

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Automatizovaný sběr dat pomocí samohybného robota

Aleš Doležal

Bakalářská práce

2011

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2010/2011

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Aleš DOLEŽAL**
Osobní číslo: **I07510**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Komunikační a mikroprocesorová technika**
Název tématu: **Automatizovaný sběr dat pomocí samohybného robota**
Zadávající katedra: **Katedra elektrotechniky**

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

Teoretická část bude obsahovat problematiku sběru a statistického zpracování dat, prokládání funkcemi atd.

V praktické části bude realizováno měření světelného nebo tepelného pole za pomoci autonomního robota propojeného s počítačem. Robot měří data a počítač rozhoduje v kterých místech je potřeba měřit. (měřící robot fyzicky existuje, je potřeba propojit s počítačem a vytvořit aplikaci pro ukládání a zpracování dat C).

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

Mobilní roboty Novák P.; BEN - technická literatura, 2005, ISBN: 80-7300-141-1

Měřicí technika; Koropečká H.; Pardubice 1989, ISBN 80-85113-05-8

Myslíme v jazyku C, Tom Archem; Grada Publishing, ISBN 80-247-0301-7

Vedoucí bakalářské práce:

Ing. Pavel Rozsival
Katedra elektrotechniky

Datum zadání bakalářské práce: **16. prosince 2010**

Termín odevzdání bakalářské práce: **13. května 2011**



prof. Ing. Simeon Karamazov, Dr.
děkan



L.S.



Ing. Zdeněk Němec, Ph.D.
vedoucí katedry

V Pardubicích dne 31. března 2011

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 13. 5. 2011

Aleš Doležal

Poděkování

Tímto bych rád poděkoval rodině a mému blízkému okolí za podporu a případnou radu při průběhu zpracování bakalářské práce.

Anotace

Cílem této práce bylo proměření světelného nebo teplotního pole za pomoci autonomního robota, který řídí mikroprocesor propojený s počítačem. Robot proměří pole obdélníkového tvaru senzorem např. pomocí fotorezistoru, fotodiody, termistoru. Naměřená data odešle do počítače přes sériový port, kde se tyto data zpracují a vyhodnotí průběh měření.

Klíčová slova

intenzita světla, teplota, fotorezistor, sériový port, robot

Title

Automatic data acquisition with autonomous robot

Annotation

The aim of this study was to measure the intensity of heat or light field using autonomous robot, controlled by microprocessor connected to a computer. Robot measures light or heat field using photoresistor, photodiode or thermistor. Measured data are sent to a computer via serial port, where the data processed and evaluated during the measurement.

Keywords

Light intensity, temperature, photoresistor, serial port, robot

Obsah

Seznam zkratk	9
Seznam obrázků	10
Seznam tabulek	11
1 Úvodní informace	12
2 Sběr dat	13
2.1 Měření intenzity světla	13
2.1.1 Fotorezistor	14
2.1.2 Fotodioda	14
2.2 Měření teploty	15
2.2.1 Kovové odporové snímače teploty	15
2.2.2 Polovodičové odporové snímače teploty	16
2.2.3 Termoelektrické snímače teploty.....	17
3 Robot Boe Bot	18
3.1 O robotu.....	18
3.2 Obsah stavebnice	19
3.2.1 Základní modul.....	19
3.2.2 Základní deska	19
3.3 Basic Stamp Editor v2.5(dále jen Basic)	22
3.3.1 Pohyb robota.....	25
3.3.2 Měření světla za pomoci fotorezistoru	27
4 Program robota	28
4.1 Vývojový diagram	29
4.2 Navigace robota.....	30
4.2.1 Trasa robota bez proměření	30
4.2.2 Mezi dvěma hlavními měřeními na řádku, bez nutnosti proměření	31
4.2.3 Proměření na řádku mezi dvěma hlavními měřeními.....	31
4.2.4 Pravá zatáčka bez proměřování	32
4.2.5 Pravá zatáčka s proměřováním	32
4.2.6 Levá zatáčka bez proměření	33
4.2.7 Levá zatáčka s proměřováním	33
5 Uživatelský program	35

5.1	Vývojový diagram	35
5.2	Ovládání uživatelského programu	36
5.3	Uložení dat.....	37
5.4	Vykreslení plochy.....	40
5.4.1	Nastavení barvy RGB	40
5.4.2	Řádek buňky	41
5.4.3	Buňky	42
5.4.4	Možnosti vykreslení plochy	43
6	Naměřené hodnoty a grafy	45
7	Závěr	48
	Literatura	49
	Příloha A – Obsah CD.....	50

Seznam zkratek

LED	Light-Emitting Diode
PN	Positive-Negative
PIN	Positive-Intrinsic-Negative
NTC	Negative Temperature Coefficient
PTC	Positive Temperature Coefficient
RAM	Random access memory
EEPROM	Electrically Erasable Programmable Read-Only Memory
PC	Personal Computer
RGB	Red Green Blue
COM	Hardware Interface
PIXEL	Picture Element

Seznam obrázků

Obrázek 1 – Pasivní polovodičové součástky	13
Obrázek 2 – Schématické značky	13
Obrázek 3 – Schéma zapojení fotorezistoru	14
Obrázek 4 – Fotorezistor	14
Obrázek 5 – Schéma zapojení fotodiody (odporové)	15
Obrázek 6 – Schéma zapojení termistoru NTC	16
Obrázek 7 – Schéma zapojení termistoru PTC.....	16
Obrázek 8 – Robot Boe Bot.....	18
Obrázek 9 – Základní modul	19
Obrázek 10 – Základní deska	20
Obrázek 11 – Podvozek.....	20
Obrázek 12 – Servo	21
Obrázek 13 – Box na baterie	21
Obrázek 14 – Kolečko	21
Obrázek 15 – Kulička.....	21
Obrázek 16 – Vycentrované servo.....	23
Obrázek 17 – Nejvyšší rychlost otáčení serva po směru hodinových ručiček	23
Obrázek 18 – Nejvyšší rychlost otáčení serva proti směru hodinových ručiček.....	23
Obrázek 19 – Servo ladící šroubek.....	24
Obrázek 20 – Schéma zapojení fotorezistoru	27
Obrázek 21 – Vývojový diagram robota	29
Obrázek 22 – Trasa robota.....	31
Obrázek 23 – Proměření mezi 2 hlavními měřeními.....	32
Obrázek 24 – Robot v zatáčce	34
Obrázek 25 – Vývojový diagram C#.....	35
Obrázek 26 – Uživatelský program	36
Obrázek 27 – Spektrum barev zobrazitelné v grafu	37
Obrázek 28 – Matice "Dej", stejně dlouhé řádky	38
Obrázek 29 – Matice "Dej", různě dlouhé řádky.....	38
Obrázek 30 – Hodnota < 255.....	41
Obrázek 31 – Hodnota < 510, hodnota > 255	41
Obrázek 32 – Hodnota > 509.....	41
Obrázek 33 – Možné tvary buňky	43
Obrázek 34 – Buňky bez proměření	43
Obrázek 35 – Buňky, proměřené řádky.....	44
Obrázek 36 – Buňky, proměřené sloupce.....	44
Obrázek 37 – 1. proměřená plocha.....	45
Obrázek 38 – Graf 1. proměřené plochy	45
Obrázek 39 – 2. proměřená plocha.....	46
Obrázek 40 – Graf 2. proměřená plocha.....	46
Obrázek 41 – 3. proměřená plocha.....	47

Obrázek 42 – Graf 3. proměřená plocha.....	47
--	----

Seznam tabulek

Tabulka 1 – Stejně řádky	39
Tabulka 2 – Řádky jsou různě dlouhé	39

1 Úvodní informace

Úkolem této bakalářské práce bylo realizování měření světelného nebo tepelného pole za pomoci autonomního robota propojeného s počítačem. Součástí robota je Mikrokontroler určující směr trasy a místa, kde se proměří intenzita dopadajícího světla za pomoci fotorezistoru. Naměřená data se odesílají po sériové lince do počítače, který data zpracuje a použije pro zobrazení grafu.

Druhá kapitola je věnována možnostem sběru dat. Data získáme změřením intenzity světla nebo teploty. Intenzita světla je měřena polovodičovými součástkami, a to fotorezistorem, fotodiodou. Teplota je měřena termistory, termoelektrickými snímači.

Třetí kapitola představuje robota. Popisuje jednotlivé části, ze kterých je robot sestaven a způsob jeho ovládání. Dále se zabývá měřením a odesíláním těchto hodnot do počítače.

Ve čtvrté kapitole je popsána funkce a chování robota při navigaci do bodů měření. Navigace má 16 hlavních bodů měření se standardní vzdáleností od sebe, po každém 4. hlavním bodě měření následuje zatáčka o 180°. V případě proměrování na řádcích může robot proměřit až 52 bodů. Mezi dvěma hlavními body na řádku mohou být až 3 proměření. Zatáčka je tvořena dvěma hlavními body se standardní vzdáleností od sebe, což jsou krajní hodnoty řádků. Proměření zatáčky způsobí, že řádky mezi sebou budou mít menší vzdálenost.

Pátá kapitola obsahuje ovládání uživatelského programu. Dále způsob ukládání dat pro následující zpracování. Barevné složky RGB se používají pro nastavení barvy¹ prvního sloupce buňky, kde první a poslední hodnota sloupce jsou naměřené hodnoty. Řádky buňky se vykreslí po pixelech. Po jednotlivých řádcích se vykreslí buňky, ze kterých se skládá plocha. Proměřený řádek zmenší buňky, ale zvýší jejich počet, vzdálenost mezi hlavními body měření je nezměněna. Proměřená zatáčka sníží výšku buňky a celé plochy grafu.

Šestá kapitola obsahuje fotky měřené plochy a naměřené hodnoty s grafy. Na fotkách je znázorněn základní tvar plochy s jednotlivými řádky bílými samolepkami.

¹ Barva může být v rozmezí červená – modrá – černá

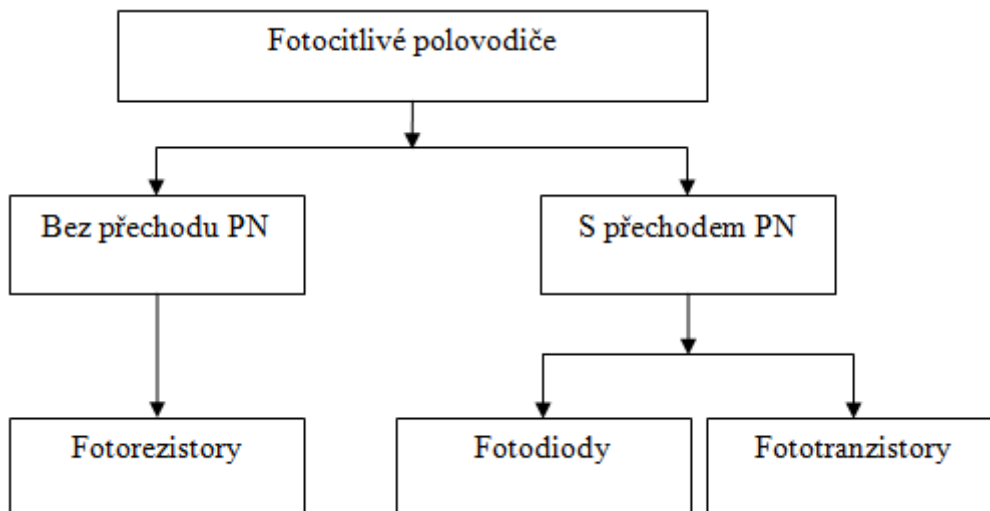
2 Sběr dat

2.1 Měření intenzity světla

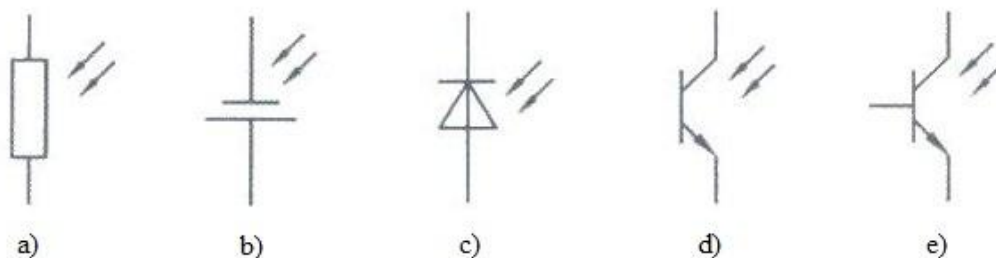
Světlo je elektromagnetické vlnění, jeho největším přirozeným zdrojem je slunce. Umělý zdroj světla může být LED, žárovka, zářivka. Elektromagnetické vlnění, které lidské oko vnímá je pouze nepatrná část ze širokého spektra frekvencí (Frohn, a další, 2006).

Intenzita světelného záření dopadající na polovodičové senzory zvýší vodivost polovodičového materiálu. Senzory se dělí na aktivní a pasivní, schopné pracovat jak v oblastech viditelného světla, tak i v oblastech mimo viditelné světlo.

Aktivní senzory přeměňují světelnou energii v energii elektrickou. Mezi aktivní senzory patří fotoelektrické články a solární články. Mezi pasivní senzory (Obrázek 1) patří fotorezistory (Obrázek 2a), fotodiody (Obrázek 2c) a fototranzistory (Obrázek 2d,e).



Obrázek 1 – Pasivní polovodičové součástky



Obrázek 2 – Schématické značky

Polovodičové materiály mají různou vlastní vodivost, která se mění se vzrůstající teplotou nebo zvyšováním dopadajícího světelného záření. Zvyšováním tepla nebo světla dochází k uvolnění valenčních elektronů z krystalové mřížky polovodiče, což zvýší jeho vodivost. Diody a tranzistory mají pouzdra, která nejsou pro světlo prostupná, aby u těchto

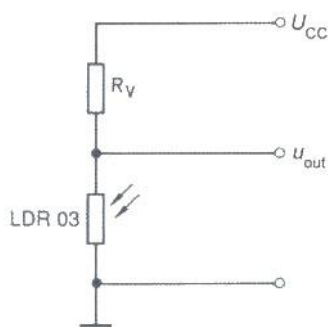
prvků nepůsobil rušivě vnitřní fotoelektrický jev. Ostatní fotoelektrické prvky mají pouzdra s okénkem, aby mohlo světlo účinně působit na polovodičový materiál.

2.1.1 Fotorezistor

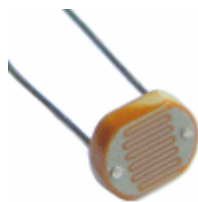
Fotorezistory jsou tvořeny směsí krystalů polovodičů. Funkce fotorezistorů není závislá na polaritě připojeného napětí, protože fotorezistor nemá žádný přechod PN. Právě proto lze fotorezistory zapojovat do stejnosměrných i střídavých obvodů (Obrázek 3). Světlo dopadající na fotorezistor uvolní valenční elektrony, které zvýší vodivost a sníží odpor polovodičového materiálu.

Fotorezistory pracují ve spektrální oblasti viditelného světla. Pro výrobu se používá sulfid kadmnatý (CdS) a selenid kadmnatý (CdSe).

Typické provedení fotorezistoru s meandrovitým průběhem (Obrázek 4).



Obrázek 3 – Schéma zapojení fotorezistoru



Obrázek 4 – Fotorezistor

Možné použití:

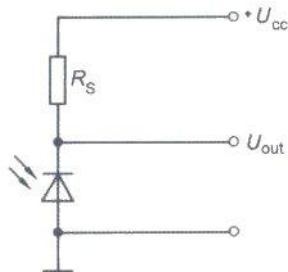
- měření osvětlení expozimetry (fotografické přístroje),
- ovládání přístrojů fotobuňkou (ovládání dveří),
- zabezpečovací zařízení.

2.1.2 Fotodioda

Fotodiody mají dva typy provedení s přechodem PN a PIN. Přechodem PN bez osvětlení protéká proud v závěrném směru I_{R0} . Je-li přechod PN osvětlen, dojde k roztrhání vazeb v krystalu. Vzniknou pohyblivé páry elektron – díra, které působením elektrického pole ihned odtékají. Díry putují k oblasti P a elektrony k oblasti N. Diodou začne protékat

proud I_{ph} , který lineárně narůstá s dopadajícím světlem. Fotodiody jsou zapojeny podle obrázku (Obrázek 5).

Fotodiody jsou vyráběny z křemíku.



Obrázek 5 – Schéma zapojení fotodiody (odporové)

Možné použití:

- V odporovém zapojení, použití stejné jako u fotorezistoru,
- V hradlovém zapojení, fotodioda se chová jako zdroj elektrické energie.

2.2 Měření teploty

Senzory pro měření teploty, pracující na různých fyzikálních principech. Podle požadavků a vlastností je zvolen správný senzor (Frohn, a další, 2006).

2.2.1 Kovové odporové snímače teploty

Odporové snímače jsou vyráběny z materiálů jako je platina (Pt), nikl (Ni) a iridium (Ir). U všech odporových senzorů je téměř lineární závislost odporu materiálu se širokým rozsahem teplot.

Na keramická tělíska jsou navinuty vodiče o průměru od 0,05 mm do 0,3 mm. Nosná destička obsahuje vrstvičku materiálu senzoru. Tento materiál je zapouzdřen nebo pokryt ochrannou vrstvou, která ho ochrání před okolními vlivy. Vinutí může být lineární nebo bifilární.

Měření teploty pomocí odporových snímačů je založeno na měření ohmického odporu senzoru. Jelikož se jedná o pasivní prvky, tak je potřeba zdroj napětí. Proud procházející obvodem senzor ohřívá, což je chyba.

Teplotní rozsah:

- platina -250 °C ... +1000 °C,
- nikl -60 °C ... +250 °C,
- iridium -200 °C ... +600 °C.

Elektrický odpor kovových vodičů vzrůstá s teplotou.

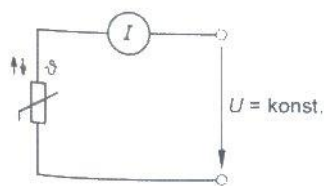
2.2.2 Polovodičové odporové snímače teploty

Termistory se vyrábí z polykrystalické keramiky. S rostoucí teplotou odpor polovodiče klesá. Polovodiče vykazují vyšší citlivost než kovové.

Termistory NTC

Změna odporu s teplotou je u termistorů větší, než u kovových vodičů, což je vlastnost určená výrobou termistorů.

Pokud je termistor elektricky pouze málo zatížen, jeho teplota je shodná s okolní teplotou. Termistor je zapojen podle obrázku (Obrázek 6).



Obrázek 6 – Schéma zapojení termistoru NTC

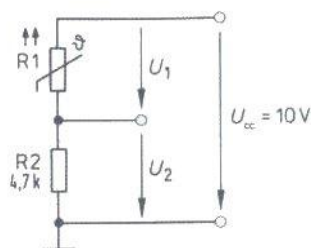
Možné použití:

- kompenzační a měřicí termistory (odpor termistoru určen teplotou okolí) např. měření a regulace teploty v chladničkách, mrazničkách nebo pračkách,
- rozběhové termistory např. k omezení proudových rázů ve výbojkách, žárovkách. Často se používají ke zpomalení průběhu spínání a rozpínání relé.

Tento typ termistorů měří 2 nebo 3 stavy.

Termistory PTC

Termistory se zapojují do obvodu podle následujícího obrázku (Obrázek 7).



Obrázek 7 – Schéma zapojení termistoru PTC

Možnosti použití:

- měření teploty (velmi citlivé senzory),
- aplikace, při nichž dochází ke zvýšení odporu pozistoru vlastním ohřevem procházejícím elektrickým proudem. Např. ochrana proti přetížení, přeplnění.

Křemíkové senzory

Tvoří je křemíková destička s pokovenými ploškami, které představují kontakty. Jedna strana je pokovená celá, druhá strana je pokovená pouze částečně. Křemíkové senzory mají skleněná nebo plastová pouzdra. Používají se k měření teploty.

Integrované polovodičové senzory

Integrované teplotní senzory využívají vlastnosti přechodu PN, dále vyhodnocovací obvody, které na svém výstupu dodávají teplotně závislý proud nebo napětí.

2.2.3 Termoelektrické snímače teploty

Termoelektrický článek je tvořen dvěma vodiči z různých kovových materiálů, které jsou na obou koncích vodivě spojeny. Jestliže teplota měřicího spoje bude různá od teploty srovnávacího spoje, tak vzniká termoelektrické napětí a obvodem prochází proud.

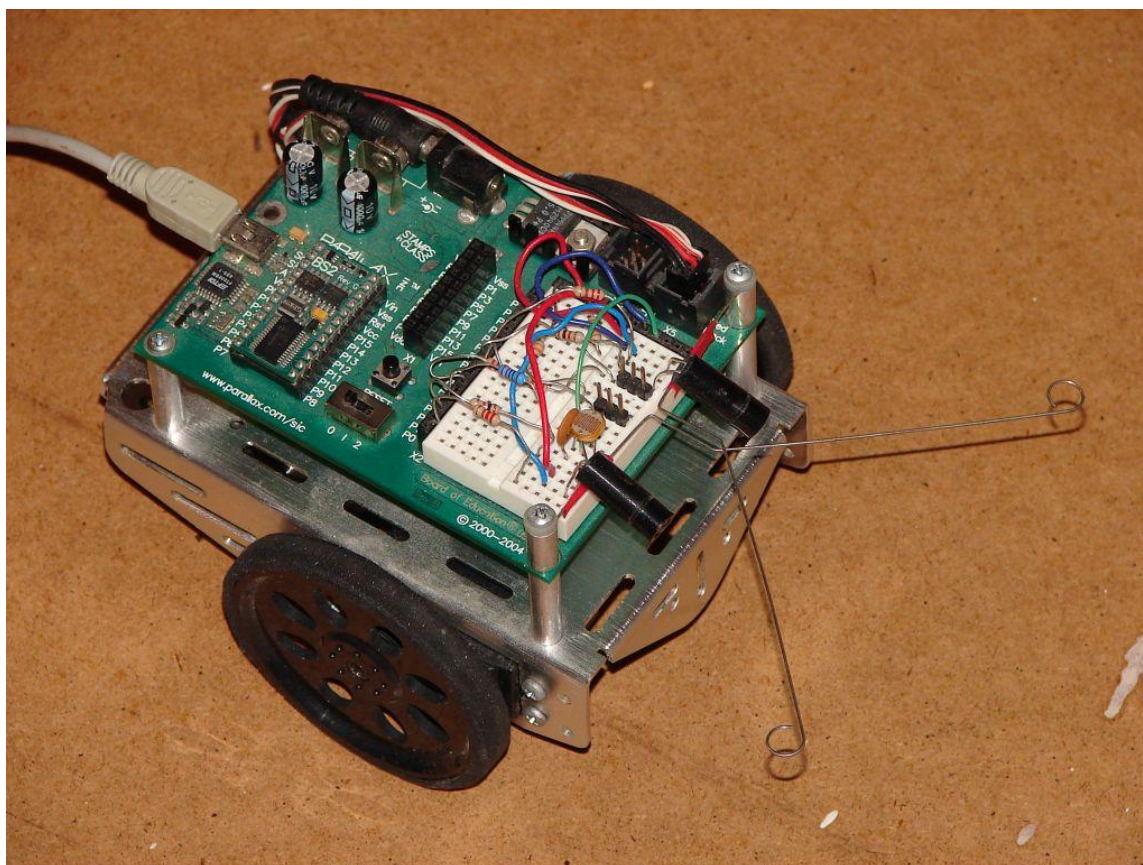
3 Robot Boe Bot

3.1 O robotu

Robota Boe Bot vyrábí společnost Parallax Inc. Jedná se o elektronickou stavebnici, ze které lze postavit programovatelný robot. Stavebnice obsahuje Základní modul, Základní desku, Podvozek, 2 Serva, Box na baterie, 2 Kolečka, Kuličku, Závlečku, šroubky (Parallax, 2010).

Do nosné konstrukce robota přišroubujeme 4 distanční sloupky pro montáž Základní desky, poté přišroubujeme serva k podvozku robota. K servům přišroubujeme kolečka a kuličku uchytkou závlečkou na zadní část podvozku robota. Sestavení robota dokončíme připojením bateriového boxu. Celou kompletaci robota provádíme dle instrukcí výrobce stavebnice uvedených v manuálu.

Pro sestaveného robota použijeme jako zdroj elektrické energie baterie, případně síťový adaptér. Na obrázku (Obrázek 8) je robot vybaven 2 infračervenými diodami, 2 drátovými dotykovými senzory a fotorezistorem.



Obrázek 8 – Robot Boe Bot

Ke komunikaci s robotem programátor používá programový editor “BASIC Stamp Editor v2.5“. Po napsání a kompilaci kódu se nahraje program přes sériový port do paměti

robota EEPROM. Dojde-li k úspěšnému nahrání kódu, měl by být robot schopen vykonávat naprogramované akce.

3.2 Obsah stovebnice

3.2.1 Základní modul

Základní modul (Obrázek 9) je vložen do patice (Parallax, 2010), (Parallax, 2011).

Specifikace:

- Procesor 20 MHz
- RAM 32 B
- EEPROM 2 KB
- Počet I/O pinů: 16 + 2 pro sériový port



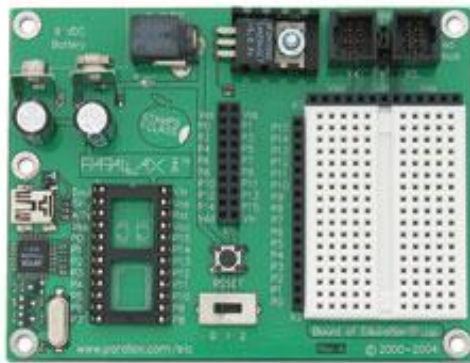
Obrázek 9 – Základní modul

3.2.2 Základní deska

Základní deska (Obrázek 10) je umístěná na podvozku robotu (Parallax, 2010), (Parallax, 2011).

Disponuje následujícím:

- Patice – pro osazení Základním modulem
- Piny P12 a P13 – připojení serv
- Nepájivé pole – připojení senzorů
- Přepínač – poloha:
 - 0 vypnuto
 - 1 aritmetické operace, nahrání kódu
 - 2 pohyb, nahrání kódu
- Reset
- Konektor Přepínač – připojení napájení



Obrázek 10 – Základní deska

Podvozek

Podvozek (Obrázek 11) je nosná konstrukce robota (Parallax, 2011), ke které lze připojit:

- 2 serva,
- 2 kolečka,
- kulička,
- box na baterie,
- základní destičku.



Obrázek 11 – Podvozek

Servo

Serva (Obrázek 12) (Parallax, 2011) slouží jako pohonná jednotka robota.



Obrázek 12 – Servo

Box na baterie

Bateriový box (Parallax, 2011) – pro 4 baterie AA (Obrázek 13).



Obrázek 13 – Box na baterie

Kolečko

Kolečka (Obrázek 14) (Parallax, 2011) slouží pro pohyb robota. Každé kolečko je přišroubované k servu, které je uchyceno v podvozku.



Obrázek 14 – Kolečko

Kulička

Kulička (Obrázek 15) (Parallax, 2011) slouží k udržení stability robota. Kulička je umístěná v zadní části robota.



Obrázek 15 – Kulička

3.3 Basic Stamp Editor v2.5(dále jen Basic)

Programový editor slouží pro zapsání a kompilaci programového kódu (Parallax, 2005), (Parallax, 2010) pro robota. Přehled příkazů editoru:

Podmínky

- 1) IF (true) THEN
...
ELSEIF(true)THEN
ENDIF
- 2) SELECT znak
CASE "F": i = 1
CASE ELSE: i = 0
ENDSELECT

Cykly

- 1) DO...LOOP – je cyklus, který není nikdy ukončený (nekonečná smyčka),
- 2) DO UNTIL (true) ...LOOP – cyklus je ukončen, když je podmínka pravdivá,
- 3) DO...LOOP UNTIL () – cyklus je ukončen, při nesplnění podmínky,
- 4) FOR...NEXT – provede počet cyklů, který má uvedený.

Skoky

- 1) GOTO – skočí na uvedenou adresu,
- 2) GOSUB – skočí na podprogram.

Ukládání do EEPROM

- DATA – ukládá instrukce (navigace) do EEPROM od adresy 0 (hodnota 1, hodnota 2, hodnota N),
- Program – ukládá se od nejvyšší (2047) k nejnižší adrese.

Ovládání serv

```
DO
  PULSOUT 12, 750    \ zastaví servo na Pinu P12
  PAUSE 20           \ 20 ms
LOOP
```

Serva se ovládají příkazem PULSOUT, přičemž první parametr určuje ovládané servo. Druhý parametr směr a rychlost otáčení. PAUSE je čas mezi jednotlivými impulsy.

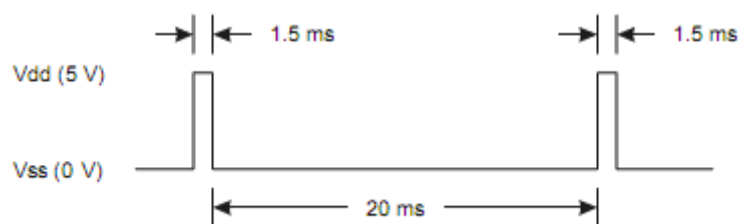
Vzorec pro výpočet doby trvání příkazu:

$$\frac{\text{šířka pulsu}}{2\mu\text{s}} = \text{Trvání příkazu}$$

kde *šířka pulsu* – ovládá rychlost a směr točení serva,

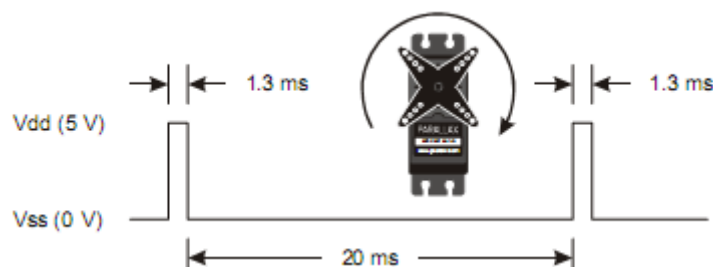
Trvání příkazu – je hodnota, kterou se nastaví směr a rychlost ovládání serva.

Při délce trvání příkazu 750 je šířka pulsu 1,5 ms a pauza 20 ms, perioda trvá 21,5 ms. Vycentrované servo na obrázku (Obrázek 16) je bez jakéhokoliv pohybu, v případě že se pohybuje, je potřeba ho vycentrovat.



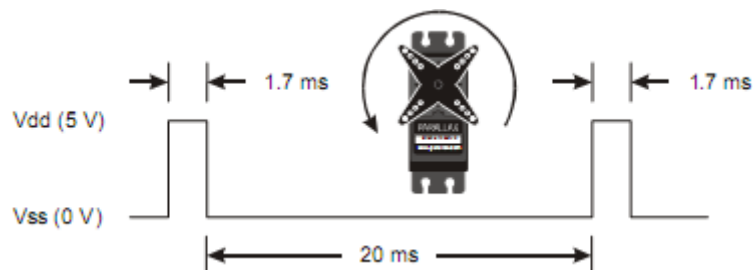
Obrázek 16 – Vycentrované servo

Při délce trvání příkazu 650 je šířka pulsu 1,3 ms a pauza 20 ms, perioda trvá 21,3 ms. Servo se otáčí nejvyšší rychlostí po směru hodinových ručiček (Obrázek 17).



Obrázek 17 – Nejvyšší rychlost otáčení serva po směru hodinových ručiček

Při délce trvání příkazu 850 je šířka pulsu 1,7 ms a pauza 20 ms, perioda trvá 21,7 ms. Servo se otáčí nejvyšší rychlostí proti směru hodinových ručiček (Obrázek 18).



Obrázek 18 – Nejvyšší rychlost otáčení serva proti směru hodinových ručiček

Centrování serva

Důležitou operací než se kompletně fyzicky robot sestaví, je centrování serv. Důvodem je sjednocení rychlosti otáčení, tím zajistíme nejpřesnější možný pohyb robota.

Pro vycentrování serv se vloží základní modul do základní desky, připojí se servo a napájení. K centrování je potřeba nainstalovat do počítače BASIC Stamp Editor v2.5, pomocí něhož se programuje robot.

Do Basicu napíšeme kód:

```
DEBUG "Program Běží!"    \ zobrazí na monitor zprávu

DO
  PULSOUT 12, 750        \ zastaví servo na Pinu P12
  PAUSE 20                \ 20 ms
LOOP
```

Připojíme základní desku k počítači pomocí sériového portu, přepínač posuneme do polohy 2 (zapnutí serv) a nahrajeme kód z PC. Po nahrání kódu dojde k jeho automatickému spuštění, na monitor je vyslaná zpráva “Program Běží“ a zároveň je servům posílán příkaz stát. Jakmile se některé servo pohybuje, provedeme jeho seřízení pomocí ladícího šroubku, který je umístěn nad napájecím kabelem serva (Obrázek 19) (Parallax, 2011).

Po seřízení serv je možno pokračovat v kompletaci robota.



Obrázek 19 – Servo ladící šroubek

3.3.1 Pohyb robota

Vpřed

Pohyb vpřed robot může vykonávat, když levé kolečko má směr otáčení proti hodinovým ručičkám a pravé kolečko po směru hodinových ručiček.

Do Basicu napíšeme kód:

```
DO
    PULSOUT 13, 850    \ proti směru (levé servo)
    PULSOUT 12, 650    \ po směru (pravé servo)
    PAUSE 20           \ 20 ms
LOOP
```

Vzad

Robot se pohybuje vzad, otáčí-li se levé kolečko po směru hodinových ručiček, pravé proti směru hodinových ručiček.

Do Basicu napíšeme kód:

```
DO
    PULSOUT 13, 650    \ po směru (levé servo)
    PULSOUT 12, 850    \ proti směru (pravé servo)
    PAUSE 20           \ 20 ms
LOOP
```

Vlevo

Odbočení vlevo dosáhneme nastavením směru otáčení koleček po směru hodinových ručiček u obou serv.

Do Basicu napíšeme kód:

```
DO
    PULSOUT 13, 650    \ po směru (levé servo)
    PULSOUT 12, 650    \ po směru (pravé servo)
    PAUSE 20           \ 20 ms
LOOP
```

Vpravo

Pro odbočení vpravo se naopak nastaví směr otáčení koleček proti směru hodinových ručiček.

Do Basicu napíšeme kód:

```
DO
    PULSOUT 13, 850    \ proti směru (levé servo)
    PULSOUT 12, 850    \ proti směru (pravé servo)
    PAUSE 20           \ 20 ms
LOOP
```

Stop

Obě kolečka se nastaví na stop, což po vycentrování není žádný problém.

Do Basicu napíšeme kód:

```
DO
  PULSOUT 13, 750  \ zastaví servo (levé servo)
  PULSOUT 12, 750  \ zastaví servo (pravé servo)
  PAUSE 20         \ 20 ms
LOOP
```

Tyto kódy pro uvedení robota do pohybu se vkládají většinou do cyklu FOR, aby se mohl nastavit čas nebo počet cyklů, po který se příkaz má provádět. Pro cykly FOR je nutné nastavit proměnnou v našem případě pojmenovanou “pocitadlo“, která určuje počet cyklů.

Určení času pro vykonání jednoho cyklu je potřeba znát šířky pulzu obou serv, které mohou být v rozmezí od 1,3 ms do 1,7 ms, pauza je dlouhá 20 ms a kódová režie trvá 1,6 ms.

Kde – čas, za který se provede jeden cyklus,
Šířka pulzu P13 – ovládá rychlost a směr točení serva,
Šířka pulzu P12 – ovládá rychlost a směr točení serva,
Pauza – pauza mezi pulzy,
Kódová režie – čas, za který se vykoná smyčka.

Počet cyklů je potřebný k určení vzdálenosti, které má robot docílit. Pro výpočet cyklů je potřeba znát čas, za který robot ujede určenou vzdálenost a čas vykonání jednoho cyklu.

$$\text{počet cyklů} = \frac{\text{čas}}{\text{čas Cyklu}}$$

Kde počet cyklů – počet cyklů, pro vykonání pohybu za určený čas,
– čas po který se bude robot pohybovat,
– čas, za který se provede jeden cyklus.

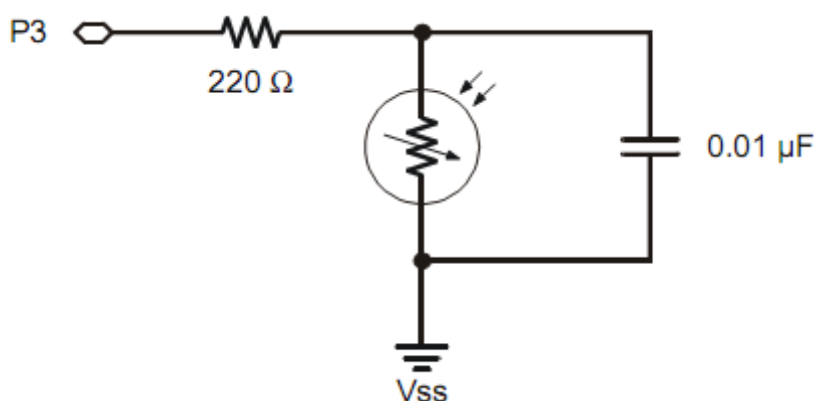
Ukázka kódu pohybu vpřed s cyklem For po dobu 1s:

```
pocitadlo          VAR          Word  \ počet cyklů

FOR pocitadlo = 1 TO 40
  PULSOUT 13, 850  \ proti směru (levé servo)
  PULSOUT 12, 650  \ po směru (pravé servo)
  PAUSE 20         \ 20 ms
NEXT
```

3.3.2 Měření světla za pomoci fotorezistoru

Do pinu P3 je zapojen odpor $220\ \Omega$ a sériově k němu paralelní spojení kondenzátoru $0.01\ \mu\text{F}$ a fotorezistoru (jehož odpor závisí na množství dopadajícího světla), které je připojeno na zem (Obrázek 20) (Parallax, 2010).



Obrázek 20 – Schéma zapojení fotorezistoru

Fotorezistor měří světelné podmínky tak, že čím větší světlo na něj dopadá, tím je menší jeho vnitřní odpor.

Řídící jednotka robota vyhodnotí dobu vybíjení (ms) kondenzátoru. Tato doba je závislá na odporu fotorezistoru. Čím větší intenzita světla dopadá na fotorezistor, tím je doba vybíjení Kondenzátoru menší.

Jakmile robot změří 0 ms, znamená to, že na fotorezistor nedopadá žádné světlo, je v úplné tmě. Při naměření nízkého času vybití je robot na světle, jinak je robot v šeru nebo větší tmě.

Do Basicu napíšeme kód:

```
timeLeft          VAR          Word   \ Pro naměřenou hodnotu

DO
    HIGH 3          \ Nabití kondenzátoru
    PAUSE 2         \ Čas na nabití
    RCTIME 3,1, timeLeft \ Uložení naměřené hodnoty
LOOP
```

Příkaz HIGH zajistí nabití kondenzátoru na 5 V, PAUSE zajišťuje dostatek času na nabití. RCTIME je příkaz, který měří pokles napětí na 1,4 V a ukládá čas do proměnné “timeLeft”.

4 Program robota

Program robota je uložen v paměti EEPROM. Po kliknutí na tlačítko *Zapni* v uživatelském programu, začne vykonávat uložené instrukce. Spuštěný robot načte instrukci na prvním adresovém řádku a adresu zvýší o jedničku, podle načtené instrukce provede akci, po provedení se načte další instrukce a provede. Robot bude pokračovat do té doby, než narazí na poslední příkaz, který ukončí program.

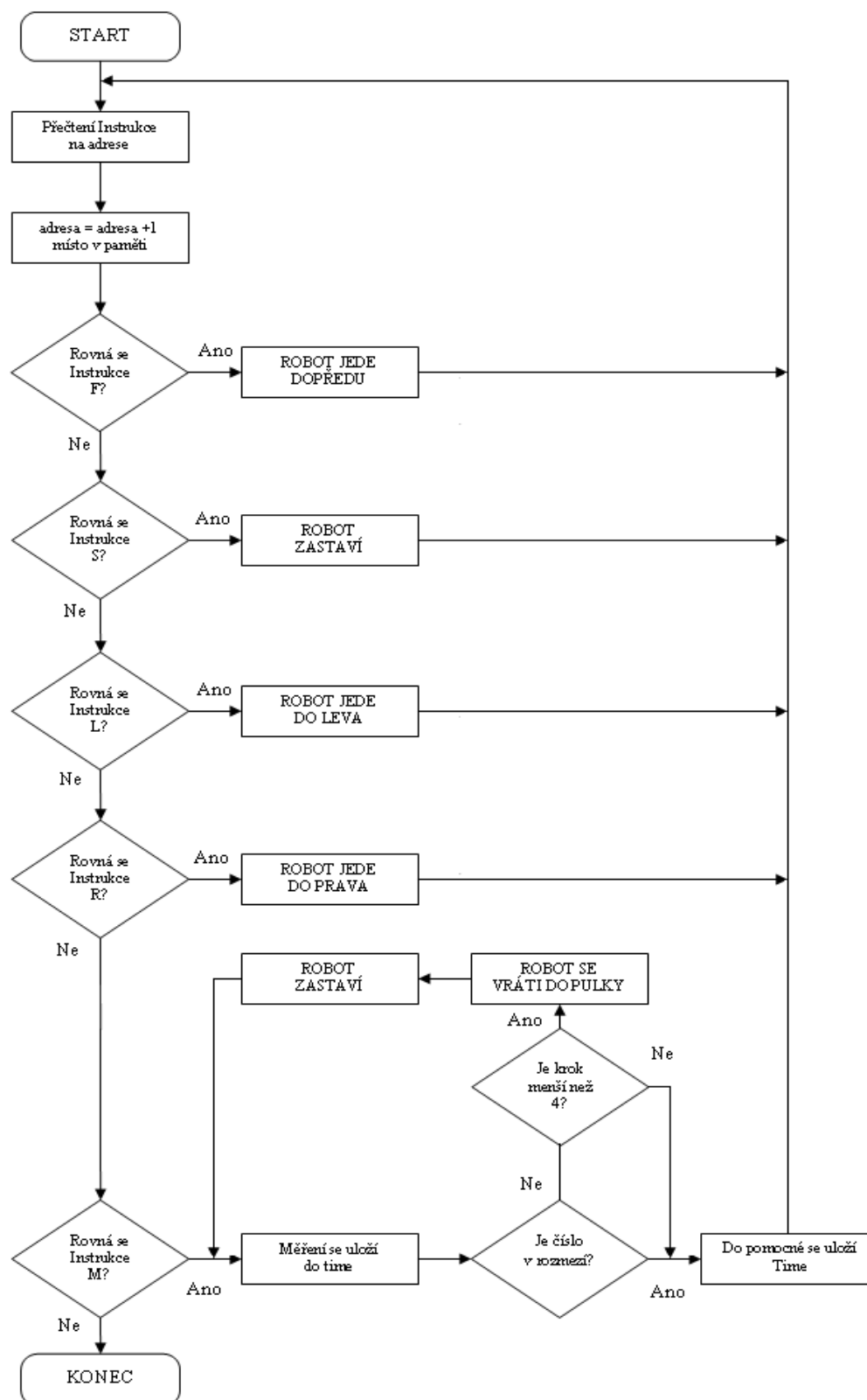
Pro spuštění robota je potřeba mít připojené napájení a přepínač v poloze 2. Aktivovaný robot čeká na zprávu od uživatelského programu, jakmile zachytí na sériovém portu příkaz, provede buď nastavení rozdílu (100, 500, 1000) nebo spuštění souboru instrukcí měření, které jsou uloženy v EEPROM.

Instrukce uložené v EEPROM

- “M” – měření
- “F” – jede vpřed
- “B” – jede zpět
- “L” – zatáčí vlevo o 90°
- “R” – zatáčí vpravo o 90°
- “S” – stojí
- “Q” – zastaví navigaci

Instrukce jsou vykonávány podle vývojového diagramu (Obrázek 21).

4.1 Vývojový diagram



Obrázek 21 – Vývojový diagram robota

4.2 Navigace robota

Instrukce “M” změří intenzitu světla, naměřenou hodnotu uloží do proměnné “pomocná”, odešle do uživatelského programu a následuje další instrukce. Při následném “M” (měření) se hodnota porovná s hodnotou v proměnné “pomocná”, je-li rozdíl mezi hodnotami menší než 100 ms (500 ms, 1000 ms dle nastavení v uživatelském programu), tak se hodnota odešle do uživatelského programu, v opačném případě se robot o polovinu vzdálenosti vrátí zpět a opětovně provede sekvenci měření.

Pokud je opět hodnota mimo limit, robot se vrátí do čtvrtiny standardní vzdálenosti, provede sekvenci měření. Další půlení nelze, protože by byl překročen minimální krok. Návrat na pozici hlavního měření probíhá ve stejných bodech měření, jen v opačném pořadí, poté se může načítat další instrukce.

Robot vykonává instrukce uložené v paměti EEPROM, kde je nastavený pohyb a měření (jedná se o trasu míst, kde bude provedeno měření) Trasa pohybu směřuje do 16 hlavních bodů měření se standardní vzdáleností od sebe. Vždy po 4. hlavním měření následuje zatáčka.

4.2.1 Trasa robota bez proměření

Robot na řádku vykoná 4 hlavní měření. Mezi dvěma hlavními měřeními má robot pohyb vpřed (instrukce “M”, “F”, “S”, “M”, “R”, “S”, “M”).

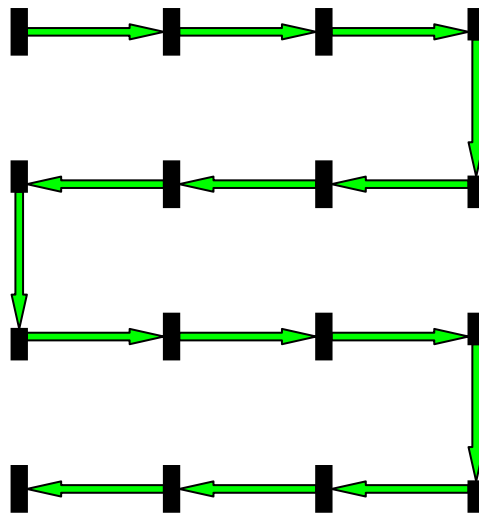
Po 4. hlavním měření následuje otočení o 90° doprava, jízda vpřed a zatáčka o 90° doprava (celkově se otočí o 180°), kde se provede 5. hlavní měření (instrukce “R”, “S”, “F”, “S”, “R”, “S”, “M”).

Po 8. hlavním měření následuje otočení o 90° doleva, jízda vpřed a zatáčka o 90° doleva (celkově se otočí o 180°), kde se provede 9. hlavní měření (instrukce “R”, “S”, “F”, “S”, “R”, “S”, “M”).

Po 12. hlavním měření následuje otočení o 90° doprava, jízda vpřed a zatáčka o 90° doprava (celkově se otočí o 180°), kde se provede 13. hlavní měření (instrukce “R”, “S”, “F”, “S”, “R”, “S”, “M”).

Na 16. hlavním měření robot dospěl do konce trasy navigace. Celý průběh trasy měření je znázorněn na obrázku (Obrázek 22).

Při měření se ukládají hodnoty do dvou proměnných “time” a “pomocná”. Pro porovnání těchto proměnných nám slouží proměnná “rozdil”, pomocí které se zjišťuje, zda jsou naměřené hodnoty v toleranci. Naměřená hodnota z proměnné “pomocná” se vždy odesílá přes sériový port. Při proměření robot couvá.



Obrázek 22 – Trasa robota

4.2.2 Mezi dvěma hlavními měřeními na řádku, bez nutnosti proměření

První instrukce “M” provede 1. hlavní měření, změří intenzitu světla, uloží zjištěnou hodnotu do proměnné “time” a “pomocná”. Proměnnou “pocetMereni” zvětšíme o jedničku (identifikace konce řádku a umožnění zatočení robota).

Druhá instrukce “F” popojede dopředu o nadefinovanou vzdálenost (což je standardní vzdálenost mezi Hlavními měřeními), na místo 2. hlavního měření.

Třetí instrukce “S” zastaví na místě 2. hlavního měření.

Čtvrtá instrukce “M” provede 2. hlavní měření, které se uloží do proměnné “time” a vyvolá podprogram Porovnávání, který porovnává proměnné “pomocná”, kde se nachází první měření a “time” s nově naměřenou hodnotou. V našem případě jsou hodnoty v nadefinovaném rozmezí hodnot (proměnná “rozdil” je rovna jedné z hodnot: 100 ms, 500 ms, 1000 ms). Do “pomocna” se uloží hodnota z proměnné “time”. Proměnnou “pocetMereni” zvětšíme o jedničku (vysvětleno výše).

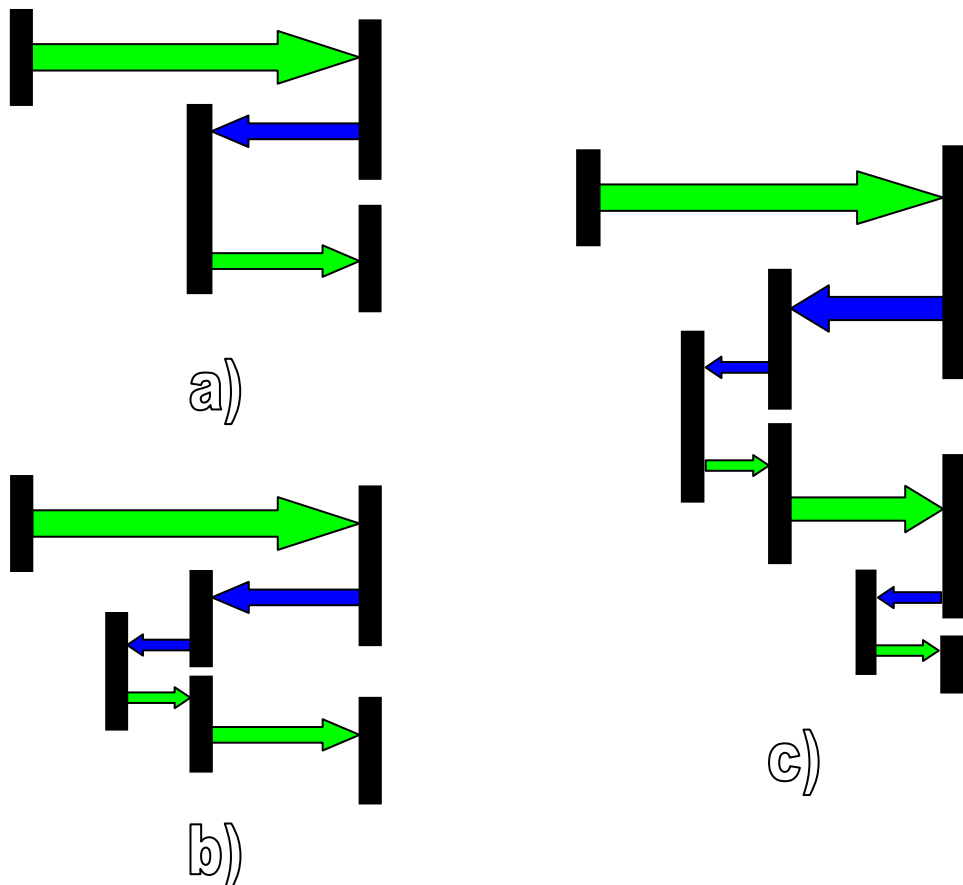
4.2.3 Proměření na řádku mezi dvěma hlavními měřeními

První čtyři instrukce jsou shodné s popisem v předchozím bodě (0) Avšak výsledek porovnání proměnných je mimo stanovený interval hodnot. V tomto případě se robot vrátí na půlku vzdálenosti mezi hlavními měřeními a provede nové proměření (princip měření je shodný s hlavním měřením).

- a) **Jedno proměření** (Obrázek 23a): - proměřená hodnota uložená v proměnné “time“ je v nadefinovaném rozmezí hodnot s proměnnou “pomocna“. Do “pomocna“ se uloží hodnota z proměnné “time”. Robot jede do pozice 2. hlavního měření.
- b) **Dvě proměření** (Obrázek 23b): - proměnné nejsou v rozmezí hodnot, robot provede návrat na $\frac{1}{4}$ ze standardní vzdálenosti mezi jednotlivými hlavními měřeními, kde se opět provede měření. Jelikož je $\frac{1}{4}$ vzdálenosti minimální krok, tak se do proměnné “pomocná” uloží hodnota z proměnné “time”. Robot pokračuje na polovinu standardní vzdálenosti, kde opět proměří a porovná hodnoty. Robot jede do pozice 2. hlavního měření, kde provede

nové proměření (princip měření je shodný s hlavním měřením). Hodnoty jsou v rozmezí.

- c) **Tři proměření** (Obrázek 23c): - robot postupuje stejně jako v bodě b), až na to, že hodnoty nejsou v rozmezí hodnot, tak se musí přeměřit $\frac{3}{4}$ vzdálenosti od předešlého hlavního měření, robot se tedy vrátí o $\frac{1}{4}$ vzdálenosti zpět a proměří, jelikož je $\frac{1}{4}$ vzdálenosti minimální krok, tak v případě že hodnota není opět v rozmezí, tak se robot vrátí na 2. hlavní měření. Poté pokračuje robot podle navigace.



Obrázek 23 – Proměření mezi 2 hlavními měřeními

4.2.4 Pravá zatáčka bez proměřování

Pravá zatáčka robota je vykonána v okamžiku, když se instrukce rovná “R”. Robot zatočí o 90° doprava, následují instrukce “S” zastavení, “F” popojetí na standardní vzdálenost, “S” zastavení, “R” zatočení o 90° doprava a “M”, které je v tomto případě v rozmezí hodnot. Zobrazeno na obrázku (Obrázek 24a).

4.2.5 Pravá zatáčka s proměřováním

Vykoná se předchozí bod (0), až na měření, které není v rozmezí hodnot.

- a) **Jedno proměření** (Obrázek 24b): - robot se zatočí doleva, jede zpět na polovinu standardní vzdálenosti (jede dozadu), zatočí doprava (směrem dopředu) a proměří hodnoty v rozmezí.
- b) **Dvě proměření** (Obrázek 24c): - stejný děj jako v bodě a), jen s rozdílem, že proměnné nejsou v rozmezích, robot se zatočí doleva, jede zpět na $\frac{1}{4}$ standardní vzdálenosti (jede dozadu), zatočí doprava (směrem dopředu) a proměří, jelikož je to minimální krok, tak už se robot nevrací.

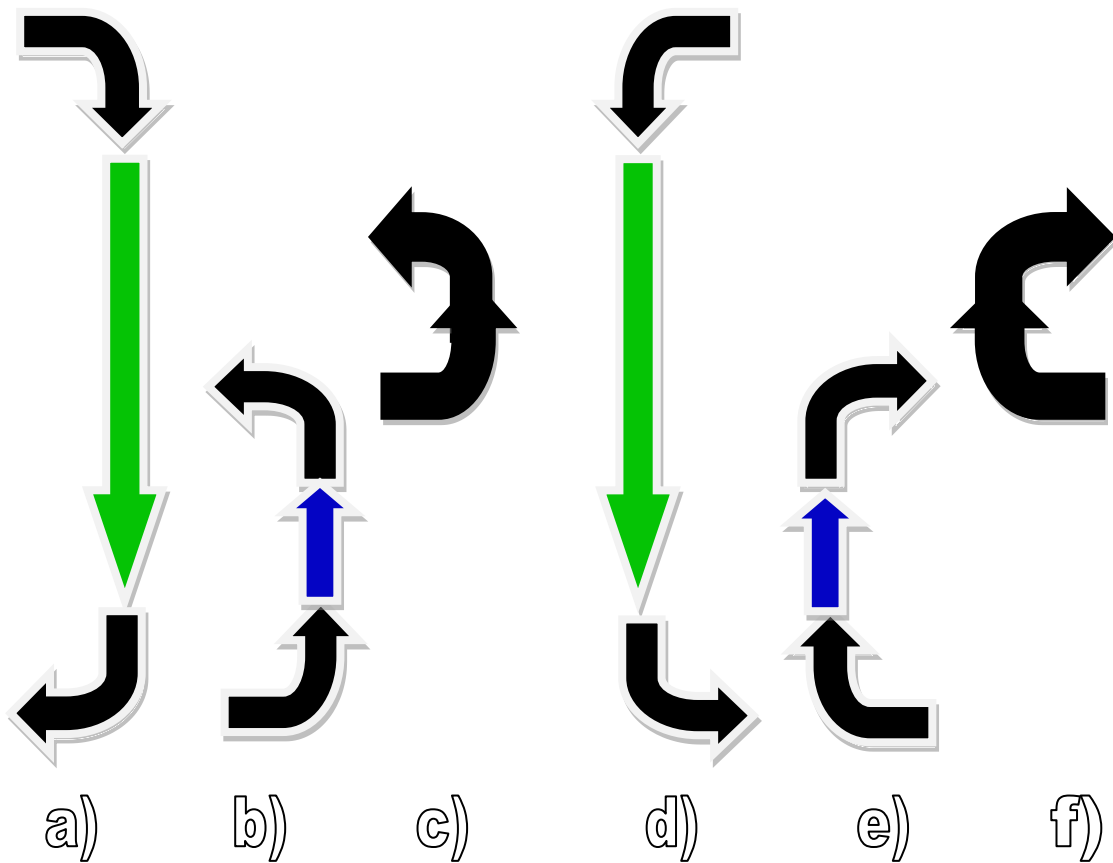
4.2.6 Levá zatáčka bez proměření

Levá zatáčka robota je vykonána v okamžiku, když se instrukce rovná "L". Robot zatočí o 90° doleva, následují instrukce "S" zastavení, "F" popojetí na standardní vzdálenost, "S" zastavení, "L" zatočení o 90° doleva a "M", které je v tomto případě v rozmezí hodnot. Zobrazeno na obrázku (Obrázek 24d).

4.2.7 Levá zatáčka s proměřováním

Vykoná se předchozí bod (4.2.6), až na měření, které není v rozmezí hodnot.

- a) **Jedno proměření** (Obrázek 24e): - robot se zatočí doprava, jede zpět na polovinu standardní vzdálenosti, zatočí doleva (směrem dopředu) a proměří hodnoty v rozmezí.
- b) **Dvě proměření** (Obrázek 24f): - stejný děj jako v bodě a), jen s rozdílem, že proměnné nejsou v rozmezí, robot se zatočí doprava, jede zpět na $\frac{1}{4}$ standardní vzdálenosti, zatočí doleva (směrem dopředu) a proměří, jelikož je to minimální krok, tak už se robot nevrací.



Obrázek 24 – Robot v zatáčce

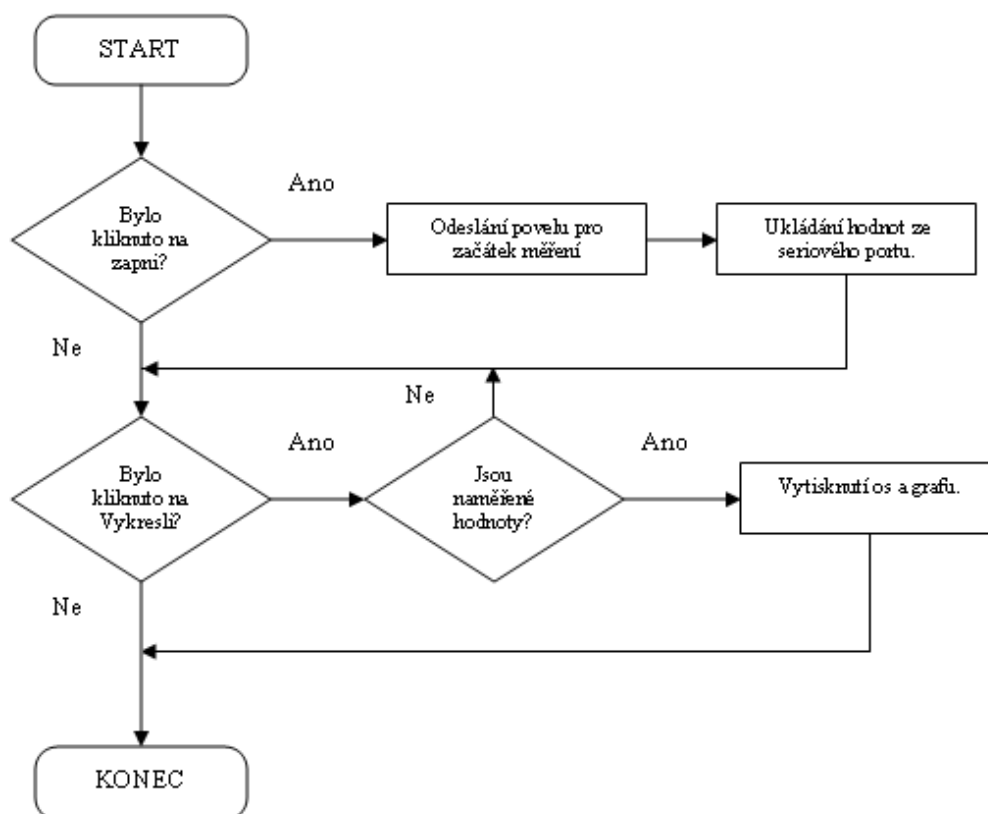
5 Uživatelský program

Uživatelský program je napsán v programovacím jazyce C# s vývojovým diagramem (Obrázek 25).

Po kliknutí na tlačítko *Zapni* se aktivuje sériový port PC a odešle příkaz, který spustí program robota. Program robota obsahuje naplánovanou trasu jízdy a měření po této trase. Naměřené hodnoty robot porovná, věděním zjistí, zda se má vracet kvůli přeměření. Vyhovující hodnoty a zprávy o průběhu měření odesílá po sériovém portu do PC, kde se hodnoty ukládají do pole čísel, zprávy do pole řetězců pro další zpracování.

Spuštěním *Vykresli* se uloží naměřená data do matic, ze kterých se naměřené hodnoty vykreslí. Na grafu se zobrazuje proměřená plocha, která je tvořena kombinací barev červená, modrá a černá.

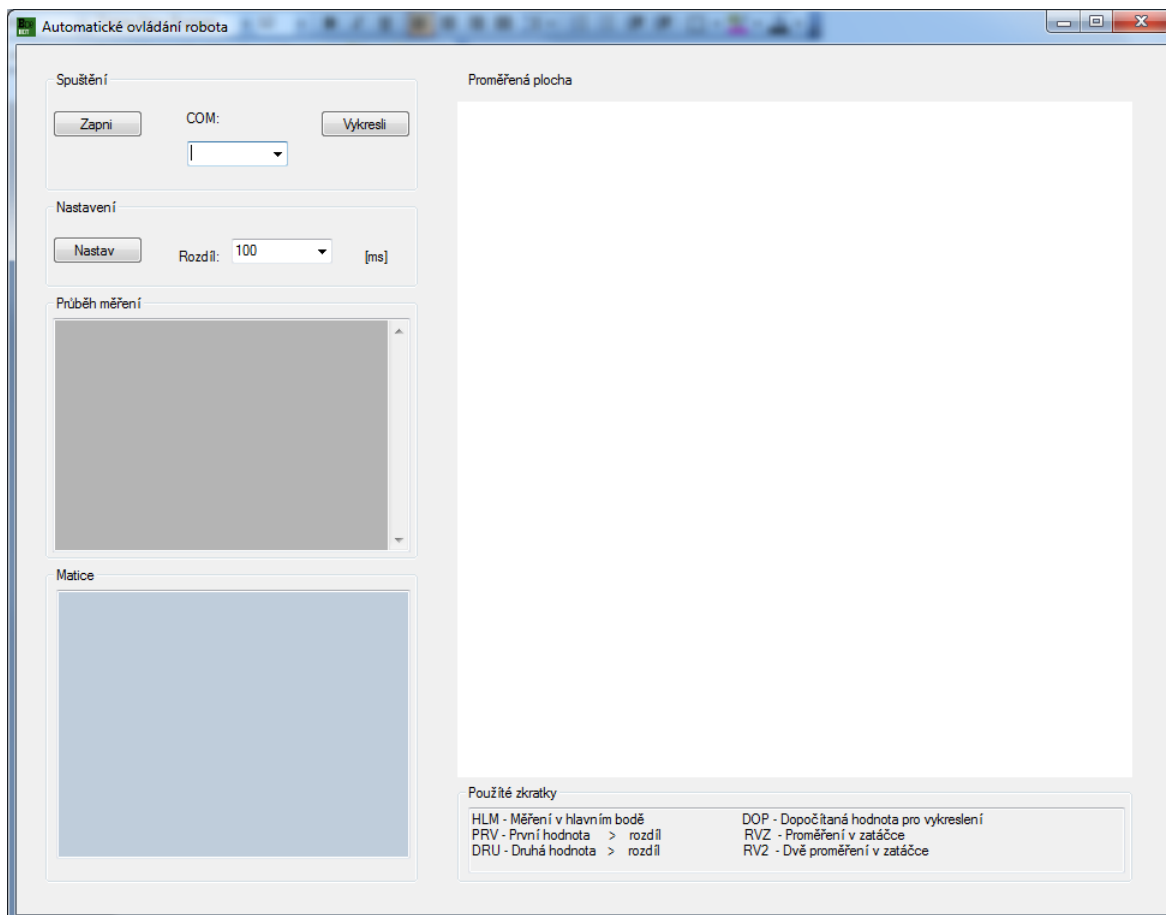
5.1 Vývojový diagram



Obrázek 25 – Vývojový diagram C#

5.2 Ovládání uživatelského programu

Uživatelský program obsahuje ovládací tlačítka, TextBoxy pro vytisknutí naměřených hodnot a vysvětlivky k použitým zkratkám. Dále bitmapový obrázek, pro vykreslení naměřené plochy (Obrázek 26).



Obrázek 26 – Uživatelský program

Po spuštění uživatelského programu se načte do “ComboBoxCOM“ seznam dostupných sériových portů PC. V případě, že uživatel nemá připojeného robota na výchozím sériovém portu COM16, tak zvolí správný port z načteného seznamu, na kterém se robot nachází. Rozdíl je automaticky nastaven na hodnotu 100 ms, což je čas vybití kondenzátoru, o který se může lišit hodnota předchozí a aktuální. Pokud není robot připojen, tak při kliknutí na tlačítka *Zapni* a *Nastav*, bude do textBoxu Průběh měření vypsána chybná hláška: “Port COM16 neexistuje. Připojte zařízení na COM16 nebo vyberte jiný sériový port!“.

Nastavením rozdílu se určí přesnost měření a vykreslení grafu. Rozdíl hodnot lze nastavit v “ComboBoxRozdil“ na 100, 500 a 1000 ms. Kliknutím na tlačítko *Nastav* je potvrzené nastavení.

Po kliknutí na tlačítko *Zapni* se aktivuje sériový port PC a odešle příkaz pro spuštění programu robota, který má za úkol měření intenzity světla na určené ploše.

Robot odesílá hodnoty přes sériový port a data se uloží uživatelským programem do pole. V textBoxu se zobrazí “Průběh měření“. V programu robota je naplánovaná trasa a místa měření.

Tlačítko *Vykresli* zpracovává naměřená data, ze kterých se vykreslí na bitmapu plocha, zobrazující intenzitu světla v měřeném prostoru. Tento jev je interpretován barevným spektrem od červené po černou (Obrázek 27).

Intenzita světelného záření

- Červená barva je vysoká intenzita světla (měření pod zdrojem světla)
- Modrá je nižší intenzita světla (měření v šeru)
- Černá je nejnižší intenzita světla (měření ve stínu)



Obrázek 27 – Spektrum barev zobrazitelné v grafu

Do textBoxu “Průběh měření“ se vypisují aktuálně změřené hodnoty a akce, které robot provedl, případně chybné hlášky programu.

Data jsou vytisknuta v textBoxu “Matice“, ve formě dvou matic. První matice značí provedené úkony, druhá matice zobrazuje seznam naměřených hodnot.

Graf se zobrazí na bitmapovém obrázku vloženým do “Proměřena plocha“, který odpovídá světelným podmínkám měřeného prostoru. Přesnost grafu závisí na nastavení rozdílu před začátkem měření. Výška jednotlivých řádků měření může být jiná.

V textBoxu “textBoxZkratky“ je vyjádřen význam jednotlivých zkratek použitých ve vedlejších textBoxech “Průběh měření“ a “Matice“. Podle těchto zkratek je možno zjistit, jak probíhalo měření a kde byla dopočítaná hodnota.

5.3 Uložení dat

Proměnné

“Akce“ – jednorozměrné pole pro ukládání akcí,

“Pole“ – jednorozměrné pole pro ukládání naměřených hodnot,

“PocetSloupcu“ – jednorozměrné pole pro uložení počtu sloupců v jednotlivých řádcích,

“Dej“ – matice akcí,

“Matice“ – matice naměřených hodnot.





Po kliknutí na tlačítko *Zapni* se aktivuje sériový port PC a odešle příkaz pro spuštění programu robota, který má za úkol měření intenzity světla na určené ploše. Robot data odesílá přes sériový port. Data se postupně ukládají do polí:

- Akce do "Akce",
- Naměřené hodnoty do "Pole".





Zjištěním počtu řádků a sloupců, které robot proměřil, lze data uložit do matic.

Každý řádek, který robot proměřil, je tvořen 4 hlavními měřeními a poté následuje odřádkování. Sečtením všech měření na řádku se získá počet sloupců, tento způsob platí pro všechny řádky. Počet jednotlivých sloupců se ukládá do pole "PocetSloupcu", potřebný k uložení akcí a hodnot do matic.

Z jednorozměrných polí "Akce" a "Pole" se uloží data do matic "Dej" a "Matice". Pole "PocetSloupcu" slouží pro uložení akcí a hodnot do matic ve správném směru². V případě stejně dlouhých řádků se naplní matice podle znázornění (Obrázek 28), zbytek matice "Dej" je nenaplňen a má hodnotu null³. Různě dlouhé řádky jsou znázorněny na obrázku (Obrázek 29).

				null	null	null
				null	null	null
				null	null	null
				null	null	null

Obrázek 28 – Matice "Dej", stejně dlouhé řádky

				null	null	null
					null	null
						null
						

Obrázek 29 – Matice "Dej", různě dlouhé řádky

² Data se ukládají do matice z levé strany, platí pro 1. a 3. řádek. Z pravé strany, platí pro 2. a 4. řádek

³ Řádky matic "Dej" a "Pole" mají 13 sloupců

Pro zjištění nejdelšího řádku slouží ArrayList “nejRad“, do kterého se přidají akce 1. řádku. Pokud je proměření v některém z následujících řádků a v “nejRad“ není, následuje vložení do “nejRad“. Nejdelší řádek je potřeba pro uložení akcí do matice “AkceM“, kde budou všechny řádky stejně dlouhé.

Jestliže “nejRad“ i aktuální řádek obsahuje akci proměření, tak se uloží do matice “AkceM“ akce aktuálního řádku. Je-li na nejdelším řádku proměření a na aktuálním není, do “AkceM“ se uloží DOP (Značí dopočítanou hodnotu, která je potřeba k vykreslení grafu). Všechny sloupce matice “AkceM“ budou naplněny akcemi, přitom budou mít řádky stejnou délku.

Možné uložení akcí v matici “AkceM“

- Všechny řádky jsou stejné (Tabulka 1)
- Řádky jsou různě dlouhé a doplněné akci DOP (Tabulka 2)

Tabulka 1 – Stejně řádky

Měření	1. bod	2. bod	3. bod	4. bod
1. řádek	HLM	HLM	HLM	HLM
2. řádek	HLM	HLM	HLM	HLM
3. řádek	HLM	HLM	HLM	HLM
4. řádek	HLM	HLM	HLM	HLM

Tabulka 2 – Řádky jsou různě dlouhé

Měření	1. bod	2. bod	3. bod	4. bod	5. bod	6. bod	7. bod
1. řádek	HLM	HLM	HLM	DOP	DOP	DOP	HLM
2. řádek	HLM	HLM	HLM	DOP	DRU	DOP	HLM
3. řádek	HLM	HLM	HLM	PRV	PRV	DOP	HLM
4. řádek	HLM	HLM	HLM	DRU	DRU	DRU	HLM

Matice “Maticka“ se naplní naměřenými hodnotami podle matice “AkceM“, kde jsou uloženy akce. Při akci DOP je nutné dopočítat průměrnou hodnotu do matice “Maticka“.

$$DOP = \frac{(Větší - Menší)}{2} + Menší$$

kde *DOP* – dopočítaná hodnota mezi naměřenými hodnotami,
Větší – je to větší naměřená hodnota vedle dopočítané hodnoty,
Menší – je to menší naměřená hodnota vedle dopočítané hodnoty.

5.4 Vykreslení plochy

Plocha tvoří matici buněk, jejichž počet závisí na počtu řádků a sloupců. Plocha se vykresluje po sloupcích matice. Sloupce se vykreslují po buňkách. Buňku tvoří dvě hodnoty z 1. a 2. řádku, přitom jsou hodnoty řádků pod sebou. Buňka se vykresluje po řádcích, které jsou tvořeny první a druhou hodnotou. První hodnota je začátek řádku a druhá je konec řádku. Výška řádku se mění, je-li proměření v zatáčce.

5.4.1 Nastavení barvy RGB

Barva se skládá ze tří základních barev červená–zelená–modrá. RGB hodnoty jsou kódované ve 24 bitové hloubce. Každá základní barva má 8-bitovou hloubku, jejíž intenzitu určuje rozsah (0 – 255) (Wikipedia).

Intenzita světla v grafu je znázorněná barevným přechodem červená–modrá–černá. V tomto barevném přechodu není potřeba zelené barvy, mění se jen červená a modrá barva v RGB. Počáteční barva se nastaví pro 1. pixel řádku buňky.

Barevné přechody mají krajní hodnoty:

- Červená barva (255, 0, 0),
- Purpurová barva (255, 0, 255),
- Modrá barva (0, 0, 255),
- Černá barva (0, 0, 0).

Naměřená hodnota je menší než 255, nastavení barevné hloubky (255, 0, hodnota), výsledná barva bude v rozsahu obrázku (Obrázek 30).

Naměřená hodnota je menší než 510 a větší než 254, barevná hloubka má nastavení (510 - hodnota, 0, 255), výsledná barva bude v rozsahu obrázku (Obrázek 31).

Naměřená hodnota je větší než 509, barevná hloubka je (0, 0, 765 - hodnota), výsledná barva bude v rozsahu obrázku (Obrázek 32).

První hodnota řádku je menší než 255, poslední hodnota řádku je větší než 510. Musí dojít k překročení hodnot 255 a 510. První hodnota je menší než 255, barva vypadá následovně (255, 0, hodnota). Barva se postupně mění zvýšením Modré složky RGB na hodnotu 255, čímž je tvořen přechod na purpurovou barvu (255, 0, 255). S dalším zvyšováním hodnoty klesá Červená složka RGB, jakmile se dostane na další hranici tvořenou modrou barvou (0, 0, 255), začne klesat i Modrá složka RGB k poslední hodnotě řádku.

První hodnota je větší jak 510, poslední hodnota řádku je menší než 255. Musí dojít k překročení hodnot 510 a 255. Barvy následujících pixelů se postupně mění zvyšováním Modré složky RGB, dosažením hranice tvořené modrou barvou (0, 0, 255), začne růst

Červená složka RGB a tímto se dosáhne hranice purpurové barvy (255, 0, 255), Modrá složka RGB začne klesat na hodnotu posledního řádku.



Obrázek 30 – Hodnota < 255



Obrázek 31 – Hodnota < 510, hodnota > 255



Obrázek 32 – Hodnota > 509

5.4.2 Řádek buňky

Pro vykreslení 1. řádku je nutné znát dvě změřené hodnoty, mezi kterými je řádek buňky. Hodnoty by měli být v rozsahu 0 – 765, což je barevné spektrum zobrazené v grafu. Barevný přechod mezi barvami červená–modrá–černá⁴ (Obrázek 27). V případě, že hodnota je vyšší než 765, nachází se robot ve tmě a na grafu se zobrazí hodnota 765, znázorňující nejnižší intenzitu světla.

Hodnoty mají různou velikost: - barva se vykresluje po jednom pixelu. 1. Pixelu se nastaví počáteční barva, následujícím pixelům se vypočte barva sečtením nebo odečtením barvy předchozího pixelu a Složky. Pro Složku platí následující rovnice (Parallax, 2010):

$$\text{Složka} = \frac{k}{\text{vzdálenost}} \times \text{barevnáŠířka}$$

kde *Složka* – je hodnota, o kterou se bude měnit barva pixelů,
k – index pixelu v buňce,
vzdálenost – vzdálenost bodů měření,
barevnáŠířka – rozdíl naměřených hodnot.

Hodnoty mají stejnou velikost: - barva bude platit pro celý řádek, který se vykreslí najednou.

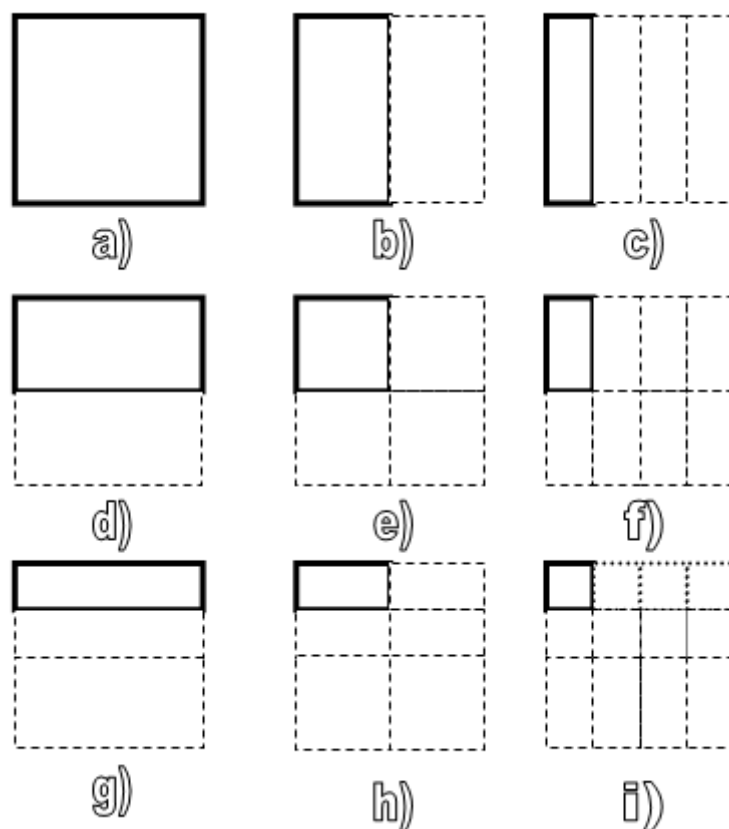
⁴ Červená – vysoká intenzita světla, Černá – velmi nízká intenzita světla

Pro vyplnění buňky se musí zjistit hodnoty krajních sloupců, které se počítají podle předchozího vzorce. Tím se zjistí první a poslední hodnoty řádků, sloužící stejně jako v 1. řádku.

5.4.3 Buňky

Buňku tvoří 4 body měření, 2 body na každém řádku. Možné tvary buněk jsou znázorněny na obrázku (Obrázek 33), kde každá buňka má alespoň jedno hlavní měření, což je 1. hodnota na 1. řádku.

- a) Mezi všemi body je standardní vzdálenost.
- b) Jedno proměření na řádku, od hlavního měření k proměření je poloviční vzdálenost.
- c) Dvě nebo tři proměření na řádku, od hlavního měření k 1. proměření je čtvrtěční vzdálenost.
- d) Mezi body na řádku je standardní vzdálenost. Sloupec byl jednou proměřen, buňka má poloviční výšku.
- e) Jedno proměření na řádku, od hlavního měření k proměření je poloviční vzdálenost. Sloupec byl jednou proměřen, buňka má poloviční výšku.
- f) Dvě nebo tři proměření na řádku, od hlavního měření k 1. proměření je čtvrtěční vzdálenost. Sloupec byl jednou proměřen, buňka má poloviční výšku.
- g) Mezi body na řádku je standardní vzdálenost. Sloupec byl dvakrát proměřen, buňka má čtvrtěční výšku.
- h) Jedno proměření na řádku, od hlavního měření k proměření je poloviční vzdálenost. Sloupec byl dvakrát proměřen, buňka má čtvrtěční výšku.
- i) Dvě nebo tři proměření na řádku, od hlavního měření k 1. proměření je čtvrtěční vzdálenost. Sloupec byl dvakrát proměřen, buňka má čtvrtěční výšku.



Obrázek 33 – Možné tvary buňky

5.4.4 Možnosti vykreslení plochy

U základního tvaru plochy jsou změřené jen body hlavního měření. Proměřené řádky a sloupce mají 4 hodnoty, na šířku i výšku mají 3 buňky, plocha má 9 buněk. Buňky se vykreslují podle obrázku (Obrázek 34).

1	2	3
4	5	6
7	8	9

Obrázek 34 – Buňky bez proměření

Na základním tvaru plochy robot proměřoval hodnoty na řádcích. Proměření mezi hlavními hodnotami bude zobrazeno jako na obrázku (Obrázek 35).

- 1 proměření, 1. a 2. buňka
- 2 proměření, 3. 4. a 5. buňka
- 3 proměření, 6. – 9. Buňka

1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27

Obrázek 35 – Buňky, proměřené řádky

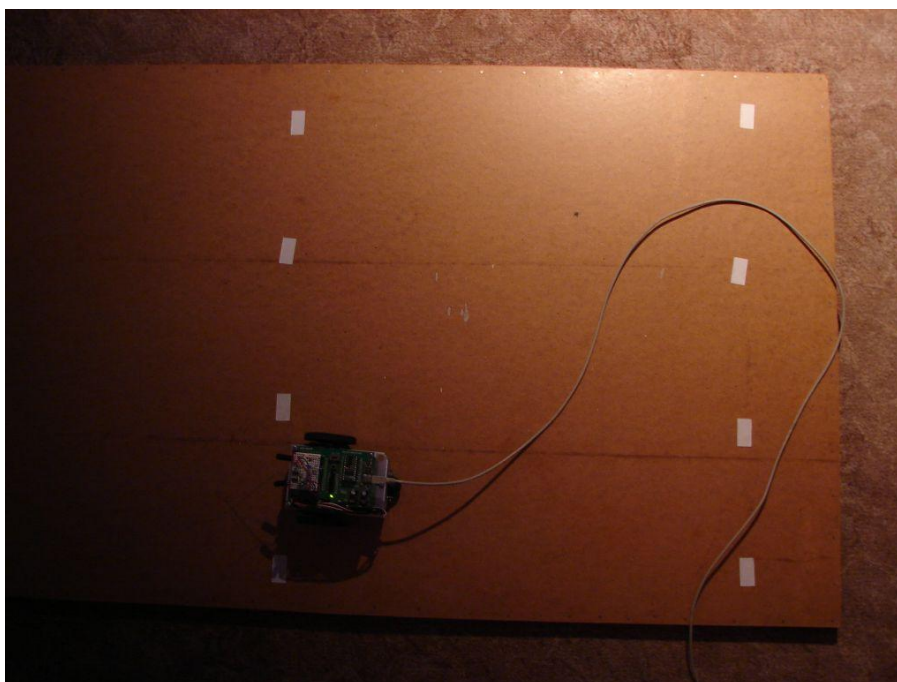
Plocha nebude mít základní tvar (Obrázek 36), proměření sloupců změní výšku grafu:

- žádné proměření, 1. – 3. buňka
- 1 proměření, výška sloupce poloviční, 4. – 6. buňka
- 2 proměření, výška sloupce čtvrtinová, 7. – 9. buňka.

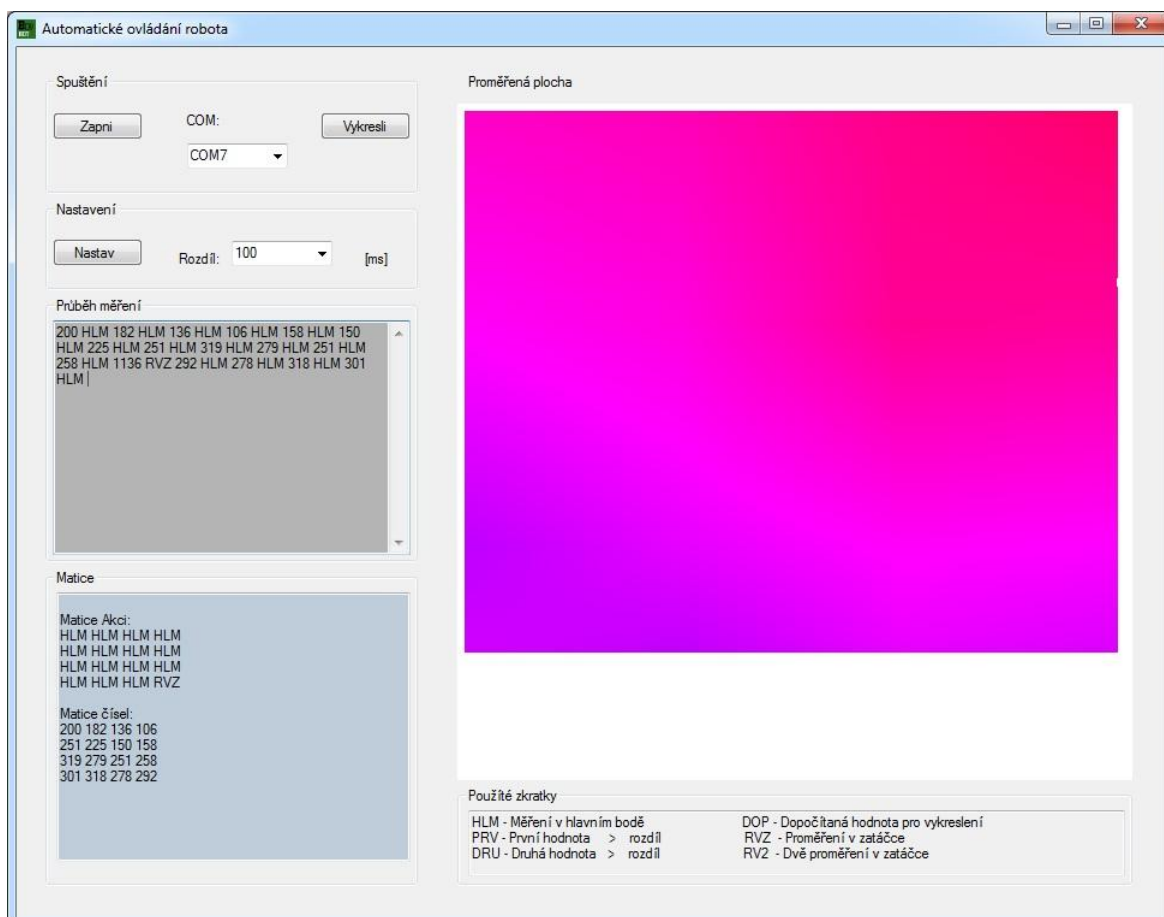
1	2	3
4	5	6
7	8	9

Obrázek 36 – Buňky, proměřené sloupce

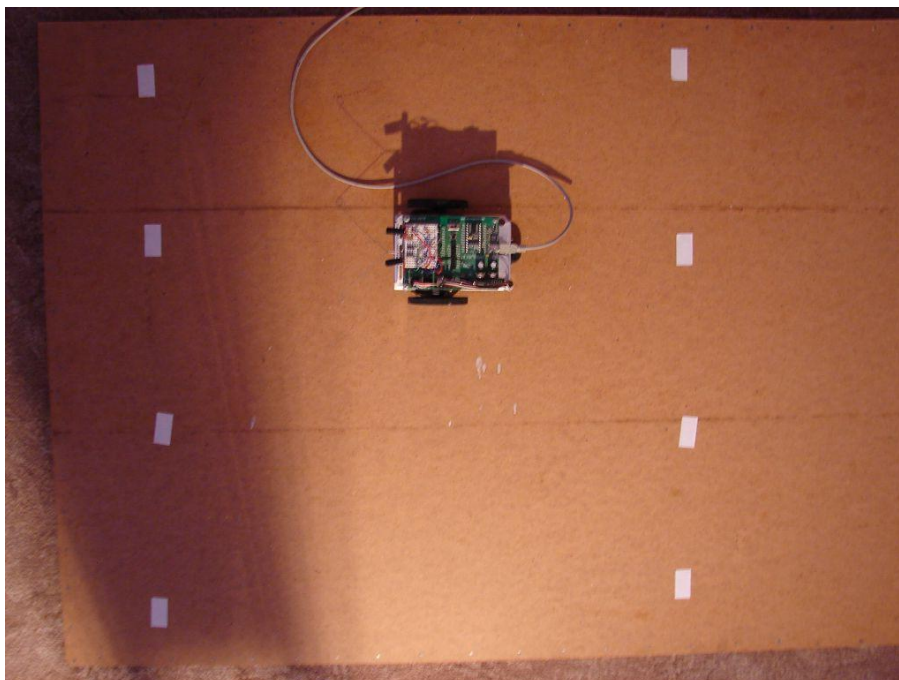
6 Naměřené hodnoty a grafy



Obrázek 37 – 1. proměřená plocha



Obrázek 38 – Graf 1. proměřené plochy



Obrázek 39 – 2. proměřená plocha

Automatické ovládání robota

Spuštění

Zapni COM: COM7 Vykresli

Nastavení

Nastav Rozdíl: 100 [ms]

Průběh měření

```

362 HLM 201 PRV > 100. Robot se o pul kroku vrati.
315 210 PRV > 100. Robot se o pul kroku vrati.
261 203 HLM 156 HLM 178 HLM 146 HLM 136 HLM 251
DRU > 100. Robot se o pul kroku vrati.
143 229 HLM 382 DRU > 100. Robot se o pul kroku vrati.
343 DRU > 100. Robot se o pul kroku vrati.
317 342 367 HLM 396 HLM 127 PRV > 100. Robot se o
pul kroku vrati.
197 PRV > 100. Robot se o pul kroku vrati.
295 PREKROCENI MINIMALNIHO KROKU.
206 113 HLM 115 HLM 128 HLM 102 HLM 105 HLM 167
HLM 314 DRU > 100. Robot se o pul kroku vrati.
293 DRU > 100. Robot se o pul kroku vrati.
275 PREKROCENI MINIMALNIHO KROKU.

```

Matice

Matice Akci:

```

HLM PRV PRV HLM DOP HLM HLM
HLM DRU DRU HLM DRU HLM HLM
HLM PRV PRV HLM DOP HLM HLM
HLM DRU DRU HLM DOP HLM HLM

```

Matice čísel:

```

362 315 261 203 179 156 178
367 342 317 229 143 136 146
396 295 206 113 114 115 128
309 291 275 167 136 105 102

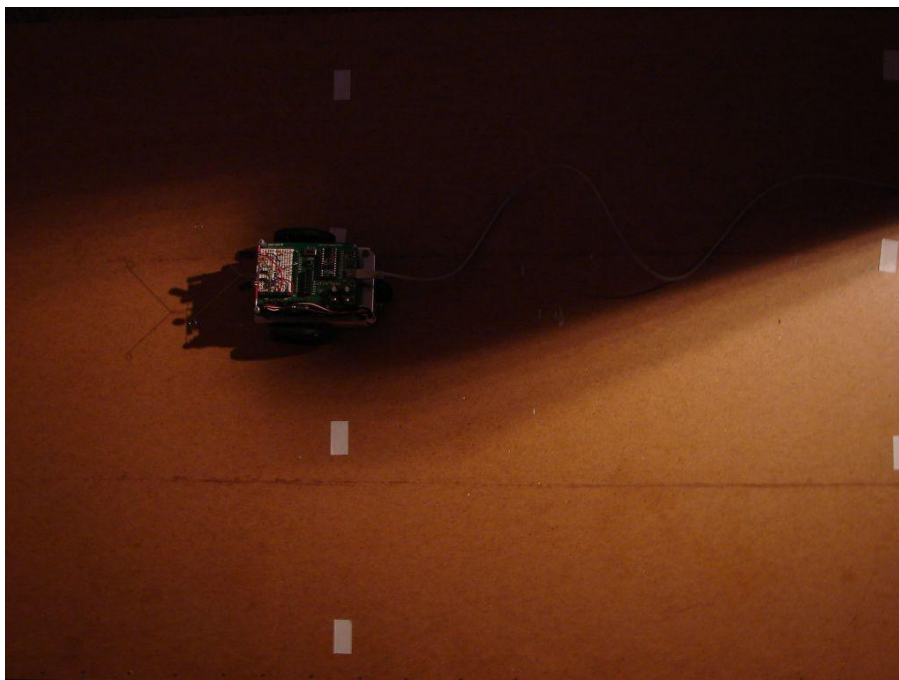
```

Proměřená plocha

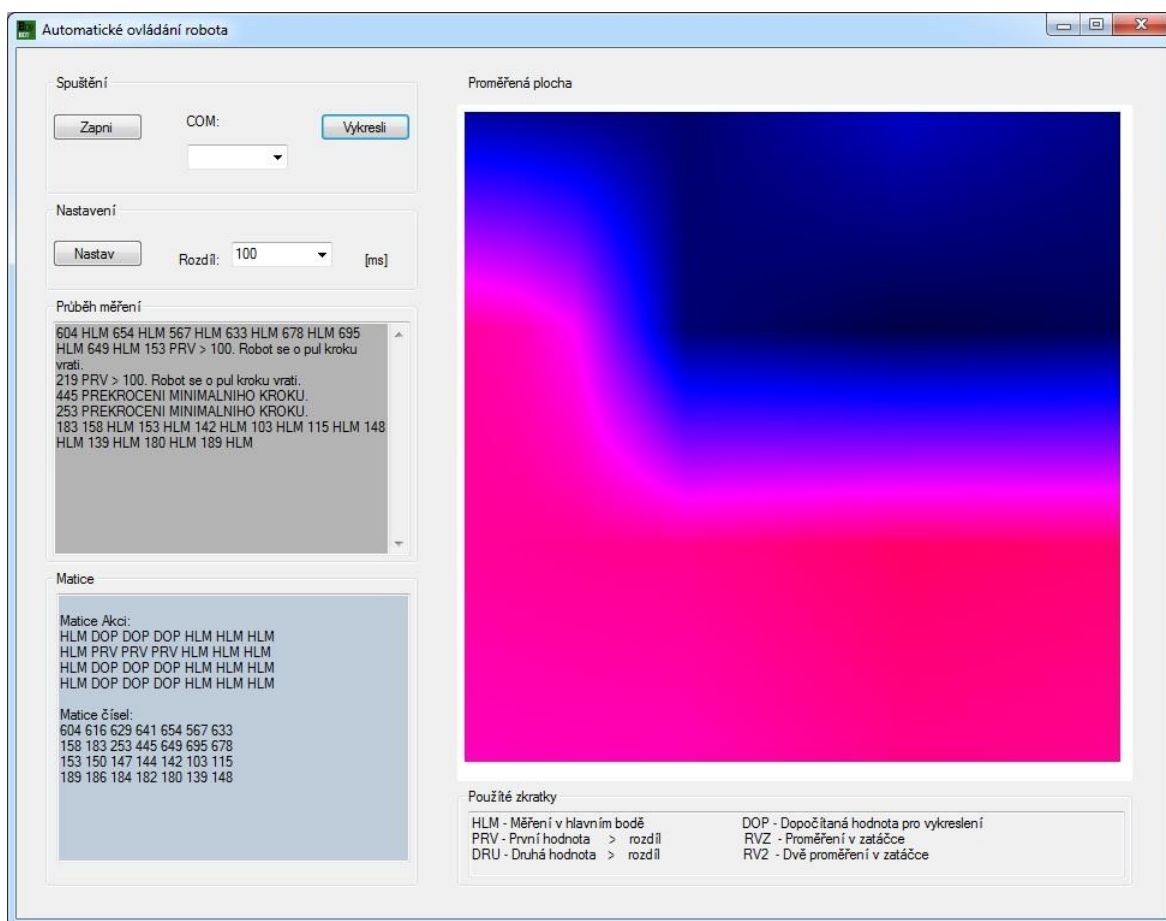
Použité zkratky

HLM - Měření v hlavní bodě	DOP - Dupočítaná hodnota pro vykreslení
PRV - První hodnota > rozdíl	RVZ - Proměření v zatáčce
DRU - Druhá hodnota > rozdíl	RV2 - Dvě proměření v zatáčce

Obrázek 40 – Graf 2. proměřená plocha



Obrázek 41 – 3. proměřená plocha



Obrázek 42 – Graf 3. proměřená plocha

7 Závěr

Pro měření světelného pole za pomoci autonomního robota Boe Bot byl napsán kód v editoru BASIC Stamp, který byl nahrán do paměti robota EEPROM po sériové lince. Kód obsahuje navigaci s hlavními body měření, je-li mezi změřenými hodnotami velký rozdíl, potom následuje proměření mezi body.

Měření světelného pole je realizováno paralelní kombinací fotorezistoru a kondenzátoru, který plní funkci paměti. Měří se čas, za jak dlouho se vybije kondenzátor v ms.

Zpracování naměřených hodnot a vykreslení grafu je naprogramováno v jazyku C#. Před samotným měřením je nutné propojit robota a počítač přes sériový port. Přepínač robota dát do pohotovostního režimu. V uživatelském programu se nastaví port COM, na kterém je robot připojen, v případě že byl vybrán jiný port, tak po kliknutí na *Zapni* se do textového pole Průběh měření vypíše hláška „Port COM16 neexistuje. Připojte zařízení na COM16 nebo vyberte jiný sériový port!“. Po spuštění měření se začnou ukládat data. Jakmile je pole proměřené, tak se graf vykreslí kliknutím na *Vykresli*.

Pohyb robota ovlivňuje povrch, kabel a baterie. Odpor povrchu je na hladkém povrchu menší např. plovoucí podlaha, robot se pohybuje snadno, jakmile se dostane např. na koberec, tak vznikne větší odpor povrchu a nedotáčivost robota, tím dojde k nechtěné změně trasy. Právě proto je potřeba použít individuální nastavení pro každý povrch. Jakmile robot za sebou tahá kabel, tak dojde k nechtěné změně trasy vlivem toho, že serva nemají takovou sílu, aby se robot pohnul tam, kam je potřeba. Slabé baterie způsobí pomalejší chod serv, což se projeví na postupném zkracování vzdálenosti mezi body a nedotáčivostí v zatáčkách.

Literatura

Frohn, Manfred a další. 2006. *ELEKTRONIKA polovodičové součástky a základní.* [překl.] Jiří Nobilis a Vlastislav Kazda. Praha : BEN, 2006. ISBN 80-7300-123-3.

Microsoft. MSDN. [Online] [Citace: 5. 5 2011.] <http://msdn.microsoft.com/cs-cz/vcsharp/default.aspx>.

Parallax. 2005. BASIC Stamp Syntax and Reference Manual. [Online] 2005. [Citace: 5. 5. 2011.] <http://www.parallax.com/Portals/0/Downloads/docs/prod/stamps/web-BSM-v2.2.pdf>. ISBN 1-928982-32-8.

— . **2010.** Robotics with the Boe-Bot. [Online] 2010. [Citace: 5. 5. 2011.] http://www.parallax.com/Portals/0/Downloads/docs/books/edu/Roboticsv3_0.pdf. ISBN 9781928982531.

— . **2011.** Robots. [Online] 2011. [Citace: 5. 5. 2011.] <http://www.parallax.com/Store/Robots/tabid/82/Default.aspx>.

Wikipedia. *RGB*. [Online] [Citace: 5. 5. 2011.] <http://cs.wikipedia.org/wiki/RGB>.

Příloha A – Obsah CD

Příložené CD obsahuje:

- Tato práce,
- program robota ve složce Robot,
- uživatelský program ve složce Zpracování_Dat.