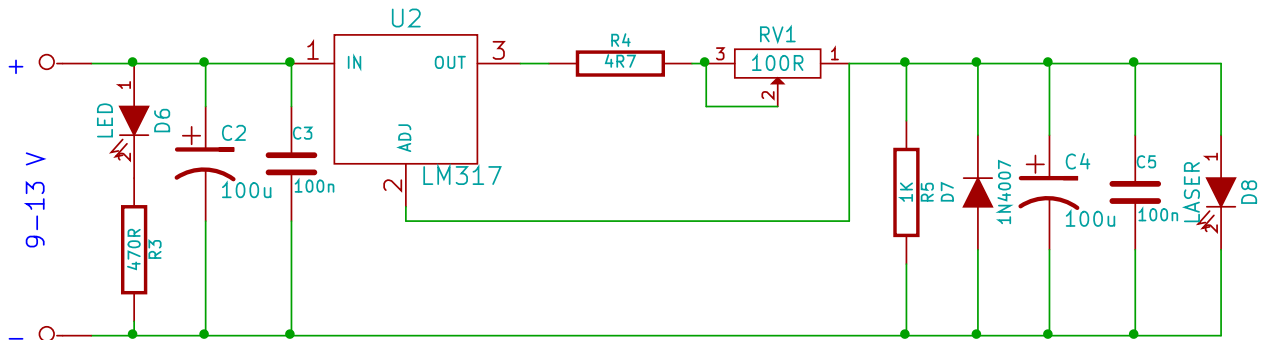
Obr. 10: Zvolený laserový modul F-LASER 5mW

Tab. 1: Parametry laserové diody (převzato z [5])

$U_F$	$P_O$	$I_F$	$\lambda$	$T_{SL}$
3 V	4 mW	35 mA	650 nm	3000 h

Modul vysílače by měl být vzhledem k předpokládanému užití na běžeckých a jiných závodech schopen provozu jak ze sítě, tak i na baterie. Při provozu na baterie je vhodné, aby byl nezávisle na stavu baterií dodáván laserové diodě pokud možno konstantní proud,

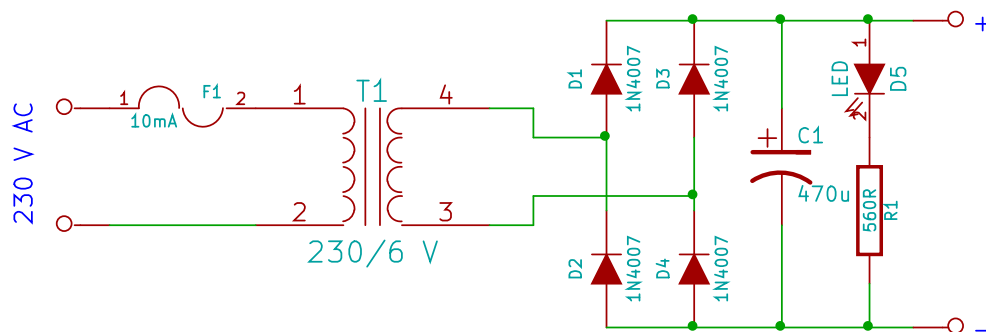
tedy aby se při poklesu napětí na bateriích v důsledku částečného vybití nesnižoval proud protékající diodou. Pro stabilizaci proudu při napájení z baterií byl použit obvod LM317T v jednom z jeho obvyklých zapojení. Obvod je navíc vybaven ochranným článkem převzatým z [6], který se skládá z rezistoru, diody a dvou kondenzátorů. Dioda je takto chráněna před statickým nábojem a před přepólováním.



Obr. 11: Schéma zapojení stabilizátoru proudu a ochranného článku laserové diody

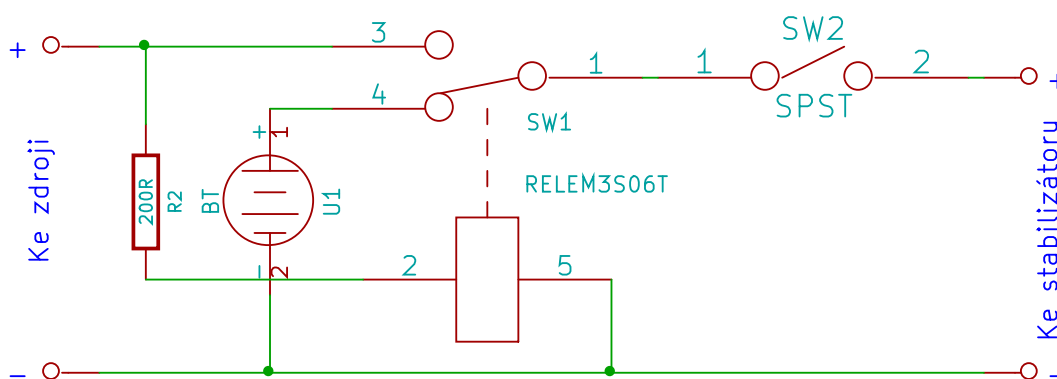
Proud dodávaný stabilizátorem do zátěže se určí z rovnice (rov.2). Pro požadovaný proud 30 mA je třeba, aby velikost odporu mezi vývody č. 2 a 3 byla přibližně 41  $\Omega$ . Odporový trimr je tedy třeba nastavit přibližně na hodnotu 36  $\Omega$ . Odporový trimr je v zapojení umístěn kvůli případné pozdější výměně laserové diody za výkonnější. Umožňuje plynulé nastavení velikosti proudu protékajícího zátěží. LED dioda D6 slouží k indikaci zapnutého laseru.

Vysílač lze také napájet ze sítě 230 V. Pro tyto účely je ve vysílači zabudován klasický usměrňovač s vyhlazovacím kondenzátorem.



Obr. 12: Schéma zapojení síťového zdroje napětí

LED dioda D5 slouží k indikaci připojeného síťového napájení. Výstupní napětí síťového zdroje v zatíženém stavu je přibližně 11 V. Napětí použité baterie je 9 V. Dále bylo třeba navrhnout způsob přepínání mezi síťovým zdrojem a baterií, nejlépe tak, aby se při zapojení zařízení do sítě automaticky odpojila baterie. To bylo vyřešeno pomocí relé.



Obr. 13: Schéma zapojení přepínání napájení

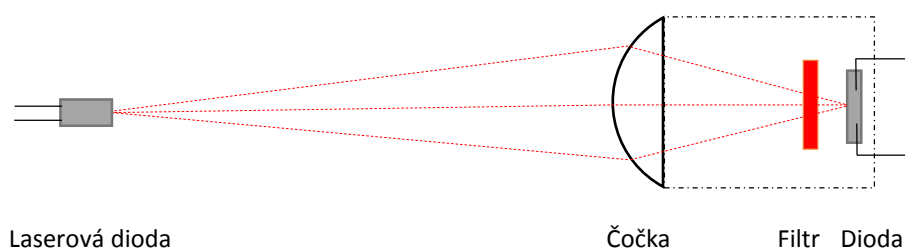
Pokud je síťový zdroj nezapojen, jako zdroj napětí se používá baterie. V okamžiku zapojení síťového zdroje relé sepne, baterie se odpojí a zařízení se napájí ze sítě. Spínač SW2 slouží k zapnutí napájení laseru nezávisle na použitém zdroji.

## 2.2 Přijímač

### 2.2.1 Optika

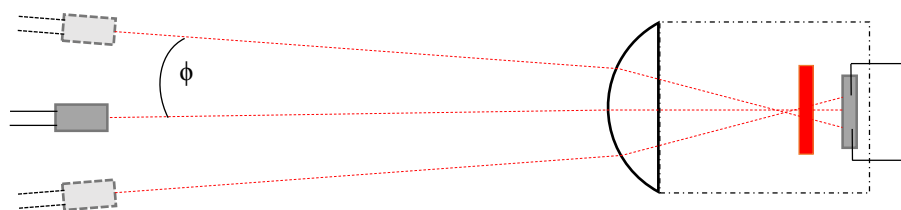
Jedním ze zásadních bodů při návrhu přijímače byla použitelná optická část. Ta sestává z čočky, barevného filtru a fotodiody. Důvodem k použití čočky – spojky, byla skutečnost, že čím větší je vzdálenost mezi oběma moduly, tím hůře se zaměřuje laserový

paprsek na jeden bod. Pokud uvážíme obvyklé rozměry fotodiody, bylo by přesné namíření paprsku velmi obtížné. Je také třeba počítat s možnými vibracemi modulů. Z tohoto důvodu se ukázalo jako nutné použít čočku, která by rozšiřovala fotocitlivou plochu a paprsek směřovala do jednoho bodu na fotodiodu. Dále bylo třeba omezit dopad denního světla z okolí na fotodiodu, aby bylo možné rozpoznat osvětlenou a neosvětlenou fotodiodu. K tomuto účelu byl použit barevný filtr, který propouští světlo o vlnové délce přibližně 650 nm. Optická komora byla navíc důkladně zatemněna. Tím bylo dosaženo výrazného snížení protékajícího proudu diodou v neosvětleném stavu.



Obr. 14: Principiální nákres uspořádání a funkce optiky přijímače

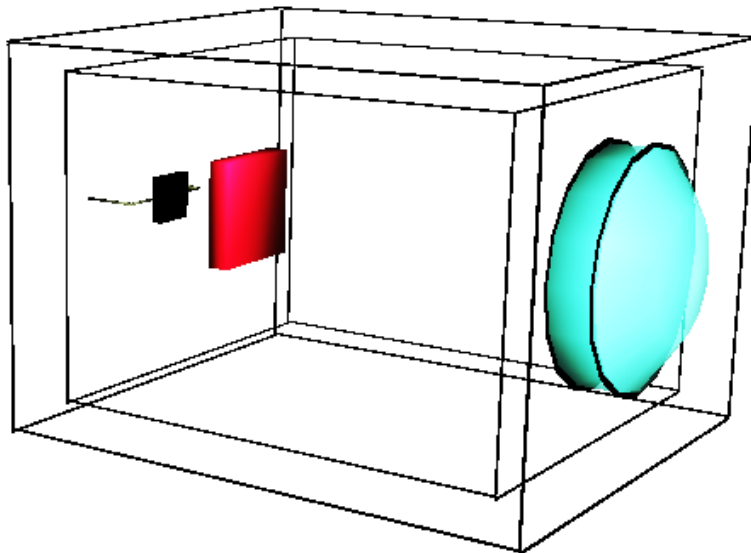
Tato jednoduchá optika má však i své zápory. Vysílač i přijímač musí být pokud možno v ose. Úhel, který svírá osa čočky s paprskem, musí být minimální, neboť při větších úhlech paprsek nedopadá na fotodiodu. Znázorněno je to na dalším obrázku.



Obr. 15: Vliv nenulového úhlu dopadu na rozbíhavost paprsku

Bylo tedy třeba použít fotodiodu s dostatečně velkou fotocitlivou plochou a také najít kompromis mezi velikostí čočky a maximálním možným úhlem  $\phi$ . Při pozdějších pokusech totiž větší čočky se vzdálenějším ohniskem vykazovaly menší maximální úhel dopadu  $\phi$  než čočky s menším průměrem a menší vzdáleností ohniska. Navíc, maximální možná vzdálenost ohniska byla v tomto případě omezena velikostí konstrukční krabičky.

Jako nejvhodnější pro tyto účely se ukázala být spojka o průměru 18 mm s ohniskem ve vzdálenosti 50 mm. Následující obrázek naznačuje uspořádání všech prvků v optické komoře ve 3D.



Obr. 16: 3D pohled na uspořádání optických prvků v komoře

### 2.2.2 Mikroprocesor

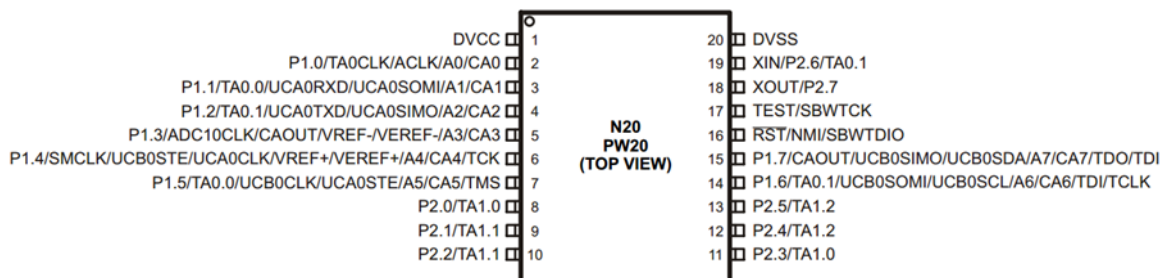
Základním bodem elektronické části je volba vhodného mikroprocesoru. Pro tyto účely byl zvolen mikroprocesor MSP430G2553 od firmy Texas Instruments, která nabízí rovněž jednoduchou vývojovou desku Launchpad.

Americká firma Texas Instruments, která se zabývá výrobou polovodičových součástek, nabízí několik rodin mikroprocesorů pro různá použití. 16 bitové mikroprocesory MSP430 jsou navrženy pro užití v nízkonapěťových aplikacích, nabízí minimální spotřebu proudu a pět různých úsporných provozních módů, ve kterých jsou podle potřeby vypnuté některé části mikroprocesoru. Pro přechod z úsporného módu do plně aktivního módu stačí procesoru méně než 1  $\mu$ s. [6] Zvolená varianta mikroprocesoru MSP430G2553 disponuje těmito vlastnostmi:

- Nízké napájecí napětí 1.8V až 3.6 V
- Velmi nízká spotřeba (v aktivním módu 230  $\mu$ A)
- Pět úsporných módů

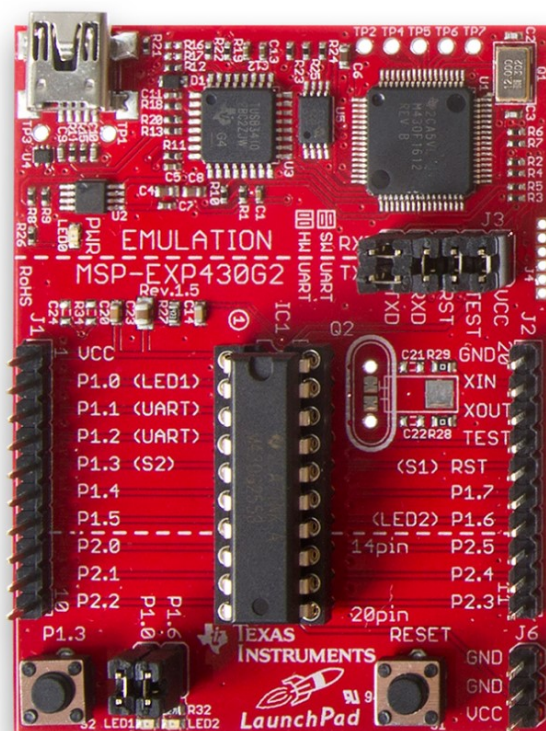


- 16 bitová RISC architektura, doba jedné instrukce 62,5 ns
- Pracovní frekvence do 16 MHz
- 512 B RAM
- 16 KB Flash
- Dva 16 bitové časovače
- Připojitelný externí zdroj časování
- 16 I/O pinů
- Univerzální sériové komunikační rozhraní (USCI)
- Vestavěný komparátor
- 10 bitový ADC
- 20 pinové pouzdro DIP

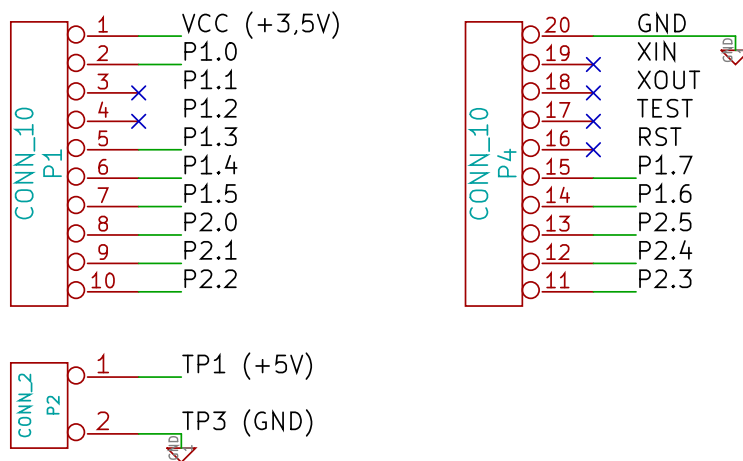


Obr. 17: Rozložení pinů mikroprocesoru MSP430G2553 (Převzato z [7])

Vývojová deska Launchpad slouží jako jednoduchá základní deska pro mikroprocesory MSP430G2xxx, která umožňuje programování a odlaďování programu v procesoru. O komunikaci s PC přes USB se starají podpůrné obvody na desce. Deska je zároveň dostatečně malá a jednoduchá, takže bylo rozhodnuto nevytvářet později novou DPS pro mikroprocesor, ale použít v zařízení tuto desku a k ní připojit případné přídatné moduly. Všechny piny mikroprocesoru jsou vyvedeny na desce, takže je k nim snadný přístup.



Obr. 18: Deska Launchpad osazená procesorem MSP430G2553 (převzato z [8])



Obr. 19: Použité vývody desky Launchpad

Deska nabízí celkem 16 vstupů / výstupů, z nichž P1.1 a P1.2 jsou použity pro sériovou komunikaci a nelze je využít pro jinou funkci. Piny XIN a XOUT jsou použitelné též jako I/O. Ke zbývajícím pinům je připojena fotodioda, signalizační diody, konektor pro externí branku, LCD displej a tónový generátor. Testovací ploška TP1 je zde využita k

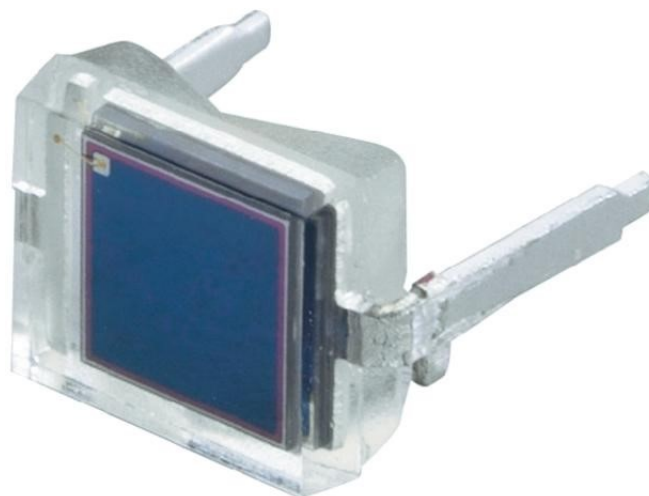
získání napětí 5V potřebného pro obvod 555 a LCD. Napájecí napětí mikroprocesoru (3,5 V) by totiž bylo nedostatečné.

### 2.2.3 Periferie

V této části je věnována pozornost všem součástkám a modulům připojeným k desce Launchpad, které jsou zde nazývány periferiemi. Mezi ně je zahrnuta fotodioda, signalizační diody pro indikaci osvětleného nebo neosvětleného přijímače, konektor pro externí startovací branku nebo pro druhou světelnou závoru, informativní LCD displej a tónový generátor s obvodem 555, který slouží ke generování zvukových signálů při intervalových startech a při průběhu závodníka.

#### Fotodioda

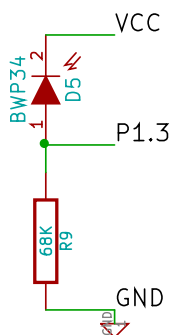
Použitá fotodioda BPW34 je rychlá dioda typu PIN, která je citlivá na viditelné světlo a na blízké infračervené záření. Vrchol citlivosti má tato dioda v oblasti 950 nm. Na 650 nm je dioda méně citlivá, což zde ale nevádí kvůli použitému filtru.



Obr. 20: Fotodioda BPW34 (převzato z [9])

Tab. 2: Parametry fotodiody (převzato z [10])

$I_{R0}$	$I_{RA}$	$T_R$	$U_0$	$\lambda$
2 nA	50 $\mu$ A	100 ns	350 mV	430-1100 nm



Obr. 21: Připojení fotodiody a rezistoru k desce Launchpad

Při určení velikosti rezistoru R9 vycházíme z dostupného katalogového údaje IRA, který udává proud procházející diodou v závěrném směru při intenzitě osvětlení  $E = 1 \text{ mW/cm}^2$ , vlnové délce  $\lambda = 950 \text{ nm}$ , napětí na diodě  $U_R = 5 \text{ V}$ . Tyto podmínky se sice neshodují s našimi, ale pro tyto účely je možné rozdíly zanedbat. Požadujeme, aby při plně osvětlené diodě bylo na rezistoru dostatečně velké napětí, v ideálním případě  $3,5 \text{ V}$ . Z Ohmova zákona určíme přibližnou velikost odporu R9.

$$R = \frac{U}{I} = \frac{3,5}{50 \times 10^{-6}} = 70 \text{ K}\Omega \quad (\text{rov.3})$$

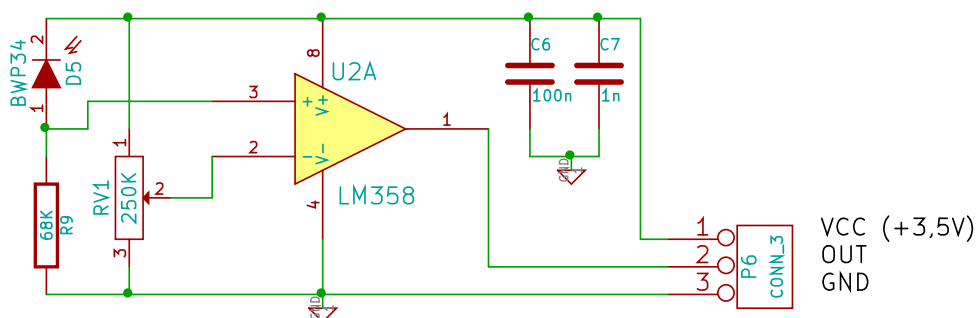
Nejbližší dostupná hodnota je  $68 \text{ K}\Omega$ .

V praxi napětí na rezistoru R9 dosahuje při osvětlení laserem hodnoty přibližně  $2 \text{ V}$ , což je pro další zpracování dostatečné.

Napětí na rezistoru R9 se zavádí do interního komparátoru mikroprocesoru. Jako referenční napětí byla zvolena hodnota  $0,25 \times V_{cc}$ , což je  $0,875 \text{ V}$ .

Komparátor umožňuje jednoznačně rozpoznat, kdy je fotodioda osvětlena laserem. Navzdory tomu, že optická komora byla důkladně zatemněna a byl použit barevný filtr, na diodu dopadá část slunečního spektra v blízkosti  $650 \text{ nm}$ . Diodou tedy protéká za denního světla nezanedbatelný proud a napětí na R9 je nenulové. Později bylo ověřeno, že pokud není optika namířena přímo proti Slunci, je napětí na R9 vždy menší než referenční hodnota  $0,875 \text{ V}$ . Pomocí komparátoru tedy takto lze spolehlivě odfiltrovat nežádoucí rušení v podobě slunečního světla.

V průběhu návrhu bylo uvažováno i použití samostatného externího komparátoru s operačním zesilovačem. Toto zapojení umožňovalo použití trimru pro manuální nastavení referenčního napětí. Ten bylo sice možné použít i u interního komparátoru, ale bylo by tím využito o jeden pin desky více. Nakonec z důvodu nepotřebnosti manuálního nastavení referenčního napětí a z důvodu snahy o ušetření součástek bylo rozhodnuto o užití interního komparátoru s pevně nastaveným referenčním napětím. Následující obrázek ukazuje schéma zapojení původního externího komparátoru.

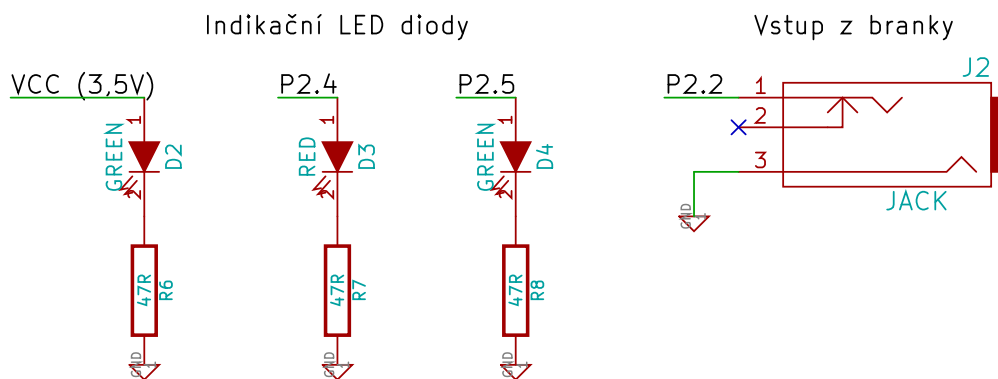


Obr. 22: Původní koncepce komparátoru

### Signalizační diody a externí branka

Na modulu přijímače se nacházejí celkem tři, dvě z nich slouží k indikaci stavu osvětleno/neosvětleno. Pokud není mezi vysílačem a přijímačem žádná překážka a paprsek je správně namířen, svítí zelená LED dioda. V opačném případě svítí červená LED dioda. Třetí, zelená dioda svítí, pokud je zařízení v provozu.

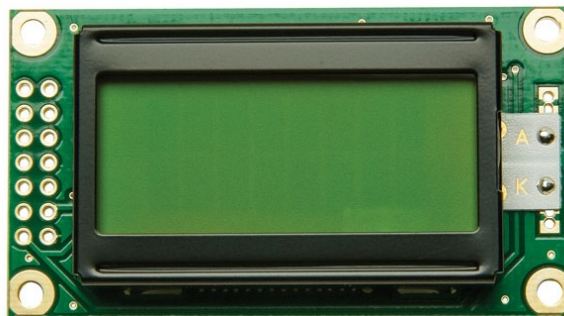
Pro připojení startovací branky je zde vyhrazen samostatný konektor typu Jack 3,5 mm. Branka, kterou závodníci procházejí na startu, funguje jako spínač. Při sepnutí dojde k zaznamenání času stejně, jako kdyby závodník procházel světelnou závorou. Ke konektoru lze v případě potřeby připojit i druhou světelnou závoru, jejíž modul přijímače už nemusí být vybaven mikroprocesorem, ale pouze komparátorem a spínacím tranzistorem.



Obr. 23: Připojení indikačních LED diod a konektoru branky k vývodům Launchpadu

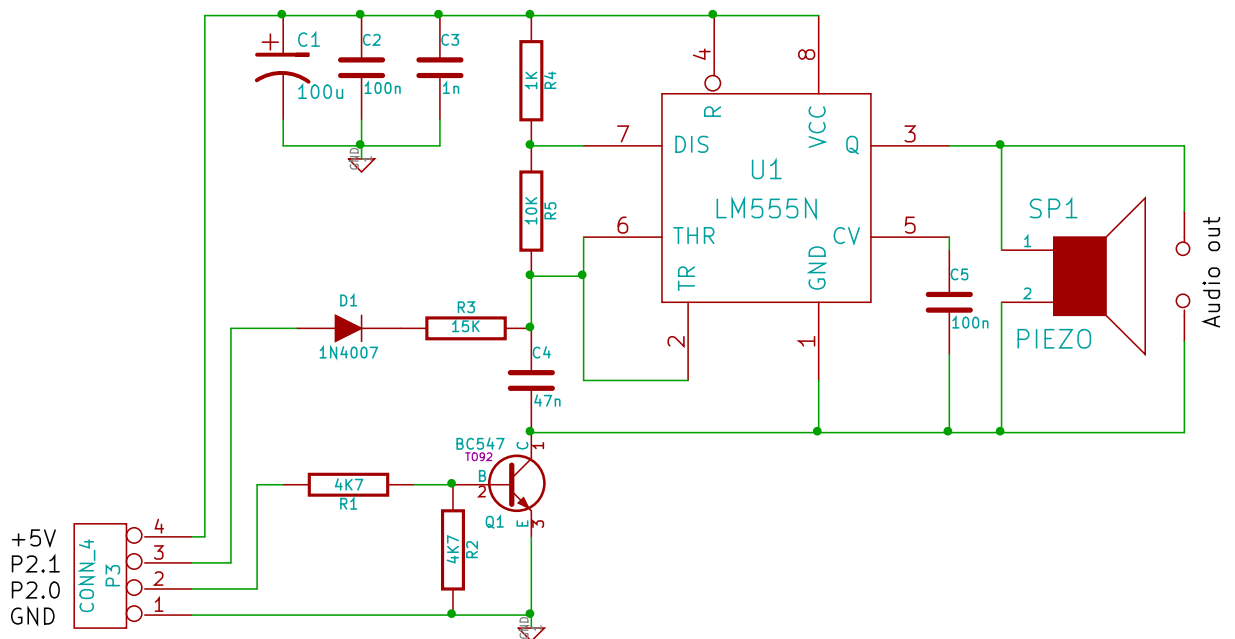
## LCD displej

Jednotka přijímače disponuje také LCD displejem, který slouží k zobrazování aktuálního času a stavu (start/stop). LCD disponuje integrovaným řadičem kompatibilním s Hitachi HD44780 a umožňuje zobrazit 8x2 znaků. Displej dále nabízí podsvícení, nastavení kontrastu a s řadičem displeje je možné komunikovat ve 4 bitovém nebo 8 bitovém paralelním módu. Vzhledem k omezenému počtu vývodů Launchpadu zde byl zvolen 4 bitový mód, ve kterém je přenos dat pomalejší, neboť se každý bajt odesílá po dvou částech.



Obr. 24: LCD displej Winstar 8x2 znaků (převzato z [11])





Obr. 26: Schéma zapojení spínaného tónového generátoru s časovačem 555

Obvod umožňuje generovat dva tóny o různé frekvenci. Nižší tón se zde používá pro odpočítávání a vyšší tón je signálem ke startu. Zvýšení tónu se zajišťuje přivedením napětí 3,5 V z výstupu P2.1 na Launchpadu. Při požadavku na zvýšený tón díky tomu teče kondenzátorem při jeho nabíjení větší proud, kondenzátor se nabíjí rychleji a tím také roste frekvence generovaného tónu. Napájecí napětí 5 V pro časovač se získá z testovací plošky Launchpadu TP1, která je připojena k +5 V USB portu. V případě nedostatečné hlasitosti zabudovaného piezoměniče je k dispozici konektor Jack 3,5 mm pro připojení externího nízkofrekvenčního zesilovače a odpovídajícího reproduktoru.

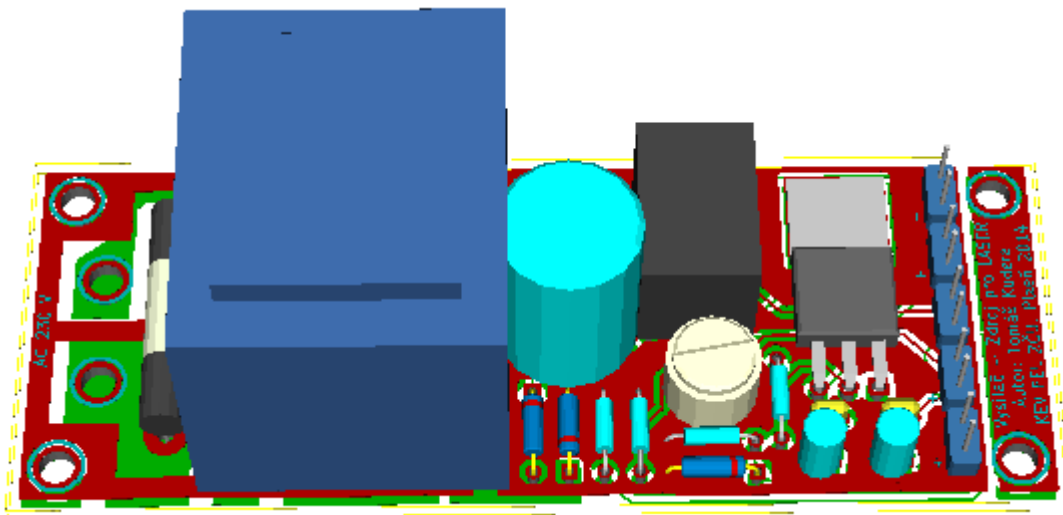
## 2.3 Realizace

### 2.3.1 Návrh DPS

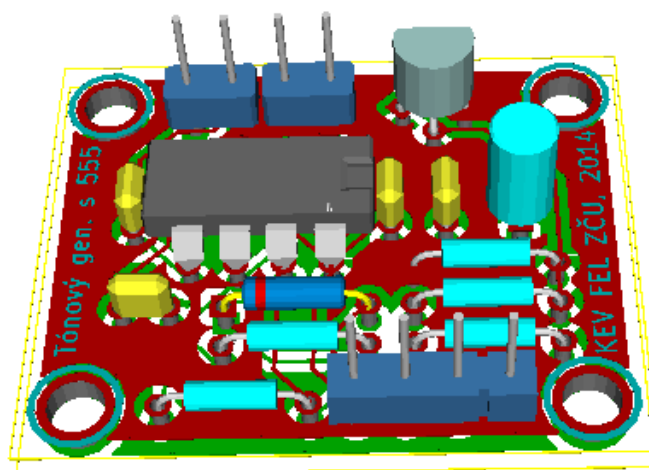
Pro návrh DPS vysílače a tónového generátoru byl použit volně dostupný program KiCAD.



Obě desky jsou dvouvrstvé, součástky jsou umístěny z vrchu. Potisk DPS je umístěn na vrchní straně, nepájivá maska je na vrchní i spodní straně. Nebyly zde použity SMD součástky z důvodu vlastnoručního osazení desky prototypu a usnadnění pájení. Výstupní soubory gerber pro výrobu DPS lze nalézt na CD přiloženém k bakalářské práci. Zde jsou uvedeny pouze 3D modely navržených a osazených desek plošných spojů.



Obr. 27: 3D model osazené DPS vysílače



Obr. 28: 3D model osazené DPS tónového generátoru s 555

### 2.3.2 Konstrukce

Při samotném návrhu hardwarové části zařízení, modulů vysílače i přijímače bylo třeba brát v úvahu více požadavků:

- Dostatečně pevná a odolná konstrukce
- Stabilní poloha nad zemí
- Eliminace kývání a vibrací
- Nastavitelná výška

Z těchto důvodů se již od počátku nabízelo použití stativu, ke kterému by bylo možné modul přijímače a vysílače upevnit. Pro tyto účely byl zvolen stativ Hama Star 05, který umožňuje nastavení výšky v rozmezí 36,5 – 106,5 cm. Destička s aretačním šroubem pro fotoaparát je odnímatelná pomocí jednoduchého mechanismu. Bylo tedy třeba pouze vytvořit na modulech výstupek, který by zapadal do upínacího mechanismu.

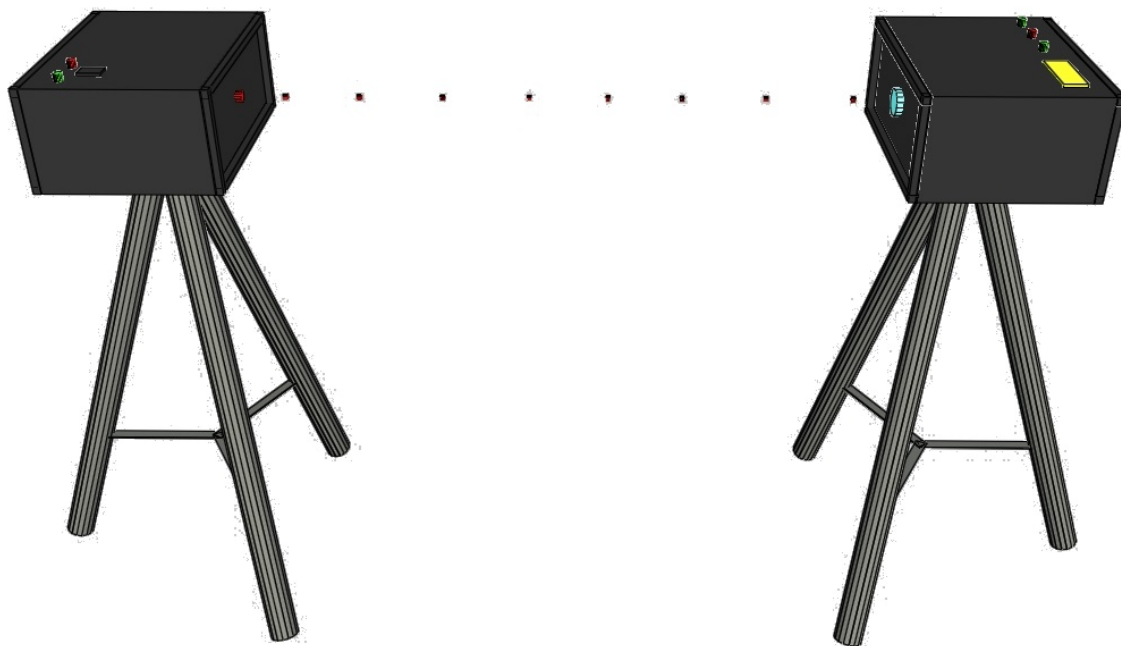


Obr. 29: Stativ Hama Star 05 (převzato z [12])

Dále bylo třeba odhadnout rozměry obou modulů a podle tohoto odhadu zvolit příslušnou konstrukční krabičku. Jako vhodná se ukázala být krabička KP 8 od firmy GES-

ELECTRONICS, a.s. o rozměrech 149x110x71 mm. Pro zvýšení stability přijímače byly k vnitřním stěnám krabičky připevněny kovové pláty, které zvyšují váhu. Váhu vysílače nebylo potřeba zvyšovat, neboť je již dostatečná díky bateriím a transformátoru.

Fotografie obou zhotovených modulů jsou dostupné v části Přílohy.



Obr. 30: Ukázka konstrukčního řešení zařízení

### 3 Návrh software

Tato část se zabývá návrhem a popisem funkce softwarové části zařízení. Tu je možné dále dělit na program mikroprocesoru a na uživatelský program Time Terminal.

#### 3.1 Program mikroprocesoru

Program mikroprocesoru MSP430 byl vytvářen v jazyce C ve vývojovém prostředí Code Composer Studio, které je poskytováno firmou Texas Instruments pro její procesory. Vývojové prostředí může být používáno pod vícero licencemi. Při vývoji softwaru pro procesory MSP430 je možné použít Free licenci, která uživatele omezuje pouze v maximální velikosti kódu na 16 KB. Pro tyto účely však 16 KB bohatě stačí, program zabírá ve flash paměti mikropočítače přibližně 3,2 KB.

##### 3.1.1 Náhled na funkci

V hlavním while cyklu programu se mikroprocesor stará o přepočítání časů, pípání v režimu intervalových startů, aktualizaci údaje na LCD každou sekundu, provádění příkazů po přijetí řídicího znaku a také o volání funkce pro odeslání času, pokud byl zaznamenán průchod světelnou závorou.

O počítání času se zde stará časovač, který opakovaně vyvolává přerušování, při kterých dochází ke změně interního časového údaje, který je uložen ve čtyřmístném poli. Po dosažení hodnoty 60 sekund se v hlavním cyklu programu sekundy vynulují a přičte se jedna minuta. Podobně je tomu u minut a setin.

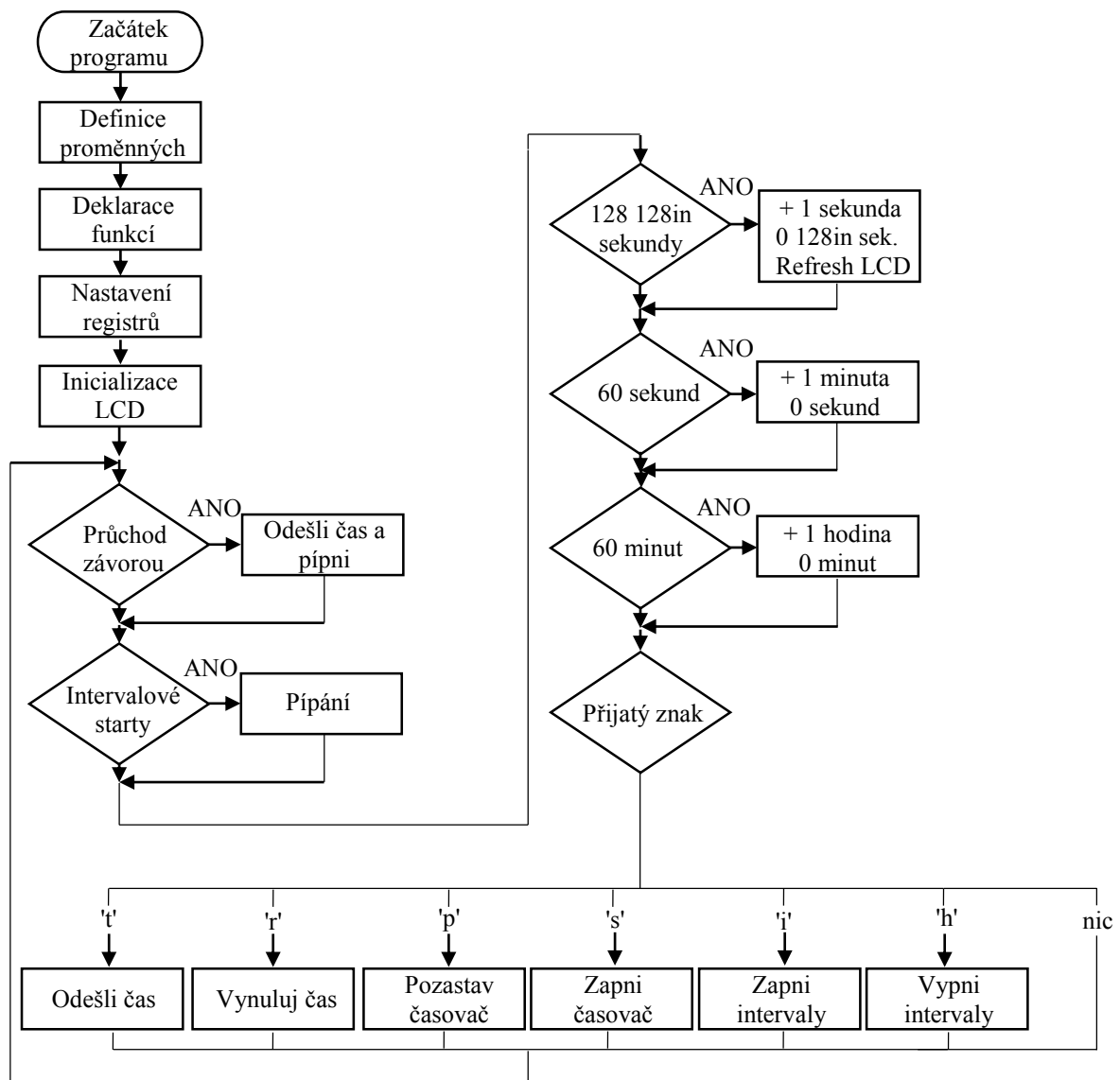
Přerušování může být též vyvoláno příjmem příkazu v podobě znaku z PC. Přijatý znak se uloží a v hlavním cyklu se provedou patřičné příkazy.

Tab. 3: Řídící znaky sloužící k ovládání přijímače

Znak	't'	'r'	'p'	's'	'i'	'h'
Příkaz	vynucené odeslání času	vynulování času	pozastavení počítání času	start počítání času	režim intervalových startů	režim hromadných startů

Další událost vyvolávající přerušeni je průběh světelnou závorou, případně sepnuti branky. V tomto případě se v obsluze přerušeni komparátoru zaznamená, že došlo k průběhu a po návratu do hlavní smyčky se v patřičné funkci vyvolá přerušeni pro odeslání patřičného času přes UART.

Pro obsluhu LCD displeje byl použit samostatný kód převzatý z [13].



Obr. 31: Základní vývojový diagram programu mikroprocesoru

### 3.1.2 Nastavení registrů

Po deklaraci proměnných a konstant je nutné ještě před vstupem do while cyklu programu ve funkci main nastavit registry mikroprocesoru pro správnou funkci všech modulů a částí, které chceme používat. Následují výňatky z kódu s dodatečným popisem.

```
WDTCTL = WDTPW + WDTHOLD;           // Stop WDT
```

Tento příkaz slouží k deaktivaci Watchdog timeru, který resetuje procesor v případě softwarové chyby.

```
P2DIR = (BIT0 | BIT1 | BIT4 | BIT5); // P2.0, P2.1, P2.4 a P2.5 jako výstupy
P2IFG &= ~(BIT2);                   // clear interrupt flag P2.2
P2IE |= (BIT2);                     // P2.2 interrupt povolen
P2OUT |= BIT2;                      // na P2.2 logická 1
P2REN |= (BIT2);                   // pullup rezistor na P2.2
```

Porty P2.0, P2.1, P2.4 a P2.5 jsou nastaveny jako výstupy. Port P2.2 je vstup připojen k brance a požadujeme, aby mohl vyvolat přerušení. Přivedeme na něj pomocí čtvrtého příkazu napětí 3,5 V a pomocí pátého příkazu do obvodu zařadíme interní rezistor pro omezení proudu při sepnutí branky.

```
TACTL |= TASSEL_1 + ID_3 + MC_1;    // ACLK z krystalu, upmode
BCSCTL3 |= XCAP_3;                 // 12.5pF - nastavení pro 32768Hz krystal
CCTL0 |= CCIE;                     // CCR0 interrupt povolen
CCR0 = 31;                          // 4096/128 = 32. Pocita se od 0 do 31.
```

První příkaz nastavuje kontrolní registr časovače. TASSEL\_1 značí řízení časovače přesným externím krystalovým oscilátorem o frekvenci 32768 Hz. Ten není na desce umístěn přímo z výroby, je nutné jej dodatečně připájet. [14] Frekvence oscilátoru je dělena osmi, což určuje hodnota ID\_3. Hodnota MC\_1 pak značí, že časovač počítá v tzv. upmode, přičítá se od nuly po jednom až do stanovené hodnoty CCR0. Po jejím dosažení se počítá znovu od nuly. Druhý příkaz slouží k volbě velikosti kapacity interního kondenzátoru potřebného pro správnou funkci krystalu. Třetím příkazem povolujeme

časovači vyvolání přerušení. Posledním příkazem nastavujeme hodnotu, při které dojde k vyvolání přerušení.

Vzhledem k tomu, že frekvence oscilátoru je dělena osmi, pracuje se zde s frekvencí 4096 Hz. Časovač tedy zvýší svou hodnotu o 1 každou  $\frac{1}{4096}$  s. Potřebujeme, aby došlo k přerušení ideálně každou  $\frac{1}{100}$  s pro umožnění měření setin sekundy. K tomu bychom ovšem potřebovali krystal s frekvencí dělitelnou 100. Dostupné krystaly s frekvencí v řádech MHz jsou pro účely přesného měření času nevhodné a krystal s nižší frekvencí dělitelnou 100 nebyl dostupný, a tak bylo rozhodnuto zůstat u frekvence 32768 Hz a řešit situaci jinak. Hodnota CCR0 byla nastavena na 31, tím pádem každou  $\frac{1}{128}$  s dojde k vyvolání přerušení, neboť po dobu jedné sekundy se od 0 do 31 počítá 128 krát. Sekunda je tedy rozdělena na 128 intervalů, které se před odesláním převedou na setiny. Princip převodu je vysvětlen v části 3.1.3.

```
CACTL1 = CAON + CAEX + CAIE + CAREF1;           // komp. zapnut, exchange, interrupt, 0,25xVcc
CACTL2 = P2CA2 + P2CA1;                         // CA3 - vstup je na P1.3
```

Interní komparátor se nastavuje pomocí registrů CACTLx. Bit CAON slouží k zapnutí komparátoru. CAEX slouží k reverzaci invertujícího a neinvertujícího vstupu. Pomocí CAIE povolujeme komparátoru vyvolání přerušení. Hodnota CAREF1 značí, že jako referenční napětí je použito  $0,25 \times V_{cc}$ , což je 0,875 V. Toto napětí je připojeno k invertujícímu vstupu komparátoru. Hodnoty v druhém řádku značí, že porovnávané napětí přivádíme na CA3, což je port P1.3.

```
P1SEL = BIT1 + BIT2 ;                          // P1.1 = RXD, P1.2=TXD
P1SEL2 = BIT1 + BIT2 ;                         // P1.1 = RXD, P1.2=TXD
UCA0CTL1 |= UCSSEL_1;                          // CLK = ACLK
UCA0BR0 = 0x03;                                // 32kHz/9600 = 3.41
UCA0BR1 = 0x00;
UCA0MCTL = UCBSR1 + UCBSR0;                    // Modulace UCBSRx = 3
UCA0CTL1 &= ~UCSWRST;                          // Inicializace USCI
IE2 |= UCA0RXIE;                               // Povolit USCI interrupt
```

V nastavení sériové komunikace je nejdříve portu P1.1 přiřazena funkce RX a portu P1.2 funkce TX. Třetí příkaz nastavuje časování pomocí krystalu 32768 Hz. Čtvrtý a pátý

příkaz slouží k nastavení baud rate na 9600 dělením frekvence krystalu. Šestým příkazem nastavujeme modulaci. Sedmý příkaz slouží uvedení USCI modulu do provozu. Posledním příkazem povolujeme vyvolání přerušení při přijmutí znaku. Nastavení sériové komunikace bylo převzato z tutoriálů poskytnutých firmou TI v CCS.

```
__bis_SR_register(GIE);           // Global interrupt enable, aktivni rezim
```

Tímto závěrečným příkazem povolujeme obsluhu přerušení. Procesor disponuje pěti různými provozními módy umožňujícími výrazné snížení spotřeby díky vypnutí některých částí mikroprocesoru. Zde přecházíme do aktivního režimu, kde jsou všechny části zapnuté.

### 3.1.3 Odesílání časů

Při průběhu světelnou závorou dojde k vyvolání přerušení komparátorem a v obsluze přerušení se proměnné sepnuto přiřadí hodnota 1. Jak vyplývá z vývojového diagramu, v hlavním while cyklu programu je umístěna podmínka, která volá funkci pro odesílání časů, vždy když je proměnná sepnuto rovna 1.

Samotná funkce *odesli\_cas* pak slouží k převodu časů z pole typu int do pole typu char, které je následně možné odeslat pomocí UART. Neméně důležitý je převod 128in sekundy na setiny před odesláním. Pokud vydělíme  $\frac{100}{128}$ , získáme číslo 0,78125, což je hodnota, kterou je třeba násobit 128iny při převodu na setiny. Pro snížení počtu prováděných aritmetických operací při převodu bylo předem vytvořeno pole o velikosti 128 prvků, které obsahuje zaokrouhlené hodnoty 0-100, některé vícekrát. Jedná se o pole konstant, které se ukládá do flash paměti, nikoli do RAM.

Tab. 4: Ukázka způsobu převodu na setiny

128iny	0	1	2	3	4	5	6	7
setiny	0	0,781	1,562	2,343	3,125	3,906	4,687	5,468
zaokrouhl.	0	1	2	2	3	4	5	5

Tímto převodem vzniká aditivní chyba  $\pm 0,005$  sekundy.



Samotné časové údaje, které jsou umístěny ve čtyřprvkovém poli, je nutné převést do formátu vhodného k odesílání přes UART, což je pole typu char. Pro tyto účely byla zpočátku používána funkce printf, která je součástí knihovny stdio.h. Texas Instruments dodává knihovny standardně používané v jazyce C v úsporných verzích pro své mikroprocesory. I přes svou úspornost je ale funkce printf náročná na paměť a výpočetní výkon procesoru, a proto bylo později rozhodnuto vytvořit vlastní algoritmus, který rozloží dvojmístné číslo na jednotlivé cifry a každou zvlášť převede do formátu char následujícím způsobem:

Dvojciferné číslo  $x$ , které je zapsáno ciframi  $x_1$  a  $x_2$  lze upravit:

$$x_2 = x \bmod 10, \quad (\text{rov. 4})$$

$$x_1 = \frac{(x-x_2)}{10}, \quad (\text{rov. 5})$$

kde mod je zkratka pro operaci modulo.

V případě jednociferného čísla je hodnota  $x_1$  rovna nule.

Těmito operacemi jsme získali jednotlivé cifry. V jazyce C lze typu char přiřadit i celočíselnou hodnotu, pokud chceme znak definovat pomocí ASCII kódu. Čísla 0 až 9 mají ASCII kódy 48 až 57. Pokud tedy chceme cifru 0-9 převést na char, stačí nám k tomu:

$$ch = 48 + x_n, \quad (\text{rov. 6})$$

kde ch je proměnná typu char a  $x_n$  je cifra typu int získaná předchozími operacemi.

Jednotlivé časy se odesílají ve tvaru HH:MM:SS:SE, kde SE jsou setiny. Před odesláním je vyvoláno přerušení UART TX, ve kterém se po jednom znaku odešlou jednotlivé znaky z pole. Po dokončení přenosu je proměnné sepnuto přiřazena 0.

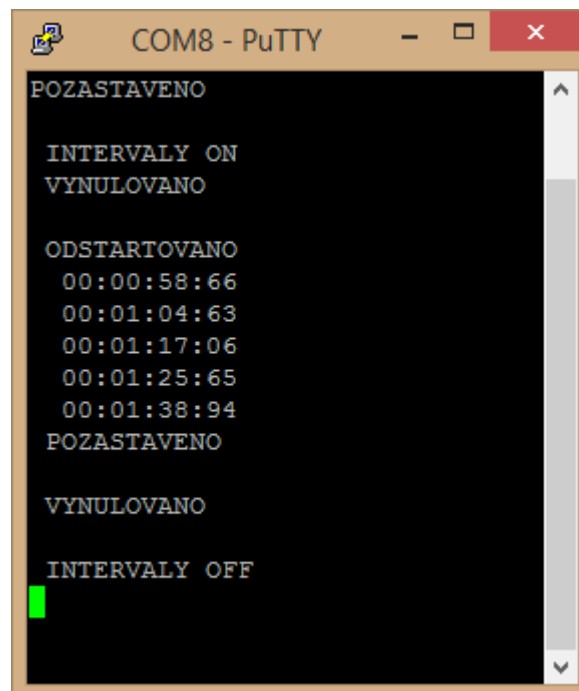
### 3.1.4 Využití programu Putty pro komunikaci

Po zkompileování programu a jeho nahrání do flash paměti mikroprocesoru je zařízení připraveno k činnosti. Systém Windows by měl být schopen automaticky po připojení USB k jednotce přijímače nainstalovat ovladače pro sériový port a zařízení následně přidělit některý z COM portů. S jednotkou přijímače lze komunikovat v libovolném terminálovém

programu podporujícím sériovou komunikaci. Parametry sériové komunikace jsou následující:

- Baud rate 9600 Bd
- 8 datových bitů
- 1 stop bit
- Bez parity
- Bez flow control

Samotná obsluha jednotky je díky šesti příkazů nenáročná. Po stisknutí dané klávesy zařízení odešle zpět informaci o provedení příkazu, tedy například po stisknutí 's' vidíme v terminálu vypsáno ODSTARTOVANO. Názorně to ukazuje následující obrázek.



Obr. 32: Komunikace s jednotkou v programu PuTTY

Sériová komunikace přes terminál byla také odzkoušena v linuxovém operačním systému, konkrétně v distribuci Fedora 21 x86\_64. Zprovoznění zařízení pod Linuxem by se též mělo obejít bez větších obtíží. Zařízení se po připojení objeví ve složce /dev pod názvem ttyACM0, případně ttyACM1, pokud došlo k opakovanému připojení a odpojení. Pro komunikaci v terminálovém okně lze použít například program screen. Při jeho spuštění stačí zadat cestu k zařízení a modulační rychlost.

```
$ sudo screen /dev/ttyACM0 9600
```

Pro nenáročné použití může takováto forma komunikace stačit. Časy lze z terminálu standardně kopírovat a vkládat je například do tabulkového procesoru, kde s nimi lze dále pracovat. Pro náročnější použití byl vytvořen obslužný program Time Terminal, který posouvá komunikaci s jednotkou na jinou úroveň.

## 3.2 PC program (Time Terminal)

Program Time Terminal, který byl vytvořen v jazyce Python 3, přináší uživatelsky přívětivější obsluhu zařízení a několik vlastních funkcí navíc. Program využívá několika Python modulů, z nichž nejdůležitější jsou moduly pySerial, Tkinter a configparser. První z nich má na starosti sériovou komunikaci, druhý modul umožňuje pracovat s grafickým uživatelským rozhraním. Třetí modul má na starosti práci s konfiguračními soubory. Aby mohl být skript vytvořený v Pythonu spuštěn, je třeba mít na počítači nainstalovanou kompatibilní verzi Pythonu a příslušné moduly. Je však také možné skript takzvaně zmrazit a vytvořit tak spustitelný exe soubor pro Windows, doplněný několika knihovnamí a dalšími soubory. Pro tyto účely je použit modul cx\_Freeze, který podporuje Python verze 3. Aplikace je po vytvoření balíčku spustitelná na počítačích s operačním systémem Windows, přičemž Python nemusí být na daném počítači nainstalován. Pro spuštění pod Linuxem (Fedora 21 x86\_64) bylo třeba program mírně upravit, neboť se zde sériové porty označují jinak než ve Windows.

### 3.2.1 Náhled na funkci

Vytvořený program umožňuje zaznamenávat přijaté časy a zobrazovat je v seznamu. Ten je dále možné v podobě výsledkové listiny exportovat do textového souboru. K dispozici je možnost mazání jednotlivých časů, duplikace času, manuální přidání času, přiřazování ID (startovních čísel) k jednotlivým časům a také spárování dvojic časů se stejným ID. Tato funkce je praktická zejména v případě, pokud jsou zaznamenávány startovní i cílové časy závodníků. Při párování se u každé označené dvojice odečte startovní čas od cílového a vytvoří se exportovatelná výsledková listina.

Při běžeckých závodech dochází k zaznamenání dvou časů při průběhu jednoho závodníka, což je způsobeno zaznamenáním časů obou nohou. Je tedy vhodné

zaznamenané časy filtrovat. Pro tyto účely byl vytvořen filtr s následující funkcí: Pokud je rozdíl dvou libovolných po sobě následujících časů menší než uživatelsky definovaná hodnota, druhý čas je automaticky smazán. Pokud časovou konstantu filtru nastavíme například na 10 setin, je možné zaznamenat další čas až po uplynutí 10 setin od záznamu předchozího času. Tento způsob filtrace je však pouze základní a pro dokonalejší filtraci by bylo třeba použít složitějších algoritmů. Experimenty bylo zjištěno, že rozdíl časů obou nohou se pohybuje v případě běhu od 0 do 20 setin. Pokud nastavíme časovou konstantu na 20, je sice možné relativně spolehlivě odfiltrovat časy druhých nohou, ale zaplatíme za to zhoršenou rozlišovací schopností a jeden čas pak bude možné zaznamenat jen každých 20 setin.

Využití automatických filtrů pro zpracovávání naměřených časů je tedy problematické a v mnoha případech je třeba zdvojené záznamy filtrovat ručně, případně použít cílovou kameru pro analýzu sporných údajů.

### 3.2.2 Vývojový diagram

Program po spuštění nejdříve zkontroluje dostupné sériové porty. Existující porty vloží do pole, ze kterého si uživatel může později vybrat používaný port, který si program pamatuje i při dalších spuštěních. V případě neexistence sériového portu je program ukončen.

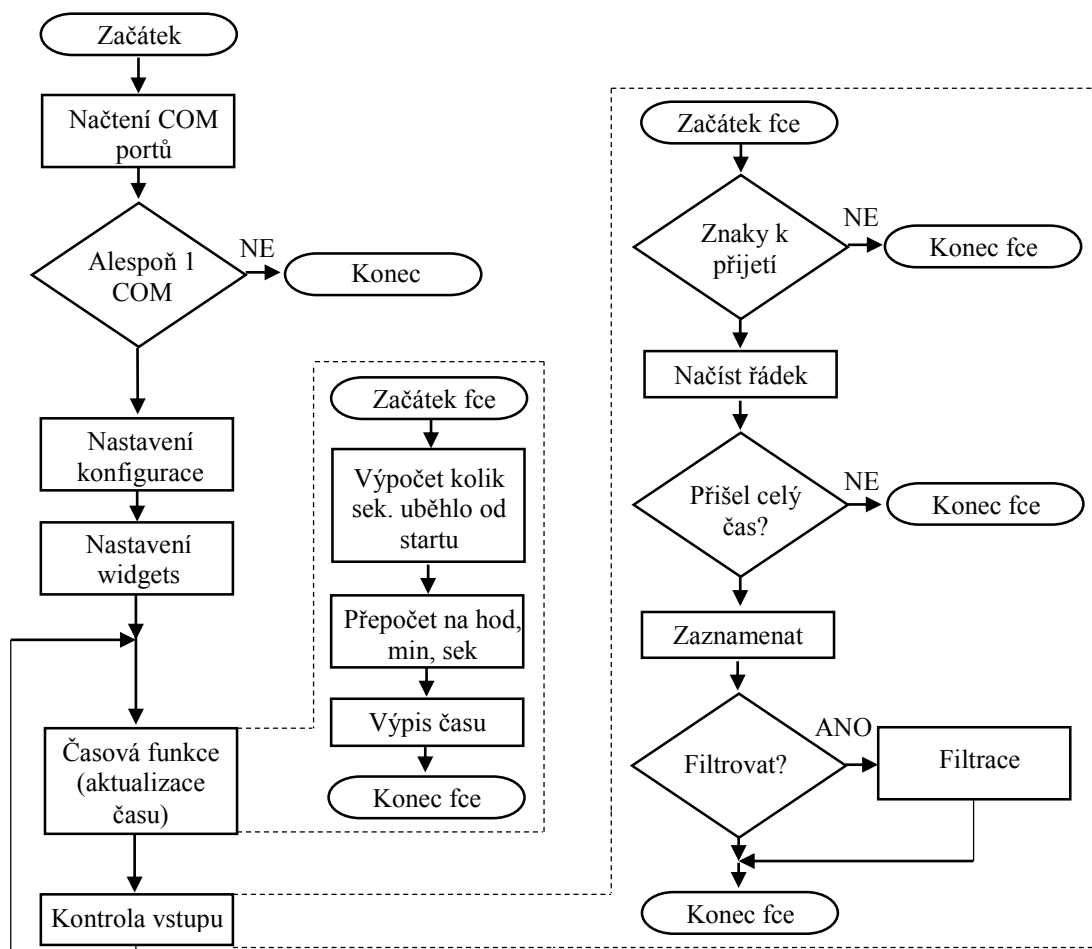
V dalším kroku se načtou nastavení z konfiguračního souboru, vytvoří se potřebné základní proměnné a nastaví se klávesové bindy. Program lze totiž ovládat jak tlačítky v aplikaci, tak i klávesovými zkratkami.

Pro vytvoření uživatelského rozhraní je třeba nastavit všechny takzvané widgety. Mezi widgety patří například okna, tlačítka, seznamy, menu nebo vstupní pole.

Opakovaně každých 10 ms dochází k volání dvou hlavních funkcí, časové funkce, která ukazuje orientační hodnotu aktuálního času a funkce pro kontrolu a načtení příchozího údaje.

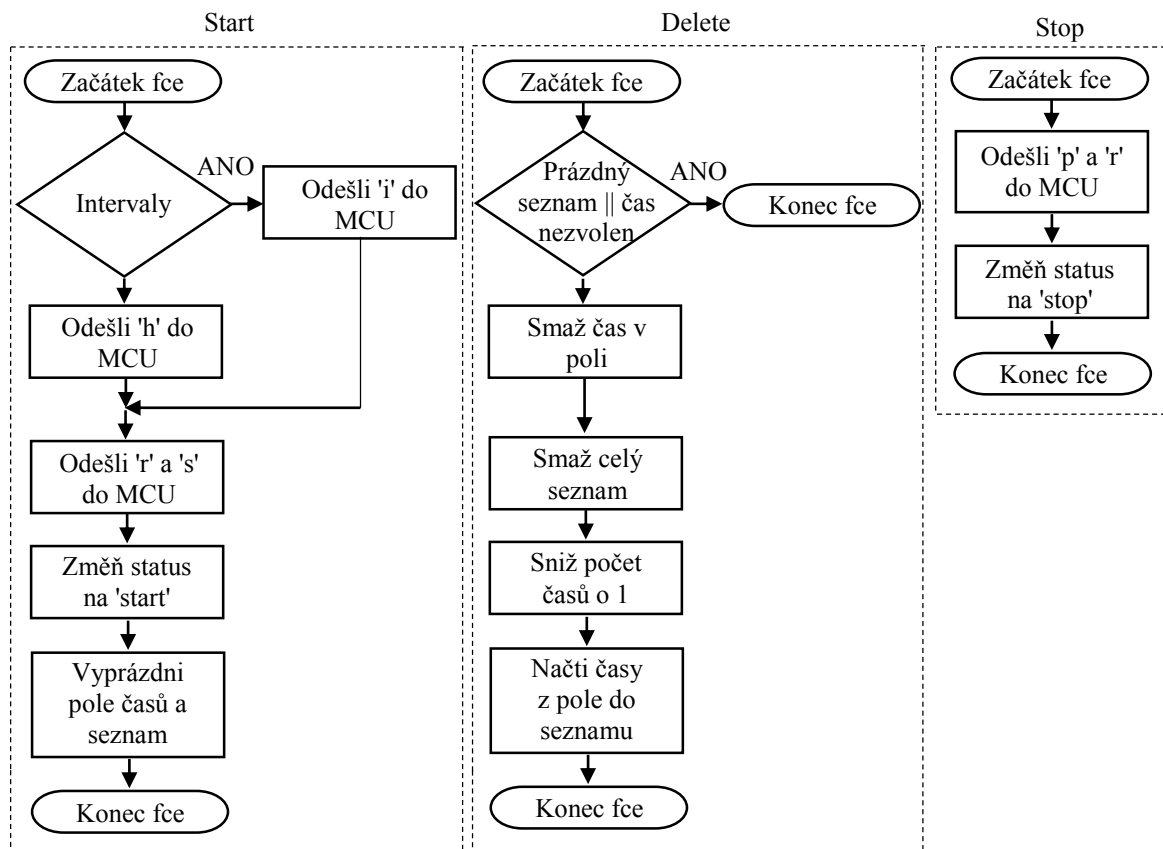
Pokud čekají (ve vstupním bufferu) znaky na přijetí, jsou načteny a následně zpracovány. Program zpracovává pouze celé řetězce začínající a končící mezerou.

Případné šumy v podobě neukončených řetězců jsou zahozeny. Přijatému času program přidělí pořadové číslo, uloží ho do pole přijatých časů a zobrazí ho v seznamu. Se záznamy v seznamu pak může uživatel dále pracovat.



Obr. 33: Zjednodušený vývojový diagram hlavní části programu

Po stisknutí některého z hlavních řídicích tlačítek dochází k volání funkcí, které mají na starosti vykonat danou činnost. Následující vývojový diagram naznačuje činnosti, které jsou vykonávány po stisknutí některého z řídicích tlačítek, konkrétně tlačítek Start, Stop nebo Delete, která jsou pro účely běžeckých závodů nejdůležitější.

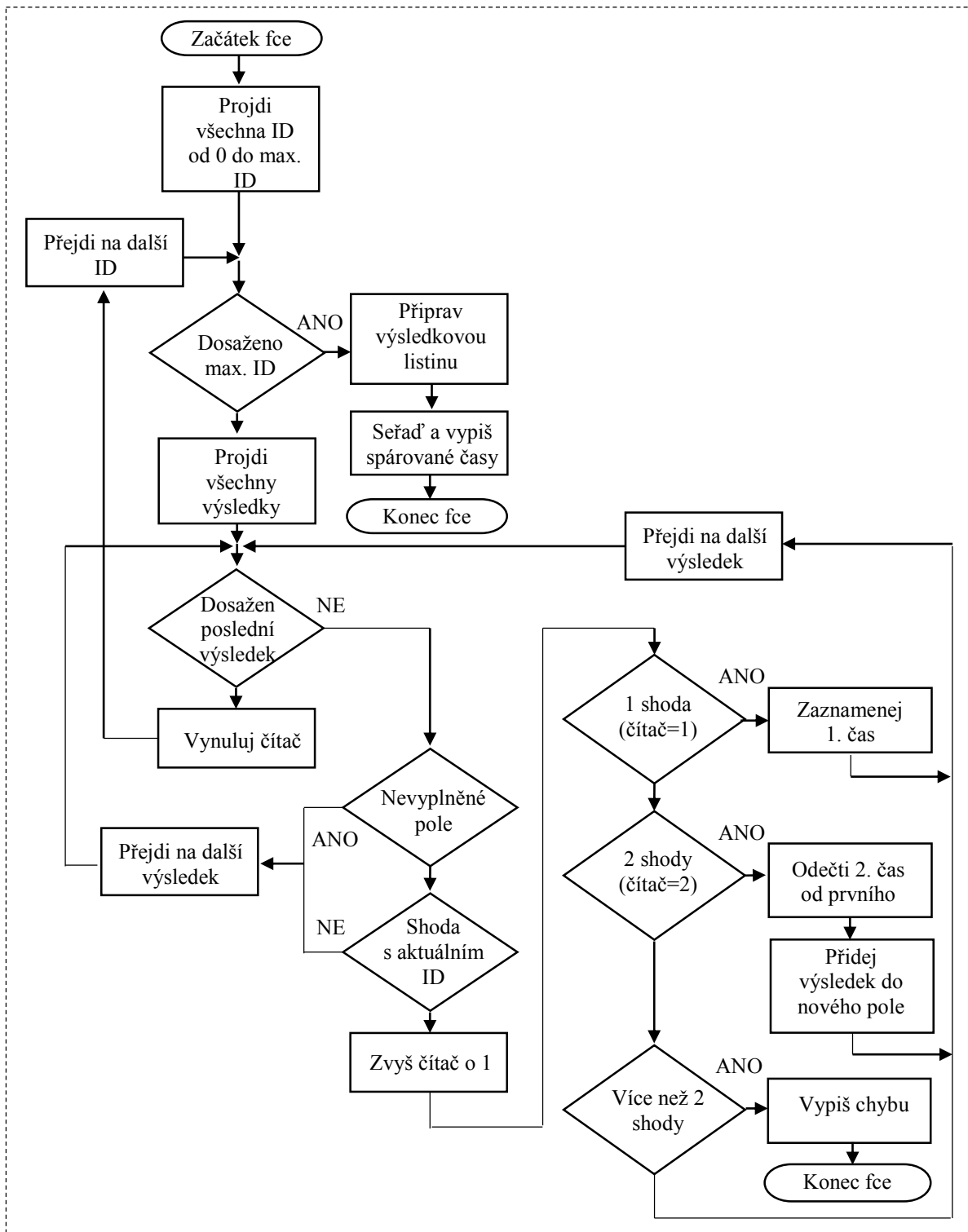


Obr. 34: Vývojové diagramy funkcí tlačítek Start, Delete a Stop

Jak je zřejmé z Obr. 34, při stisknutí tlačítka Start se do mikropočítače odešlou právě tři znaky. Nejdříve se navolí režim, následně se vyresetuje čas a spustí se časovač. V případě stisknutí tlačítka stop se odesílá znak pro pozastavení časovače a následné vyresetování.

Složitějšího algoritmu si lze všimnout u funkce pro párování časů. Funkce zkouší shodu jednotlivých generovaných ID s hodnotou v seznamu přiřazených ID a takto postupně prochází jednotlivé výsledné časy. Při nalezení právě dvou shod, tedy dvou časů se stejným ID funkce, časy odečte a vloží je do nového pole. Po dosažení hodnoty maximálního ID uvedeného uživatelem se připraví výsledková listina a jednotlivé časy s patřičnými ID se seřadí vzestupně.

## Párovat

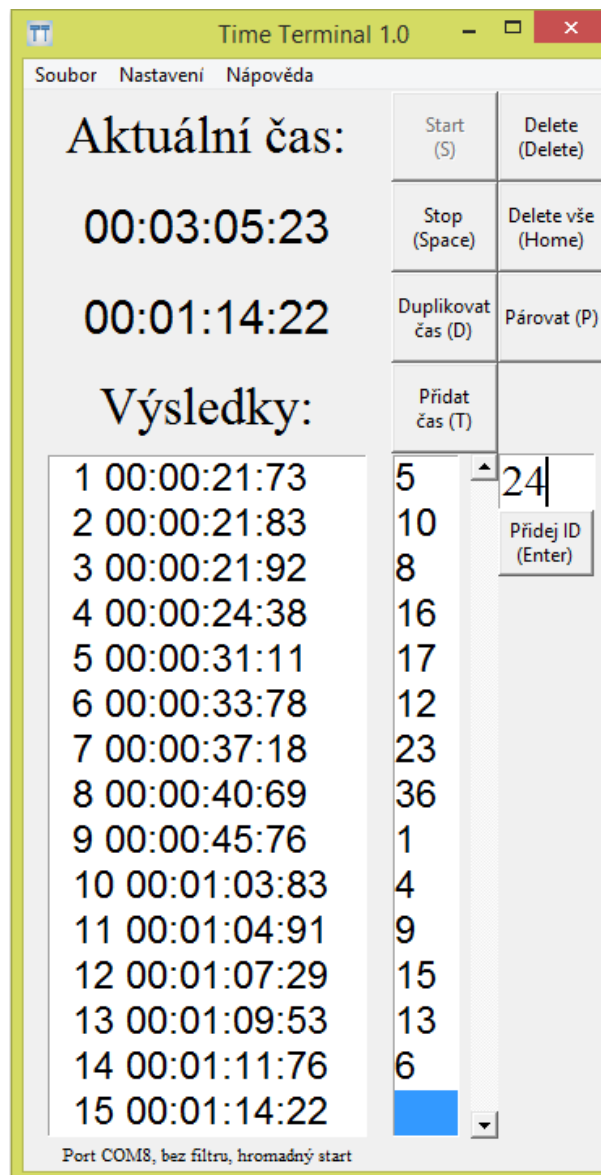


Obr. 35: Vývojový diagram párovací funkce



### 3.2.3 Ukázka použití aplikace Time Terminal

Program Time Terminal je možné spustit v jeho zkompilované verzi pro Windows, pro jejíž spuštění není nutno mít na počítači nainstalován Python. Případně je možné spustit přímo soubor .py , ovšem je třeba mít nainstalován Python verze 3 s patřičnými moduly. Na Obr. 36 je vidět program Time Terminal v provozu.



Obr. 36: Vzhled programu Time Terminal

Nad výsledkovou listinou jsou vidět dva časové údaje. První časový údaj uživatele informuje o aktuálním čase, druhý údaj udává poslední zaznamenaný čas průchodu světelnou závorou. Funkce, která má na starosti generování aktuálního času, se řídí systémovým časem počítače. K jediné synchronizaci s časovým údajem v mikropočítači dochází v okamžiku startu. Vzhledem k tomu, že tento časový údaj je pouze orientační, dalších synchronizací není třeba.

Zaznamenaným časům ve výsledkové listině se automaticky přiřazuje pořadí. V případě smazání nebo duplikace některého z časů se pořadí automaticky aktualizuje.

Exportovat výsledkovou listinu je možné v Soubor / Export časů. Export se provádí do textového souboru, uživatel si může v samostatném dialogovém okně zvolit umístění a název souboru. Volbu COM portu, režimu (hromadný start, intervalové starty) a filtru je možné provést v Nastavení. Program uživatele informuje v hlavním okně pod seznamem časů o zvoleném nastavení. V sekci Nápověda je možné nalézt základní informace o programu a pokyny pro ovládání a nastavení programu.

## Závěr

Úkolem této práce bylo navrhnout zařízení umožňující měření reálného času s vhodným mikroprocesorem. V první části práce byly popsány základní navrhovaného zařízení a byla vysvětlena jejich funkce.

V druhé části práce byla v souladu se stanovenými požadavky na zařízení navržena elektronická, optická i konstrukční část jednotek, která byla vytvořena nad rámec zadání práce. Výsledkem návrhu jsou robustní moduly vysílače a přijímače s nastavitelnou výškou světelné závory, schopné funkce v terénu a bez externího síťového napájení. Modul vysílače je možné napájet z 6x1,5 V baterií a modul přijímače je napájen z USB portu. Modul přijímače byl vybaven indikačními LED diodami a LCD displejem, informujícím uživatele o aktuálním stavu. Pro účely intervalových startů na běžeckých závodech bylo zařízení navíc doplněno možností zvukové signalizace.

Dále byl vytvořen software pro mikroprocesor a pro PC. Software mikroprocesoru se stará o záznam času, jeho zaznamenávání a odesílání do PC, výpis na LCD a dále také o řízení pípání při průběhu závodníka nebo při intervalových startech. Komunikace mezi PC a mikrokontrolérem byla vytvořena na sběrnici UART a odzkoušena pomocí terminálového programu. Pro PC byla navržena a naprogramována uživatelská aplikace Time Terminal sloužící k přehlednému zaznamenávání přijatých časů, jejich editaci a tvorbě výsledkových listin.

Vzniklé zařízení jako celek bylo navrženo, sestaveno a správná funkce odzkoušena za různých provozních podmínek. Navržené zařízení je použitelné na běžeckých trénincích nebo méně náročných závodech jako náhrada za manuální záznam času stopkami. S drobnými úpravami lze zařízení použít i pro záznam časů a počtu prošlých výrobků na lince v průmyslu.

### Hlavní přínosy práce:

- 1) V práci byla navržena a popsána koncepce zařízení pro měření času průchodu osob nebo předmětů světelnou závorou a byla popsána funkce jednotlivých bloků.

- 2) Byla navržena hardwarová část zařízení včetně konstrukční části. Pro moduly byly navrženy desky plošných spojů.
- 3) Byla navržena softwarová část zařízení, která zahrnuje program mikroprocesoru a uživatelský program Time Terminal pro PC.
- 4) Navržené zařízení bylo zkonstruováno a ověřeno na správnou funkci.

### **Perspektivní směry dalšího vývoje:**

Zařízení je možné dále vylepšovat a upravovat. Mezi další možná vylepšení patří zejména:

- 1) Přijímač nezávislý na PC, vybavený vlastním napájením a s ukládáním časů do interní paměti.
- 2) Vybavení přijímače Wi-Fi konektivitou.
- 3) Vytvoření uživatelského programu pro OS Android pro ovládání pomocí mobilního telefonu nebo tabletu.

## Seznam literatury a informačních zdrojů

- [1] LM217, LM317 1.2 V to 37 V adjustable voltage regulators. [online]. 2014 [citováno 2014-05-09]. Dostupné z <http://www.farnell.com/datasheets/1815554.pdf>
- [2] AC Line powered LEDs [online]. [citováno 2014-09-27]. Dostupné z <http://www.bowdenshobbycircuits.info/page10.htm> .
- [3] Photo resistor [online]. [citováno 2014-09-27]. Dostupné z <http://www.resistorguide.com/photoresistor/> .
- [4] HW server představuje - Sériová linka RS-232. [online]. 2005 [citováno 2014-04-17]. Dostupné z <http://www.hw.cz/rozhrani/hw-server-predstavuje-seriova-linka-rs-232.html> .
- [5] 650 LASER MODULE. [online]. [citováno 2014-09-27]. Dostupné z <http://www.gme.cz/img/cache/doc/759/040/f-laser-5mw-datasheet-1.pdf> .
- [6] Modrý laser 1W 445nm. [online]. 2011 [citováno 2014-09-27]. Dostupné z <http://danyk.cz/laser5.html> .
- [7] MSP430G2x53, MSP430G2x13 Mixed Signal Microcontroller (Rev. J). [online]. 2013 [citováno 2014-09-27]. Dostupné z <http://www.ti.com/lit/ds/symlink/msp430g2553.pdf> .
- [8] TI LaunchPad. MSP-EXP430G2 [online]. [citováno 2014-09-27]. Dostupné z <http://www.ti.com/ww/en/launchpad/launchpads-msp430-msp-exp430g2.html#tabs>
- [9] Photo-Diode-Pin Bpw34 Osr. [online]. [citováno 2014-09-27]. Dostupné z <http://www.voelkner.de/products/142663/Photo-Diode-Pin-Bpw34-Osr.html> .
- [10] VISHAY SEMICONDUCTORS BPW34, BPW34S. [online]. 2011 [citováno 2014-09-27]. Dostupné z <http://www.vishay.com/docs/81521/bpw34.pdf> .
- [11] 8 x 2 Alphanumeric LCD Displays. [online]. [citováno 2014-09-27]. Dostupné z <http://www.rapidonline.com/electronic-components/8-x-2-alphanumeric-lcd-displays-500158>.
- [12] Hama "Star 5" Tripod. [online]. [citováno 2014-09-27]. Dostupné z <https://de.hama.com/00004105/hama-star-5-tripod> .
- [13] Interface MSP430 Launchpad with LCD Module (LCM) in 4 bit mode. [online]. 2011 [citováno 2014-09-27]. Dostupné z <http://cacheattack.blogspot.cz/2011/06/quick-overview-on-interfacing-msp430.html>
- [14] Using ACLK and the 32kHz Crystal. NJC's MSP430 Launchpad Blog [online]. 2012 [citováno 2014-09-27]. Dostupné z <http://www.msp430launchpad.com/2012/03/using-aclk-and-32khz-crystal.html> .
- [15] Tutorial 12: Making Comparisons. Scientific instruments using the TI MSP430 [online]. 2010 [citováno 2014-09-27]. Dostupné z <http://mspsci.blogspot.cz/2010/12/tutorial-12-making-comparisons.html> .
- [16] 555 Astable Circuit Calculator. [online]. [citováno 2014-09-27]. Dostupné z [http://www.ohmslawcalculator.com/555\\_astable.php](http://www.ohmslawcalculator.com/555_astable.php) .
- [17] MSP430x2xx Family User's Guide (Rev. J). [online]. 2013 [citováno 2014-09-27]. Dostupné z <http://www.ti.com/lit/ug/slau144j/slau144j.pdf> .

## Přílohy

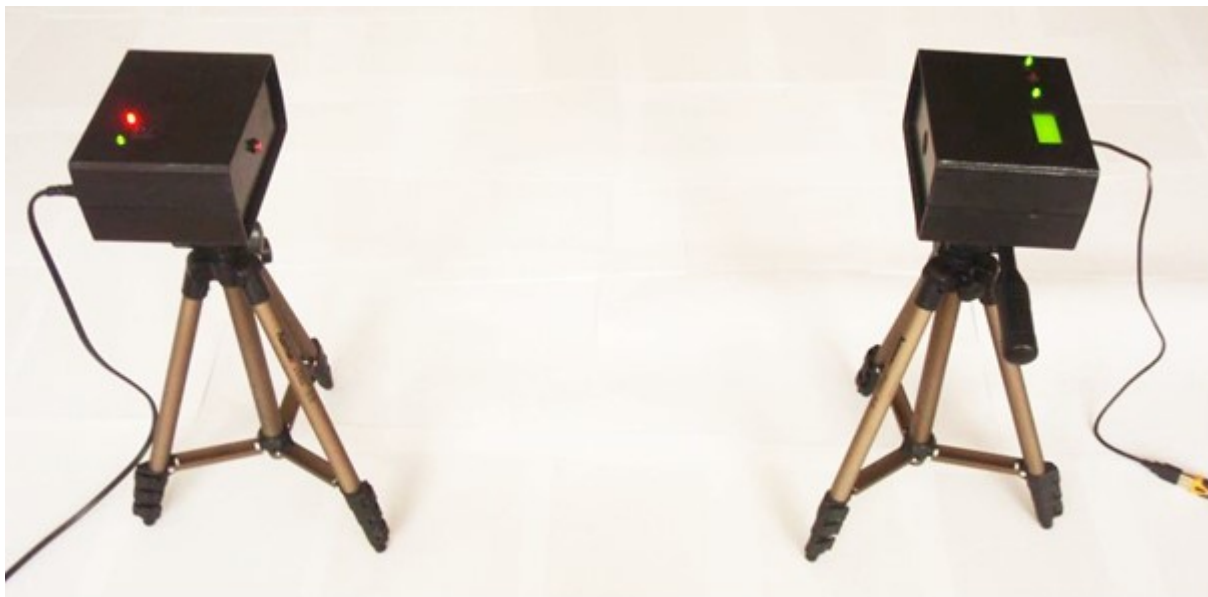
### Příloha A – Modul vysílače v provozu



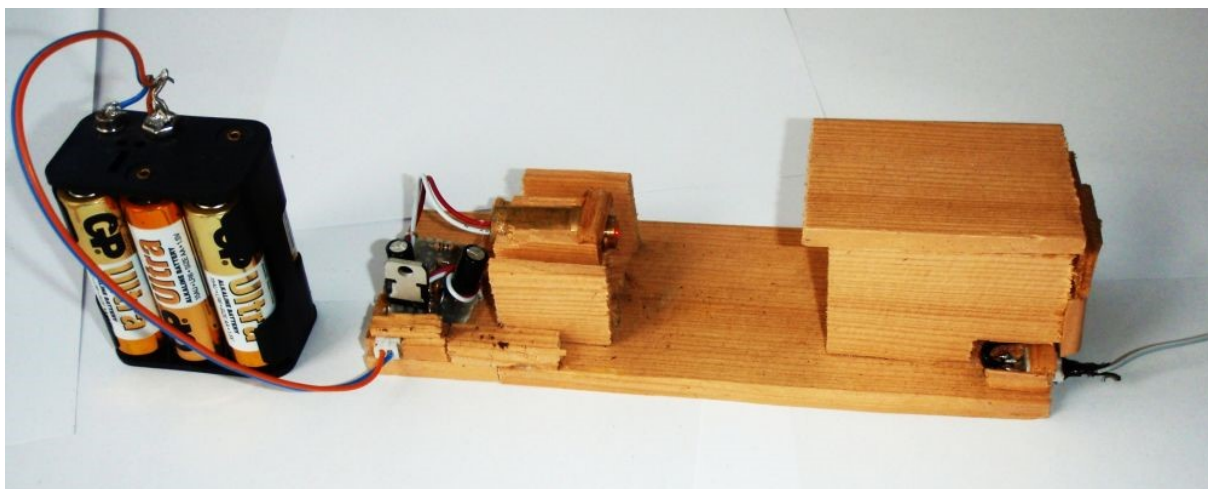
## Příloha B – Modul přijímače v provozu



### Příloha C – Obě jednotky v provozu

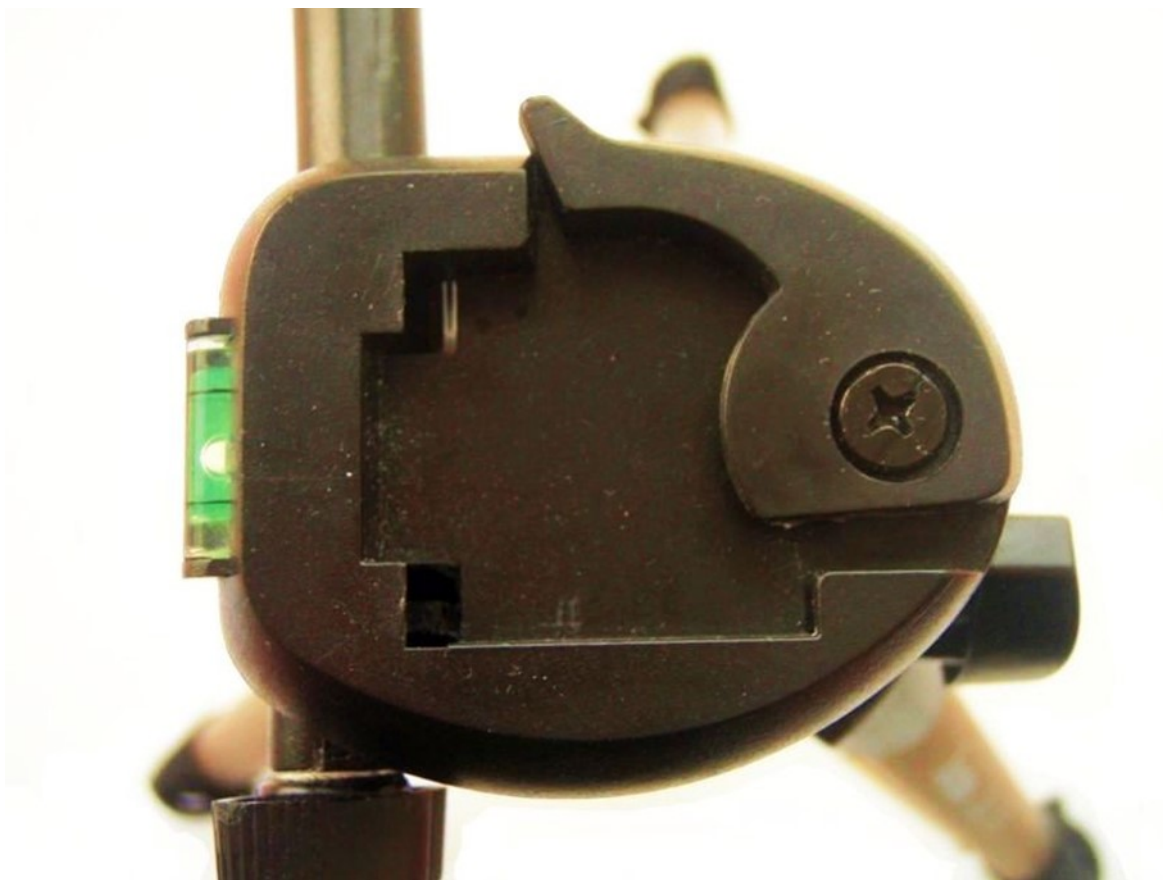


### Příloha D – Testovací přípravek

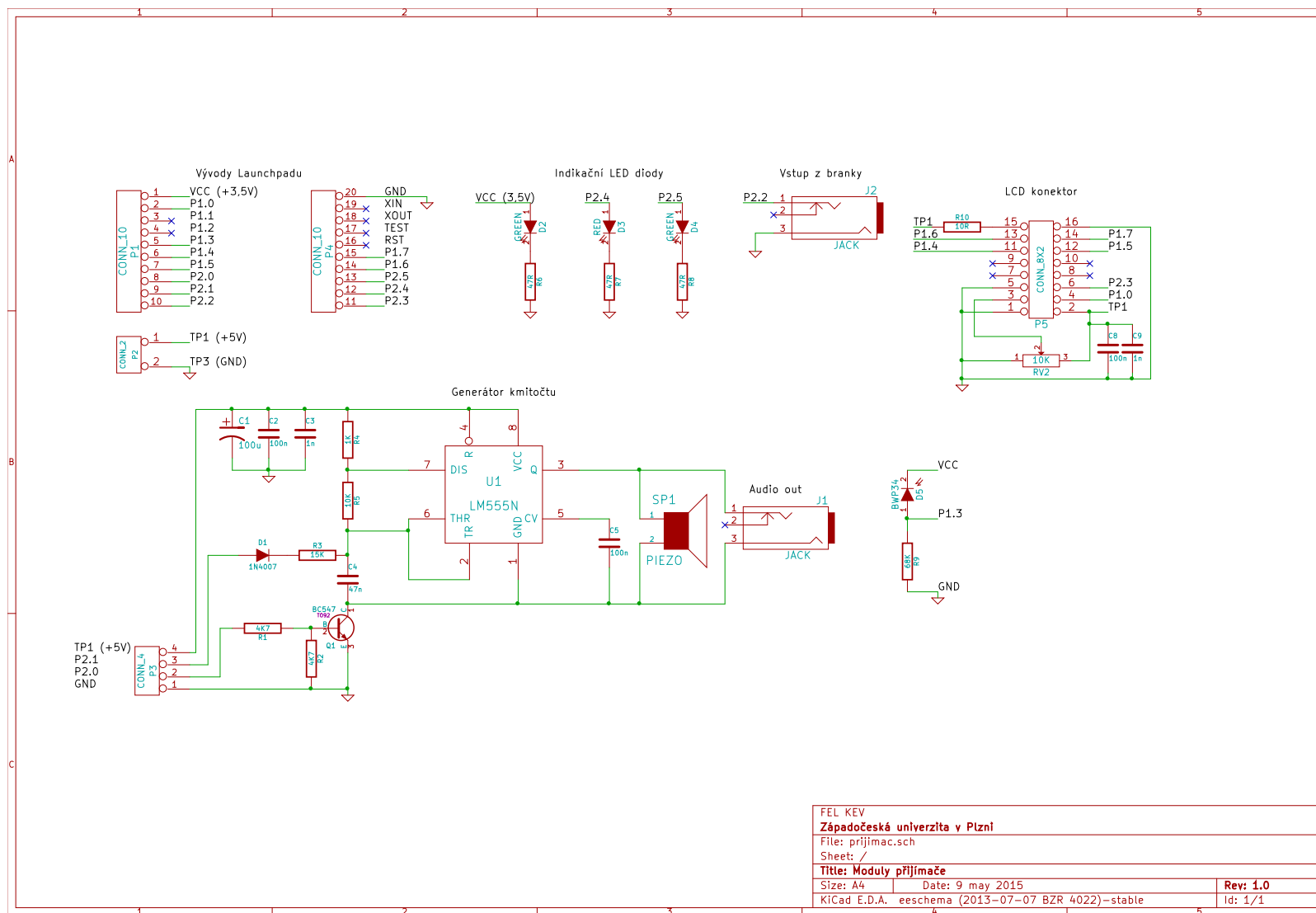




**Příloha E - Systém uchycení aretační destičky na stativ**

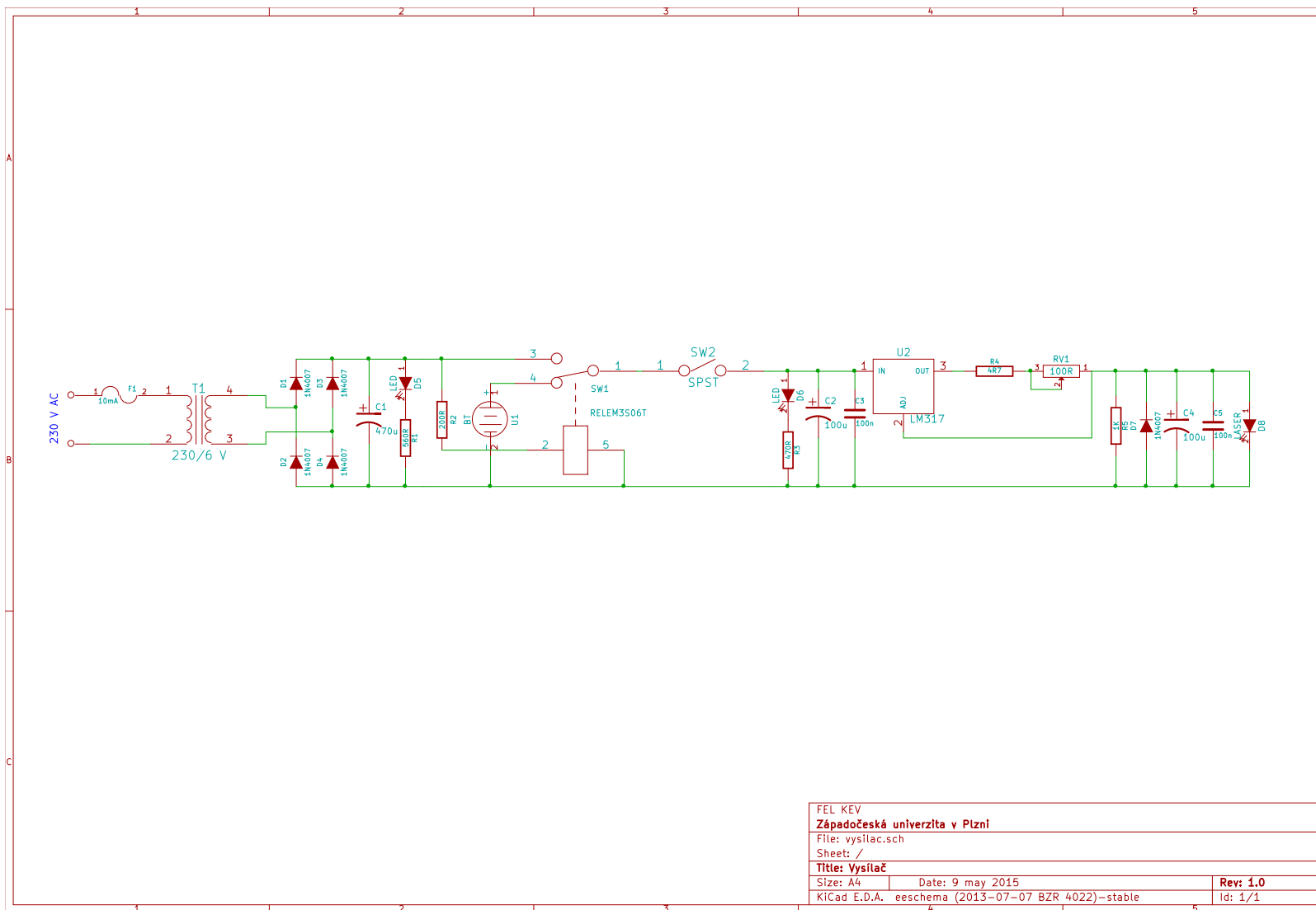


### Příloha F – Schéma zapojení modulu přijímače



FEL KEV		
Západočeská univerzita v Plzni		
File: prijimac.sch		
Sheet: /		
Title: Modul přijímače		
Size: A4	Date: 9 may 2015	Rev: 1.0
KiCad E.D.A. eeschema (2013-07-07 BZR 4022)-stable		Id: 1/1

**Příloha G – Schéma zapojení modulu vysílače**



FEL KEV		
Západočeská univerzita v Plzni		
File: vysilac.sch		
Sheet: /		
Title: Vysílač		
Size: A4	Date: 9 may 2015	Rev: 1.0
KiCad E.D.A. eeschema (2013-07-07 BZR 4022)-stable		Id: 1/1

