



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

MANIPULÁTOR PRO SKLENÍKOVÝ SYSTÉM

GREENHOUSE MANIPULATOR

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Martin Sukdolák

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Tomáš Beneš

BRNO 2023

Bakalářská práce

bakalářský studijní program **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

Student: Martin Sukdolák

ID: 230176

Ročník: 3

Akademický rok: 2022/23

NÁZEV TÉMATU:

Manipulátor pro skleníkový systém

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je navrhnout manipulátor s možností automatizace operací ve skleníku.

1. Seznamte se s komerčními řešeními a porovnejte jejich výhody a nevýhody.
2. Navrhněte mechanickou část vlastního modulárního řešení.
3. Navrhněte elektroniku pro vaše řešení.
4. Implementujte a otestujte Vámi vytvořený manipulátor.
5. Zhodnoťte dosažené výsledky a zdokumentujte řešení.

DOPORUČENÁ LITERATURA:

JAZAR, Reza N., c2010. Theory of applied robotics: kinematics, dynamics, and control. 2nd ed. New York: Springer. ISBN 14-419-1749-7.

Termín zadání: 6.2.2023

Termín odevzdání: 22.5.2023

Vedoucí práce: Ing. Tomáš Beneš

doc. Ing. Václav Jirsík, CSc.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato bakalářská práce se věnuje automatizaci procesu ve skleníku s využitím manipulátoru typu SCARA za účelem zefektivnění práce ve skleníku se snahou snížit spotřebu vody. Práce rozebírá návrh takového systému a výrobu SCARA manipulátoru pro tento systém.

KLÍČOVÁ SLOVA

Manipulátor, SCARA, Skleník, Raspberry Pi, Codesys, Siemens NX, 3D model, automatické zalévání

ABSTRACT

This bachelor thesis deals with the automation of the process in the greenhouse using a SCARA manipulator in order to make the work in the greenhouse more efficient and to reduce water consumption. The thesis discusses the design of this system and the manufacturing of the SCARA manipulator for this system.

KEYWORDS

Manipulator, SCARA, Greenhouse, Raspberry Pi, Codesys, Siemens NX, 3D model, automatic watering

SUKDOLÁK, Martin. *Manipulátor pro skleníkový systém*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2023, 57 s. Bakalářská práce. Vedoucí práce: Ing. Tomáš Beneš

Prohlášení autora o původnosti díla

Jméno a příjmení autora: Martin Sukdolák
VUT ID autora: 230176
Typ práce: Bakalářská práce
Akademický rok: 2022/23
Téma závěrečné práce: Manipulátor pro skleníkový systém

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Tomáš Beneš za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

| | |
|---------------------------------------|-----------|
| Úvod | 11 |
| 1 Chytré farmaření | 12 |
| 1.1 Internet věci | 12 |
| 1.2 Drony | 12 |
| 1.3 Zemědělstí roboti | 13 |
| 1.3.1 Farmbot | 13 |
| 1.3.2 Root-Ai | 13 |
| 1.3.3 Certhon Harvest Robot | 13 |
| 1.3.4 Srovnání | 14 |
| 2 Koncepce | 15 |
| 2.1 Koncepce manipulátoru | 15 |
| 3 Návrh mechanické části | 16 |
| 3.1 Kinematika | 16 |
| 3.2 CAD/CAM systémy | 17 |
| 3.2.1 Virtuální dvojče | 17 |
| 3.3 Siemens NX | 18 |
| 3.3.1 Model | 18 |
| 3.3.2 SCARA robot | 19 |
| 3.3.3 Základna | 20 |
| 3.3.4 Virtuální zprovoznění | 20 |
| 3.4 3D tisk | 22 |
| 4 Elektronika | 23 |
| 4.1 Řídící člen | 23 |
| 4.1.1 Raspberry Pi 4 model B | 24 |
| 4.1.2 SPI a I2C komunikační protokoly | 24 |
| 4.2 Plošný spoj | 25 |
| 4.2.1 DRV8825 | 25 |
| 4.2.2 H-můstek | 27 |
| 4.3 Senzory | 27 |
| 4.3.1 AD převodník | 27 |
| 4.3.2 Koncové spínače | 27 |
| 4.3.3 Senzor vzdálenosti | 28 |
| 4.3.4 Senzor vlhkosti | 28 |
| 4.4 Akční člen | 28 |

| | | |
|----------|--------------------------------------|-----------|
| 4.4.1 | Krokové motory | 29 |
| 4.4.2 | Peristaltické čerpadlo | 29 |
| 5 | Softwarové vybavení | 30 |
| 5.1 | Soft PLC | 30 |
| 5.2 | HMI | 30 |
| 5.3 | PLC-program | 32 |
| 5.3.1 | Komunikace | 33 |
| 5.3.2 | Softwarová PWM | 33 |
| 5.3.3 | Pozicování | 34 |
| 5.3.4 | Automatický mód | 34 |
| 5.3.5 | Funkce měření vlhkosti | 35 |
| 6 | Implementace | 36 |
| 6.1 | Sestavování manipulátoru | 36 |
| 6.2 | Elektronika | 36 |
| 6.2.1 | Osazování plošného spoje | 37 |
| 6.2.2 | Modul DRV8825 | 38 |
| 6.2.3 | Vstupy/výstupy Raspberry Pi | 38 |
| 6.3 | Implementace programu | 39 |
| 7 | Plány do budoucna | 40 |
| 7.1 | Strojové vidění | 40 |
| 7.2 | Hlava manipulátoru | 40 |
| 7.3 | Konstrukce | 40 |
| 8 | Zhodnocení | 41 |
| 8.1 | Parametry | 41 |
| | Závěr | 42 |
| | Literatura | 43 |
| | Seznam symbolů a zkratk | 45 |
| | Seznam příloh | 46 |
| | A Schéma zapojení elektroniky | 47 |
| | B Návod | 49 |
| | C Obsah elektronické přílohy | 57 |

Seznam obrázků

| | |
|---|----|
| 3.1 Geometrie SCARA manipulátoru | 16 |
| 3.2 Výsledný 3D model | 19 |
| 3.3 Lineární vedení | 21 |
| 3.4 FDM tištěné díly | 22 |
| 4.1 Kapacitní snímače vlhkosti | 28 |
| 5.1 HMI-Verze dva-hlavní obrazovka | 31 |
| 5.2 HMI-Verze dva-druhá obrazovka | 31 |
| 5.3 Stavový automat PLC programu | 32 |
| 6.1 Sestava manipulátoru | 36 |
| 6.2 Plošný spoj | 37 |
| 6.3 Rozložení pinů | 38 |
| 6.4 Spojení Raspberry Pi a programu Codesys | 39 |
| 8.1 Celý systém | 41 |

Seznam tabulek

| | |
|--|----|
| 3.1 Tabulka součástek | 20 |
| 3.2 Signály MCD | 21 |
| 4.1 Seznam elektroniky | 23 |
| 4.2 Krokování motoru na základě přepínače | 26 |
| 4.3 Výstupy z expandéru a vstupy do DRV8825 modulu | 26 |

Úvod

S rostoucí populací roste i spotřeba potravin ve společnosti. Průměrný zdravý člověk potřebuje 2700 kcal za den. V dnešní době zemědělství spotřebuje 1 litr vody na vypěstování 1 kcal. V důsledku toho je 70 % vody z hydrogeologických kolektorů, řek a jezer použito v zemědělství k vypěstování potřebného množství potravin. [1] K snížení spotřeby vody na polích by mohlo pomoci přesunutí části výroby na zahrady rodinných domů nebo chat. Vzhledem k časové vytíženosti většiny lidí je tento přístup možný pouze v případě plné automatizace daného procesu. To by umožnilo minimalizovat spotřebu vody a mělo by mít za následek zvýšení kvality potravy, vzhledem k tomu, že by nebyly potřebné v takové míře různé pesticidy a herbicidy.

Tato práce se věnuje robotizaci pěstování rostlin. Hlavním cílem této práce bylo vytvořit robota, který bude nezávisle na člověku kontrolovat stav vlhkosti půdy a na základě předem nastavené hodnoty vlhkosti pro konkrétní rostlinu dávkovat množství vody do záhonu. Implementací tohoto robota do skleníku předpokládáme snížení spotřeby vody a zvýšení produkce potravin. Jako základ robota jsme si zvolili SCARA manipulátor s přídatnou osou pohybu, která mu umožňuje pohybovat se po skleníku. K tomuto manipulátoru bylo vytvořeno virtuální dvojčete za pomoci programu Siemens NX, na kterém byly otestovány hlavní problémy, které by mohly vzniknout při návrhu manipulátoru a programu. Celý systém jsme také zrealizovali.

V první části této práce prezentuji průmyslová řešení dané problematiky od firem FarmBot, Root-Ai a Certhon Harvest Robot. Následně se zabýváme CAD návrhem samotného manipulátoru a zprovozněním virtuálního dvojčete. Třetí kapitola se zabývá návrhem elektroniky, která se skládá z řídicí části, senzorů a akčních členů. Čtvrtá kapitola obsahuje softwarovou část manipulátoru, popisuje PLC program programovaný v prostředí Codesys a návrh HMI. Dále se práce věnuje celkové realizaci tohoto manipulátoru a všem krokům pro úspěšnou implementaci. Na závěr jsou naznačena další možná rozšíření, které je možné s manipulátorem realizovat a zhodnocení celého návrhu.

1 Chytré farmaření

S rostoucí populací přichází potřeba zvýšení produkce potravin. V důsledku toho, je potřebné se zaměřit na výzvy, které povedou ke zvýšení zemědělské produkce. Může to být například zlepšování kvality půdy, minimalizace spotřebované vody, ochrana proti škůdcům nebo odstraňování plevelů. [16]

První snahy o implementaci chytrého farmaření ("smart farming") byly již v letech 1970-1980. V dnešní době, se zlepšující se automatizací, přichází příležitost vyvíjet mnoho flexibilních zemědělských přístrojů, které zmenšují nežádoucí odpad na životní prostředí a zvyšují udržitelnost výroby potravin. Robotické platformy v terénu mohou také ze sebraných dat poskytnout do budoucnosti mnoho informací o půdě, semenech, plodinách, nákladech a využití vody. Hlavní využití chytrého farmaření je na opakujících se úkonech, kde automatictí roboti jsou efektivnější než lidé nebo velké stroje. Další možnost využití je kooperace lidské síly a robota. [5]

1.1 Internet věci

Pod tímto pojmem se skrývá velká škála aplikací. V zemědělství internet věci pomáhá hlavně ve smyslu sběru dat. Chytré systémy efektivně monitorují stav rostlin a data následně posílají na cloudová úložiště, kde se zpracují a mohou dát signál zemědělcům pro vykonání potřebných procesů. Podle "Food and Agriculture Organization (FAO 2017)" je přibližně 20-40% úrody zničeno v důsledku škůdců a pesticidů. Tomu by se dalo zabránit implementováním chytrých senzorů. Aby se toto řešení dalo použít je potřebné stabilní a rychlé připojení k internetu pro přenos velkého množství dat a následně umělá inteligence pro jejich zpracování. Všechny tyto věci by mohly vést k automatizaci následných procesů. [16]

1.2 Drony

Drony byly používány v omezené míře v zemědělství už od roku 1980. Po zlepšení komunikace se drony staly velice užitečným a používaným strojem. Začaly se používat například pro monitorování zdravých rostlin, sázení, postřiky nebo pro analýzu půdy. Drony jsou vybaveny mnoha senzory a kamerami, díky kterým dokážou odhalit škůdce, choroby, či sledovat výšku a počet rostlin, potřebu vody a mnoho dalších. Všechny tyto operace stojí na základním předpokladu real-time sbírání dat a jejich vyhodnocování. Hlavní nevýhoda použití dronů je jejich krátká letová výdrž. Většina dronů dokáže pracovat méně než hodinu a následně je potřebné je opětovně nabít. Další nevýhody jsou vysoká cena, nedostatečně vyřešená legislativa a nestálost při změně počasí. [16]

1.3 Zemědělní roboti

V aktuální době je velmi málo robotů, které jsou vyvinuty specificky pro zemědělství. Většina robotů je převzatých z jiných odvětví a upravena ke konkrétnímu účelu. Tyto roboty můžeme rozdělit do dvou hlavních skupin. Jsou to roboti, kteří dokážou vykonávat pouze specifický úkon, a roboti, kteří dokážou nést různé moduly a vykonávat tak množství různorodé práce. Hlavním problémem ale zůstává, že většina robotů zatím není dostatečně robustních k tomu, aby odolávaly reálným podmínkám jako je například odolnost proti prachu, dešti, mlze a vysokým/nízkým teplotám. Například, většina manipulátorů nedokáže odolávat velkému množství vlhkosti ve sklenících. [5] Následující odstavce se věnují konceptům robotů používaných v zemědělství.

1.3.1 Farmbot

System Farmbot je open-source projekt, zabývající se automatizací na venkovních záhonech s využitím volně dostupných dílů a 3D tisku. Tento koncept funguje na principu CNC obráběcích strojů. Nad celým záhonem jezdí rameno a vykonává předem naprogramované činnosti jako je například zalévání, sázení nebo dokonce odstraňování plevelů. Obsahuje také human-machine interface, díky kterému si může uživatel nastavit, kde a jak chce danou rostlinu pěstovat. [2]

1.3.2 Root-Ai

Tohoto robota vlastní firma AgTech a AppHarvest. Tyto firmy jsou jedny z největších světových stavitelů vnitřních farem na přírodní pěstování zeleniny a ovoce. Tento robot má za úlohu sběr rajčat ve sklenících, ve kterých funguje hydroponický systém zalévání. Mechanismus tohoto robota je na principu SCARA manipulátoru, díky čemu se jednoduše dostane na přesnou pozici, kde následně pomocí chapadla odtrhne plod bez poškození a zanechání jediné stopy. Taktéž disponuje 3D skenováním celé rostliny a vyhodnocováním zralosti plodu. [13]

1.3.3 Certhon Harvest Robot

Tento robot byl vyvinut na základě existující technologie firmy Dento a pomocí firmy Certhon byl upraven ke sběru rajčat v hydroponických sklenících. Jako základ byla použita 5-ti osá robotická ruka, která na základě integrovaných kamer vyhledá a odstříhne celý trs rajčat a opatrně uloží do přepravního boxu. Díky integrovanému světlu může robot pracovat ve dne i v noci. Robot se hlavně snaží o zlepšení výnosu, kvality a zisku implementací plně autonomní farmy. [17]

1.3.4 Srovnání

Roboti Root-Ai a Certhron Harvest Robot fungují na podobném principu. Základní rozdíl, ve kterém se tyto dva systémy liší, je druh manipulátoru. Root-Ai používá SCARA manipulátor a Certhron Harvest používá 5-ti osou robotickou ruku. Oba tyto roboti mají integrovanou umělou inteligenci na rozpoznávání zralosti a disponují možností sklizení zralých plodů. V důsledku toho jsou tyto dva roboti v cenové relaci dostupné pouze pro velké firmy. Na druhé straně je open-source projekt Farmbot, který můžeme vzhledem k jeho ceně a náročnosti na obsluhu použít i na domácích záhonech. Jeho nevýhoda spočívá v přístupu shora, kdy nemá dobrý přístup ke všem plodům, a není tedy možné použít umělou inteligenci na rozpoznávání zralosti plodů nebo ho implementovat do skleníku.

2 Koncepce

Základní koncepce automatizovaného skleníku spočívá v umístění několika senzorů vlhkosti a kapénkové závlahy k rostlinám. Tato koncepce má nevýhodu v tom, že vlhkostní senzory umístěné v půdě časem ztrácejí přesnost, a tím pádem je potřebné je měnit. Samotné zavlažování zavlažuje i místa, kde se nemusí ani nacházet rostliny, což má za důsledek zbytečné plýtvání vodou. Po každé sklizni je potřebné tento systém znovu předělat na základě nově vysázených rostlin. Pro efektivní zavlažování je potřebné minimalizovat povrch půdy, která zůstává vlhká, aby co nejméně docházelo k vypařování vody. Z tohoto důvodu jsme se rozhodli vytvořit modulární systém, který bude na základě vysázených rostlin zalévat pouze dané pozice s přesným množstvím vody tak, aby každá individuální rostlina měla jen tolik vody, kolik potřebuje. Zároveň tento systém nezanechává vlhkostní senzor v půdě a tak tolik nedochází ke ztrátě přesnosti senzoru.

2.1 Koncepce manipulátoru

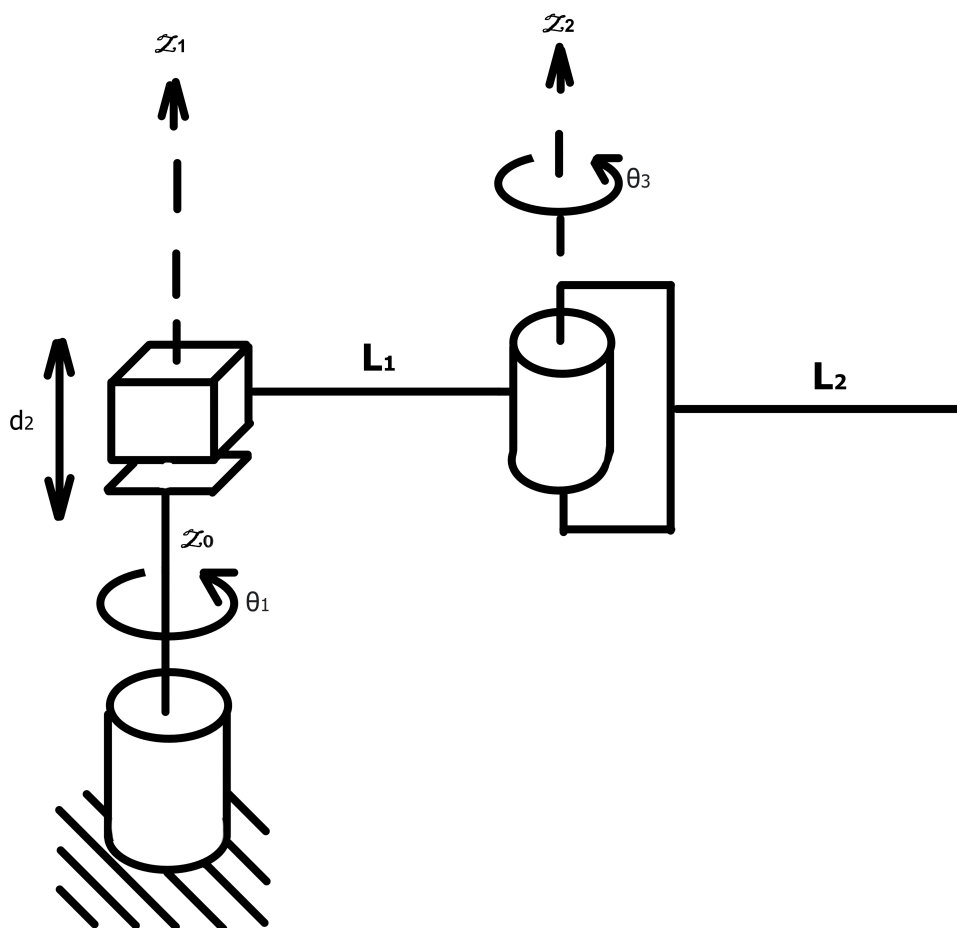
Při návrhu našeho manipulátoru jsme se snažili využít všechny výhody komerčně vyráběných systémů uváděných v kapitole 1.3 a odstranit co nejvíce jejich nevýhod. Nejprve jsme se rozhodovali odkud bude mít manipulátor přístup k rostlinám. V případě že bychom umístili manipulátor nad záhon, omezili bychom si tím možnost pohodlně zalévat rostliny přímo ke kořenům. V případě, že bychom zalévali rostliny shora, by na nich zůstávala voda, což by mělo za následek spálení částí listů a nežádoucí šíření plísní. Další výhodou umístění manipulátoru do úrovně rostlin spočívá v jednoduchosti konstrukce, pomocí které se manipulátor pohybuje po skleníku. Dále jsme uvažovali, jaký typ konstrukce pro daný manipulátor použít. Uvažovali jsme nad robotickou rukou a SCARA manipulátorem. V případě, že bychom použili robotickou ruku, zvýšila by se náročnost na program a celková cena. Jako základ jsme tedy použili manipulátor typu SCARA, abychom docílili dobrý přístup k rostlinám za dobrou cenu. Základní konstrukci jsme použili z volně dostupné a otestované koncepce, kterou je možné vytisknout z větší části na 3D tiskárně.

3 Návrh mechanické části

Jako základ jsme si tedy zvolili manipulátor typu SCARA s přídatnou možností se pohybovat po středu skleníku. Název SCARA vychází ze slov "Selective Compliant Articulated Robot for Assembly" což by se dalo přeložit jako selektivní kloubový robot pro montáž. [19]

3.1 Kinematika

Geometrie manipulátoru se rozděluje podle toho, kolik má posuvných a kolik rotačních os. Základní typy jsou například: otočné(RRR), sférické(RRP), válcové(RPP) nebo kartézské(PPP). Námí zvolený manipulátor typu SCARA spadá do kategorie RRP. Na obrázku 3.1 je znázorněna konfigurace našeho manipulátoru. Můžeme zde vidět, že má osu Z_0 a Z_2 rotační a osu Z_1 posuvnou nahoru a dolů. [19]



Obr. 3.1: Geometrie SCARA manipulátoru

Pro výpočet pozice ramena manipulátoru vzhledem k základně můžeme použít dopřednou nebo inverzní metodu kinematiky. Dopředná metoda kinematiky nám udává údaj v jaké pozici se manipulátor nachází při konkrétní pozici všech os. Inverzní metoda kinematiky nám na druhou stranu říká, jak máme nastavit osy, aby se manipulátor nacházel na konkrétní pozici. Pro náš projekt budeme používat inverzní metodu (3.1, 3.2) vzhledem k tomu, že chceme manipulátor dostávat na konkrétní pozici.

$$\theta_1 = \arctan\left(\frac{x}{y}\right) - \frac{\arctan(L_2 \cdot \sin(\theta_3))}{L_1 + L_2 \cdot \cos(\theta_3)} \quad (3.1)$$

$$\theta_3 = \arccos\left(\frac{x^2 + y^2 - L_1^2 - L_2^2}{2 \cdot L_1 \cdot L_2}\right) \quad (3.2)$$

V případě potřeby je možné do manipulátoru implementovat taktéž přímou metodu kinematiky (3.3, 3.4).

$$x = L_1 \cdot \cos(\theta_1) + L_2 \cdot \sin(\theta_1 + \theta_3) \quad (3.3)$$

$$y = L_1 \cdot \sin(\theta_1) + L_2 \cdot \cos(\theta_1 + \theta_3) \quad (3.4)$$

Výška (d2) je v našem případě nastavena na základě měření ultra sonického senzoru. Pozici celého manipulátoru vzhledem ke skleníku určujeme na základě předem definované pozice vysázených rostlin ve skleníku. [4]

3.2 CAD/CAM systémy

CAD/CAM systémy jsme používali na vytvoření virtuálního dvojčete celého systému. Celý systém jsme následně používali na kontrolu chyb, virtuální programování PLC programu a pro výslednou vizualizaci.

3.2.1 Virtuální dvojče

První zavedení pojmu virtuální dvojče bylo v technologickém plánu NASA z roku 2010, kdy byl definován jako: “an integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its flying twin”. Tato definice by se dala interpretovat jako virtuální prezentace fyzického systému za použití fyzikálních modelů, díky kterému je možné oboustranně sledovat různá data virtuálního a reálného systému a tímto způsobem simulovat samotný život systému. Tato prezentace je základní kámen průmyslu 4.0, vzhledem k tomu, že spojuje různé odvětví výroby dohromady. Hlavními výhodami virtuálního dvojčete jsou:

- rychlost prototypování a re-designování výrobku,
- vzhledem k převážně virtuální tvorbě méně nákladné,
- je možné předvídat chyby a adekvátně na ně reagovat,
- zmenšení odpadu při digitálním testování a prototypování,
- účinnější a názornější školicí prostředek,

Dle míry integrace virtuálního dvojčete můžeme daný model rozdělit do 3 kategorií:

- model - V této fázi se informace mezi fyzickým a virtuálním modelem vyměňují manuálně.
- shedow - V této fázi virtuální model přijímá informace z fyzického modelu, ale nikoli naopak.
- twinx - V této fázi probíhá obousměrná výměna informací mezi fyzickým a virtuálním modelem, a na základě těchto dat se systémy navzájem ovlivňují.

[18]

Náš model dosáhl integrace na úrovni shedow, vzhledem k tomu, že virtuální systém přijímá data z reálného modelu, ale nijak reálný model neovlivňuje. Případné úpravy je stále potřebné dělat manuálně.

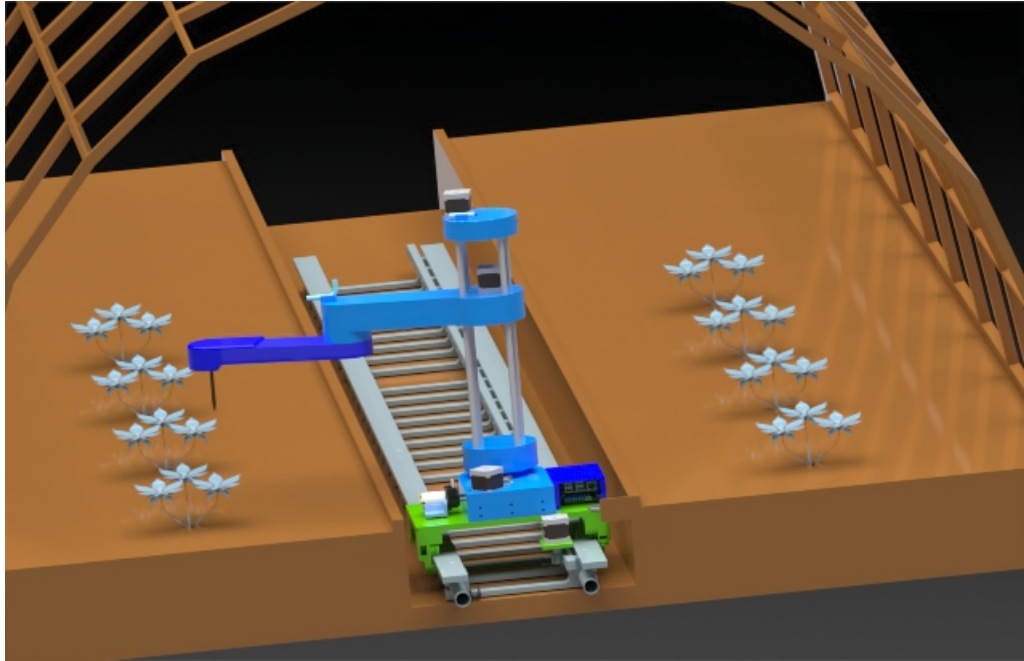
3.3 Siemens NX

Pro vytvoření virtuálního dvojčete jsme si vybrali modelovací program Siemens NX, vzhledem k tomu, že spojuje dohromady modelování, montáž a zpracování. Tento program má navíc možnost spojit výsledný model s PLC a otestovat v bezpečném prostředí simulace funkčnost kódu a modelu. Taktéž tento program obsahuje obsáhlou databázi různých spojovacích materiálů. [15]

3.3.1 Model

Pro vytvoření modelu jsme použili klasické parametrické modelování. Při tomto druhu modelování jsme si nejdříve nakreslili 2D sketch a následně jsme ho vytáhli do prostoru. Tímto způsobem jsme postupně vytvářeli díly, které jsme následně pomocí vazeb spojili do jedné montážní soustavy.

Tabulka 3.3.1 obsahuje kompletní seznam potřebných součástek, které není možné vytisknout na FDM 3D tiskárně, vzhledem k tomu, že by nesplňovaly požadovanou přesnost, pevnost nebo ohebnost. V seznamu není uvedena elektronika. Tu naleznete v kapitole 4.



Obr. 3.2: Výsledný 3D model

3.3.2 SCARA robot

Hlavní část celého modelu tvoří SCARA manipulátor, který jsme převzali z webové stránky howtomechatronics.com [4] a následně upravili. Hlavní úpravy z převzatého modelu spočívaly ve změně koncového ramena a samotné hlavy manipulátoru. Koncové rameno je navrženo tak, aby se hlava robota dala jednoduše vyměnit. Snažili jsme se docílit toho, aby celý robot byl modulární a mohl vykonávat více činností. Na konec samotného ramena jsme umístili ultra sonický senzor, díky kterému dokáže robot určit, jak daleko se nachází od povrchu nebo překážky. Hlavu manipulátoru jsme vymodelovali tak, aby byla schopná měřit vlhkost, a dále obsahuje průtok, přes který dokáže pomocí peristaltického čerpadla dávkovat množství vody. Samotný SCARA manipulátor je poháněn 3 krokovými motory. Motory na změnu úhlu jsou pomocí řemenic zpřevodovány pro přesnější otočení jednotlivých částí. Pohyb nahoru a dolů zabezpečuje další krokový motor, který je připojen ke kuličkovému šroubu a pomocí matice zvedá rameno. Volili jsme tento druh tyče vzhledem k jejímu plynulému pohybu a možnému většímu namáhání v této ose. Rameno jezdí za pomoci lineárních ložisek po hladkých tyčích připevněných k podstavě. Poloha se zjišťuje pomocí homing funkce, k čemu slouží koncové spínače na každé ose volnosti.

| Ložiska | kusy |
|-----------------------------|-------------|
| skf 4200 atn9 | 4 |
| lm10uu | 4 |
| vrl 607-z | 8 |
| 608rs | 5 |
| axiální ložisko 35x52x12 mm | 2 |
| axiální ložisko 40x60x13mm | 2 |
| Pojezdy | |
| přesné tyče h6-10mm-500mm | 4 |
| kuličkový šroub 1204 | 1 |
| ozubený hřeben | 1 |
| ozubená matice | 1 |
| Šrouby a matice | |
| m3,m4,m5,m8 | x |
| Řemeny 6mm | |
| 200mm | 1 |
| 300mm | 2 |
| 400mm | 1 |
| hadička 8mm | 1 |

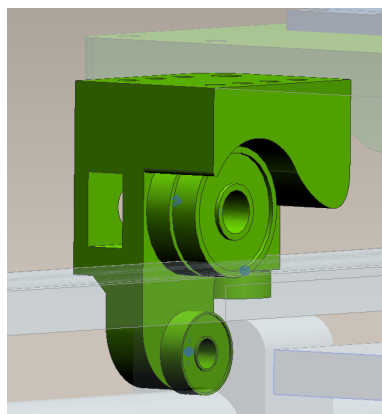
Tab. 3.1: Tabulka součástek

3.3.3 Základna

Základna zabezpečuje lineární posun robota po středu skleníku. K tomu používá ozubený hřeben, který je připevněn na kolejnicích, a ozubené kolo, kterým otáčí další krokový motor. Lineárním vedením jsme se inspirovali u firmy HepcoMotion modelem MHD. Lineární vedení 3.3 se skládá ze 4 kusů, kde každý kus obsahuje 2 ložiska, která zabraňují pohybu nahoru a dolů, a jedno ložisko, které zabraňuje pohybu do strany. Tento princip jsme vybrali, protože jeho jednotlivé díly se dají vytisknout na 3D tiskárně, čímž jsme se snažili zmenšit celkovou výrobní cenu. S tímto systémem je možné vytvářet mírné zatáčky a není nutné, aby byl povrch dokonale hladký.

3.3.4 Virtuální zprovoznění

Po vytvoření funkčního modelu jsme přešli k vytváření virtuálního dvojčete pomocí modulu Mechatronics Concept Designer (součást Siemens NX). Nejprve jsme všem pohybujícím se tělesům přidělili tuhost ("Rigid Body") a následně jsme jim přidělili omezení na konkrétní osy pohybu. Následným spojováním jednotlivých os pomocí



Obr. 3.3: Lineární vedení

ozubených kol jsme vytvořili požadovaný pohyb virtuálního dvojčete. Pro kontrolovaný pohyb jsme použili nastavení na konkrétní rychlost. Požadovaná poloha se zjišťuje pomocí kolize koncového senzoru a modelu. Pro měření vzdálenosti ramena od povrchu jsme použili senzor vzdálenosti, který simuluje ultra sonický senzor. Pro spojení virtuálního dvojčete s PLC jsme použili signály v tabulce 3.3.4. Pro nastá-

| List signálu | | | |
|-----------------|--------------|-------------|-------------------------|
| INPUT | OUTPUT | Typ | Funkce |
| Senzory | | | |
| triger SenOsa2 | Sen osa2 | BOOL | koncový senzor výšky |
| triger SenOsa3 | Sen osa3 | BOOL | koncový senzor ramena 1 |
| triger SenOsa4 | Sen osa4 | BOOL | koncový senzor ramena 2 |
| triger Sen Base | Sen base | BOOL | koncový senzor základny |
| sonic height | Sen sonic1 | INT | výška od půdy |
| Krokové motory | | | |
| base dir/step | base speed | BOOL to INT | rychlost základny |
| height dir/step | height speed | BOOL to INT | rychlost zvedání |
| arm1 dir/step | arm1 speed | BOOL to INT | rychlost ramena 1 |
| arm2 dir/step | arm2 speed | BOOL to INT | rychlost ramena 2 |

Tab. 3.2: Signály MCD

vení rychlosti motoru jsme použili podmínku, kde false signál nám nastaví rychlost na 0 a true signál step nám na základě true signálu dir nastaví kladnou rychlost, nebo na základě false signálu dir zápornou rychlost.

Všechny signály jsou následně pomocí UPC UA komunikace spojené a ovládané pomocí virtuálního PLC.

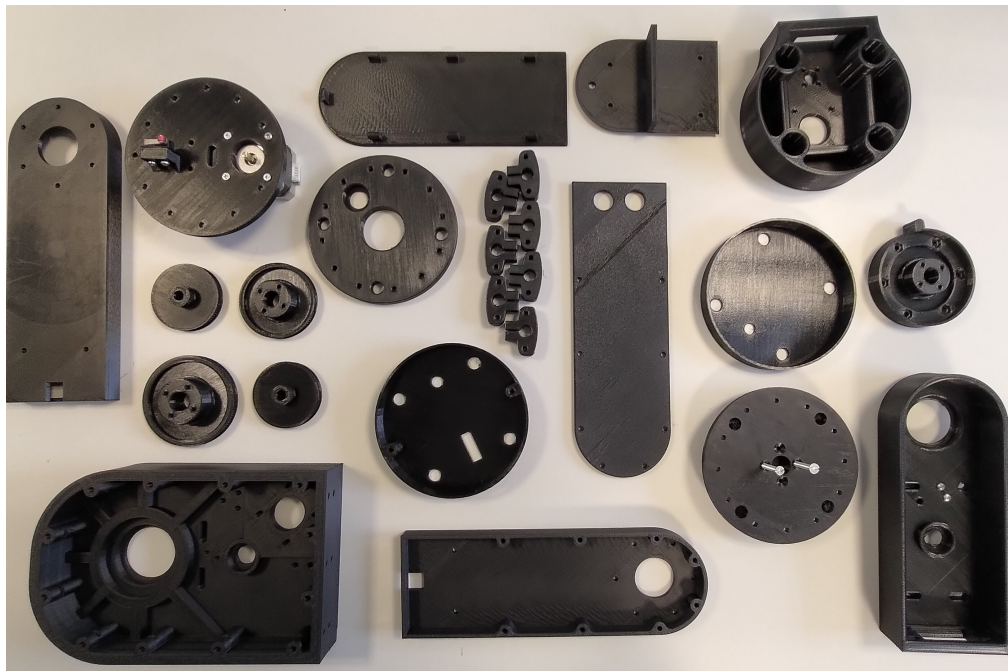
3.4 3D tisk

3D tisk je aditivní metoda obrábění, která umožňuje vytvoření komplexních tvarů a struktury s minimem odpadního materiálu. Taktéž umožňuje sériovou výrobu unikátních modelů za minimální cenu oproti masové výrobě. [8]

Na výrobu namodelovaných dílů, pokud nepočítáme díly z tabulek [3.3.1,4] jsme si vybrali metodu FDM 3D tisku. Tuto metodu výroby jsme si vybrali vzhledem k dostupnosti, ceně výsledného kusu a jednoduché obsluze samotné tiskárny. Model jsme namodelovali tak, abychom mohli použít materiál PET-G vzhledem k jeho mechanickým a teplotním vlastnostem.

Základem FDM tisku je rozpouštění termoplastických polymerů a jejich formování do nového tvaru. Samotné polymery jsou dostupné ve strunové formě stočené na kotoučích, ze kterých se pomocí hnacích kol vtlačují do hlavy zahříváné na kontrolovanou teplotu, kde se tento materiál dostává do polo-tekutého stavu. Následně se tento polo-tekutý materiál protlačuje skrz tenkou trysku, čímž vytváří ultra tenké vrstvy, které následně vrstva po vrstvě vytváří daný model. Postupnost vrstev je specifikovaná na základě programu ve formátu gcode, který je nahrán do FDM tiskárny. [8]

Na obrázku [3.4] jsou všechny tištěné díly potřebné k sestavení SCARA manipulátoru dostupné v příloze C.



Obr. 3.4: FDM tištěné díly

4 Elektronika

Elektroniku můžeme rozdělit na řídicí člen, akční člen a snímače. Hlavní napájecí napětí je DC 12/24V, které pohání akční členy celého manipulátoru s využitím čipu DRV8825, který spouští motory na základě SPI komunikace z Raspberry, která jde do GPIO expandéru, který následně určuje signály dir, step, enable, reset a sleep. Pro logickou část manipulátoru je přivedeno napětí 5V, pomocí kterých se napájí Raspberry Pi a následně samotné Raspberry distribuuje napětí do senzorů a povelování motorů. Celkové schéma zapojení se nachází v příloze A. V tabulce 4 se nachází seznam použité elektroniky.

| Rídící člen | Typ | Kusy |
|--------------------------------|-----------|------|
| Raspberry Pi | 4 model B | 1 |
| Akční člen | | |
| krokový motor | Nema 17 | 4 |
| peristaltické čerpadlo | 24V | 1 |
| Senzory | | |
| koncové spínače | 10T85 | 4 |
| ultra sonický senzor | GY-US42 | 1 |
| kapacitní senzor vlhkosti pudy | v2.0 | 1 |
| Moduly | | |
| Buck step down měnič | 12/24 | 1 |
| AD převodník | DFR0553 | 1 |
| Driver pro krokové motory | DRV8825 | 4 |
| GPIO expandér | MCP23S17 | 1 |
| H-můstek | DRV8871 | 1 |
| zdroj napájení | 24V | 1 |

Tab. 4.1: Seznam elektroniky

4.1 Řídicí člen

Pro řízení celého manipulátoru jsme použili desku Raspberry Pi 4 model B, která pomocí SPI a I2C komunikace přijímá jednotlivé signály ze senzorů a vysílá signály do motorů. Pouze koncové spínače jsou k Raspberry Pi připojeny na přímo, vzhledem k tomu, že potřebují co nejmenší zpoždění.

4.1.1 Raspberry Pi 4 model B

Hlavními důvody, proč jsme si zvolili jako hlavní řídicí jednotku Raspberry Pi 4 je jeho nízká cena, dostupnost, Open Source hardware/ software, drátové/bezdrátové spojení a možnost rozšíření o různé kompatibilní moduly. V tomto typu Raspberry Pi najdeme 4 jádrový ARM64, který má 2 rychlá izolovaná jádra vyhrazená na komunikaci. Na zbylých dvou jádrech běží operační systém. [\[9\]](#)

Jako operační systém jsme zvolili Raspbian. Celý program jsme následně vytvářeli pomocí rozšíření pro Raspberry Pi od firmy Codesys, které nám umožní používat Raspberry Pi jako "soft PLC". Na desku je možné se připojit pomocí webového rozhraní a kontrolovat tak běh programu pomocí HMI na jakémkoliv zařízení disponující webovým připojením.

4.1.2 SPI a I2C komunikační protokoly

Tyto komunikační protokoly jsme v tomto projektu použili na komunikaci s perifériemi manipulátoru. Vzhledem k tomu, že není potřebné přenášet velké množství dat za krátký čas, není potřebné používat protokoly jako jsou například Ethernet nebo USB. Pro komunikaci s akčními částmi jsme zvolili SPI komunikační protokol vzhledem k tomu, že je full duplex, a tedy je o něco rychlejší. Sensorická část na druhou stranu plně postačuje na half duplex komunikaci, a proto jsme pro ni zvolili I2C komunikační protokol. [\[10\]](#)

SPI komunikační protokol

SPI je zkratka pro "Serial Peripheral Interface" což by se dalo interpretovat jako rozhraní pro sériovou komunikaci s perifériemi. Tento protokol spojuje digitální zařízení čtyřmi signálovými linkami na principu lídr/následovník, takzvaný Master/Slave. Jsou to:

- **SCLK** synchronizační hodinový signál posílaný lídrem všem následovníkům
- **SS_n** navazování komunikace s konkrétním následovníkem.
- **MOSI** data od lídra následovníkovi (Master Out-Slave In)
- **MISO** data od následovníka lídrovi (Master In-Slave Out)

Toto uspořádání lídr/následovník dovoluje pouze jednoho lídra, který je v našem případě Raspberry Pi. Následovník je pro náš projekt pouze GPIO expandér mcp23s17, avšak je stále možné připojit další moduly, které by používaly toto rozhraní. [\[10\]](#)

I2C komunikační protokol

I2C je zkratka pro "Inter-integrated circuit" a původně byl navržen pro komunikaci s perifériemi televize. Na rozdíl od SPI funguje toto rozhraní na multi-lídr principu

a používá pouze dvě signálové linky, napájení a zem. Signálové linky jsou:

- **SCL** sériový hodinový signál
- **SDA** sériová data

Vzhledem k tomu, že toto rozhraní nemá žádný výběr čipů je potřebné, aby každý čip měl přidělenou unikátní 7-bitovou následovníkovu adresu. Komunikace probíhá pomocí stanoveného protokolu po 8-bitech. Jakmile nějaké zařízení vyšle startovací podmínku všechna ostatní zařízení se stávají následovníkem. V následujících 7 bitech lídr pošle adresu zařízení, se kterým chce komunikovat. Poslední bit z prvních 8-bitů rozhoduje o tom, jestli chce lídr číst nebo posílat data. Po tomto bitu následují samotná data, která jsou prokládána příznakovým bitem. Příznakový bit slouží ke kontrole, zda byla správně poslána poslední sekvence 8-bitu. Bit je validní pouze v impulzech hodinového signálu. Na konci je poslaný stop bit a všechna zařízení znovu čekají na další start bit. [\[10\]](#)

4.2 Plošný spoj

Pro ovládaní krokových motorů a peristaltického čerpadla jsme si navrhli vlastní plošný spoj. Tuto variantu jsme volili z důvodu, že Codesys umožňuje použít pouze 12 vstupně/výstupních pinů, což je pro náš projekt nedostačující. Na desku jsme umístili expandér, který nám na základě SPI komunikace povoluje 16 výstupů, které spínají jednotlivé vstupy modulu DRV8825 a h-můstek pro spínání peristaltického čerpadla. Na návrh jsme použili program Eagle. Celý návrh se nachází v příloze C ve složce PCB.

4.2.1 DRV8825

Tento modul slouží k otáčení krokového motoru v bipolárním zapojení. Umožňuje povelování pomocí 3,3/5V logiky a motor dokáže otáčet s pomocí až 45 V v našem případě pomocí 24 V. Na námi navržené desce plošných spojů se nachází 3 přepínače, které slouží k nastavení krokování. Všechny moduly se nastavují na stejné krokování jedním přepínačem. Všechny moduly mají spojené vstupy RESET a SLEEP. Tyto dva piny je potřeba držet na logické jedničce, pokud nechceme, aby se nacházely ve stavu, kdy pouze čekáme a nic nevykonáváme (SLEEP) nebo, když potřebujeme resetovat moduly například po nějakém chybném stavu. Tyto piny se dají ovládat pouze v servisním stavu v HMI. Další ovládací piny jsou STEP, DIR a ENABLE. Pouze pin ENABLE má obrácenou logiku, takže pokud chceme používat modul musí být ENABLE na logické 0. Pomocí pinu DIR určujeme směr otáčení a pomocí pinu STEP na základě impulsu nastavujeme následující krok. Na desku jsou vyvedeny testovací plošky, díky kterým je možné zjistit, zda se modul nenachází ve FAULT

| | | | |
|------------|------------|------------|-----------|
| 1.přepínač | 2.přepínač | 3.přepínač | Krokování |
| Vypnutý | Vypnutý | Vypnutý | Full Step |
| Zapnutý | Vypnutý | Vypnutý | Half Step |
| Vypnutý | Zapnutý | Vypnutý | 1/4 Step |
| Zapnutý | Zapnutý | Vypnutý | 1/8 Step |
| Vypnutý | Vypnutý | Zapnutý | 1/16 Step |
| Zapnutý | Vypnutý | Zapnutý | 1/32 Step |
| Vypnutý | Zapnutý | Zapnutý | 1/32 Step |
| Zapnutý | Zapnutý | Zapnutý | 1/32 Step |

Tab. 4.2: Krokování motoru na základě přepínače

stavu. Do tohoto stavu se modul může dostat v případě přehřátí, při příliš velkém proudu, při příliš malém napětí nebo při zkratu. Pod každým modulem je umístěn jeden 100uF elektrolytický oddělovací kondenzátor pro přívodní napětí do motoru. Pro správné napětí na motoru je potřebné správně nastavit potenciometr na modulu. Krokové motory jsou připojeny na poslední 4 volné výstupy modulu. [6]

Výstup z GPIO expanderu do jednotlivých vstupů DRV8825 modulu a H-můstku jsou v tabulce [4.2.1](#).

| Pin MCP23S17 | Pin DRV8825 | Funkce |
|--------------|---------------|-------------------------------------|
| A0 | height-dir | změna směru nahoru/dolu |
| A1 | height-step | PWM impulzy kroku |
| A2 | height-enable | povolení motoru výšky |
| A3 | base-dir | změna směru dopředu/dozadu |
| A4 | base-step | PWM impulzy kroku |
| A5 | base-enable | povolení motoru podstavy |
| A6 | arm1-dir | směr otáčení první půlky ramena |
| A7 | arm1-step | PWM impulzy kroku |
| B0 | pump-p | dopředný chod pumpy |
| B1 | pump-n | zpětný chod pumpy |
| B2 | arm1-enable | povolení otáčení první půlky ramena |
| B3 | reset | resetování všech modulu |
| B4 | sleep | sleep mód všech modulu |
| B5 | atm2-dir | směr otáčení druhé půlky ramena |
| B6 | atm2-step | PWM impulzy kroku |
| B7 | arm2-enable | povolení otáčení druhé půlky ramena |

Tab. 4.3: Výstupy z expandéru a vstupy do DRV8825 modulu

4.2.2 H-můstek

Pro řízení peristaltického čerpadla jsme použili modul od firmy Adafruit, který používá čip s H-můstkem DRV8871. Tento čip umožňuje řídit jeden kartáčový DC motor pomocí 5 voltové logiky na vstupu. Napájecí napětí pro tento motor může být v rozmezí 6,5 V až 45 V s nejvyšším možným proudem až 3,6 A. Na desce je umístěn rezistor, který omezuje výstupní proud na přibližně 2A. Tento rezistor se dá nahradit a omezit tak proud na konkrétní hodnotu. Do čipu není potřebné přivádět 5 V napájení pro logické vstupy. V případě, že bychom chtěli čerpadlo zpomalit, je možné na vstup posílat PWM signál, čímž se omezí rychlost. Tento signál může dosahovat maximálně 200kHz. Pro naši aplikaci dáváme na vstup pouze dvoustavovou logiku vzhledem k tomu, že je potřeba, aby nám čerpadlo jelo co nejrychleji. [1]

4.3 Senzory

Senzory snímají konkrétní fyzikální, chemickou nebo biologickou veličinu. Následně jsou tyto signály převedeny na elektrické signály, které jsou pomocí AD převodníku převedeny na číselnou hodnotu a následně poslány do řídicího členu. V řídicím členu probíhá normalizace, kde z číselné hodnoty AD převodníku vzniká pomocí přepočtu hodnota měřené veličiny. Na základě této veličiny jsou následně vykonávány konkrétní úkony. [14]

4.3.1 AD převodník

Většina použitých senzorů převádí měřenou fyzikální hodnotu na napětí 0 až 5 V. Pro převod na číslo jsme použili 16-bitový AD převodník, který dokáže měřit až 4 analogové hodnoty nebo 2 diferenciallyně, které převádí pomocí delta-sigma převodníku. Tuto hodnotu následně prostřednictvím I2C komunikace posílá do Raspberry Pi.

4.3.2 Koncové spínače

Pro zjišťování polohy jsou na robota připevněny na konkrétních pozicích koncové spínače. Díky nim dokáže robot určit nulovou pozici a následně od ní na základě známých převodů určit konkrétní polohu. Jako koncové spínače byly použity mikrospínače, které jsou přímo připojené do Raspberry Pi.

4.3.3 Senzor vzdálenosti

Pro kontrolu vzdálenosti od povrchu jsme použili ultra sonický senzor, který na základě neslyšitelného zvuku měří vzdálenost od nejbližšího objektu. Senzor funguje tak, že vyšle zvukové vlny o konkrétní frekvenci a měří počet impulzů do doby, než se vrátí vlna o stejné frekvenci jako byla vyslána. Následně na výstup pošle počet naměřených impulzů, který se jako číslo pošle po sběrnici I2C do řídicího členu. Tento senzor jsme zvolili pro jeho cenu, dostatečnou přesnost, jednoduchost použití a malé rozměry. [12]

4.3.4 Senzor vlhkosti

Pro měření vlhkosti půdy se dají použít dva typy senzorů. První měří změnu vodivosti půdy na základě měření odporu mezi dvěma plochami, a druhý měří změnu kapacity mezi dvěma plochami. Měření pomocí odporu má nevýhodu v tom, že je potřebné, aby měřené plochy byly odkryté. Tím může během času docházet ke korozi, a tak i ke změně přesnosti snímače. Z tohoto důvodu jsme si pro tuto práci vybrali kapacitní snímač vlhkosti, který může mít zakryté měřící plochy, a tím nedochází ke korozi.



Obr. 4.1: Kapacitní snímače vlhkosti

4.4 Akční člen

Jako akční člen jsou v této práci použity krokové motory, které zabezpečují pohyb robota, a peristaltické čerpadlo, díky kterému dokáže robot zalévat.

4.4.1 Krokové motory

Krokový motor je bezkartáčový DC motor, který mění úhel natočení na základě programu. Rotace nastává v důsledku změny magnetického pole v cívkách statoru a díky permanentnímu magnetu v rotoru. Krokové motory jsou digitální akční členy, které přijímají signály z programovatelné jednotky, neboli inkrementální pohybový kontrolní systém. [3]

Robota pohání 4 krokové motory Nemma 17, které jsou ve 4-drátovém provedení a jsou zapojeny do DRV8825 modulu jako bipolární krokové motory. Pomocí signálu z SPI komunikace povoluje GPIO expandér jednotlivé vstupy čipu DRV8825, které následně napájí postupně jednotlivé cívky krokového motoru na základě nastavené hodnoty přepínače, který nastavuje krokovaní (plný krok/poloviční krok/mikrokrok) [4.2.1] a výsledné polohy motorů.

4.4.2 Peristaltické čerpadlo

Peristaltické čerpadlo funguje tak, že rotující váleček stlačuje pružnou hadičku, která se následně snaží vrátit do původní polohy, čímž se vytváří podtlak a čerpadlo tak dokáže tlačít kapalinu. Hlavní výhoda tohoto čerpadla je v tom, že dokáže držet konstantní tlak v hadičce bez potřeby energie a dokáže díky tomu nasát kapalinu do hadičky bez potřeby mít v hadičce kapalinu. Nevýhodou je vcelku malý průtok, avšak k našemu použití je průtok dostačující. Vzhledem k tomu, že toto čerpadlo dokáže přesně dávkovat kolik kapaliny načerpá není potřebné do trubice instalovat snímač pro měření průtoku.

5 Softwarové vybavení

Vzhledem k tomu, že kontrolér Raspberry Pi běží na operačním systému Raspbian, který má základ v Linux operačním systému je možné tento kontrolér programovat různými způsoby. Nejčastěji se tento kontrolér programuje pomocí programu MicroPython, Python nebo C. Vzhledem k tomu, že jsme se snažili co nejvíce přiblížit průmyslovému řešení, abychom mohli využívat výhody například virtuálního dvojčete, rozhodli jsme se programovat kontrolér pomocí PLC programovacího prostředí Codesys, který obsahuje rozšíření pro Raspberry Pi, díky kterému je možné tento kontrolér programovat jako "soft PLC". První verze programu byla vytvořena hlavně pro simulování vlastností manipulátoru v programu Siemens NX. Z tohoto důvodu jsou zde zanedbány měřené vstupy, jako je například měření vlhkosti. Druhá verze už používá rozšíření pro Raspberry Pi, a tím pádem tam jsou nastaveny všechny vstupy, výstupy a komunikace. Obě verze jsou dostupné v příloze C ve složce Codesys.

5.1 Soft PLC

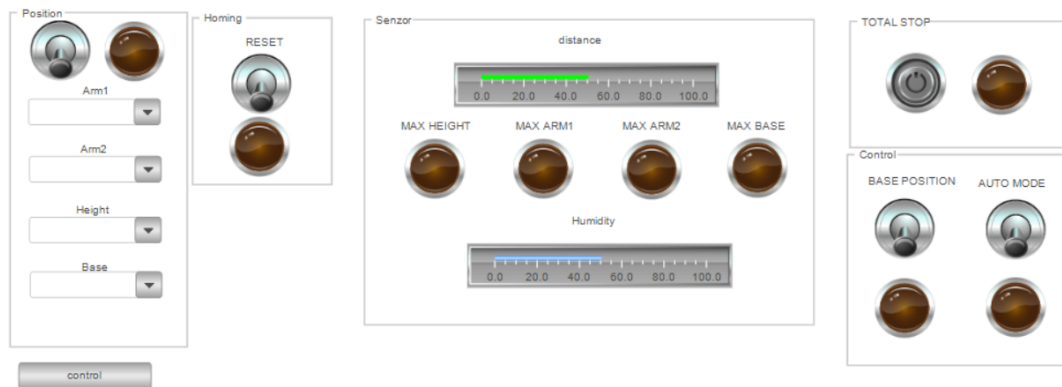
Nainstalováním programu emulující funkce klasického PLC do počítače, v našem případě Raspberry Pi, můžeme vytvořit softwarové PLC, pro které následně platí stejná pravidla programování jako u standardních PLC. K takto vytvořenému soft PLC je většinou potřebné připojit vstupně/výstupní jednotku, což v našem případě je již součástí Raspberry Pi desky. Výhody tohoto řešení spočívají v nízké ceně, rychlosti a velké operační paměti, avšak zároveň systém ztrácí oproti klasickému PLC robustnost a částečně real-time zpracování dat. Vzhledem k tomu, že hlavní řídicí část je počítač, je jednodušší propojit řídicí a informační část. [7]

Rozhodli jsme se řešit celý problém pomocí soft PLC hlavně kvůli nízké ceně, možnosti jednoduše propojit řídicí a informační systém, popřípadě do budoucna možnosti zpracování obrazu přímo v kontroléru. Nevýhody spočívající v real-time zpracování částečně řeší program codesys, který si vyhraduje jedno jádro pro operační systém a zbylé používá na zpracování dat.

5.2 HMI

HMI je prostředí vytvořené pro komunikaci mezi robotem a člověkem. V první verzi programu lze vykonávat základní funkce, které sloužily pro testování pomocí virtuálního dvojčete. Zapnutí funkce Homeing uvede manipulátor do pozice, kdy jsou všechny koncové spínače sepnuty a nastaví nulovou pozici. Bez první kalibrace nelze manipulátor ovládat vzhledem k tomu, že hrozí náraz na koncích os. Ve funkci Position uživatel nastavuje konkrétní pozici všech os manipulátoru. Tato funkce se

dá použít například pro servisní úkony nebo pro manuální manipulaci s robotem. Zapnutím funkce Base Position se robot přesune do pozice, kde jsou všechny osy nejméně mechanicky namáhány. Po zapnutí funkce Auto mode robot vykoná předem vytvořený program, který má za úlohu zalévat postupně všechny rostliny. Při vzniku problému je možné zmáčknout tlačítko TOTAL STOP, které zastaví všechny motory a do doby než se toto tlačítko opětovně zmáčkne nelze s robotem nijak manipulovat. Ve středu se nachází informační část, která ukazuje sepnutí koncových spínačů a posuvníky, které ukazují vzdálenost hlavy manipulátoru od povrchu a vlhkost půdy. Ve druhé verzi jsme v HMI přidali druhou obrazovku pro manuální



Obr. 5.1: HMI-Verze dva-hlavní obrazovka

řízení jednotlivých výstupů GPIO expandéru. Tato obrazovka slouží hlavně pro testování, opravu chyb a případné ladění na reálném manipulátoru. Mezi obrazovkami je možné přepínat pomocí tlačítka v rohu obrazovky.



Obr. 5.2: HMI-Verze dva-druhá obrazovka

5.3.1 Komunikace

Pro komunikaci pomocí SPI a I2C používáme již předpřipravené zařízení v programu Codesys. Díky těmto předpřipraveným knihovnám je potřebné nastavit pouze základní údaje. Pro SPI komunikaci s GPIO expandérem je potřebné nastavit rychlost komunikace, a dále které piny slouží jako výstupy, a které jako vstupy. Následně je možné do jednotlivých bitů nahrávat konkrétní hodnoty (5.1).

```
1 MCP23S17.byOutputsB.3:=gvl.reset;
2 MCP23S17.byOutputsB.4:=gvl.sleep;
```

Výpis 5.1: příklad zapisování proměnných pro SPI

Následnou komunikaci si program moderuje sám.

Pro I2C komunikace je potřebné do zařízení pro I2C přidat konkrétní součástky, se kterými chceme komunikovat a nastavit jim jejich unikátní adresu. Následně je možné na základě příkazů pro danou součástku sledovat dané hodnoty. Ve výpisu 5.2 je ukázka kódu pro použitý AD převodník s čipem ADS1115. První řádek říká, který vstup AD převodníku je aktivní. Druhý řádek říká, zda se jedná o diferenciální hodnotu ze dvou vstupů nebo single hodnotu z jednoho vstupu. Třetí řádek nastavuje úroveň napětí. Na čtvrtém řádku je znázorněna standardizace pro senzor vlhkosti, kdy při hodnotě 36000 je senzor ponořen do vody a při hodnotě 58000 je úplně suchý. Tyto hodnoty jsme následně převedli na procentuální hodnoty vlhkosti.

```
1 ADS1115.ChannelConfigs[0].Active:=TRUE;
2 ADS1115.ChannelConfigs[0].InputMux:=Single_0;
3 ADS1115.ChannelConfigs[0].PGA:=FSR_6144mV;
4 gvl.sen_humini :=(ADS1115.Channels[0].Value-38000)*0.005;
```

Výpis 5.2: příklad proměnných pro AD převodník

5.3.2 Softwarová PWM

Vzhledem k tomu, že modul DRV8825 potřebuje pro přechod na další krok náběžnou hranu, vytvořili jsme pomocí časovačů softwarovou PWM (5.3). Vzhledem k tomu, že jsme neimplementovali rampu pro krokové motory z důvodu zjednodušení programu, a vzhledem k tomu, že nepotřebujeme dosahovat větší rychlosti, mohli jsme použít stejnou PWM pro všechny krokové motory, a pouze při zapnutí jakéhokoli STEP zapisovat proměnou "TMP" na konkrétní výstup.

```
1 t1(IN:=TMP, PT:=INT_TO_TIME(10));
2 t2(IN:=NOT TMP, PT:=INT_TO_TIME(10));
3 IF t1.Q THEN
4   TMP:=FALSE;
5 END_IF
6 IF t2.Q THEN
7   TMP:=TRUE;
8 END_IF
```

Výpis 5.3: Softwarová PWM

Aktuální perioda je pevně nastavena na 20ms, ale zároveň je možné tuto hodnotu nahradit proměnnou a ovládat ji pomocí HMI. Maximální hodnota SPI přenosu je nastavena na 50 000 bitů za sekundu. Jedna zpráva musí obsahovat hodnotu všech 16 výstupů, což odpovídá 16 bitům. Pro jistotu počítejme radši s rezervou a tedy jedna zpráva obsahuje 256 bitů. Jedna zpráva se tedy odešle každých 5ms. V případě potřeby je možné přenos SPI komunikace zrychlit až na hodnotu 1 000 000 bitů za sekundu, avšak v tu chvíli používáme větší výpočetní výkon.

5.3.3 Pozicování

V aktuálním programu není implementovaná kinematika a manipulátor se otáčí pouze na základě předem daného času pro konkrétní úhel natočení. V případě změny krokování je potřebné přepočtení konstantu změnit v poměru změny krokování.

```

1 A1TON(IN:=arm1, PT:=arm1time);
2 IF gvp.arm1val < gvp.arm1new THEN
3   arm1time := 1000 * INT_TO_TIME(gvp.arm1new - gvp.arm1val);
4   GVL.arm1_dir := FALSE;
5   arm1 := TRUE;
6   GVL.arm1_step := TRUE;
7 END_IF
8 IF gvp.arm1new < gvp.arm1val THEN
9   arm1time := 1000 * INT_TO_TIME(gvp.arm1val - gvp.arm1new);
10  GVL.arm1_dir := TRUE;
11  arm1 := TRUE;
12  GVL.arm1_step := TRUE;
13 END_IF
14 IF A1TON.Q = TRUE THEN
15   GVL.arm1_step := FALSE;
16   arm1 := FALSE;
17   gvp.arm1val := gvp.arm1new;
18 END_IF

```

Výpis 5.4: Pozicování na základě času

5.3.4 Automatický mód

Automatický mód je navržený jako sekvenční stavový automat. V jednotlivých stavech se nastavují konkrétní nové pozice, které jsou následně pomocí funkce "position" vykonávány. Jakmile jsou všechny osy na správném místě program přechází do dalšího kroku. Samotné zalévání vykonává funkce "measure", která jakmile vykoná všechny potřebné úkony automaticky posouvá automat do dalšího stavu. Před každým vykonáním automatického módu se manipulátor zkalibruje na nulovou pozici, aby byla zaručena stejná trajektorie pro každé spuštění. Na konci se manipulátor vrací do základní pozice. V případě jakéhokoli neočekávaného stavu se aktivuje total stop tlačítko. V případě změny rozložení rostlin je potřebné tento kód upravit na základě nových pozic.

```

1 CASE GVP.istav_auto OF
2   0:gvp.kalibrace:=TRUE;
3   1: gvp.arm1new:=14;
4     gvp.arm2new:=16;
5     gvp.basenew:=3;
6     gvp.heightnew:=15;
7     gvp.position:=TRUE;
8     IF gvp.arm1new=gvp.arm1val AND gvp.arm2new=gvp.arm2val AND gvp.basenew=gvp.
baseval AND gvp.heightval=gvp.heightnew THEN
9       gvp.istav_auto:=2;
10    END_IF
11   2:
12     gvp.measure_height:=TRUE;
13

```

Výpis 5.5: základ stavového automatu pro automatický mód

5.3.5 Funkce měření vlhkosti

Měření a zalévání zabezpečuje funkční blok "measure", který nejdříve sníží výšku hlavy tak, aby byl snímač vlhkosti vždy stejně hluboko zapíchnutý do půdy. Následně se na základě výsledku měření tohoto senzoru přepočítá čas zapnutí peristaltického čerpadla, které nadávkuje přesné množství vody do záhonu.

```

1 IF heightok=FALSE THEN
2   gvl.height_step:=TRUE;
3   gvl.height_dir:=FALSE;
4   IF gvl.Sen_sonic1 < 15 THEN
5     gvl.pump_p:=TRUE;
6     heightok:=TRUE;
7   END_IF
8 END_IF
9
10 humidity_T(IN:=heightok, PT:=100*LREAL_TO_TIME(gvl.Sen_humini));
11 IF humidity_T.Q THEN
12   gvp.heightval:=20;
13   gvl.pump_p:=FALSE;
14   gvl.height_step:=FALSE;
15   gvp.istav_auto:=gvp.istav_auto+1;
16   gvp.measure_height:=FALSE;
17 END_IF

```

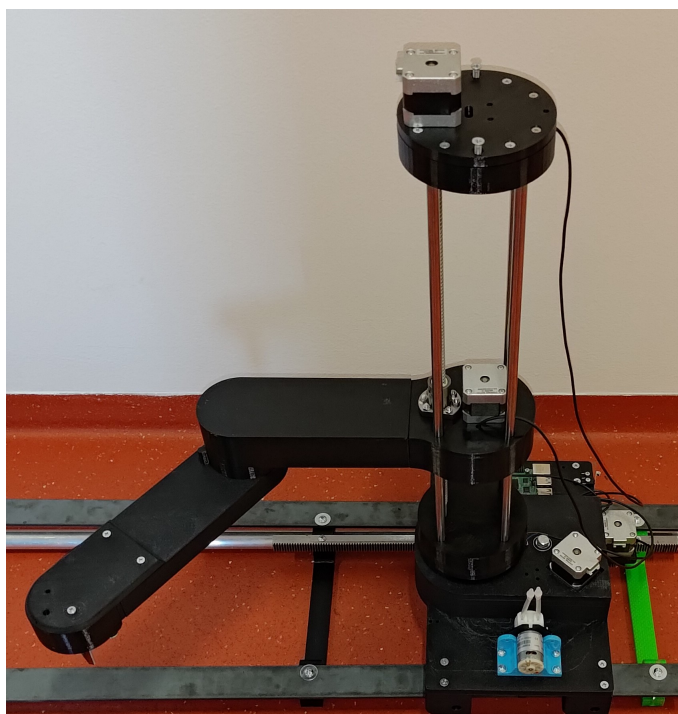
Výpis 5.6: Pozicování na základě času

6 Implementace

Celková implementace se dá rozdělit do několika kroků. První krok byl 3D tisk všech dílů a objednávání všech potřebných součástí. Druhým krokem bylo postavení celého robota a vytvoření kabeláže. V posledním kroku jsme zprovoznili elektroniku a implementovali program. Příloha B obsahuje stručný návod, jak zprovoznit celý manipulátor včetně implementování programu a ovládání HMI.

6.1 Sestavování manipulátoru

Po vytištění všech dílů jsme nejdříve sestavili odděleně ruku, věž a základnu manipulátoru. Separátně jsme také sestavili posuvnou desku a kolejnici. Následně jsme tyto díly spojili, umístili krokové motory a napnuli jsme řemeny. Po natažení kabelů jsme celý manipulátor umístili na posuvnou desku a připevnili na ni elektroniku.



Obr. 6.1: Sestava manipulátoru

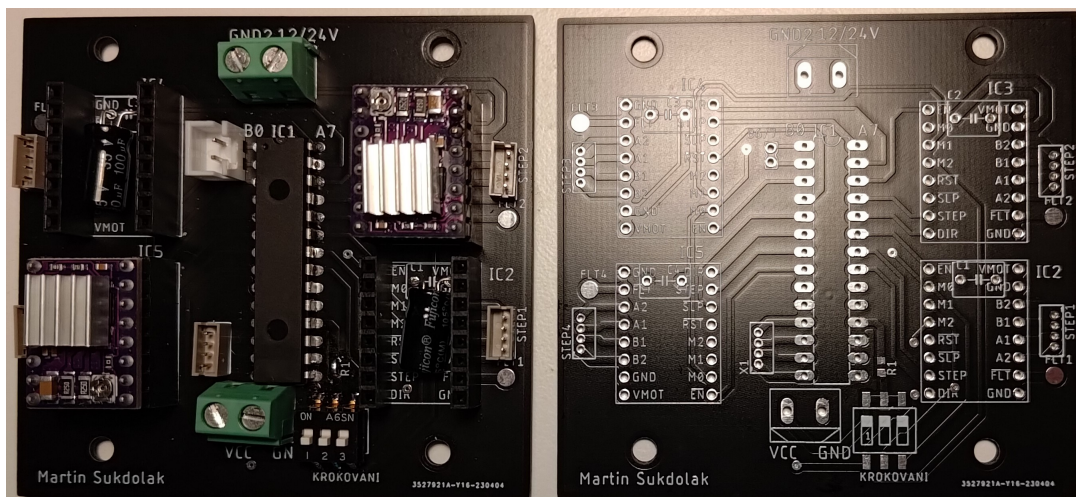
6.2 Elektronika

Hlavní napájecí napětí je 24 V, které vede do h-můstku, který následně napájí čerpadlo. Abychom eliminovali co nejvíce propojů, využili jsme toho, že h-můstek

obsahuje dvě možnosti na připojení napájecího napětí, a proto jsme druhý vstup použili jako spojovník pro napájecí napětí. Následně jsme toto napětí přivedli do svorkovnice námi navrženého modulu. Z této svorkovnice jsme zároveň brali napětí do modulu, který napájí samotné Raspberry Pi. Na celou 24V větev jsme použili 1.5mm široké kabely. Jak se píše v kapitole [4.1.2](#) pro komunikaci pomocí SPI jsme použili 4 datové a 2 napájecí kabely. Pro komunikaci pomocí I2C jsme použili 2 datové a 2 napájecí kabely. SPI komunikaci používáme pouze pro komunikaci s čipem MCP23S17 a jsou tím pádem tyto kabely napojené pouze do tohoto čipu. Na druhou stranu I2C komunikace komunikuje s AD převodníkem a ultra sonickým senzorem. Obě tyto zařízení mají přidělenou unikátní adresu, a tím pádem je možné použít stejný datový kabel. Pro omezení počtu propojek jsme opět využili toho, že AD převodník má dvě možnosti pro připojení datových a napájecích kabelů, a jeden vstup jsme využili pro prodloužení kabelů do ultra sonického snímače.

6.2.1 Osazování plošného spoje

Plošný spoj byl vyroben u firmy jlcpcb bez osazení. Vzhledem k tomu, že na desce jsou umístěny teplo citlivé součástky, začali jsme s osazováním přepínače a konektorů. Na desku jsme umístili dutinkové lišty, aby bylo možné v případě poruchy vyměnit modul DRV8825. Do prostoru, který nám tímto vznikl jsme umístili elektrolytické kondenzátory. Nakonec jsme umístili samotný čip MCP23S17.

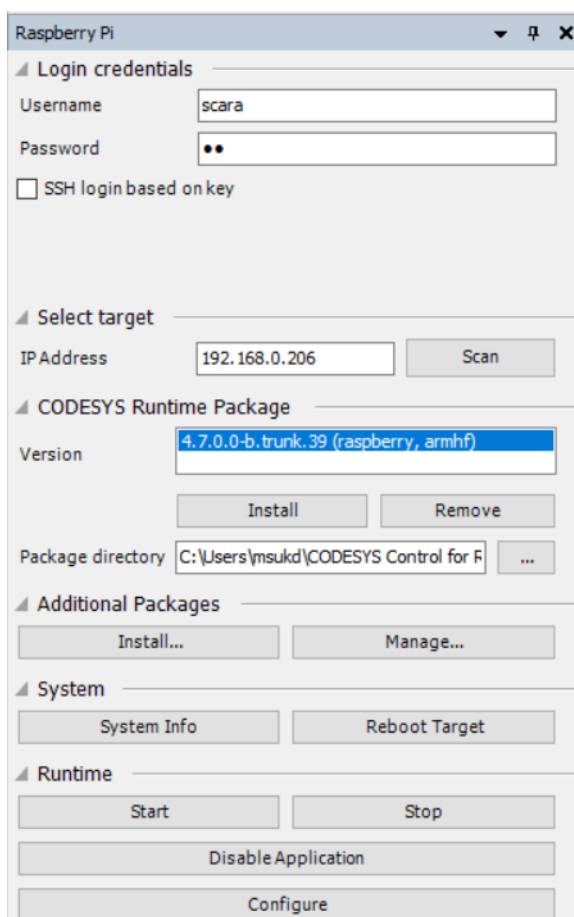


Obr. 6.2: Plošný spoj

6.3 Implementace programu

Před samotnou implementací programu je potřebné nahrát do Raspberry Pi operační systém Raspbian. Po spuštění je potřebné v operačním systému nastavit SSH spojení pro nahrávání kódu prostřednictvím wifi připojení. Taktéž je potřebné povolit v biosu výstup SPI a I2C komunikaci.

Codesys disponuje funkcí, kdy si dokáže sám najít cestu mezi počítačem a Raspberry, v případě více Raspberry Pi na stejné síti je potřeba zkontrolovat správnou IP adresu. Po nakonfigurování cesty je možné spouštět a nahrávat program obdobně jakoby bychom pracovali s PLC. Codesys taktéž umožňuje resetování a aktualizaci softwaru. Prostřednictvím manuálního módu jsme otestovali všechny výstupy z expandéru a pomocí proměnných v kódu při běžícím programu hodnoty ze senzorů. Na základě těchto hodnot jsme upravili konstanty v programu tak, aby nám reprezentovaly standardizované hodnoty v inženýrských jednotkách.



Obr. 6.4: Spojení Raspberry Pi a programu Codesys

7 Plány do budoucna

Celý manipulátor jsme koncipovali tak, aby bylo s jeho pomocí možné vykonávat další různé úkony. Manipulátor již disponuje přípravou na vyměnitelnou hlavu, aby bylo možné vykonat úlohy jako je sázení, pletí, popřípadě sběr plodů.

7.1 Strojové vidění

Raspberry Pi disponuje vstupem pro kameru, která by se dala připevnit na konec manipulátoru a na základě pořízených fotografií by manipulátor dokázal rozhodovat o potřebě přesadit, zbavit se plevelu nebo sebrat plody. Program Codesys taktéž disponuje možností pořizovat fotografie, popřípadě video. Vzhledem k tomu, že ve skleníku je minimum pohybu, bylo by možné například pomoci fotogrammetrie vytvářet 3D obraz rostlin, díky kterému bychom mohli rozhodovat o potřebě přesadit rostlinu. Tyto údaje by bylo možné zpracovávat na externím zařízení, díky čemu by se zmenšila výpočetní náročnost, kterou musí Raspberry Pi vykonávat.

7.2 Hlava manipulátoru

Aktuální model umožňuje pouze manuální výměnu hlavy, avšak jsou zde před připravené otvory pro osm pogo pinů, díky kterým by si manipulátor v případě potřeby mohl automaticky vyměnit hlavu, ve které by mohl mít elektroniku pro sběr plodu, trhání plevelu nebo hnojení. V poslední části ramene je umístěný AD převodník se čtyřmi vstupy pro analogovou hodnotu, díky čemu by manipulátor mohl snímat ještě další 3 fyzikální veličiny. V úvahu přichází snímání teploty, vlhkosti nebo tlaku. Díky těmto údajům by bylo možné upravovat podmínky ve skleníku a docílit tak větší efektivity při samotném pěstování s minimální potřebou lidského zásahu.

7.3 Konstrukce

Pojezd je navržen tak, aby zvládl i malé zatačky, díky čemu by bylo možné, aby nám manipulátor kontroloval i záhony nedaleko skleníku. Samotné tělo manipulátoru bychom chtěli vyztužit hliníkovými profily, abychom získali větší stabilitu a mohli tak lépe vykonávat potřebné úkony. K základně bychom chtěli přidat plochu pro výměnu nástrojů a přepravku na sběr plodů.

8 Zhodnocení

Výsledný manipulátor umožňuje zefektivnění lidské práce potřebné na úspěšné vypěstování rostlin ve skleníku a zároveň minimalizuje zalévanou plochu, čímž se snižuje spotřeba vody. Umožňuje ovládání prostřednictvím jakéhokoli zařízení připojeného ke stejné síti pomocí HMI.

8.1 Parametry

Celý systém je napájen 24 V a maximálním proudem 3,5 A. Logická část je řízena 5 V logikou. V případě, že se nevykonávají žádné mechanické procesy má celý systém proudový odběr 0,15 A. Manipulátor má dosah 500 mm od středu osy otáčení. V případě nutnosti je možné tuto vzdálenost zvětšit pomocí prodloužení posledního článku manipulátoru. Pro tuto operaci stačí vytisknout na 3D tiskárně dva prodloužené díly a prodloužit kabeláž do hlavy manipulátoru. Výška manipulátoru závisí na délce hlazených tyčí a kuličkové tyče. V našem návrhu je maximální výška 500 mm. Délka kolejnic je závislá pouze na délce ozubeného řemenu. Spojovací podpěry pro kolejnice je dobré dávat v maximálním rozestupu jednoho metru, aby se kolejnice neprohýbaly. Měřený průtok čerpadla činí přibližně 100 ml za minutu.



Obr. 8.1: Celý systém

Závěr

V této bakalářské práci jsem se věnoval návrhu robotického manipulátoru typu SCARA, který má za úlohu měřit vlhkost půdy a následně na základě získaných údajů zalévat.

První část práce se věnuje porovnávání již existujících robotů používaných na pěstování rostlin. Hlavní rozdíl mezi již existujícími roboty byl v přístupu k jednotlivým rostlinám. Jedni přistupují k rostlinám z vrchu, další ze strany pomocí SCARA manipulátoru, nebo pomocí robotické ruky.

V druhé části jsem se věnoval samotnému návrhu SCARA robota, který byl dle průzkumu nejvhodnější, vzhledem k tomu, že má přístup ke kořenům rostlin, má jednoduché ovládaní a je možné ho z větší části vytisknout pomocí 3D tisku, což má za následek snížení výsledné ceny. Návrh jsem vytvořil v programu Siemens NX. Vytvořil jsem virtuální dvojče, pomocí kterého jsem se snažil eliminovat co nejvíce chyb. Pomocí virtuálního dvojčete jsem také otestoval základ celého programu. Robot se skládá ze čtyř pohyblivých os, které mu umožňují přístup k jednotlivým rostlinám z uličky skleníku. Pro zlepšení přesnosti robota jsou hlavní osy otáčení zpřevodovány pomocí řemenu a ozubených kol.

Třetí část byla věnována návrhu elektroniky. Pro řízení jsem použil mikrokontroler Raspberry Pi 4 model B, a k němuž jsem pomocí SPI a I2C komunikací připojil jednotlivé moduly. Pro ovládaní krokových motorů jsem si navrhl vlastní plošný spoj, který komunikuje prostřednictvím SPI komunikace s GPIO expandérem a následně s DRV8825 moduly. Na I2C sběrnici jsou připojeny senzory.

V další části jsem se věnoval PLC programu navrhnutému pomocí programu Codesys v jazyce ST. Bylo navrženo základní ovládaní robota na pozici, funkce na zjištění nulové pozice robota, funkce na přesun robota na základní pozici, a auto mód, který zalévá všechny rostliny na základě daného programu. Pro kontrolu pozice robota jsou na HMI zasílány ukazatele na vzdálenost hlavy od povrchu a ukazatele sepnutých koncových snímačů. Nasnímaná a standardizovaná hodnota vlhkosti půdy je zobrazována pomocí posuvníku. Pro ověřování funkčnosti a ladění jsem vytvořil obrazovku pro manuální ovládaní jednotlivých výstupů. Obě obrazovky jsou přístupné z jakéhokoliv zařízení na stejné síti.

Závěr práce se věnuje samotné realizaci manipulátoru a jsou zde navrženy plány pro budoucí vývoj zařízení.

Dalším krokem je umístit zařízení do skleníku, a prověřit jeho funkčnost a spolehlivost v reálném provozu.

Literatura

- [1] Ada, L.: Adafruit DRV8871 Brushed DC Motor Driver Breakout. In *Adafruit learning system*, 2018, str. 20.
Dostupné z: https://media.distrelec.com/Web/Downloads/_t/ds/Adafruit_DRV8871_eng_tds.pdf
- [2] Aronson, R.: Farmbot. 2022.
Dostupné z: <https://genesis.farm.bot/v1.6/assembly/intro>
- [3] ATHANI, V.: *STEPPER MOTORS*. Ansari Road, Daryaganj: New Age International, reprint:2005 vydání, 1997, ISBN 81-224-1006-5.
Dostupné z: https://books.google.sk/books?hl=cs&lr=&id=Om8NTozFZL8C&oi=fnd&pg=PA1&dq=stepper+motor&ots=MFeXLnpNA5&sig=f3Hxvz-JwZ4a6TMiy8p9FbSR90g&redir_esc=y#v=onepage&q&f=false
- [4] Dejan: SCARA Robot. 2020.
Dostupné z: <https://howtomechatronics.com/projects/scara-robot-how-to-build-your-own-arduino-based-robot/>
- [5] Duckett, T.; and Bruce Grieve, S. P.; Blackmore, S.: Agricultural Robotics: The Future of Robotic Agriculture. In *UK-RAS*, UK-RAS: UK-RAS White papers, 2018, ISSN 2398-4414, str. 36.
Dostupné z: <https://arxiv.org/ftp/arxiv/papers/1806/1806.06762.pdf>
- [6] Elektronics, P. R. .: DRV8825 Stepper Motor Driver Carrier, specification.
Dostupné z: <https://www.tme.eu/Document/1dd18faf1196df48619105e397146fdf/POL0LU-2133.pdf>
- [7] Ing. Jan Pásek, C.: Programovatelné automaty v řízení technologických procesů. 30.11.2007.
Dostupné z: [Vyukovytext](#)
- [8] Kristiawan, R. B.; Imaduddin, F.; Ariawan, D.; aj.: A review on the fused deposition modeling (FDM) 3D printing: Filament processing, materials, and printing parameters. *Open Engineering*, ročník 11, è. 1, 2021: s. 639–649, doi: doi:10.1515/eng-2021-0063.
Dostupné z: <https://doi.org/10.1515/eng-2021-0063>
- [9] Laki, S.; Stoyanov, R.; Kis, D.; aj.: P4Pi: P4 on Raspberry Pi for Networking Education. *SIGCOMM Comput. Commun. Rev.*, ročník 51, è. 3, jul 2021: str. 17–21, ISSN 0146-4833, doi:10.1145/3477482.3477486.
Dostupné z: <https://doi.org/10.1145/3477482.3477486>

- [10] Leens, F.: An introduction to I2C and SPI protocols. *IEEE Instrumentation & Measurement Magazine*, ročník 12, è. 1, 2009: s. 8–13, doi:10.1109/MIM.2009.4762946.
- [11] Molajou, A.; Afshar, A.; Khosravi, M.; aj.: A new paradigm of water, food, and energy nexus. *Environmental Science and Pollution Research*, ročník 2021, 2021: str. 11, ISSN 0944-1344, doi:10.1007/s11356-021-13034-1.
Dostupné z: <http://link.springer.com/10.1007/s11356-021-13034-1>
- [12] Morgan, E. J.: HCSR04 Ultrasonic Sensor. 2014.
Dostupné z: <https://datasheetspdf.com/pdf-file/1380136/ETC/HC-SR04/1>
- [13] Nelson, J.: Root-AI. 2020.
Dostupné z: <https://root-ai.com/#tech>
- [14] Ripka, P.: *Senzory a převodníky*. Praha: Vydavatelství ČVUT, 2005, ISBN 80-010-3123-3.
- [15] Sadchikova, G. M.: Application of NX Siemens PLM software in educational process in preparing students of engineering branch. *AIP Conference Proceedings*, ročník 1797, è. 1, 2017: str. 030015, doi:10.1063/1.4972454.
Dostupné z: <https://aip.scitation.org/doi/abs/10.1063/1.4972454>
- [16] Said Mohamed, E.; Belal, A.; Kotb Abd-Elmabod, S.; aj.: Smart farming for improving agricultural management. *The Egyptian Journal of Remote Sensing and Space Science*, ročník 24, è. 3, Part 2, 2021: s. 971–981, ISSN 1110-9823, doi:<https://doi.org/10.1016/j.ejrs.2021.08.007>.
Dostupné z: <https://www.sciencedirect.com/science/article/pii/S1110982321000582>
- [17] Sharma, K.: Certhon harvest robot. 2022.
Dostupné z: <https://certhonharvestrobot.com/>
- [18] Singh, M.; Fuenmayor, E.; Hinchy, E. P.; aj.: Digital Twin: Origin to Future. *Applied System Innovation*, ročník 4, è. 2, 2021, ISSN 2571-5577, doi:10.3390/asi4020036.
Dostupné z: <https://www.mdpi.com/2571-5577/4/2/36>
- [19] Spong, M. W.; Hutchinson, S.; Vidyasagar, M.: *Robot Dynamics and Control*. John Wiley and Sons, february vydání, 2020, ISBN 978-1-119-52404-5.

Seznam symbolů a zkratek

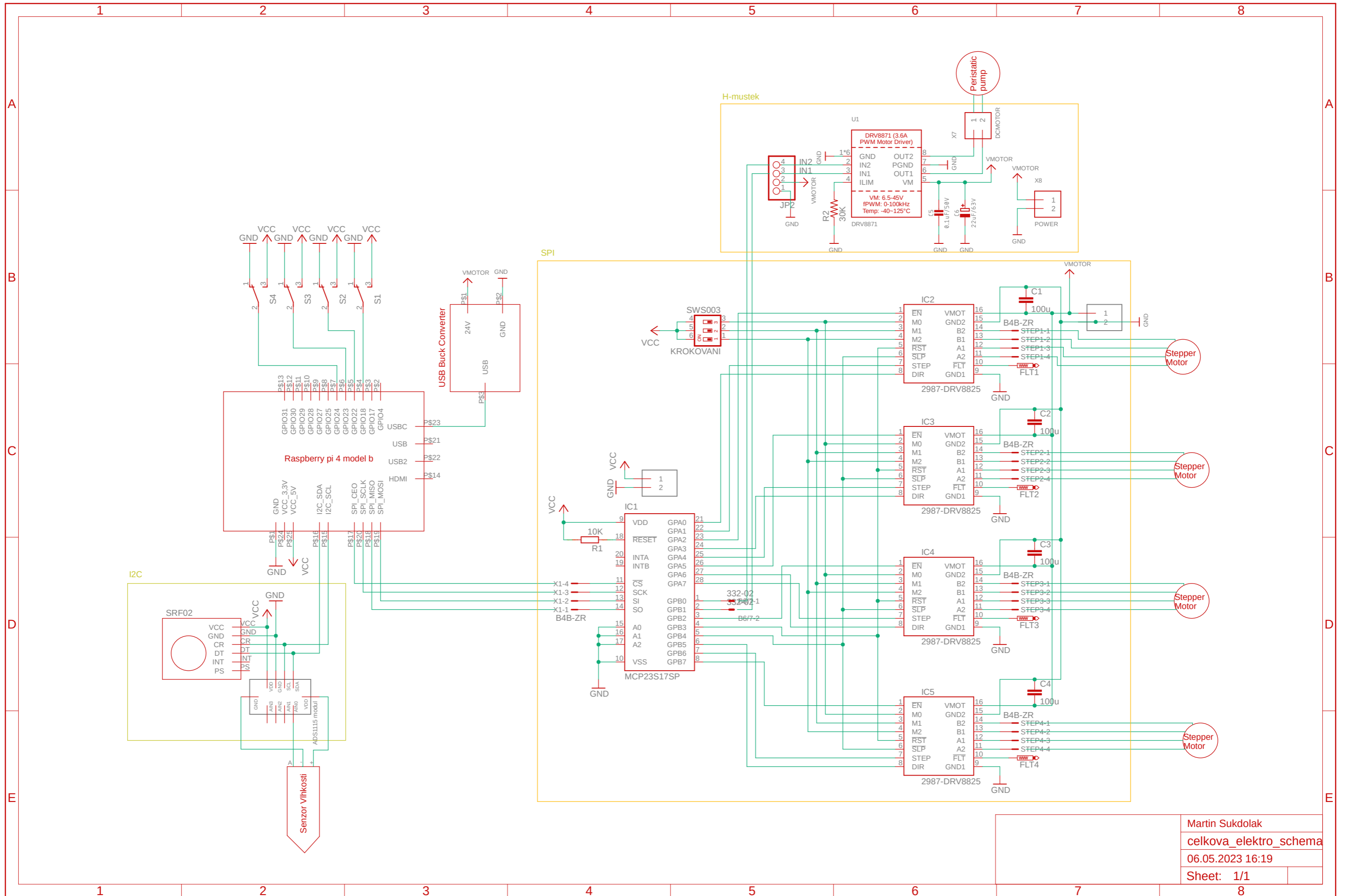
| | |
|-----------------|--|
| CAD | Computer Aided Design |
| CAM | Computer aided manufacturing |
| PLC | Programovatelný logický automat |
| Soft PLC | Softwarový Programovatelný logický automat |
| MCD | Mechatronic concept designer |
| HMI | Human Machine Interface |
| ST | Structured Text |
| FDM | Fused Deposition Modeling |
| AD | Analog to digital |
| PWM | Pulzně šířková modulace |
| GPIO | Vstupno/výstupní piny |
| V | Volty |
| A | Ampéry |
| ml | Mililitry |
| kcal | Kilokalorie |

Seznam příloh

| | |
|--------------------------------------|-----------|
| A Schéma zapojení elektroniky | 47 |
| B Návod | 49 |
| C Obsah elektronické přílohy | 57 |

A Schéma zapojení elektroniky

Schéma obsahuje 3 základní bloky elektroniky na základě komunikačního rozhraní mezi jednotlivými částmi.



Martin Sukdolak
 celkova_elektro_schema
 06.05.2023 16:19
 Sheet: 1/1

B Návod

Stručný návod pro stavbu, základní instalace softwaru a ovládaní prostřednictvím HMI.



**FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH **ústav automatizace**
TECHNOLOGIÍ **a měřicí techniky****

Manipulátor pro skleníkový systém

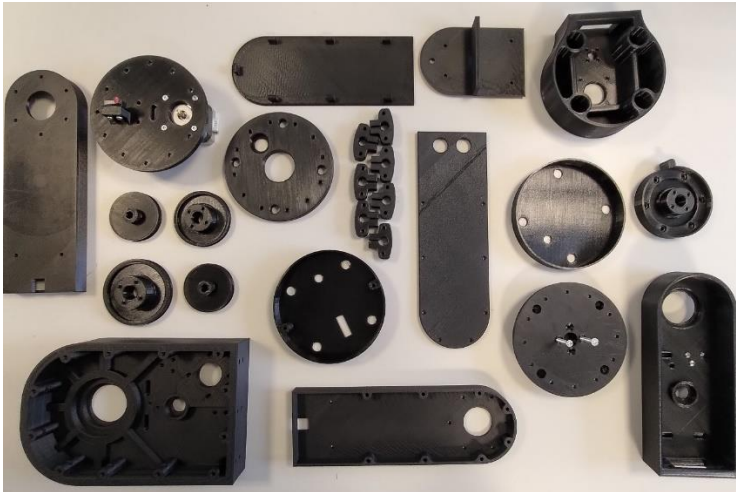
Návod na instalaci a obsluhu manipulátoru

VYPRACOVAL: Martin Sukdolák

DATUM: 12.10.2022

Sestava manipulátoru

1. Vytisknete všechny potřebné díly pomoci 3D tiskárny



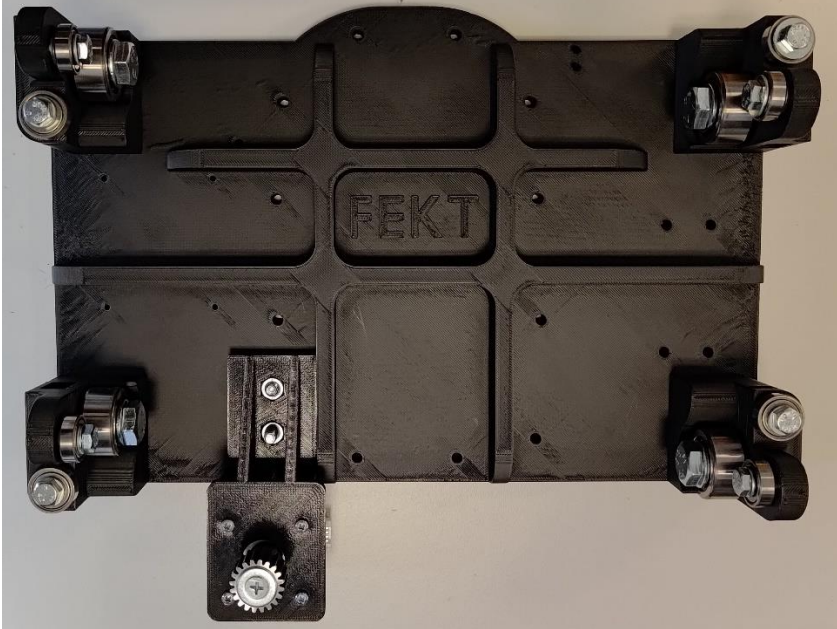
2. Sestavte rameno manipulátoru spolu s krokovým motorem
3. Napněte všechny řemeny a otestujte pohyblivost
4. Sestavte základnu manipulátoru spolu s krokovým motorem
5. Umístěte mikro switch a natáhněte si správnou délku kabelu
6. Napněte všechny řemeny a otestujte pohyblivost
7. Připevněte spodní část osy zabezpečující posun nahoru a dolu (nezapomeňte na krytku)
8. Umístěte kuličkovou tyč do ramena manipulátoru
9. Připevněte rameno k základně manipulátoru
10. Připravte si vrchní část manipulátoru včetně mikro switche s odpovídající délkou kabelu
11. Upevněte vrchní část manipulátoru a připevněte krokový motor ke kuličkové tyči. (nezapomeňte na krytku)



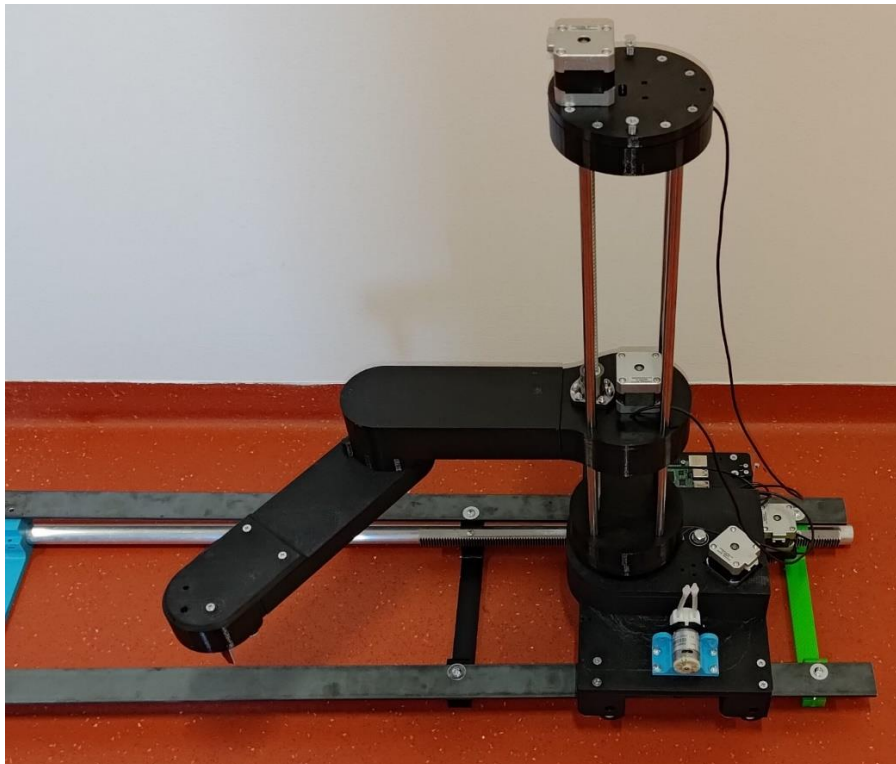
12. Zkontrolujte upevnění všech šroubů.

Sestava kolejnic

1. Vytisknete si potřebné díly pro kolejnice
2. Smontujte kolejnice a umístěte je do skleníku, tak aby na každé straně měli ještě aspoň 5 cm prostor.
3. Smontujte pojezd manipulátoru

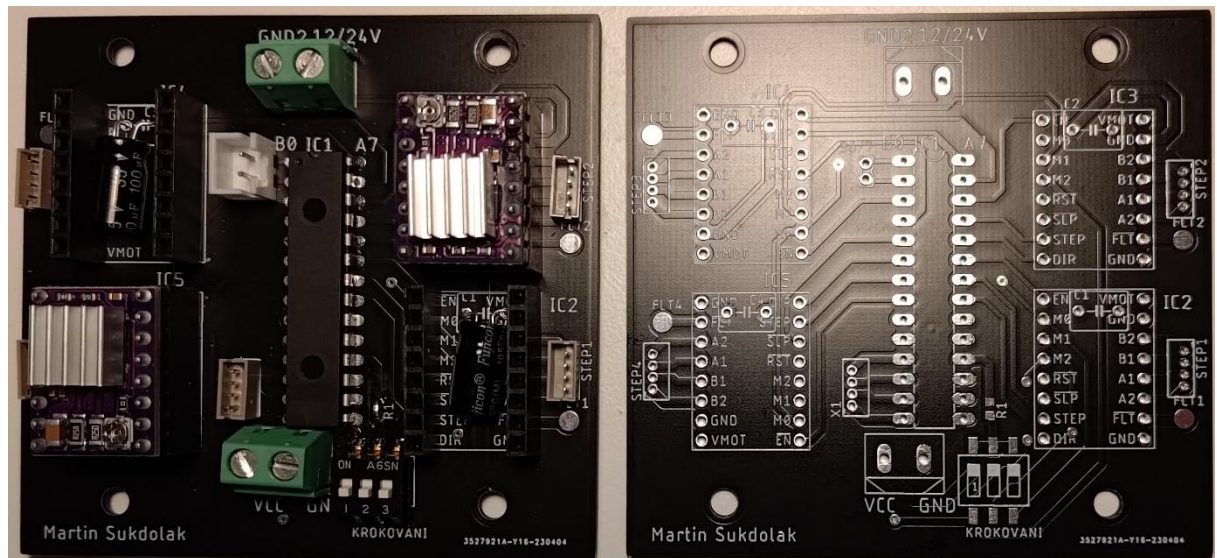


4. Nastavte vůle ložisek tak aby motor dokázal jezdit po ozubeném hřebenu na cele délce kolejnic.
5. Připevněte pojezd k manipulátoru
Doporučení: přichyťte si matky uvnitř manipulátoru pomocí lepicí pásky.



Zprovoznění elektroniky

1. Napájejte na desku plošného spoje potřebné součástky.



2. Vytvořte si společný kabel vedoucí z Raspberry pi dle schématu
3. Pevně uchyťte všechnu elektroniku
4. Zapojte všechnu elektroniku dle schématu
5. Svažte dohromady všechny kabely, tak aby nemohlo dojít k zapletení do manipulátoru.

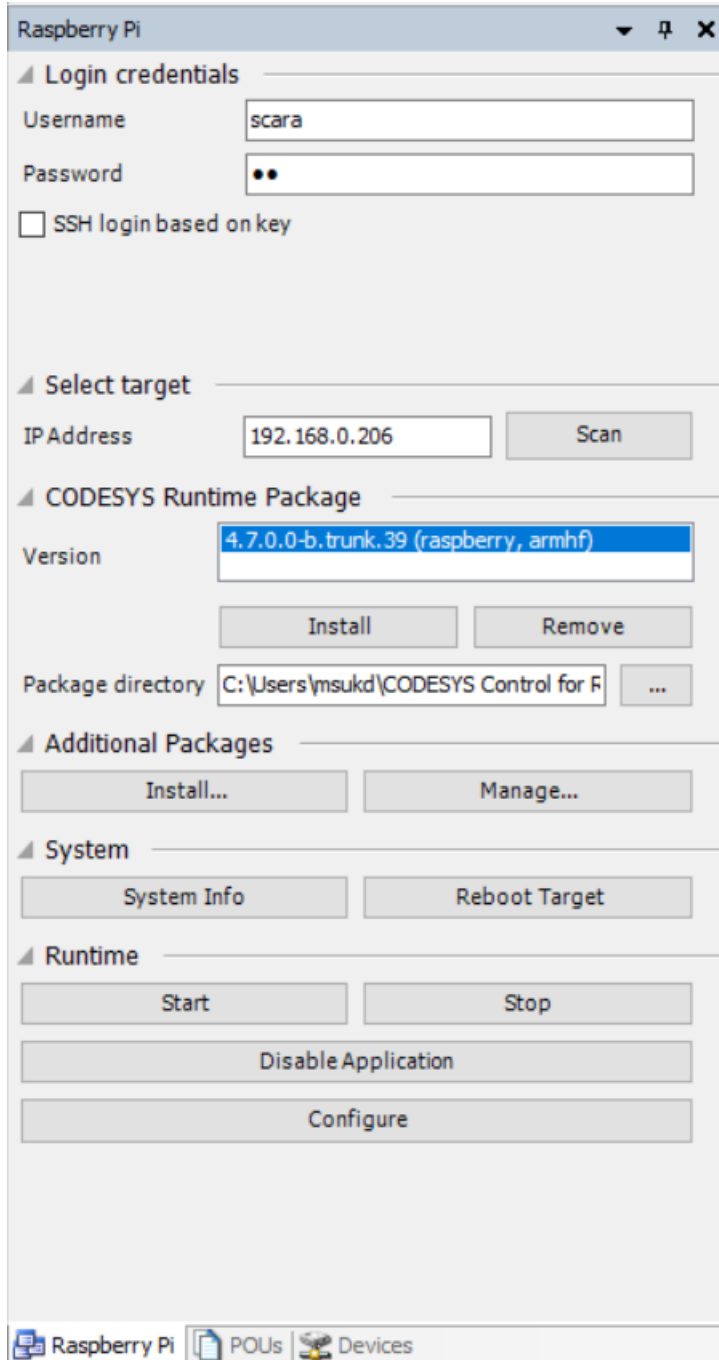
Zprovoznění Raspberry pi

1. Nahrajte do Raspberry pi oprácní systém Raspbian
2. Nastavte připojení na wifi a SSH cestu
3. Povolte v Raspberry SPI a I2C komunikace (pro případné další upgrady je možné povolit i kameru a 1-wire)
4. Vyzkoušejte spojení pomocí síťového spojení (SSH) a zjistěte IP adresu na dane síti.

Nahrání programu

1. Zapnete Raspberry pi
2. Vyzkoušejte pomocí příkazu ping <IP_adresa> v příkazovém řádku spojení s Raspberry pi
3. V prostředí Codesys s rozšířením pro Raspberry pi v záložce tool-> Update Raspberry pi otevřete nastavení Raspberry pi

4. Nastavte SSH loginy a pomoci tlačítka Scan vyhledejte vaše Raspberry pi



5. Zkontrolujte pomoci tlačítka Systém Info zda se jedná o správné Raspberry pi
6. Pomoci tlačítka Install nainstalujte daný Package do Raspberry pi
7. Následně se již Raspberry pi chová jako soft PLC a je možné do něj nahrát program

Obsluha manipulátoru

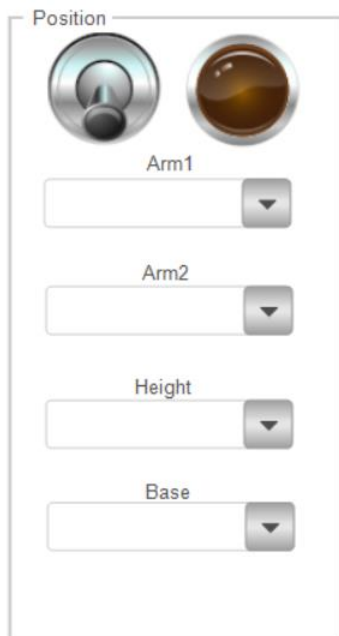
Po nahrání programu je možné manipulátor ovládat prostřednictvím HMI dostupného prostřednictvím webového prohlížeče na adrese:

[http:// <IP_adresa>:8080/webvisu.htm](http://<IP_adresa>:8080/webvisu.htm)

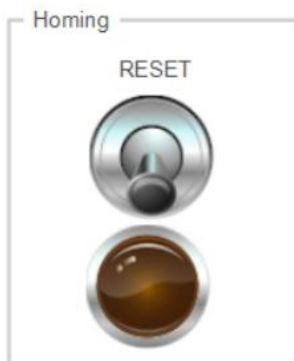
Ovládaní prostřednictvím HMI

1.1 Hlavní obrazovka

Nastavíme konkrétní číselnou hodnotu do poliček pro danou osu a následně zapneme spínač čím uvedeme manipulátor do pohybu na danou pozici.



Po každém vypnutí je potřebné, aby manipulátor zjistil v jaké pozici se nachází, což se vykoná na zaklade zapnutí spínače Homing nebo po zapnutí automatického modu

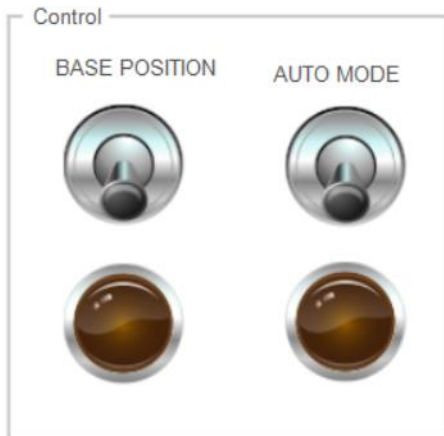


Tlačítko, které vše zastaví

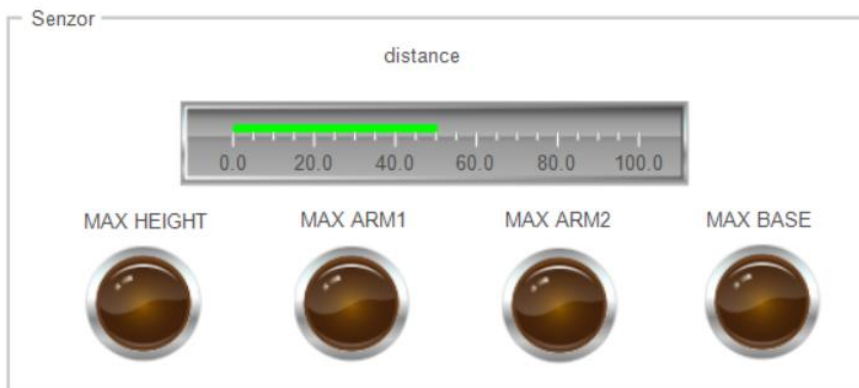


Tlačítko Base position nastaví manipulátor do základní pozice která se nachází na začátku skleníku v minimálním namáhání všech os

Tlačítko auto mode zapne automatický program zalévání všech rostlin



Kontrola všech senzoru



2.1 Control obrazovka

Spínání jednotlivých výstupu GPIO expandérů



C Obsah elektronické přílohy

Elektronická příloha obsahuje kompletní model a simulaci pro program Siemens NX. Dvě verze PLC programu a to pro řízení manipulátoru spojeného se simulací modelu a řízení reálného manipulátoru. Poslední složka obsahuje podklady pro námi navržený plošný spoj.

```
/. . . . . kořenový adresář přiloženého archivu
├── SCARA Robot . . . . . soubory pro Siemens NX
│   ├── mechanik2 . . . . . hlavní soubor pro MCD
│   └── Codesys . . . . . PLC project v programu Codesys
│       ├── V1 . . . . . řízení modelu
│       └── V2 . . . . . řízení reálného manipulátoru
└── PCB . . . . . Eagle soubory plošného spoje
```