

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ
TECHNOLÓGIE

EVIDENČNÉ ČÍSLO: FCHPT-5414-61556

Konštrukcia a riadenie inverzného kyvadla

DIPLOMOVÁ PRÁCA

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA CHEMICKÉJ A POTRAVINÁRSKEJ
TECHNOLÓGIE

EVIDENČNÉ ČÍSLO: FCHPT-5414-61556

Konštrukcia a riadenie inverzného kyvadla

DIPLOMOVÁ PRÁCA

Študijný program: automatizácia a informatizácia v chémii a potravinárstve

Študijný odbor: 5.2.14. automatizácia

Školiace pracovisko: Ústav informatizácie, automatizácie a matematiky

Vedúci práce: Ing. Martin Kalúz, PhD.



ZADANIE DIPLOMOVEJ PRÁCE

Autor práce: Bc. Peter Bakaráč
Študijný program: automatizácia a informatizácia v chémii a potravinárstve
Študijný odbor: 5.2.14. automatizácia
Evidenčné číslo: FCHPT-5414-61556
ID študenta: 61556

Vedúci práce: Ing. Martin Kalúz, PhD.

Miesto vypracovania: Ústav informatizácie, automatizácie a matematiky

Názov práce: **Konštrukcia a riadenie inverzného kyvadla**

Jazyk, v ktorom sa práca vypracuje: slovenský jazyk

Špecifikácia zadania: Cieľom projektu je vytvorenie nového laboratórneho procesu (inverzného kyvadla) a návrh algoritmov pre jeho riadenie.

Inverzné kyvadlo je prirodzene nestabilný mechanický systém, ktorý pozostáva u pohyblivej základne, na ktorej je uchytené kyvadlo, otočené smerom nahor. Takýto proces je možné mechanicky zostaviť z vopred navrhnutých stavebných dielov, ktoré tvoria hlavnú kosť konštrukcie, a z vhodných snímacích a pohonových prvkov.

Jednotlivé úlohy zadania práce sú:
návrh a zostavenie systému inverzného kyvadla; vytvorenie ovládacieho mechanizmu základne a mechanizmu na čítanie náklonu kyvadla; prepojenie systému s počítačom / riadiacou elektronikou; návrh a testovanie riadiacich algoritmov.

Rozsah práce: 50

Literatúra:

1. KOLENČÍK, M. -- ŽÁKOVÁ, K. A Contribution to Remote Control of Inverted Pendulum. In *17th Mediterranean Conference on Control and Automation: Thessaloniki, Greece, 24.-26.6.2009*. Piscataway : IEEE, 2009, s. 1433-1438. ISBN 978-1-4244-4685-8.
2. MATES, M. -- NYEKY, D. -- HONEK, M. -- ROHAČ-ILKIV, B. Control of inverted pendulum laboratory system using stepper motor. *Strojnícky časopis = Journal of Mechanical engineering*, 60. s. 211--219. ISSN 0039-2472.
3. MARGOLIS, M. *Arduino Cookbook: Recipes to Begin, Expand, and Enhance Your Projects*. Sebastopol: O'Reilly Media, 2013. 699 s. ISBN 978-1-449-31387-6.

Dátum zadania: 13. 02. 2017

Dátum odovzdania: 21. 05. 2017

Bc. Peter Bakaráč
študent

prof. Ing. Miroslav Fikar, DrSc.
vedúci pracoviska

prof. Ing. Miroslav Fikar, DrSc.
garant študijného programu

Podakovanie

Týmto by som sa chcel poďakovať môjmu vedúcemu diplomovej práce Ing. Martinovi Kalúzovi, PhD. za jeho ochotu vždy pomôcť, za jeho odborné rady, vedenie a inšpiráciu.

Abstrakt

Táto práca sa zaoberá konštrukciou reálneho laboratórneho procesu inverzného kyvadla a vytvorením návodu, podľa ktorého sa dá tento proces znova zostrojiť. Aby bolo možné tento proces riadiť, tak sú v tejto práci definované a predstavené možnosti aplikácie riadiacich algoritmov cez užívateľské rozhranie v programe MATLAB, pomocou ktorého bude možné tieto riadiace algoritmy implementovať a sledovať priebeh riadenia numericky aj graficky. Práca sa ďalej zaoberá odvodením dynamického matematického modelu pre proces inverzného kyvadla a jeho prispôbením vytvorenému reálnemu procesu. Vo fáze keď už je hotový reálny proces, užívateľské rozhranie a je k dispozícii matematický model, je možné pristúpiť k návrhu riadiacich algoritmov a ich implementácii. K tomuto procesu bol navrhnutý PID regulátor pomocou metódy umiestnenia pólov a robustný PI regulátor. Na záver sa práca v krátkosti venuje návrhu algoritmu pre nábeh kyvadla, tzv. *swing up*.

Kľúčové slová: inverzné kyvadlo, umiestnenie pólov, PID, laboratórny proces, reálny proces, swing up, MATLAB rozhranie

Abstract

This thesis deals with the construction of the real laboratory process of an inverted pendulum and the creation of an assembly guide that can be used to make a copy of the device. To be able to control this process, this thesis defines and introduces the possibilities of the application of the control algorithms via the user interface in the program MATLAB, that allows the implementation of these control algorithms. The interface provides the user with the possibility to control the process and monitor the parameters in both, the numerical and graphical way. The thesis further deals with the derivation of the dynamical mathematical model for the process of an inverted pendulum and its adaptation to the behavior of the real process. In the phase when the real process is available, and the user interface and the mathematical model are done, it is possible to proceed to the design of the control algorithms and their implementation. For this process, the PID controller was designed using the pole placement method and the robust PI controller. In the end, the thesis briefly describes the design and implementation of the swing up algorithm.

Key words: inverted pendulum, PID, pole placement, laboratory process, real process, swing up, MATLAB interface

Obsah

Úvod	13
1 Inverzné kyvadlo	16
2 Konštrukcia laboratórneho procesu	17
2.1 Komponenty potrebné na stavbu laboratórneho procesu	17
2.1.1 Akčný člen	17
2.1.2 Senzory	20
2.1.3 Elektronické časti	21
2.2 Návrh konštrukcie	23
2.2.1 Kyvadlo	25
2.2.2 Vozík	26
2.2.3 Kyvadlová časť	27
2.2.4 Pravá strana	28
2.2.5 Ľavá strana	30
2.2.6 Riadiaca časť	33
2.2.7 Dokončenie konštrukcie	35
2.3 Elektronická schéma zapojenia	35
2.4 Zoznam použitých komponentov	37
2.5 Parametre	38
3 Matematický model inverzného kyvadla	39
3.1 Linearizácia modelu	43
3.2 Verifikácia matematického modelu s reálnym procesom	45
4 Programové rozhrania	47
4.1 Implementácia riadenia v ATmega2560	50
4.1.1 Interné prerušenia	50

4.1.2	Algoritmus riadenia PSD regulátorom	53
4.2	Program na komunikáciu s PC	54
4.3	Programové rozhranie v PC	56
4.4	Zoznam a popis funkcií	57
5	Návrh riadiacich algoritmov	59
5.1	Návrh PID regulátora pomocou metódy umiestnenia pólov	59
5.1.1	Priebeh riadenia pri úlohe odstraňovania poruchy	61
5.2	Návrh robustného PI regulátora	62
6	Nábeh kyvadla	70
	Záver	73
	Literatúra	74

Úvod

Kvalita vzdelávania v odbore riadenia procesov priamo súvisí s kvalitou a rôznorodosťou laboratórneho vybavenia dostupného pre študentov. Vhodne navrhnuté laboratórne experimenty pomáhajú študentom pochopiť základy princípov správania sa systémov a osvojiť si vhodný prístup k návrhu ich riadenia. Napriek tomu, mnohé vzdelávacie inštitúcie uprednostňujú skôr využívanie počítačových simulácií. Takýto prístup je v poriadku a v určitej miere pomáha pochopiť fungovanie reálnych systémov ale neumožňuje hlbšie poznanie problémov spojených s aplikáciou riadenia na reálne procesy [1]. Preto je vhodné sústrediť sa aj na vývoj mechanických systémov, ktoré by svojimi vlastnosťami reprezentovali zariadenia s ktorými sa môžu študenti stretnúť v praxi.

Táto práca sa zaoberá vývojom takéhoto laboratórneho procesu, konkrétne laboratórneho procesu inverzného kyvadla. Tento proces pomáha pochopiť problematiku prirodzene nestabilných balančných systémov, ktoré majú rýchlu dynamiku a nelineárne správanie sa. Ich matematické modelovanie a riadenie je o to náročnejšie, preto sú tieto systémy zaujímavé nie len z pohľadu teórie riadenia ale aj informatizácie procesov, pretože sú kladené požiadavky na spoľahlivú a rýchlu výpočtovú schopnosť hardvéru.

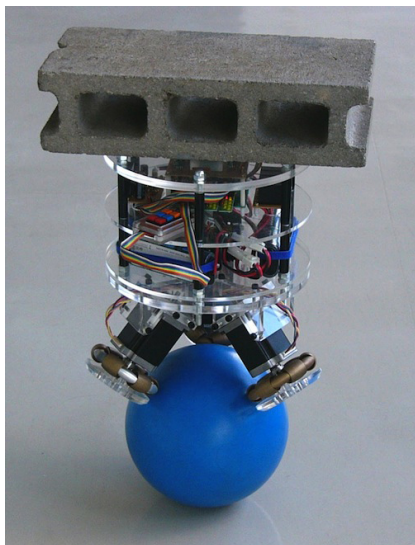
Návrh, matematické modelovanie, a riadenie balančných mechanických systémov je vo svete výskumu a vzdelávania veľmi aktuálna téma. Mnohé inštitúcie sa zaoberajú takýmito problémami, pričom ich výsledky sa priamo premietajú do praxe. Pokrok v oblasti riadenia balančných systémov umožňuje výrazne rozvíjať oblasti mechatroniky a robotiky.

Veci na Národnej univerzite v Singapúre sa zaoberajú vývojom jednokolesovej platformy na inverznom kyvadle. Jedná sa o robota, ktorý balancuje na jednom kolese s dynamikou, ktorá vychádza z inverzného kyvadla (obr. 1(b)). Títo vedci odvodili jeho dynamický matematický model, linearizovali ho a navrhli LQR riadenie, ktoré simulovali v programe MATLAB. Taktiež si uvedomujú, že simulované riadenie nemusí v praxi vôbec fungovať, kvôli praktickým problémom ako sú trenie v ložiskách a mechanických kontaktoch, obmedzený výkon a dynamika použitého akčného člena, dynamika a šum použitých snímačov, nežiadané vibrácie a pod. Z tohoto dôvodu aplikujú riadenie aj na reálnom procese, teda spomenutom jednokolesovom balančnom robote [2].

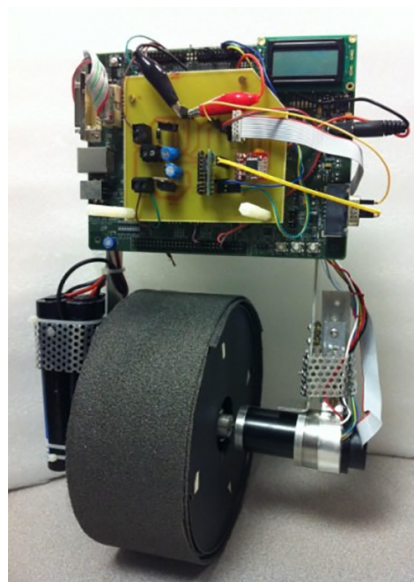
Na univerzite v Tagajo City, Tahoku Gakuin University v Japonsku vyvinuli robota, ktorý balancuje na lopte. Robot, ktorého postavili má zhruba pol metra na výšku a váži 7.5 kg (obr. 1(a)). Po stranách má tri všesmerové kolesá, ktorými balancuje na gumenej lopte. Tento robot vie viac ako len balancovať a držať stabilitu, vie sa pohybovať aj želaným smerom a otáčať sa na mieste. Jeho tvorcovia vidia jeho využitie ako mobilnú táčku, či podnos, ktorý by ľuďom pomáhal nosiť veci. Výhodou tohto robota je, že sa

nemusí otáčať keď chce meniť smer, nakoľko od dvojkoľosového balančného robota, akým je napríklad Segway, ktorý najprv potrebuje zmeniť orientáciu až potom môže ísť požadovaným smerom. Vedci a študenti na tejto univerzite vytvorili troch takýchto robotov a testovali ich tak, že museli nosiť 10 kilové betónové kocky. Roboti to nielenže zvládli ale dokázali aj spolupracovať pri nosení veľkého dreveného rámu (obr. 1(c)) [3].

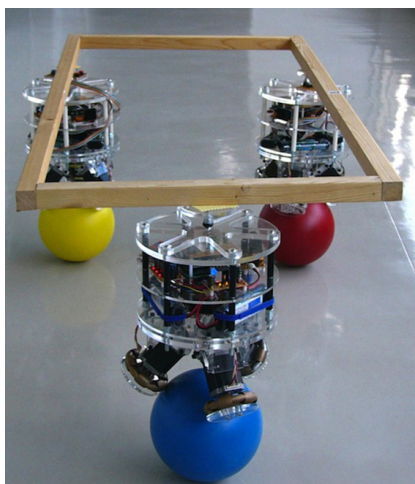
Za zmienku stojí aj poňatie problému inverzného kyvadla v podaní vedcov z univerzity ETH v Zürichu. Tí vyvinuli riadiace algoritmy, vďaka ktorým dokáže kvadrokoptéra balancovať s tyčou a udržať ju tak vo vertikálnej pozícii (obr. 1(d)). Náklon tyče monitoruje systém na snímanie pohybu v priestore. Tento systém dokáže monitorovať viacero objektov, takže je využitý aj na monitorovanie špeciálnej pomôcky, slúžiacej na dávanie pokynov pre kvadrokoptéru. Tá teda okrem toho, že dokáže balancovať s tyčou, je schopná pohybu podľa inštrukcií daných touto pomôckou [4].



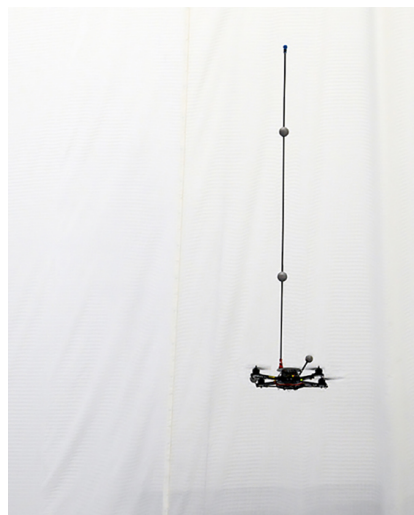
(a)



(b)



(c)

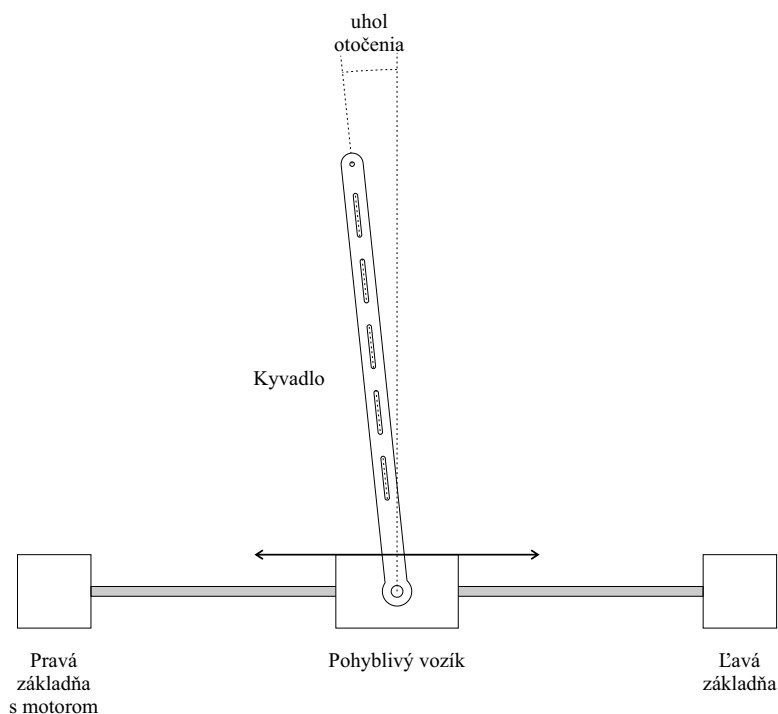


(d)

Obr. 1: Robotické zariadenia s dynamikou inverzného kyvadla

1 Inverzné kyvadlo

Zrejme už asi každý z nás sa stretol s pojmom kyvadlo. Jeho asi najbežnejšie využitie bolo v kyvadlových hodinách, kde slúžilo ako mechanický oscilátor, ekvivalent k dnešným kryštalickým oscilátorom v elektronike. Kyvadlo je teda pevný objekt, ktorý je schopný konať húpavý pohyb okolo pevnej vodorovnej osi, neprechádzajúcej jeho ťažiskom. Inverzné kyvadlo (obr. 2) je principiálne podobné tomu klasickému, s tým rozdielom, že je inverzne otočené. Inverzné kyvadlo však už nekoná húpavý pohyb ale má tendenciu spadnúť na jednu alebo druhú stranu. Ak sa však základňa kyvadla stane pohyblivou, napr. formou uchytenia na pohyblivý vozík, je možné pohybom tohoto vozíka kompenzovať pád kyvadla a udržať ho tak vo vertikálnej pozícii, teda v jeho rovnovážnej, nestabilnej polohe. Z toho vyplýva, že je to prirodzene nestabilný systém a úlohou riadenia tohto procesu je akčným zásahom meniť zrýchlenie vozíka inverzného kyvadla tak, aby nedošlo k jeho pádu na stranu.



Obr. 2: Inverzné kyvadlo

2 Konštrukcia laboratórneho procesu inverzného kyvadla

Prvým krokom v rámci tejto práce je skonštruovanie samotného laboratórneho procesu. Laboratórny proces inverzného kyvadla pozostáva z viacerých častí ako sú konštrukcia, pohon, senzory a mikroovládač. Konštrukcia je telom laboratórneho procesu, slúži ako podpora pre motor a vodiace tyče po ktorých sa hýbe vozík do ľava a do prava. Pohonom rozumieme sústavu pozostávajúcu z motora až po vozík, jednoducho celok, ktorý zaisťuje a koná pohyb. Senzory slúžia na zber dát potrebných na riadenie tohto procesu. Mikroovládač, spolu s ďalšou elektronikou tieto dáta prijíma, spracúva a posiela do PC, kde je vyhodnocovaný akčný zásah do procesu, prípadne priamo riadi proces.

Cieľom tejto kapitoly je uviesť podrobné informácie nie len o použitých komponentoch a konštrukcii, ale aj o postupe ako zhotoviť tento laboratórny proces, krok za krokom.

2.1 Komponenty potrebné na stavbu laboratórneho procesu

2.1.1 Akčný člen

Akčným členom je krokový motor (obr. 3), ktorý rotačným pohybom ťahá ozubený pás a ten potom tento pohyb prenáša na vozík. Motor je umiestnený na ľavej strane konštrukcie. Na jeho hriadeľ je umiestnené ozubené kolo s počtom zubov 40. Modul ozubeného kola je GT-2. Výhoda použitia krokového motora spočíva v jeho rýchlej dynamike a vysokom momente sily. Dynamika motora je dostatočne rýchla nato, aby ju bolo možné zanedbať, takže nie je potrebné matematicky modelovať akčný člen a silové momenty vozíka a kyvadla sú zanedbateľné vzhľadom na moment vytváraný motorom. Z týchto dôvodov je možné proces riadiť priamo prostredníctvom zrýchlenia vozíka.

Krokový motor

Krokový motor je vo svojej podstate elektromechanické zariadenie, ktoré premieňa elektrické pulzy na diskkrétne mechanické pohyby. Hriadeľ krokového motora sa točí robením diskrétnych krokov, ktoré sú výsledkom aplikácie elektrických pulzov v správnych sekvenciách. Sekvencia pulzov priamo ovplyvňuje smer otáčania hriadeľa. Frekvencia týchto pulzov je zasa určujúca pre rýchlosť otáčania a počet pulzov/krokov určuje uhol otočenia hriadeľa [5].

Existuje viacero druhov krokových motorov: pasívne, aktívne, hybridné a lineárne krokové motory. Líšia sa materiálom z ktorého je ich rotor ale princíp jeho pohybu je rovnaký

u všetkých druhov. Rotor tvorí magneticky mäkký materiál alebo permanentný magnet a stator tvoria elektromagnetické cievky, ktoré točia rotorom striedavým vytváraním magnetického poľa.

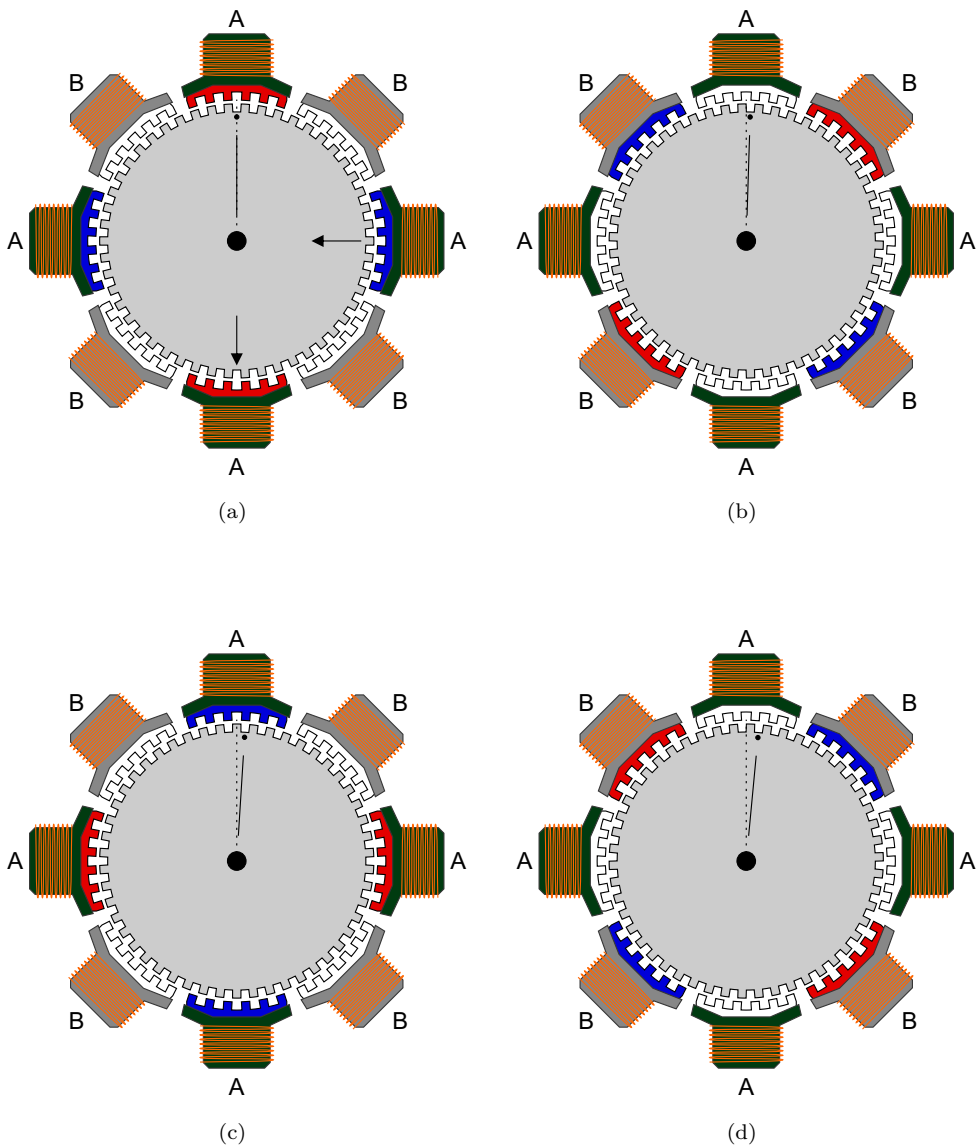


Obr. 3: Použitý krokový motor od firmy Trinamic Motion Control GmbH

Princíp činnosti krokového motora je ilustrovaný na obrázku 4:

- (a) Elektrické napätie je privedené len na dva páry cievok A. Na každý pár cievok je privedené napätie s opačnou polaritou, to znamená, že cievky vyvíjajú sily na rotor s opačnou orientáciou. Šípkami je vyznačený smer sily pôsobiacej cievkou na rotor, t.j. elektromagnet cievky zafarbený na červeno rotor priťahuje a modrý zasa odťláča.
- (b) Ak sa podobným spôsobom privedie napätie na cievky B, tak rotor motora urobí krok, pretože ho elektromagnety cievok B pootočia o 1.8° .
- (c) Opäť sa privedie napätie len na cievky A, ale tentokrát s opačnými polaritami než tomu bolo v bode (a), aby sa rotor otočil o ďalšieho 1.8° a nie naspäť.
- (d) Ten istý prípad ako bod (c) ale v rámci cievok B. Rotor sa opäť otočí o 1.8° .

Nevýhodou použitia krokového motora je nutnosť použiť aj ovládaciú elektroniku - ovládač krokového motora. Ten na základe pulzov prichádzajúcich z riadiacej elektroniky vybudzuje cievky externým napájaním tak, ako to je popísané vyššie. Naopak, jeho výhodou oproti DC motoru je, že má rýchlu dynamiku a vie otočiť svoju hriadeľ o presne stanovený uhol, s presne stanovenou rýchlosťou, ktorú mu jeho dynamika dovolí. Dynamiku DC motora by bolo potrebné matematicky modelovať a zahrnúť do modelu procesu, čím by sa tento model stal zložitejším.

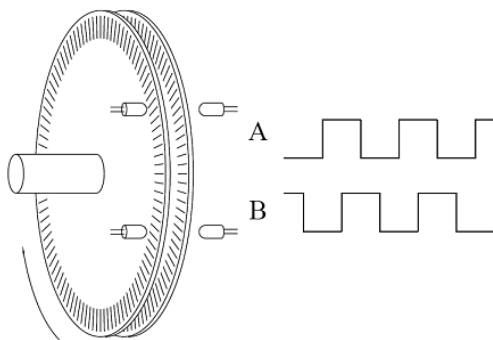


Obr. 4: Grafické znázornenie princípu činnosti krokového motora

2.1.2 Senzory

Rotačný enkóder

Rotačný enkóder je senzor, ktorý je v tomto procese použitý na meranie uhla náklonu kyvadla a polohy vozíka. Princípiálne sa jedná o dvojicu optických závor a priehladný kruh, ktorý má po svojom obvode pruhovanie. Od hustoty tohto pruhovania závisí presnosť, resp. rozlíšenie enkódera. Pri rotácii kruhu dochádza k pohybu pruhovania cez optozávory. Tieto optozávory potom vysielajú binárny signál 1 a 0. Rozostup optozávoru je taký, aby bol signál z druhej optozávory posunutý tak, ako je to znázornené na obrázku 5.



Obr. 5: Rotačný enkodér

Na základe týchto impulzov je možné určiť ktorým smerom a o aký uhol sa kruh otočil.

Rotačný enkóder je v tomto procese použitý aj na meranie polohy vozíka. Je umiestnený priamo nad krokový motor a ich hriadele sú spojené hriadeľovou spojkou, takže motor priamo otáča aj enkóderom. Keď je známy počet pulzov z enkódera a počiatočná poloha vozíka, je možné vypočítať polohu vozíka v metroch alebo iných jednotkách (cm, mm, inch a pod.). Na to je potrebné meraním určiť počet pulzov na meter.

Limitné spínače

Na každej strane konštrukcie sa nachádza jeden pár limitných spínačov. Môže dôjsť k situácii, že vozík svoj pohyb nezastaví ani na samom konci svojej dráhy a tieto spínače slúžia práve nato, aby sa predišlo nárazu vozíka na okraj a prípadnému poškodeniu zariadenia. Signál z prvého spínača je určený pre mikroovládač. Ten ho môže využiť pri návrate vozíka do východzej pozície alebo na zastavenie posielania pulzov do ovládača krokového motora - zastavenie vozíka.

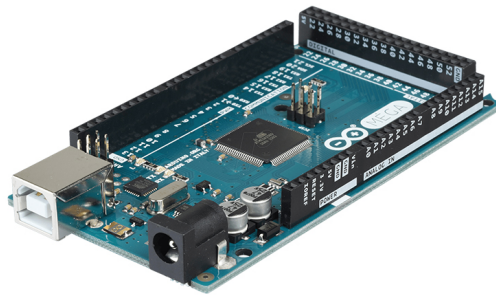
Zopnutím druhého spínača sa privedie napäťový signál na port *En* na ovládači motora a ten fyzicky odpojí cievky motora od elektrického napájania, čo znamená, že príde k

okamžitému zastaveniu motora. Druhý spínač je tam teda ako poistka, keby z nejakého dôvodu zlyhal program v mikroovládači a nezastavil pohyb motora.

2.1.3 Elektronické časti

Mikroovládač

Riadiacou elektronikou inverzného kyvadla je programovateľný mikroovládač s procesorom ATmega2560, obrázok 6. Je to 8 bitový, programovateľný mikroovládač s 54 binárnymi vstupno-výstupnými pinmi a 16 analógovými vstupmi. Tento mikroovládač prijíma signály z rotačných enkodérov a limitných spínačov. Tieto signály ďalej spracuje, teda vyhodnotí uhol otočenia kyvadla, pozíciu vozíka a prípadné stlačenie limitných spínačov. Tieto veličiny sú potom pripravené pre riadiace algoritmy. Tie môžu byť aplikované priamo v tomto mikroovládači alebo v PC s ktorým dokáže mikroovládač komunikovať cez sériovú linku.



Obr. 6: Platforma *Arduino Mega 2560* s procesorom ATmega2560

Uvedený mikroprocesor má k dispozícii kryštálový oscilátor s frekvenciou 16 MHz a 5 periférnych časovačov, ktoré vedia túto frekvenciu využiť či už na funkciu interných prerušení, ktorá je popísaná v podkapitole 4.1.1 alebo na funkciu generovania pulzov s presne zadanou frekvenciou, ktorá je využívaná na ovládanie krokového motora. Okrem interných prerušení disponuje ATmega2560 aj funkciou externých prerušení. Jedná sa o ekvivalent interných prerušení ale s tým rozdielom, že podnet na prerušenie hlavného programu prichádza cez digitálne vstupy, ktorých má ATmega2560 na túto funkciu k dispozícii šesť. Táto funkcia je využitá na správne počítanie pulzov z rotačných enkodérov, z ktorých každý používa dva tieto digitálne vstupy.

Operačné napätie procesora je 5V čo znamená, že medzi digitálnym výstupom a referenčným potenciálom je pri hornej logickej hladine výstupu napätie 5V. Podobne to platí aj pri digitálnych a analógových vstupoch s tým rozdielom, že medzi ne a nulový potenciál je možné priviesť maximálne 5V.

Maximálny elektrický prúd, ktorý je možné odoberať z digitálneho výstupu je 20mA. Tento prúd nie je dostačujúci na vybudenie DC motorov, či iných zariadení, ktoré majú odber vyšší ako 20mA. Tento prúd je však dostačujúci na prácu s rôznymi snímačmi či tranzistormi.

Mikroprocesor ATmega2560 má k dispozícii 248 kB pre takzvanú *flash* pamäť. Jedná sa o pamäť v ktorej je ukladaný kód programu od užívateľa. Veľkosť pamäte *SRAM* (z angl. Static Random Access Memory) je 8 kB a táto pamäť slúži na uchovávanie hodnôt premenných používaných v programe.

Ovládač krokového motora

Riadiaca elektronika sama o sebe nedokáže napájať ani malé motory s nízkym výkonom a momentom sily. Ak však chceme použiť motory, ktoré potrebujú väčší príkon, je nutné použiť medzistupeň v podobe ovládačov krokových motorov (obr. 7). Ovládače krokových motorov majú za úlohu prijať sériu impulzov v logických úrovniach 0 – 5V, ktoré reprezentujú jednotlivé kroky a pomocou externého napájania a vlastnej elektroniky vybudíť vinutia v krokových motoroch tak, aby sa hriadeľ otáčal požadovaným smerom a o zadaný počet krokov (počet impulzov = počet krokov).

Svorky **+24V**, **GND** sú určené pre vstupné napätie. Sem sa zapojí zdroj napájania do 24V.

Svorky **A+**, **A-**, **B+**, **B-** sú určené pre vodiče z krokového motora. Tieto svorky napájajú jednotlivé vinutia v motore.

Svorky **CLK+**, **CLK-** sú vstupom pre impulzy reprezentujúce kroky.

Svorky **CW+**, **CW-** sú vstupom pre binárny signál, ktorého logická hladina určuje smer otáčania krokového motora. Ak je logická hodnota signálu 1, potom sa motor otáča v smere hodinových ručičiek, ak je 0, tak naopak.

Svorky **EN+**, **EN-** slúžia na odpojenie napájacieho obvodu od motora. Na tieto svorky sú pripojené limitné spínače. Ich zopnutím sa privedie napätie na uvedené svorky a ovládač krokového motora okamžite prestane napájať krokový motor.



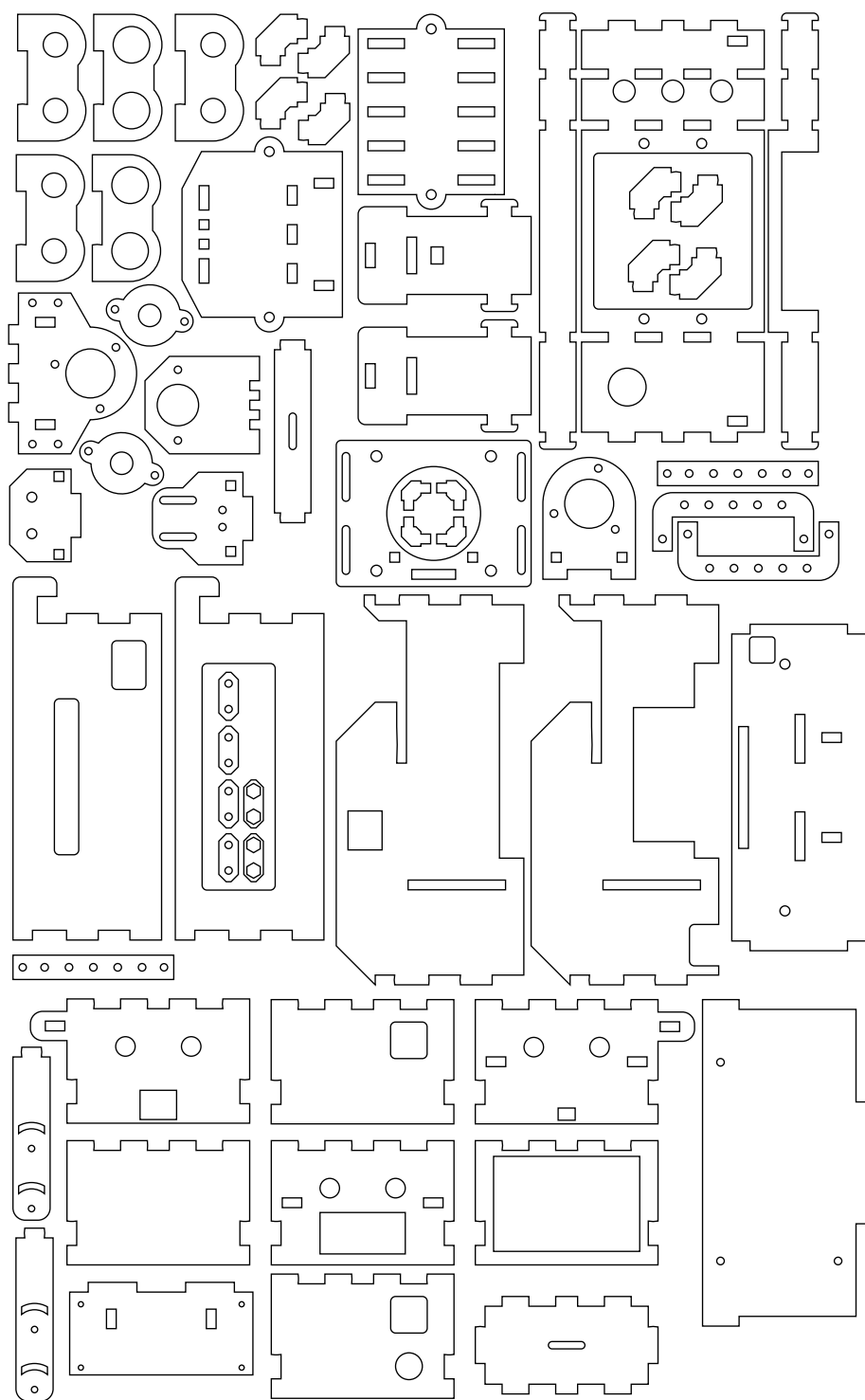
Obr. 7: Ovládač krokového motora *TB6600*

2.2 Návrh konštrukcie

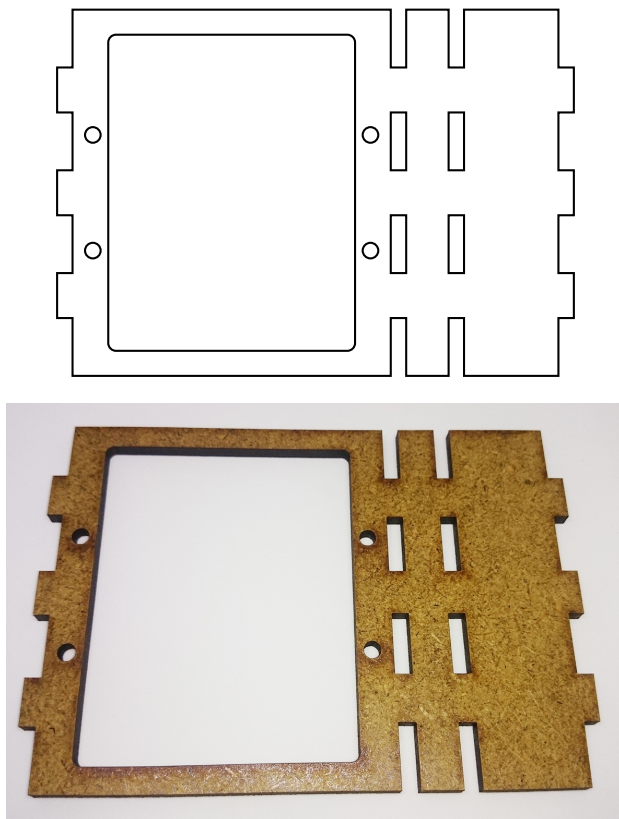
Konštrukcia je časť vyžadujúca najviac kreativity, pretože ponúka vysokú mieru voľnosti. Jediné, čomu sa musí prispôbiť sú elektronické súčiastky. V tomto prípade boli súčasti konštrukcie navrhnuté na mieru a následne vyrezané z MDF (z angl. Medium Density Fiberboard) dosky s hrúbkou 4 mm. Materiál MDF je jeden z najpopulárnejších výrobkov v rámci priemyslu s drevenými kompozitmi. Je vyrábaný lisovaním drevených štiepok, za vysokého tlaku a teploty, za vzniku drevovláknitých dosiek s veľmi kompaktnou štruktúrou [6]. Návrh všetkých súčastí je vo forme vektorových kriviek, ktoré je možné vyrezať do spomenutej dosky na laserovom CNC prístroji. Jednotlivé časti do seba zapadajú ako skladačky, čo uľahčuje a zrýchľuje stavbu zariadenia. To sa delí na dve časti, ktoré sú spojené vodiacimi tyčami, po ktorých sa pohybuje vozík. Na jednej strane je umiestnený krokový motor, ovládač krokového motora a programovateľný mikroovládač, ktorým táto časť konštrukcie slúži aj ako kryt a druhá časť slúži len ako podpora vodiacich tyčí.

Pri návrhu konštrukcie bol použitý program na tvorbu vektorovej grafiky, CorelDRAW X7. Vektorovou grafikou sa rozumie rôzne geometrické útvary ako štvorec, kruh, polkruh, mnohosten a pod, ktoré sú tvorené vektormi, takže pri zväčšovaní alebo zmenšovaní rozmerov obrázku nedochádza ku strate na kvalite, ako to býva u rasterových obrázkov. Na obrázku 8 sa nachádza kompletný vektorový návrh každej súčiastky, ktoré tvoria konštrukciu. Tieto vektorové obrazce je potom možné vyrezať na laserovom CNC prístroji. Laser kopíruje trasu vektorov a vypáli ich do materiálu pod ním. Hrúbka rezu laseru je 0.1 mm. Aby bolo možné navrhnuť nejaký objekt a vyrezať ho, bolo nutné si osvojiť prácu v spomenutom grafickom programe a byť zaškolený na prácu s laserovým CNC prístrojom. Ako vyzerá návrh vo vektorovej grafike a jeho výrez do dosky je možné vidieť na obrázku 9.

Návrh konštrukcie vyžadoval znalosť rozmerov všetkých použitých komponentov ako napríklad krokový motor či mikroovládač, kde boli dôležité vzájomné vzdialenosti dier na skrutky. Iné požiadavky na rozmery konštrukcie neboli. Na jednej strane to bola výhoda, pretože tu bola veľká miera voľnosti pri návrhu, na strane druhej táto voľnosť vyžadovala detailné premyslenie celej konštrukcie. Všetky časti konštrukcie museli korešpondovať s ostatnými, s ktorými boli spojené alebo v nejakom inom funkčnom vzťahu. Návrh jedného objektu teda závisel na ďalších a tie zasa na ňom.



Obr. 8: Kompletný návrh konštrukcie vo vektorovej grafike



Obr. 9: Vektorový návrh a jeho výrez do dosky MDF

2.2.1 Kyvadlo

Vývoj začal od samotného kyvadla (obr. 10). Kyvadlo obsahuje podlhovasté otvory, aby bolo možné k nemu pripevniť rôzne predmety a tým pádom manipulovať s jeho momentom zotrvačnosti. Kyvadlo je dlhé 42cm a široké 15cm.



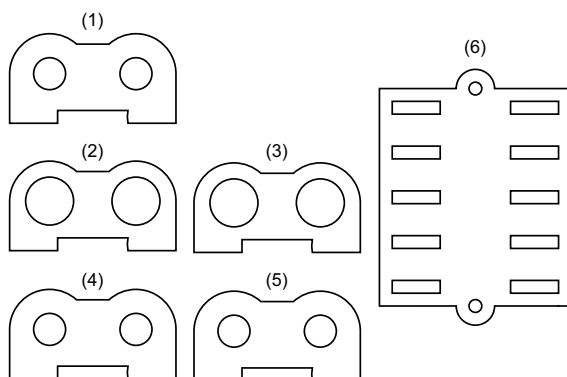
Obr. 10: Vektorový návrh kyvadla

2.2.2 Vozík

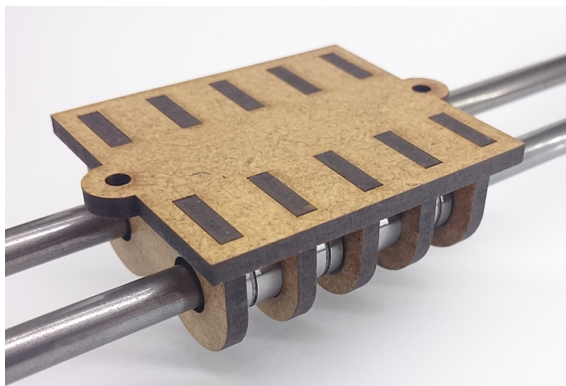
Ďalšou časťou konštrukcie bol návrh vozíka u ktorého bolo dôležité určiť rozteč vodiacich tyčí a poznať rozmery lineárnych ložísk.

Tento vozík (obr. 12) je schopný pohybu po jednej osi, t.j. doprava a doľava, kolmo na os otáčania kyvadla. Telo vozíka obsahuje lineárne ložiská, ktoré znižujú trenie medzi vozíkom a vodiacimi tyčami po ktorých sa pohybuje. Ďalej obsahuje platformu na ktorú je možné upevniť kyvadlovú časť.

Stavba vozíka pozostáva v zlepení jednotlivých súčastí z MDF dosiek, ktoré sú znázornené na obrázku 11. Časti sa skladajú z platformy a piatich útvarov, ktoré držia lineárne ložiská. Útvary s menšími otvormi, t.j. súčasti č. 1, 4, 5 patria na kraje a jeden do stredu. Slúžia nato, aby stabilizovali ložiská proti pohybu do strany. Útvary s väčšími otvormi patria medzi ne a do ich otvorov sa umiestnia lineárne ložiská, takže pred ich nalepením, je potrebné vložiť ložiská do nich a až potom ich vsunúť do platformy - č. 6.



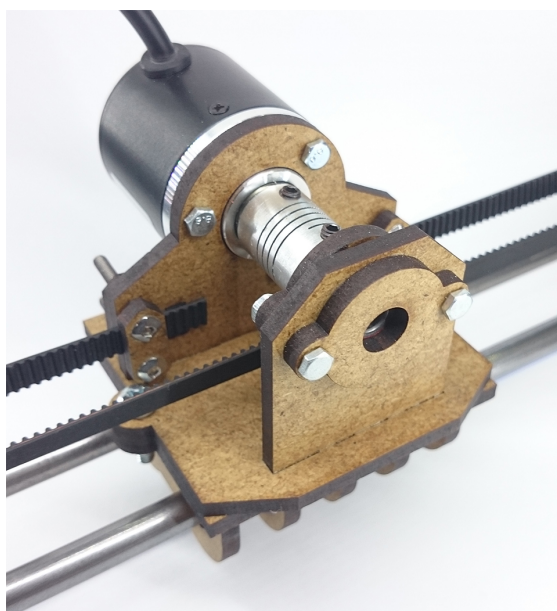
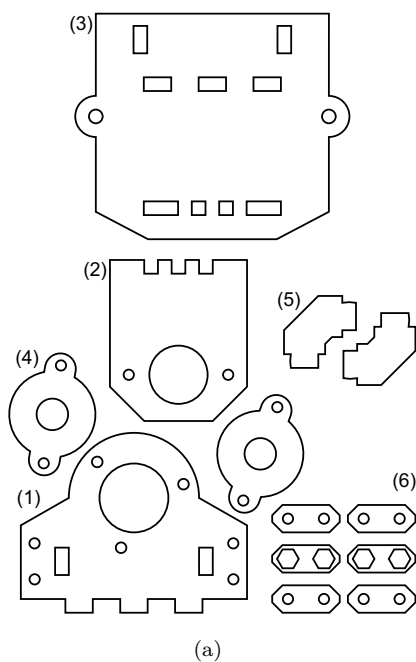
Obr. 11: Súčasti potrebné na stavbu vozíka



Obr. 12: Vozík

2.2.3 Kyvadlová časť

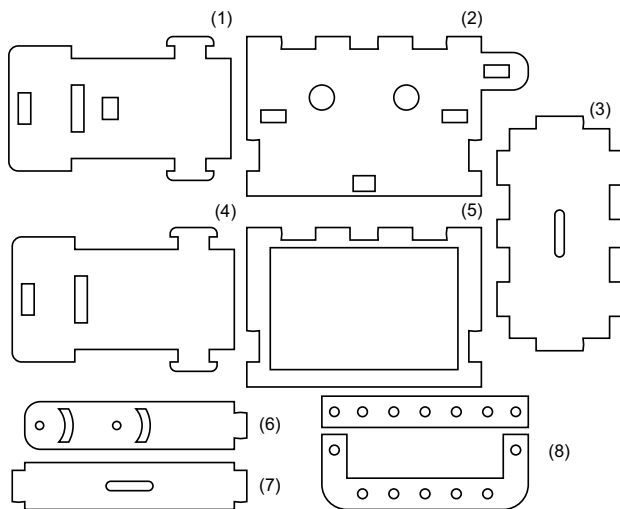
Kyvadlová časť zabezpečuje spojenie rotačného enkódera s kyvadlom. Zároveň je pripojená na vozík a obsahuje aj úchyty na pripevnenie ozubeného pásu. Prvým krokom pri jej stavbe je zlepiť všetky súčasti podľa obrázka 13(b). Tieto súčasti sú zobrazené na obrázku 13(a). Súčasť č. 1 je určená na pripevnenie enkódera a predná časť č. 2 slúži na umiestnenie guľového ložiska (tab. 1 (15)). To sa vloží do stredu a z oboch strán sa skrutkami pripevnia časti č. 4, ktoré toto ložisko stabilizujú. Keďže ložisko je hrubšie ako 4 mm, čo je hrúbka MDF dosky, tak sme najprv jednu stabilizujúcu časť prilepili a až potom sme vložili ložisko a pripevnili druhú stabilizujúcu časť. Pred pripevnením enkódera je potrebné najprv nasadiť na jeho hriadeľ hriadeľovú spojku (tab. 1 (17)). To preto, lebo po jeho pripevnení už nie je dostatok voľného miesta. Celá táto kyvadlová časť sa potom pripevní na vozík skrutkami M4. Bočné úchyty súčasti č. 1 a súčasti č. 6, slúžia na upevnenie ozubeného pásu, ktorý sa pridá až na záver stavby celej konštrukcie.



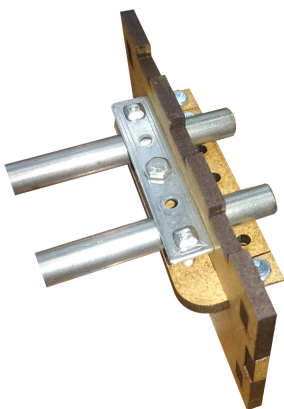
Obr. 13: Grafické znázornenie princípu činnosti krokového motora

2.2.4 Pravá strana

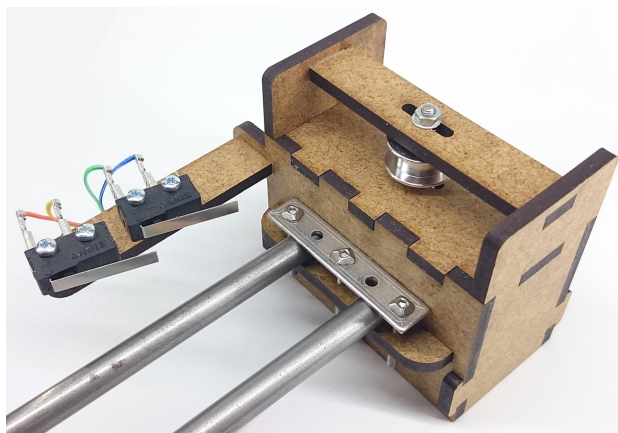
Pravá strana konštrukcie neobsahuje veľa súčastí a jej stavba je pomerne jednoduchá. Skladá sa zo súčastí, ktoré sú na obrázku 14. Keď je známa rozteč tyčí a približná výška v ktorej musia byť, je možné navrhnuť základne držiace tieto tyče po oboch stranách. Na každej základni musí byť systém, ktorý tieto tyče dostatočne silno upevní. Prvým krokom je teda poskladanie systému na upevnenie vodiacich tyčí zo súčastí č. 8 tak, ako je na obrázku 15(a). Tieto súčasti nie je potrebné lepiť a v tejto fáze ešte nie sú potrebné ani vodiace tyče. Nasleduje zlepenie ostatných súčastí a potom je možné pridať ložiskovú kladku a limitné spínače tak, ako je to na obrázku 15(b). Limitné spínače ešte nie je potrebné zapojiť. Súčasť č. 6 na ktorú sa upevňujú limitné spínače je navrhnutá tak, aby sa dal ich náklon vzhľadom na vozík meniť. Súčasti č. 1 a 4 sú identické až na pridaný otvor v prvej z nich. Tento otvor je určený pre vodiče od limitných spínačov, čiže súčasť č. 1 musí byť umiestnená na tej istej strane ako sú aj spomenuté spínače.



Obr. 14: Súčasti potrebné na stavbu pravej časti



(a)



(b)

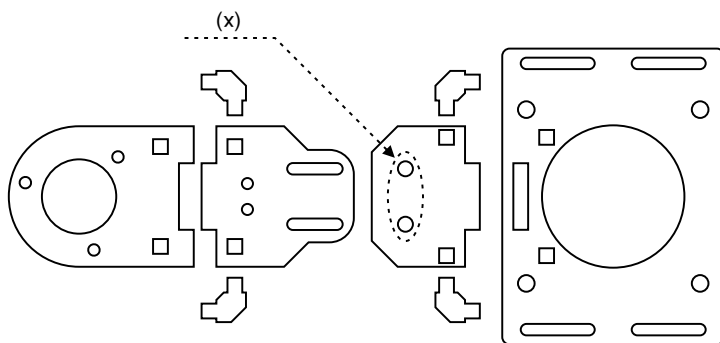
Obr. 15: Pravá časť konštrukcie inverzného kyvadla

2.2.5 Ľavá strana

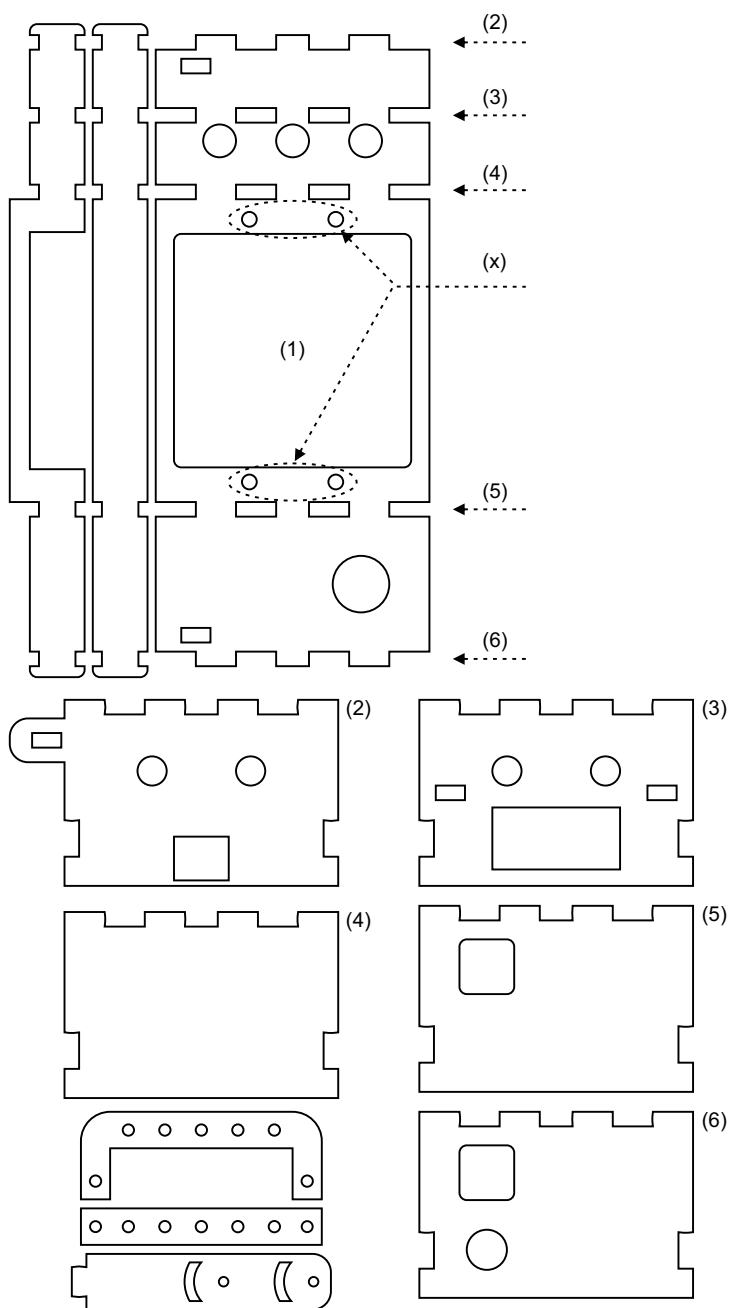
Ľavá časť základne obsahuje akčný člen, čiže túto stranu bolo treba navrhnuť tak, aby sa do nej dal uchytiť krokový motor, ten je uchytený spôsobom, ktorý dovoľuje jeho dodatočné posunutie. Táto časť základne ďalej obsahuje konektor na pripojenie napájacieho adaptéra a napájací spínač. Týmto spínačom sa zapína a vypína napájanie ovládača krokového motora. Stavba tejto časti konštrukcie je najkomplexnejšia a obsahuje najviac súčastí, preto je rozdelená na dve časti.

Prvou časťou je stavba platformy na rotačný enkóder. Potrebne súčasti na jej stavbu sú na obrázku 16 zobrazené tak, aby bolo zrejmé ako súčasti do seba zapadajú. Dvojica kruhových otvorov označených (x) je určená pre zapustené matice M3. Výška platformy na uchytenie rotačného enkódera je nastaviteľná, prípadne sa dá celá oddeliť od celku. Takže prvý sa do panelu upevní motor, následne sa pridá ozubené koleso a potom hriadeľová spojka (tab. 1 (16)). Rotačný enkóder sa pripevní na posuvnú platformu a potom opatrne na motor.

Súčasti potrebné na stavbu druhej časti základne sú zobrazené na obrázku 17 tak, aby bolo jasné poradie stien základne. Štvorica kruhových otvorov označených (x) je opäť určená pre zapustené matice M3. V rámci stavby druhej časti je prvým krokom stavba systému na upevnenie vodiacich tyčí, následne je možné pokračovať stavbou zbytku základne, ktorá je zobrazená na obrázku 18. Limitné spínače sa pripevnia tým istým spôsobom ako to bolo v rámci stavby pravej časti. Ich zapojenie ešte nie je potrebné. Po zložení tejto časti konštrukcie, je možné umiestniť prvú časť obsahujúcu krokový motor do základne.



Obr. 16: Súčasti konštrukcie tvoriace platformu pre rotačný enkóder a uchytenie motora



Obr. 17: Súčasti konštrukcie tvoriace ľavú základňu

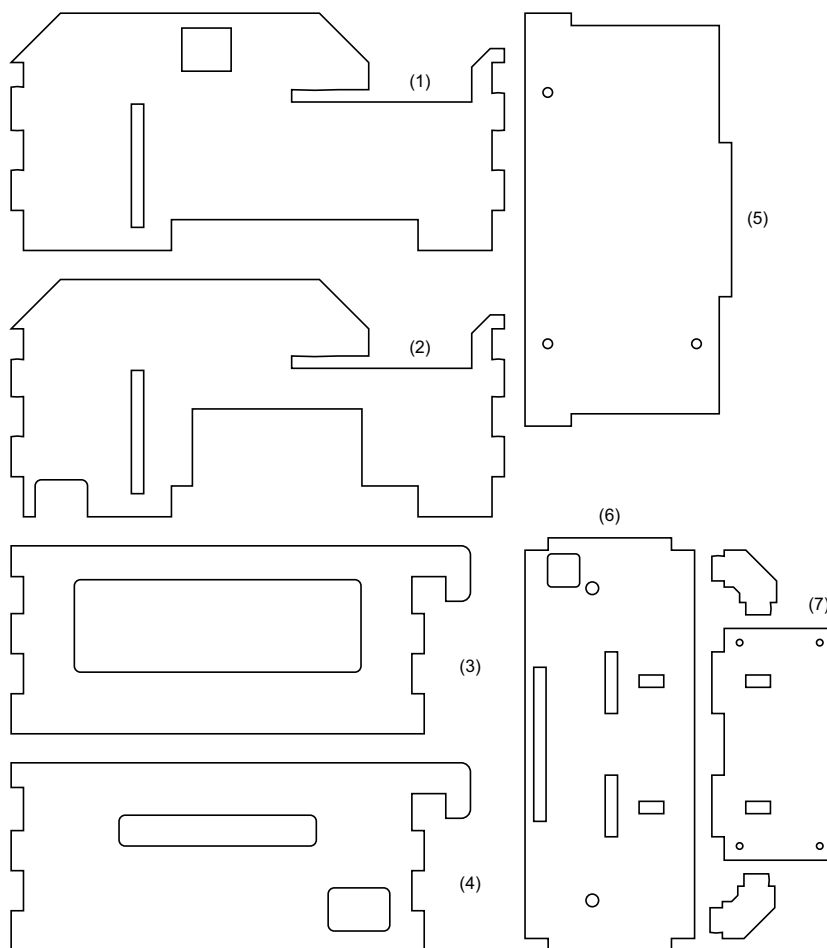


Obr. 18: Ľavá strana konštrukcie s motorom

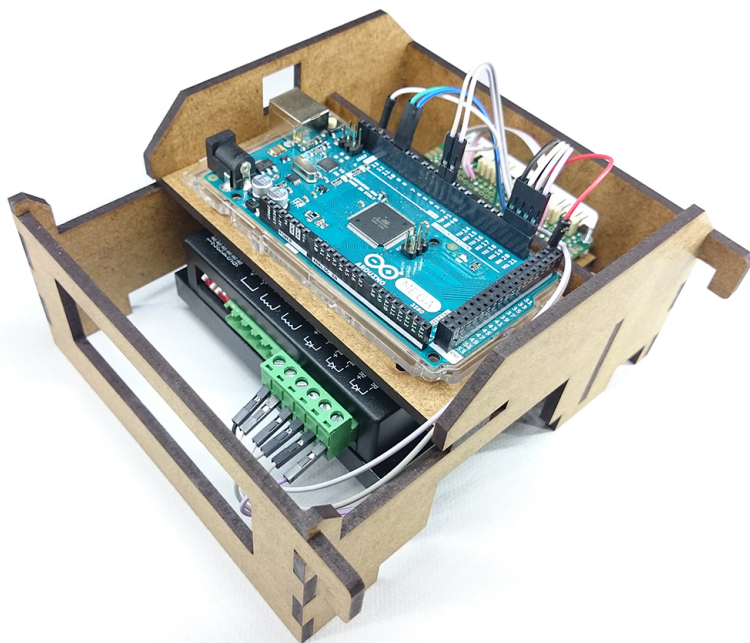
2.2.6 Riadiaca časť

Riadiaca časť slúži ako kryt pre elektronické časti zariadenia. Obsahuje ovládač krokového motora, programovateľný mikroovládač a pájkovateľnú dosku, na ktorej je prídavný elektronický obvod s konektormi pre rýchlejšie a spoľahlivejšie pripojenie limitných spínačov a rotačných enkóderov. Do spodnej časti partí ovládač krokového motora. Miesto pre mikroovládač je nad ním a prídavný obvod patrí do prednej časti, kde sú otvory pre konektory. Táto časť je zobrazená na obrázku 20.

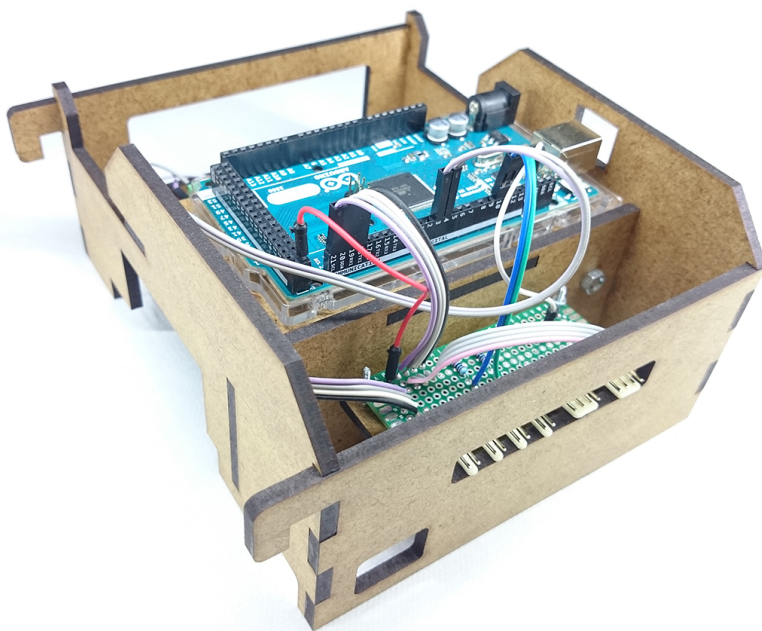
Potrebné súčasti sú zobrazené na obrázku 19. Súčasti č. 1, 2, 3, 4 sú stenami, ktoré do seba zapadajú a háky na častiach č. 3 a 4 slúžia na odnímateľné pripevnenie riadiacej časti k ľavej časti konštrukcie, v ktorej je umiestnený motor. Súčasť č. 5 je plošina, ku ktorej sa skrulkami uchyťí mikroovládač a súčasť č. 7 je zasa plošina pre prídavný elektronický obvod.



Obr. 19: Súčasti konštrukcie tvoriacej riadiacu časť



(a)



(b)

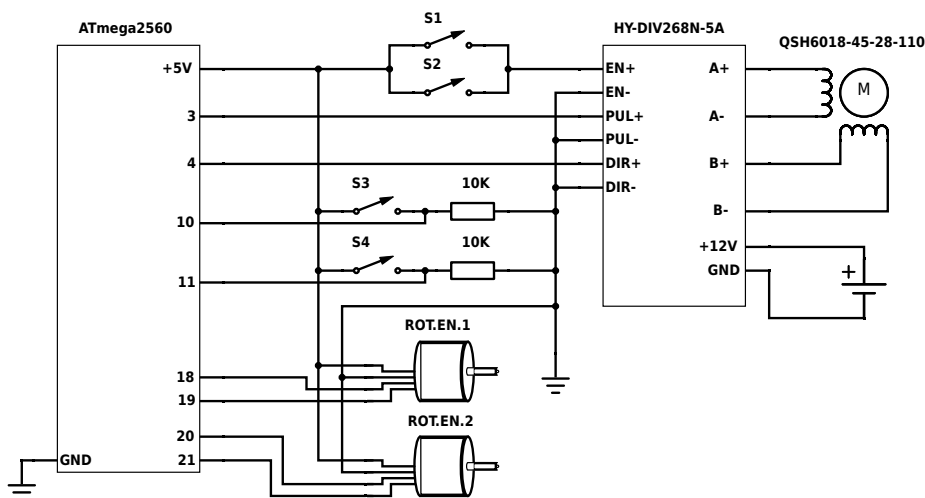
Obr. 20: Pravá časť konštrukcie inverzného kyvadla

2.2.7 Dokončenie konštrukcie

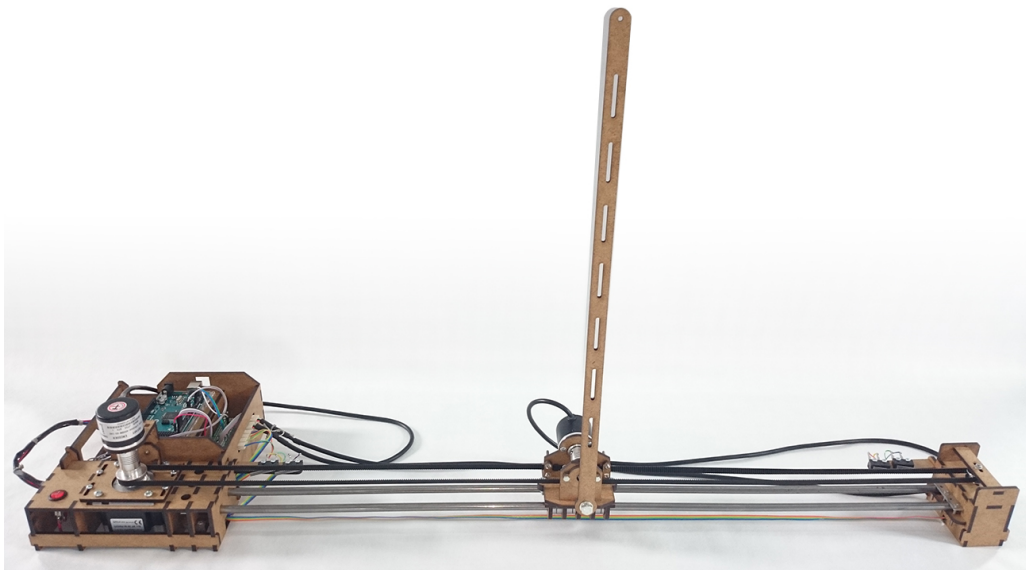
Po vytvorení jednotlivých častí je teraz možné poskladať ich dokopy a dokončiť zariadenie ako celok (obr. 22). Prvým krokom je upevnenie vodiacich tyčí do ľavej strany základne. Vozík sa nasunie na tieto tyče a tie sa potom uchytia na pravú stranu konštrukcie. Od dĺžky vodiacich tyčí závisí aj dĺžka vodičov potrebných na pripojenie limitných spínačov na pravej strane a dĺžka ozubeného pásu. Na vozík sa pripevní kyvadlová časť a ozubený pás sa pripevní na jednej strane kyvadlovej časti, prevlečie sa okolo ozubeného kola na krokovom motore a pretiahne sa cez ložiskovú kladku späť ku kyvadlovej časti, na ktorej sa upevní koniec ozubeného pásu. Ozubený pás je treba jemne napnúť, aby nedochádzalo k preklzu zubov medzi ozubeným kolom a pásom. Ku kyvadlu sa pripevní skrutka M6 a opatrne zasunie do hriadeľovej spojky pripevnenej na rotačnom enkóderi v kyvadlovej časti.

Posledná fáza dokončenia si vyžaduje aspoň základné znalosti základov elektronických obvodov a ich realizácie. V tejto fáze je totiž potrebné vytvoriť prídavný elektronický obvod a prepojiť ho s ostatnými komponentami tak, ako je to znázornené na schéme 21.

2.3 Elektronická schéma zapojenia



Obr. 21: Elektornická schéma zapojenia

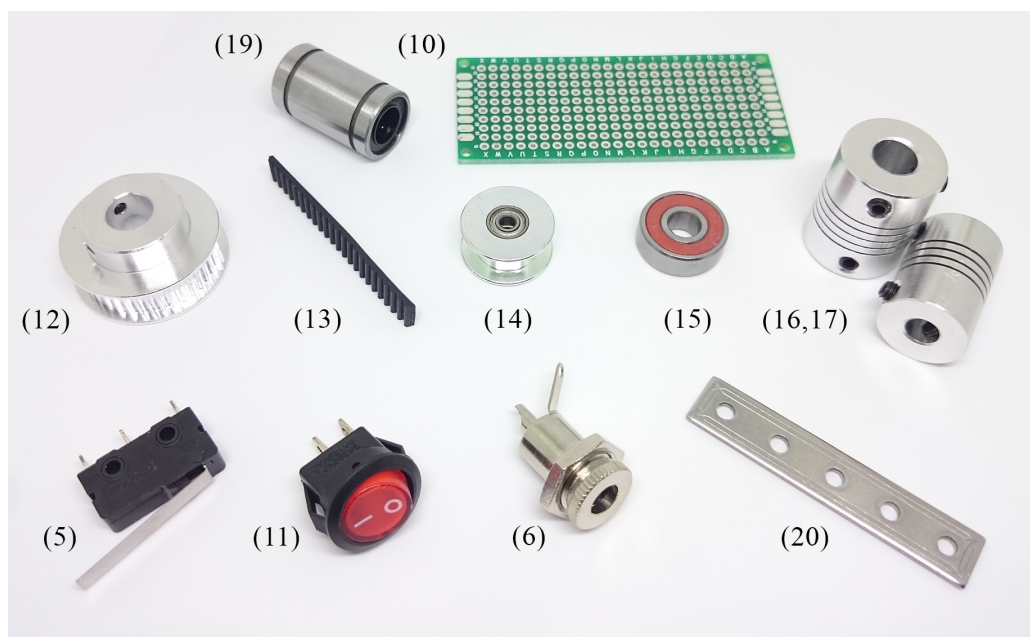


Obr. 22: Kompletne zhotovené zariadenie

2.4 Zoznam použitých komponentov

Tabuľka 1: Používané komponenty

#	koštruktčný prvok	špecifikácia	počet kusov
1	Krokový motor	QSH6018-45-28-110	1
2	Ovládač krokového motora	HY-DIV268N-5A	1
3	Mikroovládač	Arduino MEGA 2560	1
4	Rotačný enkodér	LPD3806-400BM-G5-24C	2
5	Limitné spínače	Mikrospínač 1-pólový, ON-ON	4
6	Napájací konektor	-	1
7	Rezistor	10kOhm	2
8	Konektor 4 pin	-	2
9	Konektor 2 pin	-	4
10	PCB doska na plošný spoj	-	1
11	Napájací spínač	-	1
12	Ozubené kolo 40 zubov	GT2	1
13	Ozubený pás	GT2, 6mm hrubý	2m
14	Ložisková kladka	3mm	1
15	Guličkové ložisko	606 2RS, d:6mm D:17mm B:6mm	1
16	Hriadeľová spojka	8mm - 6mm	1
17	Hriadeľová spojka	6mm - 6mm	1
18	Vodiace tyče, leštené	priemer 8mm, leštená oceľ	2x80cm
19	Lineárne ložiská	vnútorný priemer - 8mm	4
20	Priama spojka	5 otvorov s priemerom 3mm	2
21	Skrutka M2x16	-	4
22	Matica M2	obyčajná	12
23	Skrutka M2.5x12	-	8
24	Matica M2.5	obyčajná	8
25	Skrutka M3x25	-	13
26	Skrutka M3x16	-	12
27	Skrutka M3x10	-	12
28	Matica M3	obyčajná	24
29	Zapustená matica M3	-	6
30	Skrutka M4x16	-	8
31	Matica M4	obyčajná	8
32	Skrutka M6x60	-	1
33	Matica M6	obyčajná	1



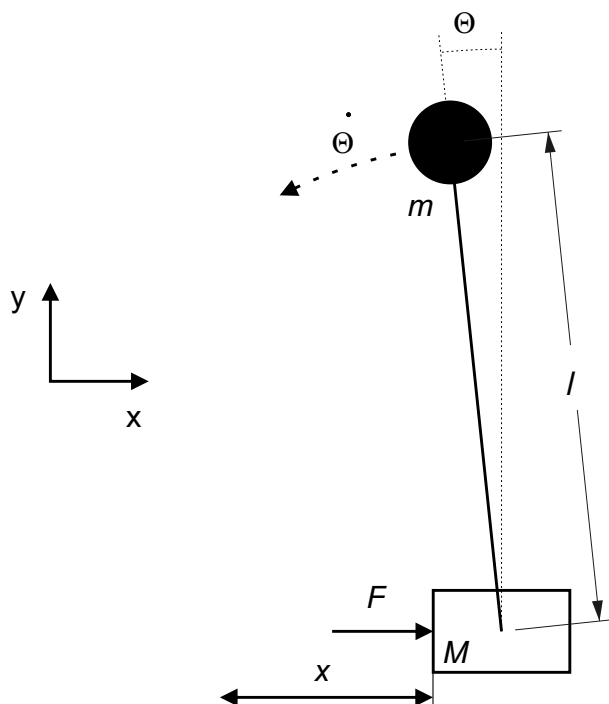
Obr. 23: Niektoré z použitých súčiastok

2.5 Parametre

Tabuľka 2: Parametre inverzného kyvadla

parameter	hodnota	jednotka
Dĺžka kyvadla	0.420	m
Hmotnosť kyvadla	0.026	kg
Rozsah pohybu vozíka	0.000 - 0.640	m
Maximálna rýchlosť vozíka	0.875	ms^{-1}
Napájanie ovládača krokového motora	12	V

3 Matematický model inverzného kyvadla



Obr. 24: Inverzné kyvadlo

Uvažujme model inverzného kyvadla tak, ako je na obrázku 24 kde:

veličina	jednotka	popis
m	[kg]	hmotnosť hmotného bodu na konci kyvadla
M	[kg]	hmotnosť vozíka
l	[m]	dĺžka kyvadla
F	[kgms ⁻²]	vonkajšia sila pôsobiaca na vozík
g	[ms ⁻²]	gravitačné zrýchlenie
\ddot{x}	[ms ⁻²]	zrýchlenie vozíka
Θ	[rad]	uhol otočenia kyvadla
$\dot{\Theta}$	[rads ⁻¹]	uhlová rýchlosť
$\ddot{\Theta}$	[rads ⁻²]	uhlové zrýchlenie

Na odvodenie matematického modelu inverzného kyvadla je použitá Lagrangeova metóda. Všeobecný tvar Lagrangeovej funkcie L je daný nasledovne [7]:

$$L = \text{CelkováKinetickáEnergia}(T) - \text{CelkováPotenciálnaEnergia}(V) \quad (3.1)$$

Potenciálna energia systému je daná ako:

$$V = mgl\cos(\Theta) \quad (3.2)$$

Kinetická energia systému je daná ako:

$$T = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}mv_p^2 \quad (3.3)$$

Rýchlosť kyvadla v_p sa dá rozložiť na rýchlosť kyvadla v smere osi x a v smere osi y .

$$T = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(v_x^2 + v_y^2) \quad (3.4)$$

$$\begin{aligned} v_p &= l\dot{\Theta} \\ v_x &= l\dot{\Theta}\cos(\Theta) - \dot{x} \\ v_y &= l\dot{\Theta}\sin(\Theta) \end{aligned} \quad (3.5)$$

Rýchlosti v_x a v_y dosadíme do rovnice 3.4 s opačným znamienkom, pretože majú záporné orientácie vzhľadom na súradnicový systém.

$$T = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x} - l\dot{\Theta}\cos(\Theta))^2 + \frac{1}{2}m(-l\dot{\Theta}\sin(\Theta))^2 \quad (3.6)$$

$$T = \frac{1}{2}(M + m)\dot{x}^2 + \frac{1}{2}m(-2\dot{x}l\dot{\Theta}\cos(\Theta) + l^2\dot{\Theta}^2(\cos^2(\Theta) + \sin^2(\Theta))) \quad (3.7)$$

$$T = \frac{1}{2}(M + m)\dot{x}^2 + \frac{1}{2}m(-2\dot{x}l\dot{\Theta}\cos(\Theta) + l^2\dot{\Theta}^2) \quad (3.8)$$

Lagrangeovu funkciu systému je teda možné vyjadriť ako:

$$L = \frac{1}{2}(M + m)\dot{x}^2 + \frac{1}{2}m(-2\dot{x}l\dot{\Theta}\cos(\Theta) + l^2\dot{\Theta}^2) - mgl\cos(\Theta) \quad (3.9)$$

$$L = \frac{1}{2}(M + m)\dot{x}^2 - m\dot{x}l\dot{\Theta}\cos(\Theta) + \frac{1}{2}ml^2\dot{\Theta}^2 - mgl\cos(\Theta) \quad (3.10)$$

Matematický model systému dostaneme využitím Eulerovej-Lagrangeovej rovnice:

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_i} \right) - \frac{\delta L}{\delta q_i} = Q_i \quad (3.11)$$

kde q_i sú všeobecné koordináty systému a Q_i všeobecná sila systému [7].

$$\underline{q_1 = x \text{ a } Q_1 = F:}$$

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{x}} \right) - \frac{\delta L}{\delta x} = F \quad (3.12)$$

$$\frac{\delta L}{\delta \dot{q}_1} = \frac{\delta L}{\delta \dot{x}} = (M + m) \dot{x} - ml \dot{\Theta} \cos(\Theta) \quad (3.13)$$

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{x}} \right) = (M + m) \ddot{x} - ml \ddot{\Theta} \cos(\Theta) + ml \dot{\Theta}^2 \sin(\Theta) \quad (3.14)$$

$$\frac{\delta L}{\delta x} = 0 \quad (3.15)$$

$$(M + m) \ddot{x} - ml \ddot{\Theta} \cos(\Theta) + ml \dot{\Theta}^2 \sin(\Theta) = F \quad (3.16)$$

$$\underline{q_2 = \Theta \text{ a } Q_2 = 0:}$$

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{\Theta}} \right) - \frac{\delta L}{\delta \Theta} = 0 \quad (3.17)$$

$$\frac{\delta L}{\delta \dot{q}_2} = \frac{\delta L}{\delta \dot{\Theta}} = -ml \dot{x} \cos(\Theta) + ml^2 \dot{\Theta} \quad (3.18)$$

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{\Theta}} \right) = -ml \ddot{x} \cos(\Theta) + ml \dot{x} \dot{\Theta} \sin(\Theta) + ml^2 \ddot{\Theta} \quad (3.19)$$

$$\frac{\delta L}{\delta \Theta} = ml \dot{x} \dot{\Theta} \sin(\Theta) + gml \dot{\Theta} \sin(\Theta) \quad (3.20)$$

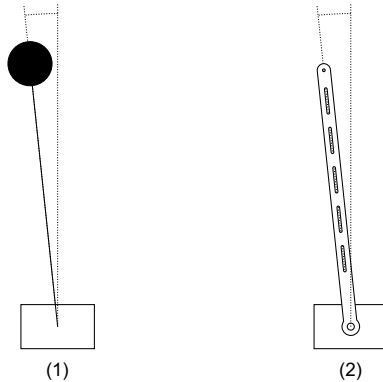
$$-ml \ddot{x} \cos(\Theta) + ml^2 \ddot{\Theta} - gml \dot{\Theta} \sin(\Theta) = 0 \quad (3.21)$$

Odvođený všeobecný matematický model inverzného kyvadla má teda tvar:

$$F(t) = (M + m) \ddot{x}(t) - ml \ddot{\Theta}(t) \cos(\Theta(t)) + ml \dot{\Theta}^2(t) \sin(\Theta(t)) \quad (3.22)$$

$$0 = -\ddot{x}(t) ml \cos(\Theta(t)) + ml^2 \ddot{\Theta}(t) - gml \sin(\Theta(t)) \quad (3.23)$$

S ohľadom na fyzikálne špecifikácie reálneho procesu - použitie krokového motora ako akčného člena, nám dovoľuje zanedbať dynamiku vozíka (rovnica 3.22), pretože je možné priamo riadiť jeho zrýchlenie. Teda použijeme len rovnicu 3.23.



Obr. 25: Dva druhy kyvadiel

Rovnica 3.23 je však odvodená pre kyvadlo, skladajúce sa z bezhmotnostnej tyče, ktorá má na svojom konci hmotný bod, obrázok 25 (1). V rámci nášho reálneho laboratórneho procesu bolo použité kyvadlo s uniformným tvarom, obrázok 25 (2). Z tohoto dôvodu je nutné urobiť v uvedenom matematickom modeli nasledujúce zmeny: Pridať moment zotrvačnosti zodpovedajúci kyvadlu uniformného tvaru:

$$0 = -\ddot{x}(t)ml\cos(\Theta(t)) + (I + ml^2)\ddot{\Theta}(t) - gml\sin(\Theta(t)) \quad (3.24)$$

kde:

$$I = \frac{ml^2}{3}, [\text{kgm}^2] \text{ moment zotrvačnosti kyvadla,}$$

l , [m] polovica dĺžky kyvadla,

po úprave a pridaní faktora trenia, dostaneme dynamický matematický model:

$$\frac{d\dot{\Theta}(t)}{dt} = \frac{3}{4l}\ddot{x}(t)\cos(\Theta(t)) + \frac{3g}{4l}\sin(\Theta(t)) - b\dot{\Theta}(t) \quad (3.25)$$

s nulovými začiatočnými podmienkami.

Vstupom do systému je zrýchlenie vozíka $\ddot{x}(t)$ a výstupom je uhol náklonu kyvadla $\Theta(t)$.

3.1 Linearizácia modelu

Kedže rovnica 3.25 obsahuje nelinearity a radiace algoritmy, ktoré budú v rámci tohoto projektu použité na syntézu regulátorov vyžadujú lineárny model systému, je potrebné ho linearizovať v nejakom ustálenom stave. Za ustálený stav systému považujeme situáciu kedy vozík stojí na mieste, čiže má nulovú rýchlosť aj zrýchlenie a kyvadlo je v jeho rovnovážnej nestabilnej polohe, t.j. vertikálnej polohe s ťažiskom nad osou otáčania:

$$\begin{aligned} \left(\frac{d^2 x(t)}{dt^2} \right)^s &= \ddot{x}^s = 0 \quad \text{ms}^{-2} \\ \Theta^s &= 0 \quad \text{rad} \end{aligned} \quad (3.26)$$

Rovnica 3.25 obsahuje dve nelinearity, ktoré linearizujeme Taylorovým rozvojom do prvého rádu:

$$\ddot{x}(t)\cos(\Theta(t)) \approx \ddot{x}^s\cos(\Theta^s) + \left. \frac{d(\ddot{x}(t)\cos(\Theta(t)))}{d\ddot{x}(t)} \right|_{\ddot{x}^s, \Theta^s} (\ddot{x}(t) - \ddot{x}^s) + \quad (3.27)$$

$$+ \left. \frac{d(\ddot{x}(t)\cos(\Theta(t)))}{d\Theta(t)} \right|_{\ddot{x}^s, \Theta^s} (\Theta(t) - \Theta^s)$$

$$\ddot{x}(t)\cos(\Theta(t)) \approx 0 + 1(\ddot{x}(t) - 0) + 0 = \ddot{x}(t) \quad (3.28)$$

$$\sin(\Theta(t)) \approx \sin(\Theta^s) + \left. \frac{d\sin(\Theta(t))}{d\Theta(t)} \right|_{\Theta^s} (\Theta(t) - \Theta^s) = \Theta(t) \quad (3.29)$$

Linearizáciou sme získali lineárnu diferenciálnu rovnicu opisujúcu dynamiku procesu inverzného kyvadla:

$$\frac{d\dot{\Theta}(t)}{dt} = \frac{3}{4l}\ddot{x}(t) + \frac{3g}{4l}\Theta(t) - b\dot{\Theta}(t) \quad (3.30)$$

Túto rovnicu je teraz možné pretransformovať na stavový opis procesu. Linearizovaný model má dva stavy:

$$\begin{aligned} x_1(t) &= \Theta(t) \\ x_2(t) &= \dot{\Theta}(t) \end{aligned} \quad (3.31)$$

$$A = \begin{pmatrix} 0 & 1 \\ \frac{3g}{4l} & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ \frac{3}{4l} \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad D = \begin{pmatrix} 0 \end{pmatrix} \quad (3.32)$$

Taktiež sa dá vyššie uvedená diferenciálna rovnica pretransformovať na prenosovú funkciu.

Ako bolo spomenuté: zrýchlenie vozíka je vstupom do systému $\ddot{x} = u$ a uhol otočenia kyvadla je výstup $\Theta = y$, potom rovnica 3.30 vyzerá po úprave nasledovne:

$$y''(t) = \frac{3}{4l}u(t) + \frac{3g}{4l}y(t) - by'(t) \quad (3.33)$$

$$\frac{3}{4l}u(t) = y(t)'' + by'(t) - \frac{3g}{4l}y(t) \quad (3.34)$$

Laplaceova transformácia:

$$\frac{3}{4l}U(s) = \left(s^2 + s - \frac{3g}{4l}\right)Y(s) \quad (3.35)$$

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\frac{3}{4l}}{s^2 + s - \frac{3g}{4l}} \quad (3.36)$$

Rovnica 3.36 predstavuje model procesu v tvare prenosovej funkcie a jej hodnoty závisia len od dĺžky kyvadla a koeficientu trenia b . Do premennej l sa dosádza polovica skutočnej dĺžky kyvadla, pretože kyvadlo s uniformným tvarom má ťažisko práve v jeho strede. Celková dĺžka použitého kyvadla v rámci reálneho procesu je 42cm, takže $l = 0.21$. Koeficient trenia sme zvolili $b = 1$. Tento koeficient bol nastavený na základe experimentálneho ladenia modelu tak, aby sa správanie modelu a reálneho procesu zhodovalo.

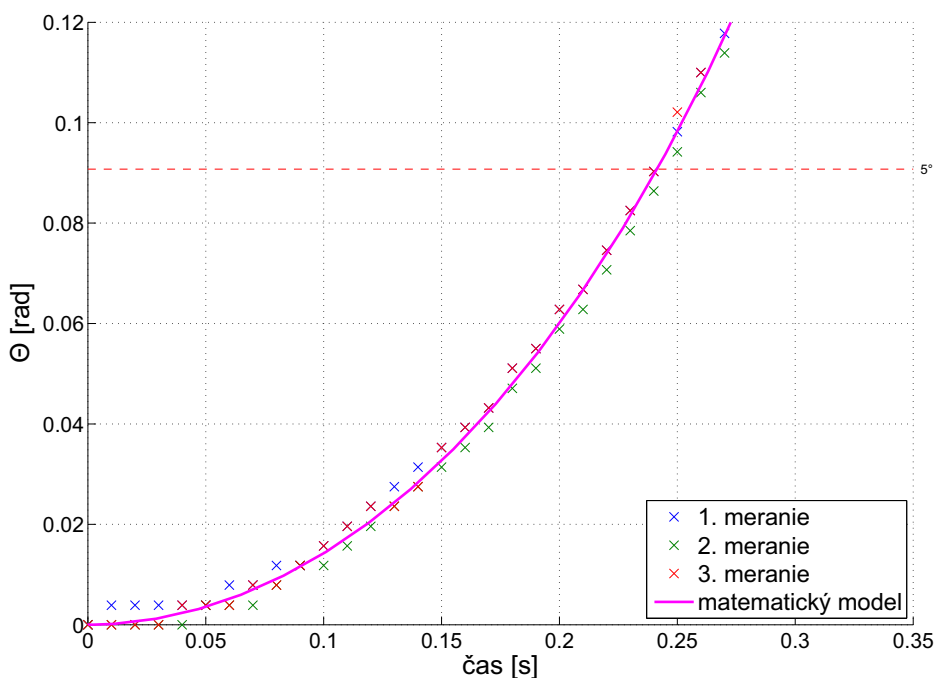
Prenosová funkcia po dosadení konkrétnych parametrov systému:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{3.57}{s^2 + s - 35.02} \quad (3.37)$$

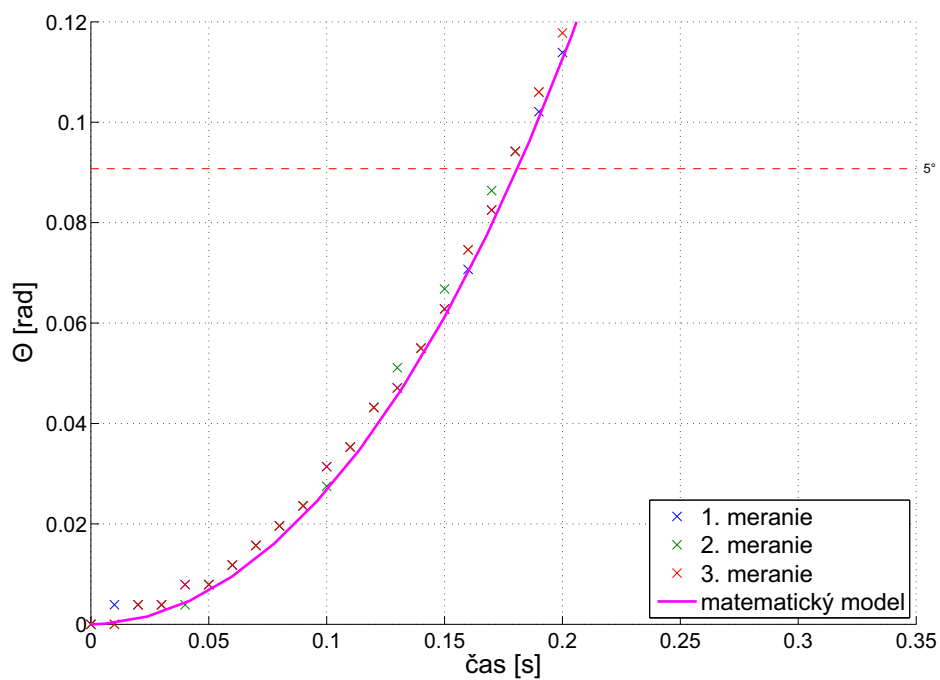
3.2 Verifikácia matematického modelu s reálnym procesom

Keďže inverzné kyvadlo je prirodzene nestabilný proces, nie je možné jeho dynamiku sledovať na prechodovej charakteristike s ustáleným výstupom, pretože tento systém má len jeden ustálený stav, ktorý je možné dosiahnuť na zostrojenom zariadení. Teoreticky ich má nekonečne veľa, pretože je možné stabilizovať kyvadlo v rôznom náklone. Na takýto úkon by však bol potrebný rádovo väčší rozsah pohybu vozíka, než je tomu teraz.

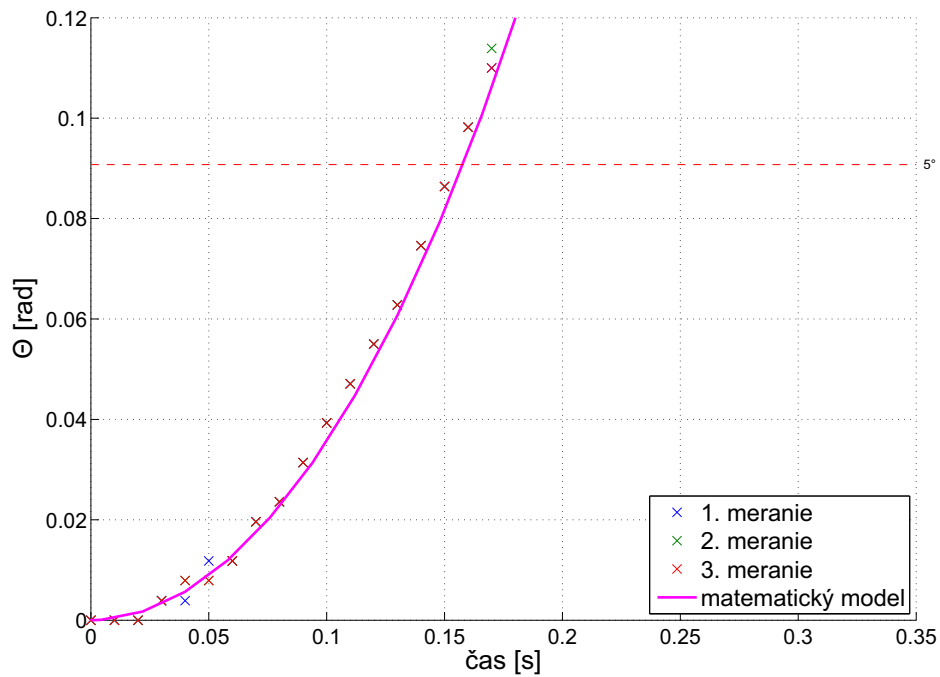
Postup verifikácie v tomto prípade bude teda pomocou porovnania prechodových charakteristík, ktorých výstupy sa neustália. To znamená, že počiatočný stav systému bude pri nulových hodnotách vstupu aj výstupu, potom sa hodnota vstupu skokovo zmení na nejakú nenulovú ustálenú hodnotu, čo zapríčini pád kyvadla. Dynamiku pádu kyvadla je potom možné porovnať s dynamikou pádu matematického modelu. Zvolené boli tri hodnoty zrýchlenia, pri ktorých prebiehalo meranie a pre lepšie porovnanie sa vždy namerali tri sady výstupných hodnôt z každej hodnoty vstupu. Na grafoch je vyznačená hranica 5° , čo predstavuje hranicu do kedy platí $\sin(\Theta) \approx \Theta$.



Obr. 26: Porovnanie dynamiky modelu a reálneho procesu, $\ddot{x} = 0.8\text{ms}^{-2}$



Obr. 27: Porovnanie dynamiky modelu a reálneho procesu, $\ddot{x} = 1.5\text{ms}^{-2}$



Obr. 28: Porovnanie dynamiky modelu a reálneho procesu, $\ddot{x} = 2.0\text{ms}^{-2}$

4 Programové rozhrania

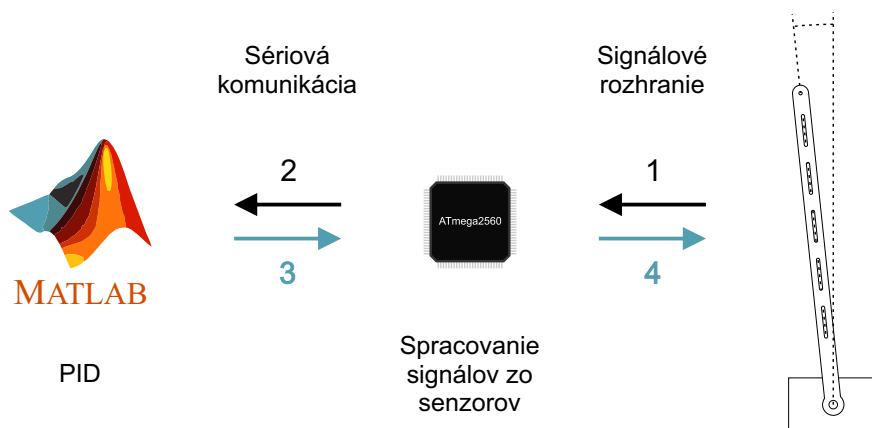
V tejto fáze práce je už hotový reálny laboratórny proces inverzného kyvadla, je určený nelineárny ale aj lineárny matematický model tohto procesu a nato, aby sa mohol začať návrh riadiacich algoritmov a ich implementácia, je potrebné vytvoriť programové rozhrania na prácu s týmto procesom.

Implementovať riadiace algoritmy na proces je možné viacerými spôsobmi. Jednou možnosťou je použiť mikroovládač na zber dát, aplikáciu akčného zásahu a komunikáciu s PC, pričom práve v PC by sa diali výpočty akčných zásahov do procesu. Princíp tejto spolupráce je naznačený na obrázku 29.

1. Dáta z procesu putujú cez signálové rozhranie do mikroprocesora ako elektrické signály. Tam sú tieto signály spracované a hodnoty procesných veličín sú premenené na inžinierske jednotky.
2. Takto spracované dáta sú potom na vyžiadanie posielané do PC pomocou komunikácie cez sériovú linku (USB kábel).

V PC sa nachádza vytvorené užívateľské rozhranie v programe MATLAB. V ňom je implementované riadenie, ktoré vypočíta akčný zásah v inžinierskych jednotkách.

3. Akčný zásah je poslaný opäť cez sériovú linku do mikroprocesora spolu s príkazom žiadajúcim nové hodnoty procesných veličín.
4. Mikroprocesor prekonvertuje inžinierske jednotky akčného zásahu na elektrický signál, ktorý aplikuje na proces cez signálové rozhranie.



Obr. 29: Možnosti riadenia procesu

Výhody:

- Flexibilná a jednoduchá implementácia riadiacich algoritmov priamo v PC.
- Plnohodnotné grafické, či numerické spracovanie nameraných dát v PC.
- Na implementáciu riadenia nie je nutné robiť zásah do programu mikroovládača, či mať nainštalovaný potrebný softvér na jeho programovanie.

Nevýhody:

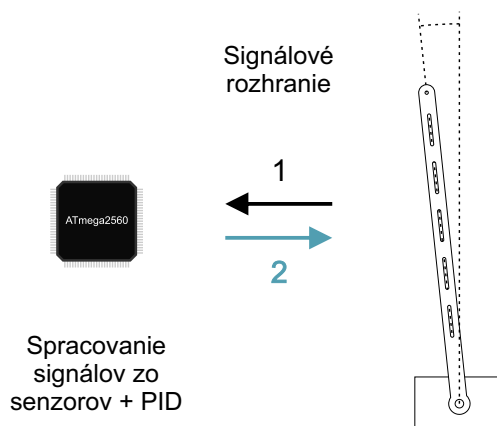
- Nutnosť naprogramovať užívateľské rozhranie v PC a komunikáciu s mikroovládačom.
- Samotná komunikácia medzi PC a zariadením zaberie určitý čas, čiže perióda vzorkovania má svoj dolný limit.

Riadiaci algoritmus je však možné implementovať aj v samotnom mikroovládači avšak za cenu toho, že na zmenu štruktúry alebo parametrov regulátora je potrebné preprogramovať ho. Princíp tejto spolupráce je naznačený na obrázku 30.

1. Dáta z procesu putujú cez signálové rozhranie do mikroprocesora ako elektrické signály.

Tam sú tieto signály spracované a hodnoty procesných veličín sú premenené na inžinierske jednotky. Ďalej je vypočítaný akčný zásah v týchto jednotkách a okamžite prekonvertovaný na elektrický signál. Parametre a štruktúra regulátora je uložená v mikroprocesore.

2. Cez signálové rozhranie je potom tento akčný zásah aplikovaný na proces.



Obr. 30: Možnosti riadenia procesu

Výhody:

- Absencia problémov v komunikácii medzi mikroovládačom a PC, pretože žiadna nie je.
- Nutnosť naprogramovať iba jeden program - odpadá programovanie PC.

Nevýhody:

- Mikroovládač ponúka mnohonásobne menší výpočtový výkon, takže aplikácia zložitejších riadiacich algoritmov by mohla byť problémom, príp. by nebola možná.
- Každá zmena riadiaceho algoritmu si vyžaduje preprogramovanie mikroovládača.
- Veľmi obmedzené možnosti grafického výstupu.
- Je možné vypisovať procesné dáta v reálnom čase, no ich spracovanie je potom pomerne pracné.

4.1 Implementácia riadenia v ATmega2560

Ako bolo spomenuté - riadiace algoritmy sa dajú implementovať priamo do mikroovládača. Každý elektronicky riadený proces pracuje v diskretnom režime, teda s určitou periódou vzorkovania. To znamená, že na zber dát, výpočet regulačnej odchýlky, výpočet akčného zásahu a jeho aplikáciu je vyhradený určitý čas a po jeho uplynutí sa celý tento proces opakuje. Z toho vyplýva, že program v mikroovládači musí byť navrhnutý tak, aby sa periodicky a opakovane vykonával určitý blok kódu obsahujúci riadiace algoritmy. Periodické opakovanie kódu sa zaistí pomocou interných prerušení a ako riadiaci algoritmus použijeme PSD regulátor.

4.1.1 Interné prerušenia

Zvolený mikroovládač, ktorý je riadiacou elektronikou procesu inverzného kyvadla obsahuje kryštálový oscilátor, ktorý kmitá s frekvenciou 16MHz. Ďalej obsahuje sadu periférnych časovačov a počítadiel, ktoré dokážu túto frekvenciu deliť a počítat. Tvoria vnútorné hodiny mikroovládača.

V programe mikroovládača sa nachádzajú dve najdôležitejšie funkcie: *setup* a *loop*. Funkcia *setup* sa vykoná len jedenkrát a to pri štarte mikroovládača. Po vykonaní funkcie *setup* sa vykoná funkcia *loop*, ktorá sa potom cyklicky vykonáva znova a znova. Ak je však potrebné nejaký kód vykonávať v presne zadanej perióde, sú potrebné interné prerušenia. V rámci interných prerušení sa pridá do programu ďalšia funkcia - *ISR* (z ang. Interrupt Service Routine). Táto funkcia má vyššiu prioritu ako funkcia *loop* a vykonáva sa v presne zadanej perióde, ktorú určuje nastavenie vyššie spomenutých časovačov a počítadiel. To znamená, že keď nastane čas vykonania funkcie *ISR*, tak sa funkcia *loop* preruší (z toho vyplýva aj ten názov "Interné prerušenia"), vykoná sa *ISR* a *loop* opäť pokračuje tam kde skončila.

Zavedme si do programu pravdivostnú premennú, nazvime ju *flag*. Aby bolo možné túto premennú používať vo funkcii *loop* aj *ISR*, je potrebné, aby bola typu *volatile*. Funkciu *ISR* využijeme nato, aby tejto premennej priradzovala pravdivostnú hodnotu 1. Vo funkcii *loop* sa nachádza náš riadiaci algoritmus, ktorého vykonanie je podmienené premennou *flag*, čiže sa vykoná jedine vtedy, ak je *flag* rovná 1. Na konci funkcie *loop* je do premennej *flag* priradená pravdivostná hodnota 0. To znamená, že po vykonaní cyklu riadiaceho algoritmu, sa nezačne okamžite vykonávať ďalší cyklus, ale až vtedy, keď *ISR* opäť vráti premennú *flag* na pravdivostnú hodnotu 1. Týmto sa zabezpečí, aby sa riadiaci algoritmus cyklicky vykonával s presne danou periódou.

Nastavenie časovačov interného prerušenia

Vyššie bolo uvedené čo sú to interné prerušenia, ich princíp a aj to ako ich využiť. Teraz si ukážeme ako nastaviť tieto prerušenia priamo v programe tak, aby sme dosiahli požadovanú periodu vzorkovania.

K dispozícii je kryštálový zdroj kmitov s frekvenciou 16 MHz, periférne časovače a počítadla. Základnú frekvenciu je možné deliť číslami 1, 8, 64, 256, 1024. Vďaka počítadlám, je možné rátať kmity tejto frekvencie a porovnávať ich počet s počtom, ktorý potrebujeme dosiahnuť. Ak je zaznamenaná zhoda, tak sa vykoná funkcia *ISR*, počítadlo sa vynuluje a toto sa opakuje cyklicky znova. Napríklad, potrebujeme, aby sa funkcia *ISR* vykonala každú jednu milisekundu. Základnú frekvenciu 16 MHz podelíme 64. Máme teraz k dispozícii kmity s frekvenciou 250 kHz, teda perióda týchto kmitov je $\frac{1}{250000} = 0.000004s$. Každú milisekundu teda počítadlo naráta $0.001/0.000004 = 250$ kmitov, takže keď nastavíme delič na 64 a počet kmitov na $250 - 1 = 249$, pretože sa berie do úvahy až pretečenie počítadla, tak sa pri takomto nastavení funkcia *ISR* vykoná presne každú milisekundu.

$$N = \frac{T_{ISR} 16 \cdot 10^6}{D} - 1 \quad (4.1)$$

kde:

N - počet kmitov, v ang. literatúre označovaný ako *Compare Match Register*

T_{ISR} [s] - žiadaná perióda vzorkovania

D - delič frekvencie (1, 8, 64, 256, 1024), v ang. literatúre označovaný ako *Prescaler*

Konfigurácia interných prerušení sa nachádza vo funkcii *Setup*, kde sa nastaví delič a počet kmitov po ktorom má počítadlo pretiecť a spustiť funkciu *ISR*.

```
void setup()
{
    noInterrupts();
    TCCR5A = 0;
    TCCR5B = 0;
    TCNT5 = 0;
    OCR5A = 249; // N - pocet kmitov (Compare Match Register)
    TCCR5B |= (1 << WGM52);

    /* Nastavenie delicov (Prescalers):
    TCCR5B |= (1 << CS52) | (1 << CS51) | (1 << CS50);

    CS52  CS51  CS50
    0      0      1      clk/1
    0      1      0      clk/8
    0      1      1      clk/64
    1      0      0      clk/256
    1      0      1      clk/1024
    */
    TCCR5B |= (1 << CS51) | (1 << CS50); // Delic = 64

    TIMSK5 |= (1 << OCIE5A);
    interrupts();
}
```

Vyššie uvedená konfigurácia časovača je nastavená na periodu vzorkovania 1 ms, užívateľ si však podľa uvedených pravidiel môže nastaviť ľubovoľnú periódu vzorkovania.

4.1.2 Algoritmus riadenia PSD regulátorom

Každú jednu periódu vzorkovania je potrebné načítať uhol otočenia kyvadla, vypočítať jeho odchýlku od referenčnej polohy, vypočítať akčný zásah do procesu v podobe zrýchlenia vozíka, prepočítať ho na rýchlosť a nakoniec rýchlosť prepočítať na frekvenciu pulzov pre krokový motor. Na výpočet akčného zásahu slúži rovnica 4.2.

$$e_k = w - y_k$$

$$u_k = u_{k-1} + Z_r \left[\left(1 + \frac{\Delta t}{T_i} + \frac{T_d}{\Delta t} \right) e_k + \left(-1 - \frac{2T_d}{\Delta t} \right) e_{k-1} + \frac{T_d}{\Delta t} e_{k-2} \right] \quad (4.2)$$

$$v_k = u_k \Delta t + v_{k-1}$$

$$f = v_k / P$$

$$u_{k-1} = u_k$$

$$v_{k-1} = v_k$$

$$e_{k-2} = e_{k-1}$$

$$e_{k-1} = e_k$$

kde:

$k = 1, 2, \dots, N$ - krok opakovania výpočtu

w - žiadaná poloha kyvadla

y_k - uhol otočenia kyvadla (v dynamickom modeli: Θ)

u_k - akčný zásah do procesu (v dynamickom modeli: \ddot{x})

e_k - regulačná odchýlka

$\Delta t = 1\text{ms}$ - perioda vzorkovania

$P = 0.025025$ - konštanta na prepočet rýchlosti na frekvenciu

f - frekvencia pulzov

Z_r, T_i, T_d - parametre PID regulátora

Aplikácia akčného zásahu znamená, generovať impulzy s určitou frekvenciou a posieľať ich do ovládača krokového motora. Frekvencia týchto pulzov je priamoúmerná rýchlosti otáčaniu hriadeľa krokového motora a tým pádom aj rýchlosti posuvu vozíka. Hodnota konštanty P závisí od počtu krokov krokového motora na otáčku a na počte zubov ozubeného kola. Vyššie uvedená hodnota platí pre 3200 krokov na otáčku a pri použití ozubeného kola s počtom zubov 40.

4.2 Program na komunikáciu s PC

Komunikácia mikroovládača s PC je dôležitá pri aplikácii riadiacich algoritmov na proces za predpokladu, že akčný zásah je rátaný v PC. V rámci budúcej práce budú na tomto procese aplikované rôzne štruktúry a druhy regulátorov. Bolo preto žiaduce vytvoriť také rozhranie, ktoré túto možnosť ponúka a zjednoduší prácu s celým laboratórnym procesom tak, že s ním bude možné pracovať bez znalosti hardvéru. Jednotlivé parciálne úlohy riadenia celého procesu boli rozdelené medzi mikroovládač a PC a medzi nimi prebieha obojsmerná komunikácia.

Mikroovládač je teda použitý iba ako prostriedok na zber dát a aplikáciu akčného zásahu. Ten by potom komunikoval s PC podľa určitého protokolu. Úlohou ATmega2560 je v rámci tohoto deja schopnosť spracovávať pokyny z PC, vykonať ich a posilať späť požadované informácie.

Komunikačný protokol

Aby sa predišlo strate informácií spôsobenej zlým spracovaním tečúcich informácií, zavedieme si v rámci našej komunikácie medzi ATmega2560 a PC určité pravidlá. Konkrétne, pravidlá ako má byť príkaz pre mikroovládač dlhý, z akých segmentov sa skladá a čo jednotlivé segmenty reprezentujú. Všetky použiteľné príkazy spĺňajúci tieto pravidlá sú prehľadne uvedené v tabuľke 3.

Tabuľka 3: Príkazy na prácu s laboratórnym procesom [1]

Príkaz	Bajt 1	Bajty 2-5	Bajt 6	Popis
Inicializácia	DEC 1	HEX 0x00000000	HEX 0x0A	reset interných premenných
Akčný člen ZAP	DEC 2	HEX 0x00000000	HEX 0x0A	povolí pohyb motora
Akčný člen VYP	DEC 3	HEX 0x00000000	HEX 0x0A	zakáže pohyb motora
Návrat domov	DEC 4	HEX 0x00000000	HEX 0x0A	presunúť vozík do východzej polohy
Absolútny pohyb	DEC 5	FLOAT X	HEX 0x0A	posun vozíka o X mm od východzej pozície
Relatívny pohyb	DEC 6	FLOAT X	HEX 0x0A	posun vozíka o X mm od aktuálnej pozície
Pohyb rýchlosťou v	DEC 7	FLOAT V	HEX 0x0A	pohyb vozíka rýchlosťou v
Pohyb so zrýchlením a	DEC 8	FLOAT A	HEX 0x0A	pohyb vozíka so zrýchlením a
Údaje zo senzorov	DEC 9	HEX 0x00000000	HEX 0x0A	žiadna akcia, len žiadosť o údaje zo snímačov

Každý príkaz musí byť dlhý 6 bajtov. Prvý bajt má preddefinovanú celočíselnú hodnotu (integer 0 až 255), odpovedajúcu konkrétnemu príkazu. Bajty 2 až 5 majú preddefinovanú hodnotu 0x00000000 alebo v prípade príkazov 5 až 8 nesú informáciu príkazu v podobe 32-bitového reálneho čísla. Posledný bajt má opäť preddefinovanú hodnotu 0x0A a slúži na uzatvorenie každého príkazu, pretože uvedená hodnota zodpovedá znaku pre nový riadok. Mikroovládač na tieto príkazy odpovedá správou dlhou najmenej tri bajty. Prvý bajt reprezentuje číslo prijatého príkazu a druhý a tretí bajt je ukončením riadku, teda

terminátorom správy. V prípade príkazu číslo 9 je spätná odpoveď mikroovládača tvorená 12 bajtmi: číslo príkazu(1 bajt), uhol náklonu kyvadla (4 bajty), pozícia vozíka (4 bajty), zopnutie limitných spínačov (1 bajt), ukončovací znak(2 bajty).

4.3 Programové rozhranie v PC

Programové rozhranie v PC bolo vytvorené vo výpočtovom programe MATLAB. Toto rozhranie pozostáva z triedy, jej premenných a metód. Vytvorením objektu - inštancie tejto triedy sa automaticky naviaže komunikácia s mikroovládačom po sériovej linke cez zadaný port. Následne je možné používať metódy tejto triedy vo vlastnom skripte. Metóda funguje ako klasická funkcia, ktorá dokáže pracovať s premennými v rámci celého objektu ako by to boli jej vlastné. K dispozícii sú metódy pomocou ktorých je možné v reálnom čase vykresľovať procesné dáta v grafe, odoslať akčný zásah do procesu, meniť alebo resetovať hodnoty z rotačných enkodérov a získavať informácie, či vozík zopol niektorý z limitných spínačov.

Správnym použitím týchto metód je možné riadiť tento proces priamo z vlastného skriptu a zároveň tak sledovať procesné dáta. Nasledujúci príklad uvádza jednu z možností ako pracovať s procesom v prostredí programu MATLAB.

```
% Vytvori sa instancia triedy - objekt s nazvom kyvadlo
% Automaticky sa nadviaze komunikacia s mikroovladacom cez zadany port
kyvadlo = Pendulum('COM8');

kyvadlo.setInit; % Nastavi hodnoty procesnych premennych na 0

% Nastavenie parametrov regulatora
% kyvadlo.setPIDparam(Zr,Ti,Td)
kyvadlo.setPIDparam(45, 1.6, 0.2)

% Parametre sa daju zobrazit pre ich kontrolu
kyvadlo.showPIDparam

% Prikazy na online vypisovanie hodnot naklonu kyvadla a pozicie vozika
kyvadlo.setVerbose('angle', 1);
kyvadlo.setVerbose('position', 1);

% Prikaz na vykreslovanie procesnych dat
kyvadlo.plotData
```


4.4 Zoznam a popis funkcií

Celá trieda vrátane názvov funkcií je napísaná po anglicky. Je totiž predpoklad, že s týmto laboratórnym procesom budú pracovať ďalší študenti medzi ktorými môžu byť aj študenti zo zahraničia, ktorí neovládajú Slovenský jazyk.

setVerbose(type, mode)		Touto funkciou je možné vypnúť alebo zapnúť vypisovanie hodnôt načítaných zo senzorov v príkazovom riadku.
<i>type</i>	'angle'	uhol otočenia kyvadla
	'position'	pozícia vozíka
<i>mode</i>	0	vypisovanie vypnúť (predvolené)
	1	vypisovanie zapnúť
setLengthUnit(unit)		Touto funkciou je možné definovať jednotky s ktorými sa pracuje v rámci vytvoreného objektu.
<i>unit</i>	'm'	metre
	'mm'	milimetre
setAcc(acc)		Príkaz určujúci zrýchlenie, ktorým sa má vozík pohybovať. Kladné číslo reprezentuje zrýchlenie smerom do prava, záporné naopak.
<i>acc</i>	float32	
setVel(velocity)		Príkaz určujúci rýchlosť, ktorou sa má vozík pohybovať. Kladné číslo reprezentuje pohyb smerom do prava, záporné naopak.
<i>velocity</i>	float32	

PID(OnOff)		Príkaz ktorým sa zapína alebo vypína riadenie procesu PID regulátorom.
<i>OnOff</i>	boolean	
setPIDparam(Zr, Ti, Td)		Cez túto funkciu je možné meniť parametre PID regulátora. Ak niektorý argument funkcie chýba, tento parameter nebude zmenený.
<i>Zr, Ti, Td</i>	float32	Parametre PID regulátora
showPIDparam		Zobrazí aktuálne parametre PID regulátora
getProcData		Vráti procesné dáta v maticovom tvare z prvých 800 periód vzorkovania. Súprava matic reprezentujú: čas, akčný zásah v ms^{-2} , rýchlosť vozíka v mms^{-1} , uhol kyvadla v rad a pozícia vozíka v mm.
setInit		Príkaz ktorým sa všetky hodnoty v rámci PID regulátora nastavujú na nulu. Taktiež sa vynuluje pozícia vozíka a uhol kyvadla.
getAngle		Vráti aktuálnu hodnotu uhlu natočenia kyvadla vzhľadom na poslednú nulovú pozíciu.
getPosition		Vráti aktuálnu pozíciu vozíka od poslednej nulovej pozície.
plotData		Príkazom sa zobrazí grafické spracovanie dát výstupu a vstupu do procesu za posledných 800 periód vzorkovania.
waste		Príkaz na vyčistenie vstupného bufferu sériovej komunikácie.
status		Vráti aktuálny stav pripojenia k ATmega2560
close		Ukončí sa periodické žiadanie hodnôt zo senzorov od ATmega2560 a zastaví sa komunikácia s ním.
disconnect		Ukončí sa spojenie s ATmega2560

5 Návrh riadiacich algoritmov

5.1 Návrh PID regulátora pomocou metódy umiestnenia pólov

Syntéza regulátora a jeho implementácia

Syntéza PID regulátora pomocou metódy umiestnenia pólov nám dovoľuje vnútiť systému nami požadované vlastnosti. Konkrétne sa jedná o stabilitu uzavretého regulačného obvodu, kmitavosť a rýchlosť regulačného pochodu. Najpodstatnejšou vlastnosťou, ktorú sledujeme, resp. snažíme sa ju dosiahnuť je práve stabilita, pretože úlohou riadenia je stabilizovať kyvadlo v jeho vertikálnej rovnovážnej polohe.

Linearizovaný matematický model procesu v tvare prenosovej funkcie:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{3.57}{s^2 + s - 35.02} \quad (5.1)$$

Z prenosovej funkcie reprezentujúcej inverzné kyvadlo je možné určiť póly pre tento proces: $s = 5.4392$ a $s = -6.4392$. Z nich je zrejmé, že tento proces je nestabilný, pretože obsahuje kladný, reálny pól a je nekmitavý, pretože neobsahuje imaginárne zložky pólov. Keďže navrhujeme PID regulátor a proces je druhého rádu, póly budú tri a umiestnime ich tak, aby uzavretý regulačný obvod bol rýchlejší ako pôvodný proces a stabilný. Póly musia byť teda záporné a ďalej od nuly než tie pôvodné, napríklad $[-7 \ -6.5 \ -6]$. Póly URO boli zvolené tak, aby URO nebol príliš urýchlenný vzhľadom na dynamiku procesu.

Polynóm obsahujúci uvedené korene:

$$0 = (s + 7)(s + 6.5)(s + 6) \quad (5.2)$$

$$0 = s^3 + 19.5s^2 + 126.5s + 273 \quad (5.3)$$

Pre potreby odvodenia všeobecných vzťahov pre výpočet parametrov regulátora sme si zadefinovali aj tento polynóm vo všeobecnom tvare:

$$0 = s^3 + K_2s^2 + K_3s + K_4 \quad (5.4)$$

Prenosová funkcia pre proces inverzného kyvadla vo všeobecnom tvare:

$$G(s) = \frac{b}{s^2 + s + a} \quad (5.5)$$

Prenosová funkcia PID regulátora s ideálnou štruktúrou:

$$G_R(s) = Z_r \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (5.6)$$

Charakteristická rovnica uzavretého regulačného obvodu:

$$1 + G(s)G_R(s) = 0 \quad (5.7)$$

$$1 + \frac{b}{s^2 + s + a} \left(Z_r + \frac{Z_r}{T_i s} + Z_r T_d s \right) = 0 \quad (5.8)$$

$$(s^2 + s + a) + bZ_r + \frac{bZ_r}{T_i s} + bZ_r T_d s = 0 \quad (5.9)$$

$$s^3 + (1 + bZ_r T_d)s^2 + (a + bZ_r)s + \frac{bZ_r}{T_i} = 0 \quad (5.10)$$

Všeobecné odvodenie vzťahov pre parametre regulátora pomocou metódy porovnania koeficientov:

$$Z_r = \frac{K_3 - a}{b} = 45.2266 \quad (5.11)$$

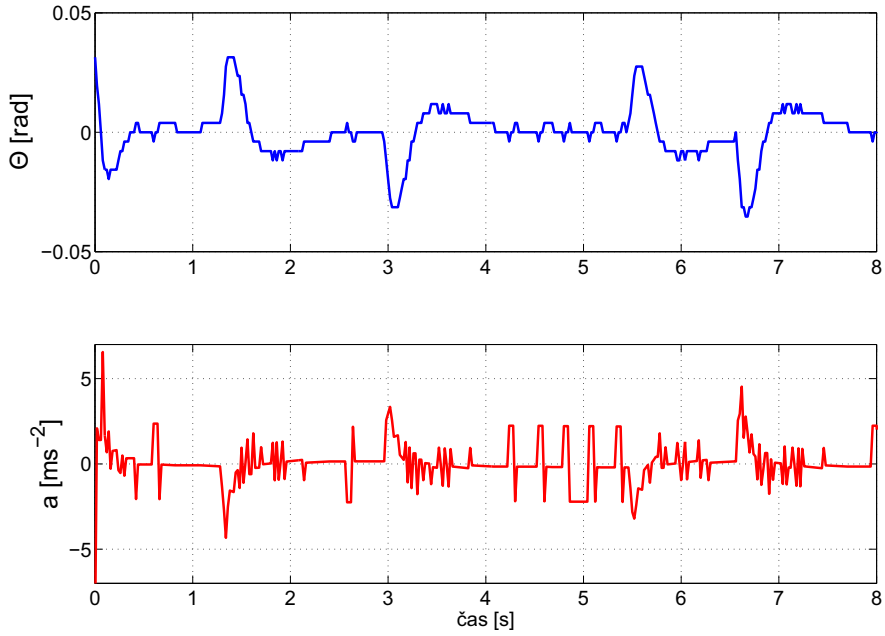
$$T_i = \frac{bZ_r}{K_4} = 0.5917 \quad (5.12)$$

$$T_d = \frac{K_2 - 1}{bZ_r} = 0.1145 \quad (5.13)$$

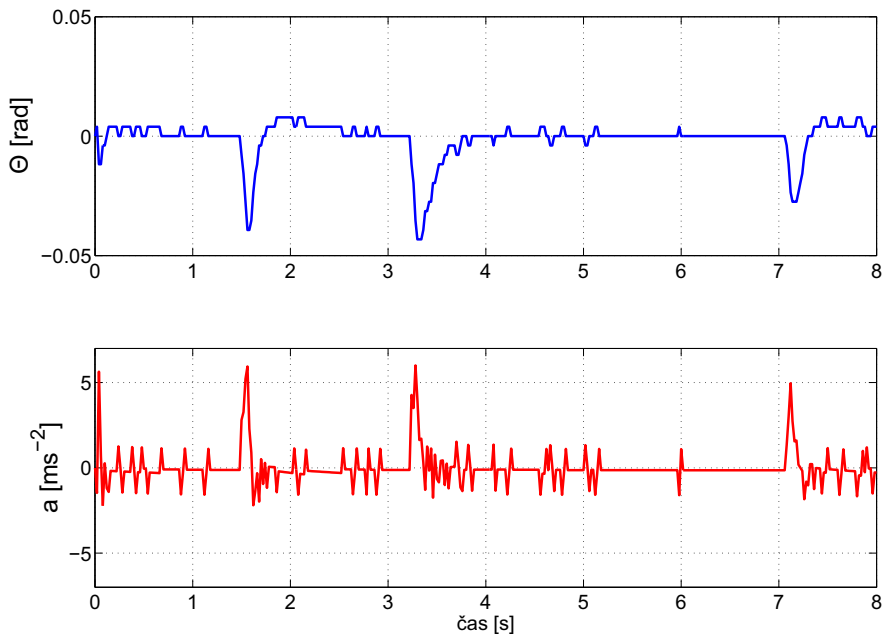
Navrhnutý regulátor bol implementovaný na reálny proces obomi spôsobmi uvedenými v predchádzajúcej kapitole, t.j. v rámci prvej aplikácie bol regulátor implementovaný priamo v ATmega2560 s periódou vzorkovania $20ms$ a druhou aplikáciou regulátora bola jeho implementácia v programe MATLAB. Toto riadenie malo k dispozícii opäť $20ms$ na jednu periódou. Je to totiž najkratšia perióda vzorkovania, ktorú sa podarilo týmto spôsobom implementácie dosiahnuť.

Priebeh riadenia pri úlohe odstraňovania poruchy znázorňuje regulačné priebehy reálneho procesu, ktorý bol vystavený poruchám. Poruchy tvorili jemné nárazy do kyvadla a ako je možné vidieť na obrázkoch 31 a 32, riadenie tieto poruchy odstraňovalo a systém zostal stabilný.

5.1.1 Pribeh riadenia pri úlohe odstraňovania poruchy



Obr. 31: Implementácia riadenia priamo v ATmega2560, $T_s = 20ms$



Obr. 32: Riadenie pomocou programu MATLAB, $T_s = 20ms$

5.2 Návrh robustného PI regulátora

Spätnoväzobný systém je robustný, ak sa zachovajú jeho základné kvalitatívne vlastnosti ako je stabilita, kvalita regulácie a pod., pri pôsobení rôznych porúch vrátane zmeny parametrov objektu. Hlavným cieľom teórie robustného riadenia dynamických systémov je rozpracovať metódy analýzy robustných vlastností reálnych objektov a navrhnúť také regulátory, ktoré zabezpečia robustnosť spätnoväzobného systému. Zvláštnosťou teórie robustného riadenia dynamických systémov je, že modelovanie, analýza vlastností objektu, ako aj syntéza regulátora sa uskutočňuje pomocou neúplného a nepresného matematického opisu riadeného procesu [8].

V prípade, že nie je známa dĺžka kyvadla, ale len približný interval v ktorom sa nachádza, je možné túto dĺžku uvažovať ako neurčitý parameter matematického modelu. Úlohou robustnej syntézy je navrhnúť taký PI regulátor, ktorý garantuje stabilitu systému v celom intervale možnej dĺžky kyvadla. Tento interval sme si zvolili v rozpätí od 30 do 54 cm.

Prenos systému s minimálnou hodnotou neurčitého parametru ($l = 0.15$):

$$G_{\min}(s) = \frac{5.0000}{s^2 + s - 49.0333} \quad (5.14)$$

Prenos systému s maximálnou hodnotou neurčitého parametru ($l = 0.27$):

$$G_{\max}(s) = \frac{2.7778}{s^2 + s - 27.2407} \quad (5.15)$$

Matematický model v tvare prenosovej funkcie s intervalovou neurčitostou:

$$G(s) = \frac{[2.7778; 5.0000]}{s^2 + s + [-49.0333; -27.2407]} \quad (5.16)$$

Charitovone polynómy menovateľa:

$$\begin{aligned} K_1(s) &= -49.0333 + s + s^2 \\ K_2(s) &= -27.2407 + s + s^2 \end{aligned} \quad (5.17)$$

Charitonove polynómy čitateľa:

$$\begin{aligned} K_1(s) &= 2.7778 \\ K_2(s) &= 5.0000 \end{aligned} \quad (5.18)$$

Celkovo sa dajú vytvoriť 4 Charitonove procesy:

$$G_1(s) = \frac{2.7778}{s^2 + s - 49.0333} \quad (5.19)$$

$$G_2(s) = \frac{2.7778}{s^2 + s - 27.2407} \quad (5.20)$$

$$G_3(s) = \frac{5.0000}{s^2 + s - 49.0333} \quad (5.21)$$

$$G_4(s) = \frac{5.0000}{s^2 + s - 27.2407} \quad (5.22)$$

Ďalším krokom je určenie charakteristického polynómu $\Delta(s)$ uzavretého regulačného obvodu s procesom a PI regulátorom v tvare:

Všeobecný tvar prenosu PI regulátora:

$$G_R(s) = k_p + \frac{k_i}{s} \quad (5.23)$$

Všeobecný tvar prenosu vrcholového systému:

$$G_i(s) = \frac{Z_i}{s^2 + s + a_i}, \quad i = 1, 2, 3, 4 \quad (5.24)$$

Na základe požiadavky stability URO: $1 + G_R(s)G_i(s) = 0$, je možné odvodiť všeobecné vzťahy na výpočet závislostí k_i a k_p od ω :

$$\begin{aligned} \Delta(s) &= 1 + G_R(s)G_i(s) = 0 \\ \Delta(s) &= s^2 + s + a_i + Z_i k_p + \frac{Z_i k_i}{s} = 0 \\ \Delta(s) &= s^3 + s^2 + (a_i + Z_i k_p)s + Z_i k_i = 0 \\ \Delta(j\omega) &= (j\omega)^3 + (j\omega)^2 + (a_i + Z_i k_p)(j\omega) + Z_i k_i = 0 \end{aligned} \quad (5.25)$$

Rozloženie na reálnu a imaginárnu časť:

Re:

Im:

$$(j\omega)^2 + Z_i k_i = 0$$

$$(j\omega)^3 + (a + Z_i k_p)(j\omega) = 0$$

$$-\omega^2 + Z_i k_i = 0$$

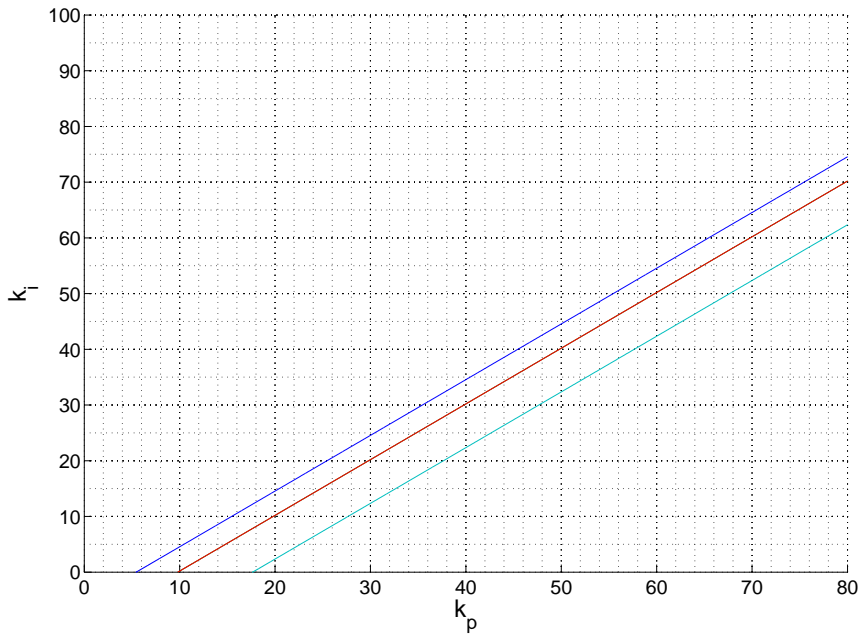
$$-\omega^3 + (a + Z_i k_p)\omega = 0$$

(5.26)

$$k_i = \frac{\omega^2}{Z_i}$$

$$k_p = \frac{\omega^2 - a}{Z_i}$$

Cyklickým výpočtom týchto vzťahov pre rôzne frekvencie $\omega \in (0; 20]$ a pre všetky Charitonove procesy, vzniknú súbory hodnôt k_i a k_p . Po vykreslení závislostí k_i od k_p je možné nájsť oblasť parametrov robustného PI regulátora, ktorá je prienikom stabilných oblastí. To, ktorá oblasť je stabilná, je možné zistiť určením k_p , odčítaním parametra k_i a otestovaním stability URO s týmto regulátorom.



Obr. 33: Závislosti parametrov k_i od k_p

Predpokladajme, že prienikom stabilných oblastí kriviek je pravá strana. Zvolíme si $k_p = 30$ a $k_i = 0.5$. Cyklickým výpočtom koreňov charakteristickej rovnice pre všetky Charitonove procesy je možné posúdiť stabilitu navrhnutého PI regulátora v URO. Podmienkou je, aby všetky korene charakteristických rovníc uzavretého regulačného obvodu mali záporné reálne časti.

$$\Delta_1(s) = s^3 + s^2 + 122.7593s + 2.5000 = 0 \quad (5.27)$$

$$\begin{aligned} s_1 &= -0.4898 + 11.0679i \\ s_2 &= -0.4898 - 11.0679i \\ s_3 &= -0.0204 \end{aligned} \quad (5.28)$$

$$\Delta_2(s) = s^3 + s^2 + 100.9667s + 2.5000 = 0 \quad (5.29)$$

$$\begin{aligned} s_1 &= -0.4876 + 10.0352i \\ s_2 &= -0.4876 - 10.0352i \\ s_3 &= -0.0248 \end{aligned} \quad (5.30)$$

$$\Delta_3(s) = s^3 + s^2 + 56.0926s + 1.3889 = 0 \quad (5.31)$$

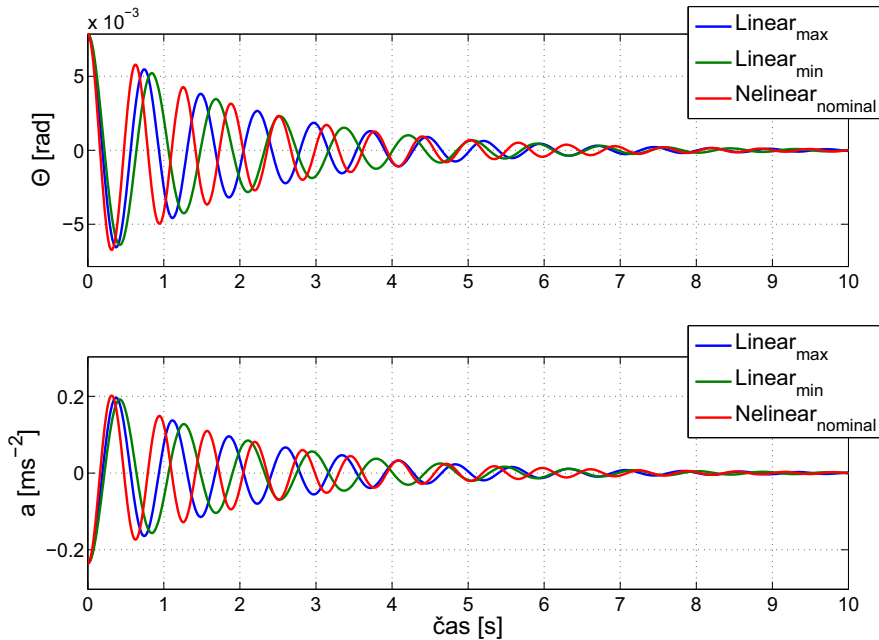
$$\begin{aligned} s_1 &= -0.4876 + 7.4720i \\ s_2 &= -0.4876 - 7.4720i \\ s_3 &= -0.0248 \end{aligned} \quad (5.32)$$

$$\Delta_4(s) = s^3 + s^2 + 34.3001s + 1.3889 = 0 \quad (5.33)$$

$$\begin{aligned} s_1 &= -0.4797 + 5.8336i \\ s_2 &= -0.4797 - 5.8336i \\ s_3 &= -0.0405 \end{aligned} \quad (5.34)$$

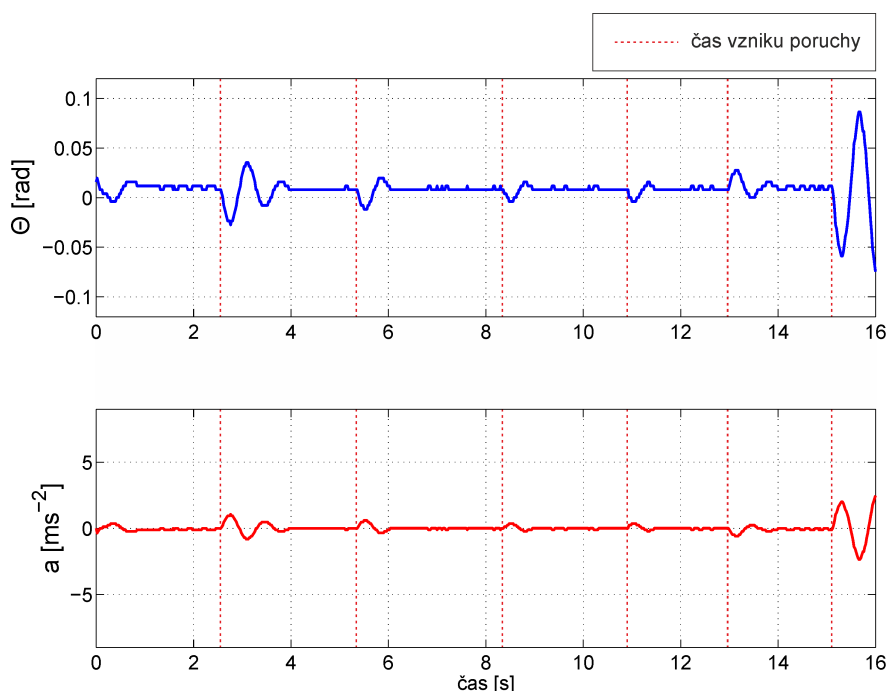
Vyššie uvedená podmienka je splnená, čo znamená, že existuje teoretický predpoklad, že parametre k_i a k_p boli zvolené tak, aby PI regulátor dokázal proces stabilizovať. Otázna zostáva kvalita riadenia, pretože póly všetkých procesov sú blízko nuly, čo naznačuje, že riadený systém sa môže blížiť k hranici stability.

Simulácia znázorňuje priebeh riadenia lineárneho matematického modelu s kyvadlom dlhým 30 a 54 cm a taktiež znázorňuje priebeh riadenia nelineárneho matematického modelu s kyvadlom dlhým 42 cm. Úlohou regulátora bolo stabilizovať proces pri nenulových počiatočných podmienkach, $\Theta_0 = 0.0079$ rad.



Obr. 34: Simulačný priebeh riadenia

Implementácia navrhnutého PI regulátora (obr. 35) potvrdila predpoklad, že systém sa môže pohybovať blízko hranice stability. Regulátor stabilizoval proces, avšak za predpokladu, že porucha - úder do kyvadla, bol pomerne malý, približne polovičný, ako tomu bolo pri riadení procesu s PID regulátorom. Pri výskyte takejto poruchy riadenie zlyhalo a systém sa stal nestabilný. Riešením by mohla byť voľba iných parametrov PI regulátora, avšak nebola nájdená taká kombinácia hodnôt týchto parametrov, ktorá by kvalitu riadenia radikálne zlepšila.

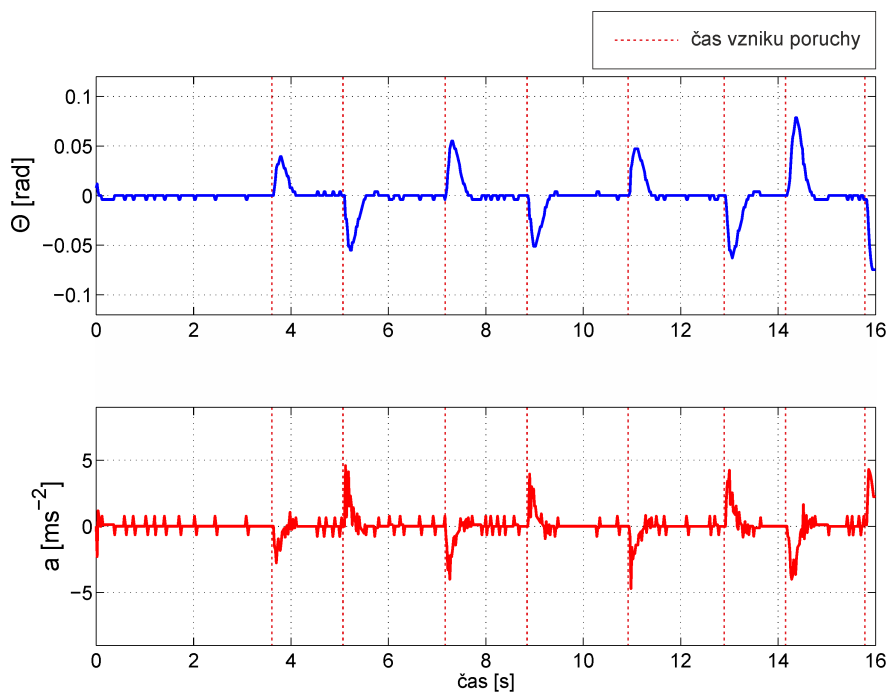


Obr. 35: Priebeh riadenia procesu PI regulátorom, pri úlohe odstraňovania poruchy

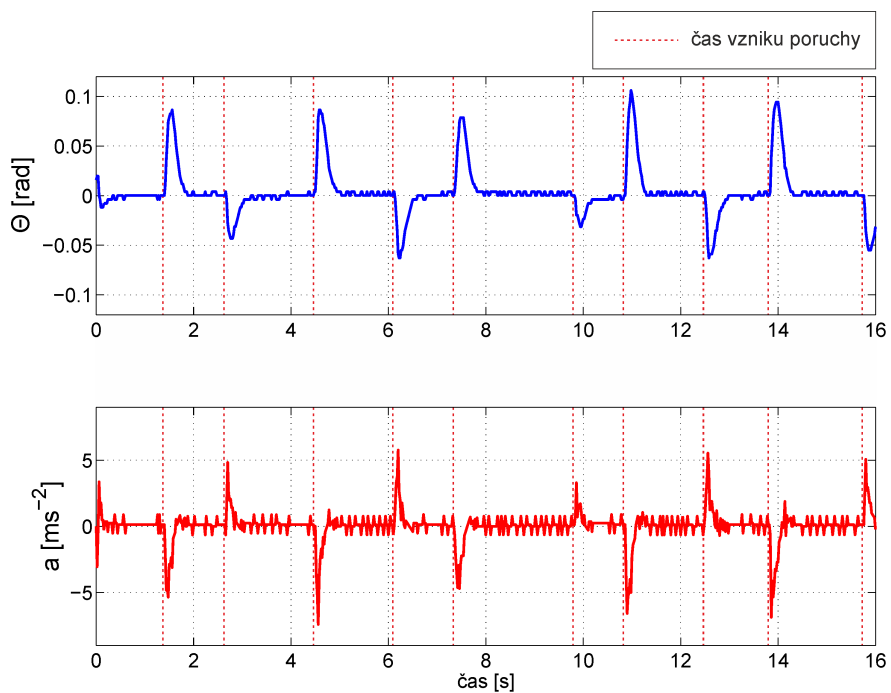
V rámci práce s reálnym procesom často krát dochádza k tomu, že teoreticky navrhnutý regulátor, je potrebné experimentálne doladiť tak, aby sa zlepšila kvalita regulácie. Keďže nebola nájdená taká kombinácia hodnôt parametrov PI regulátora, ktorá by póly uzatvorených regulačných obvodov posunula ďalej od nuly, rozhodli sme sa pridať D - zložku. Hodnoty parametrov P a I zostali zachované a hodnota D parametra bola zvolená na základe návrhu PID regulátora v kapitole 5.1.

$$D = Z_r T_d = 0.3435 \quad (5.35)$$

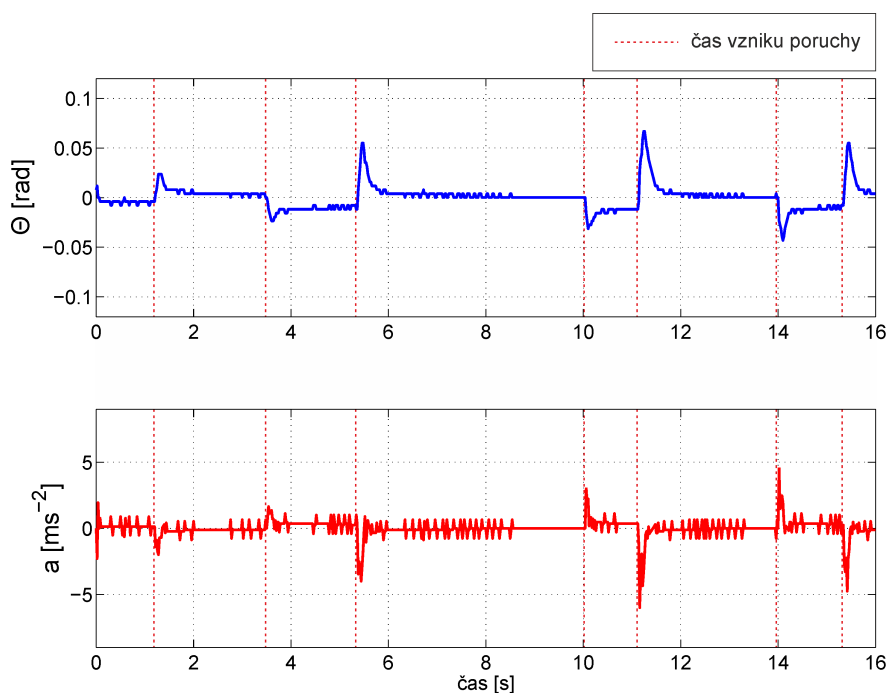
Keďže sme pri návrhu robustného PI regulátora predpokladali s intervalovou neurčitostou parametru dĺžky kyvadla, tak boli vyrobené ďalšie dve kyvadlá s dĺžkami zodpovedajúcimi krajným hodnotám intervalu - 30 cm a 54 cm. Priebeh riadenia reálneho procesu PID regulátorom, pri úlohe odstraňovania poruchy bol teda nameraný pre všetky tri kyvadlá, aby sa zistilo, či regulátor dokáže stabilizovať kyvadlá s rôznou dĺžkou v rámci uvedeného intervalu.



Obr. 36: Pribeh riadenia procesu PID regulátorom. Dĺžka kyvadla: 54cm.



Obr. 37: Pribeh riadenia procesu PID regulátorom. Dĺžka kyvadla: 42cm.



Obr. 38: Priebek riadenia procesu PID regulátorom. Dĺžka kyvadla: 30cm.

Na obrázkoch 36, 37 a 38 je možné vidieť priebeh riadenia reálneho procesu inverzného kyvadla pri úlohe odstraňovania poruchy, ktorých výskyt je vyznačený prerušovanou červenou čiarou. Uzavretý regulačný obvod bol vo všetkých prípadoch stabilný a odstraňoval poruchy prakticky bez prekmitu.

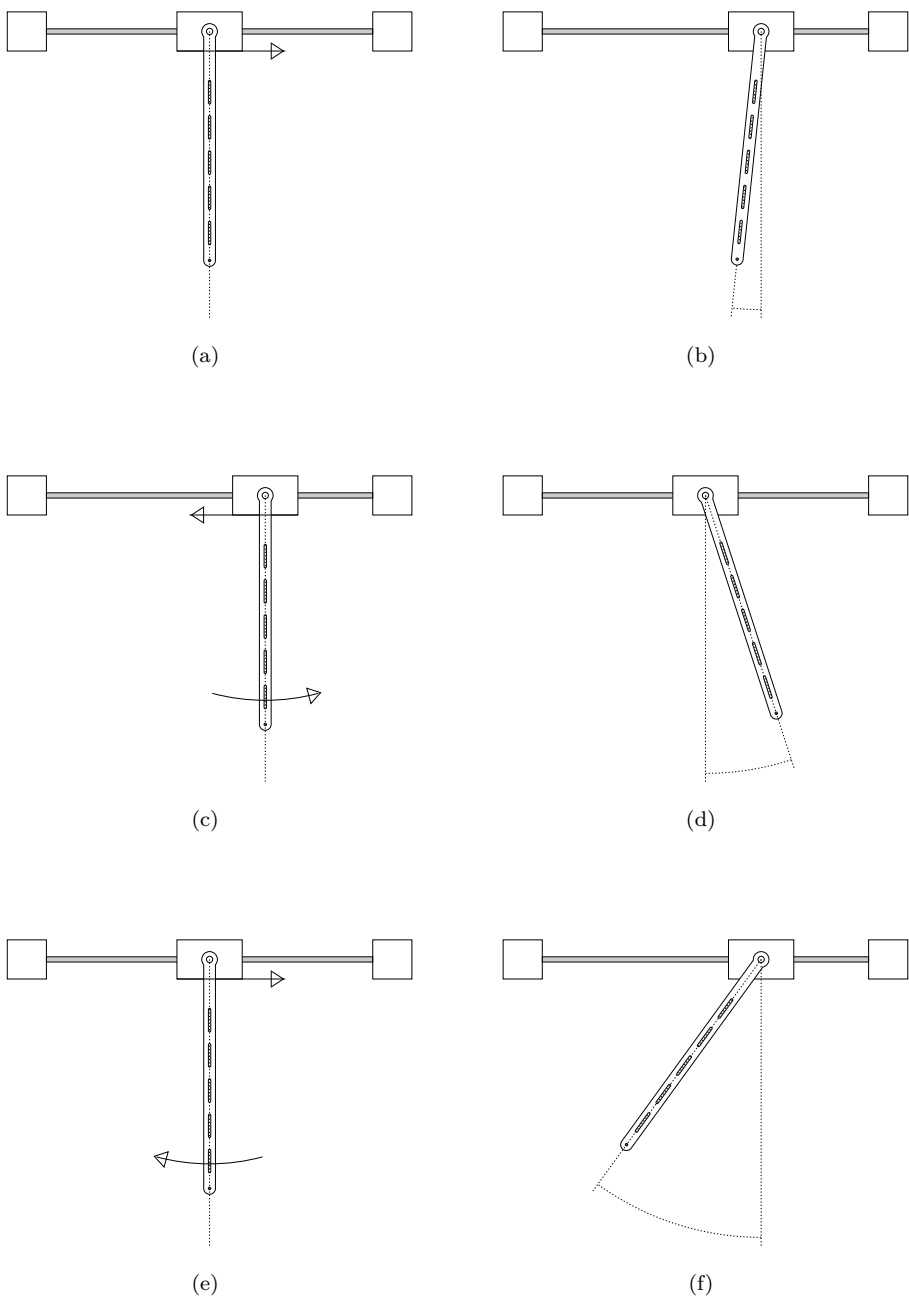
6 Nábeh kyvadla

Anglický názov *Swing up* reprezentuje snahu riadenia o vyšvihnutie kyvadla hore z jeho stabilnej rovnovážnej polohy do nestabilného rovnovážneho bodu. Zjednodušene povedané, ide o vyšvihnutie kyvadla z polohy keď visí dolu, do vertikálnej nestabilnej polohy, v ktorej bolo stabilizované aplikáciou riadení v predchádzajúcej kapitole. Vyšvihnúť kyvadlo hore a stabilizovať ho tam je pomerne náročný problém. Jednou z možností ako to dosiahnuť, je rozdeliť tento proces na dve časti: na riadenie vyšvihnutia a na riadenie stabilizácie v nestabilnom bode. To znamená, že musia byť navrhnuté dva rôzne riadiace algoritmy, ktoré sa aplikujú ako jeden hybridný regulátor, ktorý sa správa podľa zvolených prepínacích kritérií [7].

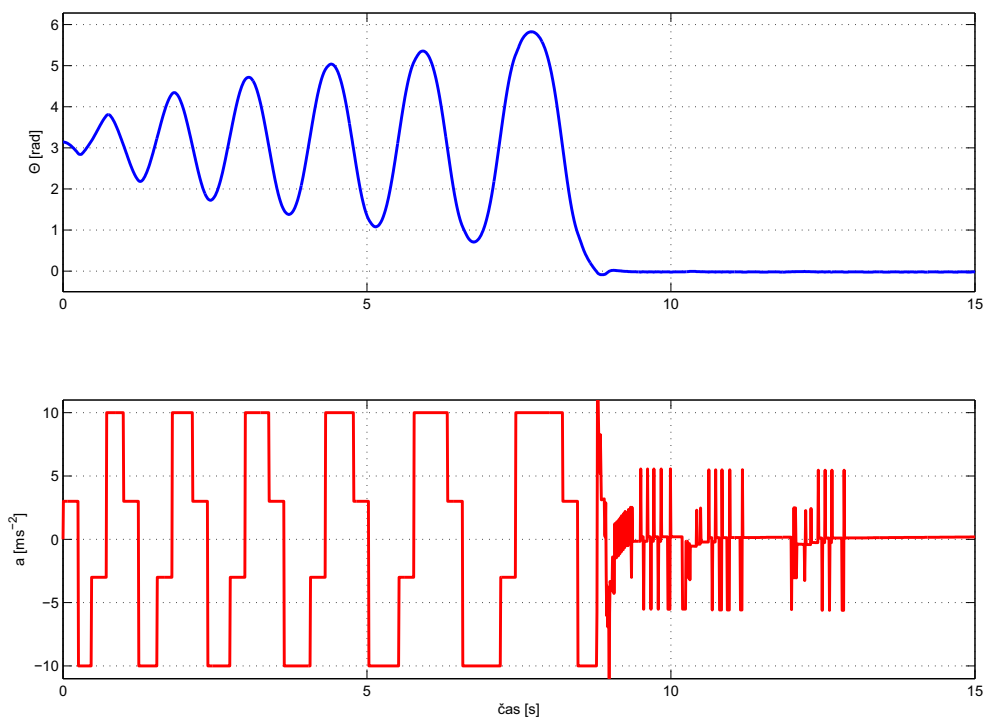
Napríklad pre vyšvihnutie hore môže byť použitý klasický PD regulátor [9]. Taktiež môže byť použitý regulátor založený na celkovej energii kyvadla [10]. Návrh riadenia, ktoré stabilizuje kyvadlo v jeho inverznej pozícii sme si ukázali v predchádzajúcej kapitole.

V rámci tejto práce bol na návrh riadenia vyšvihnutia kyvadla hore, využitý heuristický prístup. Na základe pozorovaní a experimentov s kyvadlom, bol vytvorený jednoduchý algoritmus, ktorý zabezpečí vyšvihnutie - obrázok 39. Akonáhle kyvadlo dosiahne inverznej polohy, prepne sa riadenie a kontrolu preberie PID regulátor, ktorý kyvadlo stabilizuje v jeho nestabilnej rovnovážnej polohe.

- (a) Počiatočný stav. Vozík má nulovú rýchlosť aj zrýchlenie. Kyvadlo je v dolnej vertikálnej pozícii. Prvým krokom je pohyb vozíka smerom do prava so zrýchlením $a = 3\text{ms}^{-2}$, až pokým nedosiahne maximálnu rýchlosť alebo až pokým kyvadlo nedosiahne maximálne vychýlenie spôsobené pohybom vozíka. Po splnení aspoň jednej z týchto podmienok, vozík prudko spomalí so zrýchlením $a = 10\text{ms}^{-2}$.
- (b) Vozík po spomalení stojí. Kyvadlo dosiahne maximálnej výchylky a potom padá naspäť.
- (c) Keď kyvadlo dosiahne začiatočnú polohu, vozík opäť zrýchli ale tentokrát na opačnú stranu než v bode (a), týmto dodá kyvadlu ďalšiu kinetickú energiu.
- (d) Maximálne vychýlenie kyvadla sa zväčšuje. Vozík zastavil po tom ako dosiahol maximálnu rýchlosť.
- (e) Opakuje sa bod (c), t.j. keď kyvadlo dosiahne začiatočnej polohy, vozík zrýchli a kinetická energia kyvadla a jeho maximálne vychýlenie sa opäť zvýši.
- (f) Opakovaním tejto činnosti vozíka je možné dosiahnuť maximálne vychýlenie kyvadla viac ako 180° . Ak kyvadlo prekročí túto hranicu z ktorejkoľvek strany, riadenie sa prepne na PID regulátor a stabilizuje kyvadlo vo vertikálnej hornej pozícii.



Obr. 39: Graficky znázornený algoritmus vyšvihnutia kyvadla



Obr. 40: Priebeh riadenia vyšvihnutia a stabilizácie

Z priebehu riadenia je možné vidieť, že systém potrebuje minimálne 13 pohybov vozíka s daným zrýchlením, aby dokázal vyšvihnúť kyvadlo do hornej pozície. Algoritmus vyšvihnutia nemá dopredu daný počet pohybov vozíka. Riadi sa len pohybom a polohou kyvadla. Ak kyvadlu dodáme kinetickú energiu, napríklad úderom ruky, tak sa kyvadlo vyšvihne rýchlejšie a tým pádom príde aj k prepnutiu riadení skôr. Ak kyvadlo zastavíme rukou, tak vozík bude čakať až kým ho nepustíme.

Záver

Výsledkom projektu je funkčný laboratórny proces inverzného kyvadla. Táto práca bola štylizovaná tak, aby mohla slúžiť ako návod na stavbu identického laboratórneho procesu, pretože bol opísaný kompletný postup práce, zoznam potrebných mechanických, či elektronických komponentov a elektronické schémy zapojenia.

Ďalším krokom v rámci práce s týmto procesom bolo nájdenie všeobecného matematického modelu inverzného kyvadla. Tento všeobecný matematický model bol upravený podľa špecifikácií reálneho procesu a verifikovaný s nameranými dátami, čo potvrdilo správnosť získaného, výsledného matematického modelu. Tento model opisujúci dynamiku inverzného kyvadla bol následne aj linearizovaný, pretože metódy návrhu regulátorov použité v tejto práci sú založené na použití lineárneho modelu. Pre tento proces boli navrhnuté dva riadiace algoritmy - PID regulátor navrhnutý pomocou metódy umiestnenia pólov a robustný PI regulátor, ktorý bol vylepšený a otestovaný na kyvadlách s rôznou dĺžkou. Oba tieto regulátory boli implementované na reálny proces, aby sa overila ich schopnosť stabilizácie procesu.

Poslednou časťou práce bol návrh algoritmu pre vyšvihnutie kyvadla hore a jeho následnú stabilizáciu v inverznej polohe.

Ďalšia práca s týmto laboratórnym procesom sa môže zaoberať vývojom blokov v simulačnom prostredí Simulink, čo uľahčí implementáciu riadiacich algoritmov s rôznou štruktúrou a parametrami, umožní jednoducho identifikovať proces pomocou metódy najmenších štvorcov a pod.

V rámci tejto práce boli navrhnuté dva riadiace algoritmy - PID a PI regulátor, čo znamená, že je tu ešte priestor na návrh zložitejších riadiaich mechanizmov, ako napríklad LQR riadenie (z angl. Linear Quadratic Regulator), či Prediktívne riadenie.

Literatúra

- [1] P. Bakaráč, M. Kalúz, and L. Čirka. Design and development of a low-cost inverted pendulum for control education. *Process Control 2017*, February 2017.
- [2] J. L. A. Lim, Y. Daud, J. X. Xu, and A. Al Mamun. Development of real-time control system for single-wheeled inverted pendulum platform. In *2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM)*, pages 131–136, Sept 2011.
- [3] Masaaki Kumagai and Takaya Ochiai. Development of a robot balancing on a ball. In *2008 International Conference on Control, Automation and Systems*, pages 433–438, Oct 2008.
- [4] M. Hehn and R. D’Andrea. A flying inverted pendulum. In *2011 IEEE International Conference on Robotics and Automation*, pages 763–770, May 2011.
- [5] I. Ionică, M. Modreanu, A. Morega, and C. Boboc. Design and modeling of a hybrid stepper motor. In *2017 10th International Symposium on Advanced Topics in Electrical Engineering (ATEE)*, pages 192–195, March 2017.
- [6] N. Kim, Y. S. Jeong, M. K. Jeong, and T. M. Young. Kernel ridge regression with lagged-dependent variable: Applications to prediction of internal bond strength in a medium density fiberboard process. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1011–1020, Nov 2012.
- [7] A. Rajan, A. A. Kumar, and C. S. Kavitha. Robust control methods for swing-up and stabilization of a rotary inverted pendulum. In *2016 International Conference on Emerging Technological Trends (ICETT)*, pages 1–6, Oct 2016.
- [8] V. Veselý and L. Harsányi. *Robustné riadenie dynamických systémov*.
- [9] Stelian-Emilian Oltean. Swing-up and stabilization of the rotational inverted pendulum using pd and fuzzy-pd controllers. *Procedia Technology*, 12:57 – 64, 2014.
- [10] Navin John and Koteswara Rao. Swing up and stabilization control of a rotary inverted pendulum. In *The International Federation of Automatic Control*, pages 654–659, Dec 2013.