TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií

Autonivelační systém pracovního stroje

Bakalářská práce

Studijní program: B2612 Elektrotechnika a informatika
Studijní obor: Elektronické informační a řídicí systémy

Autor práce: **Radek Boháč**
Vedoucí práce: Ing. Jan Koprnický, Ph.D.
Ústav mechatroniky a technické informatiky

Liberec 2021

TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií

Zadání bakalářské práce

Autonivelační systém pracovního stroje

Jméno a příjmení: **Radek Boháč**
Osobní číslo: M18000019
Studijní program: B2612 Elektrotechnika a informatika
Studijní obor: Elektronické informační a řídicí systémy
Zadávající katedra: Ústav mechatroniky a technické informatiky
Akademický rok: **2020/2021**

Zásady pro vypracování:

1. Proveďte analýzu nivelačních systémů stavebních strojů, především strojů určených k 3D tisku budov.
2. V praktické části navažte na ročníkový projekt [1], navrhněte a vytvořte model autonivelačního řízení platformy v Matlab Simulinku.
3. Implementujte takto vytvořený model přímo do vývojové platformy Arduino/Rasbery.
4. Otestujte navržené řešení na této platformě s inklinoměry s rozhraním analogovým i CANopen.
5. Diskutujte možnosti připojení dalších snímačů, i snímačů s jinými komunikačními rozhraními.

Rozsah grafických prací:	dle potřeb dokumentace
Rozsah pracovní zprávy:	30–40
Forma zpracování práce:	tištěná/elektronická
Jazyk práce:	Čeština

Seznam odborné literatury:

[1] BOHÁČ, Radek, Roman DOLENSKÝ. *Autonivelační systém podvozku pracovního stroje.* Liberec, 2020. Ročníkový projekt. Fakulta mechatroniky, informatiky a mezioborových studií Technické univerzity v Liberci. Vedoucí ročníkového projektu Lukáš Hubka.

[2] GERANMEHR, Behdad, Kamran VAFAEE a Aliakbar SADEQI. High precision electro-hydraulic self leveling platform system. *2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*. IEEE, 2014, 2014, , 029-034. ISBN 978-1-4799-6743-8. Dostupné z: doi:10.1109/ICRoM.2014.6990772

[3] SIUHIN, Alexandr, Mikhail ZVEREV a Pavel BALASHOV. The structural model of automated leveling system for a construction machine. *IOP Conference Series: Materials Science and Engineering*. 2020, 709, 1-6. ISSN 1757-899X. Dostupné z: doi:10.1088/1757-899X/709/2/022089

[4] DEDOV, Denis, Alexandr SIUKHIN a Artem OBUKHOV. Development of a leveling system for construction equipment. *Transport Problems*. 2019, 14(3), 49-62. ISSN 1896-0596. Dostupné z: doi:10.20858/tp.2019.14.3.5

[5] PAN, Wen. *Methodological development for exploring the potential to implement on-site robotics and automation in the context of public housing construction in Hong Kong*. München, 2020. Disertační práce. Technische Universität München.

Vedoucí práce:	Ing. Jan Koprnický, Ph.D. Ústav mechatroniky a technické informatiky
Datum zadání práce:	9. října 2020
Předpokládaný termín odevzdání:	17. května 2021

L.S.

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 9. října 2020

Prohlášení

Prohlašuji, že svou bakalářskou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Jsem si vědom toho, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

10. května 2021 Radek Boháč

Autonivelační systém pracovního stroje

Abstrakt

Bakalářská práce se zaměřuje na problematiku nivelace a realizaci autonivelačního řízení. Cílem je navrhnout a realizovat autonivelační systém pracovního stroje. Práce obsahuje teoretickou část popisující využití autonivelačních systémů, především u konstrukčních 3D tiskáren. Dále je zde obsažena část s podrobným popisem návrhu takového systému v prostředí Matlab a jeho implementace do vývojové platformy. Poslední částí je ověření jeho vlastností a možnosti jeho rozšíření.

Klíčová slova: autonivelace, nivelační systém, konstrukční tiskárny, Matlab, Matlab Simulink, Arduino

Self-leveling System of the Working Machine

Abstract

The bachelor thesis is focused on leveling systems and self-leveling control realization. The aim is to propose and realize autoleveling system of the working machine. Thesis includes theoretical part about utilisation of these systems mainly for construction 3D printers. Another part is the propose of this system in Matlab environment and its implementation to the single-board microcontroller. The last part is the verification of its features and discussion with other improvement.

Keywords: autoleveling, leveling system, construction printers, Matlab, Matlab Simulink, Arduino

Poděkování

Rád bych poděkoval panu Ing. Janu Koprnickému, Ph.D. za cenné rady, odborné vedení a vstřícnost při všech konzultacích a vypracování této bakalářské práce.

Obsah

Úvod **13**

1 Nivelační systém **14**
1.1 Princip nivelace . . . 14
1.2 Senzorika . . . 14
1.2.1 Kapalinové sklonoměry . . . 14
1.2.2 Gyroskopické sklonoměry . . . 15
1.3 Kinematika . . . 15

2 Přehled využití **16**
2.1 Těžký průmysl . . . 16
2.2 Automobilový průmysl . . . 18
2.3 Obytné vozy . . . 18
2.4 Vojenská technika . . . 19
2.5 Vesmírná technika . . . 21
2.6 Konstrukční 3D tiskárny . . . 22
2.6.1 CyBe RC 3DP . . . 22
2.6.2 Apis Cor 3D . . . 23
2.6.3 HadrianX . . . 23
2.6.4 In Situ Fabricator . . . 24
2.6.5 Constructions-3D MAXI PRINTER . . . 24
2.6.6 Digital Construction Platform . . . 25
2.6.7 CONPrint3D . . . 25
2.6.8 Genesis Dimensions . . . 26

3 Rozdělení systémů **28**
3.1 Dle typu stabilizačních ramen . . . 28
3.1.1 Vertikálně výsuvná . . . 28
3.1.2 Výsuvná ramena tvaru H . . . 29
3.1.3 Výsuvná ramena tvaru V . . . 30
3.1.4 Sklopná ramena . . . 30
3.2 Dle mechanismu akčních členů . . . 31
3.2.1 Hydraulické . . . 31
3.2.2 Pneumatické . . . 32
3.2.3 Elektromechanické . . . 33

4 Návrh vlastního autonivelačního systému **35**
4.1 Nivelační platforma . . . 35
4.1.1 Koncepce platformy . . . 35
4.1.2 Stabilizační ramena . . . 37
4.2 Řízení autonivelace . . . 38
4.2.1 Metoda a proces nivelace . . . 38
4.2.2 Režimy řízení . . . 39
4.3 Hlavní program . . . 40

5 Vytvoření programu a implementace do vývojové platformy **42**
5.1 Vývojová platforma Arduino . . . 42
5.1.1 Arduino Uno . . . 43
5.1.2 CAN Modul MCP2515 . . . 44
5.2 Snímače . . . 44
5.2.1 Snímač náklonu TMM55E . . . 44
5.2.2 Snímač náklonu TMM88D . . . 45
5.2.3 CAN komunikace . . . 46
5.3 Prostředí Matlab . . . 47
5.3.1 Matlab Support Package for Arduino HW . . . 47
5.3.2 Simulink Support Package for Arduino HW . . . 48
5.4 Simulink model . . . 48
5.4.1 Zpracování výstupu senzoru náklonu . . . 48
5.4.2 Regulace . . . 49
5.4.3 Řízení servomotorů . . . 50
5.5 Matlab program . . . 51
5.5.1 Funkce pro zpracování výstupu senzoru náklonu . . . 51
5.5.2 Funkce pro řízení servomotorů . . . 52
5.5.3 Průběh programu . . . 52
5.5.4 Uživatelské rozhraní . . . 54

6 Ověření vlastností navrženého systému **55**
6.1 Přesnost nivelace . . . 55
6.2 Rychlost nivelace . . . 55
6.3 Porovnání metod řízení . . . 56
6.3.1 Řízení s uzavřenou smyčkou . . . 56
6.3.2 Řízení s otevřenou smyčkou . . . 57
6.3.3 Kombinované řízení . . . 58

7 Možnosti připojení přídavných snímačů **60**
7.1 Snímače . . . 60
7.1.1 Inerciální senzory . . . 60
7.1.2 Snímače zatížení . . . 60
7.2 Komunikační rozhraní . . . 61

Závěr **62**

Použitá literatura **63**

Přílohy **66**

A Simulink model **66**

B Matlab kód **67**

Seznam obrázků

2.1 Základna autojeřábu [3] . . . 16
2.2 Kompaktní jeřáb Reedyk [5] . . . 17
2.3 Systém E&P Caravan Level s výsuvnými podpěrami [9] . . . 19
2.4 Nivelační systém od společnosti GES Engineering [12] . . . 20
2.5 Zdvihací systém od společnosti Haacon [14] . . . 21
2.6 Stavební 3D tiskárna CyBe RC 3DP [15] . . . 22
2.7 Stavební 3D tiskárna Apis Cor 3D [16] . . . 23
2.8 Konstrukční zařízení HadrianX [18] . . . 23
2.9 Konstrukční zařízení In Situ Fabricator [20] . . . 24
2.10 Stavební 3D tiskárna Constructions-3D MAXI PRINTER [21] . . . 25
2.11 Konstrukční zařízení Digital Construction Platform [22] . . . 25
2.12 Stavební 3D tiskárna CONPrint3D [23] . . . 26
2.13 Model stavební 3D tiskárny společnosti Genesis Dimensions [25] . . . 26

3.1 Princip vertikálně výsuvného ramene . . . 29
3.2 Princip výsuvného ramene tvaru H . . . 29
3.3 Příklady rozložení polohovacích ramen . . . 29
3.4 Princip výsuvného ramene tvaru V . . . 30
3.5 Princip sklopného ramene . . . 30

4.1 Model nivelační platformy . . . 35
4.2 Koncepce nivelační platformy . . . 36
4.3 Rozměry nivelační platformy . . . 36
4.4 Model stabilizačního ramene . . . 37
4.5 Vývojový diagram hlavního programu . . . 40

5.1 Platforma Arduino Uno [31] . . . 43
5.2 CAN Modul MCP2515 [33] . . . 44
5.3 Použité inklinometry (vlevo TMM55E, vpravo TMM88A) [34] . . . 44
5.4 Fyzická vrstva CAN sběrnice . . . 46
5.5 Ukázka komunikace se snímačem TMM88D . . . 47
5.6 Simulink model autonivelačního systému . . . 49
5.7 Výstup z inklinometru . . . 49
5.8 Simulink model subsystému pro regulaci . . . 50
5.9 Simulink model subsystému pro řízení servomotorů . . . 51
5.10 Uživatelské rozhraní NivelationGUI . . . 52

6.1 Průběh nivelace (uzavřená smyčka) 56
6.2 Průběh nivelace v jednom směru (otevřená smyčka) 57
6.3 Průběh nivelace ve více směrech (otevřená smyčka) 57
6.4 Průběh nivelace v jednom směru (kombinované řízení) 58
6.5 Průběh nivelace ve více směrech (kombinované řízení) 58

Seznam tabulek

2.1 Porovnání nivelačních systémů jednotlivých 3D tiskáren 27

3.1 Porovnání vlastností akčních členů 34

4.1 Specifikace servomotoru MG996R [30] 37

5.1 Specifikace platformy Arduino Uno [32] 43
5.2 Specifikace inklinometru TMM55E-POH010 [35] 45
5.3 Specifikace inklinometru TMM88D-PCI090 [36] 45

6.1 Porovnání metod řízení 59

Seznam zkratek

3D	Three-dimensional, trojrozměrný
AC/DC	Alternating/Direct Current, střídavý/stejnosměrný proud
BP	bakalářská práce
CAD	Computer-Aided Design, počítačem podporované projektování
CAN	Controller Area Network, datová sběrnice
DA	Digital-to-Analog
EEPROM	Electrically Erasable Programmable Read-Only Memory, přepisovatelná elektronická paměť
GPIO	General-Purpose Input/Output, univerzální vstupní/výstupní pin
GUI	Graphical User Interface, grafické rozhraní
I2C	Inter-Integrated Circuit, multi-masterová počítačová sériová sběrnice
I/O	Input/Output, vstup/výstup
ICSP	In-Circuit Serial Programming, protokol pro sériové programování mikrokontrolérů
MCU	Microcontroller Unit, jednočipový počítač
MEMS	Micro Electro Mechanical Systems, Mikro Elektro Mechanické systémy, druh technologie
PC	Personal Computer, osobní počítač
PVC	Polyvinylchlorid, druh plastu
PWM	Pulse Width Modulation, pulzně šířková modulace
SPI	Serial Peripheral Interface, sériové periferní rozhraní
SRAM	Static Random Access Memory, statická elekronická paměť
SW, HW	Software, Hardware
USB	Universal Serial Bus, univerzální sériová sběrnice

Úvod

Problematika autonivelace je důležitým tématem v oblasti stavebnictví. Tyto systémy nalézáme u různých pracovních strojů, ale jimi výčet ani zdaleka nekončí, dalšími příklady, kde se využívá autonivelace, jsou: užitkové i osobní vozy, vojenská technika, atd. Bakalářská práce se soustředí především na pracovní stroje, u nichž nivelace zajišťuje ideální pracovní podmínky, maximální bezpečnost a zvyšuje přesnost i efektivnost práce na stavbě.

První část práce analyzuje již používané nivelační systémy, zde si popíšeme jejich principy a možné způsoby využití. Podrobněji se zaměříme na stroje určené k 3D tisku budov, několik konkrétních systémů si představíme a blíže rozebereme.

V praktické části je naším úkolem, na základě zjištěných informací o aktuálně používaných systémech, navrhnout vlastní autonivelační systém. Cílem je vytvořit model systému za pomoci programového prostředí Matlab. Navržený systém by měl být uzpůsoben pro řízení testovací platformy, která je součástí ročníkového projektu. Snahou této části bakalářské práce je na zmíněný ročníkový projekt navázat. Nivelační platforma bude řízena vývojovou platformou Arduino, případně Raspberry. Navržený model systému bude třeba do konkrétního typu platformy implementovat a následně jej patřičně otestovat. Mezi cíle práce patří využití a otestování dodaných inklinoměrů s analogovým i CANopen rozhraním.

Poslední část práce zhodnocuje a analyzuje dosažené výsledky spolu s předložením následné diskuze ohledně rozšíření navrženého systému. Diskuze se má mimo jiné zabývat možnostmi připojení přídavných snímačů, včetně snímačů využívajících jiná komunikační rozhraní.

1 Nivelační systém

Nivelační systém je v principu mechatronická soustava, určená k uvedení jisté roviny (např. terénu při stavbě budov), podvozku, a nebo platformy do vodorovného stavu. V případě autonivelačních systémů jde o automatizované systémy s řídicí logikou, nivelačním algoritmem a minimálním lidským zásahem.

Konkrétním případem takového systému je systém pro nivelaci strojů, který zajišťuje plynulý přechod podvozků různých typů pracovních strojů, nebo vozidel, do vodorovné, stabilní polohy a jejich ukotvení v této poloze. Právě na tento typ systémů se práce zaměřuje.

1.1 Princip nivelace

Nivelací rozumíme postup pro vytvoření vodorovné roviny. Tento termín se používá zejména v geodézii a ve stavebnictví, kde je jím konkrétně myšleno vytyčování takovéto roviny, nebo měření výškových rozdílů v terénu pomocí nivelačních přístrojů. Nicméně základní význam zůstává stejný.

Před samotnou nivelací je potřeba určit, v jaké poloze se námi sledovaná rovina nachází. K popisu aktuálního stavu roviny se v praxi používá úhel natočení v ose x a v ose y, úhel natočení v ose z a translační pohyb roviny xy je při nivelaci irelevantní. Rovina má v tomto případě pouze dva stupně volnosti, případně tři, pokud chceme sledovat také výšku roviny. Tyto dva úhly naklopení jsou vyhodnocovány různými způsoby měření, viz podkapitolu 1.2.

Pokud je námi sledovaná rovina vychýlena, je třeba uvést ji zpět do vodorovné polohy, a to konkrétním způsobem, jenž se nazývá nivelace. K nivelaci může dojít několika způsoby v závislosti na charakteru roviny. V případě podvozků dochází k nivelaci pomocí mechanických pohyblivých částí — například výsuvných ramen.

1.2 Senzorika

Pro určení stavu nivelované roviny je zapotřebí vhodná senzorika, která je schopna měřit její sklon oproti referenční horizontální rovině.

1.2.1 Kapalinové sklonoměry

Kapalinové sklonoměry, nebo také libely, fungují na principu působení zemské gravitace. Používají se na měření malých sklonů nebo ustavení do vodorovné polohy.

Základem jsou takzvané fluidní články. Snímač je z velké části naplněn elektrolytickou kapalinou a na jeho stěnách jsou elektrody. Při změně náklonu se mění výška hladiny elektrolytu nad elektrodami, a tím se zvyšuje, nebo snižuje, vodivost mezi elektrodami. Na základě hodnoty vodivosti je poté vyhodnocen náklon senzoru.

Měření těchto snímačů je velmi přesné, nevýhodou je ovšem maximální měřicí rozsah přibližně ± 30° [1].

1.2.2 Gyroskopické sklonoměry

Tyto sklonoměry také pracují na základě působení zemské gravitace a jejich principem je gyroskopické zařízení.

Kyvadlový gyroskop se třemi stupni volnosti vytváří rovinu od níž pak měříme sklon na základě sil, které na něho působí a snaží se jej vychýlit z jeho základní polohy. Gyroskop může být tvořen mikro hmotou zavěšenou na flexibilní konstrukci, při jejímž pohybu se mění kapacita, z níž je vyhodnocen náklon. Tato technologie se také označuje pojmem MEMS.

Snímače úhlu a náklonu s gyroskopem mohou měřit v rozsahu ± 90° ve dvou osách, případně v rozsahu 360° v jedné ose [2].

1.3 Kinematika

Kinematickou částí nivelačních systémů jsou různé mechanické pohyblivé části, které zapříčiní změnu stavu sledované roviny. Koncepce těchto částí se různí dle charakteru nivelované roviny. U podvozků vozidel se může jednat o druhy nastavitelných tlumičů, dále (například u pracovních platforem) jde o pohyblivá ramena. Právě pohyblivá ramena jsou nejčastějším způsobem nivelace pracovních strojů, kterými se zabývá bakalářská práce.

Podrobnější informace ke kinematickým částem nivelačních systému jsou uvedeny v následujích kapitolách, konkrétně v podkapitole 3.1.

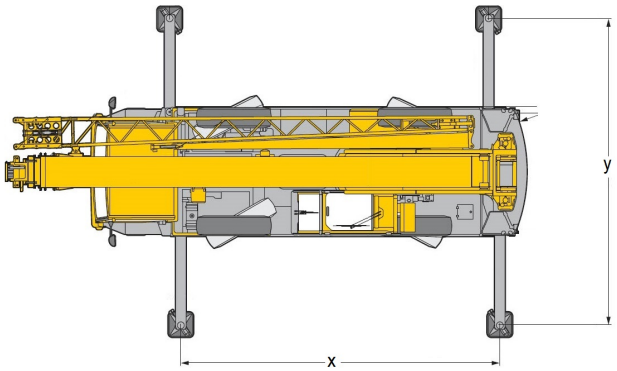
2 Přehled využití

Nivelační systémy se využívají pro zajištění ideálních pracovních podmínek. Při práci je třeba uvést základnu vozidla (podvozek, nebo jakoukoli pracovní platformu) do vodorovné polohy s konkrétní přesností. Autonivelační systémy jsou proto využívány v mnoha oblastech, například v těžkém průmyslu, ve vojenské technice, velké zastoupení těchto systémů nalezneme také u obytných vozů, ale do jisté míry i u osobních automobilů.

2.1 Těžký průmysl

Největší využití nalézáme v těžkém průmyslu, konkrétně u jeřábů, těžebních a demoličních bagrů, různých buldozerů a dalších pracovních strojů. Všechna tato pracovní zařízení potřebují ke své bezpečné a efektivní činnosti zajistit vodorovnou pracovní plochu. V jistých případech by bylo zdlouhavé upravovat okolní terén a namísto toho je efektivnější zajistit nivelaci samotného zařízení, nikoliv terénu.

Vysoká přesnost a stabilita nivelace je především vyžadována u jakékoliv jeřábní techniky, ať už jde o volně stojící, kompaktní, nebo jiné druhy jeřábů. V těchto případech může vychýlení z vodorovné polohy výrazně snížit provozní bezpečnost. Před samotným sestavením jeřábu je nutné zajistit jeho základnu a předejít tím jeho kývání či překlopení. To nedokáží obstarat samotné pneumatiky jeřábu, jednak z důvodu jejich pružnosti, a také malé vzdálenosti od sebe. Tento problém se řeší pomocí stabilizačních podpěr, které jsou umístěny ve vrcholech obdélníku (viz obrázek 2.1) tvořeného podvozkem a blízko středu je osa otáčení nástavby jeřábu. Těžiště jeřábu musí vždy ležet nad tímto obdélníkem.

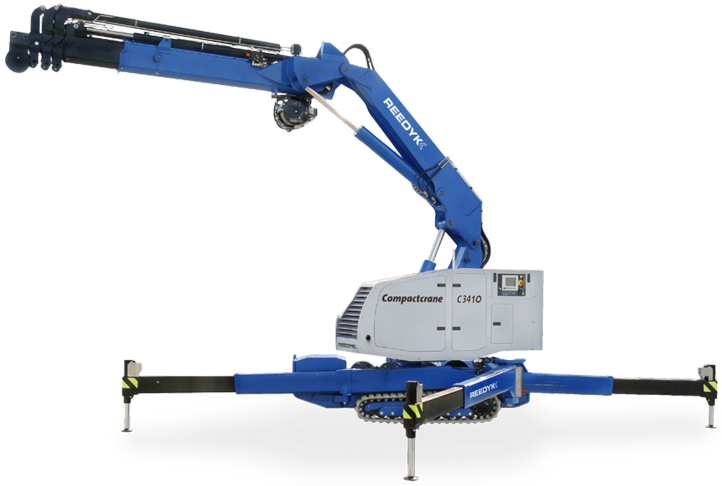


Obrázek 2.1: Základna autojeřábu [3]

S automatickými nivelačními systémy je zaručeno, že váha je optimálně distribuována přes všechny podpěry. V komplikovaných situacích mohou být podpěry případně nastaveny individuálně. U těžkých jeřábů je také velmi důležitým faktorem maximální zátěž kladená na výsuvné podpěry, které jsou v podobných případech nejčastěji řešeny hydraulickým systémem s bezpečnostním ukotvením v nastavené poloze [4].

Nivelační systém pro kompaktní jeřáby (minijeřáb Reedyk C3410)

Příkladem jeřábu s autonivelačním systémem je tento kompaktní minijeřáb od nizozemského výrobce strojů Reedyk Compact Cranes [5]. Jeřáb využívá k nivelaci čtyři hydraulická ramena typu H, která mohou být hydraulicky prodloužena, a dvě z nich jsou otočná. Postavení ramen není nijak omezeno, jsou tedy plně polohovatelná.



Obrázek 2.2: Kompaktní jeřáb Reedyk [5]

Všechny hydraulické válce na výsuvných ramenech mají individuální tlakové senzory. Kromě toho je průběžně sledována poloha každého ramene. Na základě údajů ze senzorů momentu zatížení, systém zajišťující bezpečnost nepřetržitě monitoruje, omezuje maximální nosnost ve vztahu ke stabilitě, v každé možné poloze jeřábu a konfiguraci podpěr. V úzkých prostorách to má za následek, že maximální nosnost je zachována nad plně vysunutou podpěru, a optimální, ale snížené, nosnosti nad každou zataženou podpěru. S tímto bezpečnostním prvkem je stabilita a nosnost optimální.

Využitím ovládacího displeje stroje může provozovatel stanovit maximální tlak na podpěry (založený na maximálním zatížení podvozku). Kontrola stability automaticky zajišťuje, že bezpečnostní limity stroje jsou nastaveny na tuto hodnotu [6].

2.2 Automobilový průmysl

V automobilovém průmyslu, konkrétně u osobních aut, se nevyskytují klasické nivelační systémy se stabilizačními rameny, ale systémy výškově nastavitelného odpružení, což je také určitá forma nivelace. Tyto systémy umožňují především měnit světlou výšku vozidla. Úprava výšky vozidla je jedním z možných způsobů, jak zajistit snížení aerodynamického odporu, což má za následek zlepšení výkonu a úsporu paliva při vysoké rychlosti. Co se týče nivelace, tedy přivedení vozidla do vodorovné polohy vůči vozovce, tak té se využívá ke zlepšení jízdních vlastností v obtížném terénu nebo v prudkých zatáčkách. Dále jsou tyto systémy využívány pro udržení konstantní výšky vozidla bez ohledu na jeho zatížení.

Historicky prvním sériově prodávaným vozidlem s nastavitelným odpružením, byl model Citroen 15CVH z roku 1954. Toto vozidlo mělo autonivelační, výškově nastavitelné hydropneumatické odpružení, které se později stalo nejrozšířenějším typem nastavitelného odpružení. Účelem tohoto systému bylo poskytnout citlivé, dynamické a velkokapacitní odpružení, které zvyšuje kvalitu jízdy na různých površích. Později začaly stejný typ systému používat například značky Rolls-Royce, Maserati a Peugeot [7].

Některé automobily tyto systémy používají ke zlepšení ovladatelnosti, například vozy Mercedes-Benz se systémem Active Body Control, které při jízdě rychlostí vyšší než 120 km/h sníží výšku vozidla. Podobné systémy nabízejí také novější typy automobilů značky Jeep, které zahrnují nastavitelné odpružení u modelů 4×4. Automaticky snižují podvozek, když je aktivován režim parkování a rovněž při vysokých rychlostech. Řidičům umožňují ručně zvednout odpružení s ohledem na terén [8].

Mnoho vozidel je negativně ovlivněno změnou náklonu zapříčiněnou zatížením – konkrétně velkým zatížením zadních sedadel, nebo zavazadlového prostoru. Změna zatížení, a tím i změna náklonu podvozku ovlivňuje: aerodynamické vlastnosti, zaměřování světlometů, účinnost brzdění, tlumení nárazů odpružením a výkon. Většina brzdné síly je na předních kolech vozidla. Pokud je zadní část silně zatížena, není brzdění tak účinné. Při rychlém brzdění se v této situaci přední brzdy snadněji zablokují z důvodu nedostatečného přenosu hmotnosti na přední část vozidla. Autonivelační odpružení zvedá zadní část vozidla, aby rovnoměrněji rozložila hmotnost. V konstrukci odpružení je inherentní konflikt – pokud jsou pružiny měkké, vůz bude pohodlný, ale dramaticky ovlivnitelný zatížením. Řada výrobců si uvědomuje tento konflikt a usiluje o různé cesty, jak dosáhnout pohodlí i nosnosti současně [7].

2.3 Obytné vozy

Autonivelační systémy jsou dále využívány u obytných a rekreačních vozů. Jejich účelem je zvýšení stability na nerovném povrchu. Stabilizace vozu je dosaženo pomocí několika stabilizačních podpěr, které jsou tvořeny hydraulickými válci. Podpěry vozu slouží pouze ke stabilizaci a podepření zadní části přívěsu, nikoli ke zvedání těla obytného vozu. Stabilizační podpěry mohou být přídavné, nebo integrované v samotném podvozku. Pomocí těchto podpěr lze dosáhnout v interiéru vozu naprosté

roviny. U obytných vozů je kladen důraz především na jednoduché ovládání z důvodu civilního použití. Obsluha systémů je plně, nebo alespoň částečně, automatická. Důraz je kladen i na velkou rychlost vyrovnávání.

Nivelační systém pro obytné vozy (E&P Caravan Level)

Příkladem nivelačního systému pro obytné vozy je systém od společnosti E&P Hydraulics [9]. Společnost dodává konstrukce hydraulických výsuvných nebo sklopných podpěr pro obytné vozy. Jedná se převážně o systémy se šesti opěrnými body. Hydraulické podpěry jsou připojeny k rámu vozidla pomocí montážních desek. Tím je dosaženo potřebné stability podpěr. U tohoto typu systému jsou na nápravu karavanu namontovány dvě hlavní podpěry, které slouží k nivelaci a dále jsou zde čtyři rohové klikové podpěry, které zabraňují přenosu síly přes podlahu vozidla během instalace systému. Celková nosnost podpěr je 3500 kg.

Obrázek 2.3: Systém E&P Caravan Level s výsuvnými podpěrami [9]

Obsluha systému probíhá pomocí ovládací jednotky uvnitř karavanu nebo pomocí volitelného dálkového ovládání. Hydraulické podpěry se buď automaticky vysunou (poté se upraví podle potřeby), nebo se podpěry ovládají ve dvojicích na jedné straně ručním ovládáním. To zabrání deformaci podvozku. Stabilizace obytného vozu trvá přibližně 120 sekund a je plně automatická [9].

Další nivelační systémy pro obytné vozy nabízejí například společnosti Lippert a Glide Rite [11, 10]. Principy nivelace se od výše zmíněného systému příliš neliší, jsou tedy založeny na přídavných, nebo integrovaných výsuvných podpěrách s dobou nivelace v řádech desítek sekund.

2.4 Vojenská technika

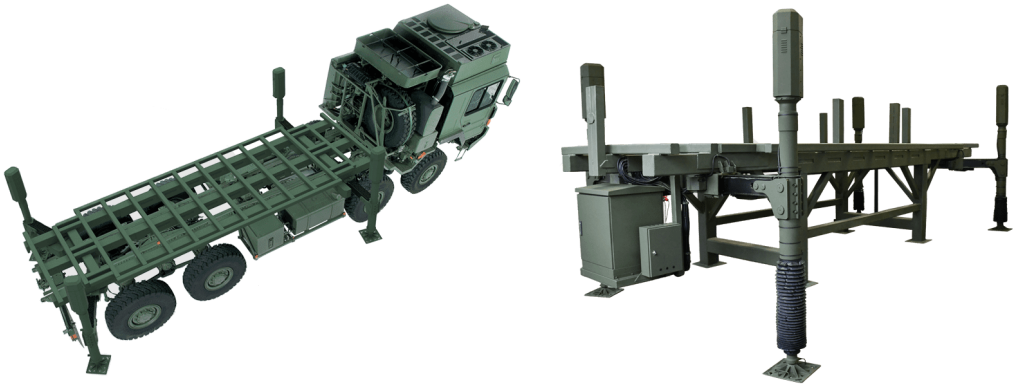
Autonivelační systémy nalézají ve vojenské technice velmi široké uplatnění a je zde stále kladeno velké úsilí na jejich vývoj. Parametry, na které je v tomto ohledu kladen velký důraz, jsou přesnost nivelace a její rychlost. Mezi konkrétní vojenská zařízení, která využívají určitou formu nivelace, patří vojenská přepravní, nákladní, nebo bojová vozidla, případně různé mobilní platformy, nebo kontejnery. Kromě toho jsou

formy nivelace nasazeny v oblasti telekomunikací a radarů, výcvikových simulačních systémů, mobilních satelitních komunikačních jednotek, startovacích jednotek bezpilotních průzkumných dronů a jiných vojenských zařízeních.

Co se týče vojenských přepravních vozidel, tak nivelační systémy jsou hojně využívány u různých zařízení přepravujících například radarové, nebo telekomunikační systémy, kde je třeba velmi přesné nivelace. Nivelace je dosaženo výsuvnými rameny, podobně jako v těžkém průmyslu, nebo u obytných vozů.

Nivelační systém pro vojenská vozidla (GES Engineering)

GES Engineering je příkladem společnosti, která vyvíjí nivelační zařízení pro obranný, letecký a automobilový průmysl. Společnost dodává systémy pro přepravní vozidla na radary, odpalovací, a nakládací zařízení. Jako jedna z mála společností se zaměřuje na elektromechanické systémy. Systém lze přizpůsobit pro vozidla vážící 40 a více tun, a lze je tedy použít v širokém spektru vojenských vozidel, od vozidel s pohonem 4×4 až po raketomety. Vyrovnání vozidla se zátěží 40 tun zvládne s přesností 0,2° za méně než 4 minuty. Jedná se především o systémy se čtyřmi výsuvnými rameny, jež jsou řízeny pomocí PLC [12].



Obrázek 2.4: Nivelační systém od společnosti GES Engineering [12]

Vozidla, u kterých je třeba nivelovat podvozek při jízdě, využívají stejných principů jako vozidla osobní. Například již zmíněný systém hydropenumatického odpružení s nastavitelnou výškou lze nalézt u moderních tanků a jiných velkých vojenských vozidlech.

Mezi další vojenská zařízení s potřebou nivelace patří kontejnerové a přístřeškové přepravní systémy, které jsou vhodné pro civilní obranu, pomoc při katastrofách, přepravu vybavení pro armády, pro služby technické pomoci, mezinárodní krizovou pomoc a všude tam, kde musí být těžké náklady přepravovány samotnou pracovní silou. Zařízení kladou největší důraz především na velký zvedací výkon a na rychlost jejich instalace, ale je zapotřebí zajistit i dostatečnou stabilitu, a proto se i zde využívá jisté formy nivelace.

Nivelační zdvihací a přepravní systém (Haacon)

Příkladem těchto zařízení jsou zdvihací a přepravní systémy od německé společnosti Haacon [13], která se specializuje právě na vojenskou techniku, ale mimo jiné také na nakládací systémy letadel, nákladních vozů a přepravních kontejnerů. Tato mobilní zdvihací zařízení jsou nejčastěji nasazena ve vojenských operacích, které zahrnují: polní nemocnice, dekontaminační zařízení, vojenské kuchyně, nebo zásobovací kontejnery pro elektřinu, vodu a plyn. Systémy umožňují zvedání a spouštění nákladu na nerovném povrchu, automatickou detekci polohy, autonivelaci s přesností 0,1° a bezpečnostní vypnutí při překročení maximálního náklonu 3,5° [14].

Obrázek 2.5: Zdvihací systém od společnosti Haacon [14]

2.5 Vesmírná technika

Nivelační systémy mají bezpochyby velké zastoupení i u vesmírné techniky. Bohužel většina společností z oblasti kosmonautiky se snaží zveřejnit co nejmenší množství technických informací. Z tohoto důvodu zde není žádné konkrétní nivelační zařízení uvedeno a popsáno.

2.6 Konstrukční 3D tiskárny

Stavební 3D tisk se týká různých technologií, které používají 3D tisk jako základní metodu pro stavbu budov nebo konstrukčních komponent. Jsou dva základní typy těchto zařízení, a to portálové tiskárny a tiskárny využívající různé typy robotických ramen. Ve druhém případě se jedná o zařízení podobná jeřábům, která mohou mít základnu pevně ukotvenou k zemi. To obnáší přípravu staveniště (zemní práce, nivelace) a zemní kotvení. Robotické rameno sedí na kovovém rámu, který je připevněn k dostatečně vodorovné podlaze, není tedy potřeba žádného autonivelačního systému s podpůrnými rameny. Na druhé straně u mobilních tiskáren s podvozkem je tento systém žádoucí. Nároky na nivelaci jsou prakticky stejné jako u jeřábní techniky, tedy vysoká přesnost a udržitelnost nivelace.

Níže v této kapitole jsou popsána vybraná konstrukční zařízení, využívající určitý typ nivelace základny. Ve většině případů se jedná o mobilní zařízení, určená k rychlému přesunu po pracovišti a k rozšíření stavebního prostoru.

2.6.1 CyBe RC 3DP

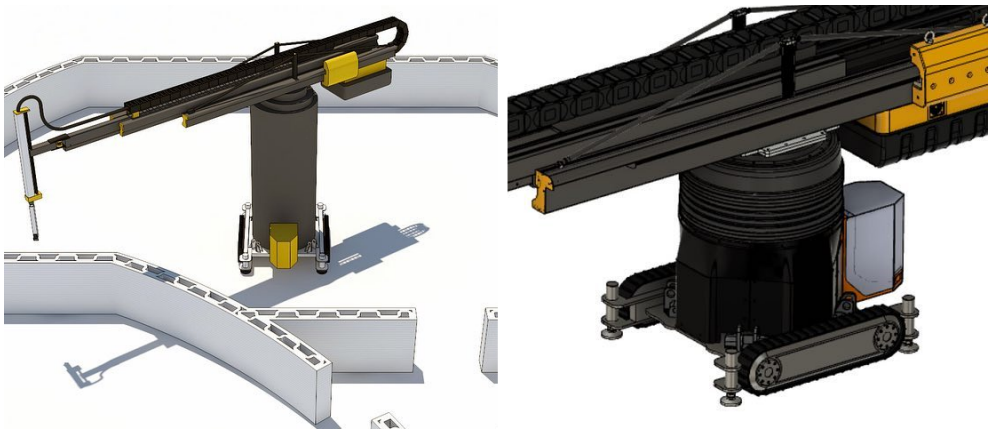
Tato stavební 3D tiskárna od nizozemské společnosti CyBe Construction [15] je mobilní variantou fixní tiskárny CyBe R 3DP, oproti které využívá nivelačních ramen pro uvedení základny do vodorovné polohy. Jedná se o stranovou 3D tiskárnu (tisk je možný pouze v jednom směru), která je uzpůsobena pro konstrukční tisk pouze na jedné straně základny, na rozdíl od běžných tiskáren, které mohou s ramenem otáčet a tisknout po celém obvodu.

Obrázek 2.6: Stavební 3D tiskárna CyBe RC 3DP [15]

Robotické rameno je usazeno na pohyblivém podvozku s gumovými pásy, které zprostředkovávají mobilitu. Lze tedy s tiskárnou manévrovat i v obtížném terénu. Pro samotnou nivelaci a stabilizaci při tisku jsou zde tři hydraulická ramena, která jsou délkově rozšiřitelná, čímž se zvyšuje celková tisknutelná výška projektů. Nivelační ramena jsou pouze vertikálně výsuvná (podrobnosti v části 3.1.1) s provozní nosností 4500 kg. Celé zařízení, včetně nivelačního systému je řízeno pomocí řídicí jednotky, na které běží software CyBe [15].

2.6.2 Apis Cor 3D

Jedná se o stavební 3D tiskárnu od Rusko-americké společnosti Apis Cor [16], která umožňuje tisk po celém svém obvodu. Lze ji pevně přikotvit k rovnému betonovému základu, nebo využít mobilního podvozku s pásy a autonivelačním systémem.



Obrázek 2.7: Stavební 3D tiskárna Apis Cor 3D [16]

Součástí podvozku jsou čtyři vertikálně výsuvná ramena. Použitá nivelační ramena jsou tvořena hydraulickými písty se servořízením a provozní nosností 2000 kg. Zařízení je možné instalovat na téměř jakýkoli povrch s výškovým rozdílem menším než 10 cm a polohu ramen lze nastavit s přesností 1 mm. Samotná nivelace je řízena dvouosým inklinometrem a PID regulátorem [16, 17].

2.6.3 HadrianX

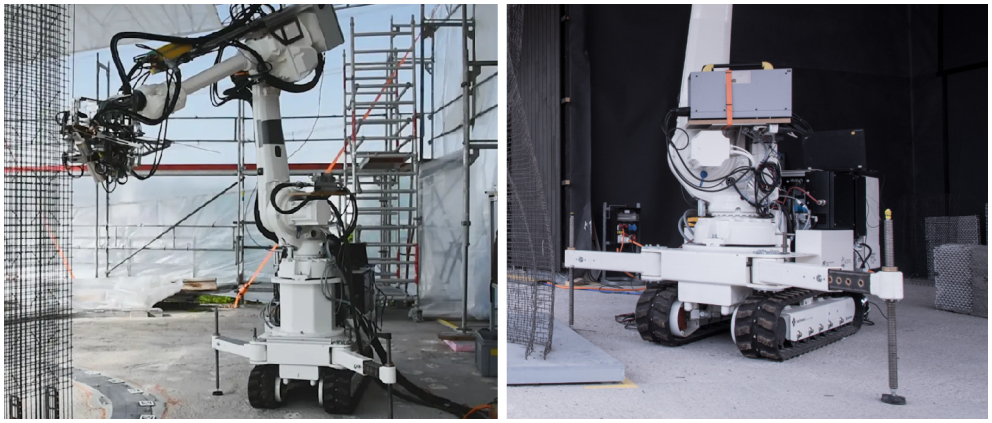
HadrianX od společnosti Fastbrick Robotics [18] je celosvětově první mobilní robotické zařízení určené k rychlému pokládání cihel a stavbě budov dle připraveného CAD modelu. Jedná se o zařízení na způsob šestikolového nákladního vozu, nebo kamionu s robotickým ramenem instalovaným na podvozku.

Obrázek 2.8: Konstrukční zařízení HadrianX [18]

Součástí zařízení jsou čtyři nivelační ramena. Jedná se o elektromechanická horizontálně a vertikálně výsuvná ramena tvaru H (podrobnosti v části 3.1.2). Tento nivelační systém je plně interní, to znamená, že stabilizační ramena jsou při pohybu kamionu skryta v podvozku. Po zastavení kamionu a zahájení samotné nivelace jsou ramena plně horizontálně vysunuta, přibližně do vzdálenosti 1 m od boční strany kamionu, čímž je zvýšena celková stabilita zařízení. Ramena není možné nijak dále polohovat [18].

2.6.4 In Situ Fabricator

Tento poloautonomní konstrukční robot je společným projektem Federálního institutu pro technologii v Curychu [19] a společnosti DFAB HOUSE [20]. Robot je určený pro automatické pokládání cihel a přípravu ocelových výztuh pro zděné konstrukce. Je navržen především pro náročnější a komplexnější konstrukce.



Obrázek 2.9: Konstrukční zařízení In Situ Fabricator [20]

Robot se pohybuje na pásovém podvozku opatřeným dvěma stabilizačními rameny, na přední straně podvozku. Ramena jsou bez automaticky řízené nivelace. Při přesunu zařízení jsou ramena plně složena a poté je lze manuálně vysunout a polohovat. I samotná nivelace je pouze manuální a založená na klasických závitových šroubech, které s výhodou využívají samosvornosti. Je tedy třeba, aby alespoň jeden pracovník, který obsluhuje zařízení ručně nastavil vysunutí jednotlivých ramen. V tomto případě se spíše jedná o prvek zvyšující stabilitu zařízení, než o nivelační systém. Použitá stabilizační ramena mají provozní nosnost 1400 kg [20].

2.6.5 Constructions-3D MAXI PRINTER

Jde o mobilní 3D tiskárnu budov od francouzské společnosti Constructions-3D [21], což je dceřinná společnost Machines-3D, nabízejícího řadu 3D tiskáren, materiálů a příslušenství pro stavební 3D tisk a sken.

Podvozek je vybaven gumovými pásy a čtyřmi sklopnými stabilizačními rameny (podrobnosti v části 3.1.4). Délka ramen je manuálně nastavitelná, jedná se o teleskopická ramena. K nivelaci dochází automatickým hydraulickým sklopením takto

Obrázek 2.10: Stavební 3D tiskárna Constructions-3D MAXI PRINTER [21]

nastavených ramen. Ramena mají provozní nosnost 2500 kg. Zařízení lze stabilizovat na povrchu s maximálním náklonem 15° [21].

2.6.6 Digital Construction Platform

Jde o experimentální zařízení umožňující rozsáhlou 3D výrobu. K aplikaci tohoto systému patří výroba nestandardních architektonických forem. Jedná se o projekt výzkumné laboratoře MIT Media Lab [22].

Obrázek 2.11: Konstrukční zařízení Digital Construction Platform [22]

Zařízení je, jako ve většině předchozích případech, umístěno na pásovém podvozku se čtyřmi nivelačními rameny. Ramena jsou hydraulicky sklopná, přičemž přední pár ramen je částečně manuálně polohovatelný [22].

2.6.7 CONPrint3D

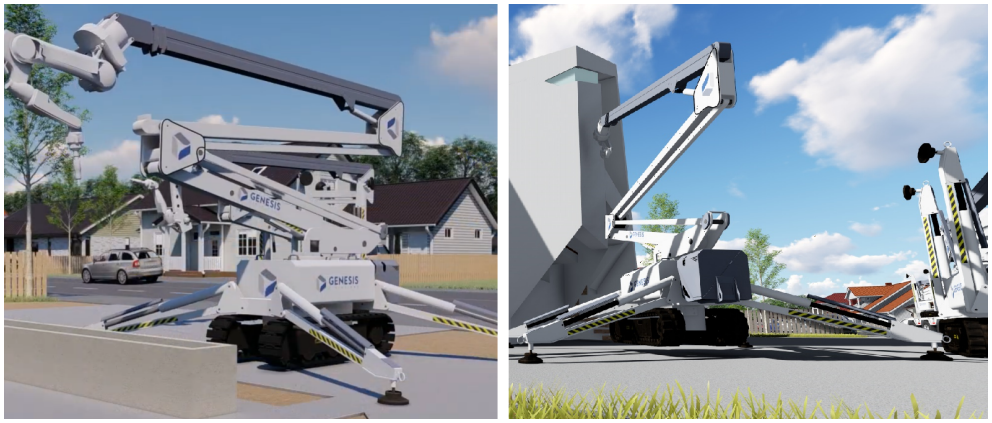
CONPrint3D je nový koncept stavební 3D tiskárny navržený výzkumníky z Technické univerzity v Drážďanech [23]. Jedná se o autojeřáb s tiskovou hlavou na místo teleskopického výložníku.

Obrázek 2.12: Stavební 3D tiskárna CONPrint3D [23]

Stabilizace tohoto autojeřábu je zajišťována čtyřmi vertikálně výsuvnými nivelačními rameny tvaru H. Tato ramena pracují na hydraulickém principu a přední pár ramen je možné automaticky horizontálně vysunout a manuálně polohovat pro zvýšení celkové stability. Na rozdíl od většiny zařízení, v tomto případě nejsou všechna ramena symetrická a umístěna v rozích podvozku. Zadní pár ramen je mohutnější a je ukotven v polovině boční strany autojeřábu. Zadní ramena lze polohovat pouze částečně [24].

2.6.8 Genesis Dimensions

Genesis Dimensions je americká společnost zabývající se výzkumem a vývojem stavebního řešení využívající technologie 3D tisku [25]. Kompletní stavební zařízení je prozatím ve vývoji, ale u dostupných ukázkových modelů lze pozorovat jeho budoucí řešení nivelačního systému.



Obrázek 2.13: Model stavební 3D tiskárny společnosti Genesis Dimensions [25]

Zařízení by mělo být klasicky vybaveno pásovým podvozkem a k nivelaci by sloužila čtyři hydraulická ramena. Ramena by byla nejspíše plně automaticky sklopná a k nivelaci zařízení by mělo docházet naklápěním ramen [25].

Tabulka 2.1: Porovnání nivelačních systémů jednotlivých 3D tiskáren

Zařízení	**Typ stab. ramen**	**Počet ramen**	**Mechanismus / Režim nivelace**	**Pracovní nosnost [kg]**
CyBe RC 3DP	vertikálně výsuvná	3	hydraulický /auto	4500
Apis Cor	vertikálně výsuvná	4	hydraulický /auto	2000
HadrianX	výsuvná tvaru H	4	elektromech. /auto	-
In situ Fabricator	výsuvná tvaru H (poloh.)	2	mechanický /manuál	1400
Construction-3D	sklopná (teleskopická)	4	hydraulický /auto	2500
DCP	sklopná (přední poloh.)	4	hydraulický /auto	-
CONPrint3D	výsuvná tvaru H (poloh.)	4	hydraulický /auto	-
Genesis Dimensions	sklopná	4	hydraulický /auto	-

3 Rozdělení systémů

Práce se zabývá především autonivelačními systémy pracovních strojů, což jsou systémy, které využívají určité druhy stabilizačních ramen. Jedná se o výše popsané systémy, především konstrukčních 3D tiskáren a jeřábní techniky.

Druhy systémů lze dělit dle konkrétních specifikací, například: dle použitého typu stabilizačních ramen, jejich počtu nebo použitého mechanismu k ovládání akčních členů ramen. Níže jsou popsány jednotlivé specifikace a podrobnosti o jejich funkčnosti.

3.1 Dle typu stabilizačních ramen

Pracovní stroje využívají různé mechanické koncepty stabilizačních ramen. Použití konkrétního typu ramen se různí dle konkrétních nároků. Hlavní rozdíly jsou v maximální dosažitelné stabilitě podvozku a v nárocích na prostor.

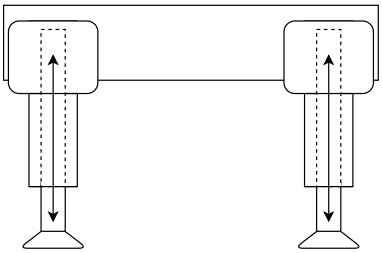
Jednotlivé typy ramen mají rozdílnou vyrovnávací schopnost, a proto lze mnohdy nalézt koncepty, které jsou tvořeny kombinací jednotlivých typů ramen. Tyto koncepty jsou konstrukčně i algoritmicky velmi složité, proto jsou níže vypsány základní typy stabilizačních ramen, které v dnešní době využívá pracovní technika:

- vertikálně výsuvná,
- výsuvná ramena tvaru H,
- výsuvná ramena tvaru V,
- sklopná ramena.

3.1.1 Vertikálně výsuvná

Nejjednodušší typ ramen, která jsou tvořena pouze lineárními členy, které jsou pevně ukotveny v rozích podvozku zařízení. Tento typ stabilizačních ramen neklade prakticky žádné nároky na volný prostor v okolí podvozku konkrétního stroje.

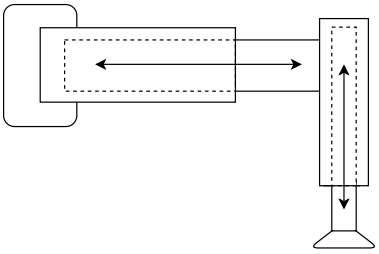
Má omezenou stabilitu a vyrovnávací schopnost při ustavování stroje v porovnání se současnou moderní technikou. Jejich největší výhodou je nízká pořizovací cena a jednoduché řízení autonivelace. Tento typ ramen využívají především nízké stroje, nebo například karavany a obytné vozy, viz podkapitolu 2.3.



Obrázek 3.1: Princip vertikálně výsuvného ramene

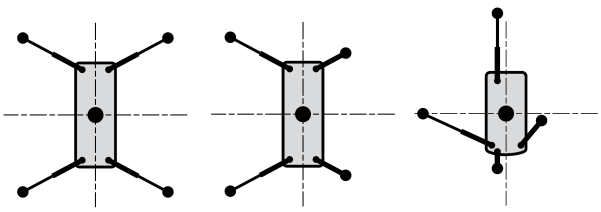
3.1.2 Výsuvná ramena tvaru H

Jde o stejný koncept jako v předchozím případě, ale bez pevného ukotvení v rozích podvozku. Tato ramena lze manuálně, nebo automaticky horizontálně vysunout do dané vzdálenosti od podvozku a zvýšit tím stabilitu celého zařízení. Tento systém stabilizačních ramen má obrovskou stabilitu, používají ho především jeřáby a těžká zvedací technika.



Obrázek 3.2: Princip výsuvného ramene tvaru H

Některá stabilizační ramena (polohovací) jsou řešena tak, že s nimi lze manipulovat ve více směrech a k horizontálnímu vysunutí může dojít pouze částečně (lze docílit nesouměrného vysunutí). Systému se využívá např. v malých a problematických prostorech. Nevýhodou je, že při změně rozložení jednotlivých ramen se změní plocha, nad kterou se musí nacházet těžiště zařízení. V obdobných případech jsou ideální ramena s individuálními senzory momentu zatížení.

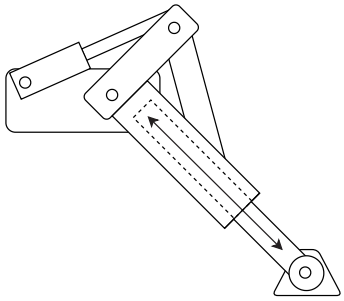


Obrázek 3.3: Příklady rozložení polohovacích ramen

3.1.3 Výsuvná ramena tvaru V

Tento koncept je podobný prvnímu typu ramen, s tím rozdílem, že lineární pohyb je v diagonálním směru k zemi. Ve většině případů jsou tato ramena navržena tak, že v počáteční poloze jsou sklopena směrem nahoru, nebo dle konstrukce jinak složena (z důvodu šetření prostoru), poté manuálně, nebo automaticky sklopena, ukotvena v této poloze a teprve poté dochází k lineárnímu posunu hydraulických válců.

Je to jeden z nejvhodnějších typů stabilizačních ramen, i přesto že má horší vyrovnávací schopnost a větší nároky na rozteč při ustavování zařízení než předchozí typy ramen.

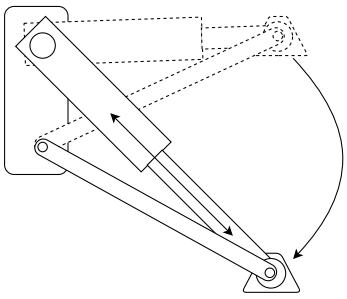


Obrázek 3.4: Princip výsuvného ramene tvaru V

3.1.4 Sklopná ramena

Jedná se o typ ramen, která ovlivňují nivelaci podvozku pouze svým úhlem naklopení. Ramena jsou pevně ukotvena v rozích podvozku a stejně jako ramena tvaru V jsou před nivelací sklopena směrem nahoru. Tyto typy ramen se mnohdy používají v teleskopickém provedení pro zvýšení stability zařízení a zvětšení pracovního prostoru.

Tento typ ramen nepatří k nejčastěji používaným z důvodu menší vyrovnávací přesnosti a větším nárokům na prostor. Využívá se spíše pro hrubou stabilizaci výškové techniky.



Obrázek 3.5: Princip sklopného ramene

3.2 Dle mechanismu akčních členů

Stabilizační ramena jsou tvořena mechatronickými částmi, z nichž nejhlavnější jsou akční členy (pohony), které uvádějí stabilizační ramena do pohybu. Prakticky u všech druhů stabilizačních ramen je využito pohonu lineárního typu. Tyto pohony lze rozdělit dle fyzikálního působení na:

- hydraulické,
- pneumatické,
- elektromechanické.

Při výběru pohonu pro nivelaci konkrétního zařízení jsou nejdůležitějšími parametry rychlost posunu, síla, přesnost a životnost. Níže jsou popsány jednotlivé typy pohonů spolu s jejich výhodami a nevýhodami v oblasti nivelace.

3.2.1 Hydraulické

Hydraulické pohony pracují na principu přenosu pracovní kapaliny. Dělí se na pohony hydrodynamické, které využívají kinetickou energii kapaliny, a hydrostatické, využívající fyzikální nestlačitelnost kapalin – tedy tlakovou energii kapalin. V případě nivelačních systémů se jedná o hydraulické válce (lineární hydromotory), jejichž výstupem je přímočarý vratný pohyb. Každé nivelační rameno tvořené hydraulickým válcem je připojeno k samostatnému hydraulickému čerpadlu se servořízením. U těchto pohonů lze jednoduše dosáhnout vysokého zvedacího výkonu, a proto se využívají především v těžké technice, například u jeřábů a bagrů. Velikosti používaných válců se pohybují až do průměru 1 m a do zdvihu 10 m. Nejčastěji jsou realizovány se systémem odpružení, kvůli lepšímu přenosu síly a tlumení. Odpružení válců je rozděleno do různých podpůrných skupin prostřednictvím otevírání a uzavírání hydraulických kulových ventilů, které je v souladu s požadavky na stabilitu v různých pracovních podmínkách.

Princip

Hydraulické pohony získávají energii z tlakové hydraulické kapaliny, kterou je obvykle olej. Hlavní částí je válec, v němž se pohybuje píst. Tyto válce jsou řízeny elektrohydraulickými proporcionálními ventily.

Pokud je potřeba co nejpřesněji zpětně detekovat nastavenou polohu pístu, jsou hydraulické válce osazeny magnetostrikčními a indukčními senzory. Magnetický prvek přenáší své magnetické pole na pístnici. Snímač namontovaný venku detekuje distribuci magnetického pole podél tyče a přemění ho na lineární výstupní signál, který odpovídá poloze pístu.

Výhody a nevýhody

Pracovní tlaky používané v hydraulických řídicích obvodech dosahují hodnot řádově až desítky MPa. Proto jsou hydraulické pohony schopné generovat velké síly nebo momenty při malých dobách přestavení a současně při malých rozměrech i tíze pohonů. Z těchto důvodů se používají u stavebních strojů a ve vojenské technice s běžnými zátěžemi v řádech desítek až stovek tun. Hydraulický pohon může, vlivem nestlačitelnosti kapaliny držet sílu nebo krouticí moment i bez provozu hydraulického čerpadla, proto není mnohdy potřeba žádného dodatečného blokování nastavené polohy, například brzdou. Hydraulické čerpadlo lze umístit i do značné vzdálenosti od pohonu s minimální tlakovou ztrátou. Navíc tyto systémy pracují s dostatečnou přesností polohování – až 0,1 mm a poměrně krátkým časem vyrovnávání, v řádech jednotek minut [26].

Nevýhodou hydraulických pohonů je vyšší cena, která se odvíjí od přesnosti výroby a jejich citlivosti na nečistoty v pracovní kapalině. Je například dokázáno, že až 80 % selhání hydrostatických systémů souvisí s kontaminací hydraulické kapaliny. Proto jsou součástí některých systémů monitorovací a čisticí zařízení. Hydraulické pohony jsou dále závislé na externích podmínkách, kde proměnná teplota vyvolává změnu viskozity kapaliny, a tím se mění průsaky i posuv válce. Z ekologického hlediska je velkou nevýhodou možné znečištění pracovní kapalinou, v případě úniku a ve většině případech je použitá kapalina hořlavá, což aplikaci hydraulických pohonů omezuje. Tyto systémy navíc vyžadují velké množství prvků – nádrže, čerpadla, ventily, chladiče, z tohoto důvodu jsou velké, těžké, drahé a složité na údržbu. Dále oproti elektromechanickým pohonům mají dlouhou reakční dobu, čímž je myšlena časová odezva na řídicí signál [26, 27].

3.2.2 Pneumatické

Pneumatické pohony pracují na stejném principu jako hydraulické, s tím rozdílem, že přenosovým médiem je pracovní plyn.

Jednou z možností jsou pneumatické písty, které se u nivelačních zařízení prakticky nepoužívají. Z důvodu obtížnějšího dosažení přesného sledování nebo polohování, kvůli poměrně velkému poměru třecí síly k výstupní síle a velké kompresi vzduchu. Dalším důvodem je nízký zvedací výkon a tuhost konvenčního pneumatického systému.

U některých nivelačních systémů se využívají takzvané pneumatické svaly, které mají až desetkrát větší zvedací výkon, než klasické pneumatické písty se stejným průměrem. Jejich hlavní využití je u podvozků automobilů, kde plní především funkci tlumičů, ale u některých vozů se začaly hojně využívat také pro nivelační účely. Využití nalézají také u malých stabilizačních zařízení, například pro lodní, nebo letecké aplikace [28].

Výhody a nevýhody

Pneumatické pohony mají velice jednoduchou konstrukci, se kterou souvisí mnoho výhod. Mezi hlavní výhody patří jejich cena, čistota provozu a možnost použití

v jakýchkoliv prostředích. Úniky stlačeného vzduchu ve vedení nebo ve válci nejsou nebezpečné pro obsluhu stroje, a proto jsou použitelné i v nebezpečných prostředích např. s nebezpečím výbuchu. S těmito pohony lze dosáhnout vysoké rychlosti a přesnosti polohování až – 0,025 mm. Lze je použít ve velkém rozmezí pracovních teplot – v běžných provedeních −20 °C až 80 °C, ve speciálních −40 °C až 250 °C.

Nevýhodou pneumatických pohonů, jak už bylo řečeno, je nízký zvedací výkon a tuhost. Tyto pohony jsou vlivem tlakových ztrát a stlačitelnosti vzduchu méně efektivní, obvykle nemohou unést zatížení větší než 300 kg a pro udržení a fixaci zátěže je nezbytný podpůrný systém. Také jsou zde vysoké náklady na výrobu stlačeného vzduchu. Stlačený vzduch jako pohonné médium je nejdražší energie pro pohonné jednotky (v porovnání s elektromotory). Většina ovládacích ventilů, redukčních ventilů a dalších částí pneumatických systémů je citlivá na nečistoty a zaolejování stlačeného vzduchu [27].

3.2.3 Elektromechanické

Tyto pohony využívají elektro-magneticko-mechanického principu a jedná se v podstatě pouze o lineární elektromotory. Elektromechanické pohony jsou podobné mechanickým pohonům, s tím rozdílem, že ovládací kotouč nebo klika je nahrazen elektromotorem, jehož rotační pohyb je převáděn na lineární posuv. Existuje mnoho konstrukcí lineárních pohonů. Tyto pohony jsou konstruovány tak, aby poskytovaly buď vysokou rychlost posunu, vysokou sílu, nebo určitý kompromis mezi těmito dvěma parametry.

Tento druh pohonu je vhodný především pro platformy se střední zátěží, tzn. s maximální pracovní zátěží v řádech jednotek až desítek tun. Nejčastější využití je proto ve vojenské technice a ve skladovacím průmyslu. Využívají se například pro různé doplňkové plošiny, umožňující snazší nakládání kamionů a jiných nákladních automobilů, nebo pro manipulaci s materiálem ve skladech.

Výhody a nevýhody

Jejich výhodou je vysoká spolehlivost a přesnost řízení polohy. S použitými lineárními převody a elektromotory lze dosáhnout vyšší citlivosti, kratší doby reakce na změnu náklonu, vysokého zrychlení a brzdicího momentu oproti využití jiných mechanismů. Jsou ideální pro polohování i mimo koncové polohy a poskytují úplnou kontrolu nad průběhem pohybu a mohou, dle typu řízení, obsahovat ovládání pro řízení rychlosti, polohy a síly. Další výhodou je jejich snadná oprava a údržba, protože se skládají pouze z mála mechanických částí a elektromotoru.

Nivelační systémy s tímto mechanismem nalézají své uplatnění pouze u platforem s nižší, až střední zátěží, proto nejsou příliš vhodné pro těžký průmysl, jsou ovšem případy, kde byly elektromechanické členy využity pro zátěže v řádech desítek tun. Nevýhodou je také počáteční cena, která je výrazně vyšší než u pneumatických pohonů. Také rozsah pracovních teplot je výrazně omezený oproti ostatním pohonům [27].

Tabulka 3.1: Porovnání vlastností akčních členů

Akční člen	**Maximální síla [kN]**	**Přesnost [mm]**	**Doba reakce [ms]**
Hydraulický	> 1000	0,1	1000
Pneumatický	< 3	0,025	100
Elektromechanický	< 100	0,001	1

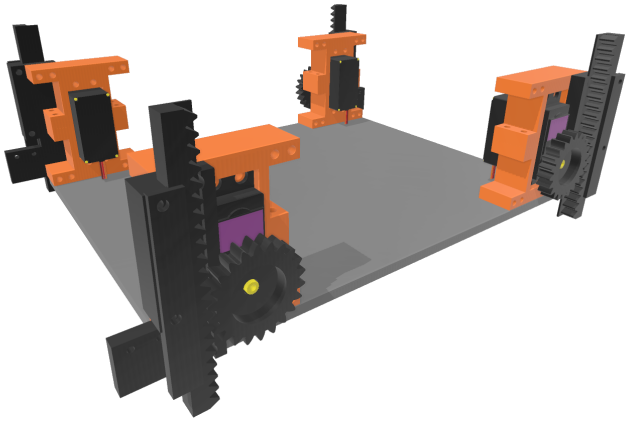
Pozn: Údaje jsou pouze orientační a dle různých zdrojů se mohou lišit

4 Návrh vlastního autonivelačního systému

V praktické části bakalářské práce se budeme zabývat návrhem vlastního autonivelačního systému, konkrétně algoritmem a způsobem řízení nivelačního zařízení. Navržený systém má za úkol řídit nivelační platformu, která byla vytvořena v rámci ročníkového projektu, viz [29].

4.1 Nivelační platforma

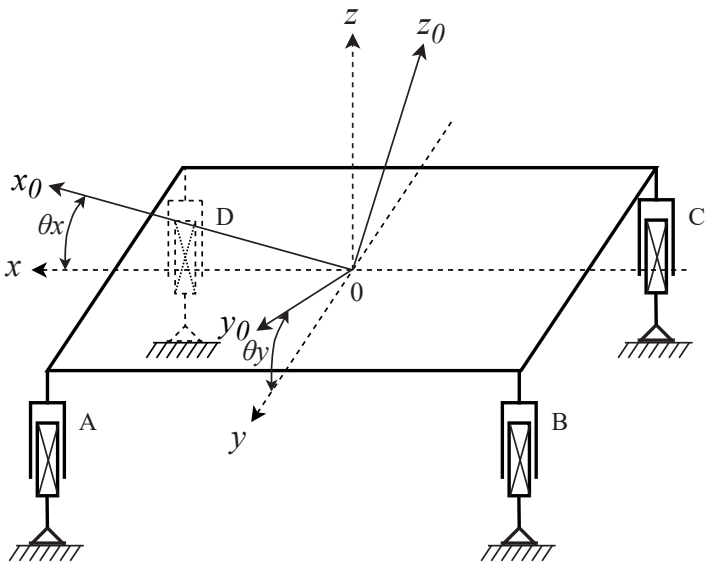
Před samotným návrhem je třeba se seznámit se zařízením, jež máme za úkol nivelovat. Jde například o: analýzu provedení konkrétního zařízení, rozpoložení a druhu použitých stabilizačních ramen. Rovněž je nutné seznámit se s hardwarovým omezením a jim přizpůsobit výsledný algoritmus. Níže si podrobně popíšeme použitý hardware – nivelační platformu.



Obrázek 4.1: Model nivelační platformy

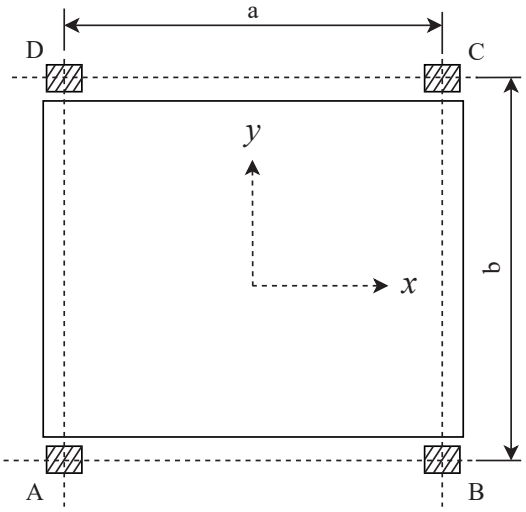
4.1.1 Koncepce platformy

Platforma je koncipována jako obdélníková deska se čtyřmi stabilizačními rameny umístěnými v pevně daných bodech, tzn. ramena není možné polohovat. Jednotlivá ramena, dále značena A, B, C a D, jsou umístěna symetricky v rozích desky, což zjednodušuje návrh řízení těchto ramen. Samotná deska je vyrobena z lehčeného PVC a rozměry desky jsou $240 \times 220 \times 4$mm. Koncepce platformy je znázorněna na obrázku 4.2.



Obrázek 4.2: Koncepce nivelační platformy

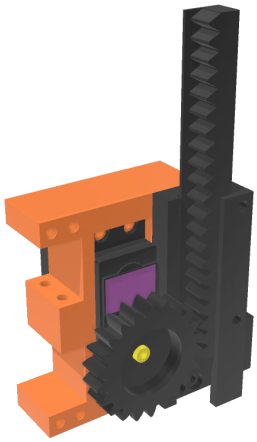
Konstrukce ramen je vytisknuta na 3D tiskárně a jednotlivá ramena jsou k desce přichycena tak, že přesahují okraj desky. Rozteč jednotlivých ramen je v ose x 240 mm (rozměr a) a v ose y 220 mm (rozměr b), viz obrázek 4.3. Tyto údaje jsou nezbytné pro návrh řídicího algoritmu.



Obrázek 4.3: Rozměry nivelační platformy

4.1.2 Stabilizační ramena

Použitá stabilizační ramena jsou výsuvná pouze vertikálně a jsou tvořena elektromechanickým lineárním členem. Zmíněný lineární člen je jednoduššího typu, v němž se využívá pouze ozubeného kola a hřebenu, na místo závitové tyče. Nevýhodou tohoto jednoduššího provedení je ztráta samosvornosti, kterou má výše zmíněný, běžně užívaný, lineární akční člen se závitovou tyčí. Samosvornost závitu se běžně využívá pro blokaci ramene v určité poloze, tato koncepce napomáhá při vysoké zátěži ramene. V našem případě jsme omezeni udržitelností hřídele použitého pohonu v nastavené poloze. Z tohoto důvodu je platforma určena pouze pro testování autonivelačního algoritmu, a není vhodné použité pohony příliš namáhat.



Obrázek 4.4: Model stabilizačního ramene

Pohonem lineárních členů jsou servomotory MG996R, přičemž konkrétní parametry těchto servomotorů jsou vypsány v tabulce 4.1. Pracovní rozsah těchto motorů je 120°. Vzhledem k rozsahu, délce ramen (přibližně 54 mm) a rozměrům desky, je s touto platformou možné vyrovnat náklon o maximální hodnotě přibližně 11,5° v ose x a 14,5° v ose y.

Tabulka 4.1: Specifikace servomotoru MG996R [30]

Rozměry	$40,7 \times 19,7 \times 42,9$ mm
Síla	9,4 kg/cm (4,8 V) – 11 kg/cm (6 V)
Rychlost	0,17 s/60° (4,8 V); 0,14 s/60° (6 V)
Operační napětí	4,8 V – 6 V
Provozní proud	500 mA – 900 mA (6 V)
Maximální proud	2,5 A / 6 V
Pracovní teplota	0 °C – 60 °C
Pracovní rozsah	cca 120°

4.2 Řízení autonivelace

Nyní se budeme zabývat postupem, kterým docílíme požadovaného vyrovnání zkonstruované platformy. Nabízí se několik postupů a úkolem je vybrat nejvhodnější pro docílení dostatečné přesnosti, opakovatelnosti a rychlosti autonivelace.

4.2.1 Metoda a proces nivelace

Jsou dvě základní metody, kterými lze řídit nivelaci, a ty jsou popsány následovně.

1. Nivelace v jednom směru – postup, při kterém dochází k úplnému vyrovnání v jedné ose, a poté v druhé, přičemž nezáleží na pořadí. Stabilizační ramena se pohybují postupně, vždy v párech. Nevýhodou je delší doba nivelace.

2. Nivelace ve více směrech najednou – při tomto postupu se pohybují všechna ramena najednou a do požadované polohy dorazí ve stejném čase. Rychlost nivelace je oproti předešlému postupu vyšší. Nevýhodou je ovšem velmi složitá a komplexní logika řízení, z důvodu rozdílných posunutí a rychlostí jednotlivých ramen. Pohyb jednotlivých ramen se při řízení navíc vzájemně omezuje. V praxi také může dojít k přetížení, protože na každé rameno působí jinak velká síla. Vzhledem k těmto omezením je vhodnější zvolit první z výše zmíněných postupů nivelace.

Vyrovnání platformy spočívá v úpravě relativní výšky čtyř bodů (rohů desky), které určují, v jakém stavu se platforma nachází. Z toho vyplývají tři možné způsoby úpravy těchto výšek.

1. Ukotvení nejvyššího bodu – nejvýše posazený bod se nepohybuje a ostatní tři body se snaží docílit úrovně nejvyššího bodu. Dochází k vysunutí ramen.

2. Ukotvení nejnižšího bodu – nejníže posazený bod se nepohybuje a ostatní tři body se snaží docílit úrovně nejnižšího bodu. Dochází k zasunutí ramen.

3. Ukotvení středního bodu – střední výška platformy je nehybná, u nižších bodů dochází k vysunutí ramen a zároveň u vyšších k zasunutí, dokud nejsou všechny body ve stejné úrovni.

Pokud jde o účinnost, třetí metoda je tou nejlepší. Nevýhodou je, že velmi často vede k vytvoření takzvaného virtuálního ramene, tzn. jedno rameno není zatíženo, nebo se nedotýká země. Druhá metoda nese podobný problém. Při použití první metody je směr pohybu všech ramen stejný, lze se takto vyhnout problémům s mechanickou vůlí použitých pohonů, takže přesnost nivelace je vyšší. Tato metoda je proto nejideálnější.

Nejprve je potřeba nalézt nejvýše položený bod ze znalosti náklonu v jednotlivých osách, to lze odvodit z následujících vztahů:

1. $\theta_x > 0, \theta_y > 0$. Bod A je nejvýše položený.
2. $\theta_x < 0, \theta_y > 0$. Bod B je nejvýše položený.
3. $\theta_x < 0, \theta_y < 0$. Bod C je nejvýše položený.
4. $\theta_x > 0, \theta_y < 0$. Bod D je nejvýše položený.

Poté, co je nalezen nejvýše položený bod, můžeme přejít k samotnému pohybu stabilizačních ramen. Výběr ovládaných ramen je závislý právě na nejvýše položenému bodu. Například, pokud je bod A vyhodnocen jako nejvýše položený, dojde nejdříve k vyrovnání v ose x, tzn. že ramena A a D zůstanou zafixovaná v počáteční poloze a ramena B a C se společně vysunou do požadované polohy. Poté, co je dosaženo nulového náklonu v ose x, dochází k vyrovnání v ose y. V této fázi ramena A a B zůstanou zafixována a k vysunutí dojde u ramen C a D.

4.2.2 Režimy řízení

Co se týče režimů řízení, existují dva základní typy – režim řízení s otevřenou smyčkou a s uzavřenou smyčkou.

Řízení s otevřenou smyčkou

V režimu s otevřenou smyčkou získáme žádanou hodnotu řízené veličiny, v našem případě úhel naklonění v jedné z os a ze znalosti aktuální hodnoty této veličiny vypočteme hodnotu řídicí veličiny, například výšku vysunutí konkrétních ramen. V praxi to vypadá tak, že každé hodnotě úhlu náklonu je přiřazena konkrétní výška vysunutí ramene. Po požadovaném vysunutí by byl proces nivelace ukončen. Tento režim řízení vyžaduje vysokou tuhost platformy, z důvodu snížení přesnosti nivelace při deformaci a následné změně geometrie platformy. Problémem může být také vysoká vůle mechanických částí stabilizačních ramen. Výhodou řízení je vysoká rychlost, a nevýhodou obtížné udržení přesnosti takového systému.

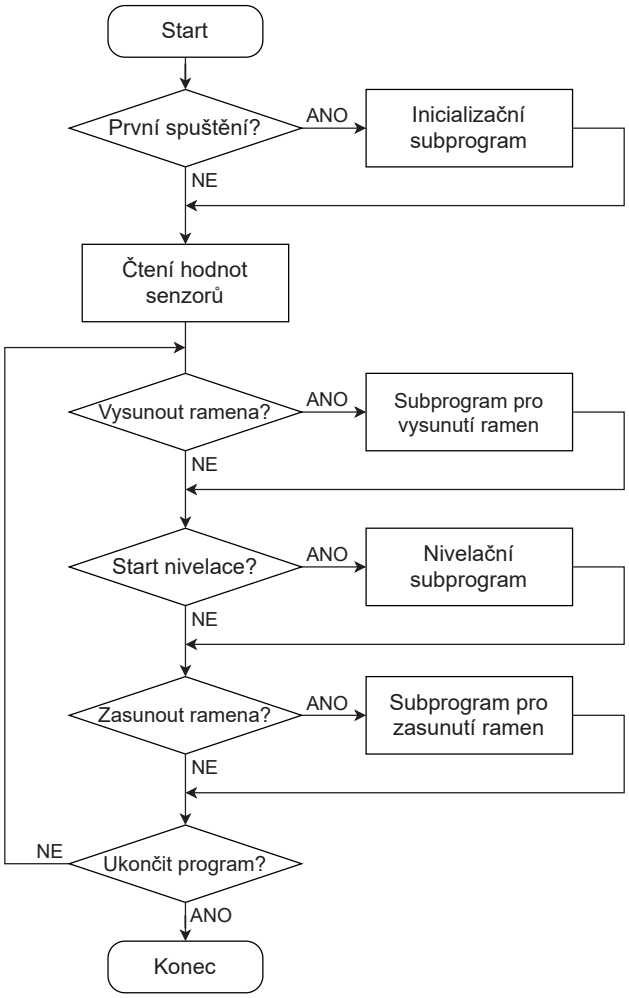
Řízení s uzavřenou smyčkou

Principem řízení s uzavřenou smyčkou je zpětnovazební regulace, kde se velikost akční veličiny řídí rozdílem žádané a aktuální hodnoty řízené veličiny, tedy regulační odchylkou. Akční veličinou v tomto případě rozumíme například nastavenou rychlost, kterou se bude rameno vysouvat. Při řízení kontinuálně sledujeme velikost regulační odchylky, pokud dosáhne dostatečně nízké úrovně, která splňuje danou přesnost, proces nivelace je ukončen. Takovýto systém se vyznačuje vysokou přesností a nižší rychlostí, z důvodu nepřetržitého sledování a zpracovávání vstupů a výstupů.

Našim cílem je oba režimy řízení využít, otestovat a vzájemně porovnat jejich dopad na celkový proces nivelace.

4.3 Hlavní program

Je nezbytné rozvrhnout jednotlivé části programu pro řízení platformy. Celé řízení se bude skládat z dílčích subprogramů, které se budou postupně vykonávat.



Obrázek 4.5: Vývojový diagram hlavního programu

Po spuštění programu je nutné ověřit, zda jde o první spuštění, a případně inicializovat potřebné parametry. Také je třeba zjistit, v jaké poloze se nacházejí jednotlivá ramena. Poté je možné přejít k samotné řídicí části programu. Předpokládejme, že při spuštění programu jsou jednotlivá ramena maximálně zasunuta. Pokud by se jednalo o typický pracovní stroj se stabilizačními rameny, nejprve by se nacházel ve stavu, kdy je postaven na vlastním mobilním podvozku, například na kolech, nebo pásovém podvozku. Prvním krokem je částečné vysunutí stabilizačních ramen a přenesení veškeré své váhy na zmíněná ramena. Tímto krokem končí první řídicí část programu.

Poté je možné spustit samotný proces nivelace. Základními kroky nivelačního subprogramu jsou: výčet výstupních hodnot konkrétního senzoru náklonu, výpočet parametrů popisujících aktuální stav platformy a následná úprava polohy jed-

notlivých ramen. Proces řízení ramen je závislý na zvolené metodě autonivelace, viz podkapitolu 4.2. V ideálním případě by mělo být možné všechna ramena zablokovat v příslušné poloze, po ukončení pohybu ramen. V našem případě možnost blokace chybí.

Po následném ukončení procesu nivelace by mělo být možné: nastavit ramena do počáteční polohy, částečně, nebo zcela zasunout ramena, celý program ukončit.

Popsaný postup programu lze znázornit zjednodušeným vývojovým diagramem, viz obrázek 4.5.

5 Vytvoření programu a implementace do vývojové platformy

Navržený postup řízení je třeba naprogramovat. Úkolem bylo pro vytvoření systému využít programové prostředí Matlab, případně jeho nadstavbu Simulink, a následně program implementovat do vybraného mikroprocesoru. Pro tento účel byla vybrána vývojová platforma Arduino s mikroprocesorem ATmega.

5.1 Vývojová platforma Arduino

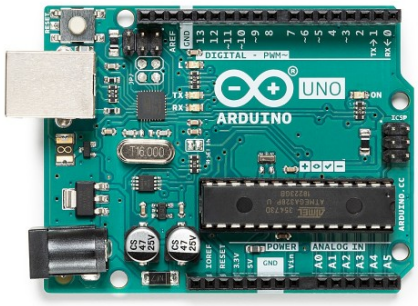
Vývojová platforma Arduino je elektronická platforma zahrnující hardware i software v rámci open-source projektu. Software zahrnuje vývojové prostředí a programovací jazyk na bázi jazyka C. Vyráběné platformy spadají do skupiny mikrořadičů s harvardskou architekturou. Jsou typické svými procesory od firmy Atmel. Mezi výhody této platformy patří:

- Nízká cena – desky Arduino jsou oproti ostatním platformám cenově dostupnější.
- Multiplatformnost – Software Arduino funguje na operačních systémech Windows, Macintosh OSX a Linux.
- Open source a rozšiřitelný SW – Software Arduino je open source nástroj, který umožňuje různá rozšíření. Samotný jazyk lze rozšířit knihovnami C++.
- Open source a rozšiřitelný HW – Hardware Arduino také umožňuje různá rozšíření, například pomocí Shieldů určených pro řízení běžných typů motorů.

V našem případě byla zvolena platforma Arduino z důvodu požadavku využít programové prostředí Matlab, pro které jsou dostupné rozšiřující balíčky právě pro komunikaci s Arduinem, viz podkapitola 5.3.1. Nabízí se možnost otestovat zmíněné rozšiřující balíčky a osvojit si práci s Matlabem v kombinaci s platformou Arduino. Pro tuto práci byly dostupné platformy Arduino Uno a Due. Výhodou modelu Due je nativní CAN port, potřebný pro zprovoznění komunikace s jedním z inklinometrů, ale stále je třeba dodat rozšiřující modul pro realizaci budícího členu tohoto portu. Nevýhodou Arduina Due je pracovní napětí, které je pouze 3,3 V oproti standardním 5 V. Pro řízení platformy byl vybrán model Uno v kombinaci s rozšiřujícím CAN modulem.

5.1.1 Arduino Uno

Arduino Uno označované také jako Genuino Uno je řízeno 8bitovým procesorem ATmega328P s taktovací frekvencí 16 MHz. Platforma disponuje 6 analogovými vstupními piny, 14 digitálními I/O piny, z nichž 6 lze použít jako PWM výstupy. Součástí jsou konektory ICSP (In-Circuit Serial Programming). Pro komunikaci s PC má rovněž zabudovaný USB B konektor, vedle něhož je napájecí konektor.



Obrázek 5.1: Platforma Arduino Uno [31]

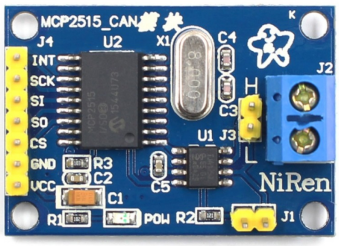
Technické specifikace desky jsou vypsány v Tabulce 5.1. Samotný procesor je předprogramovaný pomocí bootloaderu, který umožňuje nahrát nově vytvořený kód bez použití externího HW programátoru. Pro využití jiných možností je zde zmiňovaný ICSP konektor. Model Uno neobsahuje nativní rozhraní sběrnice CAN, proto bylo pro připojení jednoho z inklinometrů potřeba využít rozšiřující modul 5.1.2.

Tabulka 5.1: Specifikace platformy Arduino Uno [32]

MCU	ATmega328
Převodník USB	CH340
Pracovní napětí	5 V
Vstupní napětí	7 – 12 V (max 6 – 20 V)
I/O Piny	14 (6 PWM)
Analog. vstupy	6
DC Proud I/O Piny	40 mA
DC Proud 3,3 V Pin	50 mA
Flash	32 kB (ATmega328) 0,5 kB pro bootloader
SRAM	2 kB (ATmega328)
EEPROM	1 kB (ATmega328)
Taktovací frekvence	16 MHz

5.1.2 CAN Modul MCP2515

Modul představuje CAN převodník pro rozšíření základní desky Arduino Uno o komunikační rozhraní CAN.

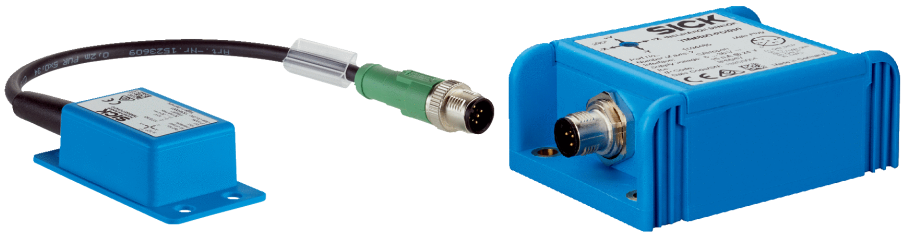


Obrázek 5.2: CAN Modul MCP2515 [33]

Jedná se o modul osazený čipem MCP2515, podporující CAN specifikaci 2.0B. Lze s ním přijímat i posílat standardní a rozšířené rámce. Čip má dvě masky a šest filtrů k odfiltrování nevyžádaných zpráv, následkem je snížení zatížení mikrokontroléru.

5.2 Snímače

Pro řízení jsou využity pouze senzory náklonu. K tomuto účelu jsou v našem případě dostupné dva inklinometry od firmy Sick [34] s různými parametry a komunikačním rozhraním. Cílem je oba tyto senzory otestovat.



Obrázek 5.3: Použité inklinometry (vlevo TMM55E, vpravo TMM88A) [34]

5.2.1 Snímač náklonu TMM55E

Prvním snímačem je dvouosý inklinometr s analogovým rozhraním. Jedná se o průmyslový snímač, využívaný právě pro pracovní stroje, například v těžkém průmyslu nebo v automatizaci. Důležitými údaji jsou pro nás: druh komunikačního rozhraní (které je v tomto případě analogové), přesnost a rozsah měření. Jeho rozsah je $\pm$ 10° a přesnost 0,15°, což jsou v případě našeho využití ideální parametry. Kompletní seznam parametrů je vypsán v tabulce níže.

Tabulka 5.2: Specifikace inklinometru TMM55E-POH010 [35]

Počet os	2
Rozsah měření	± 10°
Rozlišení	0,01°
Přesnost	0,15°
Kompenzovaná interf. citlivost (dvouosá)	Typ. ± 0,25°, max. ± 0,5°
Teplotní koeficient (nulový bod)	Max. ± 0,009°/K
Mezní frekvence	18 Hz
Vzorkovací frekvence	1,95 kHz
Komunikační rozhraní	Analogové / 0...10 V
Odpor zátěže	< 930 Ω
Inicializační čas	75 ms
Napájecí napětí	11 V DC...30 V DC
Příkon	15...25 mA

5.2.2 Snímač náklonu TMM88D

Druhý inklinometr je také dvouosý a využívá komunikačního rozhraní CANopen. Jeho pracovní rozsah je ± 90° s přesností (dynamická přesnost) 0,5°. Využívá se především v robotice a pro nivelaci pracovních strojů není příliš vhodný. V oblasti nivelace je žádána vyšší přesnost a není třeba tak široký rozsah měření.

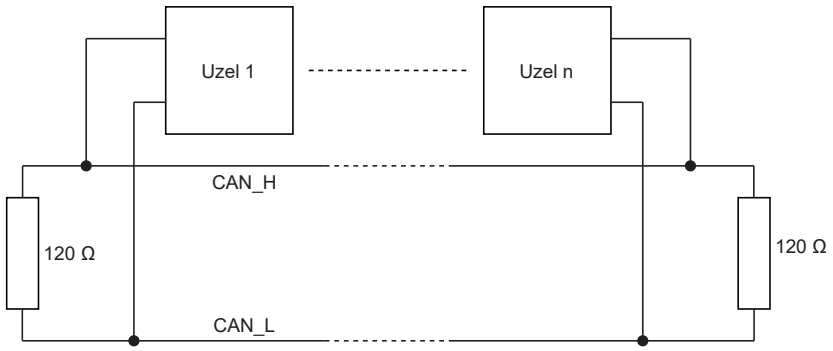
Tabulka 5.3: Specifikace inklinometru TMM88D-PCI090 [36]

Počet os	2
Rozsah měření	± 90°
Rozlišení	0,01°
Statická přesnost	0,3°
Dynamická přesnost	0,5°
Teplotní koeficient (nulový bod)	Max. ± 0,01°/K
Mezní frekvence	0,1 Hz...25 Hz
Vzorkovací frekvence	200 Hz
Komunikační rozhraní	CANopen
Rychlost přenosu dat (baud rate)	10 kbit/s...1000 kbit/s
Inicializační čas	120 ms
Napájecí napětí	8 V DC...36 V DC
Příkon	< 15 mA (24 V)

5.2.3 CAN komunikace

Druhý zmiňovaný snímač komunikuje po sériové sběrnici CAN. Jedná se o sběrnici nejčastěji využívanou pro komunikační síť snímačů a funkčních jednotek v automobilu. CAN patří k průmyslovým komunikačním sítím označovaným jako provozní sběrnice, fieldbus.

Fyzická vrstva je tvořena dvěma vodiči označenými CAN_H a CAN_L. Jedná se o symetrický spoj z důvodu zvýšení odolnosti proti rušení a komunikační rychlosti. Jsou rozlišovány dvě napěťové úrovně dominantní a recesivní, reprezentující jednobitovou informaci.



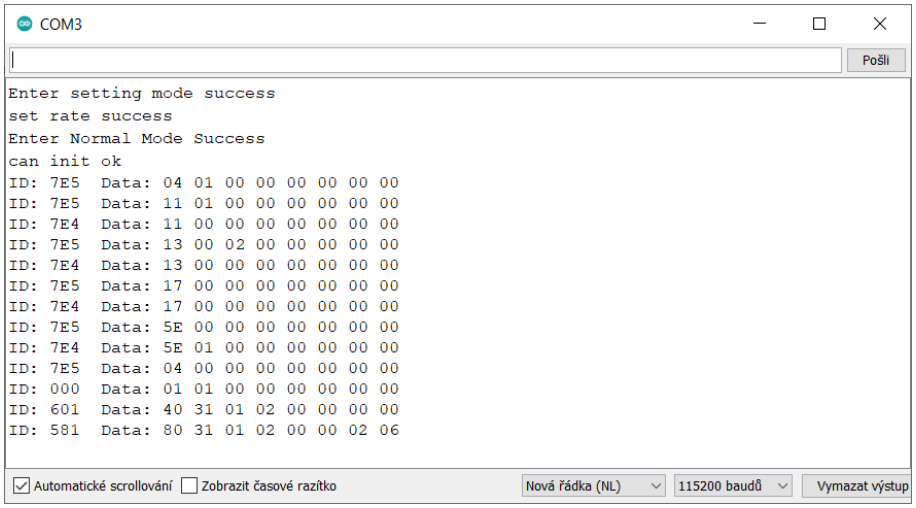
Obrázek 5.4: Fyzická vrstva CAN sběrnice

CAN představuje protokol typu multi-master, tzn. jakýkoliv uzel na sběrnici může pracovat jako master a řídit jiné uzly (typu slave). V našem případě Arduino Uno (master) má za úkol řídit inklinometr (slave).

Samotná komunikace probíhá pomocí zpráv, takzvaných rámců, které mohou být trojího typu: datová zpráva (Data Frame), žádost o data (Remote Frame), chybová zpráva (Error Frame), případně zpráva o přetížení (Overload Frame), která je podobná chybové zprávě.

Testování komunikace se senzorem probíhalo v prostředí Arduino IDE za využití knihovny Seees_Arduino_CAN, vytvořené pro příslušný CAN modul. Dle provozního manuálu k inklinometru TMM88D [37] bylo třeba snímač nejdříve nakonfigurovat, přiřadit unikátní identifikační číslo (Node ID), nastavit komunikační rychlost (modulační rychlost, tzv. baud rate) atd. Konfiguraci bylo potřebné uložit do vnitřní paměti EEPROM. Pro ověření úspěšné konfigurace byl snímač odpojen od napájení a po opětovném připojení byl příslušnými datovými zprávami požádán o informace o aktuální konfiguraci. Na obrázku 5.5 lze vidět ukázku komunikace s inklinometrem.

Komunikace se snímačem a jeho konfigurace proběhla úspěšně. Samotná naměřená data se ovšem získat nepodařilo. Pro získání dat je třeba zaslat datovou zprávu s žádostí o konkrétní naměřená data a snímač by měl zpětně poslat odpověď (princip request/response telegramu). Při pokusu o vyžádání jakýchkoliv naměřených dat bylo zasláno hlášení, že se snímač nachází v režimu Pre-operational. V tomto režimu nejspíše nedochází k měření, je proto potřeba uvést snímač do stavu Operational, což se bohužel nepodařilo. Z tohoto důvodu byl v dalších částech práce využit pouze analogový snímač náklonu.



Obrázek 5.5: Ukázka komunikace se snímačem TMM88D

5.3 Prostředí Matlab

Pro realizaci programu je využito programové prostředí a skriptovací jazyk Matlab (konkrétně verze R2021a) od společnosti MathWorks. Prostředí umožňuje numerické výpočty, práci s maticemi, vykreslování funkcí a dat, implementaci algoritmů a vytváření jednoduchých uživatelských rozhraní. Ačkoli je Matlab určen primárně pro numerické výpočty, dostupné doplňkové sady nástrojů a nadstavby nabízejí využití v široké paletě aplikací. Například nadstavba Simulink využívá Matlab a jeho funkce k simulaci dynamických systémů a přidává možnosti grafických simulací.

Pro účely práce jsou dostupné přídavné nástroje pro podporu platformy Arduino. S těmito nástroji lze přes sériovou komunikaci externě řídit vstupy a výstupy platformy.

5.3.1 Matlab Support Package for Arduino HW

Jedná se o nástroj umožňující využívat Arduino jako vstupně výstupní zařízení, přičemž řídicí proces je realizován v Matlab skriptu. Kód je psán a spouštěn v Matlab skriptu a je možné zapisovat i číst data z jednotlivých pinů bez kompilace jakéhokoliv kódu.

Komunikace s ostatními zařízeními může probíhat pomocí periferních zařízení GPIO, I2C, SPI, CAN a Serial. Komunikace s deskou Arduino je možná přes USB, bezdrátově přes Wi-Fi, nebo Bluetooth. Nástroj podporuje například řízení: servomotorů, rotačních enkodérů, ultrazvukových senzorů a jiných periferií.

Pro realizaci spojení s Arduinem je přidána funkce `arduino()`, která vrací parametry připojení a další ovládací funkce se na ni odkazují. Mezi parametry funkce patří: informace o portu (na kterém je deska připojena), informace o typu desky, I/O piny (které jsou přístupné) a typy funkcí z použité rozšiřující knihovny.

V našem programu jsou zároveň důležité funkce pro čtení a zápis I/O pinů a řízení servomotorů. Pomocí funkce `configurePin(a, pin, mode)` lze nastavit vybraný pin na konkrétní typ módu. Mód jednotlivých pinů může být například `DigitalInput`, pro čtení dat z digitálních pinů, nebo `DigitalOutput`, pro jejich zápis. Dále je zde mód `PWM` a mód servo.

Samotné čtení, nebo zápis dat na digitální vstupy/výstupy probíhá pomocí funkcí `readDigitalPin(arduino, pin)`, `writeDigitalPin(arduino, pin)` a pro čtení analogových vstupů slouží funkce `readVoltage(arduino, pin)`, která vrací hodnotu ve voltech.

Pro zmíněná serva je potřeba před samotným řízením vytvořit spojení. To lze provést funkcí `servo(arduino, pin)`, která vytvoří objekt `servo` na nějž se budeme při řízení odkazovat. Ovládání je realizováno dvěma základními funkcemi. První funkce je `readPosition(servo)`, která umožňuje určit pozici servomotoru. Výstupem funkce je hodnota v intervalu 0,00 – 1,00, která popisuje úhel natočení. Stejným způsobem funguje funkce `writePosition(servo, position)` sloužící pro nastavení úhlu natočení. Parametrem `position` určíme požadovaný úhel.

5.3.2 Simulink Support Package for Arduino HW

Tento nástroj naopak umožňuje automaticky vytvořit kód pro Arduino z vlastního Simulink modelu. Oproti prvnímu zmiňovanému nástroji je výhodou, že vytvořený program lze zkompilovat a nahrát do platformy, která funguje bez nutnosti připojení k prostředí Matlab.

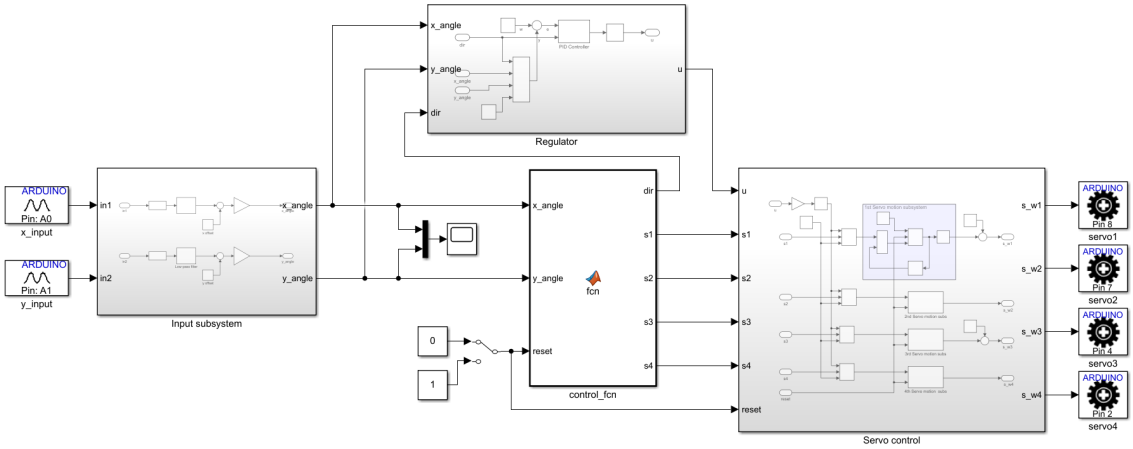
Nástroj přidává do knihovny funkčních bloků sekci s podporou pro Arduino. V ní je několik skupin s novými bloky. Například skupina Common obsahuje běžné bloky, které slouží pro jednoduché čtení, nebo zápis dat na analogové či digitální vstupy a výstupy Arduina. Součástí základní knihovny jsou bloky pro řízení servomotorů. Dále je zde například skupina Sensors pro zpracování dat z konkrétních senzorů, nebo skupina WiFi a Ethernet pro řízení shieldů pro bezdrátovou komunikaci.

5.4 Simulink model

Pro vytvoření autonivelačního programu bylo nejprve využito prostředí Matlab Simulink a nástroj Simulink Support Package pro podporu Arduina. Zpracovávání všech signálů a čtení vstupů probíhá v reálném čase, proto je prostředí vhodné pro testování řízení s uzavřenou smyčkou. Byl tedy vytvořen Simulink model pro řízení autonivelace využívající PID regulátor.

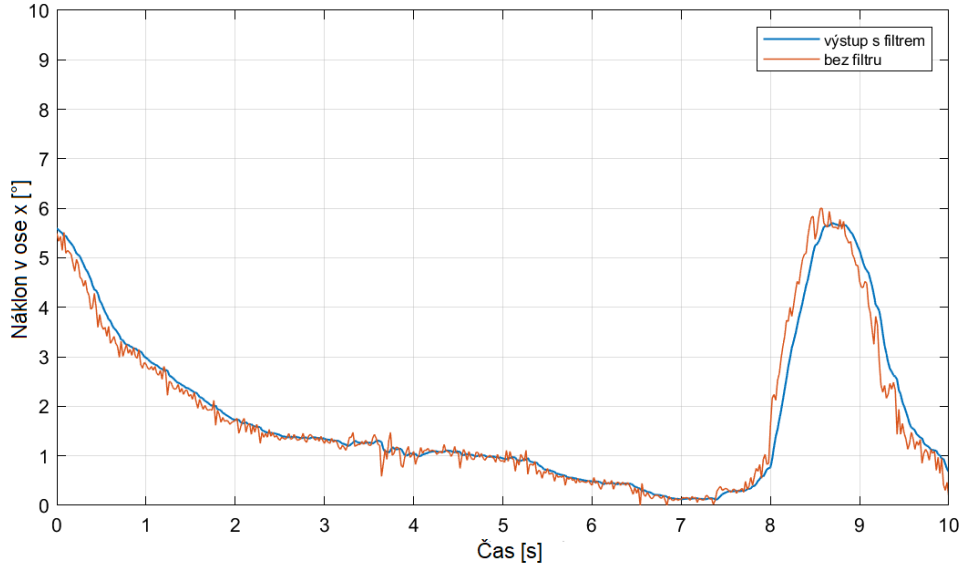
5.4.1 Zpracování výstupu senzoru náklonu

V první fázi je třeba číst výstup ze senzoru náklonu. K této operaci sloužil inklinometr s analogovým rozhraním, který vrací dvě hodnoty napětí v rozmezí 0 až 10 V pro náklon v každé ose. Výstupy inklinometru jsou převedeny na napětí v rozmezí 0 až 5 V a jsou připojeny na analogové vstupy platformy Arduino Uno. Vstupy



Obrázek 5.6: Simulink model autonivelačního systému

jsou čteny pomocí bloku `Analog Input`, který vrací 10bitovou hodnotu (0 až 1024) reprezentující velikost napětí. Na vstupy jsou připojeny bloky s funkcí Low-Pass filtru pro odstranění zákmitů na výstupech senzoru. Výstup je hladký a bez zákmitů zkreslujících skutečnou hodnotu, avšak dochází k většímu zpoždění signálu. Hodnoty jsou následně přepočteny na úhly ve stupních (signály `x_angle` a `y_angle`). Na obrázku 5.7 je pro porovnání vidět výstup z inklinometru bez filtrace a s filtrací.

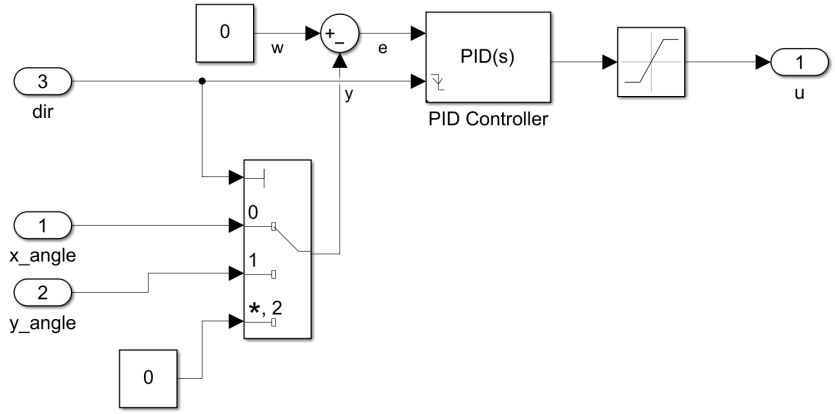


Obrázek 5.7: Výstup z inklinometru

5.4.2 Regulace

Nyní máme informaci o náklonu platformy a můžeme přejít k samotnému řízení stabilizačních ramen. Hodnota náklonu je následně vstupem dvou subsystémů. První

subsystém (viz obrázek 5.8) řídí regulaci rychlosti servomotorů na základě velikosti náklonu v jedné z os. Skládá se z PID regulátoru v regulační smyčce s žádanou hodnotou 0° a akční veličinou v rozmezí ± 100, která je následně vstupem do druhého subsystému a řídí rychlost otáčení jednotlivých servomotorů. Regulační smyčka vyžaduje řídicí logiku pro resetování bloku PID a přepínání mezi výstupy s aktuální hodnotou náklonu v ose x a v ose y. K tomu slouží blok `Switch`, jehož vstupem je proměnná `dir` z druhého subsystému, která nese informaci o aktuální ose, ve které probíhá autonivelace.

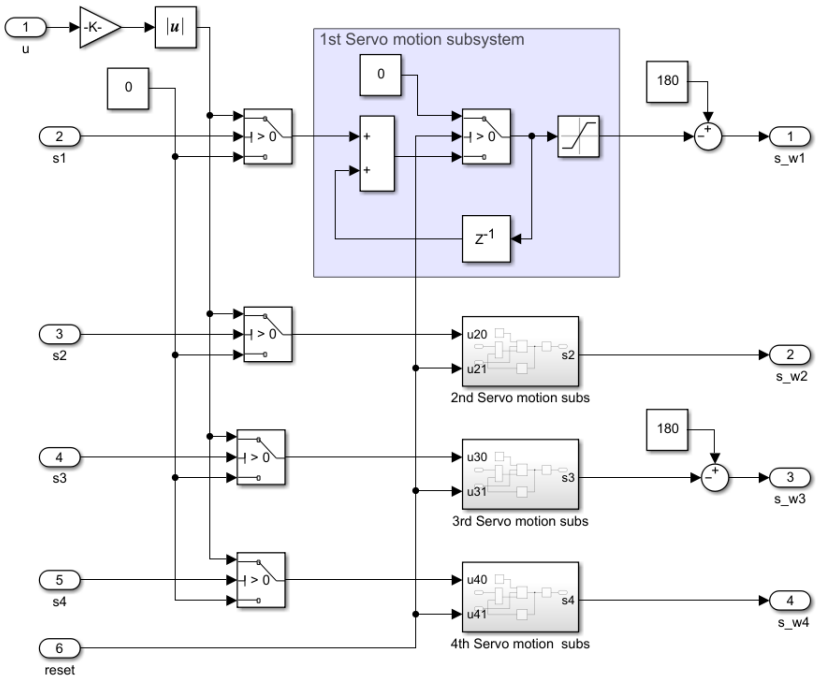


Obrázek 5.8: Simulink model subsystému pro regulaci

5.4.3 Řízení servomotorů

Druhý subsystém (viz obrázek 5.9) je připojen na funkční blok (`control_fcn`) řídící výběr konkrétních servomotorů, které budou ovládány na základě výstupu z regulátoru. Například, pokud probíhá nivelace v ose x a je třeba vysunout dvojici ramen B a C, je výstupní hodnota regulátoru nastavena na vstupy bloků ovládající odpovídající servomotory. Ovládání servomotorů je realizováno pomocí bloků `Standard Servo Write`, kterým jsou přiřazeny konkrétní výstupní PWM piny Arduina. Požadovaná pozice servomotorů v rozmezí 0 – 180 je na výstupu jednotlivých subsystémů `Servo motion subsystem`, které ukládají minulou pozici a dle hodnoty výstupu regulátoru ji inkrementují.

Pro zprovoznění modelu je třeba spustit simulaci v externím módu, kdy je na základě vytvořeného blokového schématu vygenerován kód v jazyce C, jenž je následně zkompilován a nahrán do platformy Arduino. Vygenerovaný kód je bohužel ve většině případech příliš objemný a zabírá velkou část paměti, proto bylo třeba výsledné schéma upravit a zjednodušit do aktuální podoby. Další nevýhodou je omezená využitelnost bloků pro řízení servomotorů, jelikož dochází k časté chybovosti při kompilaci programů s těmito bloky. Ze stejného důvodu dokumentace k nástroji Simulink Support Package for Arduino HW nedoporučuje používat tyto bloky. Výše popsaný model je využit pouze pro jednoduché testování řízení s PID regulátorem a pro plnohodnotné testování nivelační platformy je vhodnější program popsaný v následující kapitole.



Obrázek 5.9: Simulink model subsystému pro řízení servomotorů

5.5 Matlab program

Způsob řízení s otevřenou smyčkou je otestován pomocí programu psaném v Matlab skriptu a využívá rozšiřující nástroj Matlab Support Package for Arduino HW. Hlavní část kódu využívá jednoduché GUI, tři vytvořené funkce pro ovládání servomotorů a čtení výstupu inklinometru. Jsou vytvořeny i další pomocné funkce například pro úpravu a sběr dat pro výsledný graf zobrazený v GUI, ale těmi se nebudeme zabývat.

5.5.1 Funkce pro zpracování výstupu senzoru náklonu

První funkcí je `ReadAngles(a, x_const, y_const)`, která slouží pro zpracování výstupu z analogového inklinometru. Této funkci je jako vstupní parametr předán objekt Arduino, který je vytvořen při prvotní inicializaci platformy. Dalšími parametry jsou konstanty potřebné pro přepočet napětí na výsledné úhly náklonu. Nejdříve jsou desetkrát čteny hodnoty napětí z obou analogových vstupů, následně jsou vypočítány průměry z těchto deseti hodnot. Dochází k alespoň částečnému snížení chybovosti měření analogových vstupů. Poté jsou napětí přepočtena na úhly náklonu v ose x a y. Napětí na analogových vstupech může být v rozmezí $0 - 5$ V s krokem přibližně 4,88 mV. Po přepočtení napětí na úhly ve stupních dostaneme dvě hodnoty v rozsahu $\pm$ 10° s krokem 0,0195°. Z těchto dvou vypočtených hodnot je vytvořen dvousložkový vektor `[x_angle, y_angle]`, který funkce vrací jako výstupní parametr.

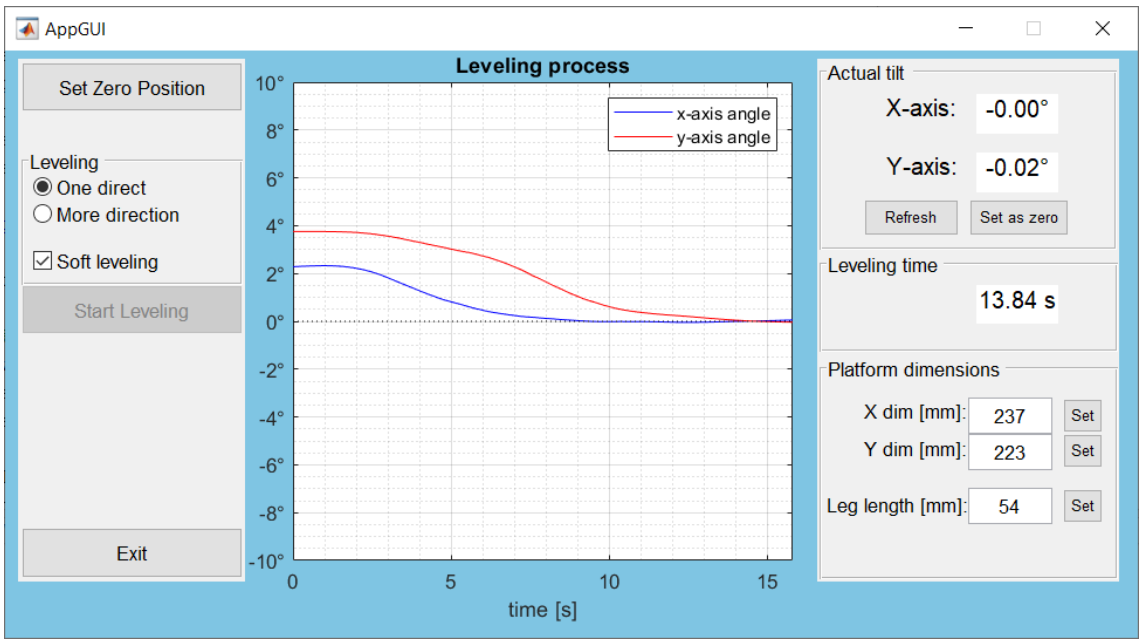
5.5.2 Funkce pro řízení servomotorů

Pro řízení servomotorů jsou vytvořeny dvě následující funkce. První funkce `SetPosition(servo, end_pos, st, vel)` slouží pro plynulý pohyb vybraných serv do požadované polohy. Parametry této funkce jsou: vektor obsahující objekty jednotlivých servomotorů, vektor s cílovými pozicemi servomotorů, nejmenší krok servomotorů a frekvence kroků. Průběhem funkce jsou čteny aktuální polohy servomotorů a je řízen jejich pohyb. Pokud jsou polohy jednotlivých ramen odlišné a do cílové polohy by dorazila v odlišných časech, je upraven krok příslušných servomotorů, což zamezuje problému s ramenem visícím ve vzduchu. Tímto způsobem je umožněno testovat autonivelaci ve více směrech najednou, navíc je ošetřeno, že nastavená poloha serv nebude mimo interval 0,00 – 1,00.

Druhá vytvořená funkce `ServoStep(servo, step)` slouží pro jemné řízení servomotorů po jednotlivých krocích. Parametry jsou: vektor s objekty servomotorů a velikost kroku. Funkce slouží pro režim jemné nivelace.

5.5.3 Průběh programu

Po spuštění skriptu s názvem `NivelationMain`, dojde nejdříve k deklaraci potřebných proměnných, následně k inicializaci platformy Arduino a konfiguraci potřebných vstupních i výstupních pinů. Součástí konfigurace je i prvotní nastavení všech servomotorů. Následuje nekonečná `While` smyčka, která spouští uživatelské rozhraní (viz obrázek 5.10) a kontroluje, jaké další akce uživatel zvolil. Rozhraní umožňuje nastavit platformu do nulové pozice a spustit autonivelaci v požadovaném režimu. Pokud je stisknuto tlačítko EXIT, program je ukončen.



Obrázek 5.10: Uživatelské rozhraní NivelationGUI

Při stisknutí tlačítka SET ZERO POSITION jsou pomocí funkce `SetPosition()` všechny servomotory nastaveny do nulové pozice. Po ukončení pohybu serv je znovu zobrazeno GUI a zpřístupní se tlačítko START LEVELING pro spuštění příslušného režimu nivelace.

Pomocí dvou přepínacích tlačítek (prvky *radio button*) lze vybrat jeden ze dvou režimů autonivelace: One direct, More direction. První možnost, One direct, nastaví režim nivelace v jednom směru (viz podkapitolu 4.2.1), při tomto výběru a následném stisknutí tlačítka START LEVELING, je spuštěna sekvence příslušných kroků. V prvním kroku je spuštěna funkce `ReadAngles()` pro získání informace o aktuálním náklonu platformy. Ze získaných hodnot je vypočtena poloha natočení příslušné dvojice servomotorů, jež jsou do této polohy nastaveny pomocí funkce `SetPosition()`. Po dokončení nivelace v jedné ose následuje stejný postup pro nivelaci ve druhé ose. Při výběru možnosti More direction dochází ke spuštění stejné sekvence kroků s tím rozdílem, že je vypočtena jedna cílová poloha všech serv, jež jsou následně všechny řízeny najednou, tzn. všechna ramena zaujmou cílovou pozici ve stejný čas.

Navíc je možné vybrat režim jemné nivelace pomocí zaškrtávacího pole s názvem Soft Leveling. Při vybrání této dodatečné možnosti dojde na konci hlavního režimu autonivelace k jednoduchému řízení s uzavřenou smyčkou. Jednotlivá ramena jsou po jemných krocích vysouvána, dokud není dosaženo nulového náklonu s určitou přesností. Dodatečná fáze slouží pro vyšší přesnost autonivelace.

Výpočet polohy servomotorů

Výpočet cílové polohy servomotorů probíhá například následovně:

Pokud je úhel náklonu v ose x i v ose y větší než 0, je nejvýše položený bod A, proto je třeba vysunout ramena B, C a D. V prvním kroku je třeba vysunout dvojici ramen B a C k provedení nivelace v ose x. Proměnné a a b vyjadřují rozměry platformy a proměnné θ_x a θ_y jsou úhly náklonu v jednotlivých osách. Potřebnou vzdálenost vysunutí spočítáme dle vztahu:

$$pos_B = pos_{C1} = a \cdot \sin|\theta_x| \tag{5.1}$$

a poté následuje výpočet pro nivelaci v ose y. Ramena D a C vysuneme o následující vzdálenost:

$$pos_D = pos_{C2} = b \cdot \sin|\theta_y|. \tag{5.2}$$

Dále máme vektor *position*, jehož prvky popisují výslednou pozici jednotlivých servomotorů, tento vektor je argumentem funkce `SetPosition()`. Dále známe délku ramen značenou l_{max}, která odpovídá zvednutí platformy při maximálním otočení servomotorů. V režimu One direct jsou servomotory přenastaveny o tuto hodnotu:

$$position = [0; pos_B; pos_{C1}; 0]/l_{max}, \tag{5.3}$$

z intervalu 0,00 – 1,00, která odpovídá otočení hřídele o 0 až 180°.

Následně jsou servomotory přenastaveny o tuto hodnotu:

$$position = [0; 0; pos_{C2}; pos_D]/l_{max}. \tag{5.4}$$

Pokud jsme v režimu More direction, jsou všechny servomotory nastaveny rovnou do této výsledné polohy:

$$position = [0; pos_B; pos_{C1} + pos_{C2}; pos_D]/l_{max}. \tag{5.5}$$

5.5.4 Uživatelské rozhraní

Součástí uživatelského rozhraní (viz obrázek 5.10) jsou dva postranní panely a graf zobrazující průběh autonivelace.

Pravý panel obsahuje tři tlačítka pro: nastavení nulové pozice, spuštění autonivelace a vypnutí programu, dále tři zaškrtávací pole pro nastavení režimu řízení. Na pravém panelu jsou informace ohledně: náklonu, době autonivelace a nastavených parametrech platformy. V horní části pravého panelu je zobrazen aktuální náklon platformy v obou osách a dvě ovládací tlačítka. Pomocí tlačítka Refresh lze znovu načíst hodnoty náklonu a tlačítkem Set as zero lze změnit referenční hodnotu náklonu. Dále je zde vypsána doba autonivelace v sekundách s přesností na dvě desetinná místa. Níže jsou parametry testovací platformy potřebné pro výpočty při řízení nivelace, které lze přenastavit pomocí tlačítek set. Je zde rozměr platformy v ose x a y a délka stabilizačních ramen, to vše v milimetrech.

V prostřední části rozhraní je graf zobrazující průběh poslední spuštěné autonivelace. V grafu jsou společně zobrazeny průběhy náklonu v jednotlivých osách a referenční nulová konstantní funkce, k níž se snažíme přiblížit.

6 Ověření vlastností navrženého systému

Po dokončení hardwaru a softwaru přichází na řadu testování vytvořeného systému. Bylo třeba ověřit základní vlastnosti navrženého řízení autonivelace, tzn. ověřit přesnost, rychlost a porovnat jednotlivé metody řízení.

6.1 Přesnost nivelace

Celková přesnost nivelace se odvíjí od chyb způsobených několika faktory, například nepřesností inklinometru, nebo servomechanismu.

V prvním případě, řízení pomocí PID regulátoru, je řízení servomotorů spojité a přesnost je vyšší. Nezanedbatelná chyba je tvořena nepřesností použitého inklinometru $\Delta\theta_1$ a konstrukcí stabilizačních ramen. Pokud bychom přepočetli chyby akčních členů ramen na chyby úrovně platformy, dostali bychom přenosovou chybu pastorku e_1, hřebenu e_2 a servomotoru e_3. Výsledná chyba akčního členu by tedy byla:

$$e = e_1 + e_2 + e_3. \tag{6.1}$$

Se znalostí rozměrů platformy (a a b) by výpočet absolutní chyby nivelace v obou osách vypadal následovně:

$$\Delta\theta_x = \Delta\theta_1 + \operatorname{arctg}(e/a), \tag{6.2}$$

$$\Delta\theta_y = \Delta\theta_1 + \operatorname{arctg}(e/b). \tag{6.3}$$

Ostatní chyby, jako například chybu AD převodníku pro čtení analogového výstupu inklinometru, nebo chybu vycházející z geometrických nepřesností konstrukce platformy, lze zanedbat. V případě druhého způsobu řízení je další nepřesnost způsobena chybou nejmenšího nastaveného kroku servomotorů.

6.2 Rychlost nivelace

Vzhledem k tomu, že použitá platforma je navržena pouze pro testování autonivelačního algoritmu, je bezpředmětné zabývat se konkrétními výsledky rychlosti nivelace. V praxi by nivelace pracovních strojů trvala v řádech minut — z důvodu použití

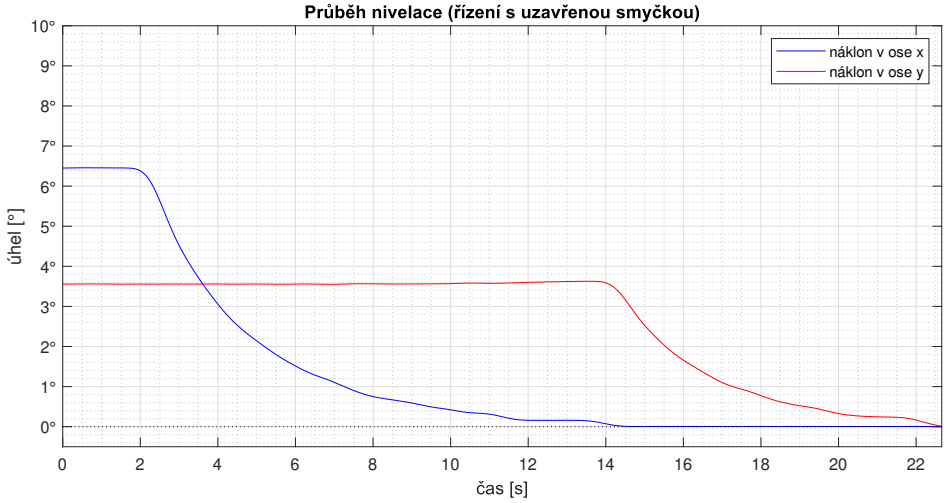
jiných druhů akčních členů a několika set kilových zátěží. V našem případě je autonivelace ukončena během několika vteřin. Zaměříme se pouze na porovnání doby nivelace při použití jednotlivých metod řízení.

6.3 Porovnání metod řízení

Při testování metod řízení byla zaznamenávána tato data: časový průběh náklonu, celkový čas nivelace a konečná hodnota náklonu. Při opakovaném testování nivelace byly konečné hodnoty náklonu zprůměrovány, a tím bylo zjištěno, s jakou přibližnou přesností se lze přiblížit k žádané (nulové) hodnotě náklonu. Dále byla ověřena opakovatelnost našeho systému, ta se týká těsnosti shody mezi výsledky při stejném pohybu.

V následujících podkapitolách jsou zobrazeny naměřené průběhy pro jednotlivé režimy řízení a za nimi následuje tabulka, která porovnává naměřená data. Pro lepší porovnání jsou zde uvedeny průběhy nivelace ze stejného počátečního stavu (přibližný náklon 6,4° v ose x a 3,6° v ose y).

6.3.1 Řízení s uzavřenou smyčkou

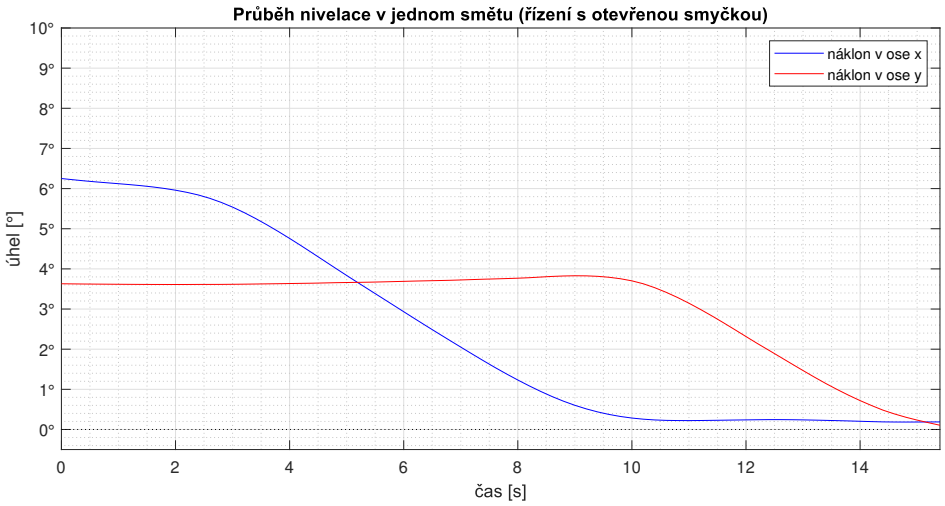


Obrázek 6.1: Průběh nivelace (uzavřená smyčka)

Průběh nivelace je exponenciální a je závislý na regulačních konstantách použitého PID regulátoru. V našem případě byly konstanty zvoleny metodou pokus omyl. V tomto případě jsou konstanty nastaveny následovně: P=5, I=1, D=0,5. Při nastavení regulátoru, bylo třeba dbát na to, aby vlivem velké proporcionální, nebo integrační konstanty nedošlo k překmitu a průběh se pomalu a plynule blížil k požadované nulové hodnotě náklonu. Přesnost řízení je velmi vysoká, řádově setiny stupně, rychlost je naopak velmi nízká.

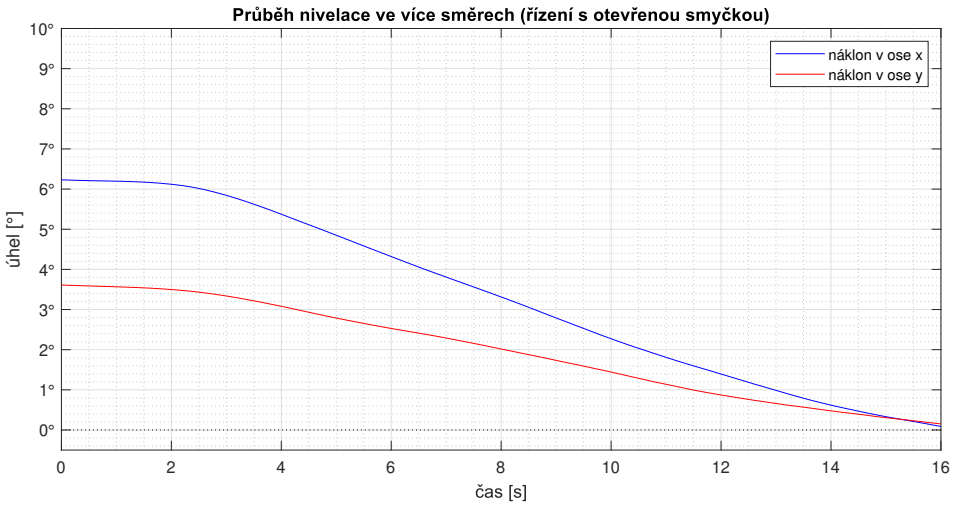
6.3.2 Řízení s otevřenou smyčkou

Nivelace v jednom směru



Obrázek 6.2: Průběh nivelace v jednom směru (otevřená smyčka)

Nivelace ve více směrech

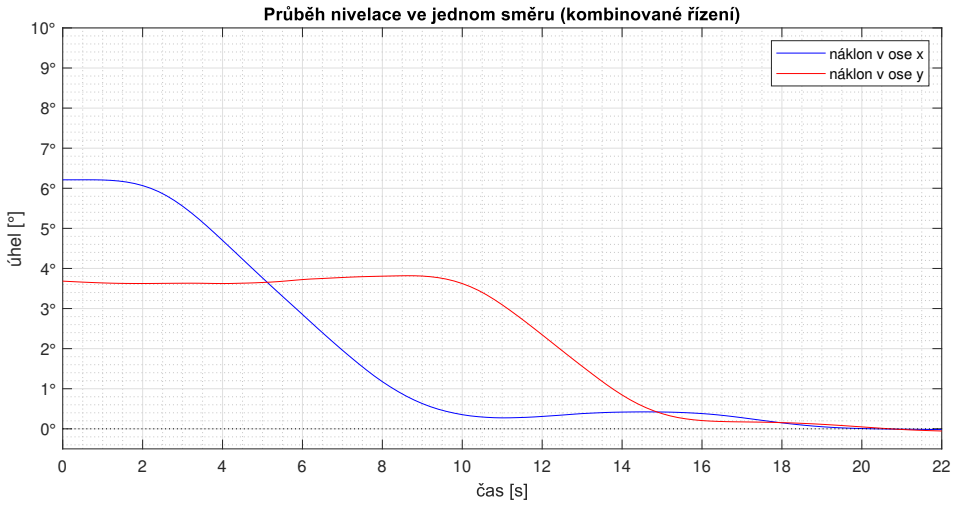


Obrázek 6.3: Průběh nivelace ve více směrech (otevřená smyčka)

V případě řízení s otevřenou smyčkou má velká část nivelačního procesu lineární průběh a oproti řízení s PID regulátorem nižší přesnost, v našem případě přibližně desetinu stupně. Rychlost je ovšem velmi vysoká.

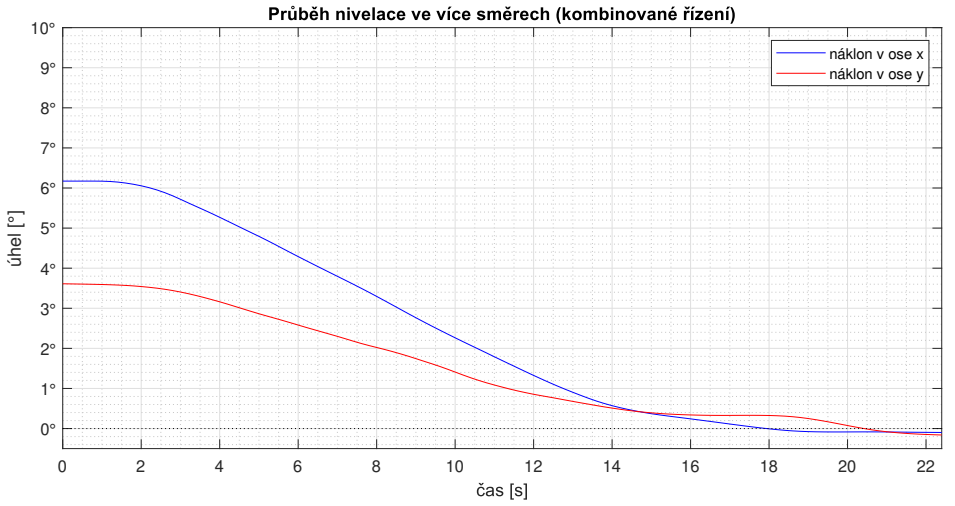
6.3.3 Kombinované řízení

Nivelace v jednom směru



Obrázek 6.4: Průběh nivelace v jednom směru (kombinované řízení)

Nivelace ve více směrech



Obrázek 6.5: Průběh nivelace ve více směrech (kombinované řízení)

Nejideálnější metodou se jeví kombinované řízení, kdy je v první fázi nivelace řízena na základě výpočtu, a po dosažení hrubého výsledku (přesnost v řádech desetin až jednoho stupně), je započata fáze řízení s uzavřenou smyčkou. Výsledná nivelace je dostatečně přesná i rychlá.

V případě otevřené smyčky a kombinovaného řízení byly také porovnány metody nivelace v jednom a ve více směrech najednou. Z naměřených dat vyplývá, že při nivelaci ve více osách najednou je přesnost nepatrně nižší. Výhodou je kratší čas nivelace, který je v případě otevřené smyčky znatelnější než při kombinovaném řízení.

Opakovatelnost byla vypočtena na základě deseti opakovaných procesů nivelace s počátečním náklonem 6,4° v ose x a 3,6° v ose y. Totéž platí pro čas nivelace, jedná se o střední hodnotu deseti opakovaných měření.

Tabulka 6.1: Porovnání metod řízení

Režim řízení	**Přesnost**[a] [°]		**Opakovatelnost** [°]		**Čas** [s]
	osa x	**osa y**	**osa x**	**osa y**	
Uzavřená smyčka (PID regulace)	0,01	0,02	0,02	0,02	20,61
Otevřená smyčka (nivelace v jednom směru)	0,10	0,12	0,05	0,04	14,03
Otevřená smyčka (nivelace ve více směrech)	0,11	0,14	0,08	0,06	12,17
Kombinované řízení (nivelace v jednom směru)	0,02	0,03	0,03	0,03	19,88
Kombinované řízení (nivelace ve více směrech)	0,03	0,03	0,02	0,03	19,07

Pozn.[a] správnost – těsnost shody střední hodnoty výsledků a referenční hodnoty (ČSN ISO 5725)

7 Možnosti připojení přídavných snímačů

Pro nenáročné aplikace je použití pouze jednoho senzoru dostačující. Ovšem lze nalézt případy, kdy je žádané využít vícero různých senzorů pro co nejlepší průběh, přesnost a rychlost nivelace. To platí především v průmyslových aplikacích.

7.1 Snímače

7.1.1 Inerciální senzory

Jedná se o senzory sloužící pro měření například: náklonu, polohy, rychlosti a zrychlení.

Pro relativně pomalé a rovnoměrné změny náklonu jsou dostačující statické senzory náklonu, důsledkem je pomalý průběh nivelace. Při vysokých rychlostech je (pro dosažení spolehlivých hodnot) zapotřebí potlačit poruchy způsobené vibracemi, otřesy nebo zrychlením. Dosažení spolehlivých hodnot lze docílit použitím dynamických senzorů náklonu, nebo využitím přídavných inerciálních senzorů.

Vhodné jsou například kombinace akcelerometrů a gyroskopů pro zajištění stabilní datové základny. Se znalostí zrychlení v jednotlivých osách lze upravit průběh a rychlost nivelace, a tím i výslednou přesnost.

7.1.2 Snímače zatížení

V oblasti autonivelace jsou často využívány snímače zatížení. Například u jeřábní techniky je žádané znát zatížení jednotlivých stabilizačních ramen. Při manipulaci s těžkými břemeny je zátěž jednotlivých ramen proměnlivá a je neustále nerovnoměrná, což může snížit pracovní bezpečnost (například překlopením nivelovaného zařízení). Se znalostí zátěže lze měnit rozpoložení jednotlivých ramen, nebo upozornit obsluhu na nebezpečnou polohu těžiště celého zařízení.

Další možnost využití snímačů je ošetření problému s takzvaným virtuálním ramenem. Jedná se o problém, který by bylo vhodné ošetřit i u naší testovací platformy. Při nulovém, nebo téměř nulovém, zatížení konkrétního ramene lze předpokládat, že zmíněné rameno visí ve vzduchu, a je třeba změnit jeho polohu.

7.2 Komunikační rozhraní

Různé senzory používají také různá komunikační rozhraní, která mají své výhody i nevýhody. Dostupná rozhraní jsou důležitým nástrojem pro čtení, zápis a vyhodnocení dat různého informačního charakteru. Pro průmyslové využití se nejčastěji jedná (vedle klasického analogového rozhraní) o digitální sériové sběrnice, které slouží pro síťovou komunikaci.

V automobilovém a těžkém průmyslu jsou hojně využívány rozhraní z kategorie Device bus, do které patří zmiňované CANopen rozhraní a dále například Interbus a DeviceNet. Jejich výhodou je, že při velkém množství zařízení, které je nutné vzájemně komunikačně propojit, není potřeba velký počet datových vodičů. Pro inerciální snímače je to nejčastěji právě rozhraní CANopen. Dále například německá firma Balluff [38] dodávající různé průmyslové snímače, včetně inklinometrů, u svých výrobků využívá rozhraní Modbus. Jedná se o otevřený protokol pro komunikaci velkého počtu zařízení a umožňuje přenášet data po různých sítích i sběrnicích. Komunikace funguje na principu předávání datových zpráv mezi klientem a serverem, stejně jako u rozhraní CANopen.

Závěr

Úvodní část bakalářské práce byla věnována rešerši různých autonivelačních systémů, přičemž konkrétněji se zaměřila na nivelační systémy stavebních strojů podrobněji pak na vybrané konstrukční 3D tiskárny.

V rámci praktické části byl na základě rešerše vytvořen vlastní autonivelační systém, který je přizpůsoben pro testování na příslušné nivelační platformě, jež byla součástí mého ročníkového projektu. Systém je vytvořen v programovém prostředí Matlab dvěma způsoby: blokovým Simulink modelem pro testování řízení pomocí PID regulátoru a Matlab skriptem, jehož součástí je jednoduché grafické uživatelské rozhraní a možnost testování nivelace vícero způsoby.

Vytvořený systém byl implementován do vývojové platformy Arduino Uno. Pro testování byly poskytnuty dva inklinometry s různými specifikacemi, přičemž inklinometr s komunikačním rozhraním CANopen se nepodařilo plně zprovoznit. Komunikace se snímačem byla realizována za pomocí modulu rozšiřujícího základní desku Arduino Uno o sběrnici CAN. Po konfiguraci snímače nebylo možné získat výstupní data, nejspíše z důvodu nepodařeného uvedení snímače do provozního stavu. K řízení systému byl tedy využit pouze inklinometr s analogovým rozhraním.

Nivelační platformu je možné řídit pomocí systému ve třech různých režimech: s uzavřenou smyčkou, otevřenou smyčkou, kombinované řízení. Tyto režimy jsou v poslední části práce otestovány a porovnány, pro přehled jsou zobrazeny průběhy jednotlivých procesů nivelace.

Možnost rozšíření tématu bakalářské práce spatřuji ve využití dalších přídavných senzorů, které jsou taktéž zmíněny a jejich použití je diskutováno v poslední části práce. Dále je vhodné otestovat různé koncepce stabilizačních ramen a akčních členů.

Použitá literatura

[1] *Kapalinové snímače úhlu a náklonu – ALPHA international* [online]. ALPHA international s.r.o., Copyright © 2020 [cit. 2021-05-05]. Dostupné z: http://www.alphaint.cz/snimace-uhlu-a-naklonu/kapalinove/.

[2] *Snímače úhlu a náklonu s gyroskopem – ALPHA international* [online]. ALPHA international s.r.o., Copyright © 2020 [cit. 2021-05-05]. Dostupné z: http://www.alphaint.cz/snimace-uhlu-a-naklonu/kapalinove/.

[3] *Liebherr LTM 1030-2 - Jeřábové služby* [MALINA - VRŠE] [online]. Praha, Copyright © 2021 [cit. 2021-05-08]. Dostupné z: https://www.jerabovesluzby.cz/jeraby/liebherr-ltm-1030-2/.

[4] *Crane (machine)* [online]. San Francisco (CA): Wikimedia Foundation, 2001--2021 [cit. 2021-03-24]. Dostupné z: https://en.wikipedia.org/wiki/Crane_%5C%28machine%5C%29.

[5] *Home - Reedyk* [online]. Klaaswaal, Copyright © 2019 [cit. 2021-03-24]. Dostupné z: https://www.reedyk.eu/.

[6] JAREŠ, Ladislav. *Kompaktní jeřáb REEDYK C 3410 – výkonný stroj pro profesionály*. 2011. Dostupné také z: https://www.stavebni-technika.cz/clanky/kompaktni-jerab-reedyk-c-3410-vykonny-stroj-pro-profesionaly.

[7] *Hydropneumatic suspension* [online]. San Francisco (CA): Wikimedia Foundation, 2001-2020 [cit. 2020-12-27]. Dostupné z: https://en.wikipedia.org/wiki/Hydropneumatic_suspension.

[8] *Height adjustable suspension* [online]. San Francisco (CA): Wikimedia Foundation, 2001-2020 [cit. 2021-03-24]. Dostupné z: https://en.wikipedia.org/wiki/Height_adjustable_suspension%5C#cite_note-AutoBlog.com-4.

[9] *The levelC caravan levelling system* [online]. Impressum, Copyright © 2021 [cit. 2021-02-04]. Dostupné z: https://www.ep-hydraulics.nl/en/caravan/.

[10] *Glide Rite* [online]. Hampshire, Copyright © 2021 [cit. 2021-03-24]. Dostupné z: https://glide-rite.com/.

[11] *Lippert: Automatic Leveling* [online]. California, Copyright © 2021 [cit. 2021-03-24]. Dostupné z: https://www.lci1.com/automatic-leveling.

[12] *Automatic Vehicle Leveling System* [online]. KuarkTek Digital, Copyright © 2019 [cit. 2021-02-03]. Dostupné z: https://www.gesmuhendislik.com/works/automatic-vehicle-leveling-system/.

[13] *Haacon: Competence in lifting technology* [online]. Germany, Copyright © 2018 [cit. 2021-03-24]. Dostupné z: https://www.haacon.com/en/home.php.

[14] *Haacon: Container and Shelter Lifting and Transport Systems* [online]. Verdict Media Limited, Copyright © 2021 [cit. 2021-02-03]. Dostupné z: https://www.army-technology.com/contractors/logistics/haacon/.

[15] *3D concrete printers: Mobile 3D Printer* [online]. Netherlands, 2020 [cit. 2020-12-26]. Dostupné z: https://cybe.eu/technology/3d-printers/.

[16] *Concrete 3D printing workshop I Apis Cor: 3D Printed Homes* [online]. Russia, Copyright © 2023 by Apis Cor [cit. 2020-12-26]. Dostupné z: https://www.apis-cor.com/concrete-3d-printing-workshop.

[17] *Apis-Cor 3D printer builds houses of nearly any size* [online]. Vienna, 2018 [cit. 2020-12-26]. Dostupné z: http://additivemanufacturing.global/index.php/en/print-en/architecture/3592-apis-cor-3d-printer-builds-houses-of-nearly-any-size.

[18] *Hadrian X: Outdoor Construction & Bricklaying Robot from FBR* [online]. Australia, Copyright © 2018 FBR Ltd [cit. 2020-12-26]. Dostupné z: https://www.fbr.com.au/view/hadrian-x.

[19] *ETH Zurich: Homepage* [online]. Zurich, Copyright © 2021 [cit. 2021-03-24]. Dostupné z: https://ethz.ch/en.html.

[20] *IN SITU FABRICATOR* [online]. Zurich: NCCR Digital Fabrication, Copyright © 2020 [cit. 2020-12-26]. Dostupné z: https://dfabhouse.ch/in_situ_fabricator/.

[21] *Constructions-3D: Technical Specifications* [online]. France [cit. 2020-12-26]. Dostupné z: https://en.constructions-3d.com/documentation-technique.

[22] KEATING, Steven. *Digital Construction Platform* [online]. Massachusetts: MIT Media Lab, 2016 [cit. 2020-12-26]. Dostupné z: https://www.media.mit.edu/projects/digital-construction-platform-v-2/overview/.

[23] *ConPrint3D — Endowed Chair of Construction Machines* [online]. Dresden: TU Dresden, Copyright © 2021 [cit. 2021-03-25]. Dostupné z: https://tu-dresden.de/ing/maschinenwesen/imd/bm/forschung/forschungsprojekte-1/conprint3d?set_language=en.

[24] O'NEAL, Bridget. *TU Dresden: CONPrint3D for Monolithic 3D Printing in Construction: The Voice of 3D Printing* [online]. Additive Manufacturing, 2019 [cit. 2020-12-26]. Dostupné z: https://3dprint.com/261167/tu-dresden-conprint3d-for-monolithic-3d-printing-in-construction/.

[25] *Genesis Dimensions: Automating Construction* [online]. Copyright © 2016 [cit. 2020-12-26]. Dostupné z: https://genesisdimensions.com/.

[26] HORNÍČEK, Tomáš. *Hydraulické a pneumatické pohony ve strojírenství* [MMspektrum] [online]. 2014 [cit. 2020-11-30]. Dostupné z: https://www.mmspektrum.com/clanek/hydraulicke-a-pneumaticke-pohony-ve-strojirenstvi.html.

[27] FILIP, Petr. *Porovnání pneumatických, hydraulických a elektrických lineárních pohonů* [online]. Praha 10, Copyright © 2013 - 2017 [cit. 2020-12-26]. Dostupné z: https://e-konstrukter.cz/novinka/porovnani-pneumatickych-hydraulickych-a-elektrickych-linearnich-pohonu.

[28] CAO, Jian; ZHU, Xiaocong; YAO, Bin; TAO, Guoliang; WANG, Qingfeng. Levelling Control of an Electro-pneumatic Stabilized Platform Driven by Pneumatic Muscles. *IFAC Proceedings Volumes*. 2013, roč. 46, č. 5, s. 445–452. ISSN 1474-6670. Dostupné z DOI: https://doi.org/10.3182/20130410-3-CN-2034.00083. 6th IFAC Symposium on Mechatronic Systems.

[29] BOHÁČ, Radek; DOLENSKÝ, Roman. *Autonivelační systém podvozku pracovního stroje*. Liberec, 2020. Ročníkový projekt. Fakulta mechatroniky, informatiky a mezioborových studií Technické univerzity v Liberci.

[30] *Servo MG996R* [GM electronic, spol. s.r.o.]. Copyright © 2019. Dostupné také z: https://www.gme.cz/data/attachments/dsh.772-293.1.pdf.

[31] *Arduino Products* [online]. USA, Copyright © 2021 [cit. 2021-05-18]. Dostupné z: https://www.arduino.cc/en/Main/Products.

[32] *Arduino Uno* [GM electronic, spol. s.r.o.]. Copyright © 2020. Dostupné také z: https://www.gme.cz/data/attachments/dsh.772-237.1.pdf.

[33] *MCP2515 CAN BUS TJA1050 RECEIVER SPI PROTOCOL MODULE* [online]. Portugal, Copyright © 2021 [cit. 2021-05-21]. Dostupné z: https://www.botnroll.com/en/others/3992-mcp2515-can-bus-tja1050-receiver-spi-protocol-module.html.

[34] *Sick - Sensor Intelligence* [online]. Waldkirch: SICK AG, Copyright © 2021 [cit. 2021-05-06]. Dostupné z: https://www.sick.com/cz/cs/.

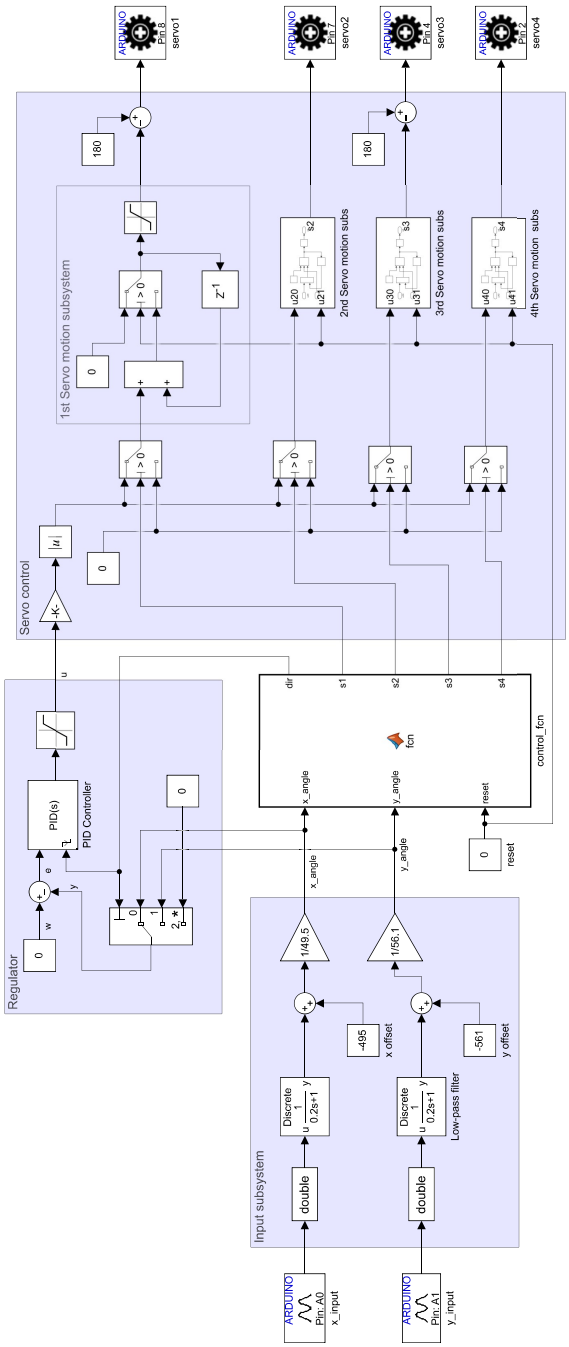
[35] *TMM55E-POH010* [SICK AG]. Waldkirch, Copyright © 2020. Dostupné také z: https://cdn.sick.com/media/pdf/9/69/369/dataSheet_TMM55E-POH010_1073783_cs.pdf.

[36] *TMM88D-PCI090* [SICK AG]. Waldkirch, Copyright © 2020. Dostupné také z: https://cdn.sick.com/media/pdf/6/46/946/dataSheet_TMM88D-PCI090_1094485_en.pdf.

[37] *Operating instructions tms tmm61 tms tmm88 tms tmm88* [online]. Waldkirch: SICK AG, 2019 [cit. 2021-05-21]. Dostupné z: https://cdn.sick.com/media/docs/3/53/853/operating_instructions_tms_tmm61_tms_tmm88_tms_tmm88_dynamic_canopen_inclination_sensors_en_im0064853.pdf.

[38] *Balluff CZ - Home* [online]. Germany: Balluff CZ s.r.o., Copyright © 2021 [cit. 2021-05-05]. Dostupné z: https://www.balluff.com/local/cz/home/.

A Simulink model



B Matlab kód

NivelationMain.m

```
clear all; clc;

exit = 0;
phase = 0;
direct_mode = 1; % 1 - one direct, 0 - more direction mode
soft_niv = 0;
active_niv = 0;

global tStart;

tEnd = 0; % tim data var for leveling time

% constants
vel = 100; % velocity (1-200)
st1 = 0.02; % servo step
st2 = 0.01; %

lmax = 54; % leg length (mm)
dimX = 237; % platform x dimension (mm)
dimY = 223; % platform y dimension (mm)

x_const = 2.5024; % inclinometer calibration constants
y_const = 2.6931;

% basic legs positions
zero_position = [0, 0, 0, 0];
middle_position = [0.5, 0.5, 0.5, 0.5];
max_position = [1, 1, 1, 1];
position_1 = [0, 0, 0, 0];

% variables
pos = 0;
[pos1, pos2, pos3, pos4] = deal(0);
x_angle = 0; % x-axis angle
y_angle = 0; % y-axis angle

% graph data
x = [];
y = [];
t = [];

% Arduino initialization
global a;
if ~isempty(instrfind)
    fclose(instrfinf);
    delete(instrfind);
end
a = arduino(); % Arduino object

% servos initialization
servo = [servo(a, 'D8'), servo(a, 'D7'), servo(a, 'D4'), servo(a, 'D2')];
%        1st (blue)      2nd (green)     3rd (yellow)    4th (orange)
```

```
%=========================================================================
% main loop
while ~exit

 [x_angle, y_angle] = ReadAngles(a, x_const, y_const);

 % show GUI
 AppGUI;
 uiwait(AppGUI);

 % exit test
 if exit
    break;
 end

 %-------------------------------------------------------------------------
 % move to zero postition
 if phase == 1
    writePosition(servo(1), 1);
    writePosition(servo(2), 0);
    writePosition(servo(3), 1);
    writePosition(servo(4), 0);

    active_niv = 1;
    phase = 0;

 %-------------------------------------------------------------------------
 % start leveling
 elseif phase == 2
    tStart = tic; % timer for Final Graph data
    GetData();
    pause(1);
    [pos1, pos2, pos3, pos4] = deal(0);

    %---------------------------------------------------------
    % start main nivelation

    [x_angle, y_angle] = ReadAngles(a, x_const, y_const);
    pos = dimX*sin(abs(x_angle)/180*pi)/lmax;

    if x_angle < 0
        pos1 = pos;
        pos2 = pos;
    else
        pos3 = pos;
        pos4 = pos;
    end

    % set one direct mod
    if direct_mode == 1
        position_1 = [pos1, pos2, pos3, pos4];
        SetPosition(servo, position_1, st1, vel);

        [x_angle, y_angle] = ReadAngles(a, x_const, y_const);
    end

    pos = dimY*sin(abs(y_angle)/180*pi)/lmax;

    if y_angle < 0
        pos2 = pos2+pos;
        pos3 = pos3+pos;
    else
        pos1 = pos1+pos;
        pos4 = pos4+pos;
    end
    position_1 = [pos1, pos2, pos3, pos4];
    SetPosition(servo, position_1, st1, vel);

```

```
121     %-------------------------------------------------------------
122     % start soft nivelation
123     if soft_niv
124         % X-axis
125         [x_angle, y_angle] = ReadAngles(a, x_const, y_const);
126         if x_angle < 0
127             while x_angle < -0.05
128                 step = [st2, st2, 0, 0];
129                 ServoStep(servo, step);
130                 [x_angle, y_angle] = ReadAngles(a, x_const, y_const);
131             end
132         else
133             while x_angle > 0.05
134                 step = [0, 0, st2, st2];
135                 ServoStep(servo, step);
136                 [x_angle, y_angle] = ReadAngles(a, x_const, y_const);
137             end
138         end
139         % Y-axis
140         if y_angle < 0
141             while y_angle < -0.05
142                 step = [0, st2, st2, 0];
143                 ServoStep(servo, step);
144                 [x_angle, y_angle] = ReadAngles(a, x_const, y_const);
145             end
146         else
147             while y_angle > 0.05
148                 step = [st2, 0, 0, st2];
149                 ServoStep(servo, step);
150                 [x_angle, y_angle] = ReadAngles(a, x_const, y_const);
151             end
152         end
153     end
154     %-------------------------------------------------------------
155     GetData();
156     pause(1);
157     GetData();
158
159     [tEnd, t, x, y] = MakeGraph(x_const, y_const);
160
161     active_niv = 0;
162     phase = 0;
163  end
164 end
```

SetPosition.m

```
function SetPosition(servo, end_pos, st, vel)
    % save actual position
    start_pos = [1-readPosition(servo(1)), readPosition(servo(2)),...
                 1-readPosition(servo(3)), readPosition(servo(4))];

    % position verif
    if isequal(start_pos, end_pos)
        return;
    end

    % number of steps calc
    path = end_pos - start_pos;
    max_path = max(abs(path));
    n_steps = max_path/st;
    step = path/n_steps;

    % direction of individ servos
    dir = sign(path);

    % loop for servos motion
    for i = 0:1:n_steps
        actual_pos = [1-readPosition(servo(1)), readPosition(servo(2)),...
                      1-readPosition(servo(3)), readPosition(servo(4))];

        next_pos = [1-start_pos(1)-i*dir(1)*step(1), start_pos(2)+i*dir(2)*step(2),...
                    1-start_pos(3)-i*dir(3)*step(3), start_pos(4)+i*dir(4)*step(4)];

        for j = 1:4
            if dir(j)*actual_pos(j) < dir(j)*end_pos(j)
                if next_pos(j) < st
                    next_pos(j) = 0;
                end
                if next_pos(j) > 1
                   next_pos(j) = 1;
                end
                writePosition(servo(j), next_pos(j));
            end
        end

        pause(1/vel);
        GetData();
    end
end
```

ServoStep.m

```
function ServoStep(servo, step)
    actual_pos = [readPosition(servo(1)), readPosition(servo(2)),...
                  readPosition(servo(3)), readPosition(servo(4))];

    next_pos = [actual_pos(1)-step(1), actual_pos(2)+step(2),...
                actual_pos(3)-step(3), actual_pos(4)+step(4)];

    for i = 1:4
        if next_pos(i) < step
           next_pos(i) = 0;
        elseif next_pos(i) > 1
           next_pos(i) = 1;
        end
        writePosition(servo(i), next_pos(i));
    end
    GetData();
end
```

ReadAngles.m

```
function [x_angle, y_angle] = ReadAngles(a, x_const, y_const)
    n = 10; % number of measurements
    r = 10; % sensor range

    x_voltage = zeros(n, 1);
    y_voltage = zeros(n, 1);

    % save n measurements
    for i = 1:n
        x_voltage(i) = readVoltage(a,'A0'); % reading analog inputs
        y_voltage(i) = readVoltage(a,'A1');
    end

    % mean of n measurements
    x_mean = mean(x_voltage);
    y_mean = mean(y_voltage);

    % voltage to angle
    x_angle = r*(x_mean/x_const-1);
    y_angle = r*(y_mean/y_const-1);
end
```

MakeGraph.m

```
function [tEnd, t, x, y] = MakeGraph(x_const, y_const)
    global x_data;
    global y_data;
    global time_data;

    if isempty(time_data)
       return
    end

    tEnd = time_data(end)-2;
    x0 = 10*(x_data/x_const-1);
    y0 = 10*(y_data/y_const-1);

    t = 0:0.2:time_data(end);

    if length(t) < 27
        return
    end

    x1 = interp1(time_data,x0,t,'linear');
    y1 = interp1(time_data,y0,t,'linear');

    w=10;
    a = 1;
    b = (1/w)*ones(1,w);
    x = filtfilt(b,a,x1);
    y = filtfilt(b,a,y1);
end
```

AppGUI.m

```
function varargout = AppGUI(varargin)
% Begin initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @AppGUI_OpeningFcn, ...
                   'gui_OutputFcn',  @AppGUI_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
%==========================================================================
% End initialization code - DO NOT EDIT

% --- Executes just before AppGUI is made visible.
function AppGUI_OpeningFcn(hObject, eventdata, handles, varargin)

% get data from workspace
dimX = evalin('base', 'dimX');
dimY = evalin('base', 'dimY');
lmax = evalin('base', 'lmax');
tEnd = evalin('base', 'tEnd');
direct_mode = evalin('base', 'direct_mode');
soft_niv = evalin('base', 'soft_niv');
x = evalin('base', 'x');
y = evalin('base', 'y');
t = evalin('base', 't');

% set data for GUI panels
set(handles.x_diameter, 'string', dimX);
set(handles.y_diameter, 'string', dimY);
set(handles.leg_length, 'string', lmax);
set(handles.text3, 'string', sprintf('%.2f s', tEnd));

if direct_mode
    set(handles.radiobutton1, 'value', 1);
else
    set(handles.radiobutton2, 'value', 1);
end

if soft_niv
    set(handles.soft_leveling_checkbox, 'value', 1);
end

a = evalin('base', 'a');
x_const = evalin('base', 'x_const');
y_const = evalin('base', 'y_const');

[x, y] = ReadAngles(a, x_const, y_const);

set(handles.text1, 'string', sprintf('%.2f°', x));
set(handles.text2, 'string', sprintf('%.2f°', y));

activ_niv = evalin('base', 'active_niv');
if activ_niv
    set(handles.pushbuttonStartLeveling,'Enable','on');
else
    set(handles.pushbuttonStartLeveling,'Enable','off');
end
```

```
67 %-------------------------------------------------------------------
68 % create graph
69 if ~isempty(t)
70     axes(handles.graph)
71     plot(t, x, 'b', t, y, 'r', [0 t(end)], [0 0], ':k');
72
73     xlim([0 t(end)]);
74     ylim([-10 10]);
75     yticks(-10:2:10);
76     ytickformat('%.0f°')
77
78     title('Leveling process');
79     legend('x-axis angle','y-axis angle');
80     xlabel('time [s]');
81
82     grid on;
83     grid minor;
84 end
85
86 % Choose default command line output for AppGUI
87 handles.output = hObject;
88
89 % Update handles structure
90 guidata(hObject, handles);
91
92 %==========================================================================
93 % --- Outputs from this function are returned to the command line.
94 function varargout = AppGUI_OutputFcn(hObject, eventdata, handles)
95
96 varargout{1} = handles.output;
97
98
99 % --- Executes on button press in pushbuttonSetZeroPoint.
100 function pushbuttonSetZeroPoint_Callback(hObject, eventdata, handles)
101     assignin('base', 'phase', 1);
102     delete(handles.figureAppGUI);
103
104
105 % --- Executes on button press in pushbuttonStartLeveling.
106 function pushbuttonStartLeveling_Callback(hObject, eventdata, handles)
107     assignin('base', 'phase', 2);
108
109     if handles.soft_leveling_checkbox.Value
110         assignin('base', 'soft_niv', 1);
111     else
112         assignin('base', 'soft_niv', 0);
113     end
114
115     delete(handles.figureAppGUI);
116
117
118 % --- Executes on button press in pushbuttonExit.
119 function pushbuttonExit_Callback(hObject, eventdata, handles)
120     assignin('base', 'exit', 1);
121     delete(handles.figureAppGUI);
122
123
124 % --- Executes on button press in refresh.
125 function refresh_Callback(hObject, eventdata, handles)
126 a = evalin('base', 'a');
127 x_const = evalin('base', 'x_const');
128 y_const = evalin('base', 'y_const');
129
130 [x, y] = ReadAngles(a, x_const, y_const);
131
132 set(handles.text1, 'string', sprintf('%.2f°', x));
133 set(handles.text2, 'string', sprintf('%.2f°', y));
134
```

```
135 function x_diameter_Callback(hObject, eventdata, handles)
136
137 % --- Executes during object creation, after setting all properties.
138 function x_diameter_CreateFcn(hObject, eventdata, handles)
139
140 if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
        defaultUicontrolBackgroundColor'))
141     set(hObject,'BackgroundColor','white');
142 end
143
144
145 function y_diameter_Callback(hObject, eventdata, handles)
146
147
148 % --- Executes during object creation, after setting all properties.
149 function y_diameter_CreateFcn(hObject, eventdata, handles)
150
151 if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
        defaultUicontrolBackgroundColor'))
152     set(hObject,'BackgroundColor','white');
153 end
154
155
156 function leg_length_Callback(hObject, eventdata, handles)
157
158
159 % --- Executes during object creation, after setting all properties.
160 function leg_length_CreateFcn(hObject, eventdata, handles)
161
162 if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
        defaultUicontrolBackgroundColor'))
163     set(hObject,'BackgroundColor','white');
164 end
165
166
167 % --- Executes on button press in set_x_diameter.
168 function set_x_diameter_Callback(hObject, eventdata, handles)
169 dimX = str2double(get(handles.x_diameter, 'string'));
170 assignin('base', 'dimX', dimX);
171
172
173 % --- Executes on button press in set_y_diameter.
174 function set_y_diameter_Callback(hObject, eventdata, handles)
175 dimY = str2double(get(handles.y_diameter, 'string'));
176 assignin('base', 'dimY', dimY);
177
178
179 % --- Executes on button press in set_leg_length.
180 function set_leg_length_Callback(hObject, eventdata, handles)
181 lmax = str2double(get(handles.leg_length, 'string'));
182 assignin('base', 'lmax', lmax);
183
184
185 % --- Executes on button press in soft_leveling_checkbox.
186 function soft_leveling_checkbox_Callback(hObject, eventdata, handles)
187
188
189 % --- Executes on button press in radiobutton1.
190 function radiobutton1_Callback(hObject, eventdata, handles)
191 assignin('base', 'direct_mode', 1);
192
193
194 % --- Executes on button press in radiobutton2.
195 function radiobutton2_Callback(hObject, eventdata, handles)
196 assignin('base', 'direct_mode', 0);
197
198
199 % --- Executes on button press in set_as_zero.
```

```
200 function set_as_zero_Callback(hObject, eventdata, handles)
201 a = evalin('base', 'a');
202 x_const = readVoltage(a,'A0');
203 y_const = readVoltage(a,'A1');
204
205 assignin('base', 'x_const', x_const);
206 assignin('base', 'y_const', y_const);
207
208 [x, y] = ReadAngles(a, x_const, y_const);
209
210 set(handles.text1, 'string', sprintf('%.2f°', x));
211 set(handles.text2, 'string', sprintf('%.2f°', y));
```