

**KATEDRA ELEKTRICKÝCH
POHONŮ A TRAKCE**

**ČESKÉ VYSOKÉ UČENÍ
TECHNICKÉ V PRAZE**



**FAKULTA ELEKTROTECHNICKÁ
POHYBOVÁ PLOŠINA SE TŘEMI
STUPNI VOLNOSTI**

DIPLOMOVÁ PRÁCE

KVĚTEN 2024

**RICHARD JOSEF
KITRELL**



ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Kittrell** Jméno: **Richard Josef** Osobní číslo: **491945**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra elektrických pohonů a trakce**
Studijní program: **Elektrotechnika, energetika a management**
Specializace: **Elektrické pohony**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Pohybová plošina se třemi stupni volnosti

Název diplomové práce anglicky:

Motion platform with three degrees of freedom

Pokyny pro vypracování:

1. Navrhněte a realizujte pohybovou plošinu o třech stupních volnosti (tripod) po které se bude valit ocelová kulička. Pohyb kuličky snímejte vhodným senzorem.
2. Navrhněte řízení plošiny tak, aby se kulička pohybovala podél předepsané trajektorie. Proveďte simulaci v prostředí Matlab-Simulink.
3. Řízení implementujte ve vhodném jednočipovém procesoru, výsledky porovnejte se simulací.

Seznam doporučené literatury:

1. CRAIG, John J. Introduction to robotics: mechanics and control. Fourth edition. Harlow: Pearson, [2022]. ISBN 978-1-292-16493-9.
2. LYNCH, Kevin M. a PARK, Frank C. Modern robotics: mechanics, planning, and control. Cambridge, United Kingdom: Cambridge University Press, 2017. ISBN 978-1-107-15630-2.
3. KURDILA, Andrew a BEN-TZVI, Pinhas. Dynamics and control of robotic systems. Hoboken, NJ: Wiley, 2020. ISBN 978-1-119-52483-0.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Petr Kočárník, Ph.D. katedra elektrických pohonů a trakce FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **14.02.2024**

Termín odevzdání diplomové práce: **24.05.2024**

Platnost zadání diplomové práce: **21.09.2025**

Ing. Petr Kočárník, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

PODĚKOVÁNÍ

Děkuji své manželce Elišce a rodině, která mě i nadále při mých studiích plně podporovala. Zvláště bych chtěl poděkovat Ing. Petrovi Kočárníkovi, Ph.D., který se mnou na této práci nechal mnoho času, plně mě podporoval v průběhu tvoření a byl vždy připraven s cenou radou, jak pokračovat dále.

Děkuji Bohu.

PROHLÁŠENÍ

Prohlašuji, že jsem předloženou práci vypracoval/a samostatně a že jsem uvedl/a veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 23. května 2024

.....

ABSTRAKT

Tato práce je součástí fyzického modelu pohybové plošiny o třech stupních volnosti. Zmíněná je kategorizace této plošiny v rámci robotických manipulátorů s krátkým motivačním vzhledem k čemu může třístupňová plošina sloužit. Pro implementaci řízení je použita inverzní kinematika, která je teoreticky rozebrána. Přehledově je pohlédnuto na proces modelování této plošiny, kdy využitým softwarem je online platforma *Onshape*. Při sestavování plošiny bylo nutné použít mnoho prvků, plastové díly byly vytisknuty na 3D tiskárně, bylo nutné objednat CNC frézované plexisklo a dále koupit prvky od rezistivního dotykového panelu s kontrolérem po *NEMA 17* krokové motory. Voleným mikroprocesorem je *Arduino Due* v kombinaci s *CNC shieldem* pro drivery *DRV8825* k řízení krokových motorů. K simulaci je využito prostředí *MATLAB & Simulink*, ze kterého je pak implementováno i řízení fyzické soustavy. Realizované řídicí metody jsou pomocí PD, PID a kaskádních regulátorů, je však krátký pohled i na jiné používané metody v dostupné literatuře. Závěrem je zhodnocení výsledků regulace.

Klíčová slova: 3-RRS paralelní manipulátor, pohybová plošina o třech stupních volnosti, kulička na desce, vyvažovač kuličky, třístupňová Stewartova platforma

ABSTRACT

This thesis is a part of a physical model of a motion platform with three degrees of freedom. Mentioned is a categorization of this platform within robotic manipulators with a brief motivating insight into where the 3-DOF motion platform can be used. Inverse kinematics is used to implement the control and is theoretically derived. An overview of the modelling process of this platform is given, where the software used is the online platform *Onshape*. Many elements were required in the assembly of the platform, plastic parts were 3D printed, CNC milled plexiglass had to be ordered and elements ranging from a resistive touchscreen with a controller to *NEMA 17* stepper motors had to be purchased. The microcontroller of choice is an *Arduino Due* combined with a *CNC shield* for the *DRV8825* drivers to control the stepper motors. The *MATLAB & Simulink* environment is used for simulation, from which the control of the physical system is then implemented. The implemented control method is using PD, PID and cascade controllers, but a brief look at other available methods in the literature is also included. Finally, an evaluation of the control results is presented.

Keywords: 3-RRS parallel manipulator, 3-DOF motion platform, ball on plate, ball balancer, 3 - DOF Stewart platform

OBSAH

ÚVOD	1
KAPITOLA 1: PARALELNÍ MANIPULÁTORY	2
1.1 POČET STUPŇŮ VOLNOSTI PARALELNÍHO MECHANISMU	2
1.2 POUŽÍVANÉ KLOUBY V PARALELNÍCH MECHANISMECH	2
1.3 PŘEHLED JINÝCH PARALELNÍCH MECHANISMŮ	3
1.4 APLIKACE PLOŠINY O TŘECH STUPNÍCH VOLNOST	5
KAPITOLA 2: PRVKY PLOŠINY O TŘECH STUPNÍCH VOLNOSTI A VYTVÁŘENÍ MODELU	6
2.1 VÝBĚR CAD SOFTWARE PRO MODELOVÁNÍ 3DOF-P	6
2.2 <i>ONSHAPE</i>	6
2.2.1 Modelování v <i>Onshape</i>	6
2.2.2 Praktické doplňky k modelování a sestavení 3DOFp	7
2.3 POUŽÍVANÉ PRVKY A TESTOVÁNÍ	9
2.3.1 Rezistivní dotykový panel s kontrolérem	9
2.3.2 Arduino CNC shield V3	12
2.3.3 Hybridní krokový motor NEMA17	14
2.3.4 Univerzální plošný spoj s tlačítky a signalizačními diodami	14
2.4 FINÁLNÍ PODOBA MODELU 3DOFP V <i>ONSHAPE</i> , URČENÍ MEZNÍCH NÁKLONŮ	15
2.5 EXPORTOVÁNÍ MODELU Z <i>ONSHAPE</i> DO <i>SIMULINK</i> A DALŠÍ ÚPRAVY	17
KAPITOLA 3: INVERZNÍ KINEMATIKA A KOMPLETACE MODELU	21
3.1 ODVOZENÍ ROVNIC INVERZNÍ KINEMATIKY	21
3.2 OTESTOVÁNÍ INVERZNÍ KINEMATIKY V PROSTŘEDÍ <i>SIMULINK</i>	27
3.3 KONTAKT KULIČKA REZISTIVNÍ DESKA	30
3.4 REGULAČNÍ SCHÉMA	31
KAPITOLA 4: REGULAČNÍ METODY	32
4.1 TESTOVACÍ TRAJEKTORIE	32
4.2 PD REGULACE	34
4.3 PID REGULACE	39
4.4 KASKÁDNÍ REGULACE	41
4.5 STAVOVÝ POPIS MODELU 3DOFP	44
4.6 REGULACE LQR	46
4.7 REGULACE SMC	47
4.8 REGULACE FLC	48
4.9 REGULACE MPC	49
4.10 STROJOVÉ UČENÍ	50
KAPITOLA 5: FYZICKÁ IMPLEMENTACE ŘÍZENÍ 3DOFP	51
5.1 POPIS MODELU PRO FYZICKOU SESTAVU 3DOFP	51
5.2 PD REGULÁTOR PRO FYZICKOU SESTAVU 3DOFP	53
5.3 PID REGULÁTOR PRO FYZICKOU SESTAVU 3DOFP	55
5.4 KASKÁDNÍ REGULÁTOR PRO FYZICKOU SESTAVU 3DOFP	57
ZÁVĚR	59
LITERATURA	60

ÚVOD

Hlavní motivací této práce je popsat proces vytvoření a řízení plošiny o třech stupních volnosti. Tento projekt demonstruje multidisciplinární činnost od modelování, testování modelu v prostředí *MATLAB & Simulink* po fyzickou realizaci a implementaci řídicí metody k balancování ocelové kuličky na ploše.

Kapitola 1 začíná motivací plošiny a za jakými účely může být použita. Zmíněný je krátký přehled paralelních manipulátorů v praxi, jejíž součástí je zkoumaná plošina o třech stupních volnosti. Zároveň je odvozeno proč daný mechanismus vlastně sestává z těchto tří stupňů volnosti a je pohlédnuto na typy kloubů, které mohou být využity. V krátkosti jsou zmíněny příbuzné mechanismy, které mohou využívat jiných kloubních struktur.

V kapitole 2 je pohled na samotné parametry modelované plošiny s přiblížením potřebných prvků, proč byly voleny a jejich základní funkčnost s případnými testovacími kódy v *Arduino IDE*.

Problematika plošiny o třech stupních volnosti obsahuje v dostupné literatuře široké matematické pozadí, kde je zkoumána inverzní a dopředná kinematika. O inverzní kinematice je pojednáno v kapitole 3, protože pak slouží nejen k prozkoumání pracovní oblasti manipulátoru, ale také k vytvoření spojitosti mezi žádanými úhly motorů a výslednými stavy (náklony, posuny) desky. Za tímto účelem byl vytvořen zjednodušený model v programu *Onshape*, který je vhodný k testování různých kinematických stavů a dává vizualizaci pro odvozené matematické rovnice.

Fyzická sestava je jen hmotou, dokud není správně aplikována řídicí metoda, jejíž účel má být polohovat kuličku na desce do žádané polohy. Řídicími metodami se zabývá kapitola 4, kde je poukázán rozdíl mezi PD, PID a kaskádní regulací a jsou představeny jiné metody používané pro problematiku polohování kuličky na plošině.

Pro porovnání chování simulované plošiny od fyzické sestavy, jsou v kapitole 5 představeny výsledky demonstračních průběhů se zhodnocením možných nedostatků regulátorů fyzické sestavy.

V závěru je pohled na možné využití vytvořené plošiny, která může sloužit jako demonstrační prostředek základů PID regulace a jejích variant, a také může být využita k ukázání funkčního principu kaskádní regulace s možným navázáním na tuto práci aplikováním složitějších regulačních metod.

KAPITOLA 1: PARALELNÍ MANIPULÁTORY

Plošina o třech stupních volnosti se řadí do kategorie paralelních manipulátorů. Jedná se o mechanismus uzavřené kinematické smyčky, kdy výsledný pohybovaný člen (pro mechanismus o třech stupních volnosti se jedná o platformu, tedy horní desku) je propojen skrze nezávislé kinematické řetězce (tři pohyblivé nohy poháněné motory) k podstavě mechanismu (spodní deska) [7].

1.1 Počet stupňů volnosti paralelního mechanismu

Pod pojmem stupeň volnosti se v robotice myslí, jaký je minimální počet proměnných, kterými lze popsat polohu mechanismu. V systému dvou dimenzí lze uvažovat obdélník. K popisu jeho umístění vůči pevnému souřadnému systému stačí tři proměnné, dvě pro translaci (v x a y), třetí pro rotaci, souřadnice x' a y' lze získat rotací souřadného systému x a y při znalosti úhlu otočení ξ

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \xi & -\sin \xi \\ \sin \xi & \cos \xi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{R}(\xi) \begin{pmatrix} x \\ y \end{pmatrix}. \quad (1-1)$$

Pro desku v prostoru už nemusí být vizualizace stupňů volnosti snadně představitelná, kdy k popisu polohy se počet proměnných zdvojnásobí na šest, tři pro translaci (v x , y , z) a tři pro rotaci (ψ , ϕ , θ v rovnicích (3-1), (3-2) a (3-3)). Pro přiřazení mechanismu do kategorie plně paralelního manipulátoru je dále podmínka, že počet stupňů volnosti výsledného pohybovaného členu odpovídá počtu kinematických řetězců. Vytvořená plošina o třech stupních volnosti se tedy řadí do této kategorie, protože u platformy jsou uvažovány dva úhly ϕ , θ pro rotaci a svislý pohyb v ose z , tedy tři stupně volnosti pohybu a počet kinematických řetězců je také tři [3].

K výpočtu počtu stupňů volnosti pro plně paralelní mechanismy se stejnými podpůrnými řetězci (nohami) lze využít Grüblerovu rovnici:

$$m = 6 \cdot (l - n - 1) + \sum_{i=1}^n d_i, \quad (1-2)$$

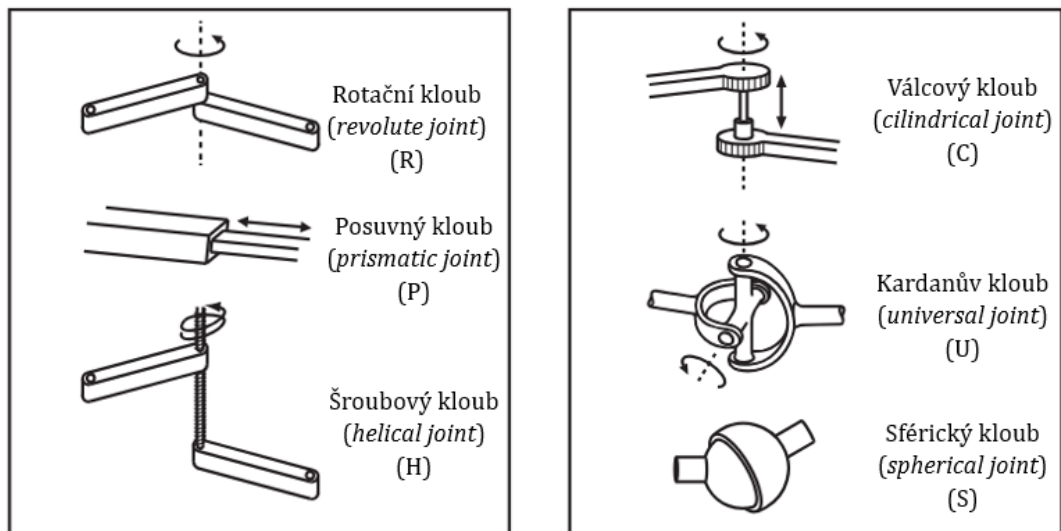
reprezentovaná pro šest stupňů volnosti v prostoru (pro planární se 6 nahradí 3) [3], [7].

1.2 Používané klouby v paralelních mechanismech

K vzájemným propojením v kinematickém řetězci je využíváno mnoho typů kloubů, kdy mezi nejčastějšími mohou být zařazeny R (rotační), P (posuvný), H (šroubový), C (válcový), U (Kardanův), S (sférický) viz obr. 1-1. V modelované plošině o třech stupních volnosti jsou zastoupeny dva typy kloubů. Jedná se o rotační kloub a sférický kloub, kdy každý otevřený kinematický řetězec, tedy tři nohy disponují symetricky ovládaným rotačním kloubem u podstavu, který je v reálném modelu tvořen hřídelí krokového motoru. Přes vyvedené rameno je propojení k dalšímu již neovládanému rotačnímu kloubu, který je dále propojen s táhlem. Na konci táhla se nachází kulový čep, který v propojení s úchytem na pohyblivé platformě reprezentuje sférický kloub. Mechanismus disponuje třemi nohami, proto je často v anglicky psané literatuře dostupný pod názvem *3-RRS parallel manipulator*.

Z tab. 1-1 vyplývá počet stupňů volnosti jednotlivých kloubů, lze z toho dosadit do Grüblerovy rovnice v (1-1) pro potvrzení počtu stupňů volnosti modelovaného paralelního mechanismu. Počet pevných těles l v mechanismu lze stanovit na osm, tedy podstavu, pohyblivou platformu a tři nohy, tedy tři ramena a tři táhla. V třech otevřených kinematických řetězcích se v každém symetricky nachází dva rotační a jeden sférický kloub, tedy celkem $n = 9$. Počet stupňů volnosti jednotlivých kloubů d_i vyplývá z tab. 1-1 na tři pro sférický kloub a jeden pro rotační tedy celkově $\sum_{i=1}^9 d_i = 9 + 6 = 15$. Po dosazení lze potvrdit očekávané tři stupně volnosti

$$m_{3\text{DOFP}} = 6 \cdot (8 - 9 - 1) + 15 = -12 + 15 = 3. \quad (1-3)$$



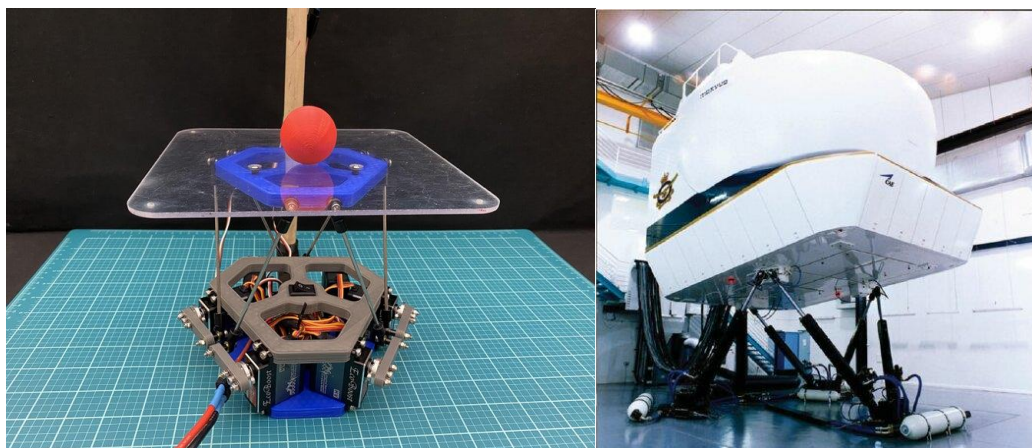
Obr. 1-1 Typy používaných kloubů v paralelních mechanismech [3].

Tab. 1-1 Počet stupňů volnosti vybraných kloubů v prostoru [3].

Typ kloubu	Stupně volnosti mezi dvěma prostorovými tuhými tělesy	Omezení mezi dvěma prostorovými tuhými tělesy
R – rotační kloub	1	5
P – posuvný kloub	1	5
H – šroubový kloub	1	5
C – válcový kloub	2	4
U – Kardanův kloub	2	4
S – sférický kloub	3	3

1.3 Přehled jiných paralelních mechanismů

V [7] je detailní přehled mnoha paralelních manipulátorů, cílem tohoto krátkého seznámení je zredukovat množství informací pro letmý přehled možných konfigurací. Obecně se mezi výhody paralelních mechanismů řadí možnosti využití v aplikacích s vyšším výkonem, kdy při aplikacích není nutně vhodným hodnotícím kritériem vyšší přesnost, za to je však hodnocena vyšší stabilita paralelního robotického celku. Obecně proto uplatnění je pro průmyslovou automatizaci [11].



Obr. 1-2 Stewartova platforma z projektu pro polohování kuličky [8] a jako letecký simulátor v [13].

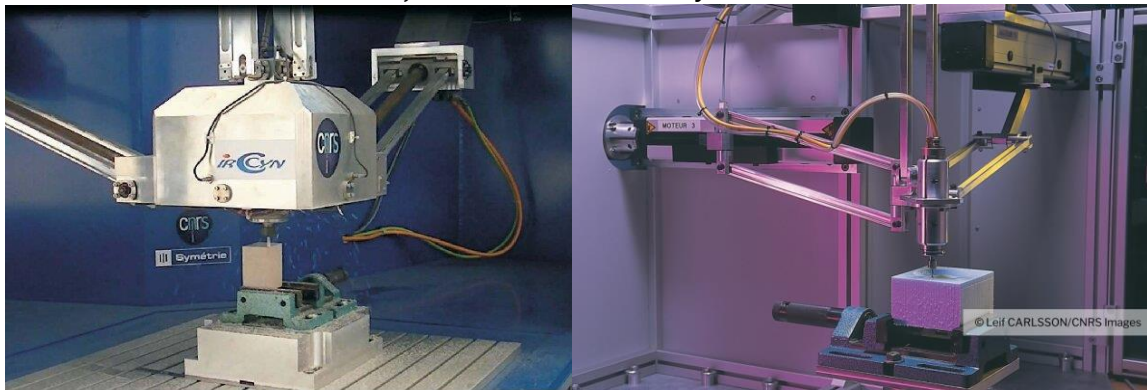
Prvním zástupcem je v literatuře často popisovaná Stewartova platforma (viz obr. 1-2) o šesti stupních volnosti (6-UPS, 6-RRS). Představena v 60. letech minulého století, disponuje šesti nohama, které mohou být složeny z různých kombinací kloubů, však se stejným cílem, a to změnou délek noh za účelem polohování pohyblivé platformy. Využití je široké ve výkonech i oblastech např. v leteckých simulátorech [13], na lodích jako stabilizační plošina kompenzující pohyby plavidla k bezpečnému přístupu na pevninu viz [14] a v neposlední řadě se jedná o možnou variantu řešení problematiky této práce polohování kuličky na plošině o šesti stupních volnosti při využití vizuální zpětné vazby (kamery) pro zpracování polohy [8].

Delta robot viz obr. 1-3 by se mohl považovat za obrácenou verzi plošiny o třech stupních volnosti, kdy pevná základna se pozičně nachází nad pohyblivou platformou často jako typ 3-RRU, či 3-RPU. Používán je od 90. let minulého století zejména v odvětvích vyžadujících rychlost a přesnost při funkci *pick-and-place* neboli zvednutí předmětu, posunu do žádoucí lokace např. v krabici, puštění předmětu. Díky této funkci je uplatnění v potravinářství, farmaceutickém průmyslu a jiných odvětvích. Časté využití mechanismu je pro pásové skládání jednotek do větších krabicových celků pro další převoz v dopravě, případně v jiných aplikacích vysokorychlostních montážních linek.



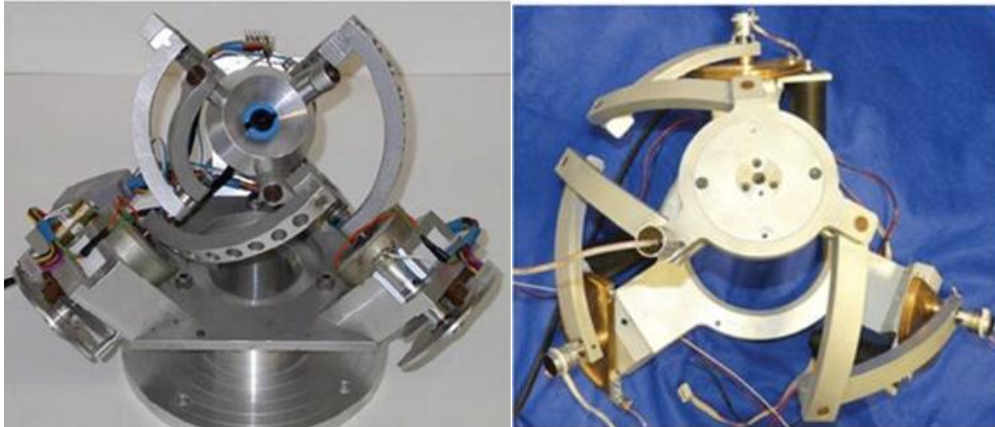
Obr. 1-3 Delta robot nabízený od ABB [15] a od ACROME [6].

Dalším vybraným zástupcem paralelních mechanismů je z anglicky psané literatury *Orthoglide* typu 3-PRP_aR viz obr. 1-4. Jedná se o novější aplikaci paralelních manipulátorů představenou v roce 2007 v kategorii robotů o třech stupních volnosti. Vyznačuje se jednoduchým a kompaktním designem, proto je hojně využíván v aplikacích obráběcích strojů. Použití je v leteckém a automobilovém průmyslu, kdy je uplatnění ve výrobě přesných obrobků. Výhodou je i robustní konstrukce, která umožňuje obrábění materiálů vyšších tuhostí.



Obr. 1-4 Orthoglide používaný pro obrábění [16], [12].

Do kategorie sférických paralelních mechanismů obdobně není do češtiny přeložen název mechanismu *Agile Eye* typu 3-RRR viz obr. 1-5. Jedná se o speciální konstrukci se třemi stupni volnosti v konfiguraci umožňující využití ve větším pracovním prostoru. Běžné je použití v lékařství, kdy může sloužit jako pomocník při chirurgických operacích, kde je vyžadovaná přesnost kamery v široké pracovní oblasti. Výhodnou aplikací tohoto mechanismu je schopnost polohování a stabilizace předmětu, např. endoskopické kamery.

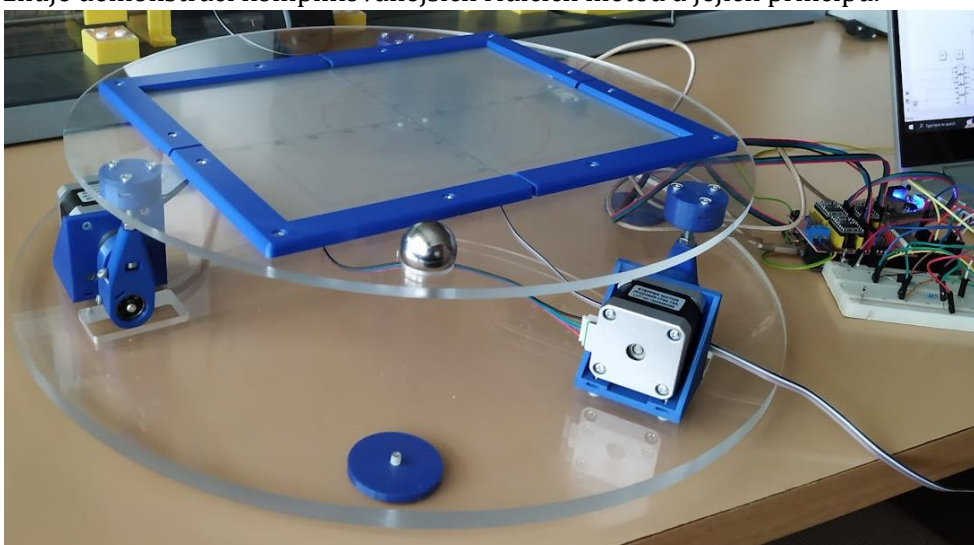


Obr. 1-5 *Agile Eye* používaný pro výzkum a méně invazivní operace [17].

1.4 Aplikace plošiny o třech stupních volnosti

Po krátkém představení vybraných paralelních manipulátorů v 1.3, je potřeba pozornost zaměřit na zkoumanou plošinu o třech stupních volnosti. Uplatnění těchto plošin je mnoho při různých výkonech, proto je časté využití v průmyslových aplikacích jako je např. letecký simulátor v [19].

Cílem této práce je vytvořit menší plošinu o třech stupních volnosti se schopností dostat kuličku do žádané polohy viz obr. 1-6. Jedná se spíše o přístroj laboratorní velikosti, který může být využit za propagačními účely řídicí techniky. Obecné principy jsou však přenositelné na plošiny větších typů. Je nutné uvést dvě vzorové práce, které slouží jako inspirace celého projektu [9], [10]. V obou je využito rezistivního dotykového panelu pro detekci polohy kuličky a pohon kinematických řetězců je přes krokové motory. Tento mechatronický celek vyžaduje znalosti konstruktérské disciplíny pro vývoj vhodného modelu pro fyzickou realizaci, zároveň však umožňuje demonstraci komplikovanějších řídicích metod a jejich principů.



Obr. 1-6 Vytvořená plošina o třech stupních volnosti pro polohování ocelové kuličky.

KAPITOLA 2: PRVKY PLOŠINY O TŘECH STUPNÍCH VOLNOSTI A VYTVÁŘENÍ MODELU

Účelem této práce je realizace plošiny schopné polohovat kuličku. Pro dosažení tohoto záměru je však potřeba nejdříve vytvořit vhodný model, který po otestování může být fyzicky realizován.

2.1 Výběr CAD softwaru pro modelování 3DOF-p

Při vytváření modelu mechanické soustavy je standardním postupem využití dostupných CAD softwarů. Tato volba byla potřeba provést již v začátku tohoto projektu, protože naučit se používat daný modelovací software vyžaduje časovou investici. Výhodou však je, že znalosti naučené v jednom modelovacím softwaru jsou často přenositelné i do softwarů jiných s menšími nuancemi. CAD softwarů je na výběr mnoho, kdy nejdříve bylo pohlédnuto na následující možnosti.

Byl vyzkoušen software od *Autodesk* s názvem *Inventor*, který je dostupný v rámci školní licence [26]. Obdobným nástrojem je *SolidWorks* [20] od francouzské společnosti *Dassault Systèmes*. Jedná se o velmi rozšířené softwarey v oblasti mechatroniky, umožňující širokou škálu testovacích funkcí na modelu s vývojem v rámci desetiletí. Výhodou je i přímá možnost exportu obou do formátu *Simscape Multibody* viz [23], [22]. Nevýhoda softwaru pro uživatele může být v požadavcích na systém ve formě grafické karty [24]. Používaným notebookem při tvoření práce je [25], který disponuje jen interní grafikou *Intel® UHD Graphics 620*, je proto tento software díky svým požadavkům na grafický výpočetní výkon pro uvedenou počítačovou soustavu nevhodný. Za tímto účelem byl vyhledáván software s menší výpočetní náročností na systém.

Možností je více a s příchodem vylepšených technologií jako virtuální realita, je snaha integrovat modelování i do prostoru přístupného např. přes brýle pro VR. Jeden z takto inovativních modelářských softwarů nabízí společnost *Onshape* [21]. Je dostupný přes studentskou licenci. Využívá online serverové úložiště (*Cloud Native CAD*), tedy požadavky na vlastní výpočetní systém jsou sníženy. Disponuje čistým designem, ve kterém se lze relativně rychle naučit, zejména díky dostupným online materiálům. Hlavní výhodou však pro tuto práci je snadnost exportu z prostředí *Onshape* do prostředí *Simscape Multibody* [27]. Z jmenovaných důvodů se jedná o software zvolený pro tvoření této práce.

2.2 *Onshape*

Účel tohoto odstavce není představení všech funkcí programu *Onshape*. Při řešení neznámé problematiky může však autor práce doporučit tři zdroje:

- otevřená dokumentace nápovědy *Onshape Help* [28],
- fórum otázek a odpovědí *Onshape Forums* [29],
- veřejně přístupné modely *Public* viz [30].

2.2.1 Modelování v *Onshape*

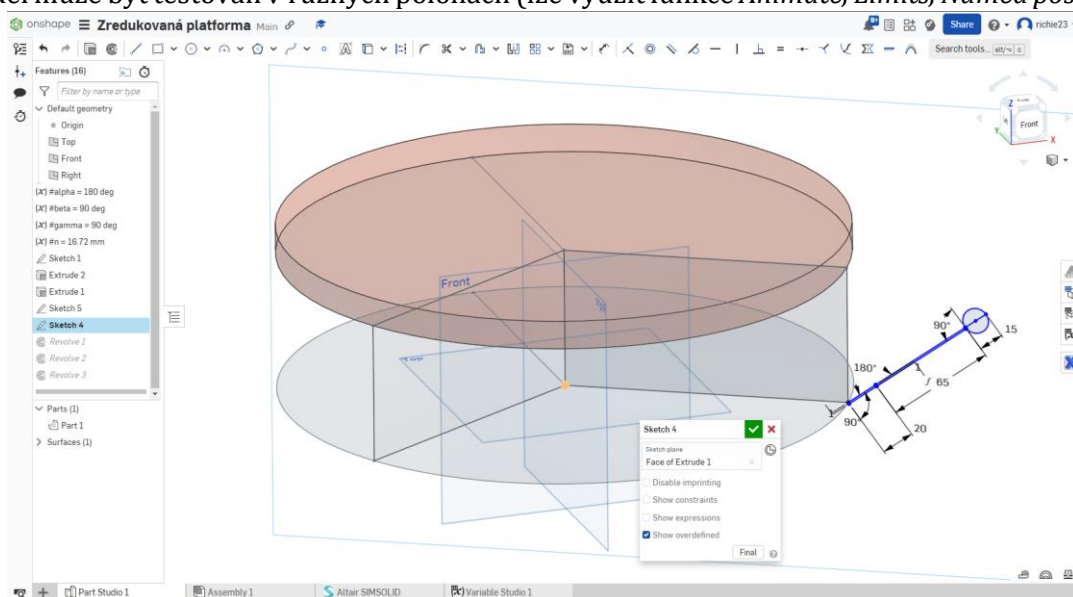
Při tvorbě plošiny o třech stupních volnosti je potřeba stanovit mechanické prvky. Po konstrukční inspiraci z [9], [10], [31]- [59] byly rozhodnuty následující předpoklady:

1. modelovaná plošina bude typu 3-RRS, tedy deska pohyblivé platformy připojená k základní desce přes tři kinematické řetězce, každý o dvou R a jednom S,
2. motory použité za účelem pohonu R kloubů u báze budou krokové, řízeny přes CNC shield [63] (nikoli stejnosměrné motory nebo servopohony kvůli menší přesnosti, složitosti při řízení),
3. využitá mikroprocesorová deska bude Arduino Due [62], která disponuje dostatečným množstvím I/O, paměti a vhodným připojením k prostředí *MATLAB & Simulink* (bylo uvažováno nad typy LiliGO s ESP 32, Raspberry Pi a STM32, ale nakonec bylo pro potřeby projektu posouzeno, že nebude potřeba větší výpočetní výkon pro využití v složitějších řídicích metodách jako MPC, pro které by např. byly vhodnější poslední dva uvedené typy),

4. zpětná vazba polohy bude realizovaná přes rezistivní dotykový panel s kontrolérem (lze využít optiku nad deskou, ale implementace je složitější, od nutnosti filtrace rušení obrazu z kamery, po výpočty vzdáleností).

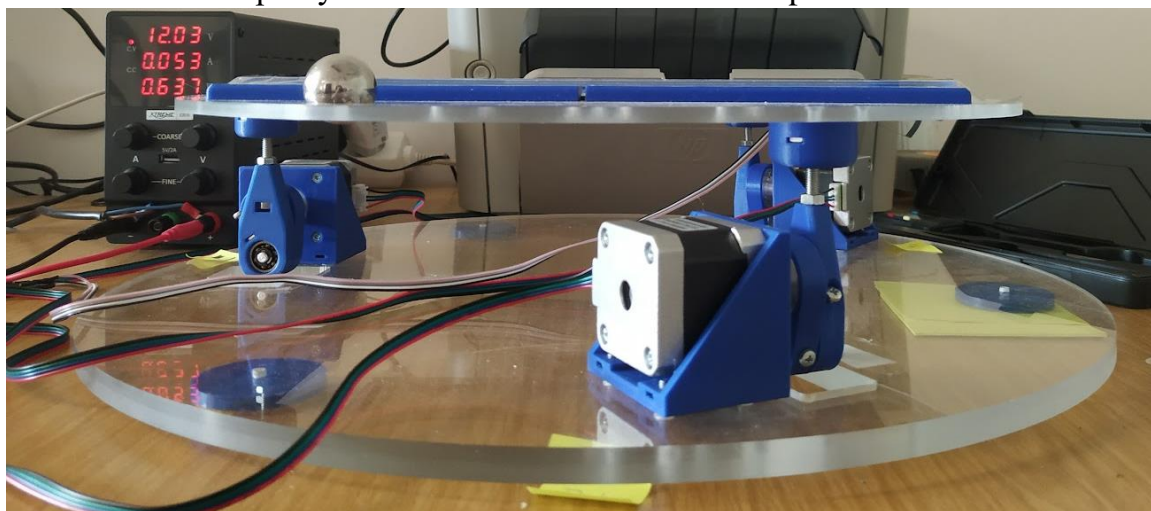
Variant implementace je mnoho, ale po pročtení článků a jiných praktických dostupných zdrojů bylo snahou vytvořit robustní řešení, které používá prověřené a dostupné prvky se snahou zajistit vzájemnou kompatibilitu. Přihlédnuto při výběru bylo i na dostupnost dokumentace pro zajištění vhodného užívání komponent. Celý seznam prvků fyzického sestavy lze zobrazit v tab. 2-1, pro simulaci v *Simulink* však není potřeba modelovat šrouby, pojistné kroužky, matice a jiné prvky, které téměř neovlivní chování modelu a mohou být nahrazeny vazbami.

Pro modelování v *Onshape* je potřeba vytvářet skici s rozměry (viz obr. 2-1), ze kterých jsou pak vytvářeny 3D prvky příslušně dostupnými funkcemi (*Extrude*, *Revolve* ...). V části sestavy (*Assembly*) jsou prvkům přiřazeny vzájemné vazby (*Fastened*, *Revolute*, *Ball Mate* ...) a složený model může být testován v různých polohách (lze využít funkce *Animate*, *Limits*, *Named positions*).



Obr. 2-1 Ukázka základních možností modelování prvků pro sestavy v Part Studio.

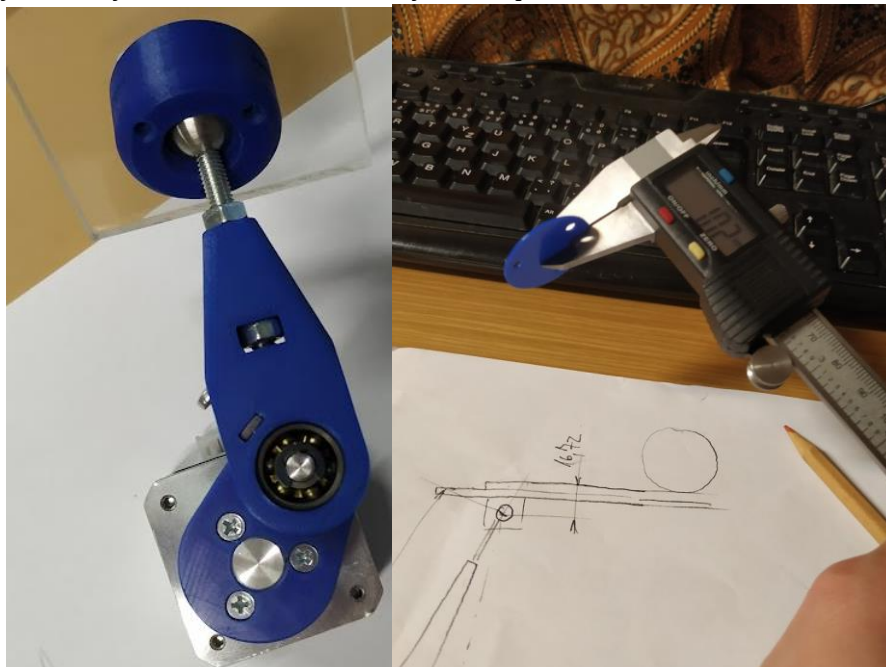
2.2.2 Praktické doplňky k modelování a sestavení 3DOFp



Obr. 2-2 Fyzická sestava polohovací plošiny o třech stupích volnosti.

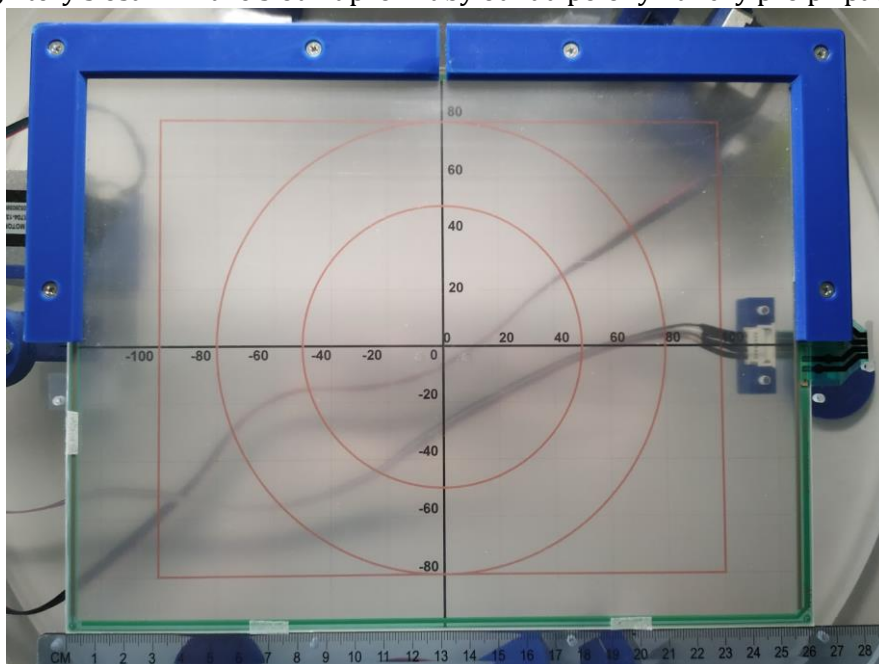
V sestavě 3DOFp je využito symetrické konstrukce pro výrobu třech identických noh viz obr. 2-2.

Nohy sestávají z 3D výtisků (modré prvky) a ze soustružených prvků (stříbrné duraluminium). Vzdálenost kliky od středu náboje do středu kulového ložiska je 20 mm a vzdálenost táhla s kulovým čepem je 65 mm ze středu ložiska do středu kulového čepu, tedy 72,5 mm od středu ložiska do vrcholu kulového čepu viz obr. 2-3. Mezi plexi pohyblivou platformou a kulovým čepem je 1mm plastový výtisk, aby při tlaku kuličky nebylo opotřebováno plexi, ale raději plastový výtisk, který může být snadně nahrazen novým viz vpravo obr. 2-3.



Obr. 2-3 Ukázka základních možností modelování prvků pro sestavy v Part Studio.

K uchycení rezistivního dotykového panelu jsou použité rohové úchytky, které mohou omezeně bránit v spadnutí kuličky viz obr. 2-4. Pod panel je také praktické vložit výtisk na transparentní fólii, který s osami může sloužit pro hrubý odhad polohy kuličky pro případné ladění.



Obr. 2-4 Ukázka rezistivního dotykového panelu uchyceného rohovými úchytkami.

2.3 Používané prvky a testování

V tab. 2-1 byl vytvořen orientační seznam většiny potřebných prvků pro projekt.

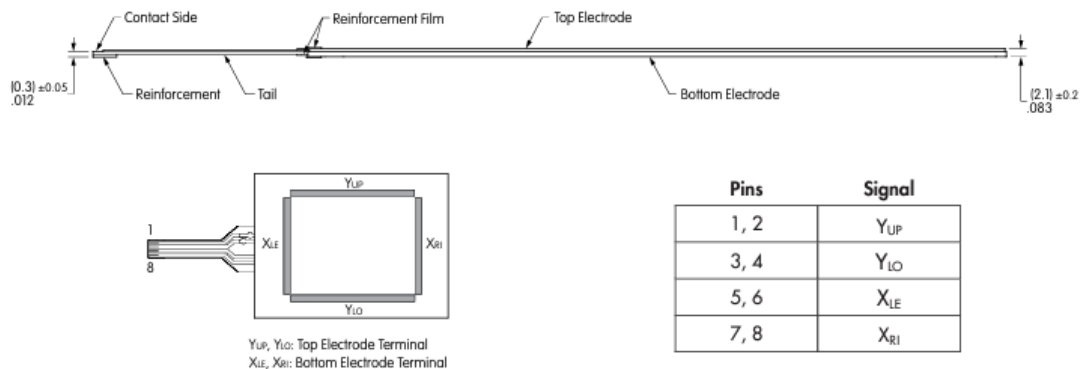
Tab. 2-1 Použité prvky s orientačními cenami včetně DPH bez částky za dopravu [3].

Název	Základní parametry	Koupeno/Vyrobeno	Cena v Kč	Počet kusů	Zdroj
Ocelové kuličky	průměr 25, 30, 34 mm	Koupeno	50	3	[64]
Krokový motor	NEMA 17, 42 mm, 1,8°	Koupeno	370	3	[65]
Dotykový panel	250 ~ 850 Ω, 1,47 N, 10 ms	Koupeno	1476	1	[67]
Vzor pod panel	Výtisk, fólie	Ze skladu školy	20	1	[70]
Kontrolér panelu	Na čtyřvodičový panel	Koupeno	318	1	[71]
CNC shield	4 sloty na 4 drivery	Koupeno	75	1	[63]
Driver krokových motorů	DRV8825, 2,2 A max, 8,2 - 36 V, 1/32 krok max	Koupeno	78	3	[73]
Arduino Due	3,3 V, 54 digitálních I/O, 12 analogových I/O	Ze skladu školy	1063	1	[62]
Tlačítko	4-pin, různé barvy	Ze skladu školy	3	3	[74]
Ledka	Různé barvy, 20 mA, 2,5 V	Ze skladu školy	2	3	[75]
Rezistor	220 Ω	Ze skladu školy	1	3	[76]
Uni. plošný spoj	Epoxid, měď	Ze skladu školy	1	190	[77]
Kabeláž	Dupont konektory různé	Ze skladu školy	1	22+	[78]
Platforma - plexi	5 mm, 380 mm kruh	CNC frézování	1700	1	[79]
Základna - plexi	10 mm, 450 mm kruh	CNC frézování	2600	1	[79]
Vrtaná kulička	Ocel, m5, 15 mm	Koupeno	120	3	[81]
Spojovací materiál	M3, M4, M5 šrouby, matice	Ze skladu školy	3	70+	[82]
Pojistný kroužek	Třmenový 4mm, 9,3 mm	Ze skladu školy	1	3	[82]
Ložisko	625, 5x16x5 mm	Ze skladu školy	100	3	[83]
Držák motoru	ABS/PETG	3D výtisk	30	3	Ons
Náboj	Duraluminim	Vysoustruženo	40	3	Ons
Klika	ABS/PETG	3D výtisk	10	3	Ons
Propoj klika/táhlo	Duraluminim	Vysoustruženo	10	3	Ons
Táhlo	ABS/PETG	3D výtisk	15	3	Ons
Pouzdro, podložka	ABS/PETG	3D výtisk	10	3	Ons
Držák konektoru	ABS/PETG	3D výtisk	5	3	Ons
Podstava podložky	ABS/PETG	3D výtisk	5	3	Ons
Protiskluzová plst'	plstěný výřezek	Ze skladu školy	2	3	[84]
Cena celkem		10 230 Kč			

2.3.1 Rezistivní dotykový panel s kontrolérem

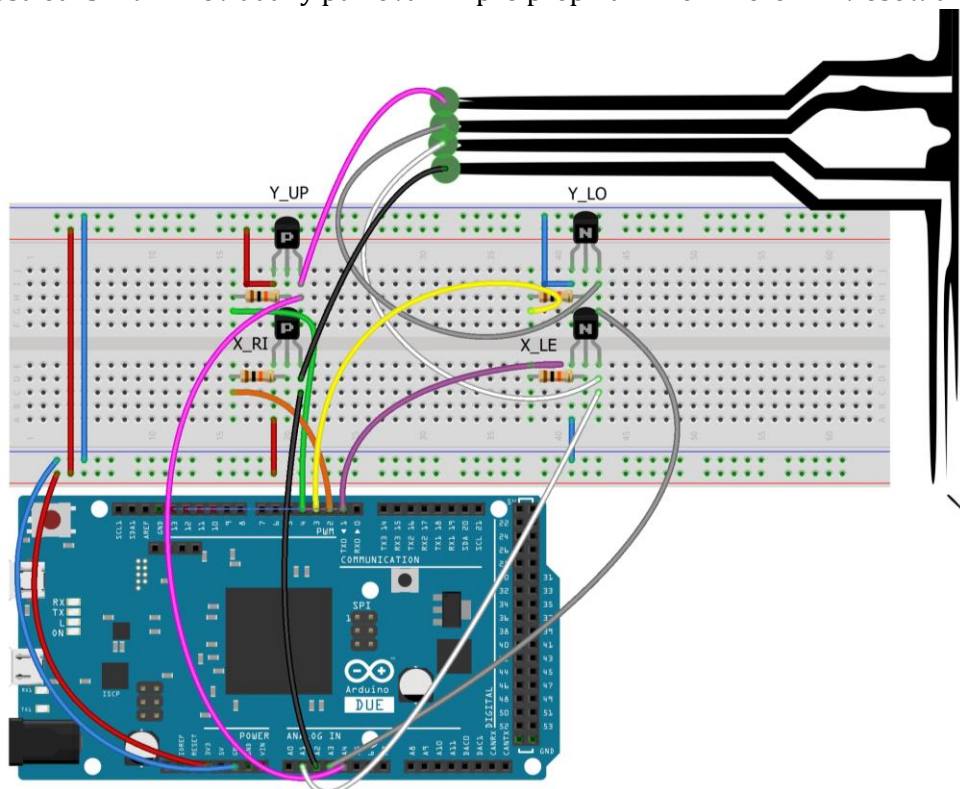
Byl zakoupen 12,1 palcový rezistivní dotykový panel s úzkým rámečkem typu 1 a horizontálním vodičovým ocasem. Tento typ není dostupný v základní dokumentaci [68], lze však získat parametry z [69] s šířkou 261 mm a výškou 198 mm.

Princip rezistivního dotykového panelu zjednodušeně spočívá v propojení horní a dolní elektrody (viz obr. 2-5). Po přiložení napětí v měřené ose x (3,3 V na X_{RI} a 0 V na X_{LE}), resp. y (3,3 V na Y_{UP} a 0 V na Y_{LO}) vzniká v bodě dotyku napěťový dělič a lze v neměřené ose na Y_{LO} , resp. X_{LE} měřit napětí vůči společné zemi (port GND Arduina), kdy dle velikosti napětí lze určit místo dotyku [85].



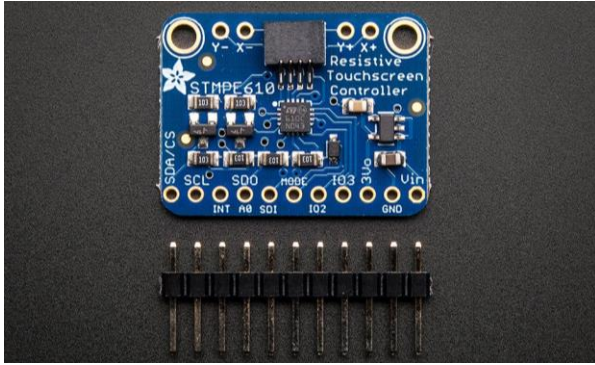
Obr. 2-5 Ukázka částí rezistivního dotykového panelu a vyvedené signály z čtyřvodičového kontaktu [68].

Pro čtení polohy nebylo nutně potřeba kontrolér kupovat. Pro prvotní testy tohoto projektu byl vytvořen sestavený měřicí celek pomocí kombinace PNP, NPN tranzistorů s odpory 10 k Ω v bázi viz obr. 2-6. V prostředí *Simulink* nelze v blocích dynamicky měnit funkci mezi vstupem a výstupem (v A IDE lze použít funkci `pinMode()` měnící funkci I/O). Pomocí schématu z obr. 2-6 lze vyvést analogové piny A1, A2, A3, A4 pro měření napětí a digitální piny D1, D2, D3, D4 jsou z prostředí *Simulink* ovládány pulzováním pro přepínání mezi měřením v ose x a v ose y.



Obr. 2-6 Ukázka zapojení rezistivního dotykového panelu bez kontroléru pro komunikaci se *Simulink*.

Využit pro finální realizaci byl však kontrolér (viz obr. 2-7), který na obdobném principu plní úlohu převádění analogově měřených napětí do digitálního signálu (A/D převodník). Uvedený kód je ze [72] a slouží k testování kontroléru v A IDE.



Obr. 2-7 Kontrolér Adafruit STMPE610 [72].

```
/*
This is an example for the Adafruit STMPE610 Resistive
touch screen controller breakout
----> http://www.adafruit.com/products/1571

Check out the links above for our tutorials and wiring diagrams
These breakouts use SPI or I2C to communicate

Adafruit invests time and resources providing this open source code,
please support Adafruit and open-source hardware by purchasing
products from Adafruit!

Written by Limor Fried/Ladyada for Adafruit Industries.
MIT license, all text above must be included in any redistribution
*/
#include <SPI.h>
#include <Wire.h>
#include "Adafruit_STMPE610.h"
// Adafruit_STMPE610(SDA_CS, SDI, SDO, SCL)
Adafruit_STMPE610 touch = Adafruit_STMPE610(31, 33, 35, 37);

void setup() {
  Serial.begin(9600);
  Serial.println("Adafruit STMPE610 example");
  Serial.flush();

  pinMode(10, OUTPUT);

  if (! touch.begin()) {
    Serial.println("STMPE not found!");
    while(1);
  }
  Serial.println("Waiting for touch sense");
}

void loop() {
  uint16_t x, y;
  uint8_t z;
  if (touch.touched()) {
    // read x & y & z;
    while (! touch.bufferEmpty()) {
      Serial.print(touch.bufferSize());
      touch.readData(&x, &y, &z);
      Serial.print("->(");
      Serial.print(x); Serial.print(", ");
      Serial.print(y); Serial.print(", ");
      Serial.print(z);
    }
  }
}
```

```

Serial.println(" ");
}
touch.writeRegister8(STMPE_INT_STA, 0xFF); // reset all ints, in this
example unneeded depending in use
}
delay(10);
}

```

Potřebným doplňkem je kromě běžně dostupných Arduino knihoven využít i dodatečně knihovny `Adafruit_STMPE610.h`, `Adafruit_STMPE610.cpp` dostupné ze [72]. Funkce kontroléru je převést analogově změřené napětí na digitální hodnoty, ze kterých lze už určit polohu. Fyzické zapojení je zobrazeno v obr. B-3. K otestování funkce panelu lze použít prst s tlakem, či ocelovou kuličku.

Výrobce v [68] uvádí aktivační přítlačnou sílu panelu jako *1,47 N maximum*. V různých pozicích dotykového panelu je potřeba jiná aktivační síla a je rozdíl mezi přítlačnou silou hrotem (např. dotykovým perem) a prstem, nebo kuličkou. Minimální přítlačná síla závisí proto na mnoha parametrech, a pravděpodobně proto není výrobcem uvedena za jakých podmínek je daná minimální přítlačná síla, ale raději uvádí hodnotu 1,47 N pro kterou by měla detekce polohy proběhnout téměř vždy.

Pomocí vztahu (2-1) lze stanovit poloměr kuličky jako definující proměnnou pro dostatečnou přítlačnou sílu.

$$F_{\text{přítlačná}} \leq F_{g,\text{kul.}} = m_{\text{kul.}} \cdot g = V_{\text{kul.}} \cdot \rho_{\text{kul.}} \cdot g = \frac{4}{3} \cdot \pi \cdot r_{\text{kul.}}^3 \cdot \rho_{\text{kul.,ocel}} \cdot g. \quad (2-1)$$

Úpravou (2-1) lze stanovit minimální poloměr kuličky:

$$r_{\text{kul.}} \geq \sqrt[3]{\frac{3 \cdot F_{\text{přítlačná}}}{4 \cdot \pi \cdot g \cdot \rho_{\text{kul.,ocel}}}} = \sqrt[3]{\frac{3 \cdot 1,47}{4 \cdot \pi \cdot 9,81 \cdot 7800}} \doteq 0,0166 \text{ m} = 16,6 \text{ mm}. \quad (2-2)$$

Kulička o nejbližším průměru celého čísla 34 mm váží kolem 160,5 g, což už je při pádu z přípravku signifikantní úder. Proto byly testovány i kuličky menších průměrů 25 mm a 30 mm. Pro kuličku o menším poloměru 12,5 mm jsou na panelu měřeny mylné hodnoty, které mohou působit nežádoucí zásahy regulace. Při testování kuličku o poloměru 15 mm již mylné hodnoty nastávají velmi zřídka, proto může být vhodnější volbou optimalizace mezi menší vahou a dostatečnou přítlačnou silou. Pro větší kuličku poloměru 17 mm však panel funguje nejlépe.

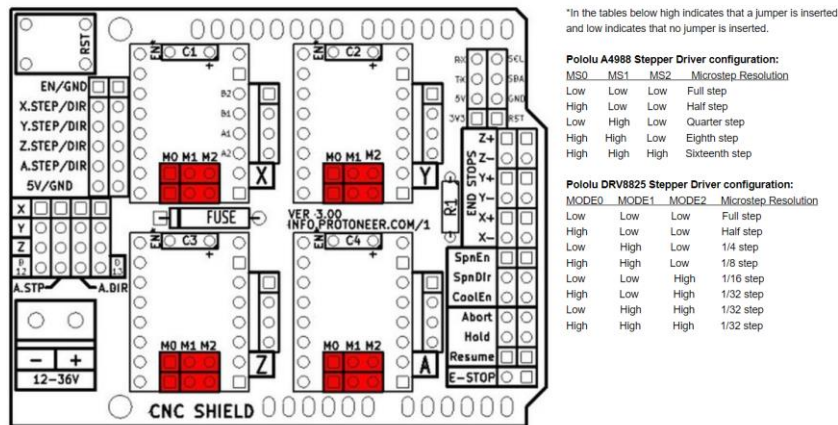
2.3.2 Arduino CNC shield V3

CNC shield je rozšiřující modul vytvořen primárně pro typ Arduino UNO, ale je funkční i pro používanou mikroprocesorovou desku v tomto projektu typu Arduino DUE. Výhodou použití tohoto modulu je snadné připojení třech driverů Pololu DRV8825 (viz [73]) pro řízení třech krokových motorů NEMA17 (viz [65]). CNC shield je elegantní, již hotové řešení, které umožňuje snadnou náhradu poškozeného driveru výměnou kus za kus, má zabudované kondenzátory pro filtraci nežádoucích špiček napětí, umožňuje snadné nastavení mikrokrokování a modul obsahuje piny pro připojení koncových spínačů v aplikacích kde to je požadováno.

Před připojením driverů do CNC shield, je potřeba zvolit jaké bude využito mikrokrokování. Mikrokrokování používá pro zvětšení rozsahu poloh otočení krokového motoru kombinaci částečného nabuzení více cívek, pro dosažení natočení rotoru do mezipoloh, tedy např. místo $1,8^\circ$ na krok (200 kroků na plnou otáčku 360°) lze např. při mikrokrokování $1/4$ dosáhnout $0,45^\circ$ na mikrokrok (800 kroků na plnou otáčku 360°). Nevýhodou mikrokrokování je snížení momentu (i přídržného ve statické poloze), a také snížená přesnost mezi mikrokroky oproti funkci plného krokování. Pro projekt bylo při testování rychlosti komunikace s prostředím *Simulink* a při uvažování požadované přesnosti a sníženého momentu stanoveno jako výhodné využít mikrokrokování $1/4$ kroku, ale pro plynulejší regulaci by bylo výhodnější poměr zvýšit.

Napájení pro projekt je volené 12 V. Z obr. 2-9 lze určit jmenovitý fázový proud krokového motoru 1,7 A pro výpočet nastaveného napětí na driveru pro proudové omezení.

$$U_{\text{ref}} = \frac{I_{fN}}{2} = \frac{1,7}{2} = 0,85 \text{ V.} \quad (2-3)$$



Obr. 2-8 CNC shield nastavení mikrokrokování [72].

Model No.	Step Angle	Motor Length	Current /Phase	Resistance /Phase	Inductance /Phase	Holding Torque	# of Leads	Detent Torque	Rotor Inertia	Mass
	(°)	(L)mm	A	Ω	mH	kg.cm	No.	g.cm	g.cm	Kg
JK42HS40-1704	1.8	40	1.7	1.65	3.2	4.0	4	150	54	0.28

Obr. 2-9 Parametry krokového motoru NEMA17 [66].

Pro testování funkce CNC shield lze v A IDE spustit následující kód.

```
// Zdroj: https://navody.dratek.cz/navody-k-produktum/arduino-cnc-shield-driver-a4988-motor-28byj-48.html
#include <SPI.h>

// nastavení čísel projevovacích pinů
#define ENABLE 8
#define X_STEP 2
#define Y_STEP 3
#define Z_STEP 4
#define X_DIR 5
#define Y_DIR 6
#define Z_DIR 7

void setup() {
  // nastavení směrů pro všechny piny
  pinMode(X_STEP, OUTPUT);
  pinMode(Y_STEP, OUTPUT);
  pinMode(Z_STEP, OUTPUT);
  pinMode(X_DIR, OUTPUT);
  pinMode(Y_DIR, OUTPUT);
  pinMode(Z_DIR, OUTPUT);
  pinMode(ENABLE, OUTPUT);
  // povolení řízení pro všechny drivery
  digitalWrite(ENABLE, LOW);
}

void loop() {
  // otočení motoru o jednu otáčku
```

```

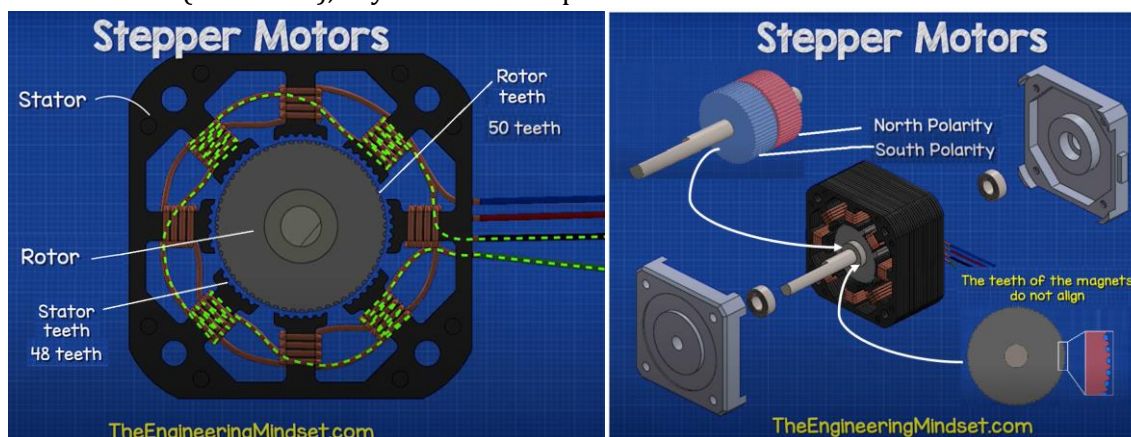
// (1,8 ° na krok, na celou otáčku 1600 kroků při mikrokrokování 1/8)
pohybOsy(false, X_DIR, X_STEP, 1600);
delay(1000);
pohybOsy(false, Y_DIR, Y_STEP, 1600);
delay(1000);
pohybOsy(false, Z_DIR, Z_STEP, 1600);
delay(1000);
// otočení motoru o jednu otáčku v opačném směru
pohybOsy(true, X_DIR, X_STEP, 1600);
delay(1000);
pohybOsy(true, Y_DIR, Y_STEP, 1600);
delay(1000);
pohybOsy(true, Z_DIR, Z_STEP, 1600);
delay(1000);
}

void pohybOsy(boolean smer, byte dirPin, byte stepPin, int kroky) {
  // zápis směru na příslušný pin DIR
  digitalWrite (dirPin, smer);
  delay(50);
  // smyčka pro provedení předaného množství kroků
  for (int i = 0; i < kroky; i++) {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds (800);
    digitalWrite(stepPin, LOW);
    delayMicroseconds (800);
  }
}

```

2.3.3 Hybridní krokový motor NEMA17

Princip funkce krokového motoru je vysvětlen např. v [86] - [88], kdy je fyzicky na statoru v ukázce obr. 2-10 osm pólových nástavců po šesti zubech, tedy celkem 48 zubů na statoru. Rotor je složen ze dvou permanentních magnetů se severní a jižní polaritou. Tyto části musí vůči sobě být pootočený, aby v planárním pohledu byl střídavě zub severní polarity a zub jižní polarity. Rotor má 50 zubů (o dva více), aby se nezasekl v poli statoru

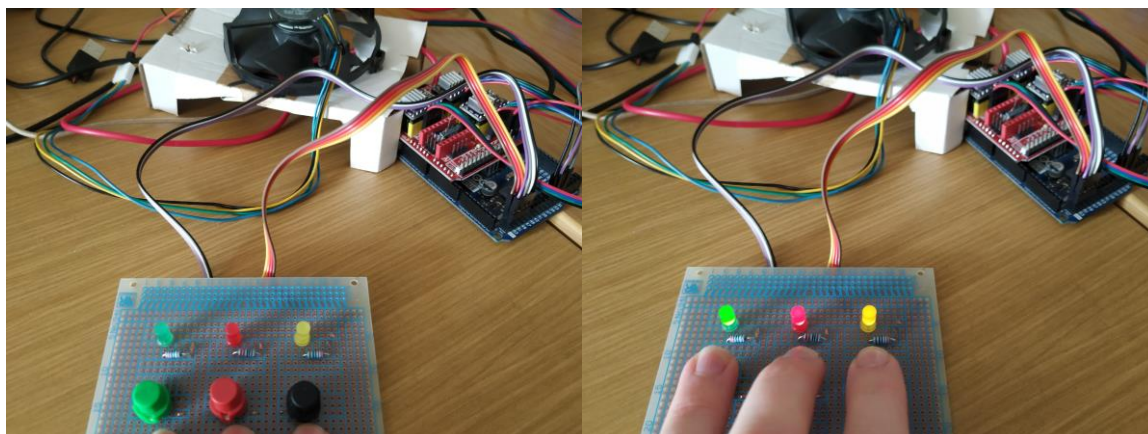


Obr. 2-10 Ukázka statoru, rotoru a zapojení cívek krokového motoru [88] (upraveno).

Pro zapojení čtyř vodičů krokových motorů lze postupovat dle obr. B-3 pro požadovaný smysl otáčení levotočivě při $DIR = 1$ a pravotočivě při $DIR = 0$.

2.3.4 Univerzální plošný spoj s tlačítky a signalizačními diodami

Deska obsahuje jednoduchou sestavu třech barevných tlačítek a diod (připojených přes rezistory 220 Ω k omezení proudu). Funkce tlačítek je popsána v části 5.1, kdy je počítán počet stisků a inkrementováno a vytvořena funkce pro resetování. Proces tvorby přípravku je od pájení a vytváření konektorů po testování viz obr. 2-11.

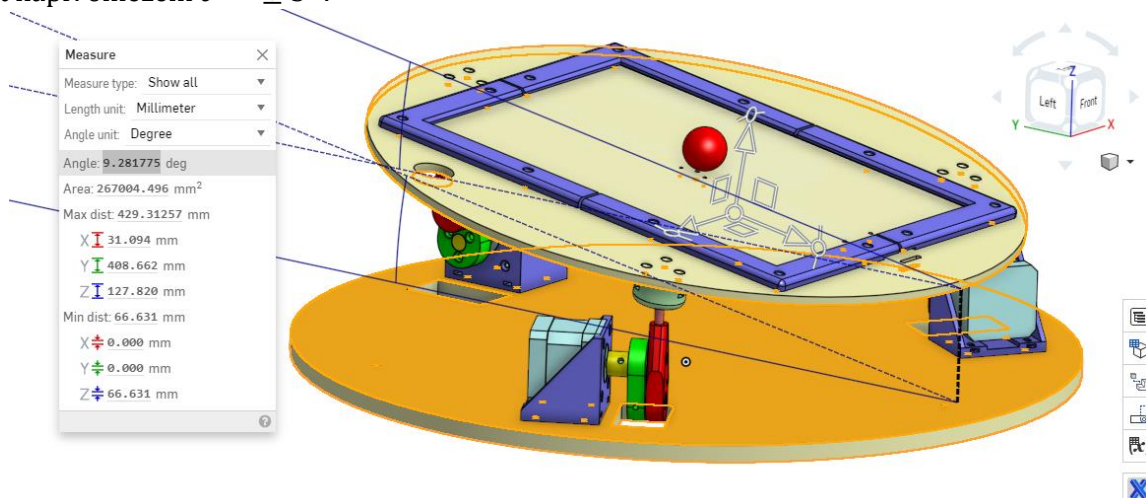


Obr. 2-11 Testování vypnutých ledek při nezmáčknutém stavu a zapnutí ledek při zmáčknutí tlačítek.

2.4 Finální podoba modelu 3DOFp v Onshape, určení mezních náklonů

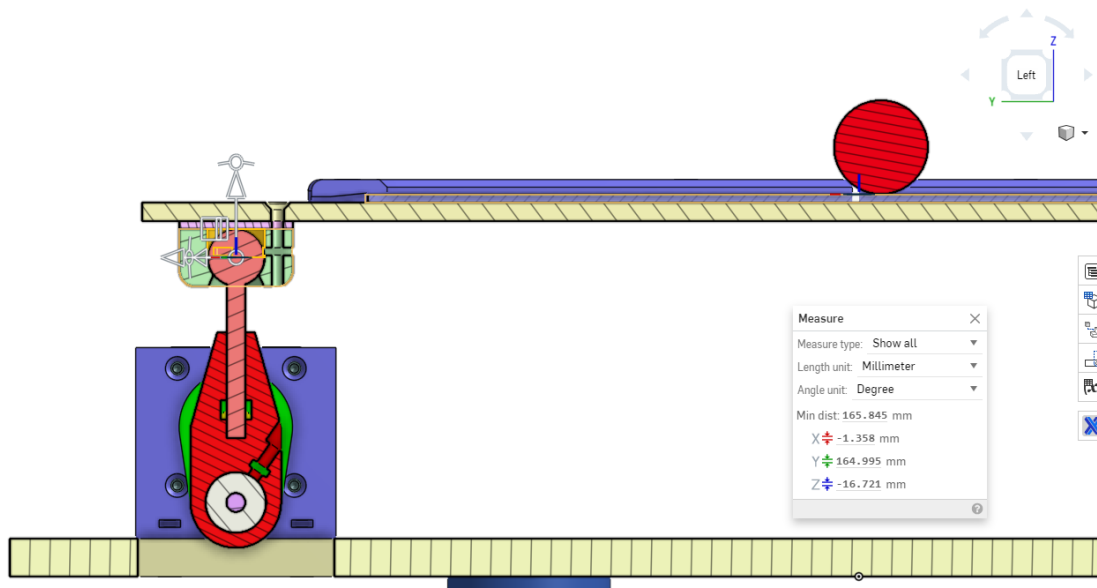
V programu *Onshape* je vytvořena zjednodušená sestava vyrobené plošiny 3DOFp. Pevné vazby nahrazují funkci šroubů a matic, rotační vazby představují otáčení hřídelí a ložisek, sférické vazby reprezentují klouby kuličky na táhle v pouzdře a planární vazbou je reprezentován pohyb ocelové kuličky na dotykovém panelu v modelu *Onshape*. K omezení pohybu kuličky v rámci rámečku panelu lze planární vazbu omezit $\pm 118,5 \text{ mm} \times \pm 87 \text{ mm}$ (rohové úchytky zmenšují aktivní rozměr dotykového panelu). Proces tvorby modelu sestavy je hrubě představen v 2.2.1, kdy jsou dále uvedeny ukázky měření ve výsledném modelu.

Pro určení maximálních dosažitelných stavů vytvořeného modelu, je potřeba nastavit žádané úhly rotačních kloubů hřídele. Vhodné je využít funkci *Named positions* (viz obr. 2-15), kdy lze po aplikaci daných rotačních úhlů měřit náklon pohyblivé platformy vůči pevné základně. Pohledem na obr. 2-12 lze určit maximální náklon kolem osy x v pravotočivém smyslu otáčení, kdy je změřen maximální úhel náklonu jako $\theta \doteq 9,282^\circ$. Obdobně lze provést měření v levotočivém smyslu otáčení a lze stanovit maximální úhel $\theta \doteq -9,32^\circ$. Úhly jsou rozdílné, protože otáčení hřídele jednoho motoru při pravotočivém náklonu je jen v ose x , zatímco hřídele dvou motorů při levotočivém náklonu jsou natočeny vůči souřadnému systému a umístění motorů je v poloviční vzdálenosti x od středu oproti samotnému motoru na levé straně. Tato omezení jsou důležitá zejména při zadávání žádaného úhlu do inverzní kinematiky a je vhodné uvažovat s rezervou, lze volit např. omezení $\theta = \pm 5^\circ$.



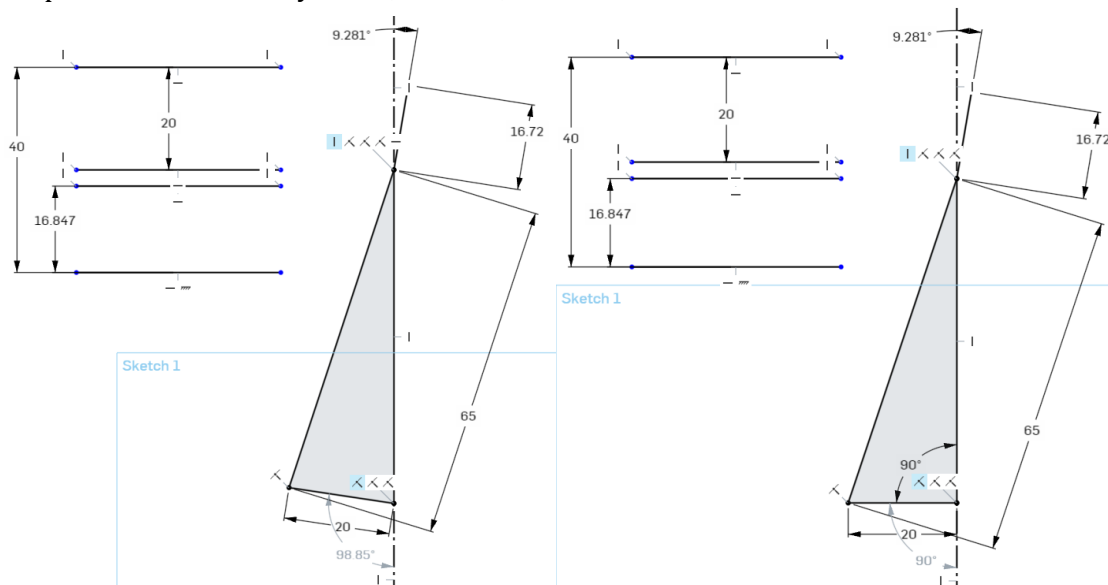
Obr. 2-12 Ukázka měření maximálního úhlu náklonu pravotočivě v ose x o úhel $\theta \doteq 9,282^\circ$.

Pro výpočet inverzní kinematiky je nutné uvažovat posun pohybové roviny kuličky vůči rovině sférických kloubů, proto lze z modelu stanovit vzdálenost 16,721 mm středu kulového čepu od horní plochy panelu z obr. 2-13.



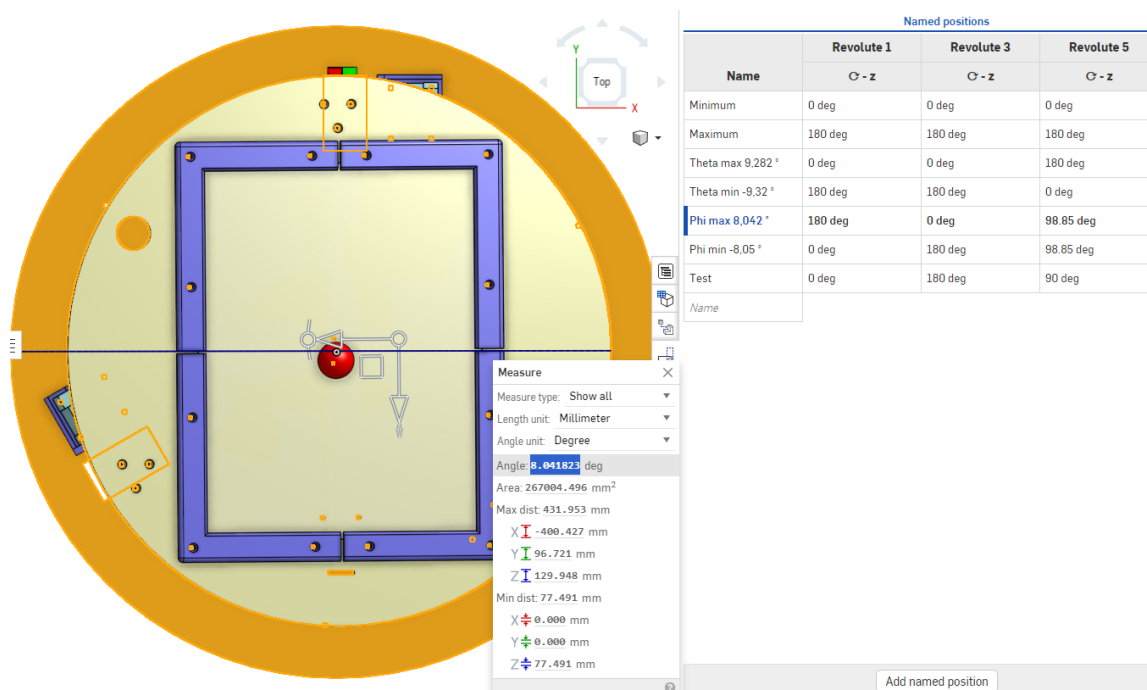
Obr. 2-13 Výška bodu dotyku ocelové kuličky vůči středu sférického kloubu 16,721 mm.

Pro měření rotace v ose y je potřeba zajistit motor vyobrazený v obr. 2-13 v poloviční výšce, která neodpovídá natočení kliky o 90°, ale o 98,85° dle obr. 2-14.



Obr. 2-14 Z planární skici v mm je střední dosažitelná výška pro úhel 98,85°, ne pro poloviční úhel 90°.

Dle obr. 2-15 je změřen maximální úhel v pravotočivém smyslu otáčení na $\phi \doteq 8,042^\circ$ a v levotočivém smyslu otáčení na $\phi \doteq -8,05^\circ$. Obdobně je vhodné omezit úhel mezi dosažitelnými hodnotami např. $\phi = \pm 5^\circ$ při nastavené poloviční dosažitelné vzdálenosti v ose z na nepohyblivém motoru.



Obr. 2-15 Ukázka měření maximálního úhlu náklonu pravotočivě v ose y o úhel $\phi \doteq 8,042^\circ$.

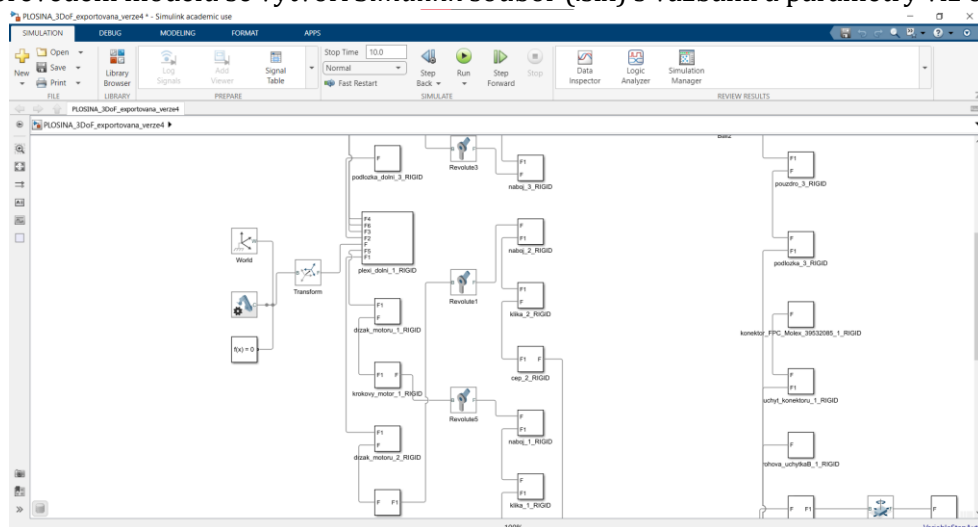
2.5 Exportování modelu z Onshape do Simulink a další úpravy

Výhodou prostředí Onshape je snadný export do Simscape Multibody [27]. Při dokončení sestavy v Onshape stačí zkopírovat URL webové stránky. Dále jen vložit zkopírované URL do následujícího kódu matlab funkce (*m.file*) a spustit (*Run*). Před spuštěním programu je vhodné si vytvořit složku, kam bude model importován a přiřadit adresu přes (*Set Path*). Program musí mít do této složky přístup k zapisování, lze pravým kliknutím myši na ikonu MATLAB a volbou *Run as administrator*.

```

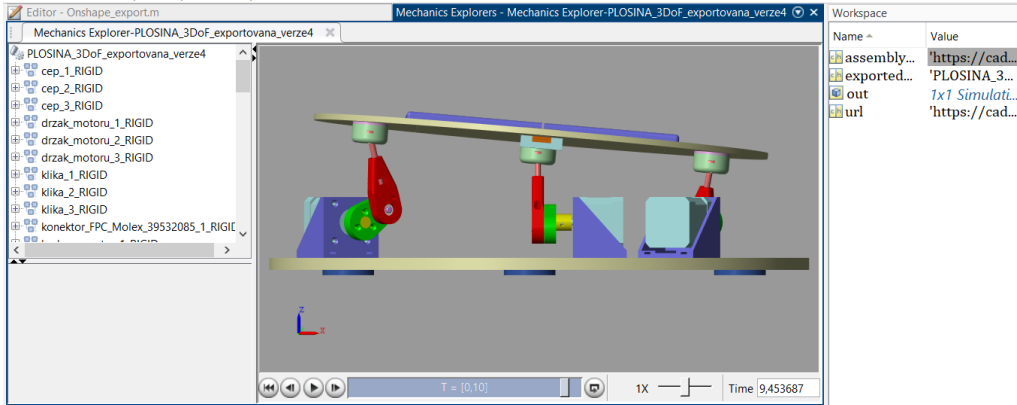
%% Onshape_export.m v příloze
assemblyURL =
'https://cad.onshape.com/documents/69a5b6acd1c7eda404dccc55/w/0d0a42b8275
1fc71461c2f65/e/c77f268dcfd30e64195fe981';
exportedModel = smexportonshape(assemblyURL);
smimport(exportedModel);
    
```

Po převedení modelu se vytvoří Simulink soubor (.slx) s vazbami a parametry viz obr. 2-16.



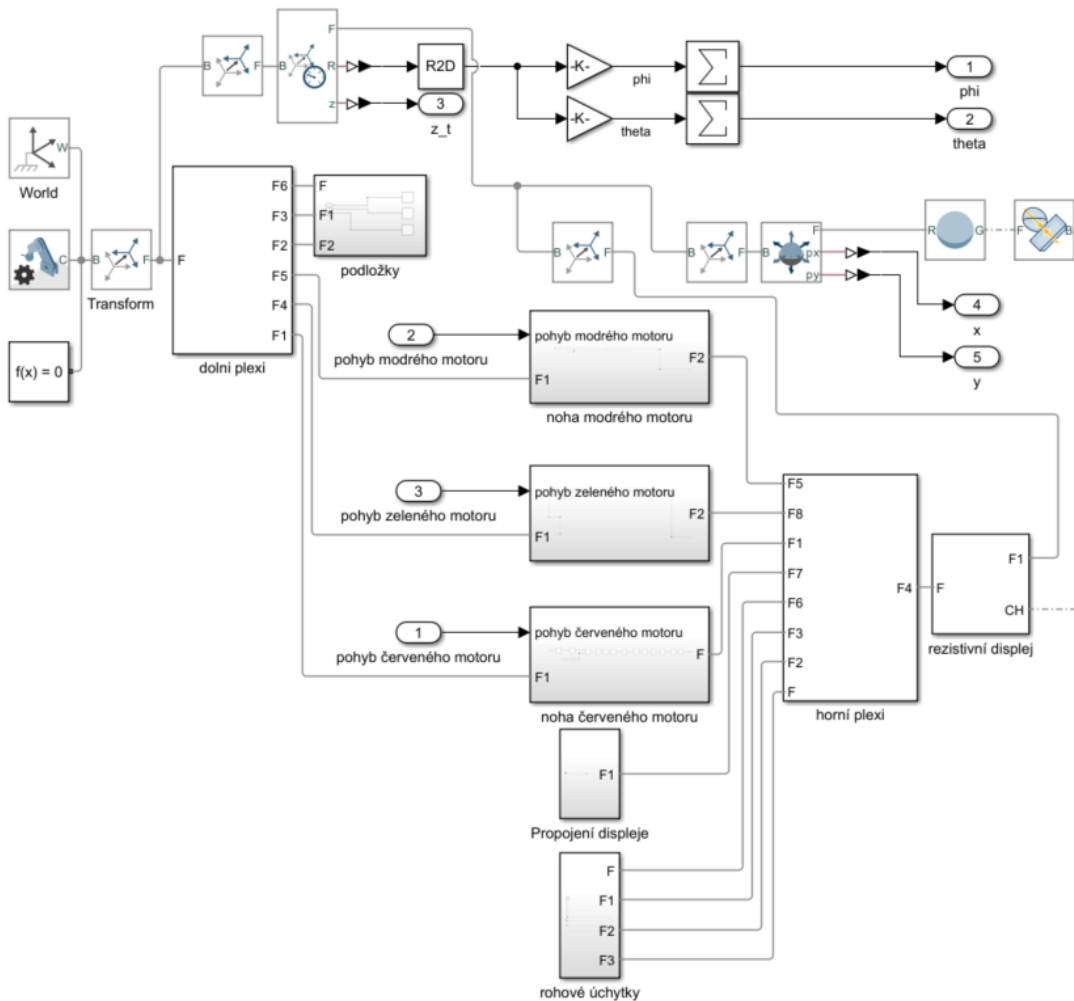
Obr. 2-16 Importovaný model z Onshape do Simulink.

Při spuštění neupraveného importovaného modelu je volným vazbám přiřazen náhodný pohyb, který lze pomocí *Mechanics Explorers* sledovat. Pokud je sestava správně propojena, tak by všechny vazby měly být v pohybu a neměly by nastávat nežádoucí polohy. Jestli se model vizuálně nechová dle představ, je to pravděpodobně indikace toho, že nějaké vazby nejsou správně definovány a je potřeba se vrátit do prostředí *Onshape* a ověřit vhodnost sestavy.

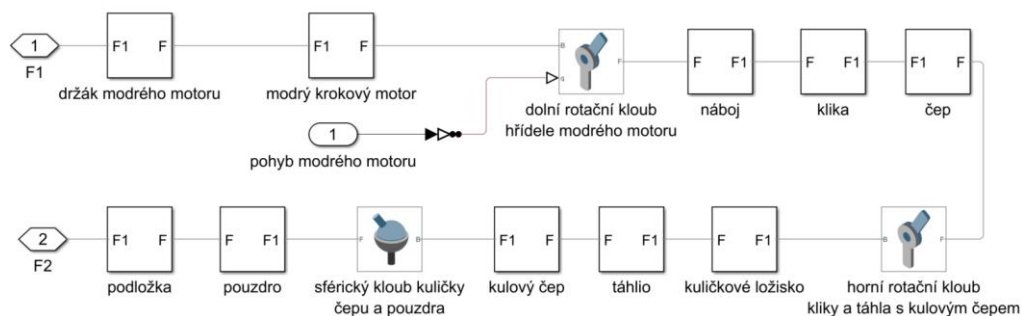


Obr. 2-17 Importovaný model z Onshape do Simulink.

Exportovaný model (viz obr. 2-16) není po exportu ve vizuálně nejpřívětivějším stavu. Je proto vhodné do modelu vložit trochu času a jednotlivé prvky a vazby přiřadit do skupin viz obr. 2-18.



Obr. 2-18 Upravený model přiřazený do podskupin pro přehlednost s členy pro kontakt kuličky.



Obr. 2-19 Subsystém nohy modrého motoru s vazbami.

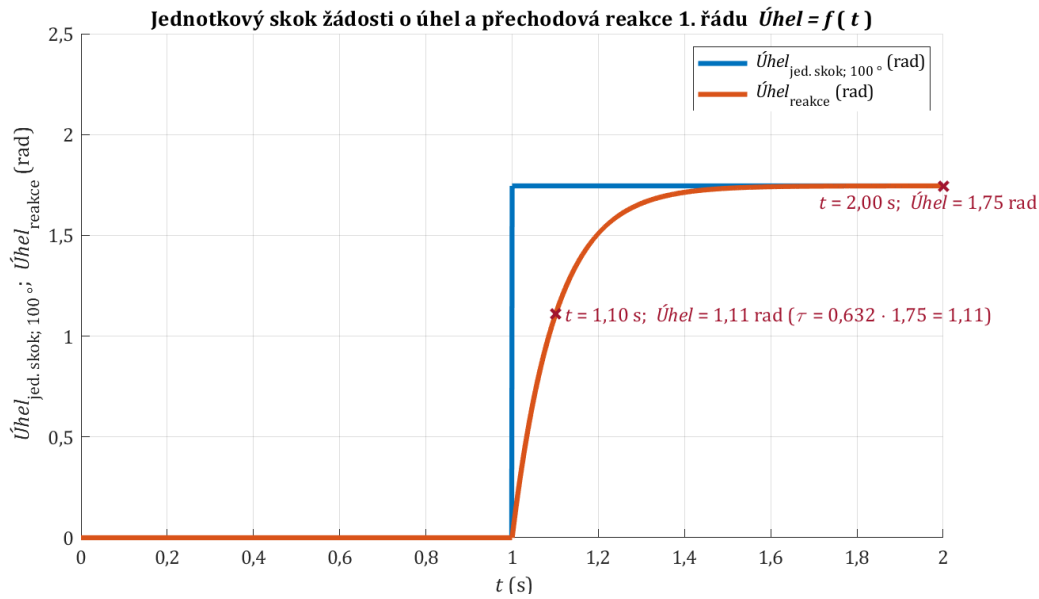
Je nutná úprava pohybovaných rotačních vazeb viz obr. 2-19 mezi krokovým motorem a nábojem. Při otevření vazby je změna definice ovládání (*Actuation*), kdy je potřeba změnit pohyb definovaný vstupem (*Motion: Provided by input*). Při definovaném pohybu je potřeba upravit nedefinovaný moment na automaticky počítaný (*Torque: místo None dát Provided by input*). Na rotačním kloubu se objeví nový vstup pro definování pohybu, zde nelze provést přímý spoj se standardním prvkem z prostředí *Simulink*, protože prvky jsou z knihovny *Simscape Multibody* a jejich vazby jsou fyzickým signálem. Pro propojení signálů fyzických prvků lze použít *Simulink-PS Converter* kde v položce *Filtering and derivatives* je nutné zvolit filtrovaný vstup při výpočtu derivace (*Filter input, derivatives calculated*) a u *Input filtering order* nastavení filtru druhého řádu (*Second-order filtering*). Časovou konstantu filtru lze ponechat na 1 ms.

Exportovaný a upravený model využívá prvků z knihovny *Simscape Multibody* a lze otestovat vložení požadavku různého úhlu pro pohyb barevně označených motorů viz obr. 2-21. Při požadavku otočení o úhel hřídele krokového motoru, není reakce okamžitá, protože z driveru je postupně krokováno (pulzy *STEP*), dokud není dosaženo požadovaného počtu kroků. Toto zpoždění je v simulaci reprezentováno přenosovou funkcí prvního řádu v obecném tvaru, kdy časová konstanta (63,2% hodnota ustálení) je uvažovaná $\tau = 100$ ms. Doba ustálení od aplikování jednotkového skoku do 99% požadované hodnoty lze uvažovat zhruba při $5 \cdot \tau = 500$ ms, což reprezentuje časové zpoždění reakce motoru při požadavku úhlu otočení 100° při komunikaci z prostředí *Simulink* s mikrokrovním $1/8$ při periodě pulzu mikrokroku 1 ms (při testování mikrokrovním $1/4$ bylo sníženo na $\tau = 20$ ms) [89]. Přechodovou funkci lze uvažovat ve tvaru:

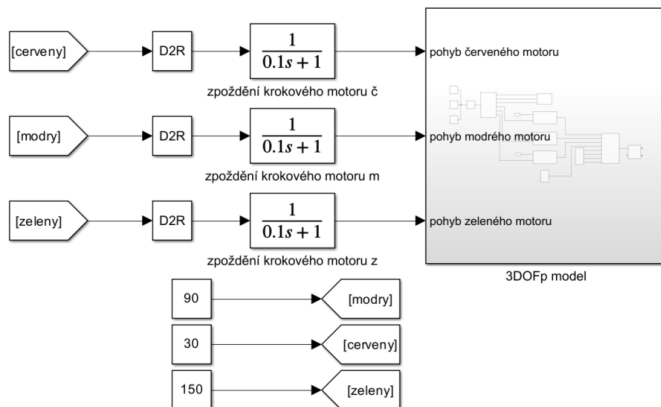
$$\frac{1}{1 + s \cdot \tau} = \frac{1}{1 + s \cdot 0,1} \quad (2-4)$$

Před spuštěním vlastního testu otáčení hřídelí motoru, je potřeba zvolit vhodný řešič, který ovlivní výsledky simulace. Pro vytváření mechanického modelu je využita knihovna *Simscape Multibody*, která využívá fyzických signálů. Pro fyzické systémy se nabízí možnost využití řešiče diferenciálních algebraických rovnic vytvořeného přímo pro systémy využívající knihovnu *Simscape*. Pro omezení výpočetního času byl volen maximální krok 10 ms a doba simulace nebyla omezena (*inf*), ale dle potřeby byla upravována na omezený čas (např. 150 s). Pro rychlejší simulaci lze volit funkci *Accelerator* (optimalizuje C – kód).

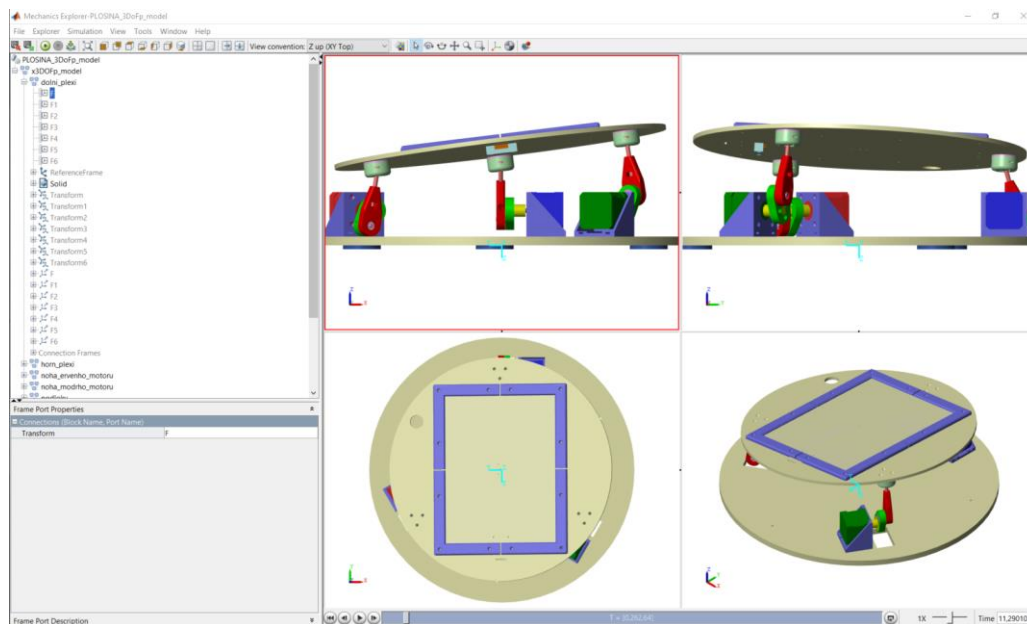
Po spuštění testování modelu z obr. 2-21 dojde k automatickému spuštění okna *Mechanics Explorer* (obr. 2-22) umožňující vizualizaci pohybů a chování systému. V tomto prostředí lze nastavit různé pohledy na celek. Umožněna je funkce přehrávání dynamických dějů systému a je funkce rychlosti přehrávání pro např. zpomalení průběhů. Je umožněn export do zvoleného video formátu, pro prezentování ukázek chování modelu. V neposlední řadě je umožněn pohled na souřadné osy prvků a celkového systému, kdy dle orientace os je potřeba pro mechanický systém vhodně nastavit vlivy jako tíhové zrychlení. Z obr. 2-22 je modře orientovaná osa z v kladném směru svisle nahoru, tedy v parametrech fyzické simulace lze dát třetí souřadnici se záporným znaménkem pro tíhového zrychlení $-9,81 \text{ m} \cdot \text{s}^{-2}$ (blok *Mechanism Configuration*).



Obr. 2-20 Přenosová funkce prvního řádu s časovou konstantou $\tau = 0,1$ s při požadavku úhlu 100° .



Obr. 2-21 Testování modelu 3DOFp (simuluje požadavky na otočení hřídele krokového motoru).

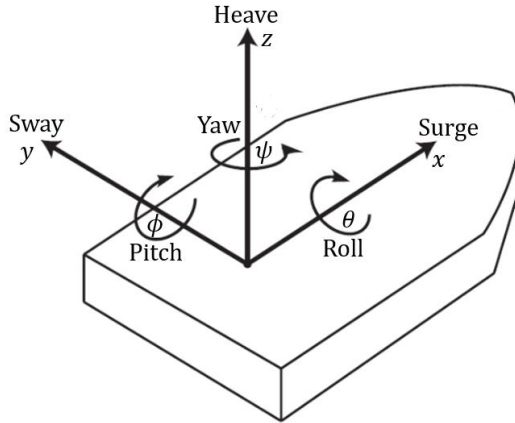


Obr. 2-22 Pohled do Mechanics Explorer při testování modelu 3DOFp.

KAPITOLA 3: INVERZNÍ KINEMATIKA A KOMPLETACE MODELU

3.1 Odvození rovnic inverzní kinematiky

V [1] je elegantní odvození inverzní kinematiky šestistupňové plošiny. Připomeňme si nyní použité matematické principy a odvození rovnic dle [2] a [101]. Je nutné definovat reprezentace Eulerových úhlů, proto je přiřazen ose x rotační úhel θ , ose y rotační úhel ϕ a ose z rotační úhel ψ . Volné těleso v prostoru má šest stupňů volnosti, kdy lze polohu souřadného systému tělesa popsat pomocí třech translačních a třech rotačních pohybů vůči pevně zvolenému souřadnému systému. V literatuře jsou rotačním pohybům kolem příslušných os přiřazeny názvy *Roll*, *Pitch*, *Yaw* (viz obr. 3-1).



Obr. 3-1 Popis šesti stupňů volnosti, tři translační pohyby x, y, z a tři rotační θ, ϕ, ψ [14] (upraveno).

Rotační matice pro jednotlivé osy lze pak reprezentovat ve tvaru 3×3 , ale při spojení translační a rotační matice přechází matice do tvaru 4×4

$$\begin{pmatrix} x'_z \\ y'_z \\ z'_z \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \psi & -\sin \psi & 0 & x_t \\ \sin \psi & \cos \psi & 0 & y_t \\ 0 & 0 & 1 & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_z \\ y_z \\ z_z \\ 1 \end{pmatrix} = \mathbf{R}_z(\psi, x_t, y_t, z_t) \begin{pmatrix} x_z \\ y_z \\ z_z \\ 1 \end{pmatrix}, \quad (3-1)$$

$$\begin{pmatrix} x'_y \\ y'_y \\ z'_y \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \phi & 0 & \sin \phi & x_t \\ 0 & 1 & 0 & y_t \\ -\sin \phi & 0 & \cos \phi & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_y \\ y_y \\ z_y \\ 1 \end{pmatrix} = \mathbf{R}_y(\phi, x_t, y_t, z_t) \begin{pmatrix} x_y \\ y_y \\ z_y \\ 1 \end{pmatrix}, \quad (3-2)$$

$$\begin{pmatrix} x'_x \\ y'_x \\ z'_x \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & x_t \\ 0 & \cos \theta & -\sin \theta & y_t \\ 0 & \sin \theta & \cos \theta & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_x \\ y_x \\ z_x \\ 1 \end{pmatrix} = \mathbf{R}_x(\theta, x_t, y_t, z_t) \begin{pmatrix} x_x \\ y_x \\ z_x \\ 1 \end{pmatrix}. \quad (3-3)$$

Při násobení rotací mezi sebou lze využít 12 sekvencí (tři rotace za sebou) pro definici Eulerových úhlů. Použitá sekvence ψ, ϕ, θ třech rotací a translace dá \mathbf{R} ve tvaru

$$\begin{pmatrix} \cos \psi & -\sin \psi & 0 & x_t \\ \sin \psi & \cos \psi & 0 & y_t \\ 0 & 0 & 1 & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \phi & 0 & \sin \phi & x_t \\ 0 & 1 & 0 & y_t \\ -\sin \phi & 0 & \cos \phi & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & x_t \\ 0 & \cos \theta & -\sin \theta & y_t \\ 0 & \sin \theta & \cos \theta & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (3-4)$$

Získaná transformace umožňuje aplikování rotační transformace s translací na libovolný bod $(x \ y \ z \ 1)^T$ pro transformaci do libovolného bodu $(x' \ y' \ z' \ 1)^T$. Platí tedy

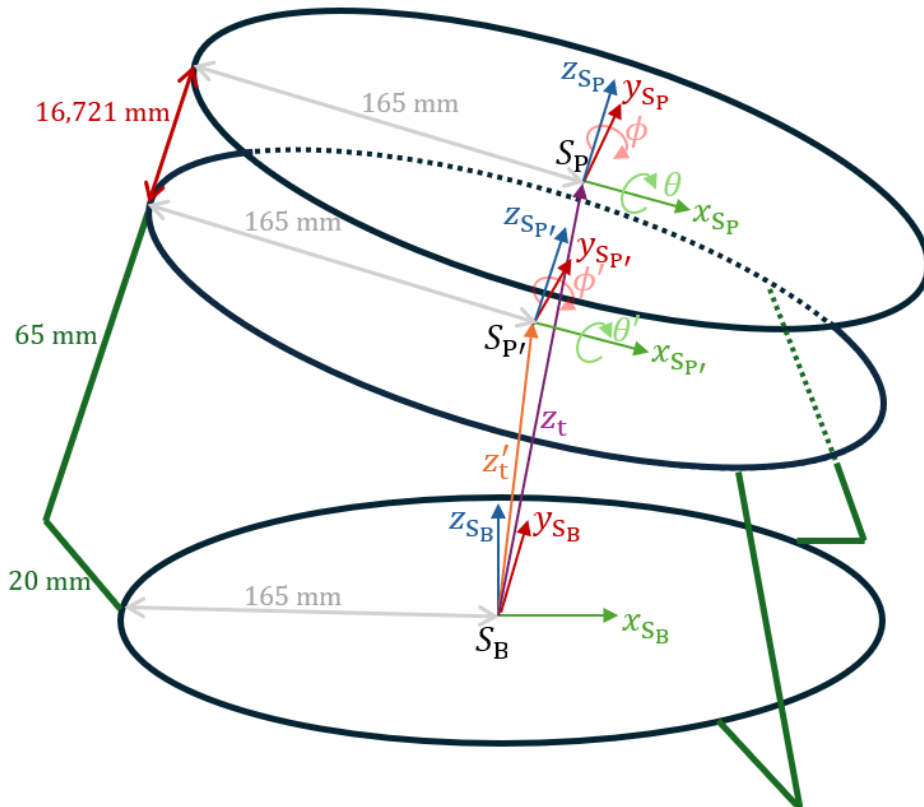
$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \mathbf{R}(\psi, \phi, \theta, x_t, y_t, z_t) \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \mathbf{R} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}. \quad (3-5)$$

Po postupném vynásobení matic zleva je pak výsledný tvar matice \mathbf{R}

$$\begin{pmatrix} \cos \psi \cdot \cos \phi & \cos \psi \cdot \sin \phi \cdot \sin \theta - \cos \theta \cdot \sin \psi & \cos \psi \cdot \sin \phi \cdot \cos \theta + \sin \theta \cdot \sin \psi & x_t \\ \sin \psi \cdot \cos \phi & \sin \psi \cdot \sin \phi \cdot \sin \theta + \cos \theta \cdot \cos \psi & \sin \psi \cdot \sin \phi \cdot \cos \theta - \sin \theta \cdot \cos \psi & y_t \\ -\sin \phi & \cos \phi \cdot \sin \theta & \cos \phi \cdot \cos \theta & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (3-6)$$

Pro plošinu o šesti stupních volnosti by bylo využito rotací kolem třech os a translací ve směru třech os, pro plošinu omezenou třemi stupni volnosti je konstrukcí potřeba volit jaké budou stupně volnosti. Pro uvažovanou 3DOFp zůstává uvažována jen translace z_t v ose z , rotace θ kolem osy x a rotace ϕ kolem osy y . V rovnici (3-6) lze dosadit za $\psi = 0$, $x_t = 0$ a $y_t = 0$ a vzniká redukovaná matice ve tvaru

$$\mathbf{R}(\phi, \theta, z_t) = \begin{pmatrix} \cos \phi & \sin \phi \cdot \sin \theta & \sin \phi \cdot \cos \theta & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ -\sin \phi & \cos \phi \cdot \sin \theta & \cos \phi \cdot \cos \theta & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (3-7)$$



Obr. 3-2 Popis souřadnicových systému s translací z_t (z_t') a rotacemi θ (θ') a ϕ (ϕ').

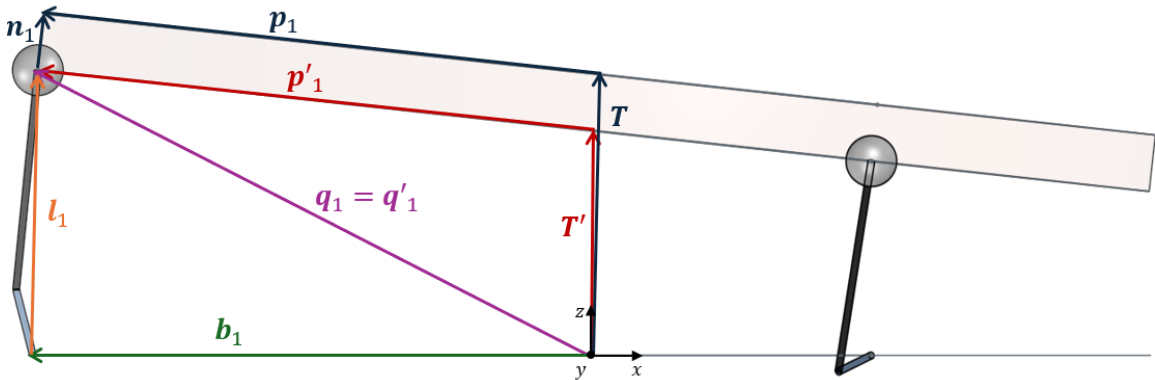
Pro výpočet inverzní kinematiky je nejdříve potřeba popsat jednotlivé body na plošině o třech stupních volnosti viz obr. 3-4. Rovině, v které jsou hřídele motorů je přiřazen název B jako *base*, rovině vrchní strany dotykového panelu P jako *platform* a rovině středů kulových čepů P'. Bázové souřadnice jsou uvažovány v S_B . Vůči soustavě S_B je uvažována rotace a translace platformy popsána vztahem (3-7) do bodů S_p , resp. S'_p viz obr. 3-2. Body P_i ($i = 1; 2; 3$) lze pak výhodně popisovat v souřadnicovém systému platformy S_p , kdy body P_i jsou ve společné rovině x_{S_p}, y_{S_p} (pro body P'_i lze uvažovat jen posun v z_{S_p} viz vektor \mathbf{n}_i z obr. 3-3), a pak transformovat vůči S_B .

Dle obr. 3-3 lze odvodit délky nohou skládáním vektorů pro tři nohy $i = 1; 2; 3$ při zanedbání tloušťky horní platformy.

$$\mathbf{q}'_i = \mathbf{T}'(z'_t) + \mathbf{R}(\phi', \theta')\mathbf{p}'_i = \mathbf{R}(\phi', \theta', z'_t)\mathbf{p}'_i \quad (i = 1; 2; 3), \quad (3-8)$$

$$\mathbf{l}_i = \mathbf{q}'_i - \mathbf{b}_i = \mathbf{T}'(z'_t) + \mathbf{R}(\phi', \theta')\mathbf{p}'_i - \mathbf{b}_i = \mathbf{R}(\phi', \theta', z'_t)\mathbf{p}'_i - \mathbf{b}_i \quad (i = 1; 2; 3). \quad (3-9)$$

Z rovnic (3-8) a (3-9) a obr. 3-2 je zřejmé, že vektory \mathbf{p}'_i jsou popisovány vůči souřadnému systému S_p . Pro transformaci vůči souřadnému S_B je potřeba zleva vynásobit vektory \mathbf{p}'_i maticí translace a rotace $\mathbf{R}(\phi', \theta', z'_t)$.



Obr. 3-3 Pomocná vizualizace pro skládání vektorů pro určení vektoru délky \mathbf{l} .

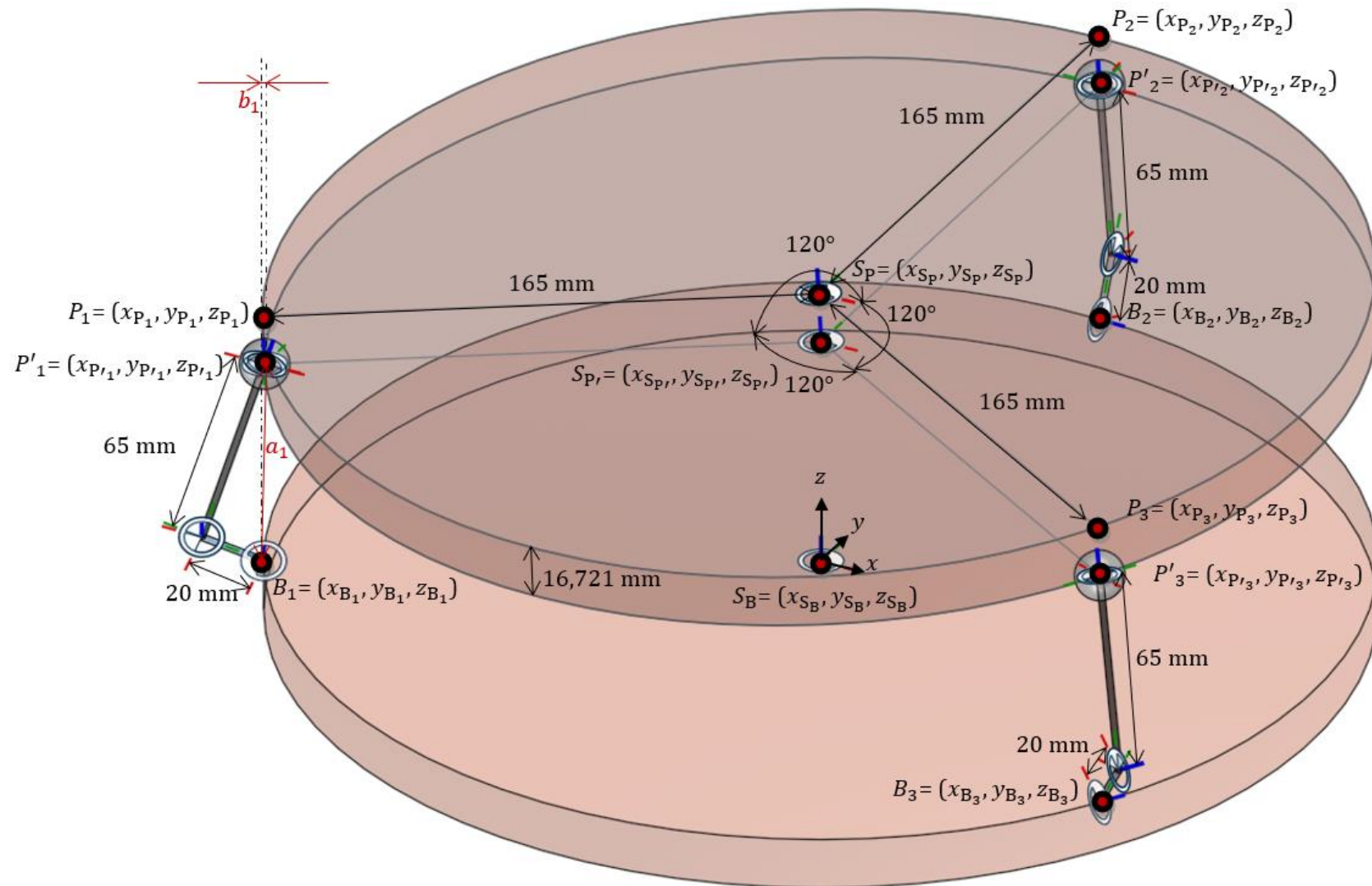
Při uvažování horní platformy jsou vztahy upraveny na

$$\mathbf{q}'_i = \mathbf{T}(z_t) + \mathbf{R}(\phi, \theta)(\mathbf{p}_i - \mathbf{n}_i) = \mathbf{R}(\phi, \theta, z_t)(\mathbf{p}_i - \mathbf{n}_i) \quad (i = 1; 2; 3), \quad (3-10)$$

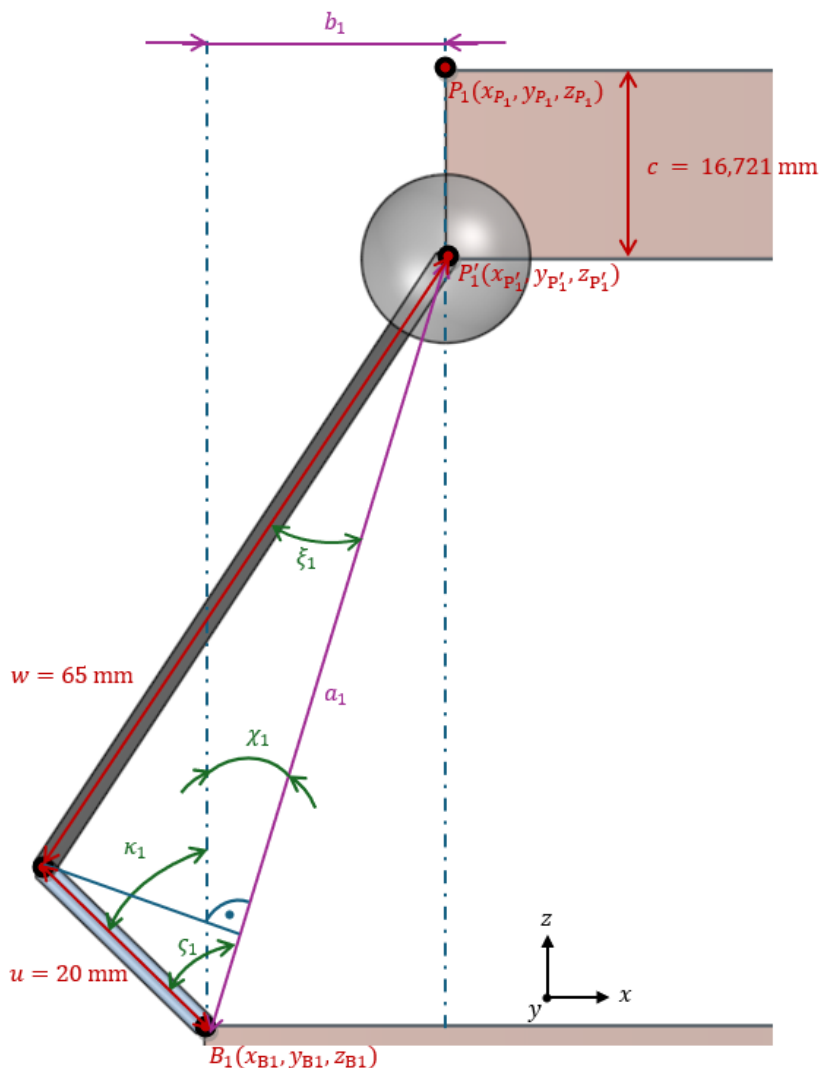
$$\mathbf{l}_i = \mathbf{q}'_i - \mathbf{b}_i = \mathbf{T}(z_t) + \mathbf{R}(\phi, \theta)(\mathbf{p}_i - \mathbf{n}_i) - \mathbf{b}_i = \mathbf{R}(\phi, \theta, z_t)(\mathbf{p}_i - \mathbf{n}_i) - \mathbf{b}_i \quad (i = 1; 2; 3). \quad (3-11)$$

Vektor \mathbf{p}_i je obdobně definován v souřadném systému S_p , stejně tak vektor \mathbf{n}_i , který se projeví jako dodatečná translace v z_p o vzdálenost $c = 16,721$ mm. Výška středu plošiny S_p vůči S_p , je rozdílně orientována translačními maticemi $\mathbf{T}(z_t)$ oproti $\mathbf{T}'(z'_t)$. Je důležité podotknout, že pro nakloněnou platformu není translace jen v ose z , ale i v osách x a y viz $\mathbf{T}(x_t, y_t, z_t)$ v obr. 3-3. Posuny v ostatních osách jsou však malé, a proto jsou zanedbávány a pro určení translace je jen zadáván posun v ose z viz model inverzní kinematiky v obr. 3-7.

Z vektoru \mathbf{l}_i se už lze dopočítat na příslušné úhly otočení hřídelí motorů vůči svislé ose z , κ_i ($i = 1; 2; 3$). Tato úprava však již nemění délku a směr vektorů \mathbf{l}_i , tedy pro 3-RRS plošinu platí stejné odvození inverzní kinematiky jako pro 3-RPS plošinu. Odlišení je však u implementace délek noh, protože u prismatických kloubů je často uvažováno využití lineárních motorů, či pneumaticky poháněných pístů pro nastavení délky noh \mathbf{l}_i přímo lineárním pohybem, zatímco u rotačních kloubů je potřeba nastavit správný úhel kliky pro dosažení vektoru \mathbf{l}_i viz obr. 3-5.



Obr. 3-4 Popis bodů a rozměrů na redukovaném modelu 3DOFp.



Obr. 3-5 Pomocná vizualizace pro výpočet žádaného úhlu otočení hřídelí motorů κ_i vůči svislé ose.

Pro plošinu 3-RRS je potřeba výpočet úhlu otočení kliky κ_i přes následující odvození vycházející z obr. 3-5. Euklidovskou vzdálenost a_i lze určit jako

$$a_i = \|P_i' - B_i\| = \sqrt{(x_{P_i'} - x_{B_i})^2 + (y_{P_i'} - y_{B_i})^2 + (z_{P_i'} - z_{B_i})^2} \quad (i = 1; 2; 3). \quad (3-12)$$

Vzdálenost b_i jen pro souřadnice x, y lze určit obdobně

$$b_i = \|P_i' - B_i\|_{x,y} = \sqrt{(x_{P_i'} - x_{B_i})^2 + (y_{P_i'} - y_{B_i})^2} \quad (i = 1; 2; 3). \quad (3-13)$$

Vzdálenost a_i lze rozdělit na dva pravoúhlé trojúhelníky (viz obr. 3-5), z Pythagorovy věty platí

$$a_i = a_{iw} + a_{iu} = w \cdot \cos \xi_i + u \cdot \cos \zeta_i \quad (i = 1; 2; 3). \quad (3-14)$$

Pro protilehlou odvěsnu společnou pro oba pravoúhlé trojúhelníky lze psát

$$w \cdot \sin \xi_i = u \cdot \sin \zeta_i \quad (i = 1; 2; 3). \quad (3-15)$$

Při využití trigonometrické identity

$$\sin^2 \xi_i + \cos^2 \xi_i = 1 \rightarrow \sin \xi_i = \sqrt{1 - \cos^2 \xi_i} \quad (i = 1; 2; 3), \quad (3-16)$$

lze vyjádřit z (3-15) a dosadit z (3-16)

$$\sin \xi_i = \frac{u \cdot \sin \zeta_i}{w} = \sqrt{1 - \cos^2 \xi_i} \rightarrow \cos \xi_i = \sqrt{1 - \left(\frac{u \cdot \sin \zeta_i}{w}\right)^2} \quad (i = 1; 2; 3). \quad (3-17)$$

Po úpravě (3-17) lze dosadit do (3-14)

$$a_i = w \cdot \sqrt{1 - \left(\frac{u \cdot \sin \zeta_i}{w}\right)^2} + u \cdot \cos \zeta_i \quad (i = 1; 2; 3). \quad (3-18)$$

Z druhého členu (3-18) je rovnice vydělena délkou kliky u a společně s délkou táhla do středu kulového čepu w je vtažen poměr do odmocniny pomocí operace umocnění w^2/u^2

$$\frac{a_i}{u} = \sqrt{\frac{w^2}{u^2} - \frac{w^2}{u^2} \cdot \frac{u^2 \cdot \sin^2 \zeta_i}{w^2}} + \cos \zeta_i = \sqrt{\frac{w^2}{u^2} - \sin^2 \zeta_i} + \cos \zeta_i \quad (i = 1; 2; 3). \quad (3-19)$$

Při odizolování členu odmocniny odečtením $\cos \zeta$ v (3-19) je získáno

$$\sqrt{\frac{w^2}{u^2} - \sin^2 \zeta_i} = \frac{a_i}{u} - \cos \zeta_i \quad (i = 1; 2; 3), \quad (3-20)$$

lze pak umocnit (3-20)

$$\frac{w^2}{u^2} - \sin^2 \zeta_i = \frac{a_i^2}{u^2} - 2 \cdot \frac{a_i}{u} \cdot \cos \zeta_i + \cos^2 \zeta_i \quad (i = 1; 2; 3). \quad (3-21)$$

Z (3-21) lze vyjádřit člen s $\cos \zeta$ a využít trigonometrickou identitu stejně jako v (3-16)

$$2 \cdot \frac{a_i}{u} \cdot \cos \zeta_i = \cos^2 \zeta_i + \sin^2 \zeta_i + \frac{a_i^2}{u^2} - \frac{w^2}{u^2} = 1 + \frac{a_i^2}{u^2} - \frac{w^2}{u^2} \quad (i = 1; 2; 3), \quad (3-22)$$

a pak při osamocení $\cos \zeta$ lze psát

$$\cos \zeta_i = \frac{u}{2 \cdot a_i} \cdot \left(1 + \frac{a_i^2}{u^2} - \frac{w^2}{u^2}\right) = \frac{u}{2 \cdot a_i} \cdot \left(\frac{u^2 + a_i^2 - w^2}{u^2}\right) = \frac{u^2 + a_i^2 - w^2}{2 \cdot a_i \cdot u} \quad (i = 1; 2; 3), \quad (3-23)$$

kdy po dosazení za a_i z (3-12) platí vyjádření ζ_i z (3-23)

$$\zeta_i = \arccos \left(\frac{u^2 + (x_{P_i} - x_{B_i})^2 + (y_{P_i} - y_{B_i})^2 + (z_{P_i} - z_{B_i})^2 - w^2}{2 \cdot \sqrt{(x_{P_i} - x_{B_i})^2 + (y_{P_i} - y_{B_i})^2 + (z_{P_i} - z_{B_i})^2} \cdot u} \right) \quad (i = 1; 2; 3). \quad (3-24)$$

Pomocí a_i lze vyjádřit

$$\sin \chi_i = \frac{b_i}{a_i} \quad (i = 1; 2; 3), \quad (3-25)$$

kdy χ_i lze vyjádřit z (3-25) po dosazení (3-12) a (3-13)

$$\chi_i = \arcsin \left(\frac{\sqrt{(x_{P_i} - x_{B_i})^2 + (y_{P_i} - y_{B_i})^2}}{\sqrt{(x_{P_i} - x_{B_i})^2 + (y_{P_i} - y_{B_i})^2 + (z_{P_i} - z_{B_i})^2}} \right) \quad (i = 1; 2; 3). \quad (3-26)$$

Úhel otočení hřídele vůči svislé ose z je dán vztahem

$$\kappa_i = \zeta_i - \chi_i \quad (i = 1; 2; 3), \quad (3-27)$$

tedy po dosazení z (3-25) a (3-26) do (3-27) byl ukázán postup, jak se přepočítat z l_i na κ_i pro odvození kompletní inverzní kinematiky, která umožňuje přepočet žádané translace v ose z a rotací θ , ϕ na úhel klik motorů κ_i pro dosažení těchto žádaných veličin.

Jiná možná odvození inverzní kinematiky 3DOFp jsou popsána v [34], [46], [61], [91] - [97], kdy je však často uvažován střed plošiny ve společné rovině jako středy sférických kloubů a je zanedbána dodatečná translace v ose z . Pro zkoumání pracovní oblasti je často uvedena v člancích i dopředná kinematika, která není uvedena (má nejednoznačné řešení a je výpočetně komplikovanější a pro tuto práci nebyla využita). Analogická odvození jsou v [98] - [100] pro Stewartovu platformu.

3.2 Otestování inverzní kinematiky v prostředí *Simulink*

Před otestováním inverzní kinematiky, je potřeba nejdříve implementace teoretického odvození do prostředí *Simulink*. Před vytvořením modelu je vhodné obecné parametry přiřadit do matlab funkce (m.file), kdy v samotném modelu (.slx) jsou příslušné funkce volány názvem bez koncového označení přílohy (.m). Používané proměnné jsou pak v následujícím kódu.

```
% 3RRS paralelní manipulátor parametry
T_sim= 0.01; %simulační perioda 10 ms
mm = 0.001; %převod z mm na m
%tři nohy, i = 1,2,3
ps = 330*mm/2; %m vzdálenost od středu S_P do bodů P_i, vršek displeje
pcarka_s = ps; %m vzdálenost od středu S_P' do bodů P'_i, kulové čepy
bs = ps; %m vzdálenost od středu S_B do bodů B_i, hřídele
u = 20*mm; %m délka kliky pro všechny nohy stejná
w = 65*mm; %m délka táhla do středu kuličky na čepu
c = 16.721*mm; %m vzdálenost od P_i do P'_i , vzdálenost ...
% středu kuličky na čepu do spodku ocelové kuličky na rezistivním
displeji

% Inverzní kinematika
psi = deg2rad(120); %rad (zadáno ve °) rotační úhel kolem osy z ...
% vzájemných bodů P_1, P_2, P_3
R_z = [cos(psi) -sin(psi) 0 0;... %rotační matice pro rotaci bodů P_i ...
sin(psi) cos(psi) 0 0;... %kolem osy z o úhel psi
0 0 1 0;...
0 0 0 1];
P1 = [ps;... %m souřadnice bodu P1 v vůči S_P
0;...
0;...
1];
```

```

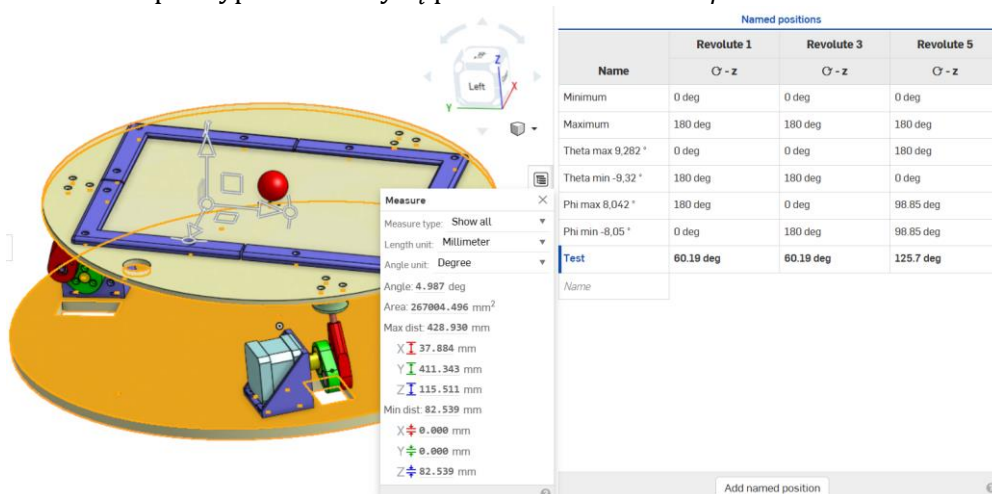
P = [P1 R_z*P1 R_z*R_z*P1]; %m souřadnice matice 4x3 P = [P_1 P_2 P_3]
vůči S_P
C = [0 0 0; ... %m souřadnice matice 4x3 P' = P - C vůči S_P'
     0 0 0; ...
     c c c; ...
     0 0 0];
B1 = [bs; ... %m souřadnice bodu B1 v vůči S_B
      0; ...
      0; ...
      1];
B = [B1 R_z*B1 R_z*R_z*B1]; %m souřadnice matice 4x3 B = [B_1 B_2 B_3]
vůči S_B
    
```

V proměnných je pojmenování a číslování bodů dle obr. 3-4, kdy je využito symetrie kolem osy z pro popis bodů vůči příslušným souřadným soustavám. V modelu bylo postupováno dle odvozeného vztahu (3-7) pro transformaci přes dvě rotace (θ, ϕ) a jednu translaci (z_t) ze souřadného systému S_B do S_P . Využity jsou operace maticového odčítání a násobení pro dosažení výsledné matice obsahující polohu bodů $P'_1 - B_1$, $P'_2 - B_2$ a $P'_3 - B_3$ vůči zvolenému souřadnicovému systému S_B .

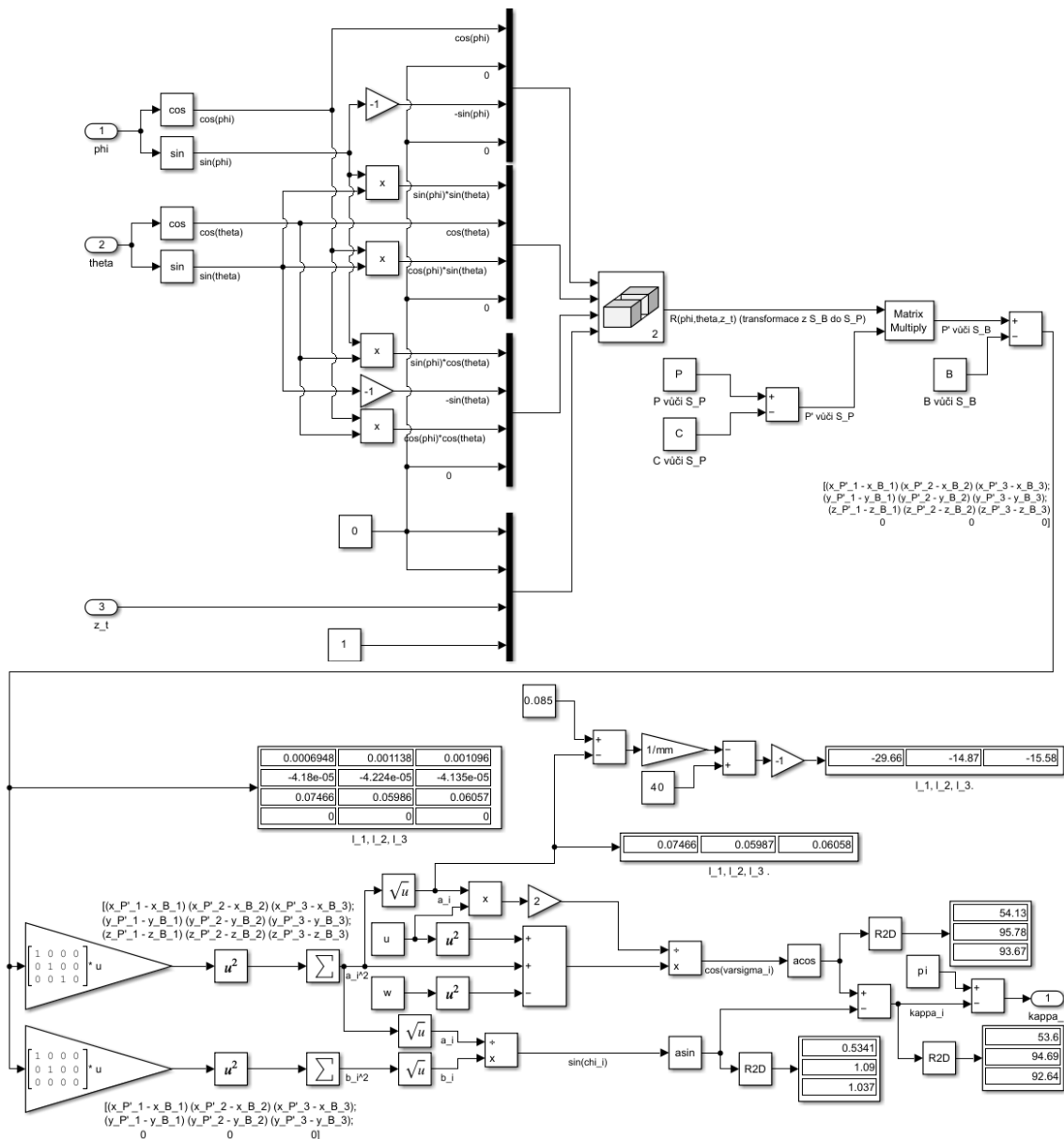
Další převod využívá vztahů (3-24), (3-26) a (3-27) pro transformaci úhlů náklonů na úhly otočení klik ve svislém směru. Protože rotační klouby jsou orientovány směrem dolů pomyslně v ose $-z$ oproti orientaci výpočtu úhlů v kladném směru osy $+z$, je potřeba upravit výsledné úhly operací $\kappa'_i = \pi - \kappa_i$ pro úhly v jednotkách radiánů (pro stupně by obdobně bylo $\kappa'_i = 180 - \kappa_i$). Obecně je pro práci v úhlech využita jako jednotka radián, kdy pro vizualizaci hodnot je preferenční zase používat úhly ve stupních, proto je používán blok *R2D* a *D2R* (*degrees to radians* a *radians to degrees*) umožňující plynulý přechod mezi jednotkami úhlů.

Výsledný model inverzní kinematiky je možné sestavit jako v obr. 3-7 s vnějšími vstupy a výstupy dle obr. 3-8. Polohování plošiny v ose z je znormováno od minimální polohy (přiřazena hodnota 0 mm) do maximální hodnoty (přiřazena hodnota 40 mm).

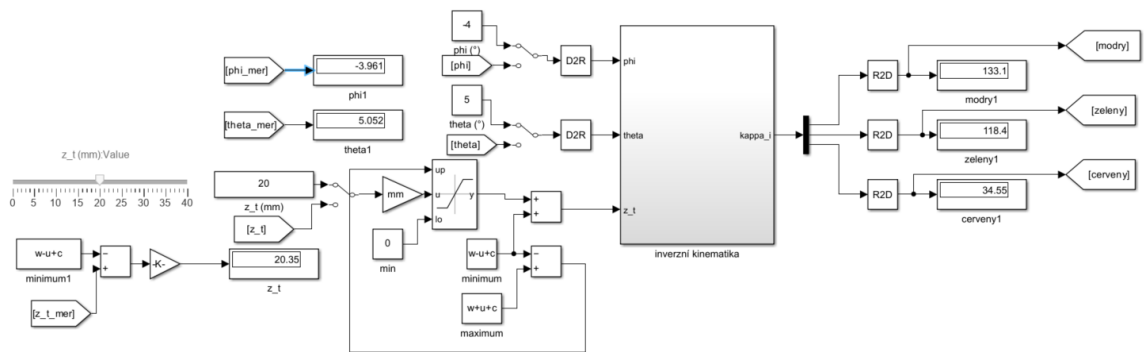
V prostředí *Simulink* je možné provést úpravu dle obr. 2-18, kdy je potřeba provést transformačním blokem translaci v ose z , aby soustava souřadnic byla přivedena do roviny ve výšce středů hřídelí. Pomocí bloku *Transform Sensor* je umožněno přímo měřit vzájemné rotace a translace různých soustav souřadnic. Ve výšce středu hřídelí je rovina v jejímž středu je první měřená soustava souřadnic, vůči které jsou měřené rotační úhly ϕ , θ a měřená translace z_t ze středu roviny dotykového panelu. Signály jsou pak zobrazeny u zadávaných hodnot v obr. 3-8. Pro ověření vypočtených hodnot úhlů κ_i , lze úhly ručně zadat do modelu 3DOFp v *Onshape* jako je demonstrováno pro vypočtené úhly κ_i při zadaném náklonu $\phi = 5^\circ$ viz obr. 3-6.



Obr. 3-6 Testování modelu 3DOFp pro kontrolu inverzní kinematiky při zadaném náklonu $\phi = 5^\circ$.



Obr. 3-7 Model inverzní kinematiky v Simulink.

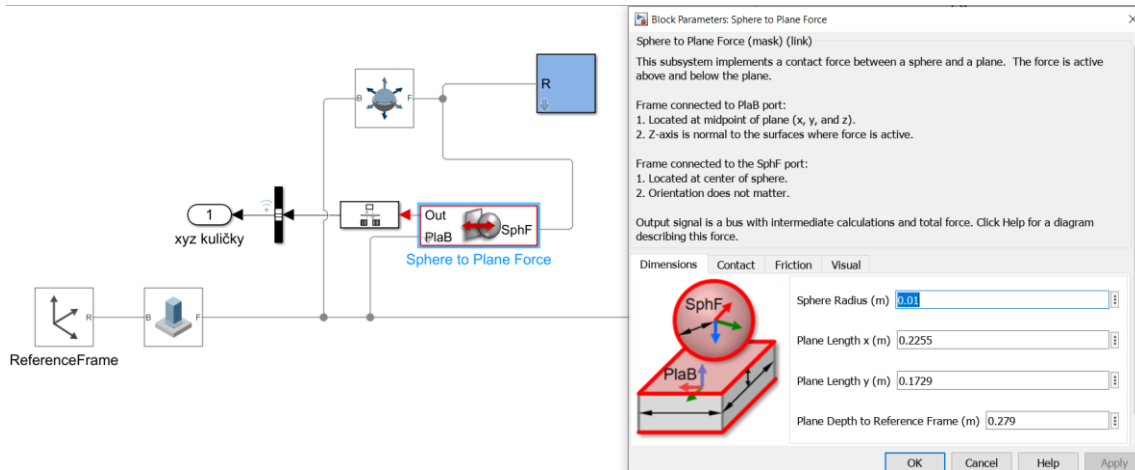


Obr. 3-8 Vnější rozhraní pro model inverzní kinematiky v Simulink.

3.3 Kontakt kulička rezistivní deska

Způsobů pro modelování kontaktu kuličky na rezistivním dotykovém panelu je mnoho. V úvodních pokusech této práce byl využit zjednodušený model využívající tíhové a odstředivé síly pro výpočet pohybu kuličky přes moment setrvačnosti. Zjednodušený model chování kuličky však uvažoval jen translační pohyb a pro reprezentaci fyzického modelu byl zavrhnut jako nedostatečný, protože bylo žádoucí uvažovat valivý pohyb kuličky. Vytvořit takový model pohybu kuličky na plošině je komplikované, ale díky dostupným zdrojům, lze v prostředí *Simulink* využít jiných způsobů než tvorbu vlastního modelu.

Jednou z možností je použití dostupné knihovny ze [103], která přímo obsahuje kontaktní blok, ve kterém jsou nastavovány příslušné součinitele kinematického a statického tření s kontaktní tuhostí a tlumením.



Obr. 3-9 Testování kontaktu Sphere to Plane ze [103].

Jiná možnost volená pro tuto práci byla již zobrazena v obr. 2-18 a jedná se o implementaci, která byla inspirována ze [102]. Postup tvorby kontaktního propojení je přímočarý, přes blok *Spatial Contact Force* přímo z knihovny *Simscape Multibody*. Ocelová kulička volená pro simulaci je o průměru 34 mm (byla zvážena na 161 g dle obr. C-2). Rezistivní dotykový panel byl importován dle příslušných parametrů z modelu v *Onshape*, kdy voleným materiálem pro jeho reprezentaci je sklo, i když se jedná o určité zjednodušení jeho materiálové struktury.

Pro reprezentaci pohybu kuličky v prostoru lze využít kloub o šesti stupních volnosti (*6-DOF Joint*), kdy pro simulaci byl pohyb omezen v rámci upravených rozměrů panelu $\pm 118,5 \text{ mm} \times \pm 87 \text{ mm}$, tedy v simulaci není uvažováno spadnutí kuličky z plošiny.

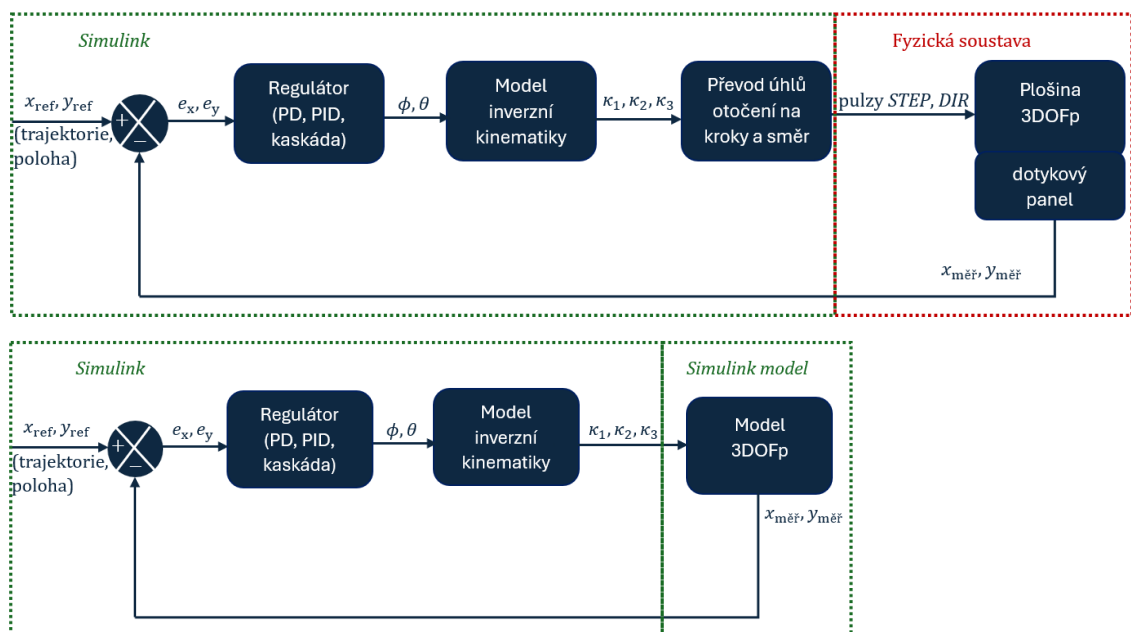
Simulace pádu kuličky na plochu vyvolá penetraci panelu a následný odraz kuličky [107]. Hertzův kontaktní model uvažuje elasticity různých materiálů a umožňuje výpočet sil mezi kuličkou a plochou a lze použít pro výpočet rozložení tlaku při kontaktu bodu na kuličce s plochou [105]. Pro představení elastického dopadu ocelové kuličky na skleněnou plochu je vytvořen model ve [106], kde byly zkoumány různé průměry kuličky a tloušťky desek pro zjištění kontaktní doby dle Hertzova modelu. Kontaktní doba je měřený čas od začátku penetrace panelu do odrazu od panelu a při každém odrazu je již nějaká energie disipována, proto se kontaktní doba prodlužuje až do ustálení na stálém kontaktu. Vlivem na kontaktní dobu má rychlost dopadu a poměr rozměrů kuličky k rozměrům skleněné desky. Chování kontaktu lze reprezentovat pružinovým modelem, kdy při známé hmotnosti jsou zkoumanými parametry tuhosti a tlumení [104]. Pro tenisový míček puštěný na zem bude poměrová odrazová výška jiná, než pro ocelovou kuličku puštěnou na skleněnou desku a doba ustálení bude také jiná. Právě tento vliv je nutné modelovat parametry tuhosti a tlumení. Pro parametr tuhosti lze využít Hertzova kontaktního modelu, ale výpočty by byly komplikované. Validní metodou, jak parametry stanovit je sledování chování

kontaktu v modelu a porovnání s reálně pozorovanými výsledky. Experimentálně proto byla puštěna ocelová kulička na nepoužívané zrcadlo pro sledování chování vzájemného kontaktu.

Sklo a ocel jsou materiály s vyšší tuhostí, a proto lze volit vyšší hodnoty tuhosti kolem řádu 10^6 Nm^{-1} . Tlumení reprezentuje disipaci energie při kontaktu, kdy častou volbou je 1 % tuhosti, tedy hodnoty v řádu $10^4 \text{ Nm}^{-1}\text{s}^{-1}$. Pro pohyb kuličky na ploše dotykového panelu byl určen statický součinitel tření a kinematický (dynamický) součinitel tření. Statický součinitel tření vychází z poměru statické třecí síly vůči normálové síle a určuje sílu potřebnou k pohybu kuličky vůči ploše skla dotykového panelu bez valení. Jakmile je kulička již v pohybu, je vyžadovaná síla pro udržení pohybu určená kinematickým součinitelem tření. Pro utržení ze statického tření, je zpravidla potřeba větší síla než síla pro udržení tělesa ve valivém pohybu. Pro různé kontakty materiálů platí různé parametry tření. Při volbě bylo čerpáno z tabulky ve [108], kde pro kontakt mědi na sklo byl změřen parametr statického součinitele tření 0,68 a parametr dynamického součinitele tření 0,53. Volené parametry byly vloženy do bloku reprezentující kontaktní silové chování viz obr. 2-18. Při vyzkoušení chování kuličky v simulaci, je dopad kuličky na plochu panelu utlumenější než při testu puštění kuličky, ale ve fyzické soustavě je kulička jen na panel položena a ne puštěna, proto jsou zejména důležité volby parametrů statického a kinematického součinitele tření, aby simulované chování kontaktního modelu kuličky na plochu vhodně reprezentovalo chování reálné soustavy.

3.4 Regulační schéma

V obr. 3-10 je představené zjednodušené regulační schéma, které je využíváno pro řízení 3DOFp. Vstupující řídicí veličinou je žádaná poloha ($x_{\text{ref}}, y_{\text{ref}}$), která může být nulová pro testování reakce přípravku při skokové změně, případně může být vložen signál žádané trajektorie. Ve zpětné vazbě je regulovaná veličina ($x_{\text{měř}}, y_{\text{měř}}$), tedy informace o současné poloze kuličky pro fyzický systém obdržena z kontroléru dotykového panelu a pro simulaci z vazby 6 – *DOF Joint* v obr. 2-18. Regulační odchylka e_x, e_y vstupuje do vybraného regulátoru (testována PD, PID a kaskádní regulace), jehož výstupní akční veličinou je úhel náklonu platformy ϕ, θ . Pomocí bloku inverzní kinematiky je požadovaný úhel náklonu převeden na žádané úhly pootočení klik $\kappa_1, \kappa_2, \kappa_3$ pro dosažení požadovaných náklonů ϕ, θ . Pro model v prostředí *Simulink* je úhel pootočení již řídicí veličinou soustavy, pro fyzickou sestavu je nutný přepočít na *STEP* a *DIR* krokových motorů.

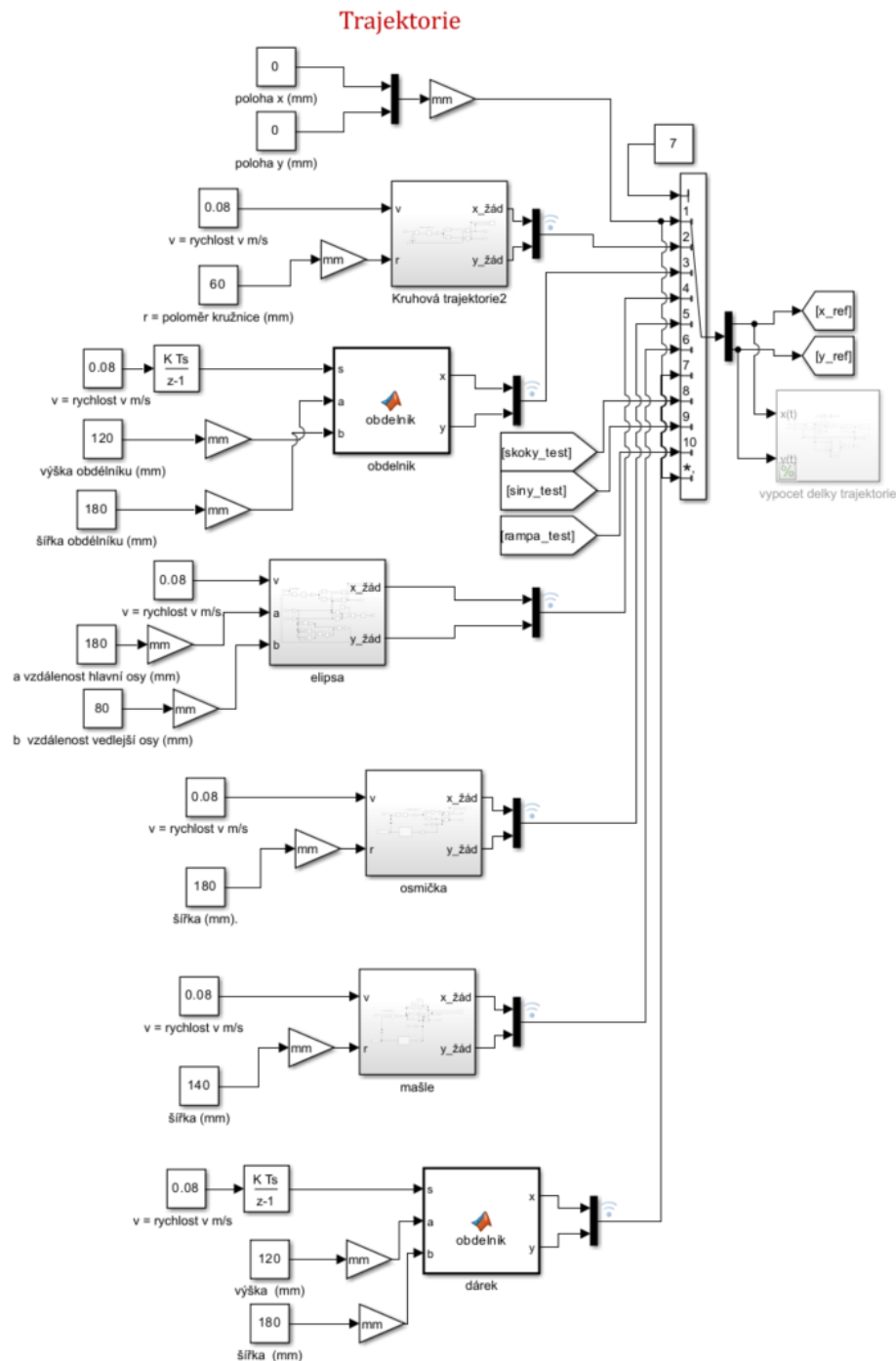


Obr. 3-10 Blokové schéma regulace fyzické soustavy ze Simulink a simulačního modelu v Simulink.

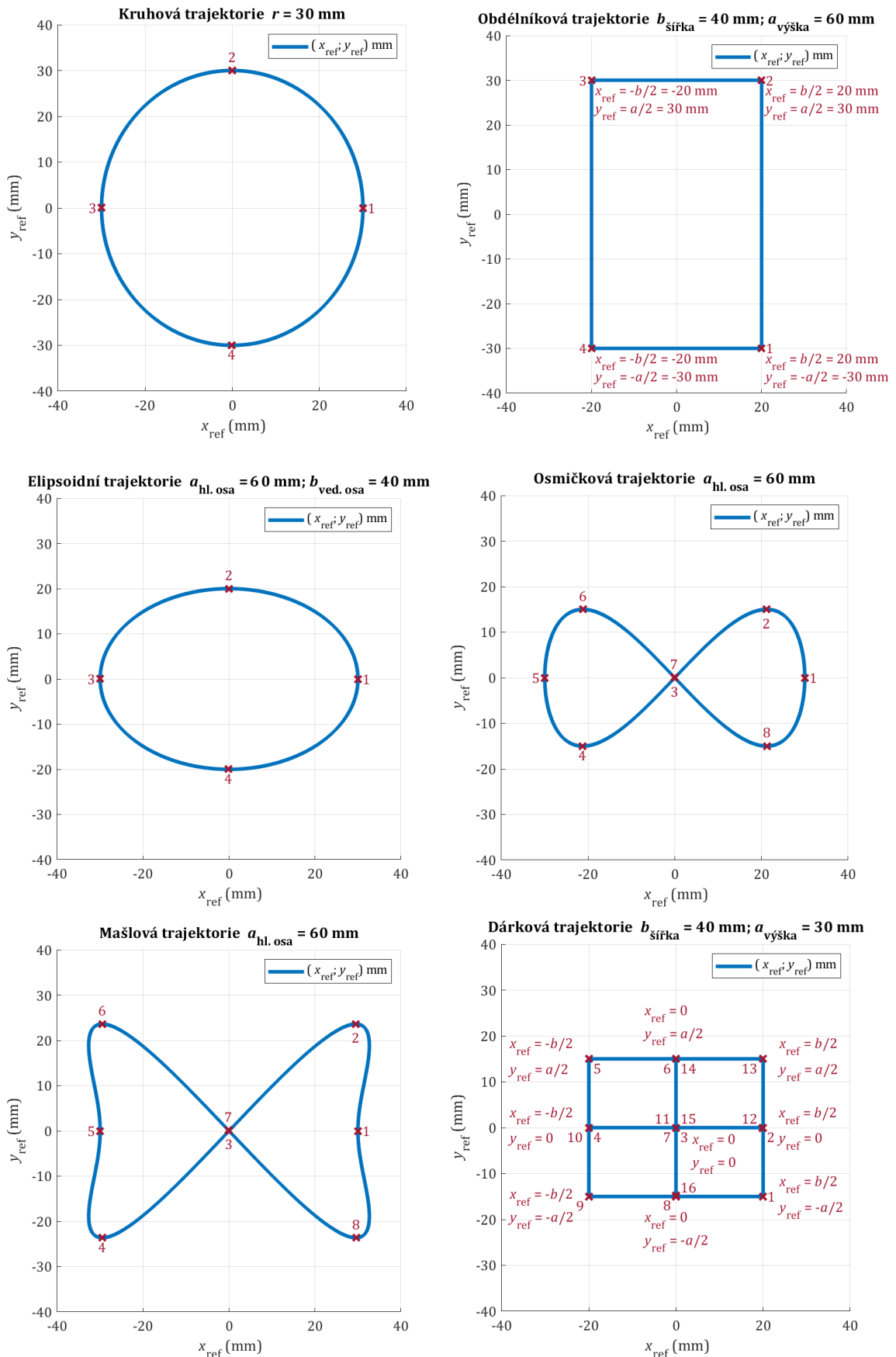
KAPITOLA 4: REGULAČNÍ METODY

4.1 Testovací trajektorie

Pro ověření funkce implementovaného řízení, je potřeba nejdříve vyladit základní funkci 3DOFp, a to polohování kuličky do žádaného bodu. Za tímto účelem stačí definovat bod žádané souřadnice $x_{\text{žad}}$, $y_{\text{žad}}$ a sledovat hodnotící parametry reakce polohy kuličky jako jsou např. překmit přes žádanou polohu, doba a dynamika k dosažení žádané polohy, trvalá odchylka od žádané polohy. Různé trajektorie testují jiné schopnosti řízení, a proto bylo vytvořeno šest průběhů ověřujících schopnosti regulátorů polohy kuličky viz obr. 4-1 a vykreslení trajektorií v obr. 4-2.



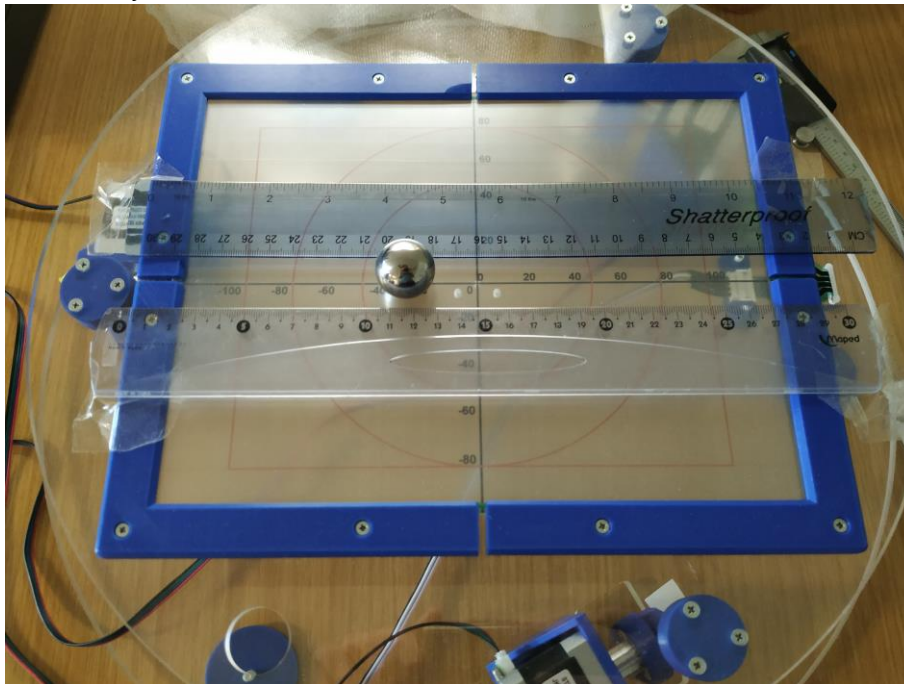
Obr. 4-1 Vytvořené trajektorie pro pohyb kuličky na plošině.



Obr. 4-2 Průběhy různých trajektorií pro polohování kuličky na plošině.

4.2 PD regulace

První využitá metoda regulace pomocí PD regulátoru je představena pro 3DOFp v [10], a pro 2DOFp ve [123]. Pro plošinu o třech stupních volnosti lze zadat pevně translaci $z_t = 20$ mm, a pak lze volně měnit rotační úhly ϕ kolem osy y a θ kolem osy x . Systém pak pomyslně nabývá chování 2DOFp, kdy lze dále soustavu zjednodušit omezením pohybu v jedné ose a ponecháním volnosti v ose měřené. V simulaci lze nastavit meze v bloku *6 - DOF Joint* v obr. 2-18 na hodnotu 0 a hodnotu blízké nule shora, např. 0,001 zafixováním pohybu ve vybrané ose a necháním volnosti v ose druhé, kde zůstává omezení rozměrů aktivní části dotykového panelu. Takto lze rozdělit pohyb v ose y a v ose x . Podobným mechanismem o jednom stupni volnosti je tzv. *Ball and Beam System*, kdy kulička je polohována v nosníku s U profilem a otáčením jednoho motoru je měněn náklon profilu viz [124]. Tohoto zjednodušení lze využít i v ladění fyzického systému při využití dvou pravítek pro omezení pohybu v příslušné ose. Úprava umožňuje pak ladit regulátor zvlášť v ose x a v ose y viz obr. 4-3.



Obr. 4-3 Omezení pohybu v ose y pro ladění regulátoru v ose x .

Po úpravě simulačního a fyzického modelu je přístup k ladění PD regulátoru podobný. Je sledována reakce systému na žádaný jednotkový skok a úpravou parametrů je snaha dosažení zvolených hodnotících kritérií chování regulovaného systému.

Při implementaci řízení do fyzické sestavy s mikroprocesorovou jednotkou, nelze již uvažovat spojitý systém, protože implementací časových konstant prvků se systém stává diskretním. Regulátor proto přechází ze spojitě formy na diskretní. Pro simulační model by bylo možné použít spojitý regulátor, ale protože snahou simulace je přiblížit vhodně model reálnému chování soustavy, je i zde využitý diskretní regulátor. Simulační perioda je volena dle periody čtení hodnot z panelu, kdy omezení je 10 ms, proto simulační doba regulátorů je nastavena na 0,01 s. Je možné využít různé modifikace rovnic PD regulátoru, kdy v bloku *PID Controller* v prostředí *Simulink* lze volit mezi ideální a paralelní formou rovnic. Odlišení je, že v ideální formě je člen zesílení (K_P) násoben až po sečtení složky $(1 + K_D)$, zatímco v paralelní formě je jen sčítání složek ($K_P + K_D$). Pro derivaci je využit filtrační člen K_N pro omezení vlivu rušení na derivační člen, který je citlivý na rychlé změny (signály vyšších frekvencí) a může způsobovat nežádoucí oscilace systému.

Spojitá forma bloku PD regulátoru ze *Simulink* v ideální formě je v (4-1), kdy spojitá paralelní forma pro oddělení vlivu proporcionální složky na derivační je v (4-2).

$$C_{PD,par.}(s) = K_P \cdot \left(1 + K_D \cdot \frac{K_N}{1 + K_N \cdot \frac{1}{s}} \right), \quad (4-1)$$

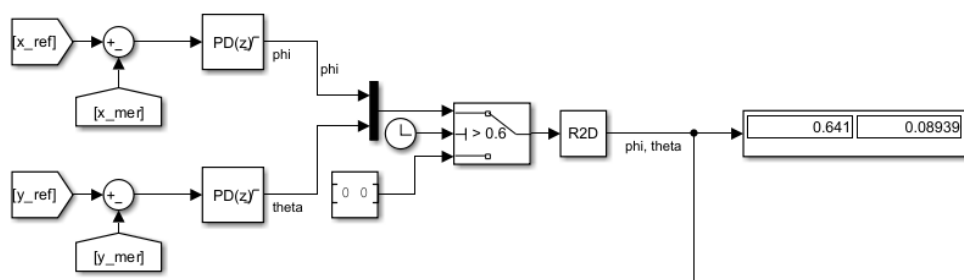
$$C_{PD,id.}(s) = K_P + K_D \cdot \frac{K_N}{1 + K_N \cdot \frac{1}{s}}. \quad (4-2)$$

Při přechodu ze spojité do diskrétní formy regulátoru, je potřeba nastavit simulační periodu ($T_s = 0,01$ s), zvolit filtrační metodu (volena lichoběžníková, *trapezoidal*) a saturovat výstupní hodnotu, kdy dle zvolené jednotky v radiánech je zvoleno omezení výstupu $\pm 5 \cdot 180/\pi$. Paralelní tvar diskrétního PD regulátoru s filtrací je pak ve tvaru:

$$C_{PD,id.}(z) = K_P + K_D \cdot \frac{K_N}{1 + K_N \cdot T_s \cdot \frac{1}{2} \cdot \frac{z+1}{z-1}}. \quad (4-3)$$

Nastavený blok PID v *Simulink* je pak se schématem řízení v obr. 4-4. Před dopadem kuličky není žádoucí mít regulaci ještě spuštěnou, proto je zpoždění 0,6 s do zapnutí.

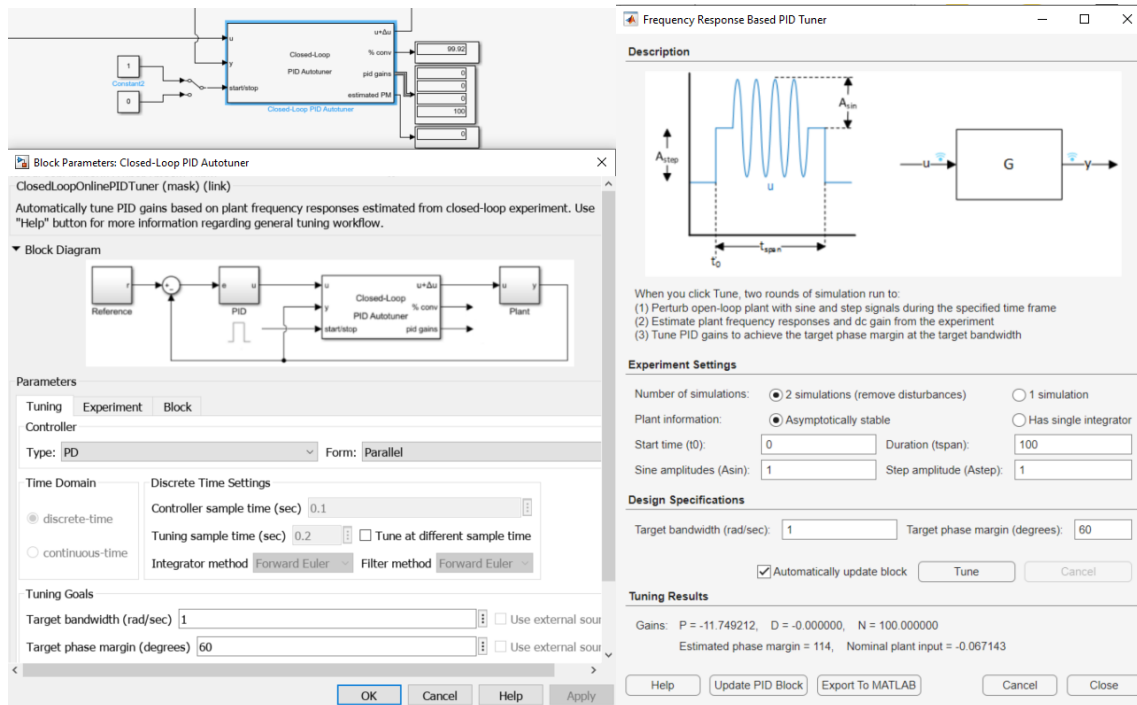
Řízení (PD regulátory)



Obr. 4-4 Schéma řízení PD regulátorů v *Simulink*.

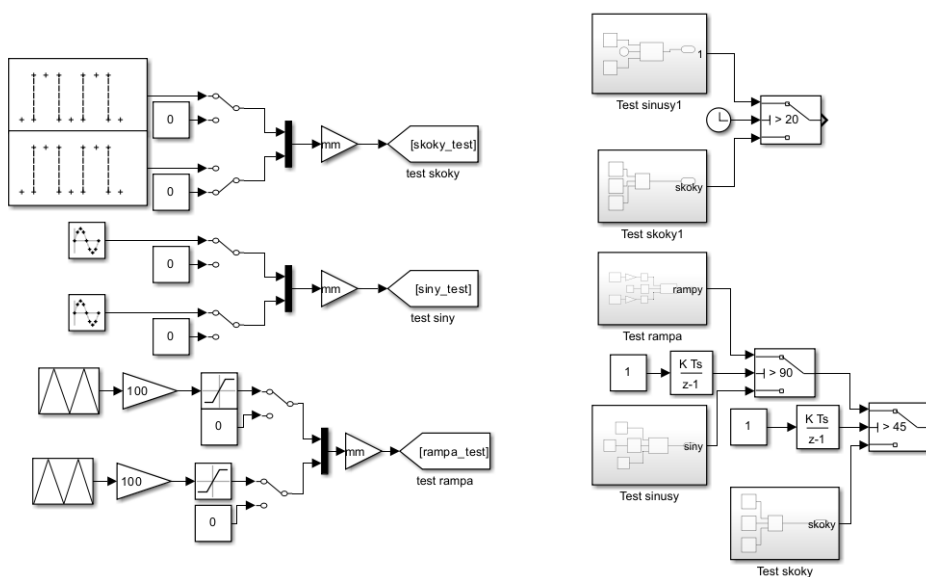
Ladících metod parametrů PD a PID regulátoru je mnoho. Často používaným přístupem je metoda Ziegler-Nichols. Přes změny parametru zesílení K_P při nastaveném $K_D = 0$ je reakce přivedena na mez oscilace, kdy je odečtena perioda mezi kmity a zesílení a dle tabulky jsou nastavené parametry jako ve [114]. Obdobné přístupy přes tabulkové určení parametrů jsou metody Cohen-Coon, Kappa-Tau, či Lambda. Další přístupy už vyžadují model procesu, jako metoda IMC nebo ladící algoritmy, které identifikují model a optimalizovanou funkcí dosáhnou žádaných parametrů. V prostředí *Simulink* lze použít funkci automatického ladění přímo v bloku (*PID controller*), kdy lze volit mezi přístupem přes přenosovou funkci (kde je systém linearizován a nelze použít pro systémy *Simscape Multibody*), či přes měřenou frekvenční odezvu. Pro fyzické systémy lze využít externí blok *Open/Closed - Loop PID Autotuner*, fungující na obdobném principu testování frekvenční odezvy pro automatické ladění regulátoru. Obě metody automatického ladění slouží pro identifikaci chování soustavy, kdy zpravidla obdržené konstanty jsou využity pro další manuální doladování žádané reakce. Bylo snahou použít zmíněné ladící metody z obr. 4-5, ale konvergované parametry z algoritmů přinášely systém do značné nestability a vytváření přenosové funkce modelu příliš zjednodušovalo chování soustavy.

Validní metodou pro ladění PD regulátoru je i manuální přístup, kdy změnou parametrů je sledována odezva systému, tedy reakce pohybu kuličky na žádaný skok, sinusový signál, či lichoběžníkový signál (viz vytvoření testovací funkce z obr. 4-6) a změnou parametrů je snaha tuto reakci dostat do žádaných průběhů. Tato metoda byla využita pro ladění PD regulátoru nejen v modelu, ale i ve fyzické sestavě.



Obr. 4-5 Blok Closed – Loop PID Autotuner a Frequency Response based PID tuner z bloku PID controller.

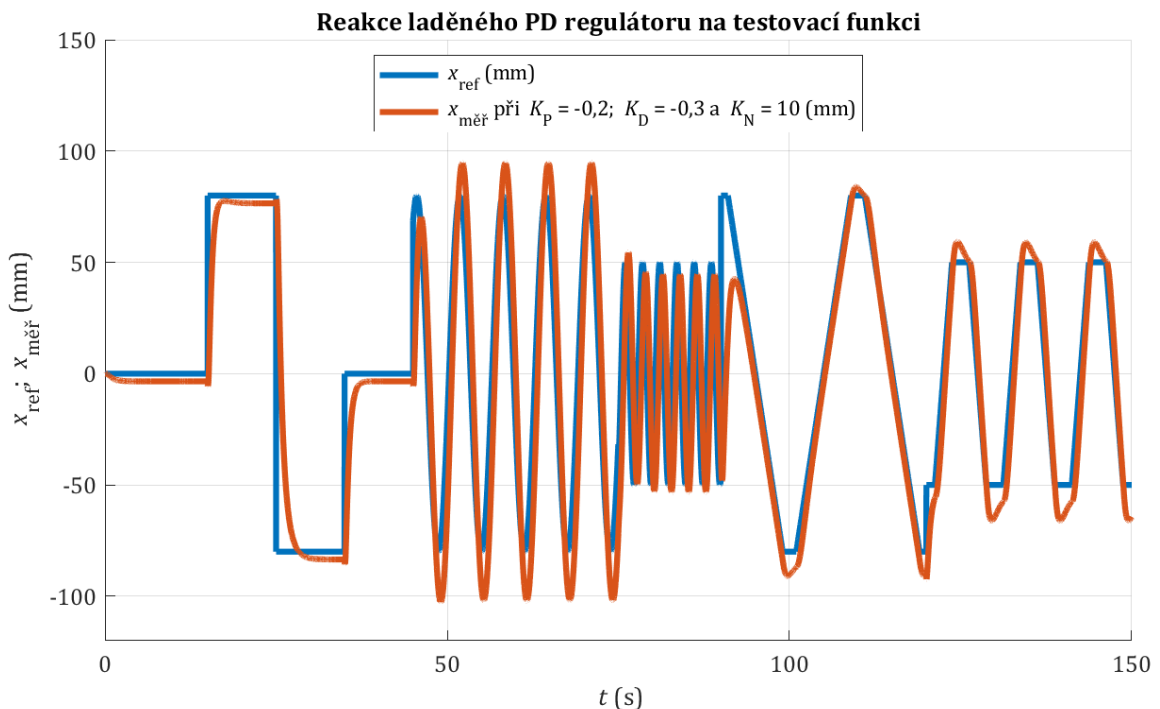
Testovací trajektorie začíná skokovou žádostí až v čase 15 s, aby při ladění fyzického systému byl čas položit kuličky do středu plochy. Skokové žádosti jsou 80 mm v čase $t = 15$ s, -80 mm v $t = 25$ s a 0 mm v $t = 35$ s. Pro trajektorie kruhové typu je vstupní signál sinusový, proto byl od $t = 35$ s do $t = 75$ s testován sinusový signál o amplitudě 80 mm a úhlové rychlosti 1 rad/s ($v = \omega \cdot r = 0,08$ m/s) a od $t = 75$ s do $t = 90$ s sinusový signál o amplitudě 50 mm a úhlové rychlosti 2,5 rad/s ($v = \omega \cdot r = 0,125$ m/s). Pro trajektorie obdélníkového typu se uplatňuje lichoběžníkový signál, kdy od $t = 90$ s do $t = 120$ s je testován signál s amplitudou 80 mm a periodou 20 s (náběh z 0 mm na 80 mm za 4 s, $v_{rampy} = 0,02$ m/s) a od $t = 120$ s do $t = 150$ s signál s amplitudou 50 mm a periodou 10 s (náběh z 0 mm na 50 mm za 1,25 s, $v_{rampy} = 0,04$ m/s) viz obr. 4-6.



Obr. 4-6 Testovací funkce pro ladění parametrů regulátorů při reakcích na různé signály.

Postup manuálního ladění může být zdlouhavý proces, který často provází stavy vazeb proti sobě, kdy změny parametrů zlepší nějaký aspekt regulace, ale oproti tomu se zhorší jiná hodnotící kritéria. Pro problematiku polohování kuličky na plošině o zvolené trajektorii, je žádoucí ladit regulaci pro různé skokové žádosti a pro různé frekvence a amplitudy sinusových signálů. Zvoleným požadavkem při ladění PD regulátoru v simulaci bylo dosažení téměř kriticky tlumené odezvy na skokovou žádost jako v [5].

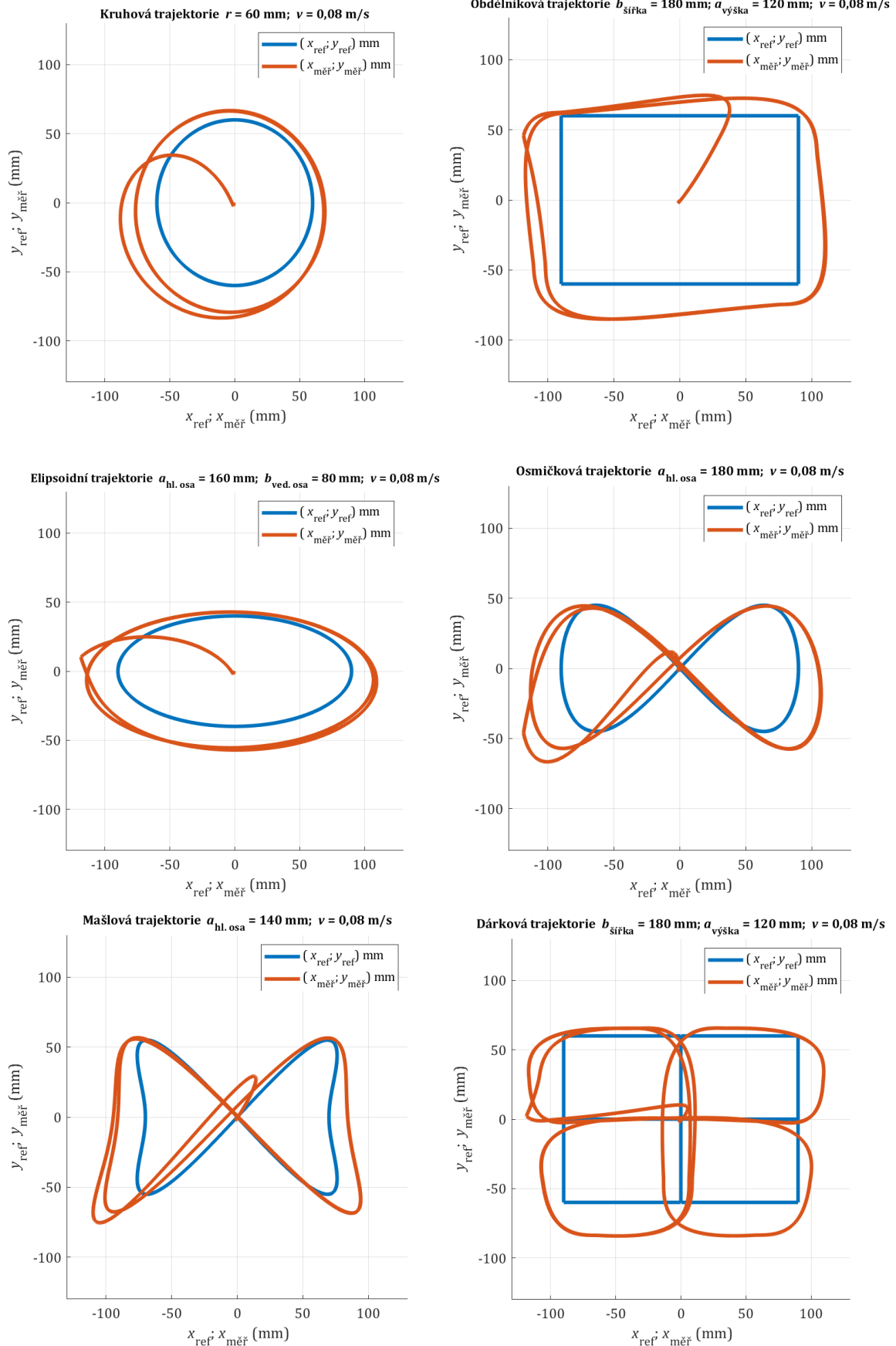
Při volbě parametrů manuálním laděním, je potřeba nejdříve stanovit, zda mají být kladné či záporné. Při pohybu kuličky v kladném směru osy x (doprava ze středu panelu), má být reakcí kompenzace pohybu levotočivou rotací v ose y , tedy zápornou hodnotou ϕ , proto parametry K_P a K_D v PD regulátoru budou nastaveny záporné. Při testování parametrů bylo ustáleno na hodnotách $K_P = -0,2$; $K_D = -0,3$ a $K_N = 10$ a to jak v ose x viz obr. 4-7, tak v ose y , protože reakce při simulaci se v osách nemění.



Obr. 4-7 Simulovaná reakce PD regulátoru na testovací funkci v ose x při zafixování $y_{\text{měř}} = 0$.

Nevýhodou PD regulátoru je trvalá odchylka viz obr. 4-7, proto je omezené použití i pro fyzický systém, kdy náklon podložky promítnutý do soustavy není nikterak kompenzován. Pro testovaný systém byla zvolena požadovaná dynamika $v = 0,8$ s. Z průběhů v obr. 4-8 lze potvrdit, že PD regulace pro simulovaný model není dostačující, protože trvalá odchylka způsobí posun středu přenášeného signálu viz posun sinusovky v obr. 4-7 a amplitudy sinusových signálů neodpovídají žádaným. Co se týče tvaru trajektorie, tak lze zhruba hodnotit tvar odpovídající zadanému, kdy u hran obdélníkovitého a dárkového průběhu je pozorovatelná omezená reakce na hrany a díky překmitu dochází k zakulacení rohů. Tento překmit je pozorovatelný v reakci na lichoběžníkové signály posledních 60 s v obr. 4-7.

Zvýšení K_D či K_N vede ke zpřesnění amplitudy sinusového signálu téměř do žádaného rozpětí (posunuto stále trvalou odchylkou), za cenu však pomalé reakce při jednotkovém skoku, a protože je vyžadována obecně funkční regulace pro signály skokových změn i pro signály sinusové, je potřeba při ladění dosáhnout vhodného kompromisu. Pokud simulovaný model vhodně reprezentuje fyzickou soustavu, lze stanovit výsledné průběhy z obr. 4-8 jako neuspokojivé, a proto by metoda nemusela být ani implementována do fyzického přípravku, dokud by například nebyla provedena vhodná úprava pro zajištění lepších výsledků jako např. *Feedforward*.



Obr. 4-8 Průběhy různých trajektorií pro polohování kuličky na plošině pomocí PD regulátoru.

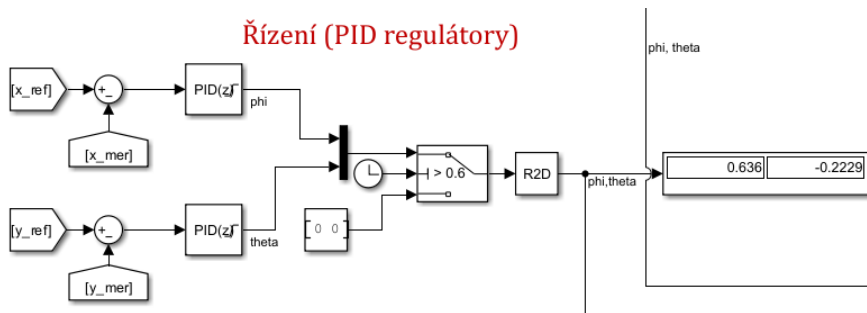
4.3 PID regulace

PID regulace je představená pro 2DOFp ve [114] a pro 3DOFp ve [118], [119], [120]. Umožňuje přes integrační složku K_I omezit trvalou odchylku. Vztah (4-2) je upraven do spojitého tvaru:

$$C_{PID,id}(s) = K_P + K_I \cdot \frac{1}{s} + K_D \cdot \frac{K_N}{1 + K_N \cdot \frac{1}{s}}, \quad (4-4)$$

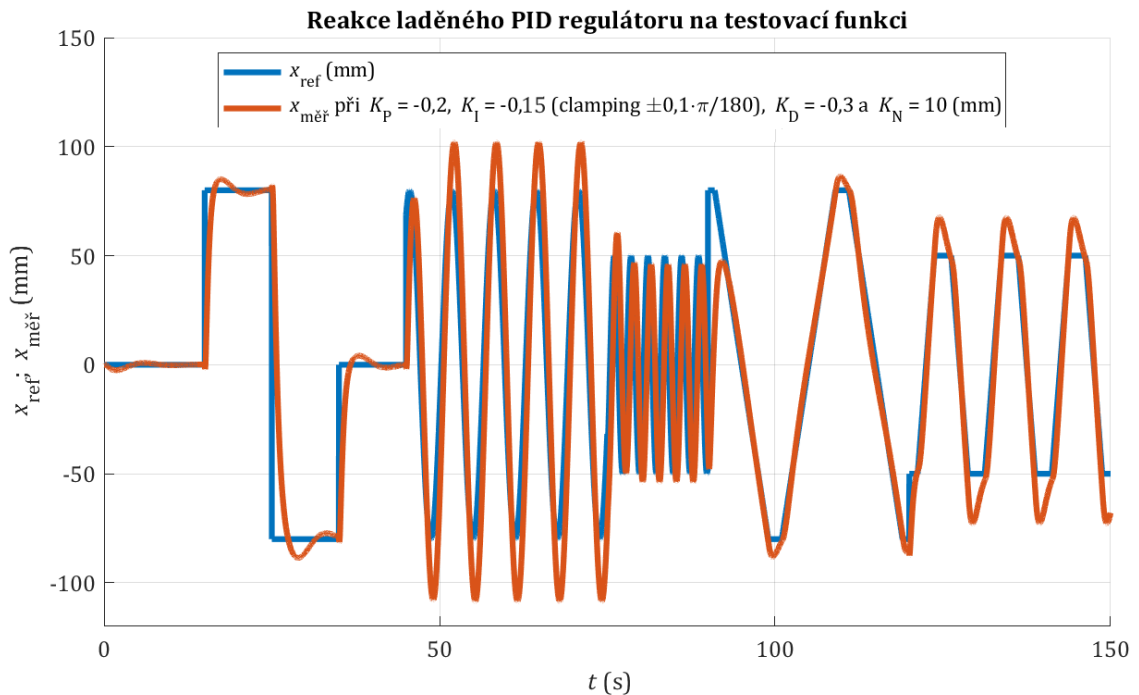
kdy obdobně jako ve (4-3), je opět přechod do diskrétního tvaru s lichoběžníkovou filtrační metodou a omezením výstupu v radiánech $\pm 5 \cdot 180/\pi$ a saturačním omezením integrátoru (*clamping*) s hranicemi $\pm 0,1 \cdot \pi/180$:

$$C_{PID,id}(z) = K_P + K_I \cdot T_s \cdot \frac{1}{2} \cdot \frac{z+1}{z-1} + K_D \cdot \frac{K_N}{1 + K_N \cdot T_s \cdot \frac{1}{2} \cdot \frac{z+1}{z-1}}. \quad (4-5)$$

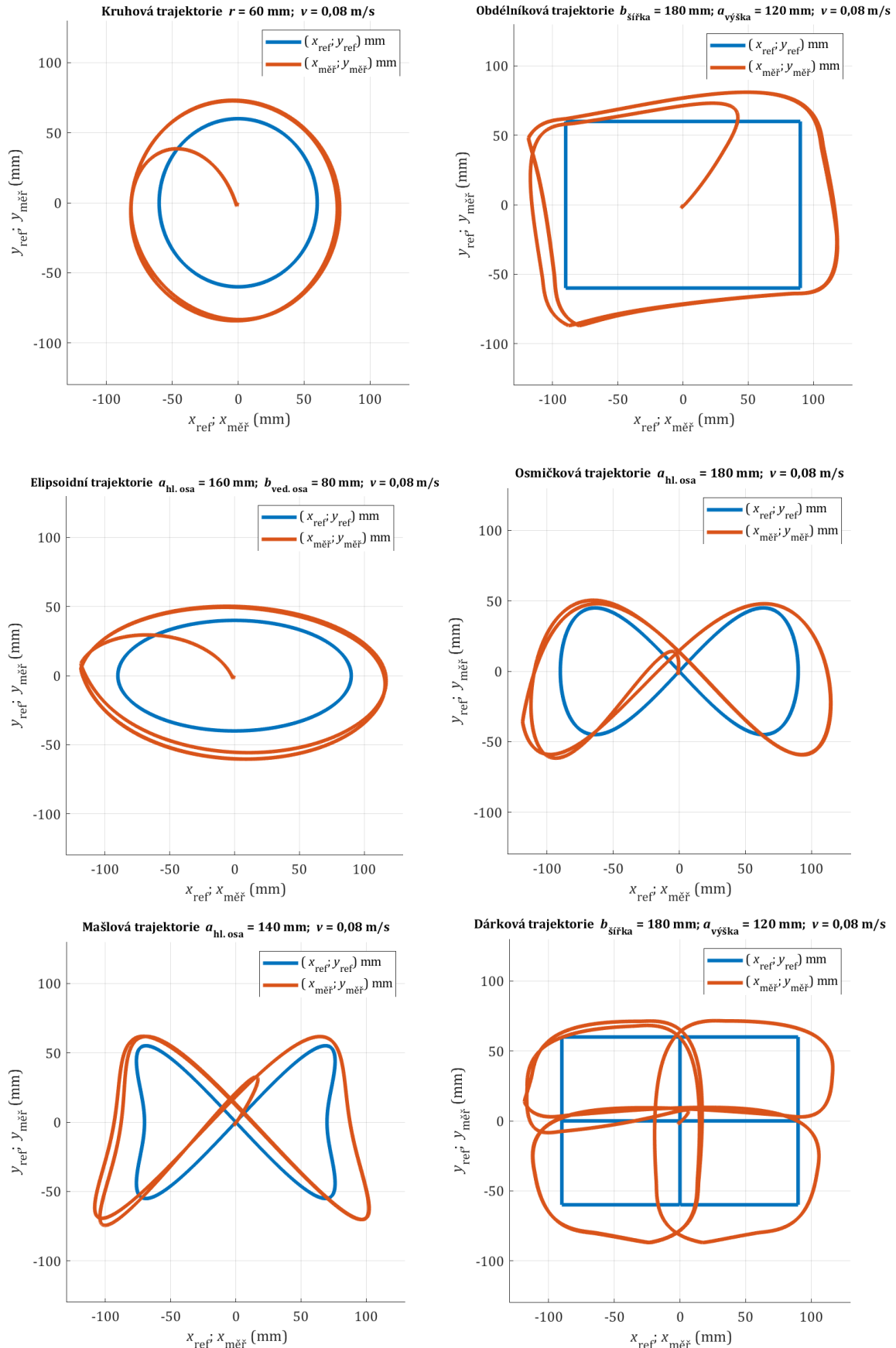


Obr. 4-9 Schéma řízení PID regulátorů v Simulink.

Pro ladění PID regulátoru byl opět využit manuální přístup, kdy při testování parametrů bylo ustáleno na hodnotách $K_P = -0,2$; $K_I = -0,15$ (*clamping* $\pm 0,1 \cdot \pi/180$); $K_D = -0,3$ a $K_N = 10$ a to jak v ose x (viz obr. 4-10), tak v ose y . Průběhy v obr. 4-11 poukazují omezení trvalé odchylky středu v porovnáním s obr. 4-8, ale stále se nejedná o vhodné výsledky pro implementaci.



Obr. 4-10 Simulovaná reakce PID regulátoru na testovací funkci v ose x při zafixování $y_{měř} = 0$.



Obr. 4-11 Průběhy různých trajektorií pro polohování kuličky na plošině pomocí PID regulátoru.

4.4 Kaskádní regulace

Pro řízení mechanických soustav je výhodné používat kaskádní regulaci jako je pro 6DOFp v [57]. V kaskádě je využíváno kombinace diskretních PID regulátorů, kde pro vnější polohovou smyčku byla využita jen složka P se zesílením $K_{P,vnější}$. a pro vnitřní rychlostní smyčku regulátor PI se zesílením $K_{P,vnitř}$. a integrační konstantou $K_{I,vnitř}$.

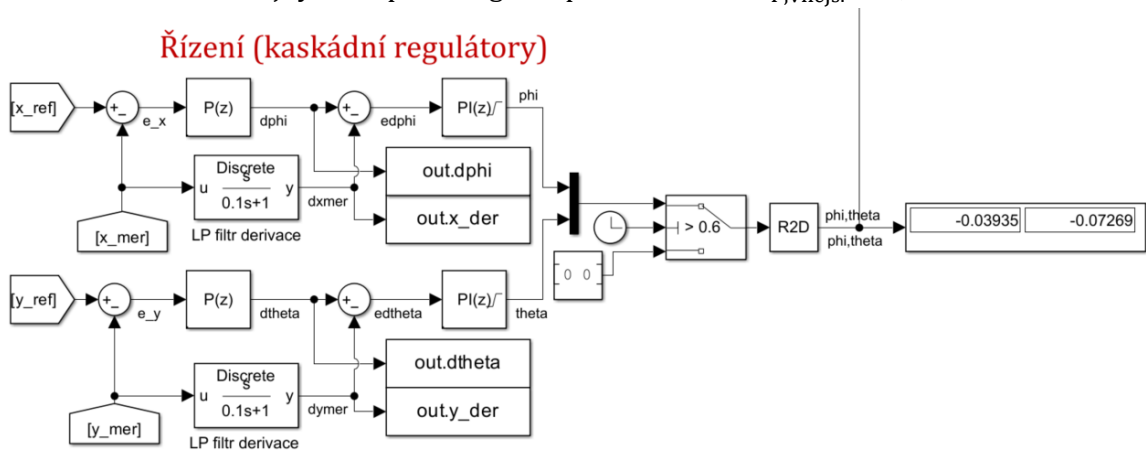
$$C_{PPI,vnější}(s) = K_{P,vnější}, \quad (4-6)$$

$$C_{PPI,vnitř}(s) = K_{P,vnitř} + K_{I,vnitř} \cdot \frac{1}{s}. \quad (4-7)$$

Pro sériovou kombinaci regulátorů převedenou do diskretního tvaru s použitím lichoběžníkové metody integrace je z blokové algebry operace násobení pro otevřený regulační obvod

$$C_{PPI}(z) = C_{PPI,vnější}(z) \cdot C_{PPI,vnitř}(z) = K_{P,vnější} \cdot \left(K_{P,vnitř} + K_{I,vnitř} \cdot T_s \cdot \frac{1}{2} \cdot \frac{z+1}{z-1} \right). \quad (4-8)$$

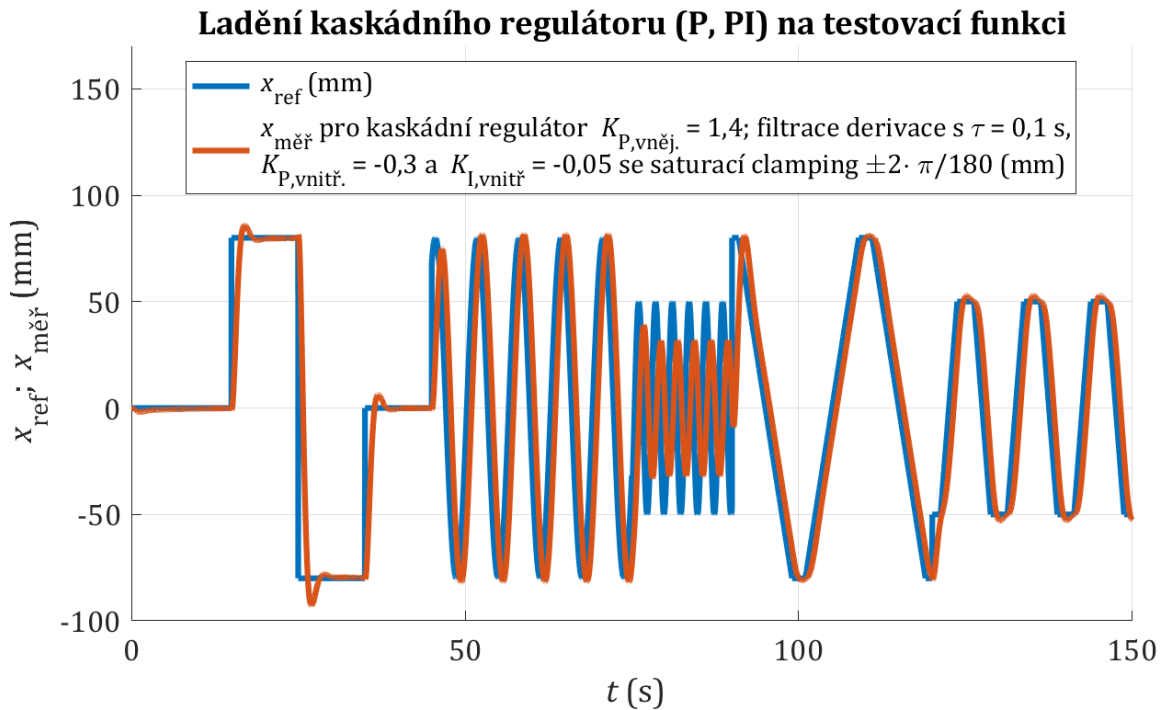
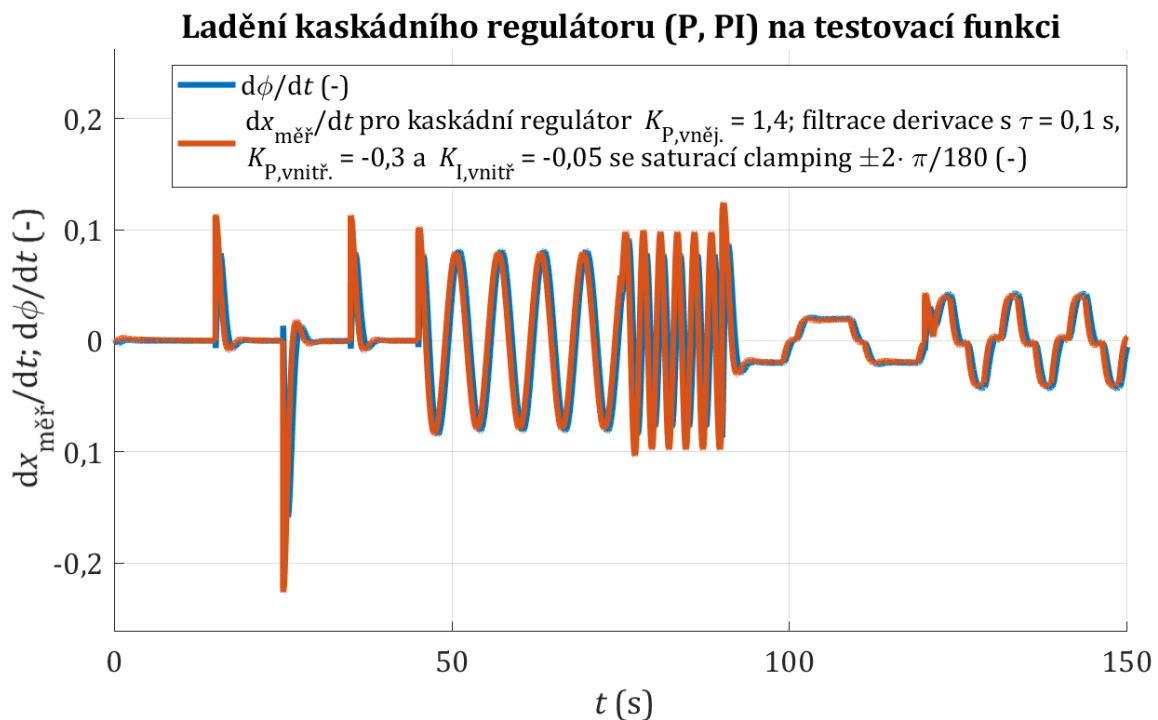
Standardním postupem pro ladění kaskádních regulátorů (viz obr. 4-12) je postup od vnitřního regulátoru, který je často volen s menší periodou. Po zajištění stabilní reakce vnitřního regulátoru je přestoupeno k regulátoru vnějšímu pro dodatečné doladění žádané reakce. Takto bylo postupováno, kdy však perioda regulátorů je stejná jako simulační doba $T_s = 0,01$ s. Pro vnitřní rychlostní smyčku bylo ustáleno na $K_{P,vnitř} = -0,3$; $K_{I,vnitř} = -0,05$ (clamping $\pm 2 \cdot \pi/180$). Pro ladění polohového P regulátoru byly pozorovány reakce ϕ a $\dot{x}_{měř} = v_{x_{měř}}$, kdy laděním zesílení bylo snahou dosáhnout stejných amplitud signálů při ustálení na $K_{P,vnější} = 1,4$.



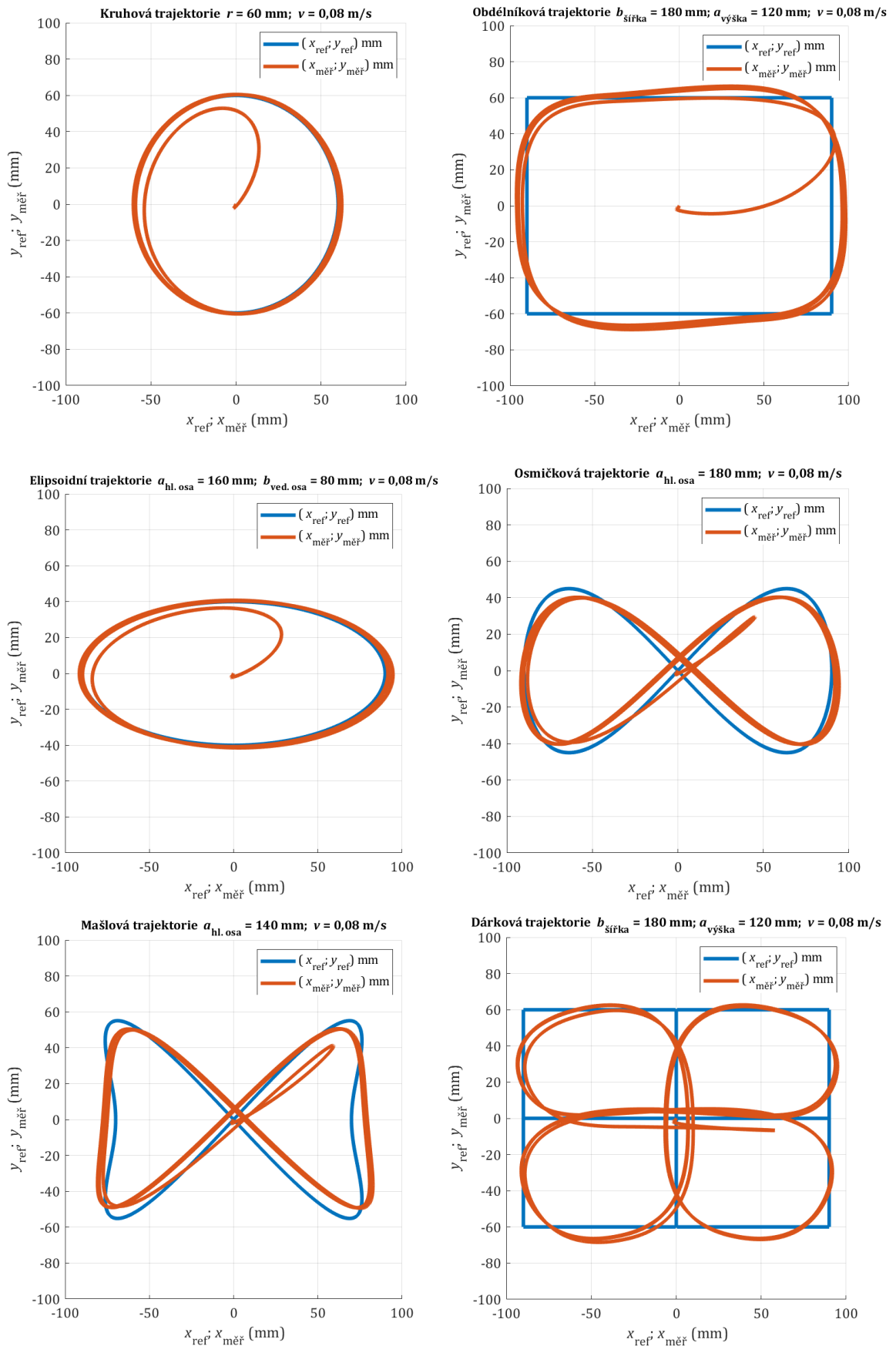
Obr. 4-12 Kaskádní regulátor v Simulink.

Pozorované reakce v ose x měřené polohy $x_{měř}$ vůči referenční x_{ref} jsou v obr. 4-13 a odpovídající $\dot{\phi}$ a $\dot{x}_{měř}$ v obr. 4-14. Je nutné podotknout, že signál polohy je ve fyzické soustavě zašuměný, proto je nutné derivaci polohy filtrovat. Pro simulaci by bylo možné použít nefiltrované derivace, ale protože je snahou přiblížit chování simulace fyzickému systému, je i zde implementována filtrovaná derivace s časovou konstantou $\tau = 0,1$ s.

Z reakcí na skokové změny v obr. 4-13 lze hodnotit omezený překmit s rychlou dobou ustálení oproti průběhu v obr. 4-10. Reakce na první sinusový průběh o rychlosti $v = 0,08$ m/s je s fázovým zpožděním zhruba půl periody (neovlivní vzhled trajektorie), ale amplituda je odpovídající. Při vyšší rychlosti $v = 0,125$ m/s už je však amplituda tlumená, proto lze hodnotit dynamické omezení regulace a pro testované průběhy proto bylo ustáleno na $v = 0,08$ m/s. Stejně jako je tomu u PD a PID regulace, nelze u trajektorií s hranami očekávat přesnou reakci, protože okamžitá změna rychlosti z rampové na nulovou způsobí reakci s překmitem jako je tomu u skokové změny.

Obr. 4-13 Simulovaná reakce kaskádní regulace na testovací funkci v ose x při zafixování $y_{\text{měř}} = 0$.Obr. 4-14 Simulované derivace kaskádní regulace na testovací funkci v ose x při zafixování $y_{\text{měř}} = 0$.

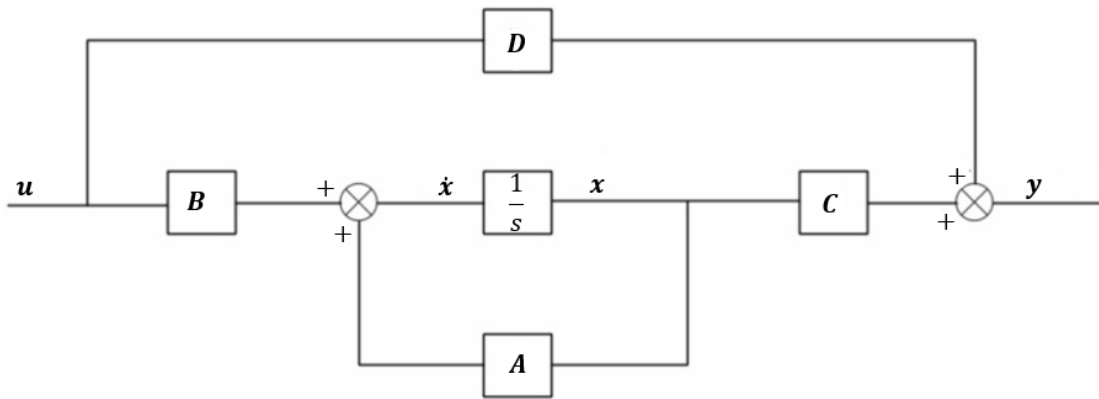
Z výsledných průběhů v obr. 4-15 lze hodnotit značně vylepšené reakce oproti obr. 4-8, obr. 4-11. V průbězích tvořených ze sinusových signálů jako kruh, elipsa, osmička a mašle lze zhruba hodnotit amplitudy odpovídající žádaným. V průbězích tvořených z lichoběžníkových signálů jako je obdélníková a dárková trajektorie jsou hrany dle očekávání zakulacené, protože vyšší volená dynamika se projeví více, ale výsledný tvar trajektorie zhruba odpovídá žádanému. Kaskádní regulace je proto metoda vhodná pro řízení kuličky po trajektorii dle výsledků simulace.



Obr. 4-15 Průběhy různých trajektorií pro polohování kuličky na plošině pomocí kaskádního regulátoru.

4.5 Stavový popis modelu 3DOFp

Při používání modelově založených řídicích metod, či algoritmu pro automatické ladění PID regulátoru, je často vyžadován popis lineárního, časově invariantního systému (LTI). Stavový popis modelu (viz obr. 4-16) je často reprezentován maticemi lineárních transformací stavu A , vstupu B , výstupu C a dopřednou maticí D , kdy vektor x reprezentuje stavy systému, u vstupy systému a y výstupy systému (někdy dodatečný vektor rušení). Počet řádků matice A odpovídá počtu řádků B a počtu sloupců C . Počet sloupců matice B určuje počet sloupců matice D a počet řádků matice C určuje počet řádků matice D .



Obr. 4-16 Model stavového popisu pro spojité, časově invariantní systém [110] (upraveno).

Vztahy popisující stavový model v obecném tvaru nabývají formu rovnice stavu [110]:

$$\dot{x} = Ax + Bu \quad (4-9)$$

a rovnice výstupu:

$$y = Cx + Du. \quad (4-10)$$

Pro vytvoření stavového modelu 3DOFp lze použít komplikovanější přístupy uvažující polohu, rychlosti, momenty setrvačnosti pohybu kuličky a rotační úhly, rychlosti změn rotačních úhlů a momenty setrvačnosti rotace desky kolem osy x a osy y , či jiné hodnotící parametry pro popis systému [113]. K přiblížení přístupu se zjednodušenými parametry, lze postupovat dle [110]. Pro 3DOFp je uvažována konstantní výška $z_t = 20$ mm, proto přechází systém na obdobný model jako 2DOFp. Úhly náklonu jsou omezeny na $\pm 5^\circ$, lze proto linearizovat sinusové funkce rotačních úhlů $\sin \phi \sim \phi$ a $\sin \theta \sim \theta$ (viz obr. 4-17), a protože změny úhlů náklonů nejsou okamžité, kdy rychlosti změn jsou malé, lze zanedbat členy $\dot{\phi}$ a $\dot{\theta}$. Zjednodušená rovnice stavu dle (4-9) nabývá podoby

$$\dot{x} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} e_x \\ \dot{e}_x \\ e_y \\ \dot{e}_y \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -\frac{m_{\text{kul}}}{m_{\text{kul}} + \frac{J_{\text{kul}}}{r_{\text{kul}}^2}} \cdot g & 0 \\ 0 & 0 \\ 0 & -\frac{m_{\text{kul}}}{m_{\text{kul}} + \frac{J_{\text{kul}}}{r_{\text{kul}}^2}} \cdot g \end{pmatrix} \begin{pmatrix} \phi \\ \theta \end{pmatrix} \quad (4-11)$$

a rovnice výstupu dle (4-10)

$$y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} e_x \\ \dot{e}_x \\ e_y \\ \dot{e}_y \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \phi \\ \theta \end{pmatrix}. \quad (4-12)$$

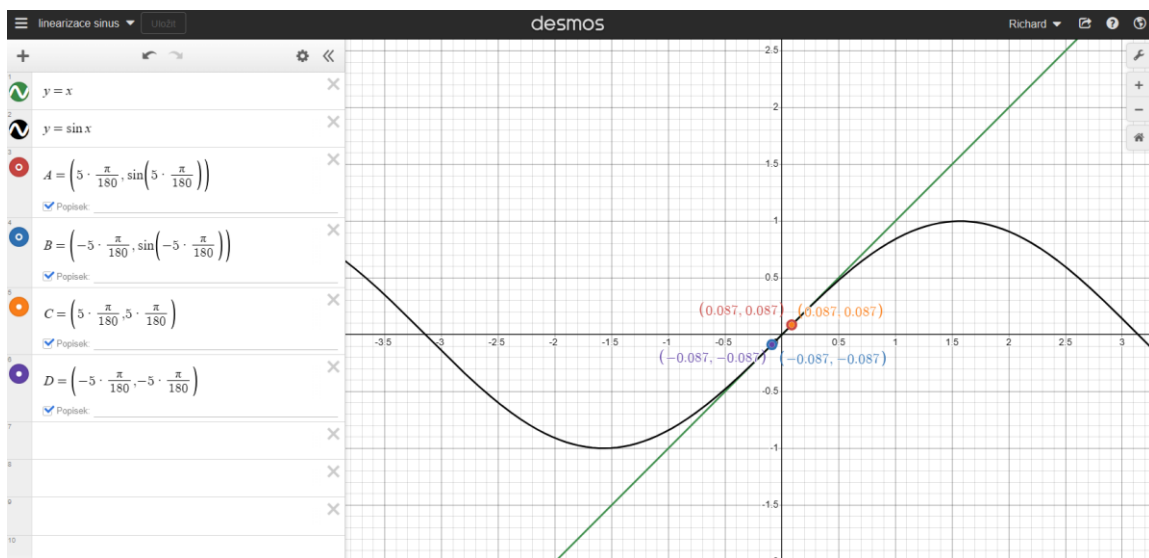
Hmotnost, poloměr a moment setrvačnosti kuličky lze uvést pro přehlednost v tab. 4-1 [113].

Tab. 4-1 Základní parametry pro linearizaci modelu 3DOFp.

Název proměnné	Symbolické označení	Hodnota s jednotkou
tíhové zrychlení	g	$9,81 \text{ m} \cdot \text{s}^{-2}$ (viz A.1)
poloměr větší ocelové kuličky	r_{kul}	17 mm (viz obr. C-1)
váha ocelové kuličky	m_{kul}	161 g (viz obr. C-2)
moment setrvačnosti kuličky	J_{kul}	4000 g·mm ² (viz (4-13))

Výpočet momentu setrvačnosti kuličky:

$$J_{\text{kul}} = \frac{2}{5} \cdot m_{\text{kul}} \cdot r_{\text{kul}}^2 = 4 \cdot 10^{-6} \text{ kg} \cdot \text{m}^2. \quad (4-13)$$



Obr. 4-17 Ukázka linearizace sinu při malých úhlech ($\sin(x) \sim x$) v programu Desmos.

Po dosazení z tab. 4-1 do (4-9) lze dle (4-11) vyjádřit matice \mathbf{A} , \mathbf{B} a ze (4-10) lze dle (4-12) vyjádřit matice \mathbf{C} , \mathbf{D} .

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad (4-14)$$

$$\mathbf{B} = \begin{pmatrix} 0 & 0 \\ -1,962 & 0 \\ 0 & 0 \\ 0 & -1,962 \end{pmatrix}, \quad (4-15)$$

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad (4-16)$$

$$\mathbf{D} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad (4-17)$$

Linearizace modelu zanedbává mnoho vlivů od skluzu kuličky po zpoždění reakce krokového motoru od zadané úhlové změny. Možným řešením je využít komplikovanější nelineární model, který může obsáhnout další vlivy působící na systém, pro 2DOFp je možné odvození ve [128].

4.6 Regulace LQR

Využívanou metodou v literatuře je regulace LQR pro 6DOFp ve [125] a pro 2DOFp ve [110], [130]. Jedná se o metodu založenou na stavovém popisu (viz 4.5). Principem řízení LQR je minimalizace kvadratické spojité penalizační funkce J_C při ladění matice penalizující stavu \mathbf{Q}_{LQR} a matice penalizující řízení \mathbf{R}_{LQR} [110]

$$J_{C,LQR} = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q}_{LQR} \mathbf{x} + \mathbf{u}^T \mathbf{R}_{LQR} \mathbf{u}) dt. \quad (4-18)$$

Postupem přes Riccatiho rovnici (odvozeno ve [129]) lze získat matici \mathbf{S}_{LQR} ze vztahu

$$0 = \mathbf{S}_{LQR} \times \mathbf{A} + \mathbf{A}^T \times \mathbf{S}_{LQR} - \mathbf{S}_{LQR} \times \mathbf{B} \times \mathbf{R}_{LQR}^{-1} \times \mathbf{B}^T \times \mathbf{R}_{LQR}^{-1} \times \mathbf{S}_{LQR} + \mathbf{Q}_{LQR}. \quad (4-19)$$

Optimalizovaný výsledek penalizační (cenové funkce) z (4-18) je pak matice zesílení \mathbf{K}_{LQR} [130]:

$$\mathbf{K}_{LQR} = (\mathbf{B}^T \times \mathbf{S}_{LQR} \times \mathbf{B} + \mathbf{R}_{LQR})^{-1} \times \mathbf{B}^T \times \mathbf{S}_{LQR} \times \mathbf{A}. \quad (4-20)$$

Pro implementaci do řízení je pak matice \mathbf{K}_{LQR} zesilovacím členem vektoru stavu pro určení vektoru vstupu dle

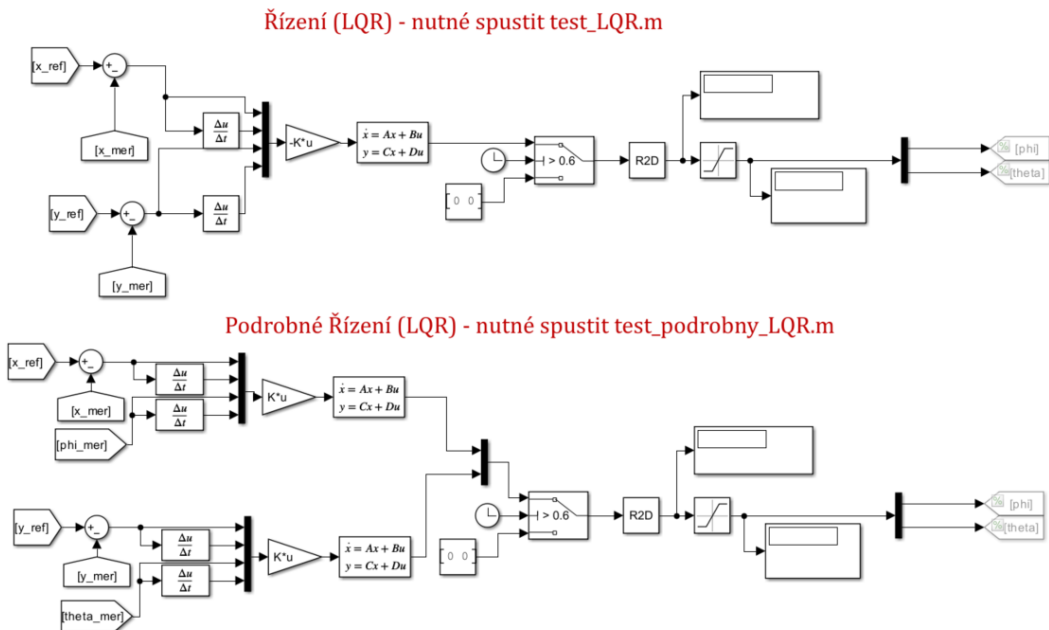
$$\mathbf{u} = -\mathbf{K}_{LQR} \times \mathbf{x}. \quad (4-21)$$

Matice \mathbf{Q}_{LQR} , \mathbf{R}_{LQR} lze ladit (mění se výpočet \mathbf{K}_{LQR}) podle váhy která má být přiřazena hodnocení měřeného stavu (např. volba vyšší priority e_x oproti e_x), kdy zvolené hodnoty byly dle [110]

$$\mathbf{Q}_{LQR} = \begin{pmatrix} 90 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 80 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (4-22)$$

$$\mathbf{R}_{LQR} = \begin{pmatrix} 20 & 0 \\ 0 & 20 \end{pmatrix}. \quad (4-23)$$

Proběhl pokus implementovat LQR do modelu (v obr. 4-18), kdy však integrátor (v obr. 4-16) stavového modelu nedovoloval ustálení výstupních hodnot a nebylo proto LQR modelováno.



Obr. 4-18 Nefunkční testovací model LQR pro linearizovaný systém.

Ukázka kódu s funkcí *lqr*, která v matlab funkci umožňuje výpočet matice K_{LQR} viz (4-23).

```

%% LQR test
g = 9.81; % m*s^-2
mm = 1/1000; % převod z mm na m
rho_kul = 7800; %kg/m^3 hustota oceli
r_kul = 34*mm/2; %m poloměr kuličky
m_kul = 4/3*pi*r_kul^3*rho_kul; %kg váha ocelové kuličky

%matice stavového popisu
A_mat = [0, 1, 0, 0;
         0, 0, 0, 0;
         0, 0, 0, 1;
         0, 0, 0, 0];
B_mat = [0, 0;
        -m_kul*g/(m_kul+J_kul/r_kul^2), 0;
         0, 0;
         0, -m_kul*g/(m_kul+J_kul/r_kul^2)];
C_mat = [1, 0, 0, 0;
         0, 0, 1, 0];
D_mat = [0, 0;
         0, 0];

% volba Q a R matice z LQR
Q_mat = diag([90, 1, 80, 1]);
R_mat = [20, 0;
         0, 20];

% výpočet K matice pomocí lqr funkce v Matlab
[K_mat, ~, ~] = lqr(A_mat, B_mat, Q_mat, R_mat);

%zobrazení výsledné K matice
disp(K_mat);

```

$$K_{LQR} = \begin{pmatrix} -2,12 & -0,81 & 0 & 0 \\ 0 & 0 & -2 & -0,78 \end{pmatrix}. \quad (4-24)$$

Obdobná regulační schémata jako LQR jsou např. diskretní LQ v [131] a [130], LQG (*Linear Quadratic Guassian Control*) ve [127], LQT (*Linear Quadratic Tracking*) ve [110], *State Feedback control* viz [113] a s využitím umístění pólů viz [110], či řídicí metoda minimálního fázového řízení představena v [132].

4.7 Regulace SMC

Klouzavé řízení (*Sliding Mode Control*) představeno pro 2DOFp ve [133] a pro 3DOFp ve [121] je hojně využívanou metodou pro regulaci nelineárních systémů. Metoda spočívá v definování klouzavého povrchu [133] z regulačních odchylek v ose x :

$$s_{klouz,x} = -\dot{e}_x + \lambda_{klouz,x} \cdot (-e_x) \quad (4-25)$$

a obdobně v ose y :

$$s_{klouz,y} = -\dot{e}_y + \lambda_{klouz,y} \cdot (-e_y), \quad (4-26)$$

kde konstanta λ musí být kladná a definuje dynamiku klouzavého povrchu. Dále je aplikována rovnice řízení pro zajištění konvergence kuličky do žádané polohy (nulové odchylky) pro obě osy. Obdobně jako ve vztahu (4-11), je nutné definovat odhad konstanty z nelineárního modelu [133].

$$b_{klouz} = -\frac{m_{kul}}{m_{kul} + \frac{J_{kul}}{r_{kul}^2}} \cdot g. \quad (4-27)$$

Používaný regulátor je hysterezní (též zvaný *bang-bang*, či *on-off*), kdy výstupní veličina z funkce *signum* (funkční hodnoty jen -1;0;1) musí být zesílena koeficientem β_{klouz} pro zajištění robustního řízení. Výsledné rovnice pro řízení s integrátorem jsou v ose *x* ze [133]:

$$u_{klouz,x} = \arcsin\left(\frac{1}{b_{klouz}} \cdot (\ddot{x}_{ref} + 2 \cdot \lambda_{klouz,x} \cdot \dot{e}_x + \lambda_{klouz,x}^2 \cdot e_x - \beta_{klouz,x} \cdot \text{sgn}(s_{klouz,x}))\right) \quad (4-28)$$

a v ose *y*:

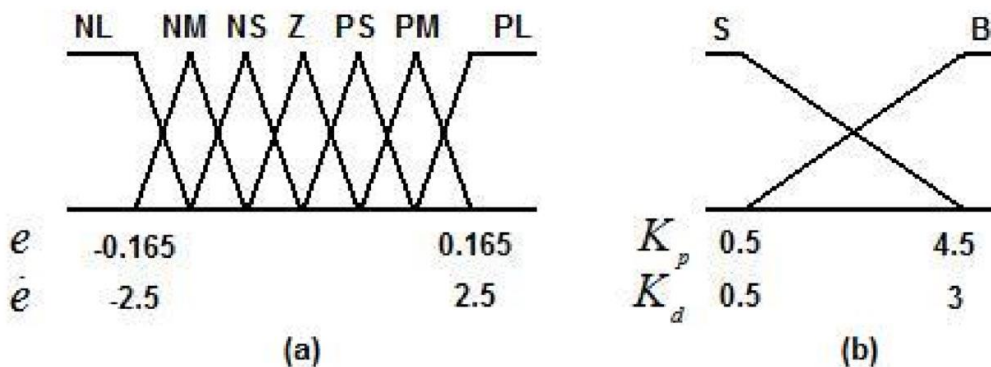
$$u_{klouz,y} = \arcsin\left(\frac{1}{b_{klouz}} \cdot (\ddot{y}_{ref} + 2 \cdot \lambda_{klouz,y} \cdot \dot{e}_y + \lambda_{klouz,y}^2 \cdot e_y - \beta_{klouz,y} \cdot \text{sgn}(s_{klouz,y}))\right). \quad (4-29)$$

Laděné parametry $\lambda_{klouz,x}$, $\lambda_{klouz,y}$, $\beta_{klouz,x}$ a $\beta_{klouz,y}$ lze získat heuristicky, tedy manuálním laděním do dosažení žádaných reakcí (stejně jako je tomu u ladění parametrů PID regulátoru). SMC je často používáno v kombinaci s jinými řídicími metodami jako např. ve [116], kde je v kombinaci s adaptivní neuronovou sítí s využitím *fuzzy* logiky.

4.8 Regulace FLC

Regulace založená na *fuzzy* logice (*fuzzy logic*) je představena pro 3DOFp ve [117], [126] a pro 2DOFp ve [134]. Častým schématickým využitím je mít *Fuzzy* regulátor jako nadřazený prvek PID regulátoru [134], či jako mezi prvek hybridního PD regulátoru jako ve [117]. Využití je pro modelování systémů, které jsou komplexní a mají stavy neurčitosti, kde může *fuzzy* logika zasáhnout bez znalosti přesného modelu systému. Jednou z problematik PD regulátoru je, že pro nějaké referenční signály by bylo výhodné využít jiné konstanty regulátoru než pro referenční signály jiné (viz obr. 4-7 s rozdílnou reakcí, kde u sinusové funkce nižší amplitudy a vyšší frekvence je signál utlumen, zatímco u vyšší amplitudy a nižší frekvence je příliš zesílen). Pro rozlišení stavů regulace, lze stanovit jako hodnotící parametry odchylky žádané polohy v obou osách e_x , e_y a jejich vývoj v čase, tedy derivace \dot{e}_x , \dot{e}_y [134].

Při větší odchylce může být výhodné mít větší hodnotu členu zesílení a menší člen derivační, ale při menší odchylce k omezení překmitu může být výhodnější opačně mít nižší člen zesílení a větší derivační složku. Pro různé vstupní proměnné lze nastavit různá pravidla pro definování různých konstant regulátoru. V principu *fuzzy* je proto potřeba stanovit vstupní a výstupní proměnné *fuzzy* logiky a jejich vztahy přes pravidla (pokud je nějaký stav, dej tyto hodnoty, *if-then rules*). Vstupy a výstupy dle pravidel ve [117] a [134] jsou rozděleny do tabulky o 49 pravidlech, kdy může být např. kategorie odchylky záporné a menší, oproti kategorii kladné odchylky a větší. Pro vizuální představení volených parametrů a v jaké *fuzzy* logice vůči sobě se nachází, je často vykreslována funkce členů dle vybraných pravidel viz obr. 4-19 [135]. Implementace *fuzzy* logiky vyžaduje znalost chování systému ke stanovení pravidel a omezení.



Obr. 4-19 (a) Funkce členů pro e , \dot{e} ; (b) funkce členu pro parametry K_p a K_d [134].

4.9 Regulace MPC

Modelově prediktivní řízení (*model predictive control*) představeno pro 2DOFp ve [111], [112], [122] a pro 6DOFp ve [109], [136], [138]. Jedná se zpravidla o výpočetně náročnější metodu, zvětšující požadavky na výpočetní rychlost hardwaru, proto ve [122], [109] je např. využita mikroprocesorová jednotka STM32, umožňující větší výpočetní výkon než DUE. V principu je podobnost metodě LQR založené na lineárním stavovém modelu, kdy pro MPC je možné i využití nelineárních modelů systému. Pro optimalizaci musí být model transformován do diskretní formy s pevným časovým intervalem mezi vzorky (konstantní vzorkovací perioda). Pro popis LTI systému přechází rovnice (4-9), (4-10) na diskretní tvar stavového popisu [136], [111]

$$\mathbf{x}_{k+1} = \mathbf{A}_d \cdot \mathbf{x}_k + \mathbf{B}_d \cdot \mathbf{u}_k, \quad (4-30)$$

$$\mathbf{y}_{k+1} = \mathbf{C}_d \cdot \mathbf{x}_k + \mathbf{D}_d \cdot \mathbf{u}_k. \quad (4-31)$$

Matice stavu, vstupu, výstupu a dopředná matice přechází ze spojitého tvaru do diskretního zpravidla pomocí [136]

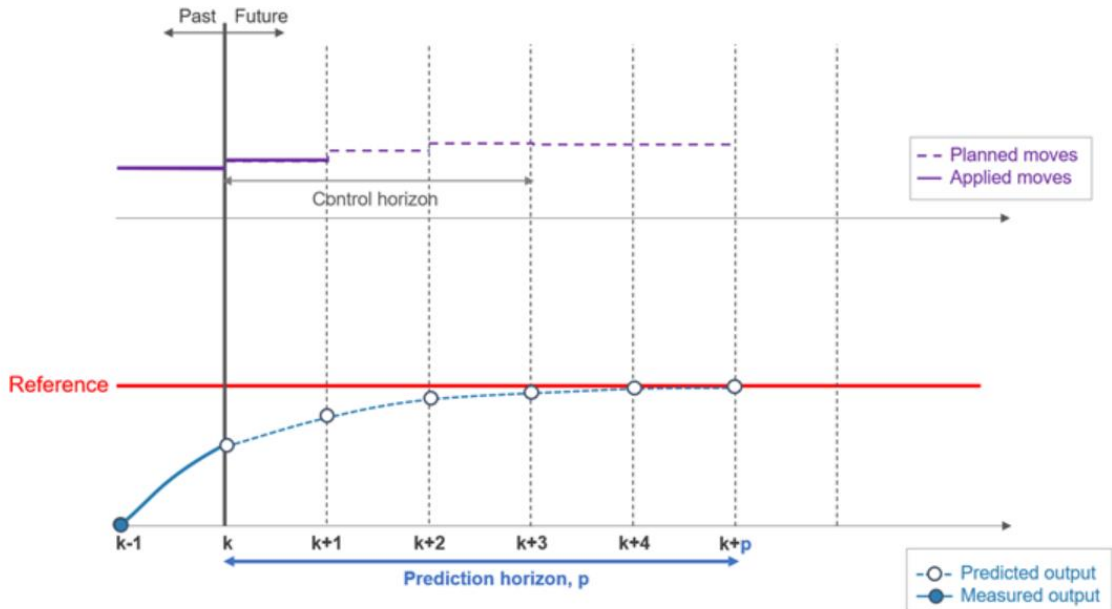
$$\mathbf{A}_d = e^{\mathbf{A} \cdot T_s}. \quad (4-32)$$

$$\mathbf{B}_d = \left(\int_{t=0}^{T_s} e^{\mathbf{A} \cdot t} dt \right) \times \mathbf{B}. \quad (4-33)$$

$$\mathbf{C}_d = \mathbf{C}. \quad (4-34)$$

$$\mathbf{D}_d = \mathbf{D}. \quad (4-35)$$

Laděnými parametry jsou prediktivní horizont N_p a horizont řízení N_u dle vzorkovací periody T_s s ekvidistantními vzorky $u(k) = u(k-1) + \Delta u(k)$ viz obr. 4-20.



Obr. 4-20 Volba prediktivního horizontu a řídicího horizontu [137].

Penalizační funkce (cenová funkce, *cost function*) pro MPC může být definována dle [109] v maticovém tvaru:

$$J_{C,MPC} = (\mathbf{r} - \mathbf{y}_p)^T \times \mathbf{Q}_{MPC} \times (\mathbf{r} - \mathbf{y}_p) + \Delta \mathbf{u}^T \times \mathbf{R}_{MPC} \times \Delta \mathbf{u}, \quad (4-36)$$

kde \mathbf{r} je reference, tedy žádaná hodnota polohy kuličky, \mathbf{y}_p je predikce výstupu (odhad pohybu kuličky na ploše), $\Delta \mathbf{u} = \mathbf{u}(k-1) - \mathbf{u}(k)$ určuje inkrementaci vstupu, diagonální matice \mathbf{Q}_{MPC} udává váhu regulační odchylky a diagonální matice \mathbf{R}_{MPC} váhu inkrementace vstupu. Matice mohou být vyjádřeny v obecném tvaru [109]:

$$\mathbf{Q}_{\text{MPC}} = \begin{pmatrix} q_1 & \cdots & 0 \\ \vdots & \ddots & 0 \\ 0 & 0 & q_m \end{pmatrix}, \quad (4-37)$$

$$\mathbf{R}_{\text{MPC}} = \begin{pmatrix} r_1 & \cdots & 0 \\ \vdots & \ddots & 0 \\ 0 & 0 & r_m \end{pmatrix}, \quad (4-38)$$

$$\Delta \mathbf{u} = \begin{pmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_u-1) \end{pmatrix}, \quad (4-39)$$

$$\mathbf{r} = \begin{pmatrix} \mathbf{r}'_1 \\ \mathbf{r}'_2 \\ \vdots \\ \mathbf{r}'_{N_p} \end{pmatrix}, \mathbf{r}'_j = \begin{pmatrix} r_1(k+j) \\ r_2(k+j) \\ \vdots \\ r_m(k+j) \end{pmatrix}, \quad (4-40)$$

$$\mathbf{y}_p = \begin{pmatrix} \mathbf{y}'_1 \\ \mathbf{y}'_2 \\ \vdots \\ \mathbf{y}'_{N_p} \end{pmatrix}, \mathbf{y}'_j = \begin{pmatrix} y_{p1}(k+j) \\ y_{p2}(k+j) \\ \vdots \\ y_m(k+j) \end{pmatrix}, \quad (4-41)$$

kde m je dimenze výstupu. Po stanovení proměnných je minimalizována penalizační funkce, lze použít algoritmus na minimalizaci jako např. matlab funkci *quadprog* [139]. Dalším krokem je sestavení omezení systému, kdy např. musí být definovány hraniční sklony desky, či rychlosti náklonů pro plynulý pohyb kuličky. Při testování MPC jsou laděny parametry matic \mathbf{Q}_{MPC} , \mathbf{R}_{MPC} a horizonty N_p , N_u pro dosažení optimální regulace nejdříve na modelu. Omezením metody je proto rozsáhlost model, tedy co vše postihuje, od tření kuličky, po momenty setrvačnosti desky. Pokud model představuje dostatečnou reprezentaci fyzického systému, lze pak z naladěného regulátoru na modelu přejít na stabilní regulátor pro fyzickou sestavu. Nuancí v implementaci MPC je mnoho od *off-line* výpočetního přístupu ve [122] po využití pozorovatelů (*observer*) jako je Kalmanův filtr pro odhad vnitřních stavů systému ve [136], [138].

4.10 Strojové učení

Strojové učení (*reinforcement learning*) je populární metodou pro řízení komplikovaných robotických systémů jako je [102]. Pro implementaci metody je potřeba stanovit vektor pozorovaných stavů jako jsou např. vzdálenosti od žádané polohy, rychlosti náklonu, parametry kuličky. Po stanovení vzorkovacího času a délky testované trajektorie, je implementace tzv. odměňovací funkce. Jedná se o podobný princip penalizační funkci, kdy je však zpravidla více pozorovaných stavů, kterým je přiřazena váha pro ovlivnění výstupní hodnoty funkce. Dalším krokem je vytvoření tzv. *soft actor - critic* agenta. Jedná se o prvek zpětné vazby, hodnotící každý průběh (*critic*) a zpravidla přes implementaci neuronové sítě (ukázka aplikace pro 2DOFp v [115]) ovlivňuje budoucí běh laděním parametrů pro získání lepšího pomyslného skóre v dalším běhu (*actor*) [102]. Proces konvergence může být zdoluhavý a volba hodnotících parametrů nemusí být zřejmá. Při např. spuštění ukázky ze [102], odhadovaná výpočetní doba při používání počítače z [25] byla kolem 23 hodin. Ve shrnutí se jedná o silnou metodu, která v aplikacích, kde nelze použít jiných alternativ, může být výhodná, ale je nutné počítat s dlouhým výpočetním časem.

KAPITOLA 5: FYZICKÁ IMPLEMENTACE ŘÍZENÍ 3DOFp

Před implementací řízení do fyzického modelu 3DOFp, je nutné *Simulink* model upravit pro komunikaci s fyzickou soustavou přes DUE. Protože se využívá digitalizované komunikace, je nutné provést definici vzorkovacích časů modelu systému. První malou úpravou je změna v kódu matlab funkce z 3.2, kdy parametry pro fyzickou soustavu zůstávají stejné jako pro soustavu simulovanou s výjimkou výměn simulační periody za vzorkovací doby fyzických prvků:

```
%%Vzorkovací časy
T_s_driver = 0.0005;% s perioda komunikace simulink pro pulzy step/dir
T_s_displej = 0.01;% s perioda čtení hodnoty z displeje
```

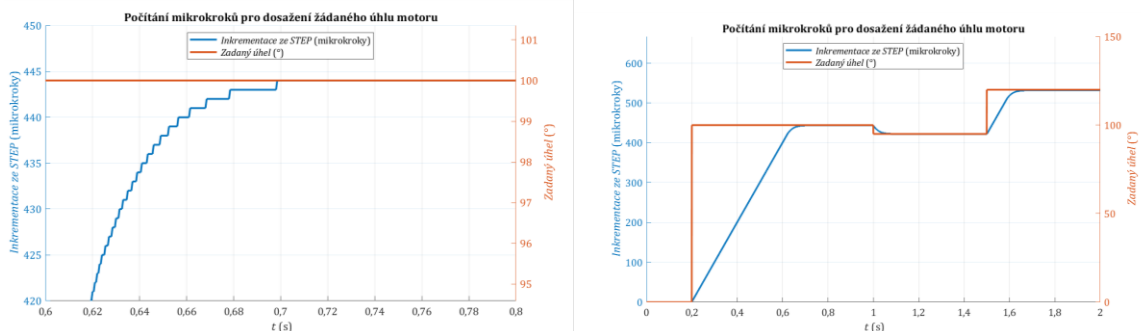
Volba vzorkovacího času rezistivního panelu vychází z technického listu výrobce [68] (*chattering time 10 ms*), kde je uvedena maximální vzorkovací perioda 10 ms pro detekci ustáleného signálu polohy dotyku na panelu (doba do ustálení kontaktu mezi vodivými vrstvami pro správné změření napětí). Sériová komunikační rychlost mezi DUE a *Simulink* je omezená, kdy je volena nejvyšší 115200 Bd (odpovídá komunikační rychlosti 115200 znaků za sekundu). Největší komunikační zátěží je signál pro otáčení motorů, kdy do driverů je posílán pulzovaný signál *STEP* a *DIR*, nelze však vyčerpat veškerou komunikační kapacitu jen na tyto signály, protože je potřeba např. ještě přijímat zpětnou vazbu polohy panelu, či stisknutí tlačítek a posílat změny parametrů jako konstanty regulátorů, či signály led diod. Experimentálně bylo zjištěno, že optimalizovanou hodnotou, kdy není komunikace přehlcena, ale zároveň je dostatečná dynamika otáčení motorů je při periodě *STEP* signálu 1 ms, což odpovídá periodě simulace 0,5 ms. Pro komunikaci DUE se *Simulink*, je využito následujících knihoven:

- *Simulink Support Package for Arduino Hardware* [140],
- *Simulink Coder* [141].

Pro komunikaci s mikroprocesorovou jednotku, je nutné použít diskretní řešič, kdy je volený typ s pevným krokem a volenou simulační dobou dle prvku s největší frekvencí, tedy vzorkovací čas driveru určuje pevný krok 0,5 ms. Pro implementaci fyzického modelu na hardware je nastaveno automatické propojení s deskou DUE při připojení USB.

5.1 Popis modelu pro fyzickou sestavu 3DOFp

Pro komunikaci s DUE je využíváno bloků z knihovny [140], kdy číslování zapojených pinů je dle obr. B-3. Pro výstup je využíván blok *Digital Write*, který posílá pulzy *STEP* a *DIR* pro ovládání počtu kroků, směru otáčení a přes proměnnou frekvenci pulzů i rychlosti otáčení krokového motoru. V blocích *Matlab Function* je použit kód, jehož funkcí je dle zadané rychlosti měnit frekvenci pulzů *STEP* a počítání počtu mikrokroků se směrem pro dosažení žádaného úhlu.

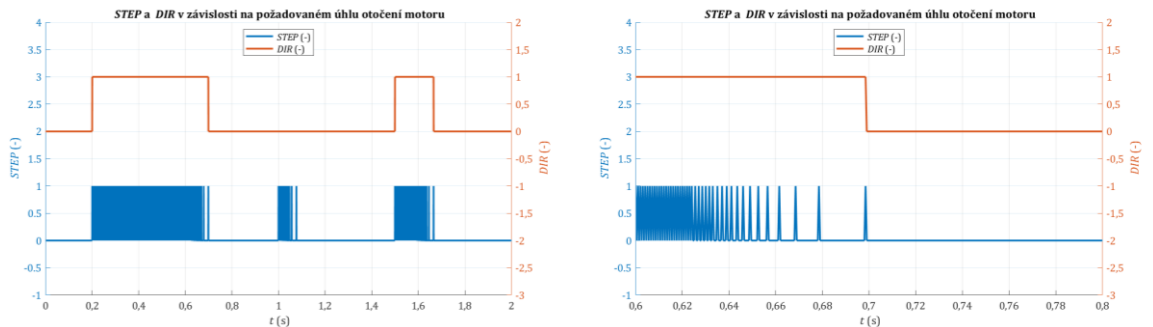


Obr. 5-1 Funkce simulující reakci krokového motoru, inkrementování mikroků k dosažení úhlu otočení.

Volbou parametru zesílení rychlosti (zvoleno 50) je určena rychlost otáčení krokové motoru omezena největší spínací frekvencí (simulační vzorkovací dobou 1 ms na periodu). Vytvořená funkce zajišťuje postupné zvětšování frekvence při doběhu do žádaného úhlu, protože na krokových motorech není zpětná vazba o úhlu otočení (inkrementální enkodér, či resolver), je vše počítáno programově, proto jakákoliv odchylka či ztráta kroku se stává trvalou. Při doběhu

motoru je proto programově zpomalováno při přiblížení k žádanému úhlu viz obr. 5-1, aby byla minimalizována šance ztráty kroku a byl plynulý pokles rychlosti a ne skokový.

V obr. 5-2 je reprezentace posílaných pulzů do CNC shield. Signály *STEP* a *DIR* jsou logického typu 0 nebo 1 (*boolean*) a změnou frekvence u *STEP* je ovládána rychlost mikrokrokování. Pro vyvážení rychlosti otáčení s přesností dosaženého úhlu byla zvolena hodnota mikrokrokování 1/4. Jednotkou žádaného úhlu otočení jsou stupně, je proto potřeba přepočítat ze stupňů na počet mikrokroků (parametr zesílení), a pokud by byl mikrokrokovací parametr upraven, je nutné upravit i tento přepočet náhradou čísla čtyři za nově zvolenou hodnotu.



Obr. 5-2 Parametry *STEP* a *DIR* posílané do driveru pro testovací úhly otočení motoru.

Funkce tlačítek slouží zejména pro demonstrační účely. Pro využití nebylo dostačující funkce stisknuto, nestisknuto, proto byla vytvořena inkrementační funkce, kdy počet stisků jednotlivých tlačítek je počítán (při stisku je rozsvícena led dioda) a resetovací funkce umožňuje nulování stavu tlačítka podržením tlačítka déle než 1 s (znamená dostatečně dlouhého stisku je zhasnutí led diody). Touto úpravou lze získat více využitelných stavů pro použití v přepínání mezi trajektoriemi (černé tlačítko), nebo mezi metodami řízení (červené tlačítko), či mezi žádanou výškou z_t (zelené tlačítko).

Pro zpětnou vazbu polohy rezistivního dotykového panelu připojeného přes kontrolér, je nutné provést úpravu IDE kódu ze [72] pro použití v bloku *S-Function Builder*. Pro správnou funkci *S-Function Builder* je nutné funkci pojmenovat, a zvolit překládaný jazyk na C++. V položce *Libraries* je vložena cesta k používaným knihovnám. Používané knihovny musí být ve stejné složce jako model (či k nim musí být přidání cesta přes *Set Path*) a je potřeba zajistit, aby byl ke každé knihovně příslušný *header file (H File)* a *source file (CPP File)*. Pod *Ports and Parameters* jsou vytvořeny proměnné pro detekování polohy a dotyku panelu a je volen diskretní mód se vzorkovacím časem 0,01 s (nelze zadat proměnnou z matlab funkce, vše musí být definováno v rámci funkce). Po spuštění funkce pomocí *Build*, jsou generovány kódy pro implementaci na hardware (přípony *.cpp*, *wrapper.cpp*, *.tlc* a *.mexw64*).

Pro čtení hodnot polohy dotyku na panelu, je využíváno digitálních vstupů, protože analogové hodnoty jsou převedeny na digitální pomocí integrovaného 12-bitového ADC převodníku přímo na kontroléru STMPE610 [72], tedy čtená digitální hodnota je v mezích 0 až 4095 ($2^{12} = 4096$ hodnot). K přepočtení polohy na panelu je využit *1-D Lookup Table* umožňující přepočet do metrů v rámci aktivní části panelu. Pro zajištění pravé nuly, byla vložena kulička dotykem přesně do středu panelu, a až pak laděny hraniční koncové body.

Funkce panelu byla ověřena v průbězích, kdy bylo zjištěno občasné naměření mylné hodnoty pro menší kuličku o průměru 25 mm. Pro omezení rušivých hodnot byl vytvořen filtr, ale pro kuličky 30 mm a 34 nastávají mylné hodnoty velice zřídka, proto není potřeba ho využít.

Blok inverzní kinematiky se nemění od modelu v obr. 3-7, kdy je implementována funkce zeleného tlačítka pro přepínání vybraných pozic z_t a červené tlačítko sloužící pro spuštění regulace po vložení kuličky na dotykový panel.

Pro řízení fyzické sestavy je obdobně jako v simulaci vytvořeno řídicí schéma s PD regulátory, s PID regulátory a kaskádní regulací složené z P regulátoru polohové smyčky a PI regulátoru

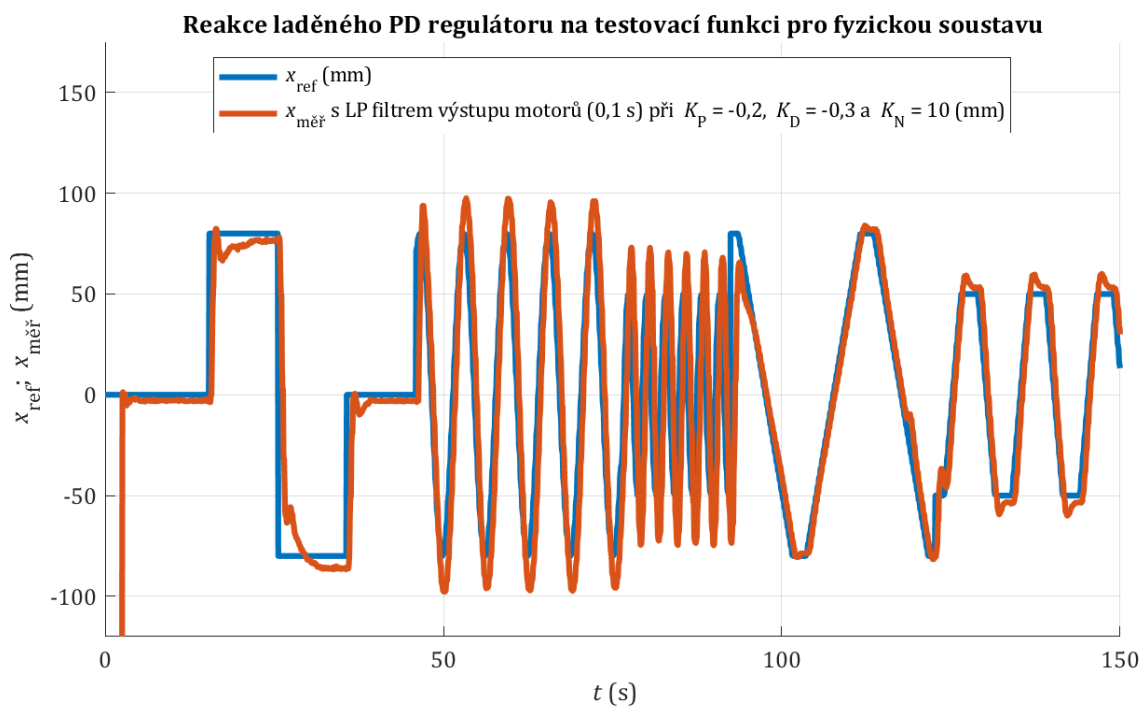
rychlostní smyčky. Testovací funkce jsou implementovány analogicky jako v simulaci z obr. 4-6. Trajektorie dle obr. 4-2 jsou také součástí modelu. Parametry regulátorů jsou voleny jako externí, aby bylo jejich ladění dostupnější za běhu programu. Vykreslování trajektorií je dle funkce exportu *To Workspace*, kde jsou data dále zpracovány a vkládány do grafů výsledných průběhů (je nutné vyjmutí časových celků trajektorií z testu).

PD regulátor je obdobný jako v simulaci se saturací výstupních hodnot úhlu natočení dotykového panelu v radiánech $\pm 5 \cdot 180/\pi$. PID regulátor je analogický k simulovanému s výjimkou funkce *External reset*, kdy hodnoty integrátoru jsou vynulovány v případě, že je detekován stav odebrání kuličky. Zvolenou tolerovatelnou trvalou odchylkou je naměřená hodnota ± 3 mm od žádané hodnoty. Při žádosti nulové polohy má regulace stále tendenci měnit výstupní parametry, i při dostatečné přesnosti, proto když je dosaženo volené odchylky (čtverec ± 3 mm se středem v nule), tak je regulace vypnuta, dokud poloha kulička není opět naměřena mimo interval (může způsobit vnější rušení jako popostrčení kuličky, či vytvoření náklonu přípravku např. pozvednutím v libovolném směru).

Při testování výsledného ustálení, byl zjištěn přílišný šum vyšších frekvencí výstupního signálu, který se projevoval zbytečnou reakcí motorů. Za tímto účelem je na vstup žádaného úhlů otočení motorů implementován *Low Pass filter*, který se zvolenou časovou konstantou $\tau = 0,1$ s omezuje šum složený ze signálu vyšších frekvencí při ponechání zesílení pro akční signály frekvencí nižších.

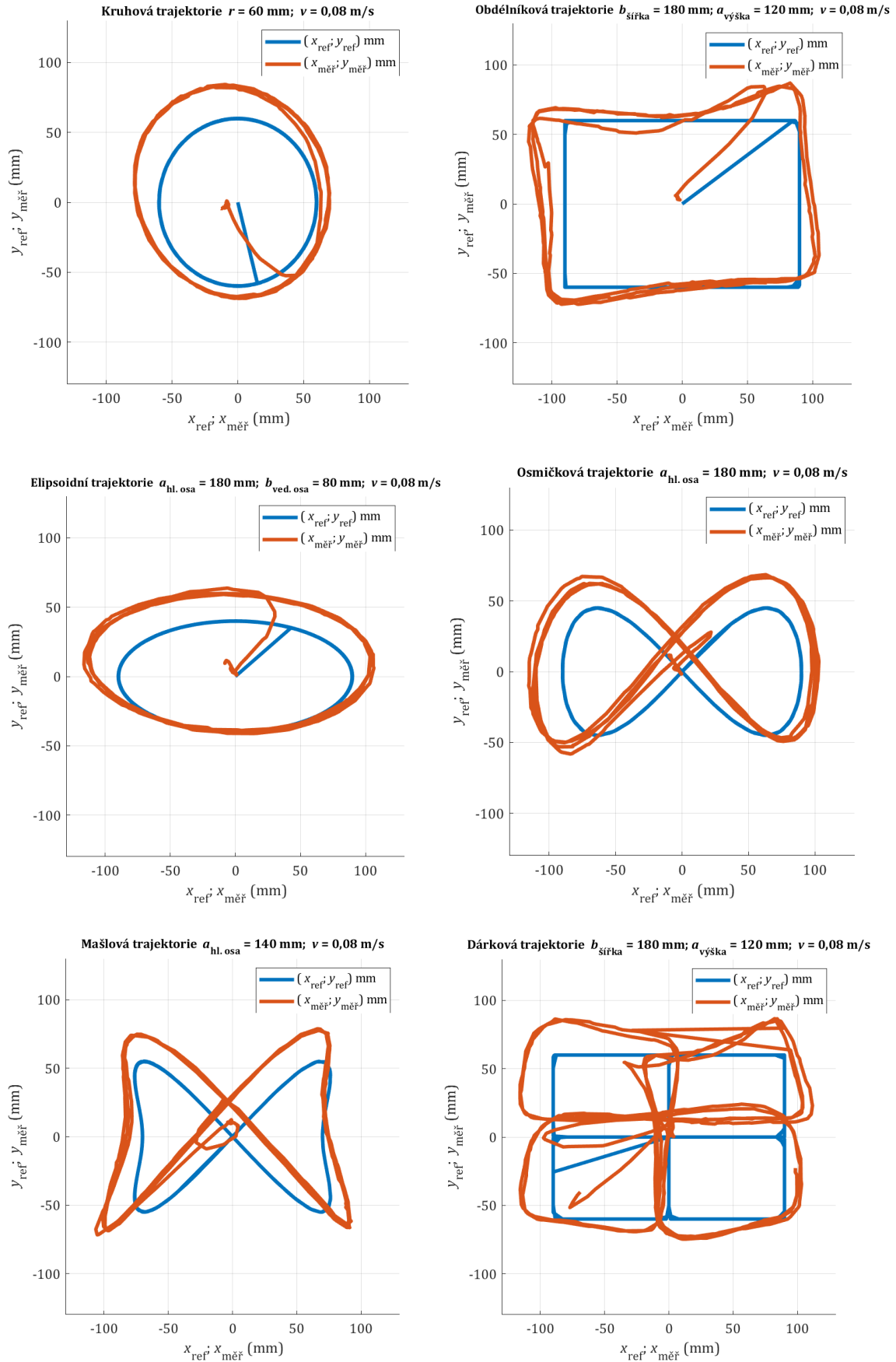
5.2 PD regulátor pro fyzickou sestavu 3DOFp

Obdobně jako v simulaci, je nejdříve laděno zvláště v ose x , kdy pohyb kuličky je omezen v ose y pomocí dvou rovnoběžně nalepených pravítek, a analogicky je postupováno pro ladění osy y . Nevýhodou PD regulátoru zůstává trvalá odchylka, která může značně ovlivnit kde je bod ustálení. Byl spuštěn test ze simulace trvajícím 150 s a kulička položena na střed viz počáteční zákmit v obr. 5-3. Zvolené konstanty regulátorů jsou jako v simulaci s dodatkem *Low Pass* filtru s $\tau = 0,1$ s, tedy $K_P = -0,2$; $K_D = -0,3$ a $K_N = 10$ v ose x i v ose y .



Obr. 5-3 Reakce PD regulace na testovací funkci v ose x při zafixování $y_{m\acute{e}r} = 0$ pro fyzickou soustavu.

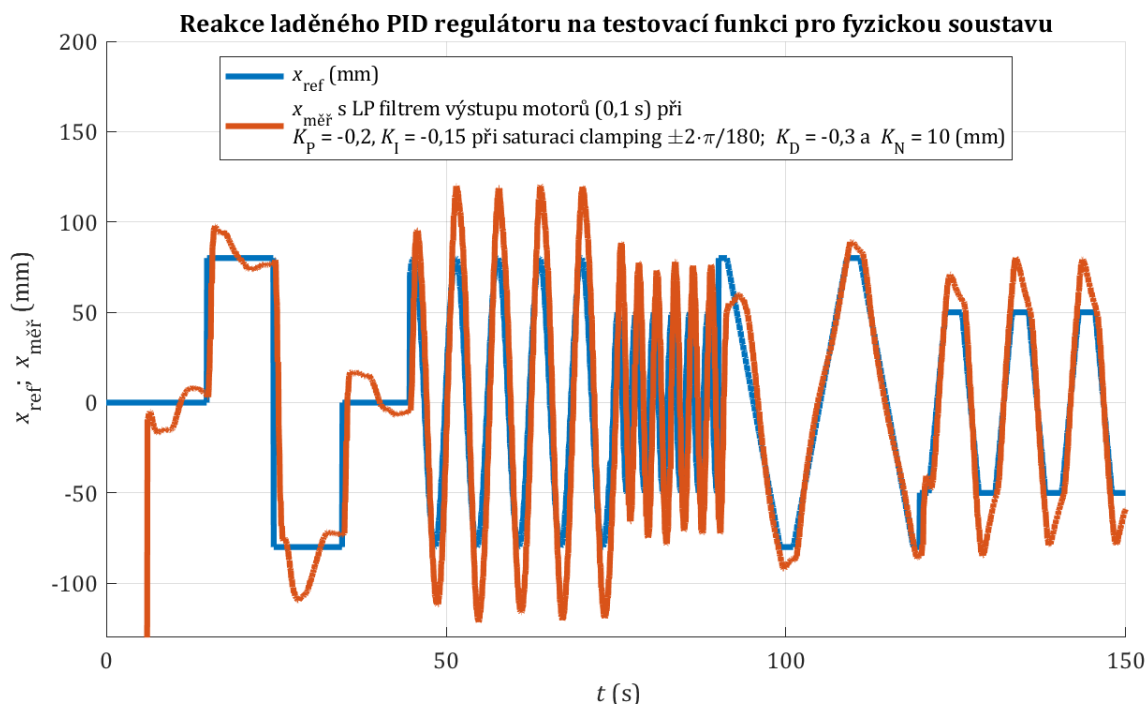
Naměřené průběhy z obr. 5-4 demonstrují trvalou odchylku a nevhodnou amplitudu signálu jako tomu bylo v předpokladu ze simulace viz obr. 4-8. Lze hodnotit regulaci jako nevhodnou.



Obr. 5-4 Průběhy trajektorií pro polohování kuličky na plošině pomocí PD regulátoru fyzicky.

5.3 PID regulátor pro fyzickou sestavu 3DOFp

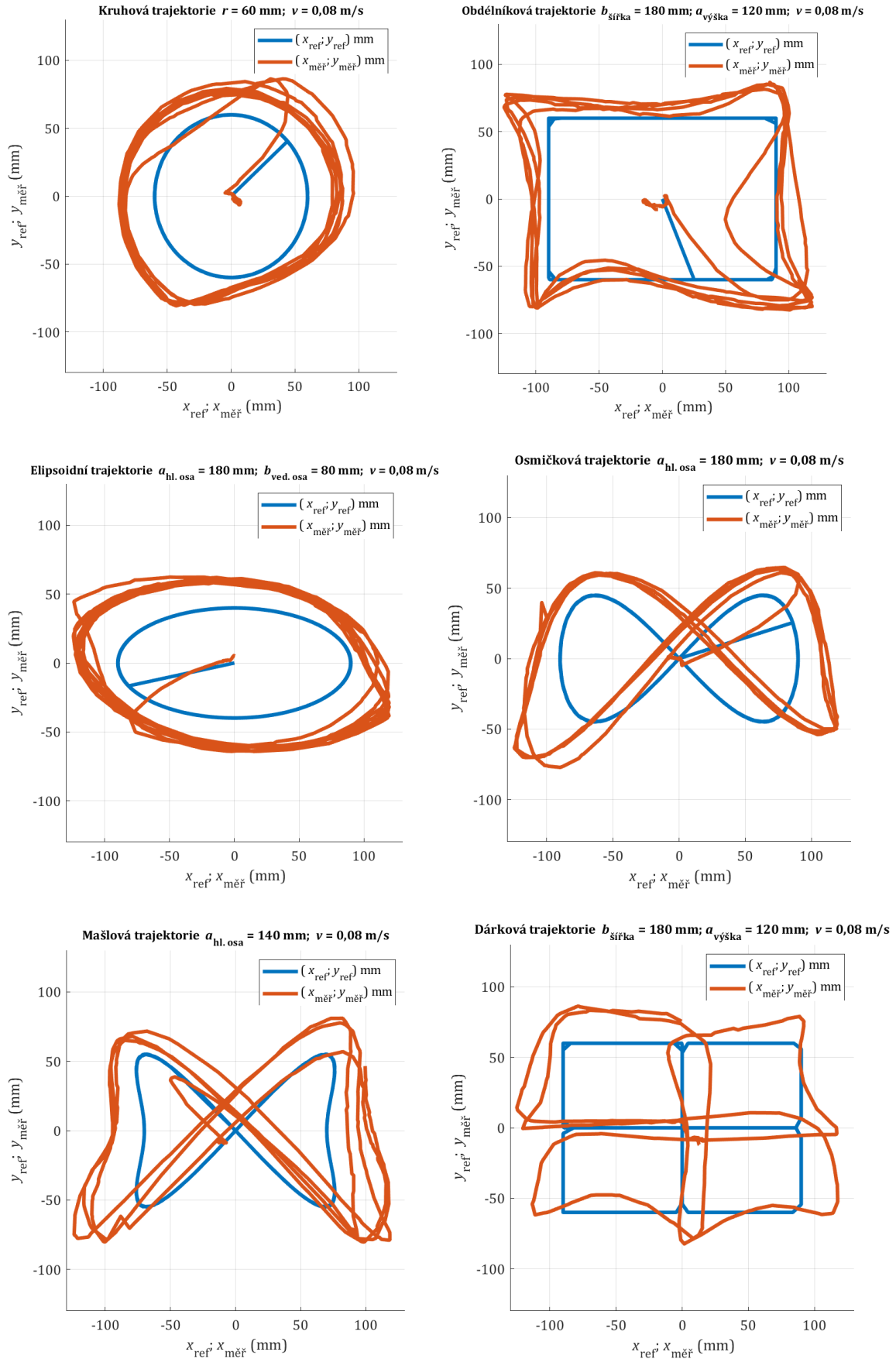
Odlišně než pro simulaci, byla zvolená větší saturační mez $\pm 2 \cdot \pi/180$ (pro porovnání, omezení výstupního žádaného úhlu natočení platformy v radiánech je $\pm 5 \cdot \pi/180$). Ostatní konstanty regulátoru jsou ponechány jako v simulaci $K_P = -0,2$; $K_I = -0,15$ a $K_D = -0,3$. Testovací průběh v obr. 5-5 demonstruje vlivy větší saturační meze oproti průběhům z obr. 5-3, kdy vzniká v průběhu skokové žádosti významnější překmit, ale je odstraněna trvalá odchyłka.



Obr. 5-5 Reakce PID regulace na testovací funkci v ose x při zafixování $y_{měr} = 0$ pro fyzickou soustavu.

Kdyby při měření na PD regulátoru byl pod desku vložen předmět pro vytvoření trvalého náklonu plošiny, v průbězích obr. 5-3 a obr. 5-4 by projev trvalé odchyłky mohl posunout střed trajektorie natolik, že poloha kulička by mohla narážet do mezí stanovené rohovými úchytkami dotykového panelu a případně při dostatečně rychlém nárazu by mohla kulička z přípravku spadnout. Protože PID regulace disponuje větším překmitem, lze pozorovat při měření na fyzickém přípravku lehké narážení kuličky do hranice stanovené úchytkami jako je tomu u pravé stěny elipsoidní trajektorie viz obr. 5-6. Lze proto potvrdit nevhodnost implementované PID regulace pro řízení kuličky po trajektorii, kdy značně rozdílné amplitudy referenčního signálu vůči měřenému působí zdeformování trajektorií.

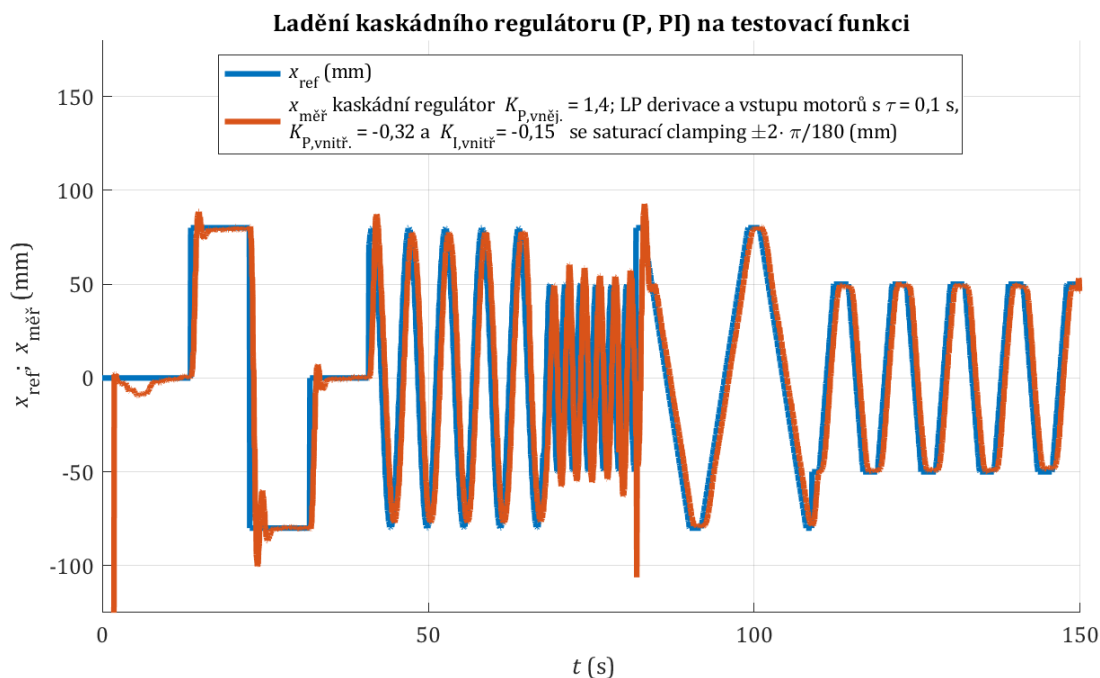
Při dodatečném doladění PID regulátoru lze dosáhnout obdobného průběhu jako v PD regulaci při snížení integrační složky a větším omezení saturačních mezí. Tato úprava však působí negativně na dobu ustálení kuličky v žádané nulové poloze po kompenzaci trvalé odchyłky, proto pro ukázky byla zvolena mez větší. Zásah integrační složky se tím stává významnější, aby při žádaní nulové polohy byl demonstrován její vliv. V dodatečných blocích *Radio Button* byly vloženy různé konstanty pro testování kombinací parametrů PD a PID regulace. Při změně parametrů lze demonstrovat, jak přílišné zvýšení derivační složky může zvýšit citlivost řízení na změnu polohy kuličky a vést až k nestabilitě systému, či zvýšení složky zesílení může vést k rychlejšímu dosažení žádané polohy, ale s významným překmitem a dlouhou dobou ustálení a obdobně může být u integrační složky demonstrováno, že příliš vysoká hodnota vede k oscilaci kolem žádané nulové hodnoty a nedochází k ustálení regulátoru. Obecně však nebylo dosaženo uspokojivých průběhů pomocí samotného PD regulátoru viz obr. 5-4, ani pomocí samotného PID regulátoru viz obr. 5-6, proto je vhodné testovat poslední simulovanou metodu kaskádní regulace.



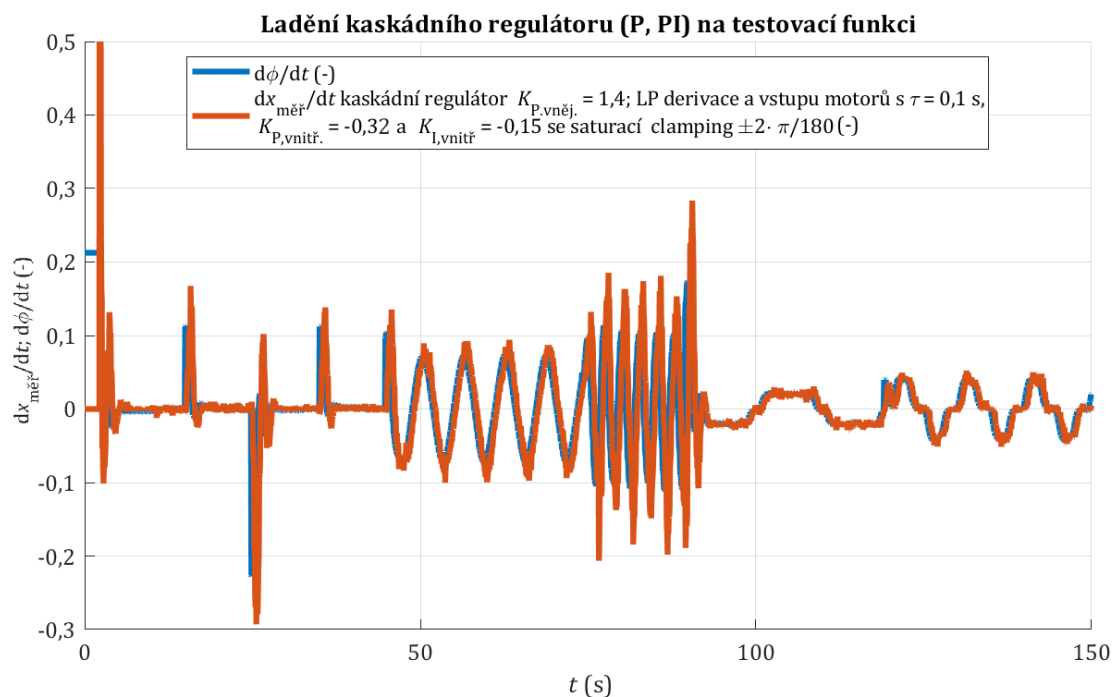
Obr. 5-6 Průběhy trajektorií pro polohování kuličky na plošině pomocí PID regulátoru fyzicky.

5.4 Kaskádní regulátor pro fyzickou sestavu 3DOFp

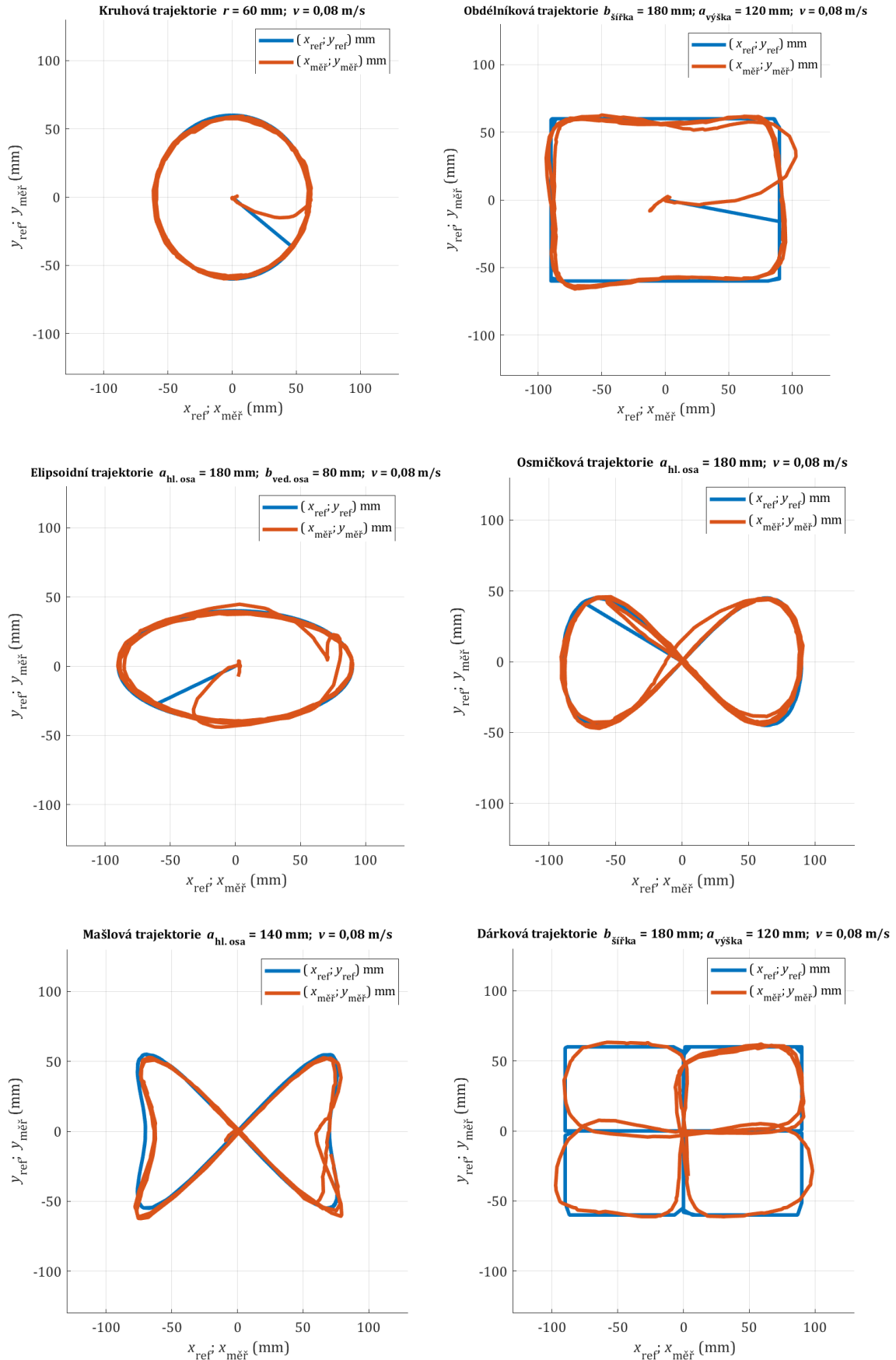
Dle simulovaných průběhů obr. 4-13, obr. 4-14 a obr. 4-15 se kaskáda polohového regulátoru P a vnitřního rychlostního regulátoru PI jevila jako jediná možná aplikovatelná pro fyzickou sestavu a po získání průběhů obr. 5-7, obr. 5-8 a obr. 5-9 je zřejmé, že se jedná o vhodnou regulaci dosahující žádaných parametrů dynamiky, přesnosti a stability při $K_{P,vnitř} = -0,32$ v ose x a $K_{P,vnitř} = -0,37$ v ose y ; $K_{I,vnitř} = -0,15$ (clamping $\pm 2 \cdot \pi/180$) a $K_{P,vnějš} = 1,4$.



Obr. 5-7 Reakce polohy kaskádní regulace na testovací funkci v ose x pro fyzickou soustavu.



Obr. 5-8 Reakce rychlosti (derivace) kaskádní regulace na testovací funkci v ose x pro fyzickou soustavu.



Obr. 5-9 Průběhy trajektorií pro polohování kuličky na plošině pomocí kaskádního regulátoru fyzicky.

ZÁVĚR

Bylo demonstrováno možného využití PD, PID regulátoru a kaskádní regulace pro polohování ocelové kuličky na plošině. Omezením PD regulátoru je chybějící integrační složka pro odstranění trvalé odchylky. Při vyrovnání roviny přípravku lze proto dosáhnout průběhů dle obr. 5-4, kdy však při představení trvalého náklonu podstavy je systém ustálený mimo pravý střed a průběhy mohou nabývat posunu a zdeformovaných tvarů jako je tomu u PID regulátoru v obr. 5-6. Přidáním integrační složky průběhy dosahují větších překmitů, ale již dochází k omezení trvalé odchylky viz obr. 5-5. Obě metody PD a PID regulace již ze simulovaných průběhů v obr. 4-8 a obr. 4-11 projevily nedostatky pro dosažení dynamické a přesné reakce a při porovnání simulovaných průběhů s průběhy fyzické sestavy v obr. 5-4 a obr. 5-6 lze hodnotit, že simulovaný model vhodně reprezentuje chování reálného přípravku.

Alternativní úprava testovaná v modelu byla kaskádní regulace, která pomocí vnější polohové smyčky s regulátorem P a vnitřní rychlostní smyčky s regulátorem PI demonstrovala v modelu přesnější reakci při požadované dynamice $v = 0,08$ m/s viz obr. 4-15. Dle předpokladu, že simulační model vhodně reprezentuje fyzickou soustavu, bylo kaskádní řízení analogicky implementováno a obdrženy srovnatelné výsledky v obr. 5-9, ze kterých lze hodnotit kaskádní regulaci jako vhodnou pro polohování kuličky na plošině o třech stupních volnosti.

Představeny v kapitole 3 jsou pokročilejší řídicí metody, které by mohly být nejdříve implementovány v simulačním modelu a testovány na fyzické sestavě. Vhodně se nabízí např. MPC, kdy na simulačním modelu by mohl být vyvinut prediktivní regulátor a po naladění by mohla jeho funkce být ověřena na fyzickém přípravku.

V komunikaci z prostředí *Simulink* s DUE je omezená přenášená komunikační rychlost, kdy je posílán pulzovaný signál pro řízení signálů *STEP* a *DIR* krokových motorů, a proto bylo zvoleno mikrokrokování 1/4 pro zachování dostatečné dynamiky systému. Možnou implementací je dodatečné mikroprocesorová deska, která by obdržela žádaný počet kroků po komunikaci a zajistila pulzování pro drivery krokových motorů s mnohem větší frekvencí, než je tomu např. dle obr. 5-2. Mohlo by tak být dosaženo přesnějšího polohování a plynulejšího otáčení motorů použitím např. mikrokrokování 1/16.

Vyvinutý fyzický model disponuje nedokonalostmi. Dosažitelný úhel náklonu by mohl být větší pro vytvoření více dynamické soustavy. Zároveň při ladění regulátorů byly zjištěny vibrace přípravku způsobené vůlemi pohybu mimo uvažované stupně volnosti. Lze proto uvažovat nad dodatečnou úpravou mechanického celku, kdy prvky mohou být vyrobeny z pevnějších a odolnějších materiálu pro pevnější vazby menších vůlí k omezení nežádoucích vlivů nedokonalých prvků na celek. K omezení vibrací a tím způsobeného hluku byla využita filtrace výstupního signálu do motorů pro potlačení nežádoucích projevů fyzické soustavy kolem ustálené polohy. V prostředí simulace tyto projevy nejsou, a proto nelze tyto potřebné úpravy ladit jinak, než testováním funkčního celku a analyzováním případných nedostatků se zjišťováním jaké úpravy by mohly být provedeny. Jedná se o často přehlíženou součást při prezentování práce, ale schopnost omezit nežádoucí vlivy mechatronického systému jako je 3DOFp vyžaduje trpělivost, zkušenosti a z pozorování systému aplikovat znalosti k zjištění jaké nežádoucí jevy mohou být přítomny a jak mohou být případně potlačeny, proto je rozhodně prostor pro další práci na přípravku, ať už je to vývoj náročnějších regulačních metod, či mechanické přemodelování.

Plošina byla vytvořena s vizí prezentace základních řídicích metod jako je využití regulátorů PID při ukázání vlivů měněných parametrů. Za tímto účelem je mechanismus vhodný, protože oproti např. ladění regulátoru střídače, je zde přímá vizuální zpětná vazba toho, jak se projevuje rušení na systém, jak dlouhou trvá regulaci odstranit odchylky, lze pozorovat vliv různých žádaných signálů na chování regulace a lze hodnotit přesnost polohování kuličky a tendence systému být v nestabilitě, kdy poloha kuličky může oscilovat až do případného spadnutí kuličky z desky pro demonstraci překročení fyzikálních omezení soustavy.

LITERATURA

- [1] Instructables. *The Mathematics of the Stewart Platform* [cit. 8.4.2024]. Dostupné z: <https://cdn.instructables.com/ORIG/FFI/8ZXW/I55MMY14/FFI8ZXWI55MMY14.pdf>.
- [2] R. Mukundan, *Advanced Methods in Computer graphics: Quaternions*. London: Springer, 2012.
- [3] K. M. Lynch, F. C. Park. *Modern robotics: mechanics, planning, and control*. Cambridge, England: Cambridge University Press, 2017.
- [4] A. Kurdila, P. Ben-tzvi. *Dynamics and control of robotic systems*. Hoboken, New Jersey: Wiley, 2020.
- [5] Acrome – acrome.net. *Understanding PID Control: 2-DOF Ball Balancer Experiments – acrome.net* [online]. [cit. 9.4.2024]. Dostupné z: <https://acrome.net/product/delta-robot>.
- [6] Acrome – acrome.net. *Delta Robot: 3-DoF Vision Guided Robotic Platform for Teaching and Research – acrome.net* [online]. [cit. 9.4.2024]. Dostupné z: <https://acrome.net/post/understanding-pid-control-using-2-dof-ball-balancer-experiments>.
- [7] J. -P. Merlet. *Solid mechanics and its applications: Parallel Robots (Second Edition)*. Dordrecht, The Netherlands: Springer, 2006.
- [8] A. Musa. *Ball balancing robot – aedmusa.com* [online]. [cit. 12.4.2024]. Dostupné z: <https://www.instructables.com/Ball-Balancing-Robot>.
- [9] A. Musa. *Ball Balancer – aedmusa.com* [online]. [cit. 12.4.2024]. Dostupné z: <https://www.aedmusa.com/projects/project-three-sng7y-gaslp>.
- [10] Autodesk, Inc – autodesk.com. *Instructables: 3DOF Ball on Plate Using Closed Loop Stepper Motors, instructables.com* [online]. ©2005-2024 Autodesk, Inc. [cit. 12.4.2024]. Dostupné z: <https://www.instructables.com/3DOF-Ball-on-Plate-Using-Closed-Loop-Stepper-Motor>.
- [11] K. -M. Lee, D. K. Shah, "Dynamic Analysis of a Three-Degrees-of-Freedom In-Parallel Actuated Manipulator," in *IEEE Journal of Robotics and Automation*, vol. 4, no. 3, pp. 361-367, June 1988.
- [12] CNRS Images - images.cnrs.fr. "*Orthoglide*" est un prototype de machine-outil rapide innovante, d'architecture parallèle. Sa conce - images.cnrs.fr [online]. [cit. 9.4.2024]. Dostupné z: https://images.cnrs.fr/en/photo/20030001_0691.
- [13] A. Weiss, R.G. Langlois, M. Hayes, "Improved Stewart Platform State Estimation using Inertial and Actuator Position Measurements," in *Control Engineering Practice* vol. 62, pp. 102-115, May 2017.
- [14] P. M. H. Van der Steen, "Ship Motion prediction for the Ampelmann System," *magisterská práce*, Delft University of Technology, Delft, The Netherlands, 2017.
- [15] ABB. *Robotics, Product specification, IRB 390*. Dec. 2022. [cit. 12. 2024]. Dostupné z: <https://search.abb.com/library/Download.aspx?DocumentID=3HAC066568-001&LanguageCode=en&DocumentPartId=&Action=Launch>.
- [16] S. Caro, D. Chablat, P. Lemoine, P. Wenger, "Kinematic Analysis and Trajectory Planning of the Orthoglide 5-axis," in *ASME*, August 2015.
- [17] T. Essomba, M. L. Laribi, S. Zeghloul, G. Poisson, "Optimal synthesis of a spherical parallel mechanism for medical application," in *Robotica* vol. 34, no. 3., pp. 671-686, March 2016.
- [18] MFS - Motion for Simulators s.r.l. - motionforsimulators.com. *2 DOF Motion Platforms Provide Pitch and Roll rotations, motionforsimulators.com* [online]. ©2024. [cit. 12.4.2024]. Dostupné z: <https://motionforsimulators.com/2dofs3dofmotionplatforms/>.
- [19] C. -C. Zhou, Y. -F. Fang, "Design and Analysis for a Three-Rotational-DOF Flight Simulator of Fighter-Aircraft," in *Chinese Journal of Mechanical Engineering*. vol. 31, no. 55., 2018.
- [20] Dassault Systèmes – solidworks.com. *3DExperience Solidworks Offers The Future of Design in Collaboration, solidworks.com* [online]. ©2002-2024 Dassault Systèmes - SolidWorks Corporation [cit. 13.4.2024]. Dostupné z: <https://www.solidworks.com/lp/3dexperience-solidworks-offers>.
- [21] Onshape – onshape.com. *Cloud-Native CAD & PDM* [online]. ©2014-2024 Onshape. [cit. 13.4.2024]. Dostupné z: <https://www.onshape.com/en/platform>.
- [22] The MathWorks, Inc. – mathworks.com. *Enable Simscape Multibody Link Plugin in SolidWorks - mathworks.com* [online]. ©1994-2024 The MathWorks, Inc.. [cit. 13.4.2024]. Dostupné z: <https://www.mathworks.com/help/smlink/ref/linking-and-unlinking-simmechanics-link-software-with-solidworks.html>.
- [23] The MathWorks, Inc. – mathworks.com. *Export an Autodesk Inventor Robot Assembly Model - mathworks.com* [online]. ©1994-2024 The MathWorks, Inc.. [cit. 13.4.2024]. Dostupné z: <https://www.mathworks.com/help/smlink/ug/export-robot-assembly-from-autodesk-inventor-software.html>.
- [24] Dassault Systèmes – solidworks.com. *SOLIDWORKS and SW Data Management System Requirements, solidworks.com* [online]. ©2002-2024 Dassault Systèmes - SolidWorks Corporation. [cit. 13.4.2024]. Dostupné z: <https://www.solidworks.com/support/system-requirements>.

- [25] HP Development Company, L.P. – support.hp.com. *HP ENVY x360 - 15-cn1065nr Product Specifications, support.hp.com* [online]. ©2024 HP Development Company, L.P.. [cit. 13.4.2024]. Dostupné z: <https://support.hp.com/us-en/document/c06401059>.
- [26] Autodesk, Inc – autodesk.com. *Autodesk Inventor: Mechanical design software for ambitious ideas, autodesk.com* [online]. ©2005-2024 Autodesk, Inc. [cit. 13.4.2024]. Dostupné z: <https://www.autodesk.com/products/inventor/overview?term=1-YEAR&tab=subscription>.
- [27] The MathWorks, Inc. – mathworks.com. *Simscape Multibody Import from Onshape, mathworks.com* [online]. ©1994-2024 The MathWorks, Inc.. [cit. 13.4.2024]. Dostupné z: <https://www.mathworks.com/products/simscape-multibody/onshape.html>.
- [28] Onshape – onshape.com. *Welcome to Onshape Help* [online]. ©2014-2024 Onshape. [cit. 13.4.2024]. Dostupné z: <https://cad.onshape.com/help/Content/EnterpriseHelp/Content/home.htm>.
- [29] Onshape – onshape.com. *ONSHAPE FORUMS* [online]. ©2014-2024 Onshape. [cit. 13.4.2024]. Dostupné z: <https://forum.onshape.com/>.
- [30] PTC Inc. – cad.onshape.com. *Onshape: Search in public.* [online]. ©2013-2024 PTC Inc. [cit. 13.4.2024]. Dostupné z: <https://cad.onshape.com/documents?nodeId=3&resourceType=filter>.
- [31] H. Guo, C. Zhao, B. Li, R. Liu, Z. Deng, J. Tian, "A Transformation Method to Generate the Workspace of an n(3RRS) Serial-Parallel Manipulator," in *Journal of Mechanical Design* vol. 141, no. 092301, pp. 1-7, September 2019.
- [32] H. Tetik, G. Kiper, "The 3-RRS Wrist: A New, Simple and Non-Overconstrained Spherical Parallel Manipulator," in *S. Zhegloul, L. Romdhane, M. Laribi, Computation Kinematics: Mechanism and Machine Science, Springer* vol. 50, July 2018.
- [33] Y. Du, R. Li, D. Li, S. Bai, "An ankle rehabilitation robot based on 3-RRS spherical parallel mechanism," in *Advances in Mechanical Engineering* vol. 9, no. 8., pp. 1-8, February 2017.
- [34] O. Carbajal-Espinosa, F. Izar-Bonilla, M. Díaz-Rodríguez, E. Bayro-Corrochano, "Inverse kinematics of a 3 DOF parallel manipulator: A conformal geometric algebra approach," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids), Cancun, Mexico*, pp. 766-771, November 2016.
- [35] Y. -H. Seo, "Design and Development of a 3-DOF Personal Motion Simulator," in *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 13, pp. 5808-5815. 2021.
- [36] M. Aminzadeh, A. Mahmoodi, M. Sabzehparvar, "Dynamic Analysis of a 3DoF Motion Platform," in *International Journal of Robotics*, vol. 1, no. 1, pp. 12-18. July 2009.
- [37] B. Gao, Z. Zhu, J. Zhao, L. Jiang, "Inverse Kinematics and Workspace Analysis of a 3 DOF Flexible Parallel Humanoid Neck Robot," in *J Intell Robot Syst*, no. 87, pp. 211-229. 2017.
- [38] L. Wenbao, L. Weijia, W. Xiao, G. Yan, W. Jinbo, "种三自由度运动平台机构的优化设计 přeloženo: Optimal construction of the mechanism of a 3DOF platform," in *China Mechanical Engineering*, vol. 22, no. 14, pp. 1728-1731. July 2011.
- [39] S. Casas, I. Coma, C. Portalés, M. Fernández, "Optimization of 3-DOF parallel motion devices for low-cost vehicle simulators," in *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 11, no. 2, pp. 1-17. April 2017.
- [40] Y. Lu, B. Hu, J.P. Yu, "Analysis of kinematics/statics and workspace of a 2(SP + SPR + SPU) serial-parallel manipulator," in *Multibody System Dynamics*, vol. 21, pp. 361-374. February 2009.
- [41] Y. Singh, V. Vinoth, M. Santhakumar, "Dynamic Modelling and Control of a 3-DOF Planar Parallel Robotic (XYθz Motion) Platform," in *International Conference on Advancer in Manufacturing and Materials Engineering AMME 2014, Procedia Materials Science*, vol. 5, pp. 1528-1539, 2014.
- [42] H. Wu, X. Lian, C. Peng, Y. Wei, X. Luo, B. Li, "Development of an Experimental Platform for 3-DOF Planar Parallel Robots," in *2022 5th World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM)*, 2022.
- [43] B. Ding, Y. Li, X. Xiao, Y. Tang, B. Li, "Design and analysis of a 3-DOF planar micromanipulation stage with large rotational displacement for micromanipulation system," in *Mechanical Sciences*, vol. 8, pp. 117-126. May 2017.
- [44] Z. Chen, J. Song, N. Li, W. Yan, C. Zhao, "Design and dynamics modeling of a novel 2R1T 3-DOF parallel motion simulator," in *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 45, no. 234, pp. 1-13. May 2023.
- [45] M. Aminzadeh, M. Sabzehparvar, "Model-Based Motion Tracking Control of an Electric 3DoF Parallel Motion Platform," in *2010 IEE Aerospace Conference, Big Sky, Montana, USA*, pp. 1-8. 2010.
- [46] R. Desai, M. Sreeksumar "A Forward, Inverse Kinematics and Workspace Analysis of 3RPS and 3RPS-R Parallel Manipulators," in *Iranian Journal of Science and Technology Transactions of Mechanical Engineering*, vol. 45, pp. 115-131. February 2020.
- [47] S. Staicu, "Inverse Dynamics of the Spatial 3-RPS Parallel Robot," in *Proceedings of the Romanian Academy, Series A*, vol. 13, no. 1, pp. 62-70. 2012.
- [48] J. Schadlbauer, D. R. Walter, M. L. Husty, "A Complete Kinematic Analysis of the 3-RPS Parallel Manipulator," in *15th National Conference on Machines and Mechanisms*, no. 121. 2011.

- [49] A. Nayak, S. Caro, P. Wenger, "Kinematic Analysis of the 3-RPS-3-SPR Series-Parallel Manipulator," in *Robotica*, vol. 37, no. 7, pp. 1240-1266. 2019.
- [50] K. A. A. Ali, Y. Liu, "Position Analysis Of 3-DOF 3-RPS Parallel Manipulator," in *International Conference on Advanced Material Science and Environmental Engineering (AMSEE 2016)*, pp. 128-131. 2016.
- [51] O. A. M. Abatan, "Balancing Ball," in *report of project*, Tshwane University of Technology, Pretoria, Jihoafrická republika, 2011.
- [52] M. M. Kopichev, A. V. Putov, A. N. Pashenko, "Ball on the plate balancing control system," in *IOP Conference Series: Materials Science and Engineering*, vol. 634, no. 012004, pp. 1-6. 2019.
- [53] M. Oravec, A. Jadlovska, *Design of virtual models of mechatronics systems with Simulink 3D animation toolbox* [online]. Technical University of Košice, Faculty of Electrical Engineering and Informatics, 2014. [cit. 16. dubna 2024]. Dostupné z: http://147.232.25.194/laboratoria/ludia/pdf/Oravec_Jadlovska_TC_2014.pdf.
- [54] B. Dasgupta, T. S. Mruthyunjaya, "The Stewart platform manipulator," in *Mechanism and Machine Theory*, vol. 35, pp. 15-40. 2000.
- [55] Z. Bingul, O. Karahan, "Dynamic Modeling and Simulation of Stewart Platform", in *S. Küçük, Serial and Parallel Robot Manipulators: Kinematics, Dynamics, Control and Optimization, InTechOpen*, 2012.
- [56] K. V. Sancak, "High Precision Motion Control of Mechatronic Systems in Presence of General Uncertainties: Application with a Heavy-Duty Parallel Robot," *disertační práce*, Istanbul Technical University, Istanbul, Turkey, 2022.
- [57] E. Thöndel, "A 6DOF Motion Platform with Permanent Magnet Linear Motors," in *Transactions on Electrical Engineering*, vol. 3, no. 4 pp. 101-105. 2014.
- [58] A. Şumnu, İ. H. Güzelbey, M. V. Çakır, "Simulation and PID control of Stewart platform with linear motor," in *Journal of Mechanical Science and Technology*, vol. 31, no. 1 pp. 345-356. 2017.
- [59] M. Furqan, M. Suhaib, N. Ahmad, "Studies on Stewart platform manipulator: A review," in *Journal of Mechanical Science and Technology*, vol. 31, no. 9 pp. 4459-4470. 2017.
- [60] P. -H. Pham, D. -H. Doan, H. -N. Vo, V. -T. Dang, H. -Q. -B. Nguyen, T. -V Le, T. -T, -Q, Truong, T, -D, Tran "Ball Balancing on 3-RRS Parallel Manipulator Using PD and LQRT Control," in *Robotica & Management*, vol. 28, no. 2, pp. 38-43. 2023.
- [61] M. Eynian, *Inverse Kinematics of a 3RPS Mechanism* [online]. University West, Trollhättan, Sweden, March 2022. [cit. 18. dubna 2024]. Dostupné z: https://www.researchgate.net/publication/332682139_Inverse_Kinematics_of_a_3RPS_Mechanism#fullTextFileContent
- [62] Arduino S.r.l. – arduino.cc. *Arduino Due – store.arduino.cc* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://store.arduino.cc/products/arduino-due>.
- [63] ECLIPSE s.r.o. – dratek.cz. *CNC Shield 3D tiskárny pro Arduino – dratek.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://dratek.cz/arduino/1132-cnc-shield-pro-3d-tiskarny-pro-arduino.html>.
- [64] Evolution – levna-loziska.cz. *Kuličky – levna-loziska.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.levna-loziska.cz/kategorie/kulicky>.
- [65] NetDirect – gme.cz. *Nema 17 DC motor krokový 5V/2A pro tiskárny – gme.cz* [online]. ©2004-2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.gme.cz/v/1500760/nema-17-dc-motor-krokovy-5v-2a-pro-tiskarny>.
- [66] NetDirect – gme.cz. *1.8° 42mm Hybrid Stepper Motor – Nema17: JK42HS40-1704 – gme.cz* [online]. ©2004-2024. [cit. 15. dubna 2024]. Dostupné z: https://img.gme.cz/files/eshop_data/eshop_data/11/775-138/dsh.775-138.1.pdf.
- [67] Mouser Electronics, Inc. – cz.mouser.com. *NKK Switches FTAS00-121AS4 – cz.mouser.com* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://cz.mouser.com/ProductDetail/NKK-Switches/FTAS00-121AS4?qs=W43f9fqXGCiHrUmf3gIKAg%3D%3D>.
- [68] NKK SWITCHES OF AMERICA, INC. - nkkswitches.com. *Series FT, 4-Wire Touch Screens with Printed Tails, nkkswitches.com* [online]. [cit. 15. dubna 2024]. Dostupné z: <https://www.nkkswitches.com/pdf/4-wireTouch.pdf>.
- [69] NKK Switches CO., LTD. – nkkswitches.com.hk. *FTAS00-12.1AS-4: Touch Screens – nkkswitches.com.hk* [online]. [cit. 15.4.2024]. Dostupné z: <https://www.nkkswitches.com.hk/products/Touch%20Screens/Touch%20Screens%20FT/FTAS00-12.1AS-4/>
- [70] VÝTVARNÝ-SHOP.CZ – vytvarny-shop.cz. *Transparentní papír A4 115 g bílý – vytvarny-shop.cz*. [cit. 15. dubna 2024]. Dostupné z: <https://www.vytvarny-shop.cz/zbozi/transparentni-papir-a4-115-g-bily/>.
- [71] Josef Hanzal – adafruit.com. *Resistive Touch Screen Controller – STMPE610 – adafruit.com* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <http://www.snailshop.cz/shieldy-lcd/1497-resistive-touch-screen-controller-stmpe610.html>.

- [72] Adafruit – adafruit.com. *Resistive Touch Screen Controller – STMPE610 – adafruit.com* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.adafruit.com/product/1571>.
- [73] LaskaKit – laskakit.cz. *DRV8825 driver pro krokové motory – laskakit.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.laskakit.cz/drv8825-driver-pro-krokovye-motory/>.
- [74] Etsy, inc. – etsy.com. *Round D6 PCB Keyboard Switches Pack of 10 Various Colours – etsy.com* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.etsy.com/uk/listing/684675613/round-d6-pcb-keyboard-switches-pack-of>.
- [75] Official Electronic, s.r.o. – official.cz. *LED Kingbrigt L-483YDT – official.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.official.cz/z3612-led-kingbrigt-l-483ydt>.
- [76] HWKITCHEN – hwkitchen.cz. *Rezistor 220Ω 0,5W – hwkitchen.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.hwkitchen.cz/rezistor-220ohm-0-5w/#:~:text=Rezistor%20220%20CE%A9%20poslou%C5%BE%C3%AD%20dob%C5%99e.rel%C3%A9%2C%20DC%20motor%20nebo%20solenoid>.
- [77] Conrad Electronic Česká republika, s.r.o. – conrad.cz. *Rademacher WR-Typ 941 eurodeska epoxid (d x š) 160 mm x 100 mm 35 μm Rastr (rozteč) 2.54 mm Množství 1 ks – conrad.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.conrad.cz/cs/p/rademacher-wr-typ-941-eurodeska-epoxid-d-x-s-160-mm-x-100-mm-35-m-rastr-roztec-2-54-mm-mnozstvi-1-ks-529568.html>.
- [78] LaskaKit – laskakit.cz. *Dupont propojovací kabel 40žil MD samec-samice (20 cm) – laskakit.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.laskakit.cz/dupont-propojovaci-kabely-m-f-40ks-samec-samice--20cm/>.
- [79] Plexiplast – plexiplast.cz. *Plexisklo XT (extrudované plexi) – plexiplast.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.plexiplast.cz/extrudovane-plexisklo>.
- [80] Plexiplast – plexiplast.cz. *CNC - frézování a gravírování plastů, tvarové obrábění plastů plexiplast.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.plexiplast.cz/frezovani-a-gravirovani>.
- [81] Amazon – amazon.com. *M5 stainless steel ball, perforated tapping steel ball, threaded handle ball, hand-tightened screw ball-15mm * m5 * 5pcs – amazon.com* [online]. ©1996 - 2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.amazon.co.uk/stainless-perforated-tapping-threaded-hand-tightened/dp/B09HYKS78T?th=1>.
- [82] Prumex s.r.o. - spojovací-material.net. *Spojovací materiál: Vyhledejte produkt podle parametrů – spojovací-material.net* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <http://www.spojovaci-material.net/>.
- [83] Arkov, spol., s.r.o. – arkov.cz. *625 SKF kuličkové ložisko – arkov.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.arkov.cz/p/625-kulickove-lozisko-skf-18285>.
- [84] Stoklasa textilní galerie s.r.o. – stoklasa.cz. *Plst'/filc – stoklasa.cz* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.stoklasa.cz/plst-filc-x2s01843>.
- [85] SparkFun Electronics – sparkfun.com. *HantouchUSA, How it works: 4 -Wire Analog-Resistive Touch Screens – sparkfun.com* [online]. ©2024. [cit. 15. dubna 2024]. Dostupné z: <https://www.sparkfun.com/datasheets/LCD/HOW%20DOES%20IT%20WORK.pdf>.
- [86] Lesics. In YouTube: *How does a Stepper Motor work? – youtube.com* [online]. Uploaded: 19. října 2016. [cit. 22. dubna 2024]. Dostupné z: <https://www.youtube.com/watch?v=eyqwlIowZiU>.
- [87] How To Mechatronics. In YouTube: *Stepper Motors and Arduino – The Ultimate Guide – youtube.com* [online]. Uploaded: 15. května 2022. [cit. 22. dubna 2024]. Dostupné z: <https://www.youtube.com/watch?v=7spK BkMjys>.
- [88] The Engineering mindset. In YouTube: *How Stepper Motors Work – Electric motor – youtube.com* [online]. Uploaded: 26. června 2022. [cit. 22. dubna 2024]. Dostupné z: <https://www.youtube.com/watch?v=7spK BkMjys>.
- [89] M. Apetaur, J. Soukup. *Mechatroniky: Přehled Problematiky pro Strojáře* [online]. Univerzita J. E. Purkyně v Ustí nad Labem, Fakulta Výrobních Technologií a Managmentu, 2014 [cit. 24.4.2024]. Dostupné z: <https://ezak.ujep.cz/document/6798/3d11669cd645182a2946fc8812d6eb34-publikace-1-mechatronika-text-v1-pdf>.
- [90] Fritzing GmbH – fritzing.org. *Fritzing is devoted to making creative use of electronics accessible to everyone – fritzing.org* [online]. ©2024. [cit. 23.4.2024]. Dostupné z: <https://fritzing.org/download/>.
- [91] K. Bürüncük, Y. Tokad, "On the Kinematic of a 3-DOF Stewart Platform," in *Journal of Robotic Systems*, vol. 16, no. 2, pp. 105-118. 1999.
- [92] M. J. Hayawi, M. A. Anuar, A. Azlan, A. R. Omar, "Research Article Kinematic and Dexterity Analysis of a 3-DOF Parallel Manipulator," in *Research Journal of Applied Sciences, Engineering and Technology*, vol. 12, no. 2, pp. 239-248. January 2016.
- [93] K. -M. Lee, D. K. Shah, "Dynamic Analysis of a Three-Degrees-of-Freedom In-Parallel Actuated Manipulator," in *IEEE Journal of Robotics and Automation*, vol. 4, no. 3, pp. 361-367. June 1988.
- [94] J. Gallardo-Alvarado, "An Application of the Newton-Homotopy Continuation Method for Solving the Forward Kinematic Problem of the 3-RRS Parallel Manipulator," in *Hindawi: Mathematical Problems in Engineering*, vol. 2019, pp. 1 - 6. January 2019.

- [95] H. Tetik, R. Kalla, G. Kiper, S. Bandyopadhyay, "Position kinematic of a 3-RRS Parallel Manipulator," in Parenti-Castelli, V., Schiehlen, W. (eds) *ROMANSY 21 - Robot Design, Dynamics and Control. ROMANSY21 2016. CISM International Centre for Mechanical Sciences*, vol. 569. Springer, Cham. 2016.
- [96] Z. Huang, W. S. Tao, Y. F. Fang, "Study on the Kinematic Characteristics of 3 DOF In-Parallel Actuated Platform Mechanisms," in *Mechanical Machine Theory*, vol. 31, no. 8, pp. 999-1007. 1996.
- [97] R. Desai, M. Sreekumar, "A Forward, Inverse Kinematics and Workspace Analysis of 3RPS and 3RPS – R Parallel Manipulators," in *Iranian Journal of Science and Technology Transactions of Mechanical Engineerig*, vol. 45, pp. 115-131. 2021.
- [98] T. Ono, R. Eto, J. Yamakawa, H. Murakami, "Analysis and control of a Stewart platform as base motion compensators - Part I: Kinematics using moving frames," in *Nonlinear Dynamics*, vol. 107, pp. 51-76. 2022.
- [99] T. Geike, J. McPhee, "Inverse dynamic analysis of parallel manipulators with full mobility," in *Mechanism and Machine Theory*, vol. 38, pp. 549-562. 2003.
- [100] R. V. V. Petrescu, R. Aversa, A. Apicella, M. Mirsayar, S. Kozaitis, T. Abu-Lebdeh, F. I. T. Petrescu, "Inverse Kinematics of a Stewart Platform," in *Journal of Mechatronics and Robotics*, vol. 2, no. 1, pp. 45 - 59. 2018.
- [101] S. M. LaValle, *Planning Algorithms*. University of Illinois: Cambridge University Press, 2006.
- [102] The MathWorks, Inc. – mathworks.com. *Train SAC Agent for ball balance control*, mathworks.com [online]. ©1994-2024 The MathWorks, Inc.. [cit. 13.4.2024]. Dostupné z: <https://www.mathworks.com/help/reinforcement-learning/ug/train-sac-agent-for-ball-balance-control.html>.
- [103] The MathWorks, Inc. – mathworks.com. *File Exchange: Steve Miller (2024). Simscape Multibody Contact Forces Library*, mathworks.com [online]. ©1994-2024 The MathWorks, Inc.. [cit. 13.4.2024]. Dostupné z: <https://www.mathworks.com/matlabcentral/fileexchange/47417-simscape-multibody-contact-forces-library>.
- [104] M. L. Nagurka, S. Huang, "A Mass-Spring-Damper Model of a Bouncing Ball," in *International Journal of Engineering Education*, vol. 22, no. 2, pp. 393-401. 2006.
- [105] V. L. Popov, M. Heß, E. Willert, *Handbook of Contact Mechanics: Exact Solutions of Axisymmetric Contact Problems*. Springer – Verlag GmbH, Germany, 2018.
- [106] P. Müller, R. Bötcher, A. Russel, M. Trüe, S. Aman, J. Tomas, "Contact time at impact of spheres on large thin plates," in *Advanced Powder Technology*, vol. 27, pp. 1233-1243. 2016.
- [107] Z.H. Jiang, Z. J. Liang, D. W. Zhou, Y. J. Deng, "Dissipative properties for a ball bouncing on a vertically vibrating plate," in *Physica A*, vol. 548, pp. 1-9. 2020.
- [108] D.D. Fuller. *Coefficients of Friction* [online]. Columbia University, 1968. Dostupné z: <https://web.mit.edu/8.13/8.13c/references-fall/aip/aip-handbook-section2d.pdf>.
- [109] H. Bang, Y. S. Lee, "Embedded Model Predictive Control for Enhancing Tracking Performance of a Ball and Plate System," in *Journal of Latex Class Files*, vol. 1, no. 1, pp. 1-8. August 2018.
- [110] J. Tamimi, "A comparative simulation and experimental study for control of a ball and plate system using model-based controllers," in *International Journal of Dynamical Systems and Differential Equations*, vol. 13, no. 2, pp. 91-107. January 2023.
- [111] M. Oravec, A. Jadlovska, "Intelligent Positioning Plate Predictive Control and Concept of Diagnosis System Design," in *Journal of Manufacturing and Industrial Engineering (MIE)*, vol. 15, no. 1-2, pp. 1-9. June 2017.
- [112] M. Oravec, A. Jadlovska, "Model Predictive Control of a Ball and Plate laboratory model," in *2015 IEEE 13th International Symposium on Applied Machine Intelligence and Informatics (SAMII)*, Herl'any, Slovensko, pp. 165-170. 2015.
- [113] A. Kastner, J. Inga, T. Blauth, F. Köpf, M. Flad, S. Hohmann, "Model-Based Control of a Large-Scale Ball-on-Plate System With Experimental Validation," in *2019 IEEE International Conference on Mechatronics (ICM)*, Ilmenau, Germany, pp. 257-262. 2019.
- [114] J. Kumar, "Design and control of ball on plate system," in *International Science Press*, vol. 9, no. 34, pp. 765-778. 2016.
- [115] A. Mohammadi, J. -C. Ryu, "Neural network-based PID compenstation for nonlinear systems: ball-on-plate example," in *International Journal of Dynamics and Control*, vol. 8, pp. 178-188. 2020.
- [116] X. Li, X. Gao, L. Sun, D. Zheng, H. Shi, C. Lei, L. Hu, Y. Zong, "The adaptive neural network fuzzy sliding mode control for the 3-RRS parallel manipulator," in *Advances in Mechanical Engineering*, vol. 14, no. 9, pp. 1-13. 2022.
- [117] M. Lashin, M. Fanni, M. Magdy, A. M. Mohamed, "PD Type of Fuzzy Controller for a New 3DOF Fully Decoupled Translational Manipulator," in *2016 The 2nd International Conference on Control, Automation and Robotics (ICCAR)*, Hong Kong, China, pp. 263-267. 2016.
- [118] G. Chaudhary, J. Ohri, "3-DOF Parallel manipulator control using PID controller," in *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, Delhi, India, pp. 1-6. 2016.

- [119] M. Vallés, M. Díaz-Rodríguez, Á. Valera, V. Mata, "Mechatronic Development and Dynamic Control of a 3-DOF Parallel Manipulator," in *Mechanics Based Design of Structures and Machines*, vol. 40, no. 4, pp. 434-452. 2012.
- [120] A. Dumlu, K. Erenturk, "Trajectory Tracking Control for a 3-DOF Parallel Manipulator Using Fractional-Order PID Control," in *IEEE Transactions on Industrial Electronics*, vol. 61, no. 7, pp. 3417-3426. 2014.
- [121] X. -M. Niu, G. -Q. Gao, X. -J. Liu, Z. -D. Bao, "Dynamics and Control of a Novel 3-DOF Parallel Manipulator with Actuation Redundancy," in *International Journal of Automation and Computing*, vol. 10, no. 6, pp. 552-562. December 2013.
- [122] K. Zarzycki, M. Ławryńczuk, "Fast Real-Time Model Predictive Control for a Ball-on-Plate Process," in *Sensors*, vol. 21, no. 12, pp. 3959. June 2021.
- [123] E. Fabregas, S. Dormido-Canto, S. Dormido, "Virtual and Remote Laboratory with the Ball and Plate System," in *IFAC PapersOnLine*, vol. 50, no. 1, pp. 9132-9137. 2017.
- [124] Wen Yu and F. Ortiz, "Stability analysis of PD regulation for ball and beam system," in *Proceedings of 2005 IEEE Conference on Control Applications (CCA)*, Toronto, Canada, pp. 517-522, 2005.
- [125] A. Kassem, H. Haddad, C. Albitar, "Comparison Between Different Methods of Control of Ball and Plate System with 6DOF Stewart Platform," in *IFAC PapersOnLine*, vol. 48, no. 11, pp. 47-52. 2015.
- [126] M. A. Rastin, S. A. Moosavian, E. Talebzadeh, M. Alaeddin, "Trajectory Tracking and Obstacle Avoidance of a Ball and Plate System Using Fuzzy Theory," in *2013 13th Iranian Conference on Fuzzy Systems (IFSC)*, Qazvin, Iran, pp. 1-5, August 2013.
- [127] H. Jafari, M. Pourrahim, A. Rahimpour, F. Hashemzadeh, "Linear Quadratic Gaussian Control for ball and plate system," in *2012 International conference on computer, control, education and management*, Dubai, United Arab Emirates, 2012.
- [128] Z. Richter, "Nonlinear Model Development and Validation for Ball and Plate Control System," *magisterská práce*, Faculty of California Polytechnic State University, San Luis Obispo, USA, 2021.
- [129] R. Tedrake. *Underactuated Robotics: Chapter 8 Linear Quadratic Regulators* [online]. MIT, Department of Electrical Engineering and Computer Science (EECS), 2016. [cit. 17. dubna 2024]. Dostupné z: <https://underactuated.mit.edu/lqr.html>.
- [130] L. Spaček, J. Vojtěšek, F. Gazdoš, T. Kadavý, "Ball & Plate model for robotic system," in *Proceedings – European Council for Modeling and Simulation, EXMS*, Wilhelmshaven, Germany, pp. 226-231. 2018.
- [131] F. Dušek, D. Honc, R. K. Sharma, "Modelling of Ball and Plate System Based on First Principle Model and Optimal Control," in *2017 21st International Conference on Process Control (PC)*, Štrbské Pleso, Slovakia, pp. 216-221. June 2017.
- [132] D. Kan, B. Xing, W. Xie, L. Zhu, "A minimum phase output based tracking control of ball and plate systems," in *International Journal of Dynamics and Control*, vol. 10, pp. 462-472. 2022.
- [133] D. Debono, M. Bugeja, "Application of Sliding Mode Control to the Ball and Plate problem," in *Proceedings of the 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO - 2015)*, pp. 412-419. 2015.
- [134] M. Moarref, M. Saadat, G. Vossoughi, "Mechatronic Design and Position Control of a Novel Ball and Plate system," in *16th Mediterranean Conference on Control and Automation*, Ajaccio, France, pp. 1071-1076. June 2008.
- [135] The MathWorks, Inc. – mathworks.com. *Foundations of Fuzzy Logic*, mathworks.com [online]. ©1994-2024 The MathWorks, Inc.. [cit. 14.4.2024]. Dostupné z: <https://www.mathworks.com/help/fuzzy/foundations-of-fuzzy-logic.html>.
- [136] B. Kempers, "Offset-free MPC for a ball-balancing Stewart platform," *magisterská práce*, Eindhoven University of Technology, Eindhoven, The Netherlands, 2013.
- [137] The MathWorks, Inc. – mathworks.com. *Choose Sample Time and Horizons*, mathworks.com [online]. ©1994-2024 The MathWorks, Inc.. [cit. 14.4.2024]. Dostupné z: <https://www.mathworks.com/help/mpc/ug/choosing-sample-time-and-horizons.html>.
- [138] F. D. Cero, "Ball and Plate MPC Control of a 6 DOF Stewart Platform," *magisterská práce*, Eindhoven University of Padova, Padova, Italy, 2022.
- [139] The MathWorks, Inc. – mathworks.com. *quadprog: Quadratic programming*, mathworks.com [online]. ©1994-2024 The MathWorks, Inc.. [cit. 14.4.2024]. Dostupné z: <https://www.mathworks.com/help/optim/ug/quadprog.html>.
- [140] The MathWorks, Inc. – mathworks.com. *Simulink Support Package for Arduino Hardware*, mathworks.com [online]. ©1994-2024 The MathWorks, Inc.. [cit. 14.4.2024]. Dostupné z: <https://www.mathworks.com/matlabcentral/fileexchange/40312-simulink-support-package-for-arduino-hardware>.
- [141] The MathWorks, Inc. – mathworks.com. *Simulink Coder: Generate C and C++ from Simulink and Stateflow models*, mathworks.com [online]. ©1994-2024 The MathWorks, Inc.. [cit. 13.4.2024]. Dostupné z: <https://www.mathworks.com/products/simulink-coder.html>.

PŘÍLOHA A: PŘÍKLAD SEZNAMU SYMBOLŮ A ZKRATEK

A.1 Seznam symbolů

g ($\text{m} \cdot \text{s}^{-2}$)	tíhové zrychlení (pro Prahu $g = 9,81373 \text{ m} \cdot \text{s}^{-2}$)
π (–)	Ludolfovo číslo (matematická konstanta $\pi = 3,141596 \dots$)
v ($\text{m} \cdot \text{s}^{-1}$)	rychlost (ve vztahu s úhlovou rychlostí a poloměrem kružnice $v = \omega \cdot r$)
l (–)	počet pevných těles v mechanismu včetně podstavy
n (–)	celkový počet kloubů v mechanismu
d_j (–)	počet stupňů volnosti j -tého kloubu ($j = 1, 2, 3 \dots n$)
m (kg)	hmotnost
F (N)	síla
U_{ref} (V)	referenční napětí driveru (měřeno mezi kovovým pouzdem potenciometru a pinem vpravo dole na driveru GND)
I_{fN} (A)	jmenovité fázové napětí krokového motoru (z technické dokumentace NEMA17)
τ (s)	časová konstanta (doba do dosažení 63,2 % z ustálené hodnoty)
s (–)	komplexní proměnná po Laplaceově transformaci
z (–)	proměnná Z-transformace (diskrétní transformace)
$\ \mathbf{x}\ $ (m)	Euklidovská norma (druhá odmocnina ze součtu kvadrátů vzdáleností jednotlivých souřadnic od zvoleného počátku)
$\ \mathbf{x} - \mathbf{y}\ $ (m)	Euklidovská vzdálenost (vzdálenost dvou bodů ve všech souřadnicích)
i (–)	proměnná číslování nohou, bodů ($i = 1, 2, 3$)
p_s (m)	vzdálenost bodů P_i od S_{P_i} ($\ P_i - S_i\ $)
p'_s (m)	vzdálenost bodů P'_i od $S_{P'_i}$ ($\ P'_i - S'_i\ $)
b_s (m)	vzdálenost bodů B_i od S_{B_i} ($\ B_i - S_i\ $)
u (m)	délka kliky
w (m)	délka táhla do středu kuličky na čepu
a_i (m)	Euklidovská vzdálenost bodů P'_i od bodu B_i ($\ P'_i - B_i\ $)
b_i (m)	vzdálenost bodů P'_i od bodu B_i bez svislé osy ($\ P'_i - B_i\ _{x,y}$)
c (m)	Euklidovská vzdálenost bodů P_i a P'_i ($\ P_i - P'_i\ $)
χ_i ($^\circ$, rad)	úhel mezi a_i a svislou osou z
ξ_i ($^\circ$, rad)	úhel mezi a_i a w
ς_i ($^\circ$, rad)	úhel mezi a_i a u
κ_i ($^\circ$, rad)	úhel mezi svislou osou z a u
K_P (–)	člen zesílení ve vybrané formě PID regulátoru (P-složka = proporcionální člen)
K_I (–)	integrační člen ve vybrané formě PID regulátoru (I-složka)
K_D (–)	derivační člen ve vybrané formě PID regulátoru (D-složka)
K_N (–)	filtrační člen ve vybrané formě PID regulátoru (N-složka filtruje D-složku)
T_s (s)	simulační perioda, vzorkovací doba (<i>sample time</i>)
J ($\text{kg} \cdot \text{m}^2$)	moment setrvačnosti
J_C (–)	penalizační funkce (cenová funkce, <i>cost function</i>)
$x_k = x(k)$ (–)	k -tý vzorek ($k \in \mathbb{N}$) pro diskrétní popis proměnné v čase
\times	maticové, vektorové násobení, pokud není mezi maticemi, vektory explicitně naznačen skalární součin tečkou, je uvažován jako výchozí maticový, vektorový součin (<i>cross product</i>)
\cdot	skalární násobení, pro matice, vektory jen členy mezi sebou (<i>dot product</i>)
$\dot{x}, \ddot{x} \dots$	označení časové derivace veličiny (počet teček je řád derivace $dx/dt, d^2x/dt^2 \dots$)
R^T, \mathbf{r}^T	transpozice matice, vektoru (převrácení prvků dle hlavní diagonály, sloupce se změň na řádky a naopak)
R, \mathbf{r}	označení matice, vektoru tučně (Bold)

R^{-1}	inverzní matice (musí splňovat podmínky pro inverzi čtvercová, regulární...)
$f(x)$	vektor jako funkce proměnné
2×3	dimenze, rozměry matice počet řádků x sloupců (v ukázce dva řádky, tři sloupce)
A	stavová matice – matice systému – vnitřní vazby systému
B	matice vstupu – matice řízení – vnější vazby systému na vstup
C	matice výstupu – vazby výstupu na stav
D	dopředná (průchozí) matice – vazby vstupu na výstup (často nulová matice)
A_d, B_d, C_d, D_d	diskrétní formy matic stavu, vstupu, výstupu a dopředné matice
x	stavový vektor
y	vektor výstupu
u	vektor vstupu (řízení)
Q_{LQR}	pozitivně semidefinitní matice definovaná pro penalizaci stavů v LQR
R_{LQR}	pozitivně definitní matice definovaná pro penalizace řízení v LQR
S_{LQR}	matice jako řešení Riccatiho rovnice v LQR
K_{LQR}	výsledná optimalizovaná matice zesílení regulátoru LQR
xy	obecný planární ortogonální systém
$x'y'$	rotovaný planární ortogonální systém
ξ	úhel rotace rotovaného systému vůči rotovanému
xyz	obecný prostorový ortogonální systém
$x'y'z'$	rotovaný prostorový ortogonální systém
x_t	translační pohyb ve směru osy x
y_t	translační pohyb ve směru osy y
z_t	translační pohyb ve směru osy z
θ	rotační úhel kolem osy x
ϕ	rotační úhel kolem osy y
ψ	rotační úhel kolem osy z
$\check{s} \times v \times h$	rozměry uváděny ve tvaru \check{s} – šířka, v – výška, h – hloubka vždy s příslušnou rozměrovou jednotkou (m - metry, cm - centimetry, mm - milimetry)
$x_{ref} = x_{\check{z}ad}$	referenční, žádaná poloha v ose x
$y_{ref} = y_{\check{z}ad}$	referenční, žádaná poloha v ose y
$x_{m\check{e}r}$	měřená poloha v ose x
$y_{m\check{e}r}$	měřená poloha v ose y
$e_x; e_y$	odchylka od žádané polohy v dané ose ($e_x = x_{\check{z}ad} - x_{m\check{e}r}$; $e_y = y_{\check{z}ad} - y_{m\check{e}r}$)

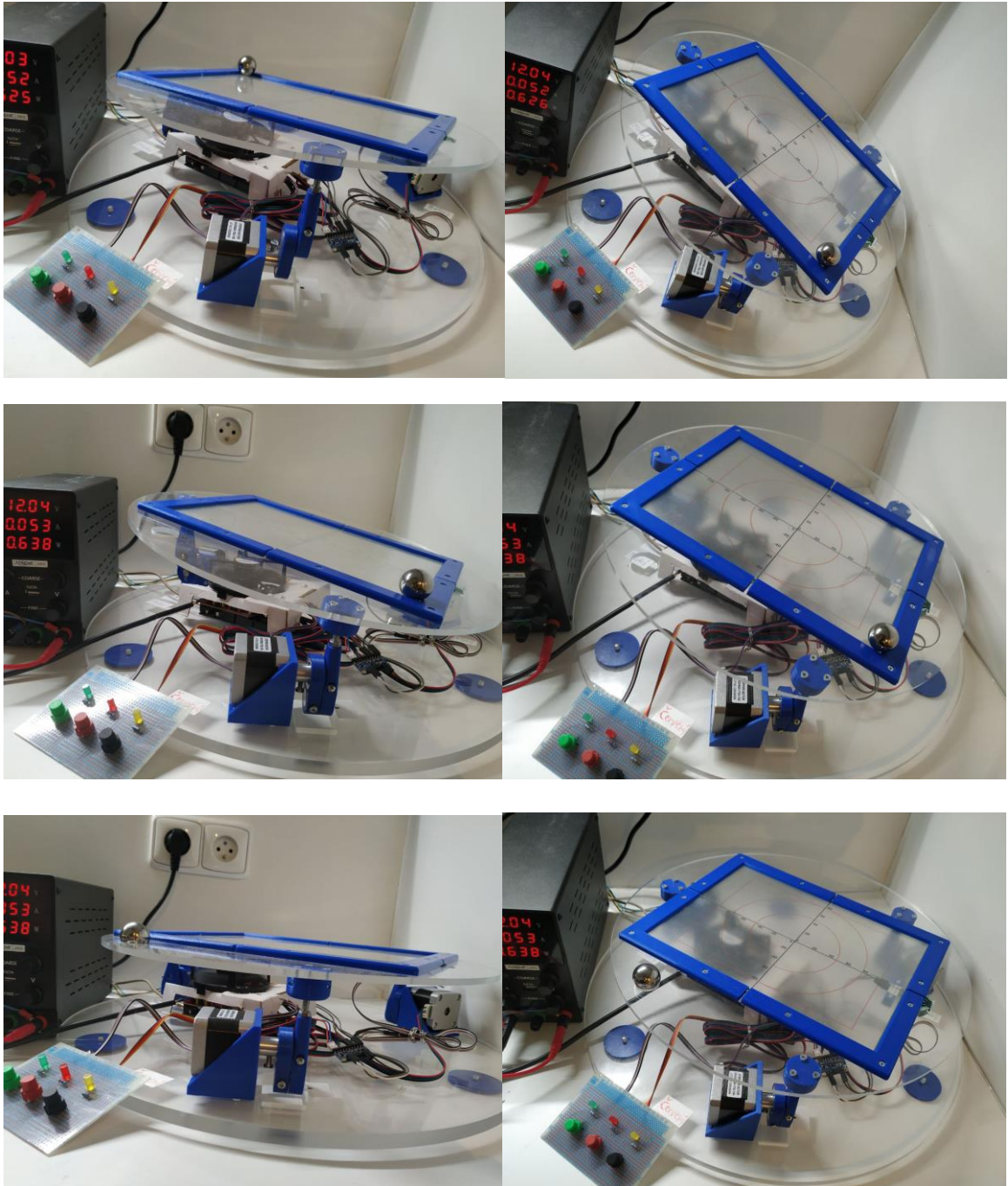
A.1.1 Seznam zkratk

R	rotační kloub (<i>revolute joint</i>)
S	sférický kloub (<i>spherical joint</i>)
P	posuvný kloub (<i>prismatic joint</i>)
U	Kardanův kloub (<i>universal joint</i>)
C	válcový kloub (<i>cylindrical joint</i>)
H	šroubový kloub (<i>helical joint</i>)
P_a	rovnoběžníkový kloub (<i>parallelogram joint</i>)
DOF	stupně volnosti (<i>degrees of freedom</i>)
3-RRS	mechanismus se třemi kinematickými řetězci (nohama), každá noha se dvěma rotačními klouby a jedním sférickým kloubem
2DOFp	plošina o dvou stupních volnosti
3DOFp	plošina o třech stupních volnosti
6DOFsp	Stewartova platforma o šesti stupních volnosti
3D	tři dimenze (často používané ve spojitosti 3D tisk, tedy v prostoru)
CNC	číslicově řízený stroj pro výrobu součástek (<i>computer numerical control</i>)

SW	software pro modelování mechanický celků (<i>SolidWorks</i>)
VR	virtuální realita (<i>virtual reality</i>)
CAD	počítačem podporované projektování (<i>computer aided design</i>)
I/O	obecně vstupy a výstupy mikroprocesoru (<i>inputs/outputs</i>)
PID	regulátor s <u>p</u> roporcionální, <u>i</u> ntegrační a <u>d</u> erivační složkou (<i>Proportional, Integral, Derivative</i>)
PD	regulátor s <u>p</u> roporcionální a <u>d</u> erivační složkou (<i>Proportional, Derivative</i>)
IMC	metoda ladění PID regulátorů využívající modelu procesu (<i>Internal Model Control</i>)
MPC	modelově prediktivní řídicí metoda (<i>Model Predictive Control</i>)
LQR	lineárně kvadratický regulátor (<i>Linear-Quadratic Regulator</i>)
FLC	regulátor založen na matematickém aparátu fuzzy logiky (<i>Fuzzy Logic Control</i>)
SMC	klouzavé řízení (<i>Sliding Mode Control</i>)
M _X	metrický závit dle normy ČSN ISO 261 ($x = 1; 1,2; 1,6; 1,8; 2; 2,5; 3; 3,5; 4; 5 \dots$)
Ons	Onshape – používaný software pro vytváření modelu
ABS	akrylonitril butadien styren – plast používaný pro 3D tisk
PETG	glykolem modifikovaný polyethylen terephthalat – odolnější plast pro 3D tisk
PMMA	polymethylmethakrylát – plexisklo (akrylátové sklo) pro CNC frézování
X _{RI}	signál osy x vpravo rezistivního dotykového panelu (right)
X _{LE}	signál osy x vlevo rezistivního dotykového panelu (left)
Y _{UP}	signál osy y nahoře rezistivního dotykového panelu (up)
Y _{LO}	signál osy y dole rezistivního dotykového panelu (low)
X+	signál osy x v kladném směru na kontroléru (hodnoty se znaménkem +)
X-	signál osy x v záporném směru na kontroléru (hodnoty se znaménkem -)
Y+	signál osy y v kladném směru na kontroléru (hodnoty se znaménkem +)
Y-	signál osy y v záporném směru na kontroléru (hodnoty se znaménkem -)
PNP	bipolární tranzistor dvě vrstvy typu P s převahou děr a jedna vrstva typu N s převahou elektronů (positive, negative, positive)
NPN	bipolární tranzistor jednou vrstvou typu P s převahou děr a dvěma vrstvami typu N s převahou elektronů (negative, positive, negative)
A/D	analogově – digitální převodník signálu, z napětí na hodnotu (analog to digital)
USB	Výrobní standard pro komunikační protokol elektroniky (Universal Serial Bus)
A IDE	Arduino IDE – otevřená softwarová platforma pro psaní kódu a nahrání do mikroprocesoru přes připojení programovacího portu (USB typ A do USB Micro)
DUE	Arduino DUE – mikroprocesorová deska používaná pro tento projekt
CNC shield	nástavbový modul pro Arduino využíváný pro usazení více driverů pro řízení více motorů pro aplikace jako např. 3D tisk, či CNC stroje.
B2	kontakt driveru DRV8825 pro první vodič první skupiny cívek
B1	kontakt driveru DRV8825 pro druhý vodič první skupiny cívek
A1	kontakt driveru DRV8825 pro první vodič druhé skupiny cívek
A2	kontakt driveru DRV8825 pro druhý vodič druhé skupiny cívek
STL	exportovaný formát pro reprezentaci 3D plochy (<i>STereoLothography</i>)
G-code	programovací jazyk používaný pro 3D tiskárny a CNC stroje
SD	paměťové médium používané pro přenos elektronických dat (Secure Digital)
DWG	univerzální formát výkresů pro CAD programy (<i>drawing exchange format</i>)
URL	specifikace umístění webové stránky (<i>Uniform Resource Locator</i>)
m-file	<i>MATLAB</i> funkce s příponou .m
slx	<i>Simulink</i> model s příponou .slx
LTI	lineární, časově invariantní (linear, time-invariant)

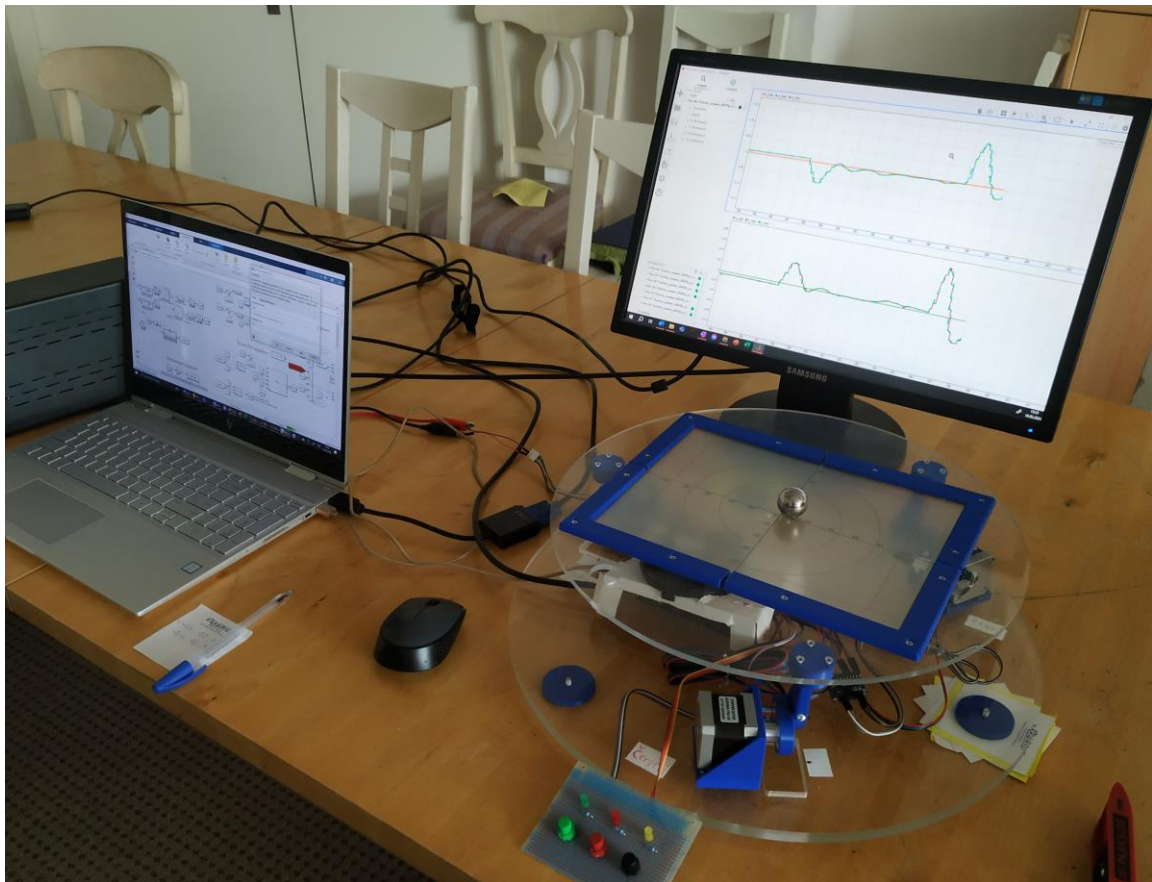
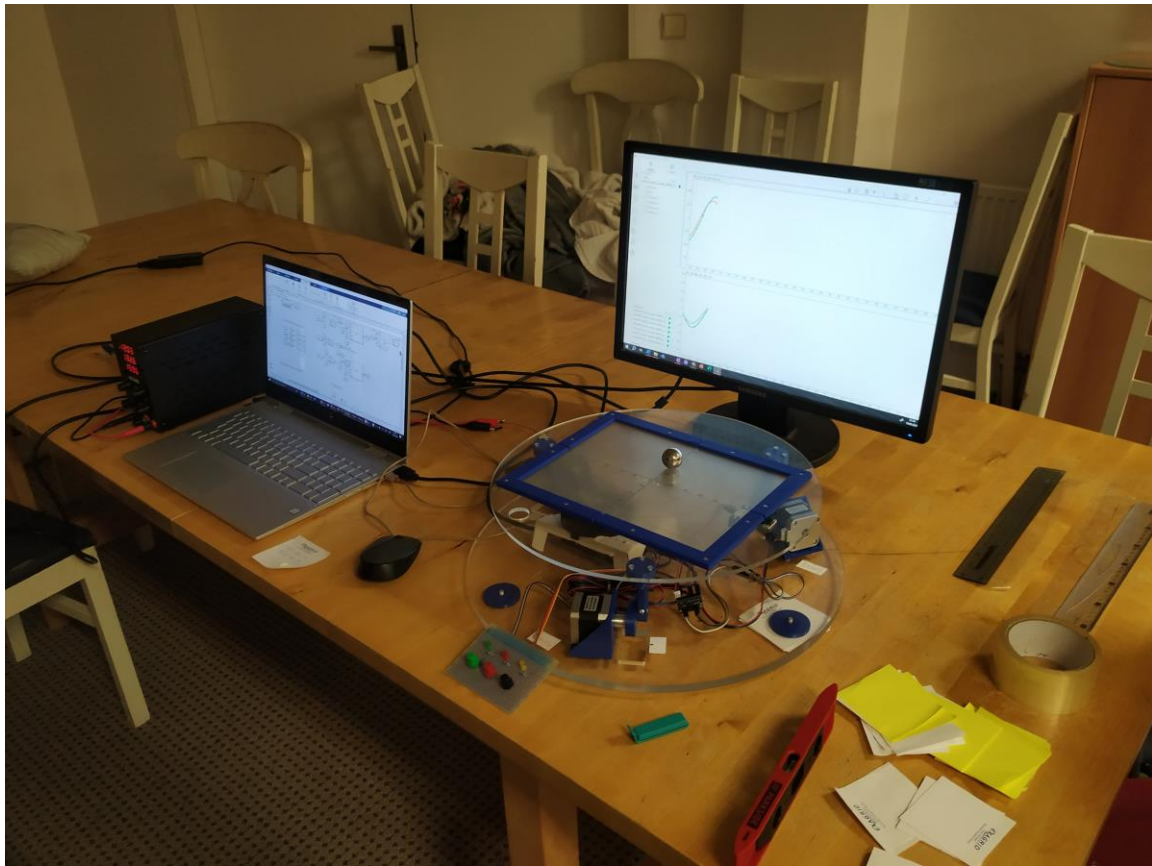
PŘÍLOHA B: FYZICKÁ SESTAVA A SCHÉMA ZAPOJENÍ

Fyzická sestava byla pospojována dle schématu v obr. B-3, kdy v obr. B-1 je představena výsledná forma vytvořeného přípravku.

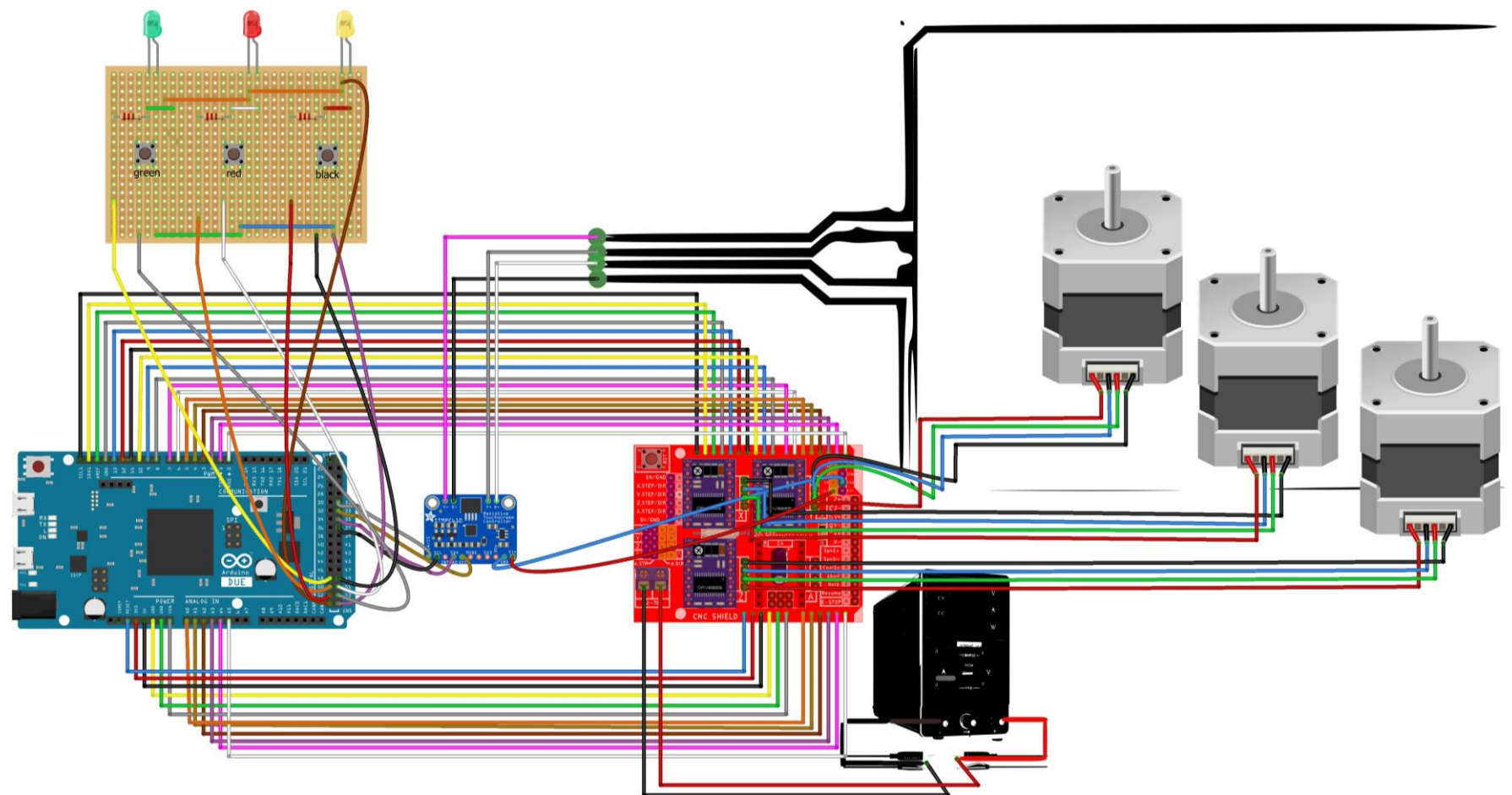


Obr. B-1 Vyrobený přípravek 3DOFp.

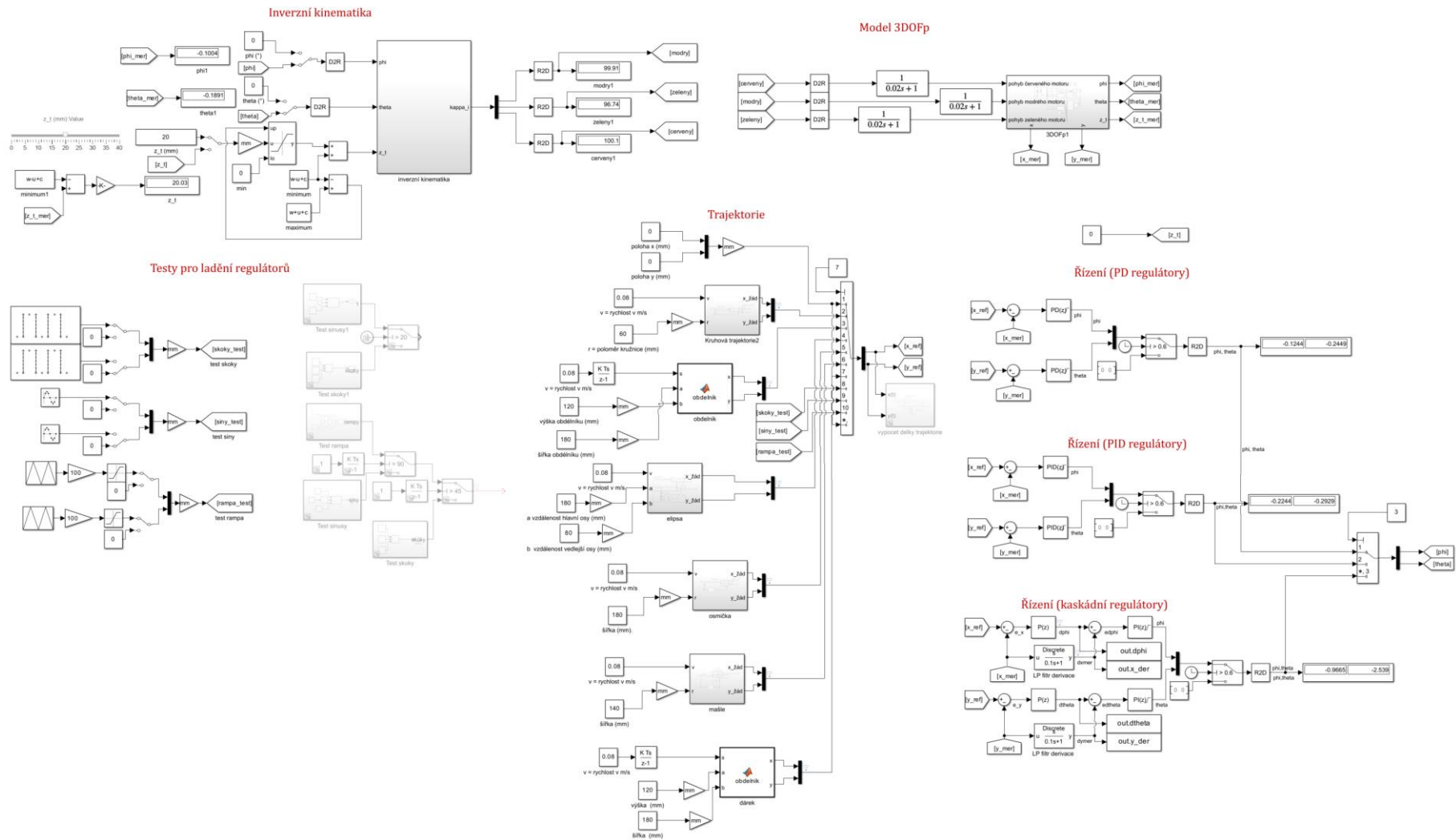
Pro ladění fyzického modelu byla vytvořena sestava dle obr. B-2, kdy pomocí *Data Inspector* byly na monitor zobrazeny signály x_{ref} a y_{ref} pro porovnání měřených signálů $x_{\text{měř}}$, $y_{\text{měř}}$. Simulační model z prostředí *Simulink* je na obr. B-4 a model pro ovládání fyzického přípravku z prostředí *Simulink* je na obr. B-5.



Obr. B-2 Ladící sestava pro polohování kuličky na plošině

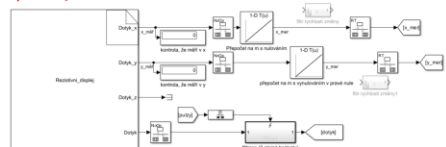


Obr. B-3 Model vytvořený pomocí software Fritzing 0 9. 3b [90].

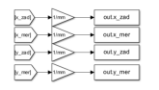


Obr. B-4 Simulační model 3DOFp v prostředí Simulink.

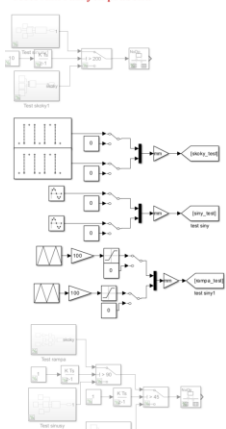
**Rezistivní displej
(nutné Build před spuštěním)**



Vykreslování trajektorií



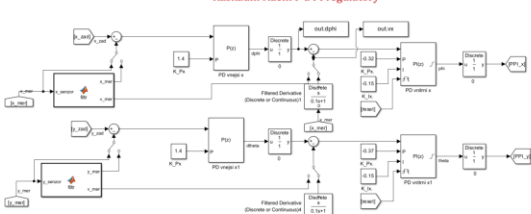
Testování různých průběhů



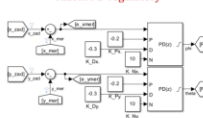
Testování konstant PID regulátoru



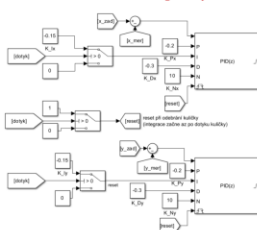
Kaskádní řízení P a PI regulátory



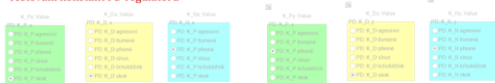
Řízení PD regulátoru



Řízení PID regulátoru



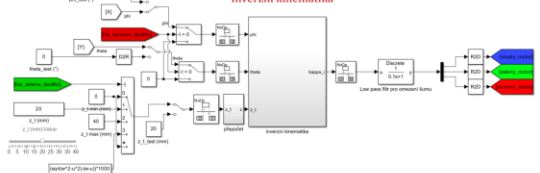
Testování konstant PD regulátoru



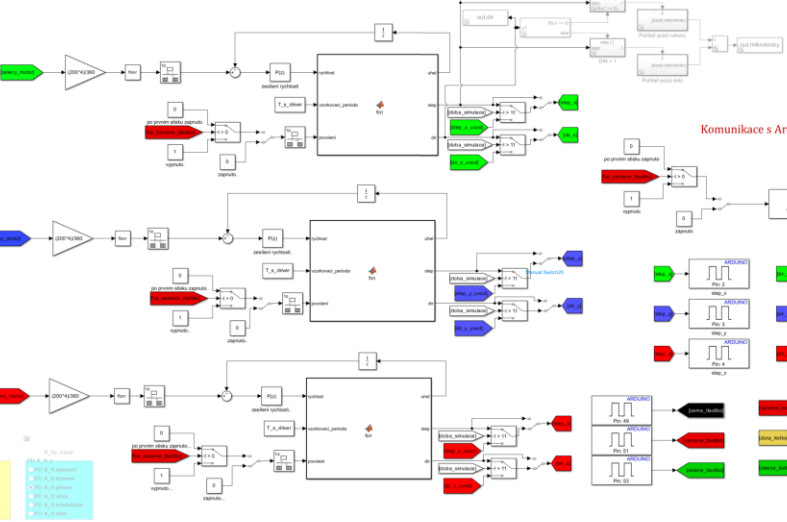
Počítání stisků tlačítek a reset



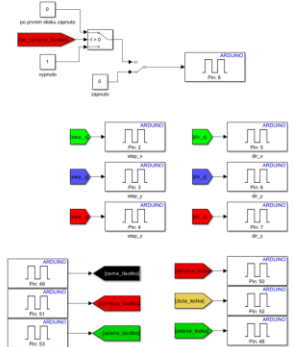
Inverzní kinematika



Generování počtu kroků pro motory



Komunikace s Arduino DUE



Úvodní melodie, lze zakomentovat



Obr. B-5 Ovládání fyzického modelu 3DOFp z prostředí Simulink.

PRÍLOHA C: PARAMETRY OCELOVÝCH KULIČEK

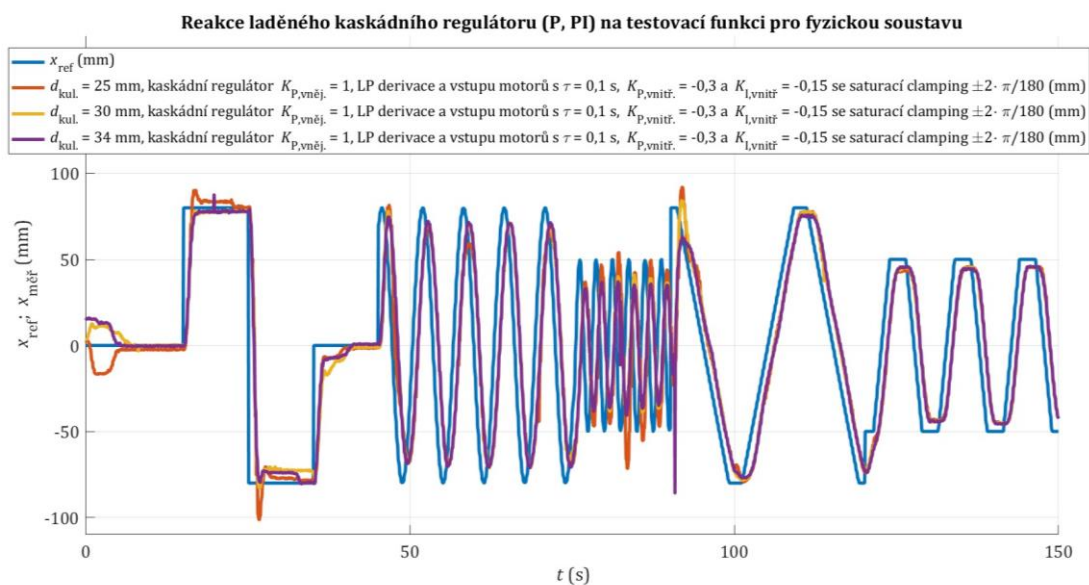
Pro projekt byly testovány tři kuličky, vliv na regulaci se téměř neprojevuje viz obr. C-3.



Obr. C-1 Naměřené průměry ocelových kuliček 25 mm, 30 mm a 34 mm.



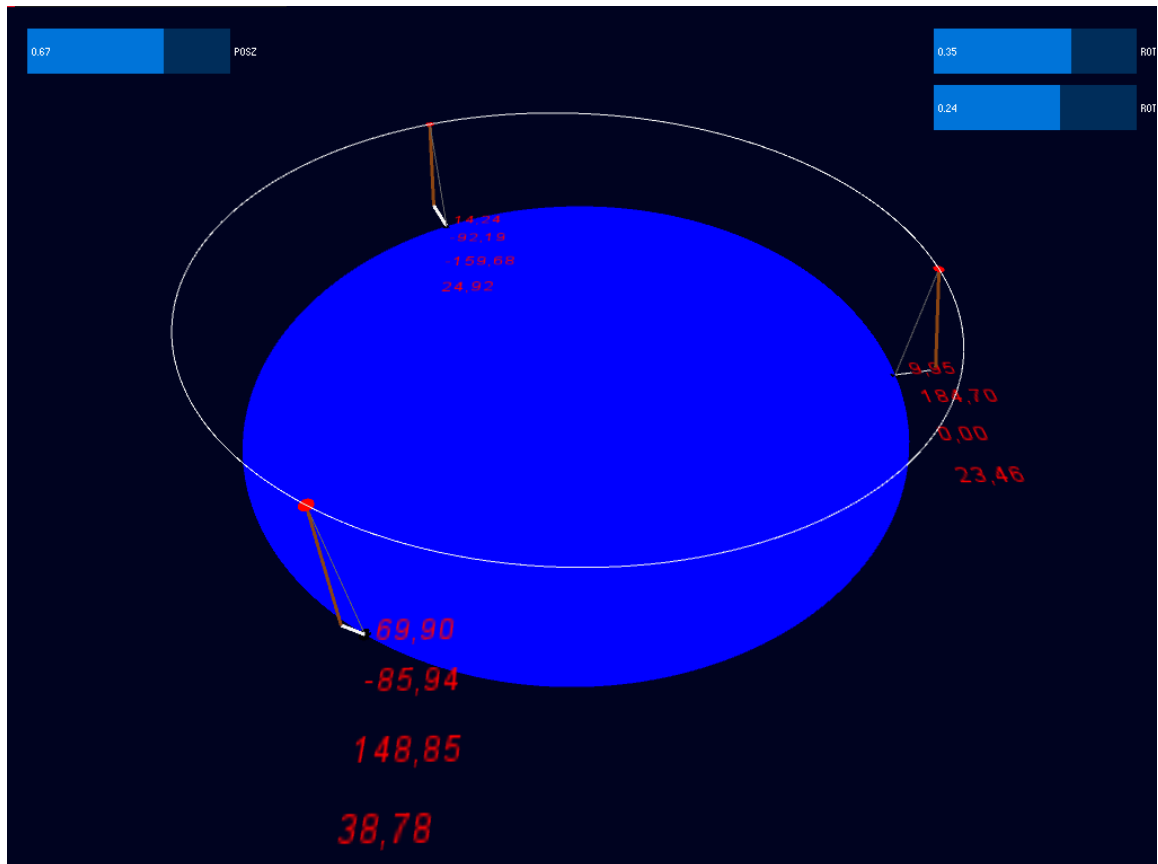
Obr. C-2 Naměřené váhy ocelových kuliček 64 g, 111 g a 161 g.



Obr. C-3 Vliv hmotnosti kuličky je zanedbatelný viz vykrácení hmotností dosažením (4-13) do (4-11)

PŘÍLOHA D: PYTHON MODEL

Za účelem testování pracovní oblasti 3DOFp byl upraven kód (viz [github](#)) v Processing Python Source Code (.pde) spustitelný po instalaci processing-4.3. Testovací model umožňuje zobrazit stavy pro zjednodušené ověření inverzní kinematiky.



Obr. D-1 Model se souřadnicemi pro testování inverzní kinematiky (inspirováno z [github](#)).