



Středoškolská technika 2023

Setkání a prezentace prací středoškolských studentů na ČVUT

NÁVOD KE STAVBĚ A PROGRAMOVÁNÍ BALANCUJÍCÍHO ROBOTA KROK ZA KROKEM

Adam Hejtmánek

VOŠ, SŠ, COP Sezimovo Ústí
Budějovická 421, Sezimovo Ústí

Anotace

Tato práce je plnohodnotný návod se zásadními informacemi potřebnými ke stavbě balancujícího robota. Je zde probráno inverzní kyvadlo, regulace PID, a samotná stavba robota krok za krokem. Kromě balancování je zde popsáno i použití ultrazvukového snímače a možnosti rozšíření o další funkce.

Klíčová slova

Inverzní kyvadlo, Regulace, balancující robot, ladění, návod

Obsah

1	Úvod.....	3
2	Požadované parametry balancujícího robota a použité nástroje	3
3	Průzkum existujících návodů a jejich využitelnosti	3
4	Proč balancující robot?.....	4
5	Stručný popis PID regulace.....	5
5.1	Jak naprogramovat P složku PID regulátoru	6
5.2	Jak naprogramovat I složku PID regulátoru	6
5.3	Jak naprogramovat D složku PID regulátoru	7
6	Trochu teorie k inverznímu kyvadlu	7
7	Balancující robot krok za krokem	8
7.1	Jak bude robot vypadat	8
7.2	Krok 1. – použité prvky	8
7.3	Výběr vhodných kol	8
7.4	Popis servomotorů	9
7.5	Popis snímače úhlu naklonění robota	9
7.6	Popis řídicí jednotky	9
7.7	Krok 2. – stavba robota.....	9
7.8	Rozchod kol.....	9
7.9	Výška robota.....	10
7.10	Uchycení gyroskopu	10
7.11	Design robota	10
7.12	Krok 3. – naprogramování robota	10
7.13	Krok 4. – ladění regulátoru	11
8	Rozšíření schopností robota	12
8.1	Jak funguje ultrazvukový snímač	12
8.2	Jak zajistit pohyb robota	12
8.3	Dálkové ovládání	13
9	Závěr.....	14
10	Použitá literatura	15
11	Seznam obrázků	15

1 Úvod

Dnes se roboti stále častěji zapojují do naší společnosti, od industriálních megastrojů až po nanoboty. Málo lidí však ví, jak roboti pracují, vznikají apod. Domnívají se, že jsou příliš složité a toto přesvědčení jim brání se o odvětví robotiky blíže zajímat. Proto vznikla tato práce, kde je popsán návod ke stavbě a naprogramování robota krok za krokem. Cílem je přesvědčit čtenáře, že postavit a naprogramovat robota není nic složitého.

Stavba a programování robota je zde ukázána na příkladu robota balancujícího na dvou kolech. I když naprogramování takového robota není úplně jednoduché, pokud budete postupovat krok za krokem podle tohoto návodu, postupně pochopíte jednotlivé principy fungování tohoto robota a vaší vlastní konstrukci dotáhnete do úspěšného konce.

Tuto práci jsem si vybral, protože věřím, že roboti jsou naše budoucnost a chtěl bych ve více lidech vzbudit zájem o tuto oblast, která roste stále rychlejším tempem. Také z důvodu, absolutního nedostatku plně obsažných návodů pro veřejnost s relevancí na stavbu reálného robota.

2 Požadované parametry balancujícího robota a použité nástroje

Cílem této práce je zhotovit robota, který dokáže balancovat na místě na dvou kolech. Současně bude schopen popojíždět ve vzpřímeném stavu vpřed a vzad v závislosti na vzdálenosti ruky nebo překážky od ultrazvukového snímače. Jako bonus přidáno dálkové ovládání. Aby stavba byla přístupná široké veřejnosti, bude předvedena na stavebnici LEGO MINDSTORMS education EV3.

Postup sestavení robota je v této práci popsán krok za krokem včetně potřebné teorie. K naprogramování robota je využito grafické vývojové prostředí LEGO MINDSTORMS EV3. Toto vývojové prostředí bylo vybráno pro názornost a snadné pochopení. Lze však určitě programovat i v textových jazycích, jako např. C, C+ nebo Python.

3 Průzkum existujících návodů a jejich využitelnosti

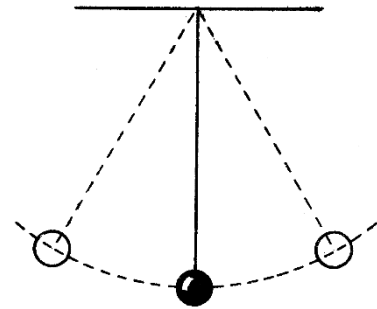
Když si vyhledáte na Googlu “Balancing robot tutorial”, tak se člověk setká s těmito hlavními nedostatky.

- Buď jde o návody, které popisují stavbu sice krok za krokem, ale jde o hotovou konstrukci bez vysvětlení principů balancování a potřebné teorie. Poté, co si postavíte robota a nahrajete do něj od autora hotový program, tak je konec. Máme sice robota, ale vůbec nevíme, jak to funguje.
- Nebo se sice dozvíte, co to je regulace a všechny další technické pojmy, ale je to jen teoreticky, bez reálné ukázky. Dále jsou tyto návody, velice zdouhavé, plné redundantních informací a detailů. Někdy jste úplně zahlceni matematikou, kterou při stavbě ani nepoužijete, pokud nebudete, vytvářet simulaci.

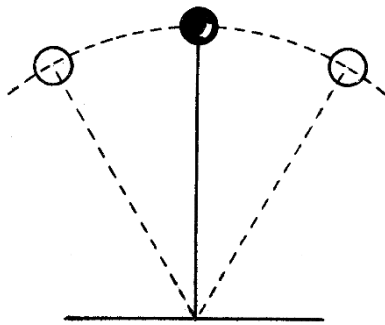
4 Proč balancující robot?

I přes to, že by člověk nejráději hned začal stavět, bez potřebných poznatků teorie by to pravděpodobně nedopadlo dobře. Nejspíše by to dopadlo tak, že bychom bezhlavě měnily regulační konstanty a modlili se, „ať už to konečně funguje!“

Všichni znají běžné kyvadlo. Je to zavěšené těleso, které se může otáčet kolem bodu zavěšení. Má těžiště pod osou úchyty. Při vychýlení z klidového stavu se samo vrací zpět do klidové polohy.



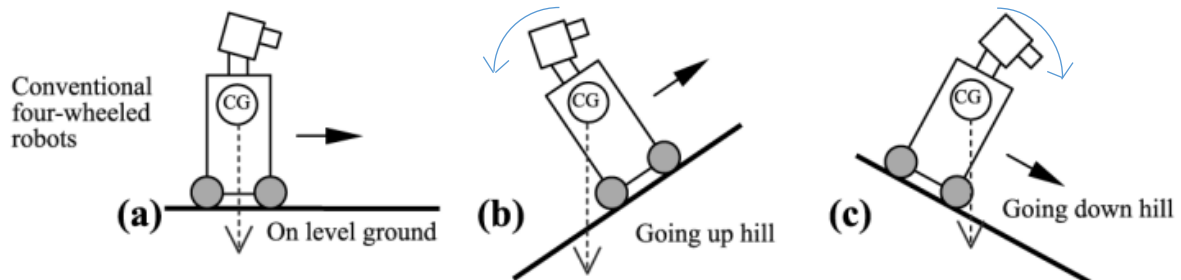
Obr. 4-2 Klasické kyvadlo [3]



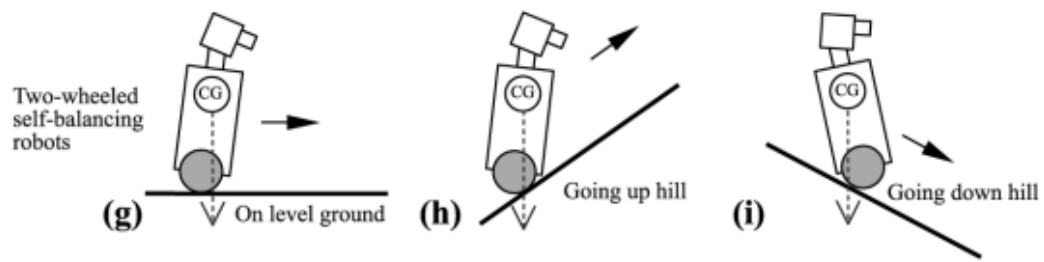
Obr. 4-1 Inverzní kyvadlo [3]

Má těžiště nad bodem uchycení. Při vychýlení z klidové polohy se do ní nevrací zpět, ale odchylka se stále zvětšuje. Tomuto chování se říká, že systém je nestabilní. Matematické pojmy, jako inverzní kyvadlo vám možná nic neříkají. Určitě jste se s ním již setkali, když jste balancovali například otočené koště na dlani a snažili se je udržet ve vzpřímené poloze. Museli jste provádět určité korekce, abyste „podběhli“ padající rameno a tím je vrátili zpět do kolmé polohy. Další poznatek, na který jste mohli přijít, je ten, že jste tyto korekce museli provádět neustále, pozorovat kam koště padá a balancovat.

Hlavní výhodou balancujících robotů oproti klasickým čtyřkolovým je v trakci kol. Balancující robot při jízdě z kopce a do kopce má pořád těžiště nad koly. Oproti čtyřkolovým robotům, kteří při stejné jízdě ztrácí smykové tření s podložkou a mohou se i převrátit. [1]



Obr. 4-3 Těžiště u čtyřkolových robotů [1]

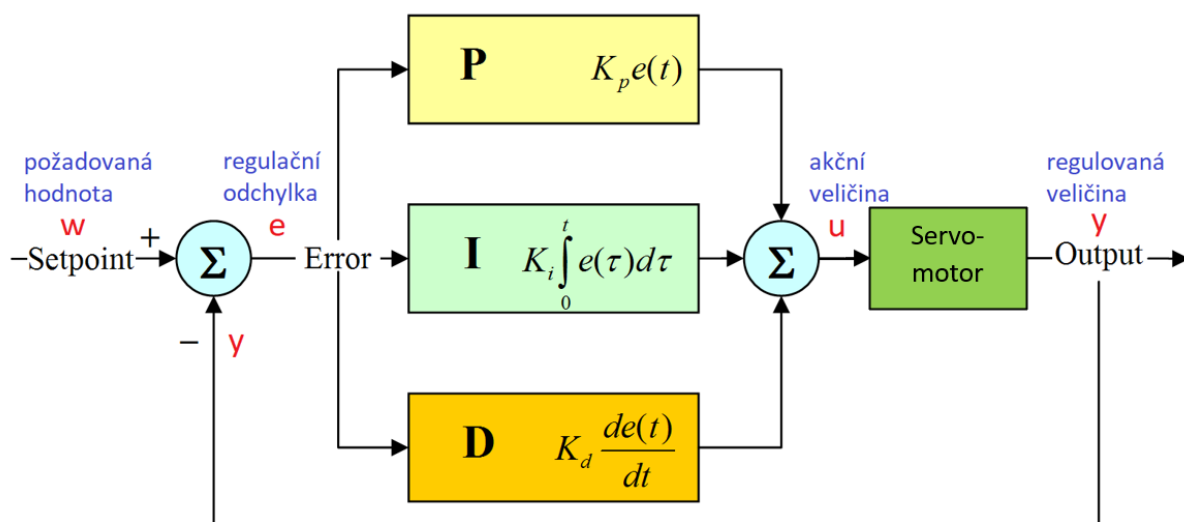


Obr. 4-4 Těžiště u balancujících robotů [1]

5 Stručný popis PID regulace

Pro člověka je naučit se udržovat košičku ve svislé poloze intuitivní. Po pár pokusech nám to jde lépe a lépe. Jak ale naučíme řídicí obvod robota snímat svou polohu a udržovat se ve vzpřímeném stavu? Existují sice samoučící se systémy, ale jejich teorie jde nad rámec cíle této práce. Strojové učení nechme na nějakou další práci do budoucna. V tomto návodu bude důležité parametry měnit programátor.

Zde přichází pojem **regulace**. V podstatě jde o způsob udržování hodnot v daném systému v požadovaných mezích. Na následujícím obrázku je schéma regulovaného obvodu.



Obr. 5-1 Blokové schéma PID regulace [2]

Na základě výstupu regulátoru akční člen reaguje na akční veličinu a opět ovlivňuje vstup zpětnou vazbou.

V našem případě bude použit PID regulátor. Jak vyplývá již z názvu, skládá se ze tří částí:

1. P – proporcionální složka;
2. I – integrační složka;
3. D – derivační složka.

1. P složka – základem každé regulace je **proporcionální** složka. Název vypovídá o schopnosti **úměrně** reagovat na změny. Reakce na poruchy je přímo úměrná velikosti odchylky od požadované hodnoty. Při malé odchylce je reakce malá a naopak. Pokud je použita samostatně, může vzniknout trvalá regulační odchylka.
2. I složka – je poněkud složitější na pochopení. Jejím cílem je vyrovnat chybu, která se akumuluje v průběhu času. Akční zásah této složky se nasčítává v čase. Nevýhodou integrační složky je, že zpomaluje rychlost reakce.
3. D složka – je poslední částí regulátoru. Tahle složka je ta nejmagičtější. Když se dobře nastaví, může se urychlit regulace. Jakoby předpovídá budoucnost. Jednoduše reaguje na rychlost změny. Tato složka reaguje na rychlost změny regulační odchylky. Čím je větší změna regulační odchylky oproti předcházející hodnotě, tím větší bude akční zásah.

Digitální počítače nepracují spojitě, ale pracují v daných časových intervalech. Proto u časových údajů není v následujícím textu uváděn symbol t , ale i -tý časový interval t_i .

5.1 Jak naprogramovat P složku PID regulátoru

Nejprve potřebujeme zjistit odchylku e regulované veličiny y od její požadované hodnoty w .

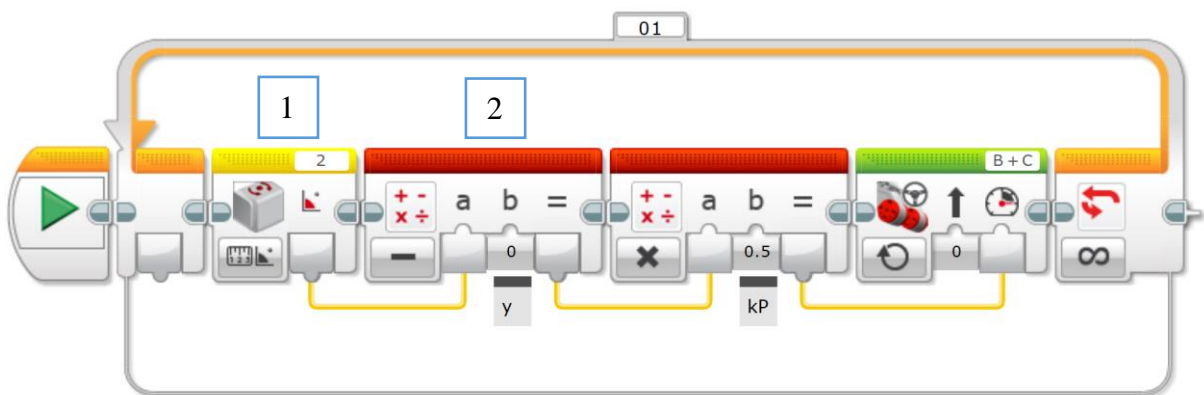
$$e(t_i) = w(t_i) - y(t_i) \quad (1)$$

Z ní regulátor následně vypočítá velikost akční veličiny u . Stačí jen odchylku vynásobit konstantou k_P .

$$P(t_i) = k_P * e(t_i) \quad (2)$$

Vypočítaná proporcionální složka P je přímo akční veličina u a určuje velikost akčního zásahu. Akčním zásahem je výkon motorů.

Ve vývojovém prostředí pro EV3 lze proporcionální složku regulátoru realizovat následovně.



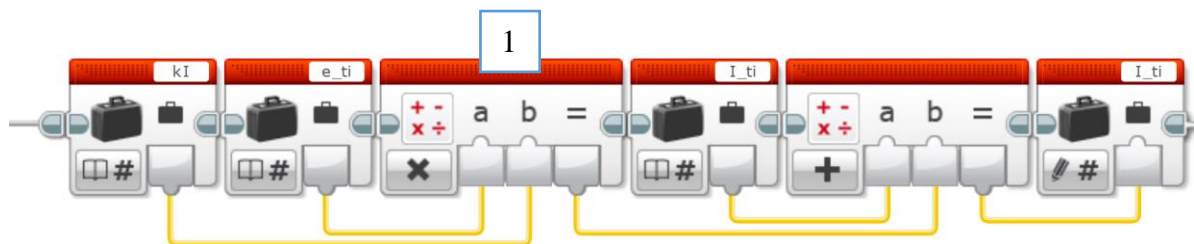
Obr. 5-2 Proporcionální regulace v prostředí LEGO MINDSTORMS EV3

Blok 1 zjišťuje velikost regulované veličiny $y(t_i)$, v našem případě je to úhel naklonění robota. Tato hodnota se v bloku 2 odečte od požadované hodnoty $w(t_i)$ a výsledkem je regulační odchylka $e(t_i)$. Ta se v bloku 3 násobí proporcionální konstantou k_P . Výsledná hodnota určuje výkon a směr otáčení motorů. Tento proces probíhá neustále v uzavřené smyčce.

5.2 Jak naprogramovat I složku PID regulátoru

Integraci musíme provádět diskretním způsobem. Postupným přičítáním dílčích odchylek v daných intervalech.

$$I(t_i) = k_i * e(t_i) + I(t_{i-1}) \quad (3)$$



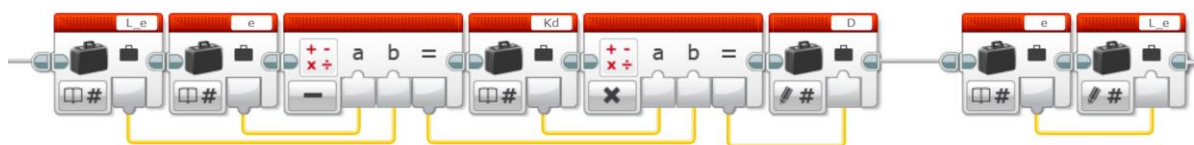
Obr. 5-3 Realizace integrační složky v prostředí LEGO MINDSTORMS EV3

Odchylku získáme stejným způsobem jako u proporcionální složky. Tu v bloku 1 vynásobíme integrační konstantou a výsledek přičteme k integrační složce z přecházejícího časového intervalu t_{i-1} .

5.3 Jak naprogramovat D složku PID regulátoru

Rychlost změny regulační odchylky $e(t_i)$ zjišťujeme podílem změny odchylky a délky časového intervalu. Délka časového intervalu je v našem případě konstantní, a proto ji můžeme zahrnout do derivační konstanty a rozdíl dělit jedničkou. Pak rychlost změny regulační odchylky získáme odečtením odchylky v předcházejícím časovém intervalu od současné.

$$D(t_i) = k_D * (e(t_i) - e(t_{i-1})) \quad (4)$$



Obr. 5-4 Realizace derivační složky v prostředí LEGO MINDSTORMS EV3

Zde si musíme vytvořit tedy dvě proměnné pro odchylku. Tu, kterou vyčteme v současném průběhu regulační smyčky (e_{ti}), a tu, kterou jsme si uložili z průběhu minulého (e_{ti-1}). Jak bylo řečeno v předcházejícím odstavci, tyto odchylky od sebe odečteme a jejich rozdíl poté vynásobíme derivační konstantou k_D a tím získáme derivační složku. Nesmíme zapomenout na aktualizování odchylky. A to tak, že odchylka v současném průběhu regulační smyčky se stane v dalším průběhu odchylkou minulou.

6 Trochu teorie k inverznímu kyvadlu

V předcházející kapitole je už naznačeno, jakým způsobem se robot (inverzní kyvadlo) udržuje ve svislé poloze. Při vychýlení o určitý úhel se těžiště ve vodorovném směru vzdálí od svislé osy o určitou vzdálenost, viz Obr. 4-1 Inverzní kyvadlo [3]

a Obr. 4-4. Aby se kyvadlo vrátilo do klidové polohy, je nutné přesunout osu otáčení (kola robota) pod těžiště.
Obr. 4-1 Inverzní kyvadlo [3]

Při regulaci inverzního kyvadla je zásadním parametrem délka ramene neboli vzdálenost těžiště robota od osy kol. Pokud budeme mít nízkého robota s nízkou uloženým těžištěm, bude regulace náročnější. Při vychýlení robota ze svislé polohy o určitý úhel se těžiště u vyššího robota odchýlí od svislé osy o větší vzdálenost v horizontálním směru než u nízkého robota. Abychom dostali osu kol opět pod těžiště, musí robot popojet o stejnou vzdálenost stejným směrem. Malý robot vyžaduje popojet o krátkou vzdálenost. Aby se servomotor vzhledem k různým vnitřním odporům a třecím silám vůbec rozjel, vyžaduje určitý základní výkon, který nezávisí na velikosti robota. To způsobí nepřiměřenou akceleraci robota a při krátké vzdálenosti snadno dojde k překmitu. V důsledku toho musí ihned regulovat na druhou stranu. Tím je regulace u nízkého robota mnohem náročnější.

Další důvod, proč zvolit robota s výše položeným těžištěm, je delší doba kyvu kyvadla s delším ramenem. Při vychýlení ramene o stejný úhel je délka oblouku, po kterém se těžiště pohybuje, delší. Z toho důvodu má robot více času na vrácení kyvadla do svislé polohy. To si můžete vyzkoušet postavením krátkého objektu na dlaň a zkusit jej udržovat ve vzpřímené poloze neboli balancovat. Zjistíte, že je to mnohem náročnější než u dlouhého objektu.

Z výše uvedených důvodů chceme mít co nejvyššího robota, ale nesmíme zajít do extrému. Takový robot by byl těžší, méně skladný a ne tak obratný.

7 Balancující robot krok za krokem

V této kapitole je postupně popsána stavba balancujícího robota krok za krokem, s bližším popisem myšlenkových pochodů. Následuje popis řídicího obvodu, také krok za krokem.

7.1 Jak bude robot vypadat

Než začneme robota stavět, musíme si ujasnit, jak bude náš robot vypadat. Potom musíme zvolit vhodnou výrobní metodu. V úvahu přichází dvě možnosti:

1. Můžeme nakreslit konstrukci robota a díly vyrobit na obráběcích strojích z polotovárů. Výhodou tohoto řešení je plná kontrola nad výslednou podobou robota, ale výroba je velmi náročná, cena bude vysoká a stavba zabere hodně času. Pro řízení robota existuje bezpočet řídicích systémů. V úvahu přicházejí například oblíbené Arduino, Teensy, Raspberry PI a bezpočet dalších. Při výběru bychom měli zvážit potřebné parametry, protože nadbytečný výkon znamená zbytečné navýšení ceny bez významného přínosu.
2. Nebo využijeme nějakou existující stavebnici, např. LEGO, Merkur, VEX apod. Některé stavebnice přímo obsahují potřebnou řídicí jednotku. Taková stavba bude mnohem jednodušší, levnější a zabere podstatně méně času. Designově jsme však omezeni díly, které stavebnice nabízí.

Pro účely tohoto návodu byla vybrána stavebnice LEGO MINDSTORMS EV3, viz předcházející odstavce, která obsahuje, jak potřebné stavební konstrukční díly, tak i snímače a řídicí jednotku. Použitím všech dílů z jednoho typu stavebnice se vyhneme případné nekompatibilitě jednotlivých prvků.

7.2 Krok 1. – použité prvky

K realizaci robota budeme potřebovat následující prvky, jež jsou pro funkci nezbytné:

1. dvě kola;
2. dva servomotory;
3. snímač úhlu naklonění robota;

4. řídicí jednotka;

7.3 Výběr vhodných kol

Pokud použijeme velká kola, robot bude jezdit rychleji. Nevýhodou bude malá citlivost a tím větší pásmo necitlivosti. V převodech servomotorů se vyskytuje vůle, kdy motor nezabírá. Ta tvoří určitý úhel. Tomu odpovídá u malých kol krátký oblouk po obvodu, u velkých je tomu naopak. Proto je vhodné použít kola s menším průměrem. Nejmenší kola u stavebnice LEGO MINDSTORMS EV3, která při stavbě použijeme, mají průměr 42 mm. Vhodné je použít kola s gumovými pneumatikami, aby na hladkých površích neklouzala.

7.4 Popis servomotorů

Ideální bude použít servomotory, protože krom elektromotoru obsahují také převodovku a snímač úhlu pootočení. Ve stavebnici jsou dva pojezdové servomotory a jeden pomocný. Aby nám ke stavbě stačila jen jedna stavebnice, použijeme pojezdové.

Pokud stavíme robota podle bodu 1. v kapitole 7.1, pak jeho pohon může být realizován také krokovými motory. Je třeba brát v úvahu jejich větší hmotnost. Toto řešení je vhodnější pro větší konstrukce.

7.5 Popis snímače úhlu naklonění robota

Pro snímání úhlu naklonění je využit Gyroskop. Tento snímač snímá úhlové zrychlení rotačního pohybu. Jedná se o elektronický senzor, jehož elektronika přepočítává úhlové zrychlení na úhel.

Umístění gyroskopu je důležité. Jedná se o nejdůležitější snímač přímo potřebný k balancování. Při použití jednoosého gyroskopu dejte pozor, aby se robot vychyloval ve snímané ose. U tříosého gyroskopu si vybereme jednu ze snímaných os. Měřená osa gyroskopu musí být rovnoběžná s osou otáčení kol. Jinak bychom museli přepočítávat naměřené údaje do osy vychylování.

Vzhledem k tomu, že je použit elektronický gyroskop, je důležité, aby byl natočen do nulové polohy v okamžiku, kdy je spojnice těžiště a osy kol ve svislé poloze. Jinak bude na gyroskop trvale působit tíhové zrychlení a úhel vlivem integrace bude narůstat, i když bude robot v klidové poloze.

7.6 Popis řídicí jednotky

Řídicí jednotka je mozek našeho robota. Je to místo, kde se odehrávají všechny výpočty. Zásadně určuje kvalitu regulace a měla by být v souladu s citlivostí použitého snímače úhlu naklonění a servomotorů. Ve stavebnici LEGO MINDSTORMS EV3 je pouze jedna řídicí jednotka, čímž se nám výběr zjednodušuje. Má zaručenou kompatibilitu se snímači, servomotory a rychlost komunikace je optimalizována.

7.7 Krok 2. – stavba robota

7.8 Rozchod kol

Tento parametr nemá na balancování přímý vliv, jelikož balancujeme jen jednu osu, a to dopředu a dozadu. Hraje však roli při otáčení robota. Když kola budou mít velký rozchod, za určitý časový interval se otočí o menší úhel.

7.9 Výška robota

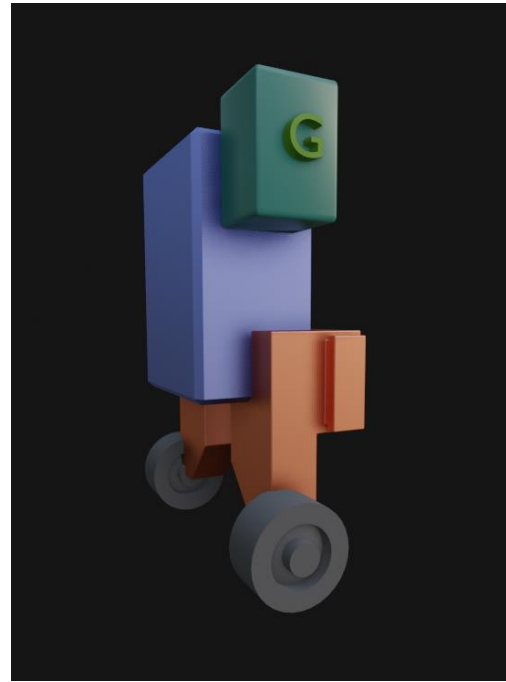
Jak bylo uvedeno již v kapitole 6, výhodnější bude vyšší robot, ale přiměřeně použité stavebnici. Výšku bude také ovlivňovat navržený design robota.

7.10 Uchycení gyroskopu

Na obrázku Obr. 7-1 můžete vidět, kam byl umístěn gyroskop. Na 3D modelu robota je označen písmenem G. Doporučené umístění již bylo popsáno v kapitole 7.5. Vzdálenost od osy kol není důležitá, rozhodující pro regulaci je úhel naklonění robota.

7.11 Design robota

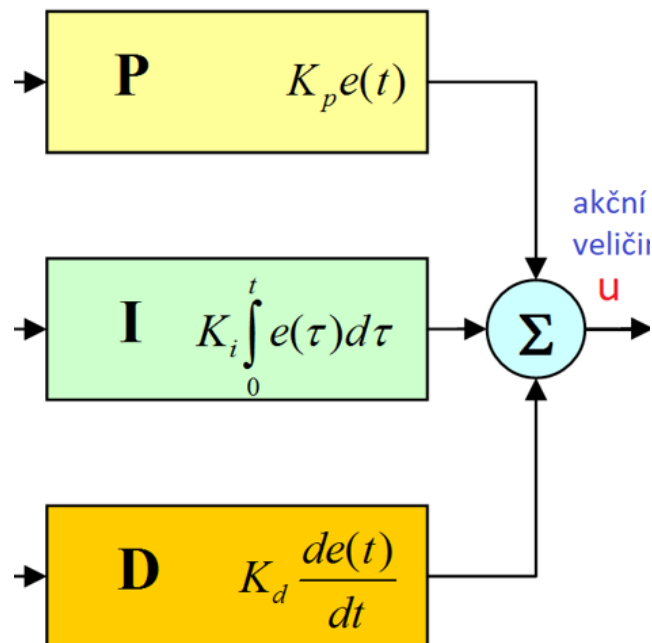
Zbytek konstrukce robota závisí pouze na představivosti konstruktéra. Po dokončení stavby můžeme přejít k dalšímu kroku a tím je naprogramování robota.



Obr. 7-1 Model základu balancujícího robota

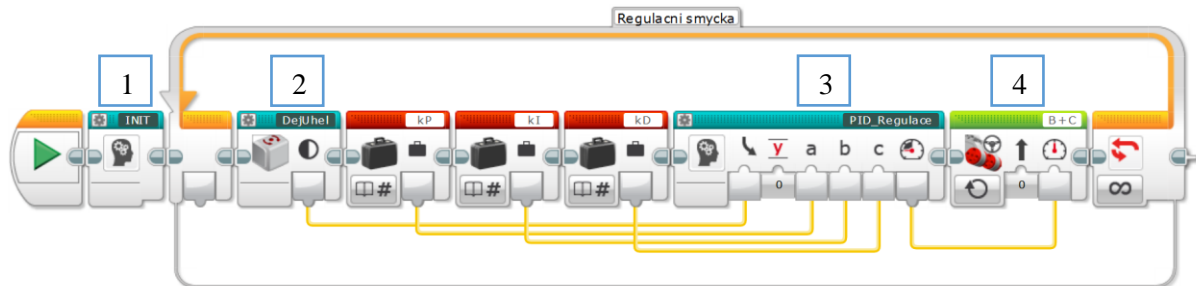
7.12 Krok 3. – naprogramování robota

Řídicí kostka slouží jako regulátor. Použijeme PID regulátor, který se skládá ze složek P, I a D. Jejich princip a realizace je vysvětlena v kapitole 5. Výslednou akční veličinu (výkon servomotorů) získáme sečtením jednotlivých složek dohromady, viz následující obrázek.



Obr. 7-2 Sečtení složek v PID regulaci [2]

Výsledná podoba programu je uvedena na obrázku Obr. 7-3. Po zapnutí robota a spuštění programu se provede první blok. V bloku 1 (INIT) se provádí inicializace. Ta zapíše počáteční hodnoty do proměnných, nastaví hodnoty konstant a resetuje se gyroskop. Po chvíli se ozve zvukový signál a zprůměruje se počáteční hodnota úhlu. V té době již musí být robot přidržován ve svislé poloze. Průměrování, a tím i INIT, se ukončí zvukovým signálem. Nyní se dostáváme do regulační smyčky.



Obr. 7-3 Hlavní program pro balancování na místě

V regulační smyčce se opakovaně vykonává následující řetězec příkazů. V bloku 2 získáváme pomocí gyroskopu aktuální hodnotu úhlu. Ta vznikne průměrováním pěti posledních hodnot. Důvodem je potlačení náhodných chyb.

V bloku 3 se provádí výpočet akční veličiny. Vstupními údaji je nejenom aktuální hodnota úhlu, ale i hodnoty konstant pro jednotlivé složky PID regulace. Postup výpočtu je uveden v kapitole 5. Výsledná akční veličina určuje výkon motorů v bloku 4.

7.13 Krok 4. – ladění regulátoru

Po spuštění výše vytvořeného programu neočekávejte bezproblémové balancování na místě. Hodnoty konstant byly zvoleny více méně náhodně podle předešlých zkušeností. Takový robot bude nejpravděpodobněji padat místo balancování na místě. Konstanty bude nutno postupně upravovat podle chování robota. Tato činnost se nazývá ladění.

Jako první začneme ladit proporcionální konstantu při vypnuté integrační a derivační složce. To zaručíme vynulováním konstanty k_I a k_D . Tím docílíme, že nebudou hrát tyto složky žádnou roli na výsledek regulace. Pro proporcionální konstantu k_P můžeme podle mých zkušeností nastavit jako výchozí hodnotu 0,5. Postupně zvyšujeme tuto hodnotu. Pro začátek doporučuji krok 0,1. Takto budeme postupovat, dokud se robot nezačne rozkmitávat. To znamená, že je hodnota k_P již moc velká. Mějte na vědomí, že po přidání zbývajících složek bude akční zásah ještě větší. To znamená, že před přidáním dalších složek budeme muset konstantu k_P snížit. Doporučuji na 80 procent prvně vyladěné hodnoty. Pojmeme rozkmitávání je myšleno nekontrolovatelné kývání ze strany na stranu bez naděje na ustálení.

Dále přidáme integrační složku. Její přínos je popsán v kapitole 5 bod 2. Pro začátek můžeme zkusit například hodnotu integrační konstanty k_I 1,5. Tuto hodnotu budeme postupně zvyšovat v krocích po 0,3. Že je nastavená hodnota již příliš velká zjistíme z opětovného rozkmitávání robota. V tomto okamžiku začneme hodnotu k_I snižovat po menších krocích, např. 0,05. Snižování ukončíme v okamžiku, kdy je kmitání co nejmenší. V mém případě se jako optimální ukázala hodnota 0,67 pro konstantu k_P a 14,2 pro k_I .

Poslední složkou k odladění je derivační. Její chování je popsáno v kapitole 5 bod 3. Tím že „předvídá“ chování robota urychluje jeho návrat do klidové polohy při vychýlení. Při jejím ladění si dejte pozor na její agresivní povahu. I při malých změnách konstanty k_D je vliv této složky obrovský. Doporučuji začít na hodnotě 0,001 a upravovat ji po jedné desetitisícině. Před laděním této složky je vhodné proporcionalní konstantu k_P ještě o 10 procent snížit. V mém případě jsem skončil s laděním na hodnotě 0,61 pro k_P a okolo 0,001 pro k_D .

Výsledné nastavení regulátoru vypadá takto:

- $k_P = 0,61$
- $k_I = 14,2$
- $k_D = 0,001$

Nyní je již robot schopen balancovat na místě.

8 Rozšíření schopností robota

Balancování na místě je zásadní, ale pro nás určitě nedostačující. Jistě budete chtít robota rozšířit o další schopnosti, např. o pohyb vpřed a vzad, kdy robot sleduje ruku, umístěnou před ním v určité vzdálenosti a o řízení robota dálkovým ovládním. V úvahu také přichází možnost orientace robota v prostoru pomocí kompasu. Pro tyto účely se hodí např. ultrazvukový snímač vzdálenosti, kompas a dálkové ovládní.

8.1 Jak funguje ultrazvukový snímač

Používáme jej pro měření vzdálenosti, např. sledování ruky před robotem nebo vyhýbání se překážkám. Funguje na podobném principu, jako když se netopýr vyhýbá překážkám. Snímač vyšle ultrazvukový impuls a měří čas do návratu po odrazu od překážky. Z naměřené časové prodlevy lze vypočítat vzdálenost, jelikož rychlost šíření zvuku ve vzduchu za normálních podmínek je konstantní.

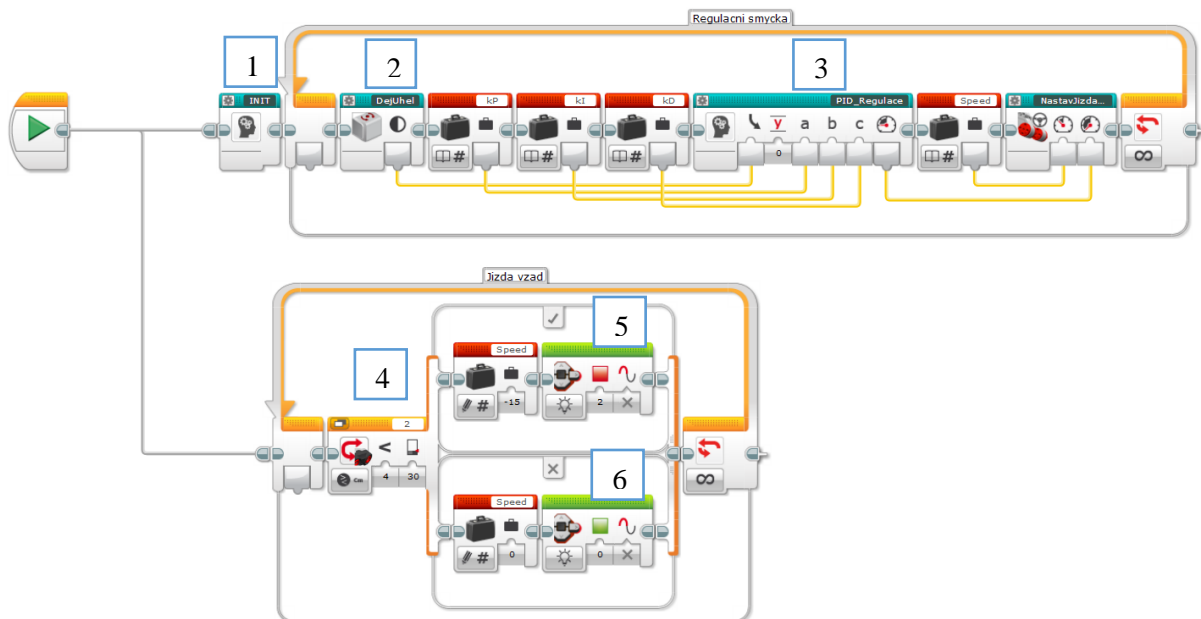
Ultrazvukový snímač ve vybrané stavebnici je schopen spolehlivě snímat vzdálenosti do dvou metrů. Pro ukázkou jízdy budeme využívat vzdálenost 30 cm. Pokud robot zaznamená objekt, v našem případě ruku, v menší vzdálenosti než 30 cm, tak začne couvat. Pokud je objekt zaznamenán v rozmezí 30 až 40 cm, robot pojedede dopředu.

8.2 Jak zajistit pohyb robota

I my lidé, když stojíme na místě, tak musíme vyrovnávat polohu těžiště, abychom zůstali ve vzpřímené poloze. Na rozdíl od běhu, kde se přirozeně nakláníme dopředu, abychom udržovali dopředný pohyb. To samé musíme zajistit u robota. Je nutné ho záměrně naklonit mimo svislou osu ve směru pohybu.

Jízdu lze vytvořit dvěma způsoby:

1. Změníme požadovanou hodnotu regulované veličiny. Tato metoda je elegantnější, ale vyladit odpovídající změnu regulované veličiny, vzhledem k požadované rychlosti pohybu, zabere více času.
2. Na konci smyčky požadovanou rychlost přičteme k akčnímu zásahu (výkon motorů). Máme tak lepší kontrolu nad rychlostí pohybu robota. Integrační složka zde způsobuje mírně kývavý pohyb. To může být případnou nevýhodou. Dále je vysvětlen tento způsob pohybu.



Obr. 8-1 Program pro jízdu vzad pomocí ultrazvukového snímače

Jak je vidět na obrázku Obr. 8-1, realizace je jednoduchá. Sečteme výslednou akční veličinu z bloku 3 s hodnotou proměnné *speed*. Proměnná *speed* je požadovaná rychlost pohybu robota. Nastavování proměnné *speed* se provádí mimo balancující smyčku ve smyčce *jízda vzad*. Blok 4 má dvě větve. Určující podmínkou, která větev se vykoná, je vzdálenost naměřená ultrazvukovým snímačem. Pokud je vzdálenost menší než 30 cm, vykonají se příkazy v horní větvi a proměnná *speed* se nastaví na požadovanou hodnotu, v našem příkladu na -15. Jízda vzad je signalizována červeným podsvícením kláves řídicí jednotky viz červený čtvereček v bloku 5. V opačném případě se vykonají příkazy spodní větve, kde se do proměnné *speed* zapisuje hodnota 0. Tento stav je signalizován zeleným podsvícením kláves řídicí jednotky, viz zelený čtvereček v bloku 6.

Obdobně by se řešil pohyb vpřed.

8.3 Dálkové ovládání

Jako jsme v předchozím odstavci proměnou *speed* nastavovali na základě naměřené vzdálenosti ultrazvukovým snímačem, podobně lze tuto hodnotu měnit na základně požadavku přijatého z dálkového ovládání. Robota lze řídit dvěma způsoby:

1. Prvním způsobem je tzv. tankové řízení, kde levým joystickem ovládáme rychlost a směr otáčení levého kola a pravým joystickem pravého kola. V tomto případě vytvoříme dvě proměnné jednu pro výkon levého kola a druhou pro výkon pravého kola.
2. Při klasickém způsobu řízení jedním joystickem ovládáme rychlost vpřed a vzad. Druhým joystickem ovládáme zatáčení doleva a doprava. Opět vytvoříme dvě proměnné. Pro každý motor jednu. Jízda vpřed a vzad se realizuje přičtením přečtené hodnoty z prvního joysticku k oběma proměnným. Pro zatáčení hodnotu získanou z druhého joysticku k jedné proměnné přičteme a od druhé odečteme. Doporučuji přičítat jen 75 procent hodnoty z druhého joysticku dálkového ovládání. Zatáčení není pak tak ostré. Dále je dobré vyzkoušet, zda robot zatáčí na správnou stranu a popřípadě změnit ke které proměnné přičítáme a od které odečítáme.

Nakonec doporučuji vytvořit pásmo necitlivosti okolo nuly, protože páčky joysticků se po uvolnění nemusí vždy vrátit úplně do nulové polohy. Hodnota ± 5 by u dálkového ovládání pro LEGO EV3 měla stačit.

9 Závěr

I když je tato práce určena spíše laikům, doporučuji alespoň minimální znalosti z automatizace a programování. Jejím cílem bylo vytvořit návod a naučit čtenáře principům stavby robotů jejich programování, což se podařilo. Poslední kapitola mohla být rozvedena více dopodrobna, ale hlavní cíl, vytvoření balancování, byl splněn. Dále jsem dal svoji práci přečíst žákům nižších ročníků a neměli problém vytvořit si vlastního robota. Přeji hodně úspěchů všem, kteří se chtějí zdokonalit v robotice.

Tuto práci by bylo vhodné doplnit o návod, jak dovybavit robota o navigaci v prostoru. Dále lze robota doplnit o funkce rukou a popřípadě komunikaci s jinými zařízeními.

10 Použitá literatura

BIERNÁTOVÁ, Olga, Jan SKŮPA. Bibliografické odkazy a citace dokumentů dle ČSN ISO 690 (01 0197) platné od 1. dubna 2011. [online]. c2004-2011 [cit. 2016-10-07]. Dostupné z: <http://citace.com/soubory/csniso690-interpretace.pdf>.

ROUBAL, Jiří a Petr HUŠEK. Regulační technika v příkladech. Praha: Sortiment BEN, 2011. ISBN 978-80-7300-260-2.

BURTON, James. Building a Two-Wheel Balancing Robot | with LG XBOOM Go Speakers. YouTube [online]. London: YouTube, 2019, 2019 [cit. 2022-12-15]. Dostupné z: <https://www.youtube.com/watch?v=TN6u-D992sw>

35, Builderdude. How to Defy Physics With LEGO 51515 - PID Self Balancing Robot. YouTube [online]. America: YouTube, 2021, 2021 [cit. 2022-12-15]. Dostupné z: <https://www.youtube.com/watch?v=JMpmpto3AM>

[1] WITZAND, Steven. Coordinated LEGO Segways [online]. New South Wales, 2009 [cit. 2023-03-24]. Dostupné z: <https://github.com/kdawgwilk/gelway/blob/master/Coordinated%20LEGO%20Segways%20-%20Thesis.pdf>. Bakalářská práce. School of Information Technology & Electrical Engineering The University of New South Wales Australian Defence Force Academy.

[2] <https://electronics.stackexchange.com/questions/295077/how-do-i-get-a-closed-loop-pid-controller-from-a-transfer-function>

[3] [http://www.theworkofgodchildren.org/collaboration/images/8/8b/Pendulum_2_\(PSF\).png](http://www.theworkofgodchildren.org/collaboration/images/8/8b/Pendulum_2_(PSF).png)

11 Seznam obrázků

Obr. 4-1 Inverzní kyvadlo [3]	4
Obr. 4-2 Klasické kyvadlo [3].....	4
Obr. 4-3 Těžiště u čtyřkolových robotů [1].....	4
Obr. 4-4 Těžiště u balancujících robotů [1]	4
Obr. 5-1 Blokové schéma PID regulace [2]	5
Obr. 5-2 Proporcionální regulace v prostředí LEGO MINDSOTRMS EV3	6
Obr. 5-3 Realizace integrační složky v prostředí LEGO MINDSTORMS EV3.....	7
Obr. 5-4 Realizace derivační složky v prostředí LEGO MINDSTORMS EV3.....	7
Obr. 7-1 Model základu balancujícího robota.....	10
Obr. 7-2 Sečtení složek v PID regulaci [2]	10
Obr. 7-3 Hlavní program pro balancování na místě.....	11
Obr. 8-1 Program pro jízdu vzad pomocí ultrazvukového snímače.....	13