



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

IMPLEMENTACE SKLADOVACÍ ENTITY V S7-1200

IMPLEMENTATION OF STORAGE ENTITY IN S7-1200

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Jan Holba

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Michal Husák

BRNO 2023



Bakalářská práce

bakalářský studijní program **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

Student: Jan Holba

ID: 228695

Ročník: 3

Akademický rok: 2022/23

NÁZEV TÉMATU:

Implementace skladovací entity v S7-1200

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je navrhnout entitu skladového systému pro inteligentní dávkovač tekutin malého objemu. Po návrhu entitu implementovat do PLC a vytvořit vhodnou vizualizaci pro HMI.

1. Seznamte se s koncepcí a aktuálním stavem testbedu.
2. Proveďte nutné úpravy hardwaru autonomní skladové buňky v kontextu budoucího automatického řízení.
3. Vytvořte rozšířitelnou a modulární řídicí logiku skladové entity v PLC dle standardů ISA S88 a S95.
4. Realizujte rozhraní člověk stroj pro provoz i údržbu skladové buňky. Zpracujte stručnou provozní a servisní dokumentaci.
5. Otestujte své řešení a popište dosažené výsledky.

DOPORUČENÁ LITERATURA:

[1] ZEZULKA, František, Jan PÁSEK, Milan FINDURA a Jaroslav PREČAN. Automatizace procesů I: Úroveň procesního řízení [online]. Brno, 2014

[2] HUSÁK, M. Návrh a realizace laboratorní úlohy na téma "Standard ISAS88". Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2018

Termín zadání: 6.2.2023

Termín odevzdání: 22.5.2023

Vedoucí práce: Ing. Michal Husák

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 částí druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Tato práce se zabývá implementací skladovací entity pro skladovou jednotku tekutin malého objemu. Modifikuje stávající konstrukci existující skladové jednotky a její způsob dávkování tekutin. Tím řeší mnoho existujících problémů a zajišťuje vyšší spolehlivost jednotky. Dále charakterizuje principy Průmyslu 4.0, koncepci testbedu Self-Acting Barman a standard ANSI/ISA-S88, na jehož základě je jednotka skladu, stejně tak její řídicí logika, vytvořena. Současně popisuje vznik komplexního digitálního dvojčete a jeho virtuálního zprovoznění v aplikaci MCD programu Siemens NX. Nechybí kompletní elektrické schéma zapojení jednotky a provozní manuál pro obsluhu skladové jednotky dotykovým operátorským panelem.

Klíčová slova

Průmysl 4.0, Dávkový proces, Digitální/Virtuální dvojče, Virtuální zprovoznění, Siemens NX, S7-1200, Testbed, Mechatronics Concept Designer, MCD, ANSI/ISA S88

Abstract

This thesis deals with implementation of storage entity for small-volume liquid storage unit. It modifies current construction of existing storage cell and its method of liquid dosage, addressing numerous issues and ensuring greater unit reliability. Furthermore, it provides a characterization of the fundamental principles of Industry 4.0, the Self-Acting Barman testbed concept, and the ANSI/ISA-S88 standard. Both the storage unit and its control logic are based on this standard. Creation of a complex digital twin and its virtual commissioning in the MCD application of the Siemens NX program is described. Complete electrical schematics of the unit's wiring and as well as an operational manual for unit's touch panel are also provided.

Keywords

Industry 4.0, Batch process, Digital/Virtual twin, Virtual commissioning, Siemens NX, S7-1200, Testbed, Mechatronics Concept Designer, MCD, ANSI/ISA S88

Bibliografická citace

HOLBA, Jan. Implementace skladovací entity v S7-1200. Brno, 2023. Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/151797>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Michal Husák.

Prohlášení autora o původnosti díla

Jméno a příjmení studenta: *Jan Holba*

VUT ID studenta: *228695*

Typ práce: *Bakalářská práce*

Akademický rok: *2022/23*

Téma závěrečné práce: *IMPLEMENTACE SKLADOVACÍ
ENTITY V S7-1200*

Prohlašuji, že svou bakalářskou 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 bakalářské 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 jsem si 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.

V Brně dne: 15. května 2023

podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce Ing. Michalu Husákovi a dalším členům Skupiny průmyslové automatizace za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne: 15. května 2023

podpis autora

Obsah

SEZNAM OBRÁZKŮ	8
1. PRŮMYSL 4.0.....	11
1.1 PRINCIPY PRŮMYSLU 4.0.....	11
1.1.1 Decentralizace	11
1.1.2 Virtualizace.....	12
1.1.3 Modularita a Rekonfigurabilita	12
1.1.4 Interoperabilita.....	13
1.2 PROSTŘEDKY PRŮMYSLU 4.0	13
1.2.1 Digitální dvojče.....	13
1.2.2 Kyberneticko-fyzikální systém (CPS – Cyber-Physical system).....	14
1.2.3 Internet věcí (IoT – Internet of Things).....	14
1.2.4 Internet služeb (IoS – Internet of Services).....	14
2. TESTBED	15
2.1 TESTBED SELF-ACTING BARMAN.....	15
2.1.1 Konstrukce testbedu	16
3. STANDARD ANSI/ISA S88	18
3.1 MODEL STANDARDU S88	18
3.1.1 Procesní model.....	19
3.1.2 Fyzický model.....	20
3.1.3 Model procedurálního řízení	21
3.1.4 Propojení modelů.....	22
4. SKLADOVÁ JEDNOTKA TEKUTIN MALÉHO OBJEMU	23
4.1 PŮVODNÍ KONSTRUKCE	23
4.2 PROBLÉMY S PŮVODNÍM ŘEŠENÍM.....	24
5. MODIFIKACE INTELIGENTNÍ SKLADOVÉ JEDNOTKY	25
5.1 PERISTALTICKÁ ČERPADLA	25
5.2 NOVÝ TYP SKLADOVÉ POZICE.....	26
5.3 ÚPRAVA KONSTRUKCE.....	27
5.3.1 Opravy původního modelu	28
5.4 FINÁLNÍ KONSTRUKCE BUŇKY	30
5.5 ELEKTRICKÉ ZAPOJENÍ	31
5.5.1 Zapojení reléového modulu pro ovládání peristaltických čerpadel	32
6. VÝVOJ DIGITÁLNÍHO DVOJČETE.....	33
6.1 KOMUNIKACE DIGITÁLNÍHO DVOJČETE S PLC	35
7. PRAKTICKÉ VYUŽITÍ STANDARDU ANSI/ISA-S88	37
7.1 PROCESNÍ MODEL SKLADOVÉ JEDNOTKY	37
7.2 FYZICKÝ MODEL SKLADOVÉ JEDNOTKY	38
7.3 MODEL PROCEDURÁLNÍHO ŘÍZENÍ SKLADOVÉ JEDNOTKY	38
8. ŘÍDICÍ LOGIKA	40

8.1	PROVOZNÍ REŽIMY	40
8.2	ČLENĚNÍ PROGRAMU	41
8.3	FUNKČNÍ BLOKY PRO ŘÍZENÍ MODULŮ ZAŘÍZENÍ	42
8.3.1	<i>AxisControl</i>	42
8.3.2	<i>Manipulator</i>	44
8.3.3	<i>PumpControl</i>	44
8.4	FUNKČNÍ BLOKY PODLE STANDARDU ANSI/ISA-S88	46
8.4.1	<i>Fáze transportu – Phase_Transport</i>	46
8.4.2	<i>Fáze čerpání – Phase_Filling</i>	48
8.4.3	<i>Fáze nabrání skleničky – Phase_Load</i>	48
8.4.4	<i>Fáze vyložení skleničky – Phase_Unload</i>	48
8.4.5	<i>Operace přípravy nápoje – Oper_DrinkPrep</i>	48
8.4.6	<i>Procedura přípravy nápoje</i>	50
8.5	POMOCNÉ FUNKCE	50
8.5.1	<i>RotToLin</i>	50
8.5.2	<i>Ostatní</i>	52
8.6	KOMUNIKACE.....	52
8.6.1	<i>Modbus</i>	52
8.6.2	<i>IO-Link</i>	53
8.6.3	<i>Komunikace s nadřazeným systémem</i>	54
8.7	SIMULAČNÍ REŽIMY SKLADOVÉ JEDNOTKY.....	55
8.8	DIGITÁLNÍ DVOJČE A REÁLNÉ NAsAZENÍ	55
9.	OPERÁTORSKÝ PANEL	56
10.	ZÁVĚR	58
	LITERATURA	59
	SEZNAM PŘÍLOH	61

SEZNAM OBRÁZKŮ

Obr. 1.1:	Struktura řídicího systému výrobního podniku [2].....	11
Obr. 1.2:	Industry 4.0 komponenta	12
Obr. 2.1:	Testbed Self-Acting Barman [4].....	15
Obr. 3.1:	Procesní model [10] inspirováno	19
Obr. 3.2:	Fyzický model [6] přeloženo	20
Obr. 3.3:	Model procedurálního řízení [6] přeloženo	21
Obr. 3.4:	Propojení modelů [7] upraveno	22
Obr. 4.1:	Model původní buňky skladu a detail stlačovacího dávkovače [8]	23
Obr. 5.1:	Konstrukce peristaltického čerpadla [11]	25
Obr. 5.2:	Nový typ skladové pozice.....	26
Obr. 5.3:	Původní pouzdro protizávaží a uchycení vodicí tyče.....	27
Obr. 5.4:	Nové pouzdro protizávaží.....	27
Obr. 5.5:	Zábrany proti kolizi	28
Obr. 5.6:	Díly vytvořené na základě fyzické předlohy	29
Obr. 5.7:	Finální stav buňky	30
Obr. 5.8:	Původní elektrické schéma zapojení [8]	31
Obr. 5.9:	Schéma zapojení reléového modulu	32

Obr. 6.1: MCD – Možnosti spojů	33
Obr. 6.2: MCD – Možnosti vazebních členů	34
Obr. 6.3: MCD – Ozubená kola	34
Obr. 6.4: MCD – Řízení rychlosti.....	35
Obr. 6.5: MCD – Blok výrazů pro řízení motoru v ose X.....	36
Obr. 7.1: Procesní model skladové jednotky	37
Obr. 7.2: Fyzický model skladové jednotky	38
Obr. 7.3: Model procedurálního řízení skladové jednotky.....	39
Obr. 8.1: UDT OperationDrinkPrep	40
Obr. 8.2: Stavový automat provozních režimů	41
Obr. 8.3: Struktura programu.....	41
Obr. 8.4: UDT Engine.....	42
Obr. 8.5: Struktura řízení reálného motoru	43
Obr. 8.6: Stavový automat řízení motorů.....	43
Obr. 8.7: Bezpečná obsluha čerpadel.....	44
Obr. 8.8: Stavový automat řízení čerpadel.....	45
Obr. 8.9: UDT Pumps	45
Obr. 8.10: UDT fáze transportu	46
Obr. 8.11: Transport do pozice na okraji manipulačního prostoru	46
Obr. 8.12: Stavový automat fáze transportu	47
Obr. 8.13: Zjednodušený stavový automat fáze transportu.....	47
Obr. 8.14: Manipulátor v pozici vyložení	48
Obr. 8.15: Stavový automat operace přípravy nápoje.....	49
Obr. 8.16: Stavový automat plnění skleničky	49
Obr. 8.17: Diagram pro výpočet transformace pohybů.....	50
Obr. 8.18: Aplikace pohybové transformace	51
Obr. 8.19: Stavový automat komunikace Modbus	52
Obr. 8.20: Funkční blok Modbus	53
Obr. 8.21: Komunikační rámec pro zápis do relé modulu	53
Obr. 8.22: Připojení IO-Link modulů	54
Obr. 8.23: Datový blok pro komunikaci s nadřazeným systémem	54
Obr. 9.1: Úvodní obrazovka HMI.....	56
Obr. 9.2: Struktura obrazovek HMI	57

Úvod

Tato bakalářská práce se zabývá modifikací jednotky pro skladování a dávkování tekutin malého objemu. Navazuje tak na diplomovou práci Ing. Lukáše Rejchlíka [1].

Cílem mé práce je návrh skladového systému, který řeší problémy spojené s původní koncepcí. Skladová jednotka bude po dokončení zahrnuta do projektu testbed Průmyslu 4.0 – Self-Acting Barman. Tento testbed reprezentuje inteligentní výrobu založenou na principech Průmyslu 4.0, proto je nutné v první části textu tyto principy popsat a vysvětlit. V další části práce je popsán aktuální stav zmíněného testbedu společně s vizí, kterou by měl naplnit.

Výrobní proces testbedu spadá do kategorie dávkové výroby, která je popsána standardem ANSI/ISA-S88. Vzniklá implementace také respektuje požadavky stanovené tímto standardem, proto je vhodné jej vysvětlit. Popis standardu ANSI/ISA-S88 se nachází ve třetí kapitole.

Podstatná část práce je věnována vývoji digitálního dvojčete skladové jednotky. To je plně virtuálně zprovozněno v aplikaci Mechatronics Concept Designer programu Siemens NX. Digitální dvojče zajišťuje komplexní kinematickou simulaci skladové jednotky, která je řízena řídicím systémem PLC Siemens S7-1200, stejně jako její reálný protějšek. Jelikož je digitální dvojče velmi detailní, po vytvoření a odladění řídicí logiky, je její nasazení na reálnou jednotku velmi jednoduché s nutností téměř nulové modifikace. Řídicí logika vzniká ve vývojovém prostředí Siemens TIA Portal V17.

Dále je navržen a fyzicky implementován nový způsob dávkování tekutin peristaltickými čerpadly. Je popsána funkce tohoto typu čerpadla, společně s výhodami, které toto řešení přináší vůči původnímu způsobu dávkování stlačovacím dávkovačem. Konstrukce buňky je modifikována, aby bylo možné nový koncept dávkování realizovat.

Je vytvořeno kompletní elektrické schéma celé skladové jednotky, které respektuje standard DIN-81346. Nechybí kompletní provozní manuál pro správné řízení skladové jednotky, pomocí dotykového operátorského panelu.

1. PRŮMYSL 4.0

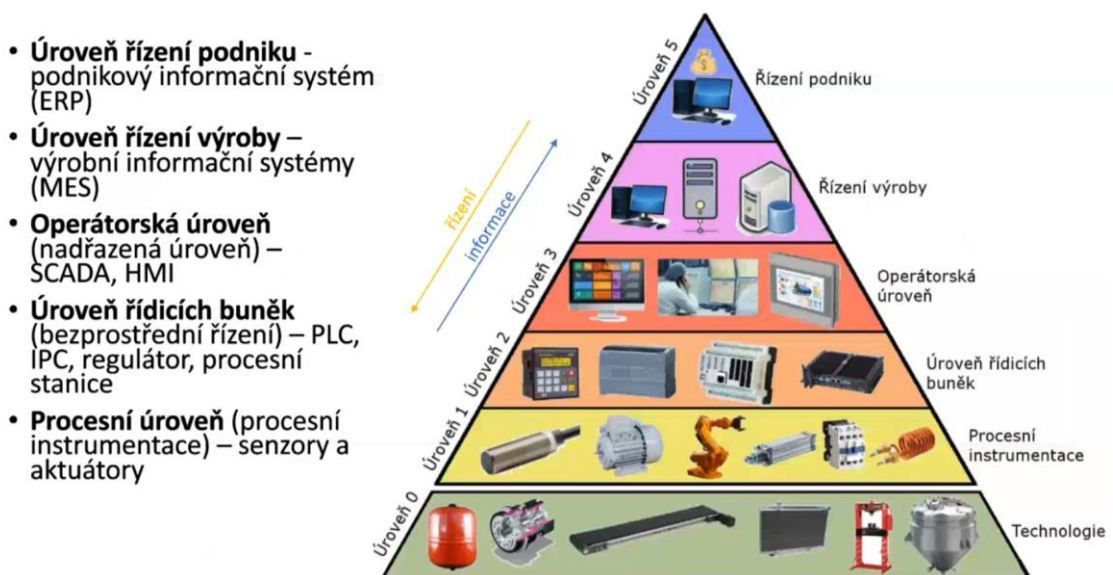
Souhrnné označení pro technologie a koncepty, jejichž cílem je, aby v rámci modulárně strukturovaných chytrých továren (Smart Factory) dokázal kyberneticko-fyzikální systém (Cyber-Physical system, dále jen CPS) monitorovat fyzické procesy, vytvářel virtuální kopii fyzického světa a činil decentralizovaná rozhodnutí. Přes IoT (Internet of Things) CPS autonomně komunikuje a spolupracuje s dalšími CPS v reálném čase a předává informace lidem. CPS nabízí své služby prostřednictvím IoS (Internet of Services). Výměna dat může být jak interní (v rámci jedné továrny), tak globální (mezi jednotlivými továrnami). Dojde tak ke kompletní vertikální i horizontální integraci dle topologie RAMI 4.0 (Reference Architecture Model Industrie 4.0). Rozšířením chytrých továren dojde k výraznému zvýšení efektivity ve výrobě, zásobování, zpracování objednávek a v mnoha dalších odvětvích, které byly dodnes závislé na lidském faktoru. To by mohlo mimo jiné vést ke snížení ceny výrobků. Výrobky budou jednoduše identifikovatelné, což umožní jejich detailní sledování po celou dobu výrobního procesu. [2] [3]

1.1 Principy Průmyslu 4.0

Myšlenka Průmyslu 4.0 je založena na několika hlavních principech. Těmi jsou decentralizace, virtualizace, interoperabilita, modularita a rekonfigurabilita.

1.1.1 Decentralizace

Aktuální způsob řízení výroby je převážně centralizovaný (centrálním systémem je ERP). Jeho struktura lze popsat pomocí následujícího pyramidového schématu (Obr. 1.1).

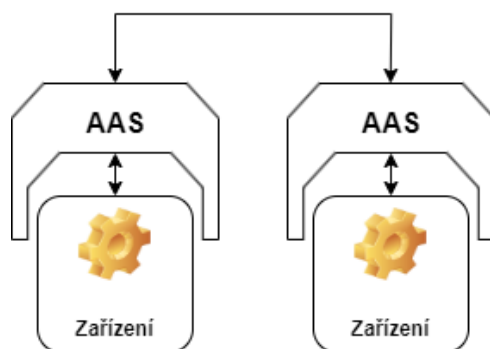


Obr. 1.1: Struktura řídicího systému výrobního podniku [2]

Cílem Průmyslu 4.0 je kompletní horizontální i vertikální integrace (propojení zařízení jak v jednotlivých vrstvách, tak i mezi vrstvami) a s tím spojená decentralizace řízení. Neexistuje tak centrální systém, který by kompletně řídil a plánoval celou výrobu, ale každé zařízení je opatřeno jednotkou pro zajištění komunikace a kompletního propojení s ostatními členy výroby. Inteligence ze systémů ERP (pro řízení podniku) a MES (pro řízení výroby) je přesunuta do jednotek na nižších úrovních. [2] [3]

Při výpadku jednoho systému tak může zbytek výroby pokračovat relativně bez omezení. Při centralizované výrobě by mohlo výpadkem hlavního systému dojít ke kolapsu celé výroby. Informace o výrobcích se také neuchovávají v centrální jednotce, ale každý výrobek si sám nese svoje požadavky a data spojená výrobou např. v RFID čipu.

Vznikají Industry 4.0 komponenty. Jednotlivé stroje, které znají své vlastnosti a schopnosti. Industry 4.0 komponenta nabízí okolním zařízením své služby, dokáže sama popotávat materiál či údržbu, tvoří výrobní frontu atd. bez nutnosti asistence nadřazených systémů a s tím spojeného zpoždění. Tyto funkce zajišťuje tzv. AAS (Asset Administration Shell), který tvoří digitální obálku daného zařízení. [5]



Obr. 1.2: Industry 4.0 komponenta

1.1.2 Virtualizace

Vytvoření kyberneticko-fyzikálního systému (viz kapitola 1.2.2) celé továrny.

1.1.3 Modularita a Rekonfigurabilita

Výrobní proces je rozdělen do jednotek, které jsou na sobě fyzicky nezávislé, tzn. jsou rozčleněny do samostatných buněk a nejsou nerozebíratelně spojeny. Často mají standardizované rozměry, aby bylo možné jednotky zaměňovat bez nutnosti zásahu do okolních jednotek.

Cílem je, aby všechny jednotky byly tzv. Plug&Produce (v doslovném překladu Zapoj a vyráběj). Po připojení nové jednotky do výrobního procesu by měla vyžadovat minimální nutnost konfigurace a měla by se dokázat relativně sama začlenit do procesu. [2] [4]

1.1.4 Interoperabilita

Všechny části výrobního procesu jsou schopny vzájemné komunikace a výměny dat. Aby toto bylo možné, je nutné, aby všechny jednotky dodržovaly jednotná syntaktická (formát přenášených dat) a sémantická (význam přenášených dat) pravidla. K tomuto účelu se výzkumné týmy, zabývající se průmyslem 4.0, snaží o standardizaci komunikace mezi I4.0 komponenty (M2M – Machine to Machine), která je nazývána I4.0 language. [4] [5]

1.2 Prostředky Průmyslu 4.0

Mezi prostředky, které jsou pro realizaci uvedených principů nezbytné, patří IoT (Internet of Things – Internet věcí), IoS (Internet of Services – Internet služeb) a CPS (Cyber-Physical systém – Kyberneticko-fyzikální systém), který souvisí s virtuálním dvojčtem.

1.2.1 Digitální dvojče

Digitální dvojče tvoří podstatnou část této práce, proto bude pojem vysvětlen detailně.

Digitální dvojče nebo DT je komplexní model reálného systému, který umožňuje jeho detailní simulaci. Díky digitálnímu dvojčeti můžeme systém jednoduše ovládat ve virtuálním prostředí. Tím jsou eliminovány problémy plynoucí z manipulace s fyzickým systémem v reálném světě. Digitální dvojče nemůže poškodit sebe ani okolí, nemůže se opotřebit či jinak znehodnotit.

V průběhu životního cyklu stroje hraje DT nezastupitelnou roli při vývojové fázi zařízení. Při tvorbě řídicí logiky pro daný systém programátor nemusí být v blízkosti reálného stroje, a dokonce daný stroj ještě nemusí fyzicky existovat. Řídicí logika může vznikat vzdáleně z pohodlí domova či kanceláře. Upadá stres a strach z poškození drahého zařízení a programátor se může plně soustředit na tvorbu řídicí logiky. Může libovolně experimentovat, simulaci může kdykoliv restartovat do libovolné pozice, či pozastavit pro debugování vznikajícího programu. To je u reálného stroje často nemožné. Po dokončení řídicí logiky pro digitální dvojče je potom potřeba minimálních modifikací programu, pro zprovoznění reálného systému.

Digitální dvojče je kromě vývoje a testování nových konceptů vhodné také pro zaškolení nového personálu, či pro výuku. Pracovníci se mohou se systémem do hloubky seznámit a mohou se naučit reakce v případě hazardních stavů. V průběhu školení reálný stroj vykonává svoji běžnou práci a výroba tak není nijak omezena. Ve výuce je vhodné využít digitální dvojče hlavně z finančních důvodů. Vytvořit a virtuálně zprovoznit virtuální dvojče je finančně mnohem méně náročné, než vytvořit fyzicky funkční stroj, na kterém se mohou studenti učit programovat. Efektivita učení, ať už ve škole či v průmyslu, se tak výrazně zvýší, protože upadá strach z možnosti poškození reálného, často drahého, stroje. Simulaci může používat několik lidí současně.

Digitálnímu dvojčeti můžeme nastavit rychlost vykonávání akcí. Můžeme ji také v určitém okamžiku zastavit a provést potřebné výpočty. Může tak běžet zpomaleně pro

detailní analýzu, nebo může zrychleně vykonat velké množství cyklů procesu pro měření a analýzu dlouhodobých dat bez nutnosti zdlouhavého měření reálného systému po dobu několika měsíců. Stejně tak lze velmi rychle otestovat obrovské množství různých modifikací procesu.

Na základě analýzy dat můžeme digitální dvojče velmi jednoduše modifikovat a optimalizovat. Při modifikaci konstrukce nespotřebujeme žádný materiál a optimalizace zařízení je tak výrazně levnější. Reálný systém se potom upravuje až po důkladném otestování na digitálním dvojčeti. [2] [4]

1.2.2 Kyberneticko-fyzikální systém (CPS – Cyber-Physical system)

Simulace digitálního dvojčete běží odděleně od reálného systému. Kompletním zpětnovazebním propojením digitálního dvojčete a fyzického systému vzniká tzv. Kyberneticko fyzikální systém (CPS). Digitální dvojče je neustále dynamicky aktualizováno na základě měřených veličin na reálném zařízení. Je tak zajištěno, že se reálný systém i jeho digitální dvojče pohybují vázaně a udržují si nulovou odchylku i v případě, kdy se reálný systém opozdí např. působením vnějších sil či opotřebením. Zprovoznění CPS je však výrazně složitější než DT. [2] [4]

1.2.3 Internet věcí (IoT – Internet of Things)

Veškerá data získaná během výroby jsou sbírána a po filtraci odesílána na cloud za účelem monitorování, analýzy, vzdálené modifikace a optimalizace výroby. Všechna zařízení jsou přes cloud propojena a vzájemně spolu komunikují.

Mimo jiné minimalizuje nutnost obsluhy být fyzicky uvnitř továrny. S IoT úzce souvisí FRID čipy zejména pro identifikaci výrobku a uchování informací o něm. [2] [4]

1.2.4 Internet služeb (IoS – Internet of Services)

Umožňuje aplikaci Architektury orientované na služby (SOA – Service Oriented Architecture). Jednotlivá zařízení znají své funkce, možnosti a kapacitu výroby. Na základě toho následně nabízí své služby okolním členům výroby. Výrobek si potom sám na základě přijatých nabídek vybere, které zařízení chce využít. Mezi rozhodovací parametry mohou patřit například rychlost, kvalita či „cena“ vykonané služby daným zařízením. Není tak potřeba centrální jednotku pro řízení výroby a přiřazování výrobků. [2] [4]

2. TESTBED

Testbed je platforma pro aplikování a ověřování nových konceptů s eliminací rizik spojených s testováním v reálném provozu. Umožňuje studování funkce komponentů a interakce mezi jednotlivými prototypy. Nabízí tedy možnost modelování reálných systémů v laboratorním prostředí a poskytuje realistické hardwarové a softwarové prostředky, na jejichž základě je možné lépe porozumět dané problematice a případně navrhnout úpravy.

Návrh testbedu můžeme rozdělit do tří fází:

- 1) Návrh a simulace prvotních konceptů
- 2) Sestavení reálných komponentů a nových prototypů, se kterými experimentujeme
- 3) Monitorování, měření, analýza a sběr dat

Testbed by měl být modulární a flexibilní jednotkou pro kompletní simulaci a možnost experimentování s novými prototypy a zároveň by měl umožňovat jednoduchou úpravu jednotlivých částí. [7]

2.1 Testbed Self-Acting Barman



Obr. 2.1: Testbed Self-Acting Barman [4]

Testbed vytvořený skupinou průmyslové automatizace nese název Self-Acting Barman. Cílem je obsažení celé výrobní pyramidy (viz kapitola 1.1) od procesní instrumentace až po systémy MES a ERP. Vizí tohoto testbedu je systém, který demonstruje základní principy Průmyslu 4.0 na praktické aplikaci.

Finální funkcionalita bude následující. Uživatel si přes webovou aplikaci zvolí požadovanou kombinaci nápojů. Objednávka je následně odeslána do systému ERP, který vytvoří pracovní požadavek a předá jej systému MES. Tento systém požadavek přijme a následně provede plánování výroby a na základě toho dynamicky vytváří výrobní frontu. Každý produkt je monitorován po celou dobu výroby od prvotního zadání uživatelem, až po vydání hotového výrobku uživateli. Receptura je společně s dalšími informacemi uložena v NFC čipu, který je součástí každé skleničky. Dokončený výrobek, tzn. naplněná sklenička se zvoleným nápojem, je následně manipulátorem umístěn na dopravník, který jej doveze až do bezpečné zóny k zákazníkovi. Zákazník následně sklenici odebere a po vyprázdnění ji může umístit zpět na dopravník, který ji za spolupráce manipulátoru umístí do zásobníku s použitými sklenicemi. [5]

2.1.1 Konstrukce testbedu

Základem testbedu je pracovní deska o rozměrech 2 000 x 1 000 mm. Na této desce je připevněno několik autonomních procesních jednotek, které budou detailně popsány v této kapitole. Pracovní deska i rámy všech jednotek jsou sestaveny z hliníkových profilů stavebnicového systému o rozměrech 30 x 30 mm. Těchto profilů je na trhu velké množství a umožňují velmi jednoduché sestavení potřebných konstrukcí.

Pro přemísťování sklenic mezi jednotlivými buňkami slouží SCARA manipulátor EPSON G55BN s chapadlem pro úchop sklenice. Jednotlivé buňky jsou k pracovní desce připevněny rozebíratelnými spoji, aby bylo zamezeno jejich posuvu. Manipulátor má tak naprogramované pevně dané pohybové trasy. Všechny buňky mají sjednocený tvar platformy pro příjem a výdej skleniček.

Dalším transportním prvkem je dopravník. Ten je poháněn krokovým motorem NEMA23. Jeho uzavřená oválná dráha je tvořena dvěma rovnými hliníkovými profily o rozměrech 20 x 20 mm a délce 500 mm a dvěma půlkulatými profily o poloměru 140 mm. Na dopravník je potom připevněno deset plastových vozíků pro uložení sklenic. Úkolem dopravníku je jednak dopravení naplněných sklenic do bezpečné zóny mimo manipulační prostor manipulátoru k uživateli a jednak odběr použitých sklenic od zákazníka.

Výrobní proces požadovaného nápoje má k dispozici následující buňky:

1) Sklad skleniček

- Pro skladování jednak čistých sklenic, které jsou připraveny k vydání manipulátoru a jednak použitých sklenic pro odebrání obsluhou.

2) Výrobník sycené vody

- Využívá stlačeného plynu CO₂, který uchovává v tlakové nádobě, k nasycení vody. To probíhá ve směšovacím kontejneru.

3) Sklad tekutin většího objemu

- Pro skladování např. džusů či limonád. Dávkování je možné v objemu 100–300 ml.

4) Sklad tekutin malého objemu

- Jednotka je předmětem této práce a je detailně popsána v kapitole 4.

5) Shaker

- Přímo na dodanou sklenici dosedne víko, které je silně upevněno a nápoj je promíchán. Pohonem je stejnosměrný motor s klikovou hřídelí, která převádí rotační pohyb na lineární, periodicky se opakující pohyb

6) Drtič ledu

- Led je drcen o mřížku pomocí rotujících nožů a padá přímo do sklenice. Aktuálně mimo provoz.

Jednotlivé buňky tvoří decentralizované kyberneticko-fyzikální systémy. Společně tvoří automatizovaný celek pro míchání a distribuci alkoholických a nealkoholických nápojů. Vzhledem k povaze finálního produktu, který je mimo jiné směsí několika ingrediencí, můžeme říct, že testbed spadá do kategorie dávkové výroby, která je popsána standardem ANSI/ISA-S88.

3. STANDARD ANSI/ISA S88

Průmyslové procesy mohou být rozděleny na diskrétní, spojitě a dávkové procesy. Toto dělení je dané povahou finálního produktu. Pokud je výsledný produkt možné rozdělit na jednotlivé rozlišitelné kusy, jedná se o diskrétní výrobu. Na opačné straně jsou spojitě procesy, jejichž výstupem je souvislý tok produktu. Např. benzín, elektřina, pivo, ... Posledním typem procesu je dávkový proces. Výstupem dávkových procesů jsou výrobky ve vzájemně ohraničených dávkách. Detailně dávkové procesy definuje standard ANSI/ISA-S88.

Standard definuje dávkové (Batch) procesy jako sekvenci chemických, fyzikálních nebo biologických aktivit, které vedou ke změně, transportu, nebo uchování materiálu nebo energie. Cílem dávkových systémů je produkce materiálu konečného množství (dávky) tak, že je na dané množství vstupního materiálu působeno jedním nebo více zařízeními v definovaném pořadí. Výstupní produkty nejsou ani spojitě, ani diskrétní, mají ale charakter obou. [6]

Přestože principy uvedené v tomto standardu je možné využít také při diskrétní nebo spojitě výrobě, není na tyto procesy standard primárně zaměřen.

Později byl vyvinut standard ISA S95 pro integraci řízení průmyslových procesů s informačním systémem (ERP). Využívá hierarchickou strukturu standardu S88 a rozšiřuje ji o informační model pro řízení podniků. Standard ISA S95 je zaměřen hlavně na nejvyšší vrstvy řízení (podnik, závod, úsek) a tato práce se jím tedy nebude zabývat.

3.1 Modely standardu S88

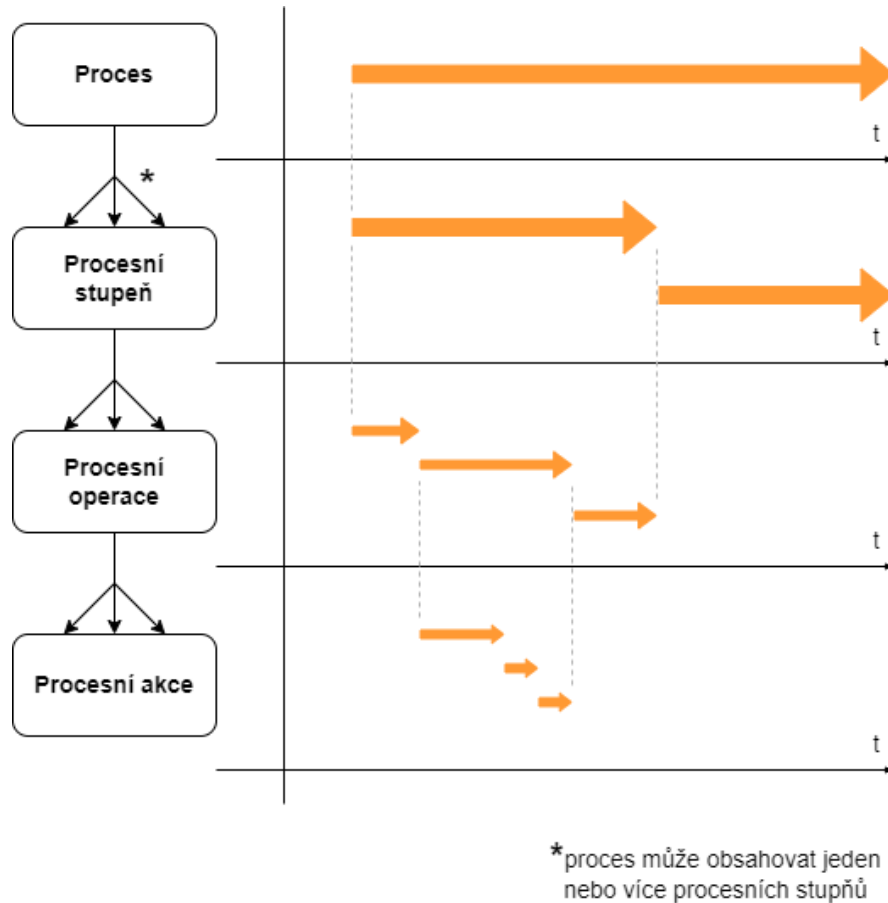
Standard je založen na rozdělení dávkové výroby do tří tzv. modelů a na jejich vzájemném propojení. Všechny modely mají stromovou strukturu, která umožňuje rozdělení složitěho problému na více jednoduchých. Celá výrobní jednotka je tak modulární a modifikovatelná.

Rozdělení na tyto modely umožní také výrazné zlepšení součinnosti mezi technologií (procesní model), integrátory (model procedurálního řízení) a pracovníky montáže (fyzický model). Práce na jednotlivých modelech může probíhat paralelně a není potřeba odborníka, který rozumí všem částem výrobního procesu.

Praktické využití všech modelů pro popis testbedu Self-Acting Barman, včetně skladové jednotky tekutin malého objemu, kterou se tato práce zabývá, je popsáno v kapitole 7.

3.1.1 Procesní model

Procesní model je slovním popisem postupu výroby. Definuje, jaké suroviny, v jakém pořadí a v jakém poměru je potřeba namíchat, ale neupřesňuje, pomocí jakého vybavení. Popisuje dávkový proces z hlediska receptury a můžeme jej přirovnat ke kuchařskému receptu. Skládá se ze čtyř úrovní a v každé úrovni se nachází jedna nebo více složek.

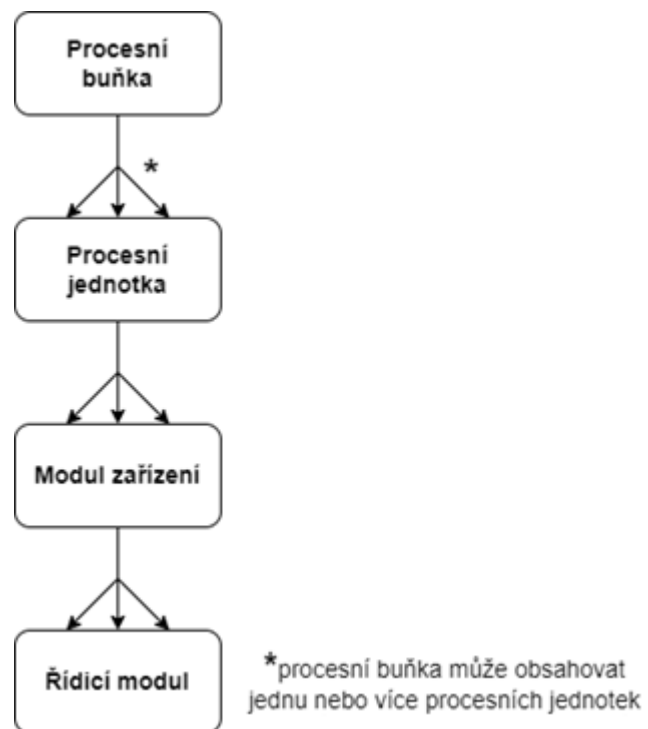


Obr. 3.1: Procesní model [10] inspirováno

Nejvyšší úrovní procesního modelu je **proces**, jehož výstupem je hotový produkt. Určuje, jaké úkony je nutné provést se surovinami k dosažení stanoveného cíle. V případě testbedu Self-Acting Barman je procesem příprava zadaného nápoje. Celý proces se skládá z několika **procesních stupňů**, které mohou být řazeny sériově nebo paralelně. Každý procesní stupeň se dále dělí do **procesních operací**. V jednom okamžiku může být na jedné procesní jednotce (popsáno ve fyzickém modelu) vykonávána jediná procesní operace. Nejnižší úrovní procesního modelu je vrstva jednotlivých **procesních akcí**. Akce jsou základní události, například zapnutí čerpadla, motoru, ...

3.1.2 Fyzický model

Popisuje přímo fyzické vybavení potřebné k vykonání postupu, který je popsán v procesním modelu. Spojováním fyzického vybavení na nízké úrovni vzniká nový prvek ve vyšší úrovni. Tento prvek se potom chová jako jeden celek a změna v nižší úrovni je už možná jedině opakovaným návrhem. Fyzický model je rozdělen do sedmi vrstev. První tři vrstvy, tj. Podnik, Lokalita a Provoz, ale nesouvisí přímo s dávkovou výrobou, a proto je dále nebudu uvažovat.

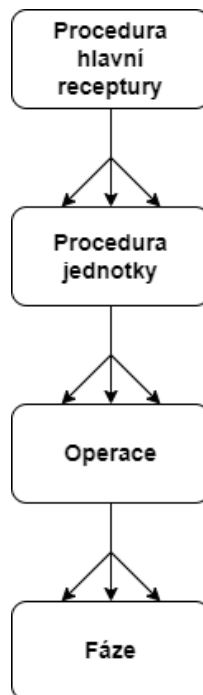


Obr. 3.2: Fyzický model [6] přeloženo

Nejvyšší vrstvou fyzického modelu je **procesní buňka**. Ta obsahuje všechno fyzické vybavení, které je potřebné ke vzniku jedné, nebo více dávek požadovaného produktu. Procesní buňka se skládá z několika **procesních jednotek**. Předpokládá se, že uvnitř jednotky se v daný okamžik bude zpracovávat vždy jen jedna dávka. Na podřazené úrovni se nachází **moduly zařízení**. Jedná se o skupiny řídicích modulů (popsáno níže), které spolupracují k dosažení stanoveného cíle v jedné jednotce. Nejnižšími elementy ve fyzickém modelu jsou **řídicí moduly**. Jedná se o jednotlivé aktuátory a sensory.

3.1.3 Model procedurálního řízení

Spojuje předchozí dva modely. Popisuje funkce, které jsou potřebné pro to, aby se využitím zařízení definovaného ve fyzickém modelu vykonal postup stanovený v procesním modelu. Model se skládá ze čtyř úrovní: procedura hlavní receptury, procedury jednotlivých jednotek, operace a fáze.

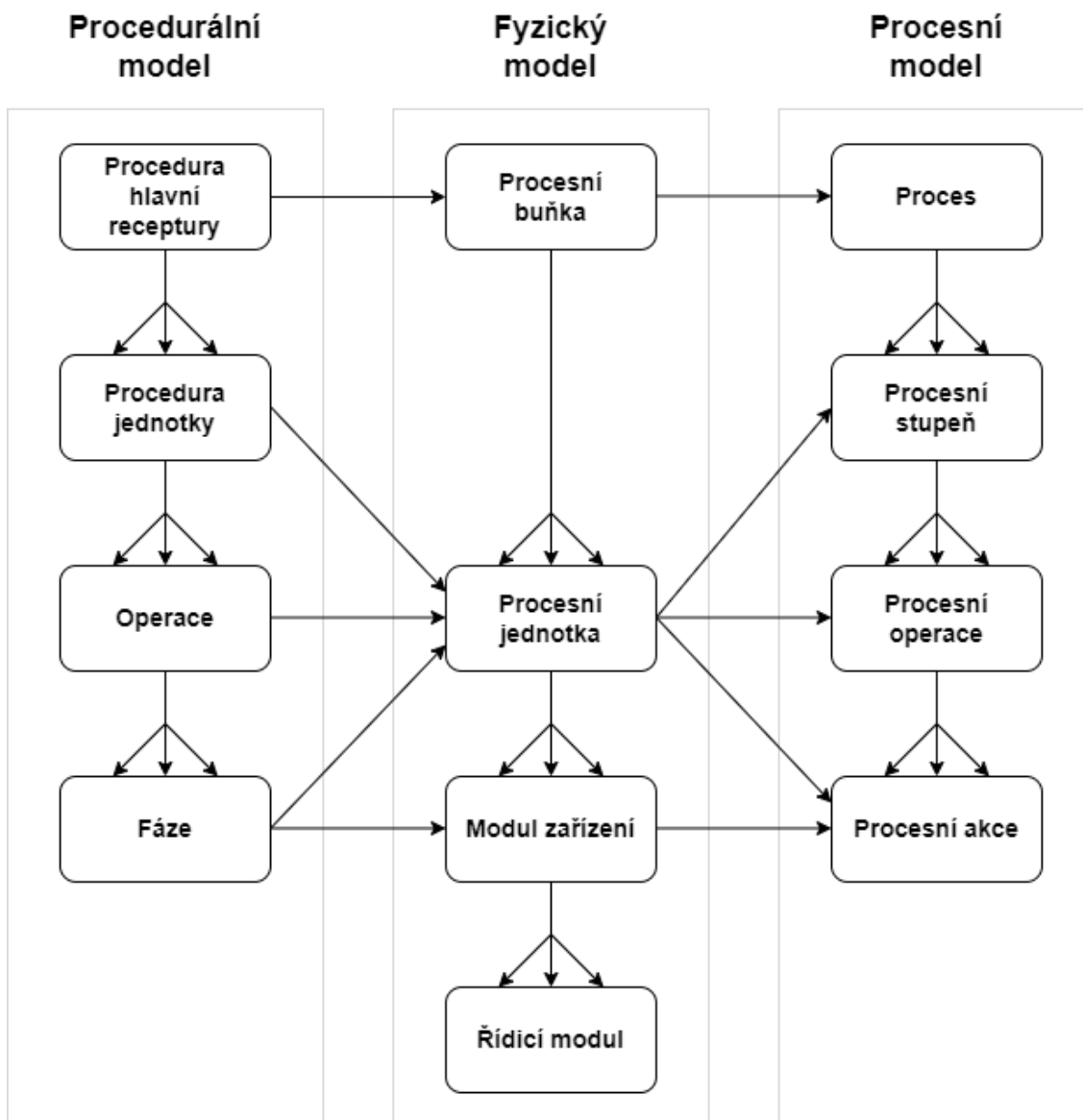


Obr. 3.3: Model procedurálního řízení [6] přeloženo

Procedura hlavní receptury řídí veškeré fyzické vybavení popsané v konkrétním fyzickém modelu. Toto komplexní řízení je rozděleno na procedury jednotlivých jednotek. Všechny **procedury jednotek** souvisí s konkrétními jednotkami definovanými ve fyzickém modelu. Procedura jednotky se skládá z jednotlivých **operací**. V jednom okamžiku může na jedné procesní jednotce probíhat jediná operace, operace tedy neprobíhají paralelně. V nejnižší vrstvě procedurálního modelu se nachází **fáze**. Ty definují, jakým způsobem se využijí jednotlivé moduly zařízení z fyzického modelu. Fáze mohou probíhat paralelně.

3.1.4 Propojení modelů

Procesní, fyzický a procedurální model jsou úzce spojeny. Změna v jednom modelu se musí projevit i ve zbylých dvou. Například pokud bychom upravili fyzický model tak, že dokáže načerpat několik různých druhů tekutin z jedné skladové pozice, operace transportu mezi pozicemi by se stala zbytečnou a procedurální model by byl upraven.



Obr. 3.4: Propojení modelů [7] upraveno

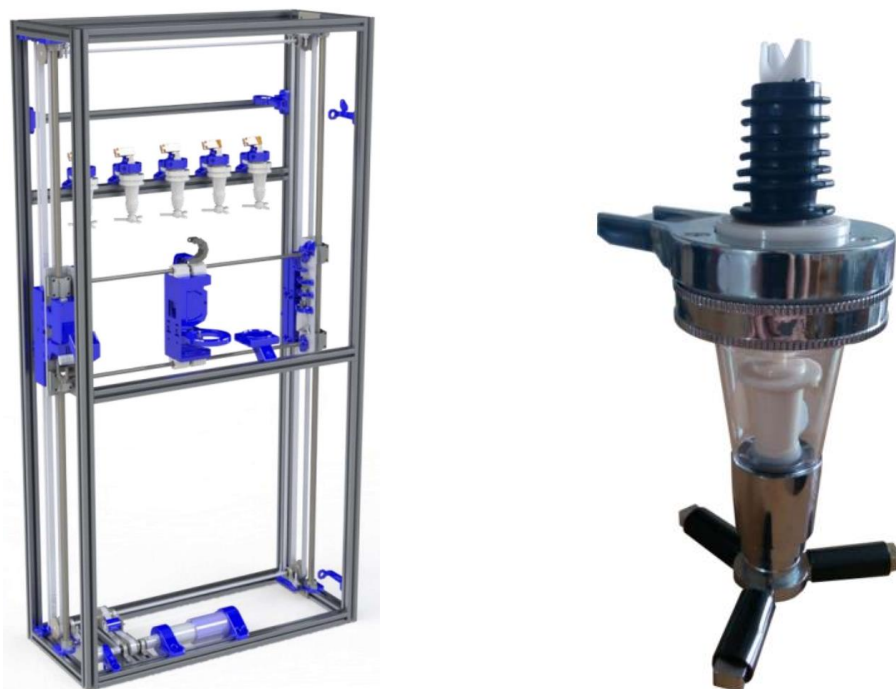
Procedura popsaná procedurálním modelem působí na fyzické vybavení popsané fyzickým modelem a účelem dosažení cílů popsanych v procesním modelu.

4. SKLADOVÁ JEDNOTKA TEKUTIN MALÉHO OBJEMU

Tato bakalářská práce se zabývá modifikací již existující skladové jednotky. Navazuje tak na diplomovou práci Bc. Lukáše Rejchlíka [8].

4.1 Původní konstrukce

Rejchlík vytvořil jednotku, která do skleničky, položené na vstupně-výstupní platformu, automaticky dodá požadované nápoje na základě receptury uložené v NFC čipu, umístěného na dně této skleničky. Tato buňka má největší konstrukci ze všech jednotek testbedu Self-Acting Barman a musí tak být umístěna mimo jeho hlavní pracovní desku. Je připevněna k zadní straně stolu tak, aby manipulátor SCARA (Selective Compliance Assembly Robot Arm) dokázal bez problému umístit a následně vyzvednout skleničku z platformy buňky. 3D model buňky vznikl v CAD (Computer Aided Design – Počítačem podporované projektování) programu SolidWorks a je zobrazen na následujícím obrázku. [8]



Obr. 4.1: Model původní buňky skladu a detail stlačovacího dávkovače [8]

Vnější konstrukce buňky má rozměry 1600 x 760 x 300 mm. Ústředním prvkem je tříosý manipulátor. Dvě osy jsou lineární a jedna rotační. Vertikální pohyb (v ose Z) je poháněn DC motorem BCI 6355 s převodovkou s poměrem 21.6:1 pro snížení jmenovitých otáček a zvýšení momentu. Horizontální pohyb (v ose X) je poháněn krokovým motorem NEMA17. Poslední pohyb je rotační. Umožňuje vyklápění ramene a je poháněn krokovým motorem stejného typu, tzn. NEMA17.

Aktuální poloha manipulátoru je snímána dvojicí enkodérů RB3500. Koncové polohy jsou kontrolovány indukčními snímači LJ12A3-4-Z/BY společně s kontaktními snímači V-153-1C25, které slouží jako havarijní snímače a odpojí napájení motorů v případě poruchy a zamezí tak případnému poškození. Množství tekutiny v láhvích snímají kapacitní snímače KQ5100.

4.2 Problémy s původním řešením

Pro dávkování nápojů jsou využity stlačovací dávkovače (Obr. 4.1). Ty obsahují prepouštěcí komoru, do níž je napuštěn obsah připevněné láhve, na kterou je dávkovač připevněn. Stlačením dávkovacího mechanismu je obsah prepouštěcí komory vypuštěn do skleničky a následným uvolněním je komora opět naplněna. Toto řešení je čistě mechanické a nevyžaduje tak žádnou elektroniku. Přináší však řadu problémů.

Hlavním problémem je to, že na dávkovací mechanismus je pro vypuštění potřeba vyvinout příliš vysoký tlak. Dávkovač je uchycen k hlavní konstrukci pomocí plastového držáku vyrobeného metodou 3D tisku a není tak na vysoký tlak vhodný. Při pokusu o dávkování dochází také k nadměrné zátěži na samotný manipulátor, který drží skleničku a hrozí tak jeho poškození.

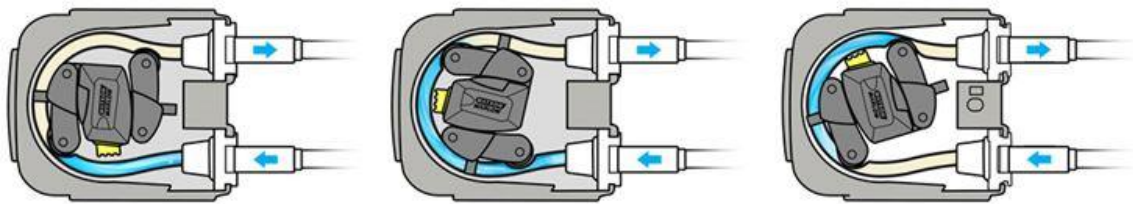
Další nevýhodou je to, že přečerpávací komora má pevně daný objem a není tak možné přesné zvolení požadovaného množství tekutiny. Při opakovaném použití dávkovače dochází také k tomu, že prepouštěcí komora není naplněna vždy přesně stejným objemem, což je způsobeno různým povrchovým napětím jednotlivých tekutin.

Komplikovaná je také výměna prázdných láhví. Lahve musí být nasunuty na dávkovač a umístěny vzhůru nohama. Problém je potlačen využitím výklopných dvířek, na kterých jsou dávkovače umístěny, přesto však není výměna láhve z pohledu rychlosti a pohodlnosti optimální.

5. MODIFIKACE INTELIGENTNÍ SKLADOVÉ JEDNOTKY

Řešením problémů zmíněných v kapitole 4.2 je využití peristaltického čerpadla. Jeho konstrukce je zobrazena na Obr. 5.1.

5.1 Peristaltická čerpadla



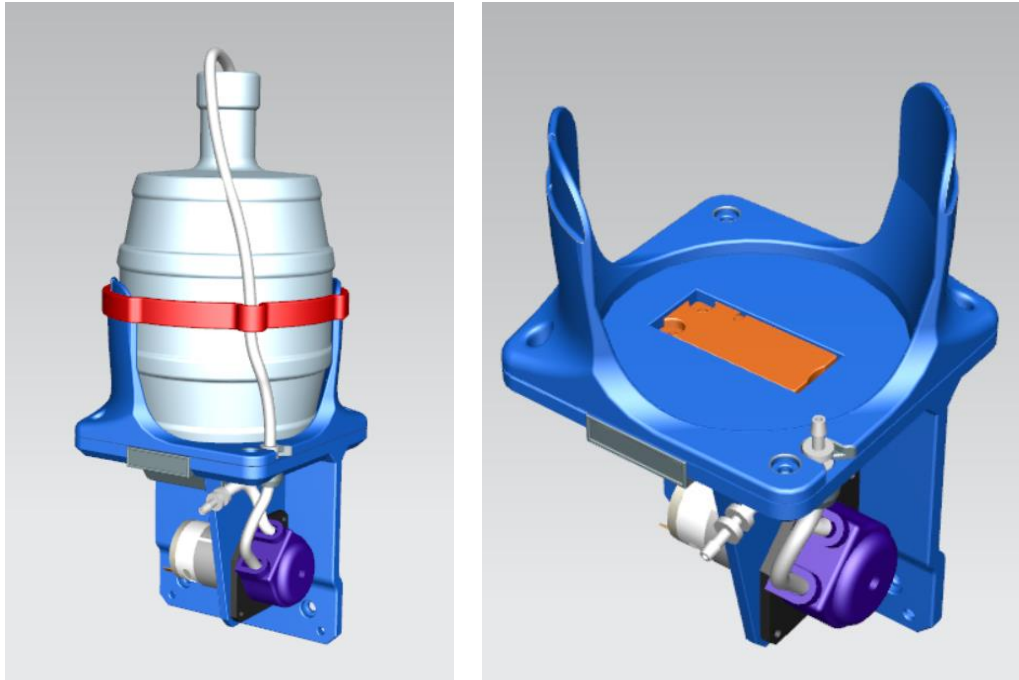
Obr. 5.1: Konstrukce peristaltického čerpadla [11]

Pohyb tekutiny je způsoben mechanickou deformací ohebné trubičky, kterou prochází požadovaná tekutina. Tento princip minimalizuje riziko kontaminace jak tekutiny, tak čerpadla samotného. Trubičku je velmi jednoduché propláchnout a čerpadlo se tím nestává hygienicky závadným.

Čerpadlo umožňuje dávkování velmi přesných objemů tekutin. Je také možné využít zpětného chodu, kdy po dokončení čerpání se zbytky natáhnou zpět a nedochází k samovolnému odkapávání. Neobsahuje žádné namáhané těsnění, ani ventily a má tak dlouhou životnost s minimální potřebou údržby. Otáčení rotačního členu čerpadla zajišťuje DC motor. Vzniká tak potřeba přivedení napájecích vodičů. [11]

5.2 Nový typ skladové pozice

Využitím nového způsobu dávkování tekutin peristaltickým čerpadlem odpadá potřeba uchycení lahví vzhůru nohama. Nová konstrukce skladové jednotky je zobrazena v následujícím obrázku. Za její konstrukci děkuji vedoucímu mé práce Ing. Michalu Husákovi.



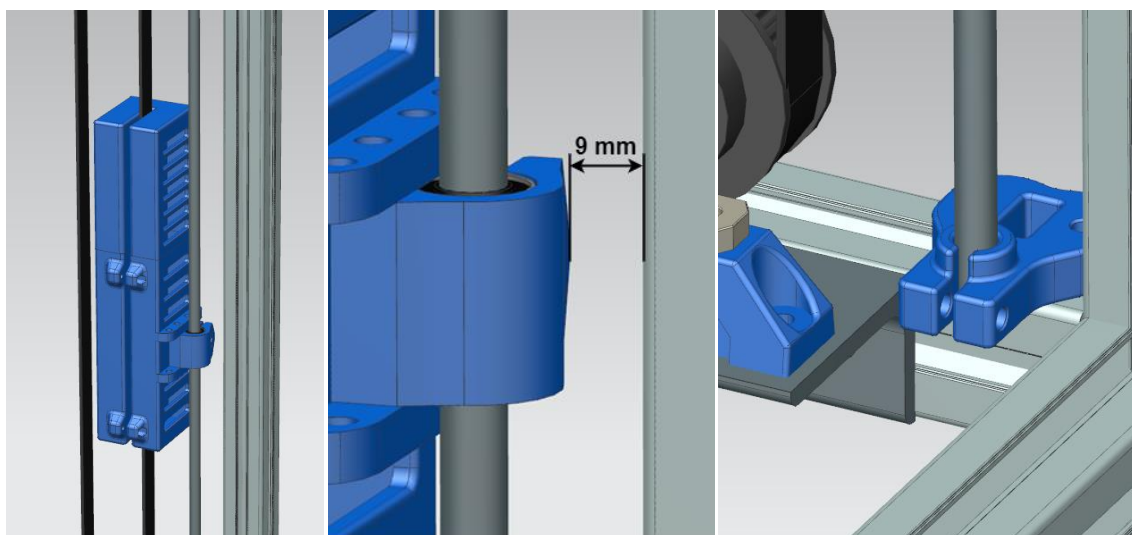
Obr. 5.2: Nový typ skladové pozice

V základně držáku je umístěn kapacitní snímač KQ5100 pro kontrolu množství tekutiny v láhvi. Tento typ skladové pozice využívá jednotný typ láhví. Díky tomu je minimalizován problém s nepřesným měřením hladiny, způsobený různou šířkou skleněného dna různých druhů láhví.

Pro doplnění obsahu láhve stačí z láhve jednoduše vysunout trubičku vedoucí do peristaltického čerpadla a láhev vyjmout a naplnit. Celý proces je jednoduchý, protože láhev není nikde pevně napojena.

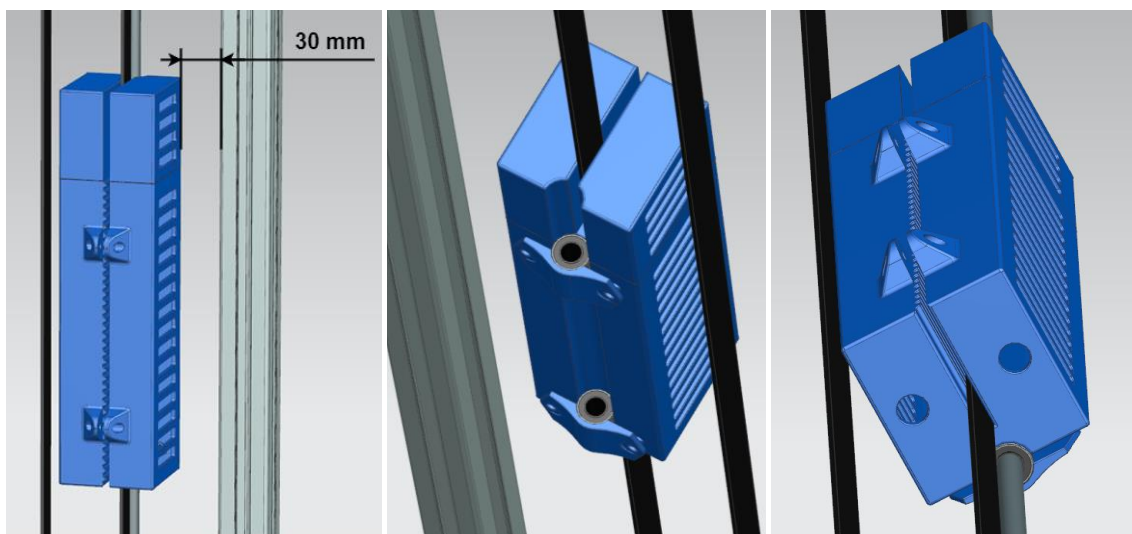
5.3 Úprava konstrukce

Původní konstrukce buňky počítá s čistě mechanickým řešením dávkování tekutin a při jejím návrhu se nepočítalo s nutností přivedení vodičů ke každé skladové pozici pro DC motor a snímače hladiny. Místo, mezi pouzdem protizávaží a ALU profilem, ke kterému jsou vodiče přichyceny, je pouhých 9 mm (měřena minimální vzdálenost). Vzniká tak kolize mezi těmito vodiči a protizávažím manipulátoru.



Obr. 5.3: Původní pouzdro protizávaží a uchycení vodičí tyče

To mne vedlo k modifikaci konstrukce. Bylo potřeba upravit pouzdro na protizávaží tak, aby se zvětšil prostor mezi ním a hliníkovým profilem konstrukce buňky, ke kterému jsou vodiče uchyceny. Stejně tak se musela změnit pozice vodičí tyče protizávaží a způsob jejího uchycení k rámu buňky. Touto modifikací došlo ke zvětšení prostoru mezi pouzdem a ALU profilem na 30 mm (měřena minimální vzdálenost).

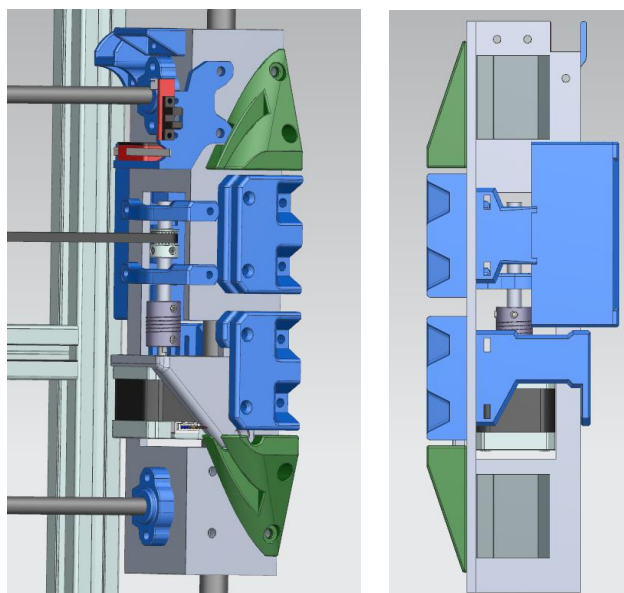


Obr. 5.4: Nové pouzdro protizávaží

Na dně pouzdra se nachází dva kruhové otvory s průměrem 10 mm pro snadné vytažení závaží v případě potřeby. K tomu mne vedl problém s původním designem, kdy vytažení závaží bez poškození pouzdra bylo v podstatě nemožné. Nyní lze na závaží zespu potlačit například šroubovákem a dojde k jeho vysunutí.

Pro uchycení vodící tyče bylo využito kovového uchycení typu SHF8. Nebylo tak potřeba dalšího modelování a 3D tisku, protože uchycení se připevní přímo k ALU rámu buňky.

Tím, že bylo protizávaží posunuto více do středu buňky, došlo k jeho přiblížení k samotnému manipulátoru. Mohlo by tak dojít k jejich vzájemné kolizi. Za celou dobu testování se tento problém nevyskytl ani jednou, přesto by však mohl být velmi závažný. Proto byly vytvořeny zábrany, které eliminují negativní dopad případných kolizí. Zábrany jsou v následujícím obrázku vyznačeny zelenou barvou pro odlišení, ve skutečnosti ale mají stejnou barvu jako ostatní díly.



Obr. 5.5: Zábrany proti kolizi

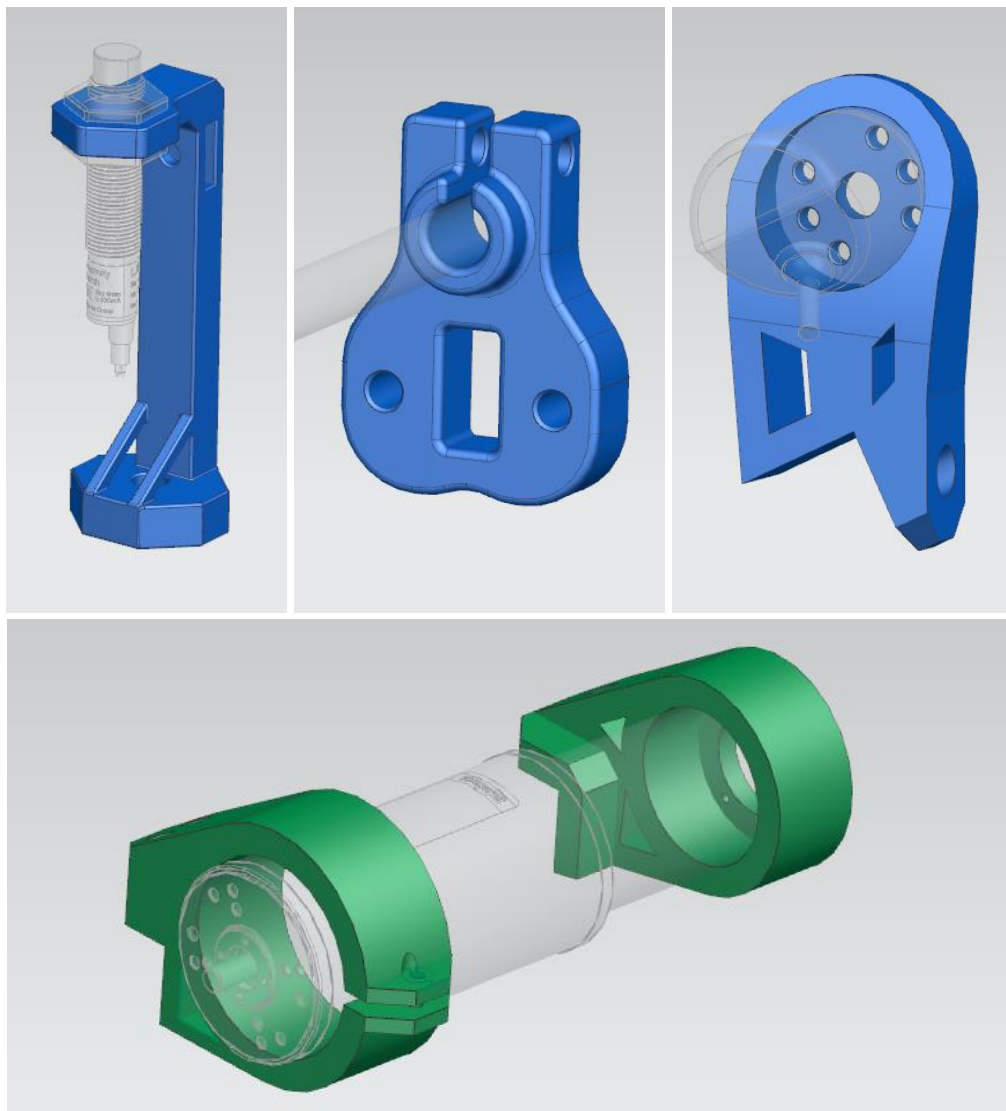
5.3.1 Opravy původního modelu

Dříve, než jsem se mohl pustit do úprav konstrukce a tvorby digitálního dvojčete, bylo potřeba provést aktualizaci modelu buňky, aby odpovídal stavu fyzického protějšku. Původní model vznikl v programu SolidWorks. Aby bylo možné vytvořit a zprovoznit digitální dvojče jednotky, bylo potřeba model převést do programu NX Siemens.

Sestava modelu buňky, kterou jsem obdržel jako výchozí bod své práce, měla nejednotný formát pojmenování jednotlivých dílů, některé pojmenování nedávala smysl a několik dílů chybělo úplně.

Jelikož bude po dokončení buňka začleněna do testbedu Self-Acting Barman, bylo nutné sjednotit formát pojmenování všech dílů a podsestav podle formátu ostatních buněk testbedu pro jednoduchou orientaci.

Díly, které chyběly v digitální formě, jsem musel vytvořit znovu na základě již fyzicky existujících kusů, jsou zobrazeny v Obr. 5.6. Jednalo se o úchyty indukčních senzorů, vodicích tyčí, enkodéru pro DC motor a DC motoru samotného.



Obr. 5.6: Díly vytvořené na základě fyzické předlohy

5.4 Finální konstrukce buňky



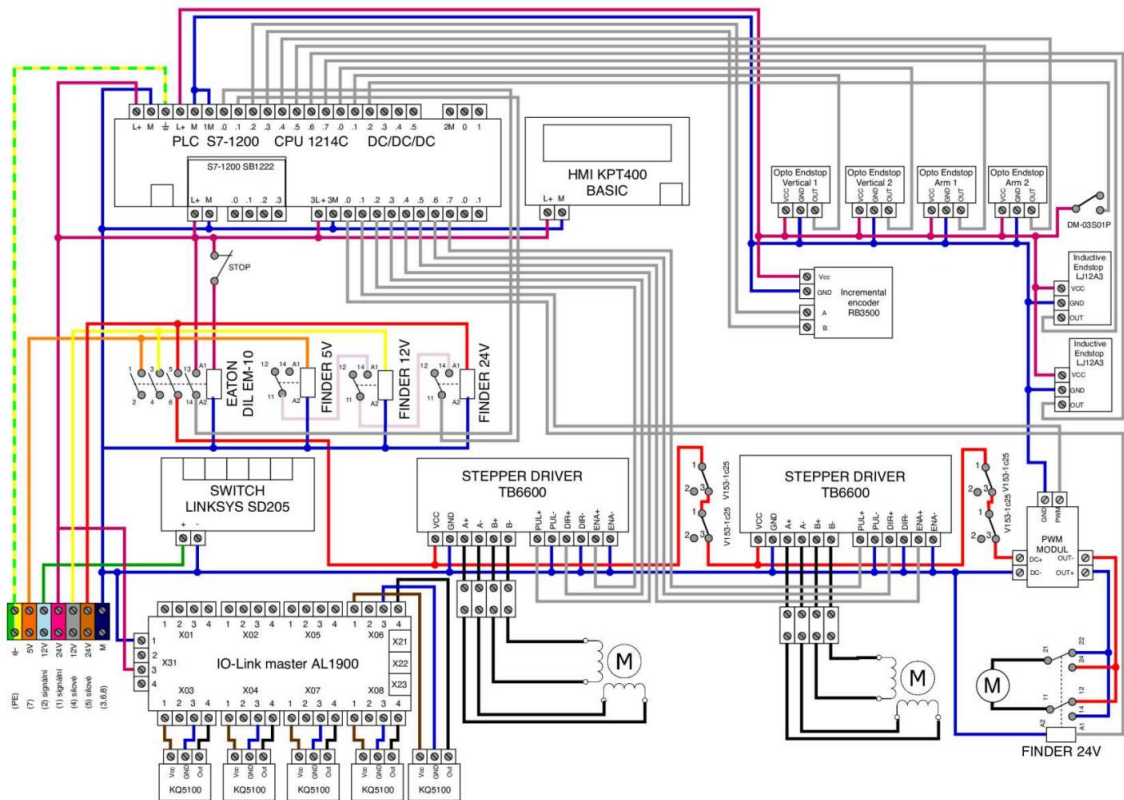
Obr. 5.7: Finální stav buňky

Skladová jednotka momentálně obsahuje šestnáct skladových pozic. Toto číslo souvisí s reléovým modulem pro připojení DC motorů peristaltických čerpadel, který bude popsán dále v textu (kapitola 5.5.1). Každý modul obsahuje osm výstupů a buňka využívá tyto moduly dva.

Samotná konstrukce buňky umožňuje rozšíření počtu skladových pozic až o sedm slotů na celkový počet maximálně dvaceti tří slotů.

5.5 Elektrické zapojení

Elektrické zapojení je modifikací zapojení, které uvádí Rejchlík ve své práci [8].



Obr. 5.8: Původní elektrické schéma zapojení [8]

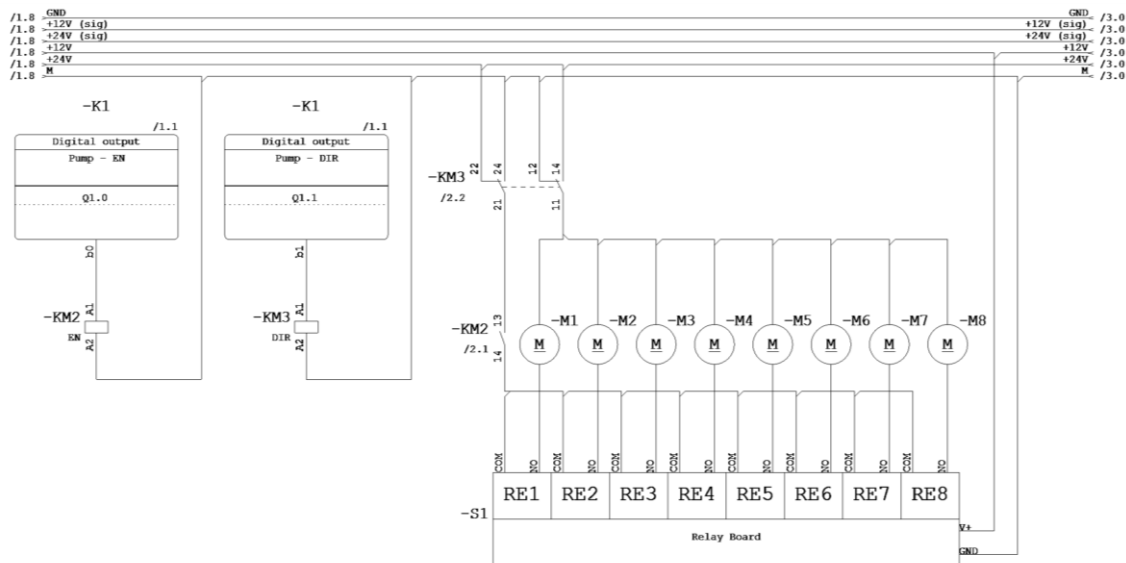
Schéma, které jsem vytvořil já, vzniklo v softwaru WS-CAD a respektuje standard DIN-81346. WS-CAD obsahuje rozsáhlou databázi standardizovaných značek a umožňuje také tvorbu značek vlastních. Při tvorbě schématu jsem využil obou možností.

Vzniklé schéma je vylepšením zejména po formální stránce. Dále upravuje drobné funkční nedostatky původního schématu, a hlavně zvyšuje celkovou přehlednost. Kompletní schéma elektrického zapojení se nachází v příloze (Příloha A - Elektrické schéma zapojení).

5.5.1 Zapojení reléového modulu pro ovládání peristaltických čerpadel

Moduly jsou napájeny stejnosměrným napětím 12 V. Každý z modulů obsahuje osm relé a pro správnou funkci je rozvaděč doplněn o dvě další samostatná relé, která jsou společná pro všechny moduly. Samostatná relé jsou připojena na PLC výstup Q1.0 pro aktivaci (EN) a Q1.1 pro změnu směru chodu čerpadla (DIR).

Relé moduly slouží jako moduly pro rozšíření počtu výstupu a s PLC komunikují pomocí Modbus komunikace. V porovnání s klasickými PLC výstupy jsou ale výrazně pomalejší. Pro tuto aplikaci je ale zásadní jejich počet, ne rychlost.



Obr. 5.9: Schéma zapojení reléového modulu

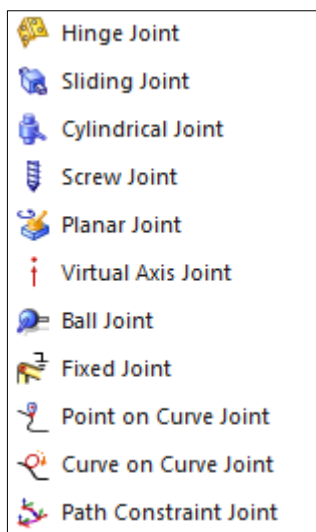
6. VÝVOJ DIGITÁLNÍHO DVOJČETE

Digitální dvojče bylo vytvořeno v programu NX Siemens. Po dokončení modelu buňky v aplikaci NX Modeling bylo možné na práci navázat v aplikaci NX Mechatronics Concept Designer (MCD). Tato aplikace nabízí nástroje pro interaktivní simulování komplexních pohybových akcí mechanických systémů.

Nejprve je nutné jednotlivým dílům přiřadit tuhá a kolizní tělesa. Tuhá tělesa představují objekty, které podléhají fyzikálním zákonům. To znamená, že na ně mohou působit síly jako je gravitace či síly vyvolané jednotlivými pohonnými jednotkami. MCD na základě tvaru objektu automaticky přiřadí tuhým tělesům těžiště a hmotnost. Předpokládá však, že celé těleso je z homogenního materiálu, pro některá tělesa je tedy nutné těžiště i hmotnost ručně upravit.

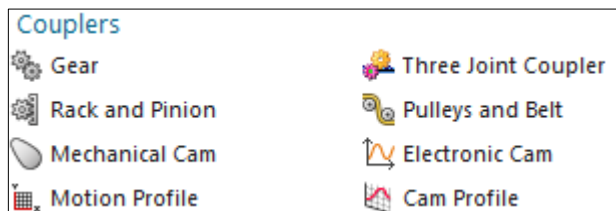
Kolizní tělesa jsou objekty, které podléhají vzájemným kolizím. Těmito kolizemi mohou být ať už nechtěné nárazy, tak například uchopení skleničky manipulátorem. Jako kolizní tělesa je tedy nutné označit všechny díly, které mají interagovat vzájemným dotykem.

Tuhým tělesům potom lze přiřadit spoje (Joints). Spoje mohou být pevné, posuvné, rotační atd. Například lineární spoj mezi protizávažím a jeho vodící tyčí zajistí, že se protizávaží bude pohybovat po přesně lineární trajektorii dané tyčí. Při volbě spoje je vhodné nastavit mu také limity pohybu. Tyto limity omezují rozsah pohybu. Pokud tedy nastavím rozsah pohybu manipulátoru pomocí limitů spojů, nemusím jeho najetí do krajních poloh řešit pomocí kolizních těles, které se negativně projevují na potřebném výpočetním výkonu pro plynulé zobrazení simulace.



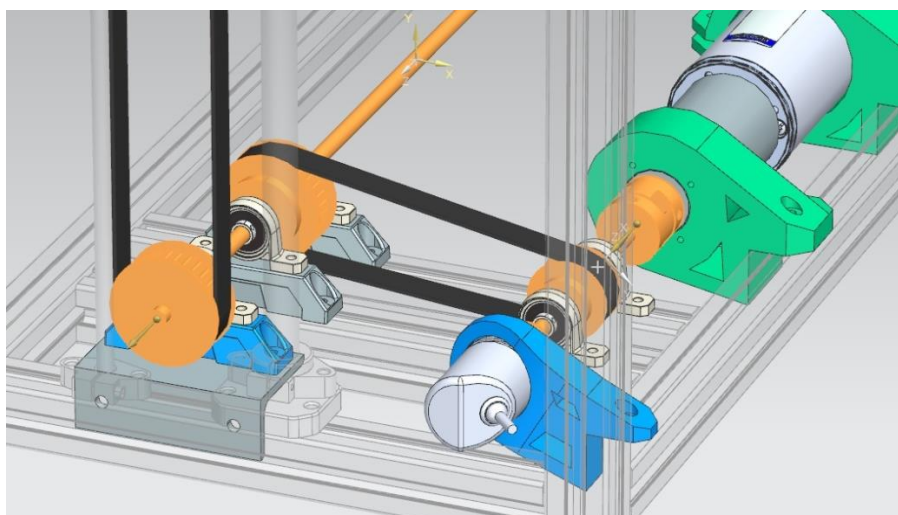
Obr. 6.1: MCD – Možnosti spojů

Jednotlivá tuhá tělesa se mohou vzájemně ovlivňovat. Toho můžeme docílit využitím vazebních členů (Couplers). V mé práci jsem využil vazební členy ozubená kola (Gears) a ozubené hřebeny (Rack and Pinion).



Obr. 6.2: MCD – Možnosti vazebních členů

Při využití ozubených kol je potřeba přiřadit, která dvě tuhá tělesa budou interagovat. Mezi tělesy lze pak nastavit převodový poměr, který můžeme poměrně jednoduše vypočítat ze znalosti průměrů jednotlivých ozubených kol.



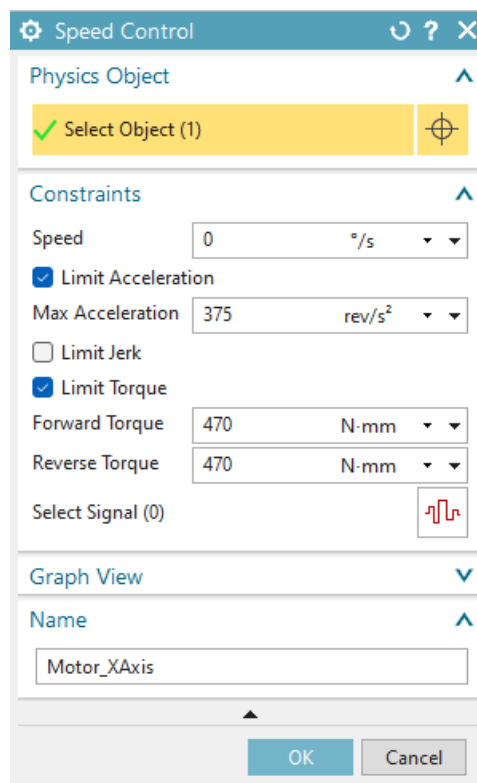
Obr. 6.3: MCD – Ozubená kola

Na stejném principu je nastaven také vazební člen ozubeného hřebenu. Zde však volíme jedno tuhé těleso, které vykonává rotační pohyb a jedno, které vykonává pohyb lineární. Jednotlivá tuhá tělesa jsou v Obr. 6.3 zvýrazněna oranžovou barvou.

Dále je potřeba nastavit snímače a akční členy. Snímače je možné nastavit na kolizním principu mezi dvěma tělesy (Collision Sensor). Tzn. při kolizi mezi snímačem a tělesem dojde k sepnutí snímače. Výpočet kolizí však bývá výpočetně náročný, proto je lepším způsobem, je-li to možné, nastavit snímače na principu překonání nastaveného limitu (Limit Switch). Tzn. že když těleso, jehož pozici snímáme, překoná zadanou polohu, dojde k sepnutí snímače. Limitní polohu potom můžeme zadat tak, aby přesně odpovídala pozici fyzického snímače. Z funkčního hlediska je toto řešení ekvivalentní koliznímu řešení, potřebný výpočetní výkon je však minimální.

Každá nová simulační funkce má negativní dopad na plynulost simulace. Cílem je tedy vytvořit co možná nejpřesnější simulaci, zároveň však musíme myslet na to, že máme omezený výpočetní výkon.

Akčním členům, tzn pohonným jednotkám, je potřeba přiřadit funkci řízení rychlosti (Speed Control). Kromě přiřazení tuhého tělesa lze nastavit i maximální zrychlení a moment sil, což jsou parametry zjistitelné z datasheetu výrobce motoru.

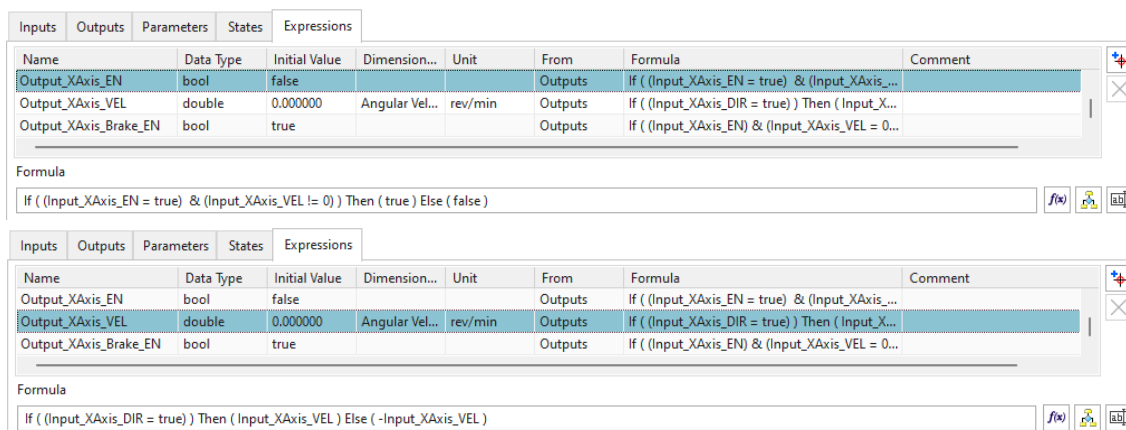


Obr. 6.4: MCD – Řízení rychlosti

6.1 Komunikace digitálního dvojčete s PLC

Komunikace mezi PLC a virtuálním dvojčetem probíhá pomocí OPC-UA serveru. Pro zajištění komunikace virtuálního dvojčete s OPC-UA serverem je potřeba definovat symboly pomocí tabulky symbolů (Symbol Table), které tvoří propojení serveru a modelu. Symboly, určené pro komunikaci se serverem je potom potřeba propojit se signály, které reprezentují vytvořené senzory a akční členy modelu.

Pokud požadujeme, aby simulační funkce plnily složitější funkci, než jen zapnuto/vypnuto, musíme využít tzv. bloků výrazu (Expression Blocks). Pomocí nich definujeme složitější funkcionalitu. V následujícím obrázku můžeme vidět blok výrazů pro řízení motoru v ose X. Motor je povolen, pokud je signál VEL, reprezentující rychlost otáček, různý od nuly. Změna směru je zajištěna dalším výrazem říkajícím, že rychlost otáčení je záporná, tzn., že se hřídel motoru otáčí opačným směrem, pokud je hodnota signálu DIR rovna hodnotě false.



Obr. 6.5: MCD – Blok výrazů pro řízení motoru v ose X

Při využití OPC-UA pro komunikaci mezi PLC S7-1200 a digitálním dvojčetem dochází k malé časové prodlevě, která se projeví při regulaci najíždění manipulátoru na požadovanou polohu a může způsobovat potíže. Pokud je potřeba minimalizovat jakoukoliv prodlevu mezi PLC a DT, je vhodné využít simulovaného PLC S7-1500, které poskytuje program PLCSIM Adv. Konverze programu z PLC S7-1200 do PLC S7-1500 nezabere ani pět minut a není složitá. Jediný zádrhel je v systémových funkcích Motion Control ve funkčním bloku AxisControl (8.3.1), které na S7-1500 fungují jinak, než na S7-1200. Tyto funkce ale slouží čistě pro řízení reálné jednotky a digitální dvojče je vůbec nevyužívá. V rámci konverze je tedy potřeba tyto funkce odstranit.

Dle dokumentace pro PLC řady S7-1200 a S7-1500 je zřejmé, že další možností zefektivnění simulace je využití příkazů PUT a GET pro přeposílání dat z reálného PLC S7-1200 do simulovaného PLC S7-1500. Vzniká tak ideální řešení problému. Řídicí program běží na S7-1200 beze změny a vstupní a výstupní signály se odesílají do simulovaného S7-1500. Není tak nutná žádná konverze programu a zároveň je využito rychlé komunikace mezi simulovanou S7-1500 a digitálním dvojčetem.

Na kompletní realizaci však nebyl dostatek času a komunikace se mi podařila zprovoznit jen částečně.

7. PRAKTICKÉ VYUŽITÍ STANDARDU ANSI/ISA-S88

Řídicí logika vzniká na základě standardu ANSI/ISA-S88, popsaného v kapitole 3. Standard umožňuje tvorbu modelů, které reprezentují stromovou strukturu, kdy je jeden složitý problém rozdělen do několika jednodušších problémů. Díky tomu je tomto celá výroba modulární a jednoduše modifikovatelná. Pro vytvoření strukturované řídicí logiky je nejprve potřeba vytvořit procesní a fyzický model skladové jednotky.

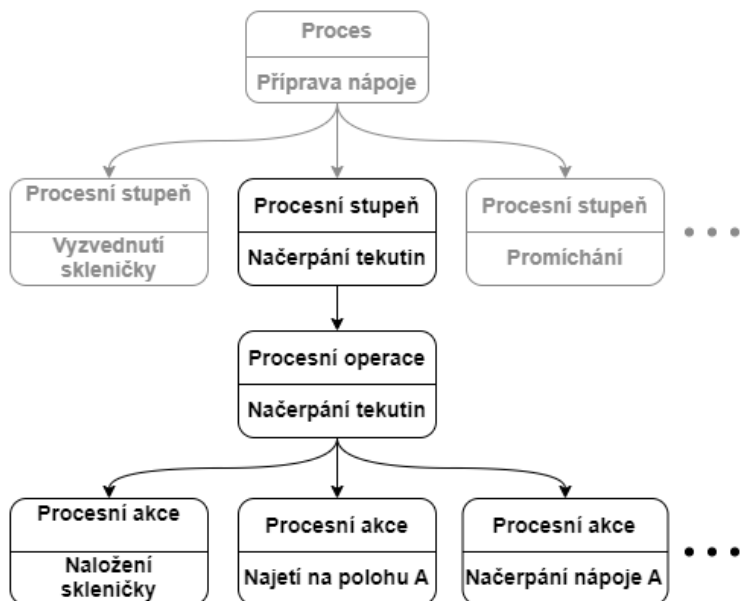
Jak už bylo zmíněno v kapitole 2.1, skladová jednotka je součástí výrobní buňky testbedu Self-Acting Barman. Na nejvyšší úrovni všech modelů se tak nachází právě zmíněný testbed, který se poté v nižších úrovních větví do jednotlivých procesních jednotek. Jednou z těchto jednotek je skladová jednotka, kterou se zabývá tato práce. Budou tedy popsány pouze tři nejnižší vrstvy modelů.

7.1 Procesní model skladové jednotky

Úkol skladové jednotky je popsán v procesním stupni. Cílem je naplnit předanou skleničku požadovanými tekutinami. Procesní stupeň se dále dělí do procesních operací.

V jednom okamžiku smí na jedné jednotce probíhat jediná operace. Operace v jedné jednotce nemohou probíhat paralelně. Členění do více operací při této aplikaci nemá smysl, protože celý postup výroby jedné dávky – plnění jedné skleničky, bude probíhat nepřerušovaně, tzn. v jediné operaci.

Pro možnost naplnění skleničky je potřeba využít několika procesních akcí. Těmi jsou naložení skleničky manipulátorem, transport skleničky na požadovanou skladovou pozici, spuštění čerpadla požadované skladové pozice, vyložení skleničky. Jednotlivé procesní akce se mohou libovolně opakovat a mohou se vykonávat paralelně.



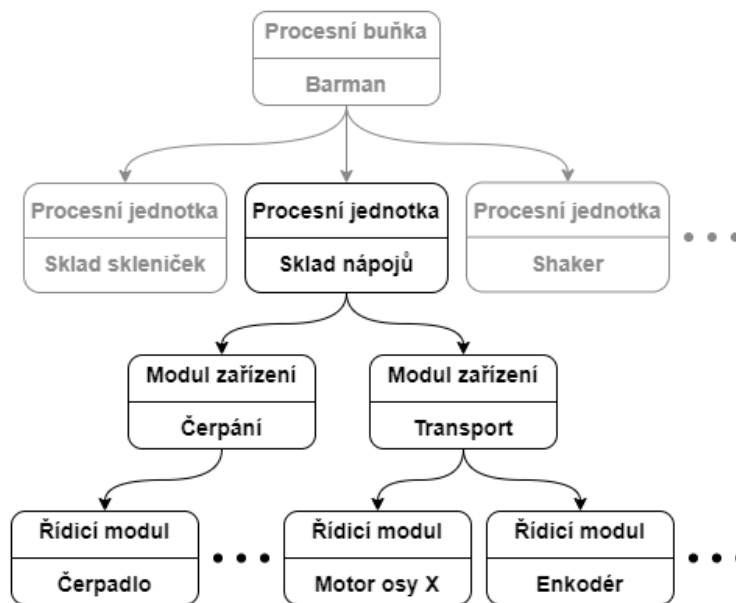
Obr. 7.1: Procesní model skladové jednotky

7.2 Fyzický model skladové jednotky

Nyní je potřeba sestavit fyzický model skladové jednotky. Ta podle standardu představuje procesní jednotku. V procesním modelu je stanoveno, že je potřeba nějakým způsobem dopravit skleničku na požadovanou pozici a načerpat do ní tekutinu.

Manipulaci se skleničkou zajistí modul zařízení transportu. Tímto modulem zařízení je tříosý manipulátor popsáný v kapitole 4.1. Skládá se z mnoha řídicích modulů, jako například motor osy X (krokový motor Hanpose 17HS440), motor osy Z (DC motor BCI 6355), snímače polohy manipulátoru (inkrementální enkodéry RB3500) atd.

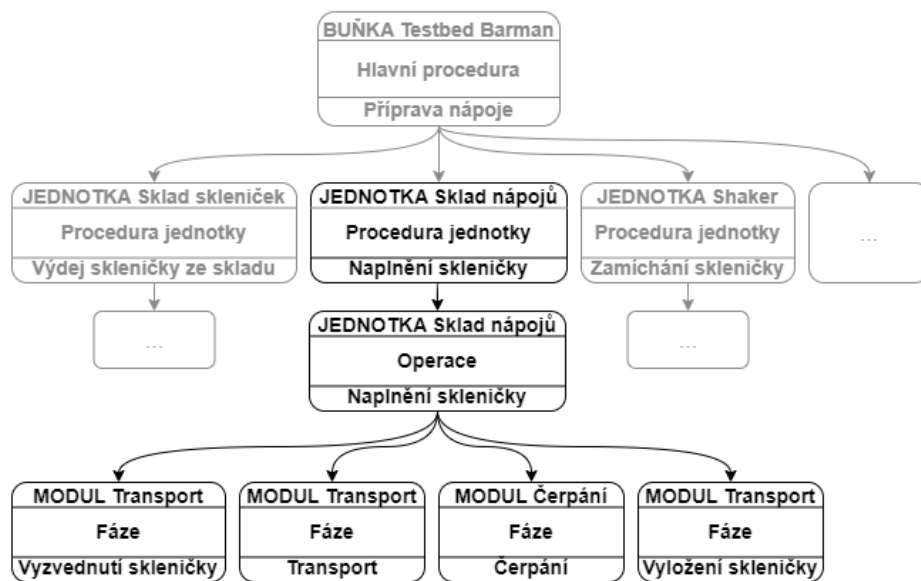
Čerpání tekutiny je zajištěno řídicími moduly čerpadel (viz kapitola 5.1) a snímači hladiny v láhvi (kapacitní snímač KQ5100)



Obr. 7.2: Fyzický model skladové jednotky

7.3 Model procedurálního řízení skladové jednotky

Po sestavení procesního a fyzického modelu, může vzniknout model procedurální. Ten popisuje, jakým způsobem je potřeba využít fyzické vybavení popsáné ve fyzickém modelu, aby byl vykonán postup popsáný v procesním modelu. Větvení řídicí logiky je popsáno v následujícím diagramu.



Obr. 7.3: Model procedurálního řízení skladové jednotky

Procedura jednotky představuje celou řídicí logiku skladové jednotky. Jak už bylo popsáno v kapitole 7.1, na jedné procesní jednotce může v jednom okamžiku probíhat jediná operace. Pokud chceme spustit jinou operaci, musí se ta aktuální přerušit. U skladové jednotky nemá smysl dělit řídicí logiku do více operací, protože se vždy zpracovává jen jeden požadavek, tedy plnění jedné skleničky. Dělení do více operací by mělo smysl například v případě, že by bylo možné skleničku v průběhu plnění odložit a vykonat jinou operaci. V této aplikaci to ale nemá význam.

Operace naplnění skleničky se větví do čtyřech fází. Tyto fáze budou popsány v kapitole 8.3.

8. ŘÍDICÍ LOGIKA

Řídicí logika pro skladovou jednotku tekutin malého objemu vzniká ve vývojovém prostředí Siemens TIA Portal V17 pro řídicí systém PLC Siemens S7-1200. Je implementována na základě standardu ANSI/ISA-S88, popsaného v kapitole 3. Struktura programu je popsána modelem procedurálního řízení (Obr. 7.3).

Každý prvek programu (motor, fáze, ...) komunikuje se zbytkem programu pomocí vlastního UDT (User Data Type – Uživatelský datový typ), který tvoří interface pro předávání dat. Tyto UDT jsou inspirovány standardem PackML a skládají se ze tří struktur:

- State – pro předání informace o aktuálním stavu
- Cmd – pro řízení dané akce
- Param – pro předání parametrů akce

Jako příklad je uveden UDT pro operaci přípravy nápoje. Všechny prvky programu musí dodržovat toto rozdělení na tři struktury, data uvnitř těchto struktur se však liší.

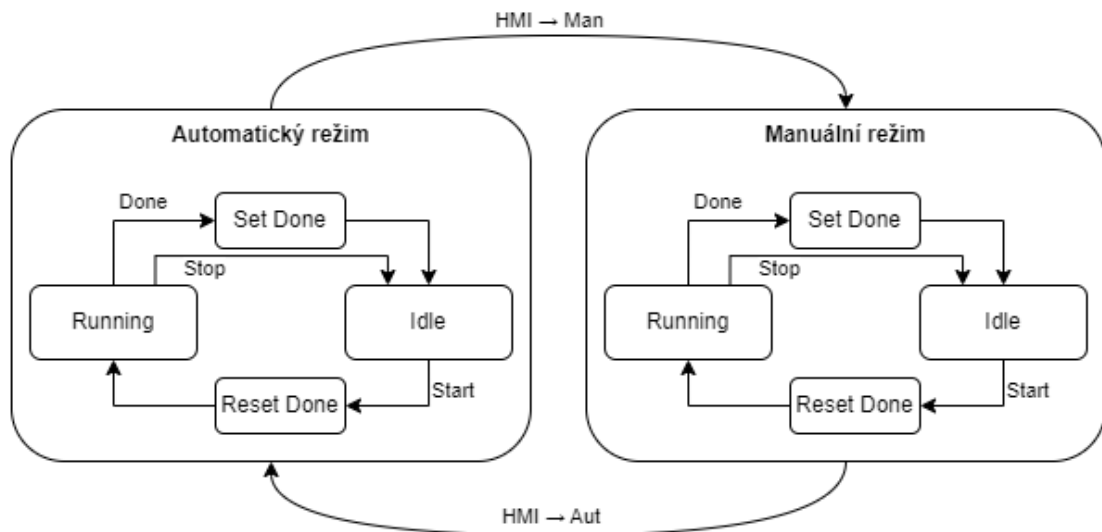
OperationDrinkPrep						
	Name	Data type	Defa...	Comment
1	State	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Zpetná vazba
2	Running	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jednotka běží
3	Idle	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jednotka čeká
4	Done	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Operace dokončena
5	Aut	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Automatický režim
6	Man	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Manuální režim
7	State	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Stav
8	RemainingVol	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Kolik ve skleničce zbývá místa
9	Cmd	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Řízení
10	StartAut	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Start, pokud se nachází v Aut
11	StopAut	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Stop, pokud se nachází v Aut
12	StartMan	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Start, pokud se nachází v Man
13	StopMan	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Stop, pokud se nachází v Man
14	ResetDone	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Resetuj stav Done
15	AutCtrl	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Přepni do Aut režimu
16	ManCtrl	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Přepni do Man režimu
17	Param	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	Liquid	Array[1..16] of Int		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Požadované tekutiny
19	Volume	Array[1..16] of Int		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Množství daných tekutin

Obr. 8.1: UDT OperationDrinkPrep

8.1 Provozní režimy

Jednotka se může nacházet v manuálním nebo automatickém režimu. Mezi stavy lze přepínat pomocí dotykového operátorského panelu HMI. Dokončení aktuální akce signalizuje příznak Done. Pokud je aktivní automatický režim, řízení přebírá nadřazený systém, tedy procesní buňka testbed Self-Acting Barman. V manuálním režimu je

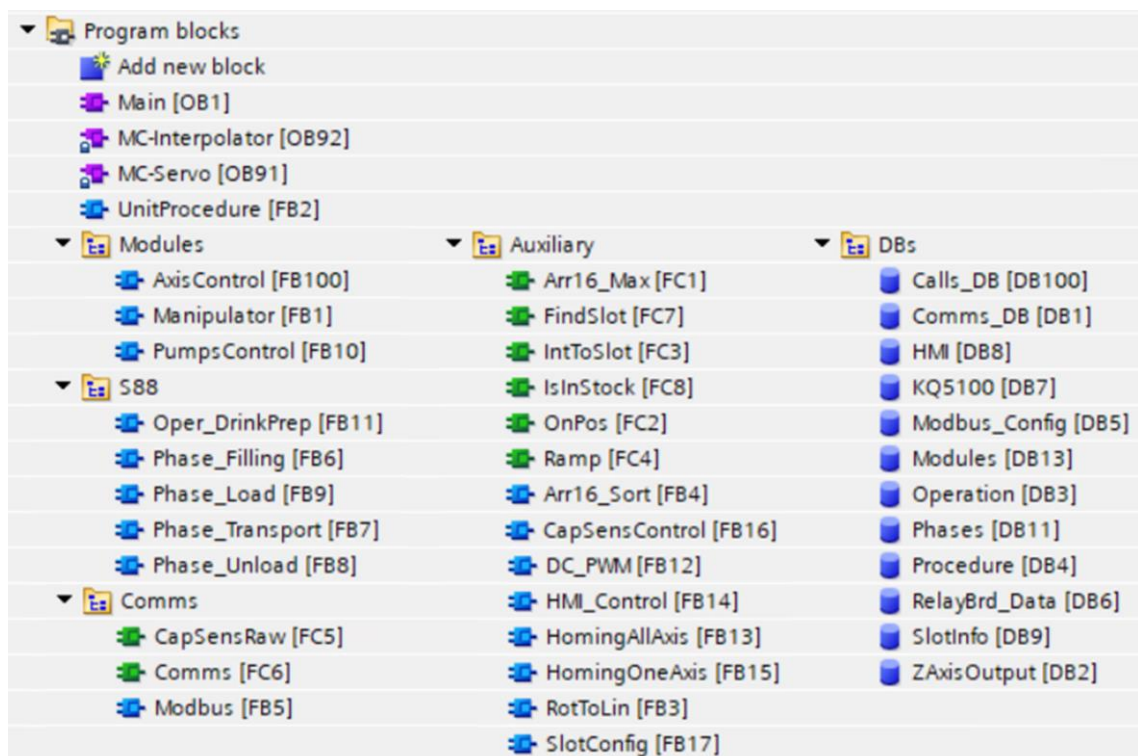
skladová jednotka řízena dotykovým operátorským panelem.



Obr. 8.2: Stavový automat provozních režimů

8.2 Členění programu

Pro přehlednost jsou jednotlivé funkční bloky a funkce rozčleněny do odpovídajících skupin podle jejich účelu.



Obr. 8.3: Struktura programu

8.3 Funkční bloky pro řízení modulů zařízení

Funkční bloky přímo související s moduly zařízení, tj. modul pro čerpání a pro transport skleničky, jsou sjednoceny ve skupině Modules. Starají se o to, aby se s podřazenými řídicími moduly, tzn. motory a čerpadly, dalo jednoduše pracovat.

8.3.1 AxisControl

Funkční blok AxisControl je bez pochyby nejsložitější částí celého programu. Slouží pro řízení jednotlivých motorů tříosého manipulátoru. Ten využívá dva krokové motory a jeden DC motor. Tyto dva typy motorů využívají naprosto odlišný princip řízení, funkční blok AxisControl však zajišťuje sjednocený interface a velmi zjednodušuje následnou práci s pohony. Při následném využití AxisControl tak vůbec nezáleží, jaký motor zrovna řídím, protože se nyní zvenčí všechny chovají stejně. Interface pro řízení a sběr dat z motoru se opět realizován s využitím UDT, který je totožný pro všechny tři motory.

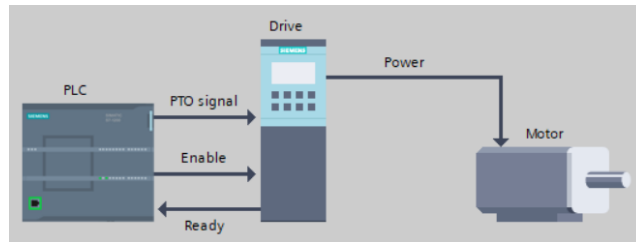
Engine						
	Name	Data type	Comment
1	State	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Zpětná vazba
2	Error	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Nastala chyba
3	Standstill	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Stojí nehnutě
4	Running	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Pohybuje se
5	HomePosition	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Pozice homing switche
6	ActualPosition	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Aktuální pozice (z encoderu)
7	ActualSpeed	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Aktuální rychlost
8	Enabled	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Motor je povolen
9	DoneHome	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Homing byl dokončen
10	DoneMoveRel	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Relativní posun byl dokončen
11	DoneMoveAbs	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Najetí na absolutní polohu je dokončeno
12	MaxVel	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Maximální rychlost motoru - dané konstrukcí
13	MinVel	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Minimální rychlost motoru - dané konstrukcí
14	Cmd	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Řízení
15	Power_En	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Spust' napájení motoru
16	Reset_Exe	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Resetuj chyby
17	Home_Exe	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Spust' funkci homingu
18	ResetHome_Exe	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Převede motor do nehomovaného stavu
19	Halt_Exe	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Zastav (pause)
20	MoveRel_Exe	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Spust' relativní pohyb (o kolik)
21	MoveAbs_Exe	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Spust' absolutní pohyb (na jakou pozici)
22	JogFwd_En	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jed' dopředu
23	JogBck_En	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jed' dozadu
24	Param	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Parametry
25	Velocity	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jakou rychlostí mm/s nebo °/s
26	RelDist	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	O kolik mm nebo °
27	AbsPos	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Na jakou pozici
28	Sensors	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Snímače
29	LowLim	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Dolní krajní snímač
30	HighLim	Bool	fa!	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Horní krajní snímač
31	SimActPos	Real	O.C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Aktuální pozice ze simulace

Obr. 8.4: UDT Engine

Funkční blok AxisControl se skládá ze dvou hlavních částí. Řízení reálných motorů a řízení simulovaných motorů. Pro řízení reálných motorů tříosého manipulátoru je využito systémových funkcí Motion Control. Tyto funkce však fungují jedině s reálným motorem.

Pro možnosti využití systémových funkcí Motion Control pro řízení reálného motoru je potřeba definovat a konfigurovat tzv. technologický objekt Positioning Axis. Tato

konfigurace obnáší nastavení výstupů PLC, snímačů krajních poloh, mechanických vlastností a dynamiky dané osy atd.

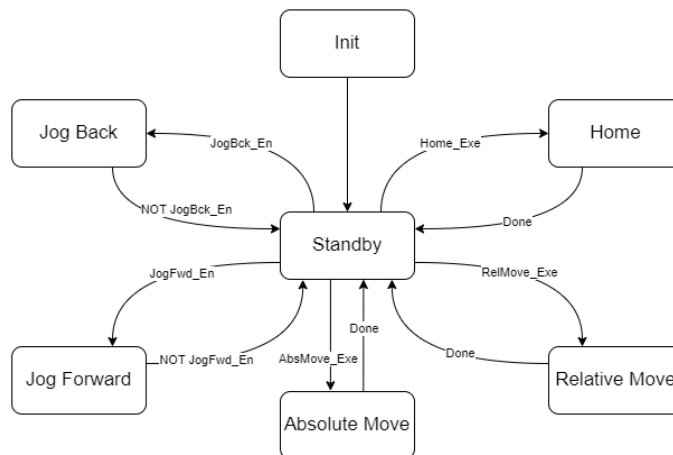


Obr. 8.5: Struktura řízení reálného motoru

Funkce Motion Control umožňují ovládat manipulátor v jednotlivých osách. Funkce MC_Home zajistí najetí manipulátoru v dané ose na inicializační polohu a tuto polohu uloží jako počátek. MC_MoveRelative definovanou rychlostí posune manipulátor o zadanou vzdálenost od aktuální polohy. MC_MoveAbsolute posune manipulátor na zadanou polohu. MC_MoveJog způsobí stálý pohyb manipulátoru dopředu/dozadu, dokud je aktivní vstupní signál.

Funkce Motion Control však fungují pouze s krokovými motory. Pohyb vertikální osy manipulátoru je ale zajištěn DC motorem. Bylo tak potřeba vytvořit kompletně nové řízení tohoto motoru, které ale musí fungovat identicky, aby byl zaručen cíl stanovený pro funkční blok AxisControl, tj. programátor už neřeší, jaký motor řídí, protože se pro něj všechny chovají totožně.

Digitální dvojčte, stejně jako vertikální osa, nedokáže využít funkce Motion Control, ale přesto se musí chovat totožně. Řízení je realizováno stavovým automatem Moorova typu, který je zobrazen v následujícím obrázku (Obr. 8.6: Stavový automat řízení motorů) a zajišťuje, že v jednom okamžiku probíhá jediná funkce. Jednotlivé stavy zajišťují funkcionalitu, která je totožná s odpovídajícími funkcemi Motion Control. Aby se vytvořený program choval stejně jako systémové funkce Motion Control, je zapotřebí složitější podmínky přechodů mezi stavy. Zobrazený konečný stavový automat je tak pouze ilustrační.



Obr. 8.6: Stavový automat řízení motorů

8.3.2 Manipulator

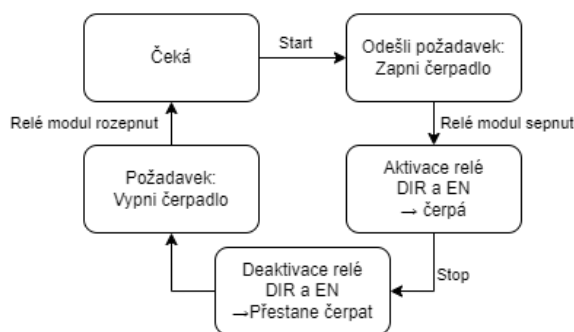
Funkční blok Manipulator slouží pro řízení tříosého manipulátoru. Stará se tak o koordinaci všech tří motorů. Využívá tedy tři funkce AxisControl.

8.3.3 PumpControl

Tento funkční blok se stará o obsluhu všech čerpadel. Čerpadla jsou připojena k relé modulu, který s PLC komunikuje pomocí Modbus. Tato komunikace ale není real-time, proto je nutné obsluhu čerpadel provádět bezpečně a zamezit nechtěnému sepnutí či pozdnímu vypnutí čerpadel.

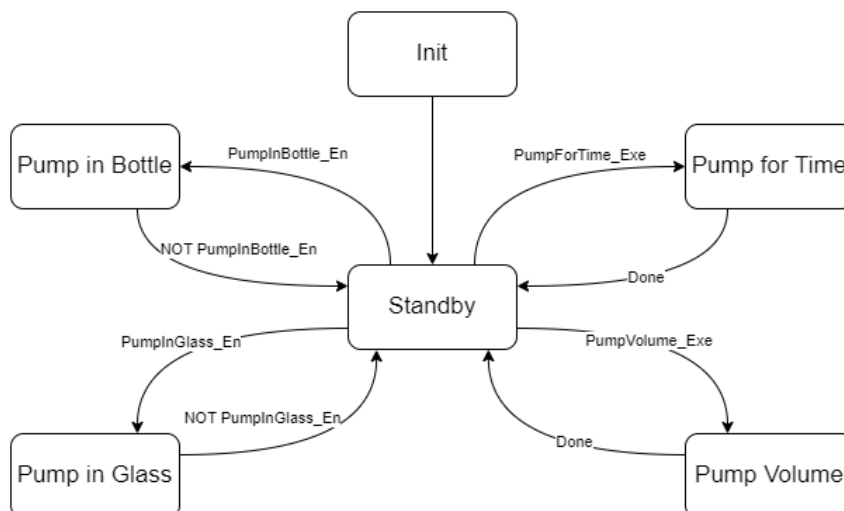
Postup řízení čerpadel je potom následující. PLC nejprve odešle jednomu z relé modulů požadavek o sepnutí jednoho relé. Poté dochází k pollingu, kdy se PLC opakovaně každých 100 ms dotazuje, zda už je dané relé sepnuto. Když modul odpoví, že je relé v režimu sepnuto, je potřeba zkontrolovat, že je sepnuto správné relé a také jediné relé. Pokud je toto splněno, dojde k výběru směru čerpání (do láhve/do skleničky) pomocí externího relé DIR a následně sepnutí dalšího externího relé EN, které přivede napájení danému čerpadlu. Po načerpání požadovaného množství, se postup vykoná opačným směrem, protože jenom externí relé EN a DIR jsou připojeny přímo na rychlé vstupy PLC a nevzniká tak časová prodleva.

Kdyby tento postup nebyl implementován, vlivem prodlevy během komunikace mezi PLC a relé modulem, by docházelo k čerpání špatných objemů. V krajních případech by mohlo dojít k tomu, že při současném požadavku na sepnutí relé EN i relé v modulu, by bylo stále zvoleno původní, nyní už nežádoucí, čerpadlo. Signál EN by na toto čerpadlo okamžitě přivedl napájení a k přepnutí na nové čerpadlo by došlo až se zpožděním. Došlo by tak k rozlití kapaliny mimo skleničku do prostoru uvnitř skladové buňky.



Obr. 8.7: Bezpečná obsluha čerpadel

Když je zajištěn způsob bezpečné obsluhy čerpadel, je možné implementovat funkce, které čerpadla využívají. Tyto funkce jsou podobně jako u AxisControl realizovány v Moorově stavovém automatu, který zajišťuje, že v jednom okamžiku probíhá jediná funkce.



Obr. 8.8: Stavový automat řízení čerpadel

Stav Pump for Time zajišťuje čerpání tekutiny po zadanou dobu. Pump Volume potom načerpá přesný objem v mililitrech. Pump in Glass a Pump in Bottle fungují podobně jako Jog Back/Forward u řízení motorů (viz kapitola 8.3.1), tzn. čerpadlo čerpá do skleničky, nebo opačným směrem do láhve, dokud je sepnutý vstupní signál.

Interface tohoto funkčního bloku je zajištěn UDT Pumps.

Pumps						
	Name	Data type	Comment
1	State	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Zpětná vazba
2	Done	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Čerpání dokončeno
3	Error	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	nastala chyba
4	Standstill	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Všechna čerpadla stojí
5	Running	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Některé čerpadlo běží
6	State	Array[1..16] of Bool	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Stav jednotlivých čerpadel
7	OneSlotState	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Stav právě sepnutého čerpadla
8	NumberOfActive	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Počet sepnutých relé
9	ManipNotOnPos	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Varování, že manipulátor nestojí pod slotem
10	Cmd	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Řízení
11	Halt_Exec	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Zastav (pause)
12	PumpForTime_Exec	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Čerpání daný čas
13	PumpVolume_Exec	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Čerpání daného množství
14	PumpInGlass_En	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Čerpání do skleničky
15	PumpInBottle_En	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Čerpání do láhve
16	OverrideSafety	Bool	fa: <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Vyřadí zabezpečení (manipulátor není pod slotem)
17	Param	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Parametry
18	PumpForTime	Real	0..C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jak dlouho [ms]
19	PumpVolume	Real	0..C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jaké množství [ml]
20	Slot	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jaké čerpadlo

Obr. 8.9: UDT Pumps

8.4 Funkční bloky podle standardu ANSI/ISA-S88

Ve složce S88 se nachází funkční bloky, které přímo souvisí se standardem ANSI/ISA-S88. Jedná se o realizaci zmíněné stromové struktury procedurálního modelu (Obr. 7.3).

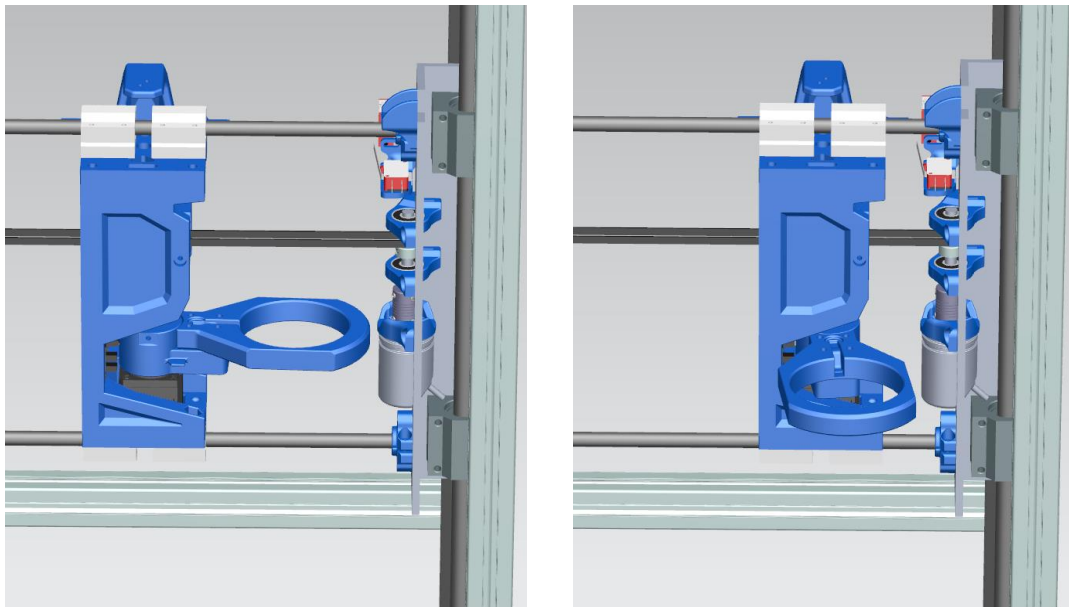
8.4.1 Fáze transportu – Phase_Transport

Fáze transportu se stará o pohyb manipulátoru na zvolené pozice. Jako vstupní parametr vyžaduje informaci o požadované poloze. Ta může být předána ve formě souřadnic, nebo jako číslo skladové pozice. Dále je potřeba zadat požadované rychlosti jednotlivých os.

PhaseTransport						
	Name	Data type	Comment
1	State	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Zpětná vazba
2	Cmd	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Řízení
3	Param	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Parametry
4	ToCoord/ToSlot	Bool	fal:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Přepínání mezi najetím na slot, nebo na souřadnice
5	ReqSlot	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Na jaký slot
6	ReqCoord	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Na jaké souřadnice
7	XCoord	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	YCoord	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	ZCoord	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	Force_RotToLin	Bool	fal:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jakým stylem - 0 = rotační pohyb, 1 = lineární pohyb. Defaultně se určí automaticky, můžu vynutit z HMI.
11	Speed	Struct	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Jakou rychlostí
12	XSpeed	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	YSpeed	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	ZSpeed	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Obr. 8.10: UDT fáze transportu

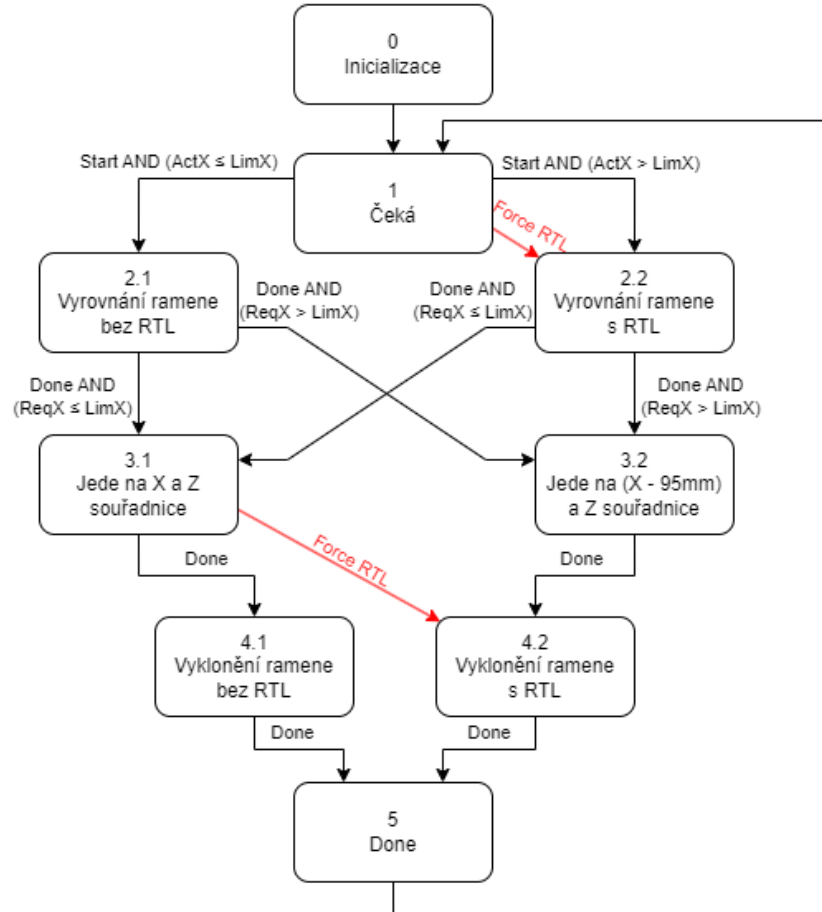
Jelikož je potřeba zajistit pohyb do komplikovaných pozic na úplném okraji rozsahu, je výsledný konečný stavový automat Moorova typu (Obr. 8.12) poměrně složitý.



Obr. 8.11: Transport do pozice na okraji manipulačního prostoru

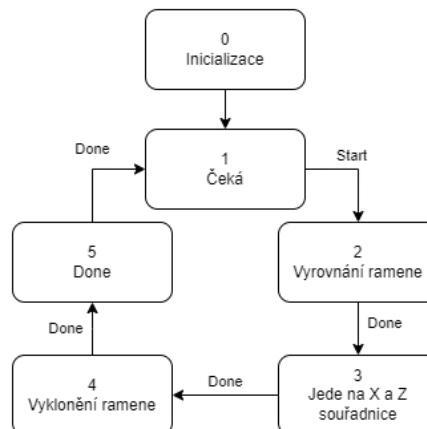
V těchto krajních pozicích se musí uplatnit transformace, při které se kombinuje rotační pohyb vyklánění ramene a lineární pohyb v horizontální ose manipulátoru, aby

byl výsledný pohyb koncového bodu ramene lineární. Tuto transformaci zajišťuje funkční blok RotToLin (RTL) detailněji popsáný v kapitole 8.5.1. Transformace se aktivuje, kdykoliv horizontální osa překročí polohu označenou jako LimX. Využití transformace se dá také vynutit aktivováním tlačítka Force RTL na dotykovém operátorském panelu.



Obr. 8.12: Stavový automat fáze transportu

Tento stavový automat je poměrně složitý. Kdybychom nebrali v úvahu ony hazardní krajní polohy, stavový automat by se výrazně zjednodušil. Prakticky to však není možné.



Obr. 8.13: Zjednodušený stavový automat fáze transportu

8.4.2 Fáze čerpání – Phase_Filling

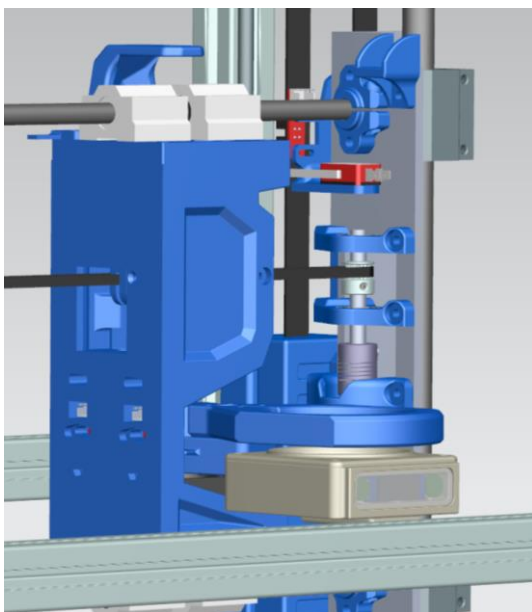
Fáze čerpání zajišťuje načerpání tekutiny do skleničky. Využívá pro to stavu Pump Volume z funkčního bloku PumpControl (viz kapitola 8.3.3). Pokud je místo reálné buňky aktivní digitální dvojče, dochází k přepočtu na čas a čerpání je nahrazeno časovačem. Kdyby digitální dvojče mělo zajišťovat také simulaci kapalin, bylo by výpočetně velmi náročné.

8.4.3 Fáze nabrání skleničky – Phase_Load

Úkolem této fáze je po najetí na startovní platformu vyzvednout manipulátorem předanou skleničku. To je zajištěno aktivováním relativního posunu manipulátoru ve vertikálním směru o 55 mm.

8.4.4 Fáze vyložení skleničky – Phase_Unload

Fáze vyložení skleničky na rozdíl od fáze nabrání skleničky využívá pohyb na absolutní polohu. Tato poloha se nachází těsně nad platformou sloužící pro vkládání skleničky. Na první pohled tento pohyb vypadá stejně jako relativní pohyb o 30 mm směrem dolů. V tomto případě je absolutní pohyb výrazně bezpečnější.

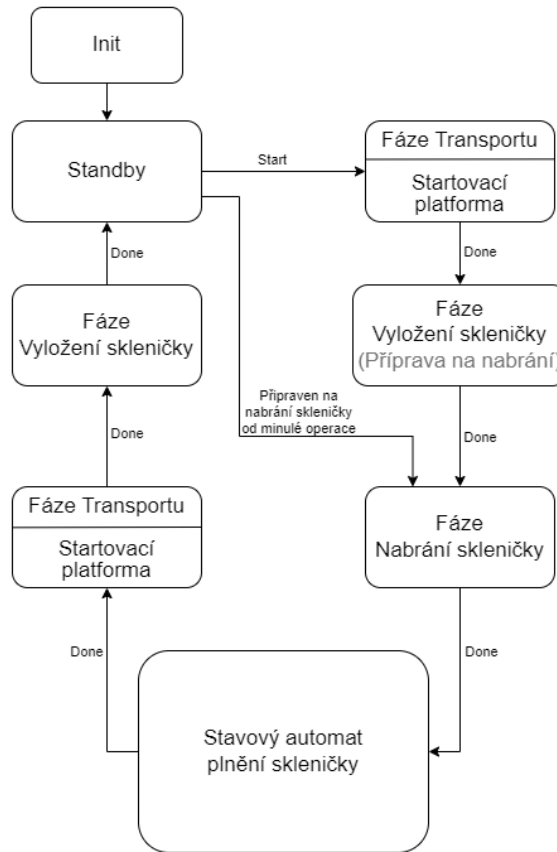


Obr. 8.14: Manipulátor v pozici vyložení

Fáze vyložení probíhá příliš blízko samotné platformy pro předání skleničky. Fázi vyložení předchází fáze transportu. Kdyby tato fáze z nějakého důvodu najela například o 10 mm níž, než měla, aktivací relativního pohybu by se tato chyba přenesla i do fáze vyložení a došlo by ke kolizi s platformou. Využitím pohybu na absolutně definovanou polohu dojde k eliminaci všech nepřesností pozice manipulátoru a ke kolizi dojít nemůže.

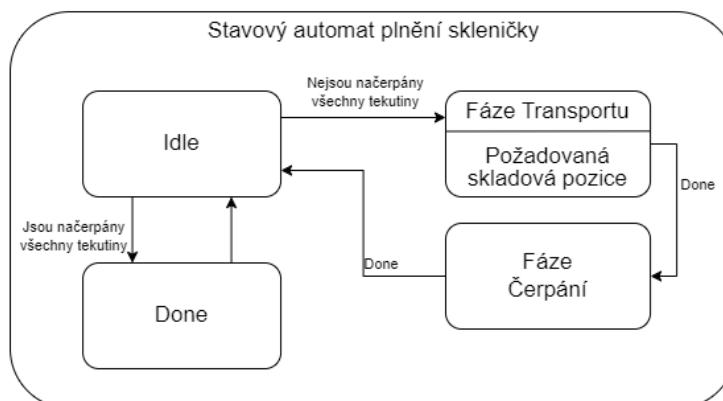
8.4.5 Operace přípravy nápoje – Oper_DrinkPrep

Jak je z modelu procedurálního řízení (Obr. 7.3) zřejmé, operace zastřešuje všechny fáze a řídí funkci celé jednotky. Je realizována opět konečným stavovým automatem Moorova typu.



Obr. 8.15: Stavový automat operace přípravy nápoje

Stavový automat operace přípravy nápoje obsahuje vnořený stavový automat plnění skleničky. Není totiž předem známo, zda uživatel bude chtít načerpat jednu tekutinu, nebo například čtyři. Tento vnořený stavový automat se tak cyklicky vykonává, dokud nejsou splněny všechny požadavky uživatele.



Obr. 8.16: Stavový automat plnění skleničky

Operace předpokládá, že je manipulátor už inicializován, tj. byl vykonán proces homování. Tento proces nelze automatizovat a musí jej vykonat pověřená osoba v manuálním režimu. Když je manipulátor poprvé spuštěn a nemá inicializované osy, tzn. nezná svoji polohu, může se nacházet v poloze, která je nebezpečná. Kdyby se proces inicializace manipulátoru vykonával automaticky, mohlo by dojít ke kolizi se skladovými pozicemi či s konstrukcí jednotky.

8.4.6 Procedura přípravy nápoje

Podle modelu procedurální výroby standardu ANSI/ISA-S88 tvoří procedura nejvyšší stupeň řízení jednotky. Nestará se jen o samotné řízení výroby, ale řídí celou skladovou jednotku včetně všech souvisejících funkcí.

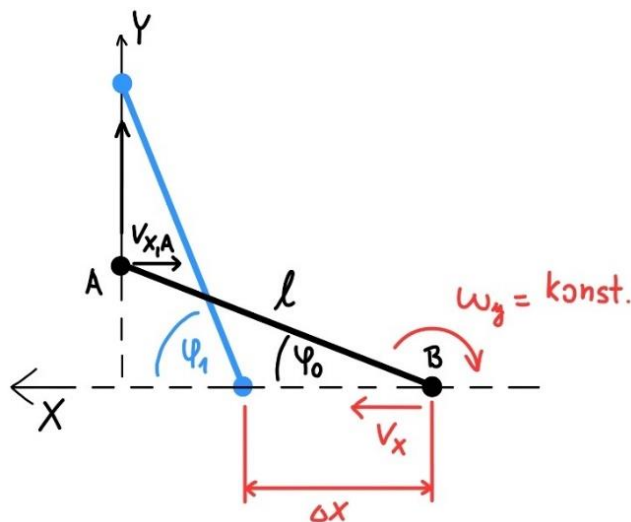
8.5 Pomocné funkce

Funkce a funkční bloky, které přímo nesouvisí se samotným řízením, ale jsou potřebné pro správný chod programu jsou umístěny ve složce Auxiliary, v překladu pomocné.

8.5.1 RotToLin

Funkční blok RotToLin se stará o transformaci rotačního pohybu v ose Y a lineárního pohybu v ose X na lineární pohyb v ose Y. Řeší tak problém s transportem do krajních poloh manipulačního rozsahu (Obr. 8.11)

Funkčnímu bloku je zadán požadovaný cílový úhel natočení v ose Y, společně s požadovanou rychlostí otáčení ramene. Na základě těchto požadavků je vypočten relativní posun v ose X (lineární) a průběžně je upravována lineární rychlost tak, aby byl výsledný pohyb v ose Y skutečně lineární.



Obr. 8.17: Diagram pro výpočet transformace pohybů

Výpočet relativního posunu i rychlosti v ose X je založen na základních znalostech z oblasti trigonometrie. Pro výpočet relativního posunu Δ_X slouží vzorec:

$$\Delta_X = l \cdot [\cos(\varphi_0) - \cos(\varphi_1)] \quad (8.1)$$

Výpočet postupné změny lineární rychlosti manipulátoru v ose X vychází z toho, že dokážeme rozložit pohyb koncového bodu A na lineární složky v osách X a Y:

$$X_A(t) = l \cdot \cos(\varphi(t))$$

$$Y_A(t) = l \cdot \sin(\varphi(t))$$

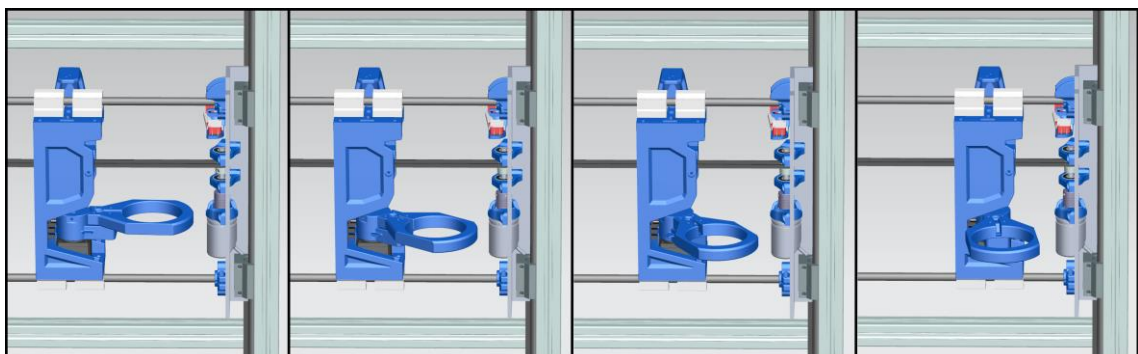
Derivací této rovnice poté získáme rychlost koncového bodu A v ose X:

$$v_{X,A} = \frac{dX_A(t)}{dt} = -l \cdot \omega_Y \cdot \sin(\varphi(t))$$

Pokud chceme, aby se z rotačního pohybu uplatnila jen lineární složka v ose Y, musíme zajistit, aby se lineární složka v ose X vyrušila:

$$v_X(t) = -v_{X,A}(t) = l \cdot \omega_Y \cdot \sin(\varphi(t)) \quad (8.2)$$

- Kde:
- Δ_X – Relativní posun manipulátoru v ose X [mm]
 - l – Délka ramene manipulátoru [mm]
 - φ_0 – Počáteční natočení ramene manipulátoru [°]
 - φ_1 – Cílové natočení ramene manipulátoru [°]
 - $\varphi(t)$ – Průběžné natočení ramene manipulátoru [°]
 - $X_A(t)$ – Lineární složka pohybu koncového bodu ramene v ose X [mm]
 - $Y_A(t)$ – Lineární složka pohybu koncového bodu ramene v ose Y [mm]
 - ω_Y – Požadovaná rychlost otáčení ramene [rad/s]
 - $v_{X,A}(t)$ – Lineární složka rychlosti pohybu koncového bodu ramene v ose X [mm/s]
 - $v_X(t)$ – Rychlost pohybu manipulátoru v ose X [mm/s]



Obr. 8.18: Aplikace pohybové transformace

8.5.2 Ostatní

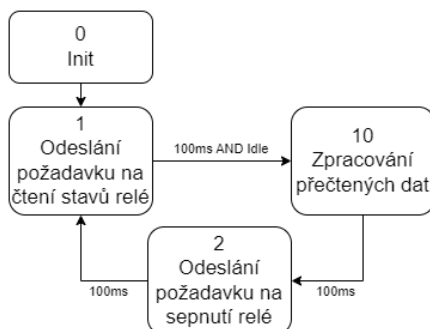
Mezi další funkce patří například OnPos, která vrací čísla {-1, 0, 1} podle toho, jestli manipulátor stojí před, na, nebo za požadovanou polohou s parametrizovanou necitlivostí. Funkční blok Arr16_Sort slouží pro seřazení požadované kombinace tekutin, aby měl manipulátor optimální trasu. HMI_Control se stará o komunikaci HMI a PLC a umožňuje tak ovládat skladovou jednotku pomocí dotykového displeje. CapSens zajišťuje komunikaci s IO-Link moduly pro kapacitní snímače hladiny v lahvích. IsInStock v kombinaci s FindSlot provádí kontrolu obsahu skladu.

8.6 Komunikace

Kromě PLC S7-1200 skladová buňka využívá také rozšiřovací moduly, se kterými musí PLC komunikovat.

8.6.1 Modbus

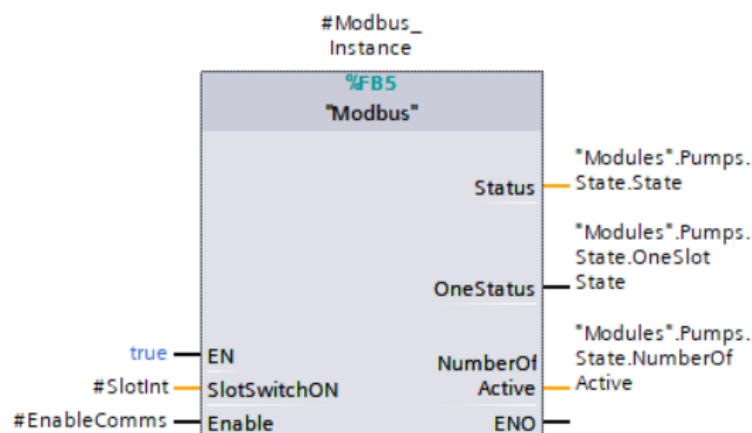
Komunikace mezi PLC a relé moduly pro řízení peristaltických čerpadel je zajištěna s využitím protokolu Modbus. Relé moduly tvoří servery a PLC představuje klienta. Komunikace potom probíhá cyklicky a je realizována stavovým automatem.



Obr. 8.19: Stavový automat komunikace Modbus

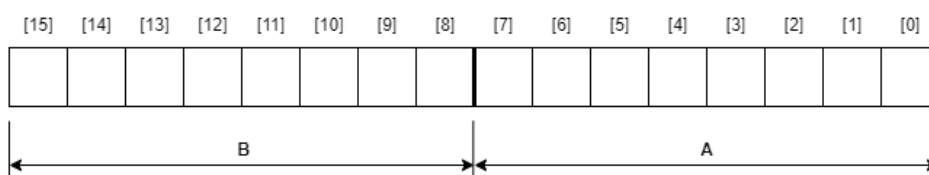
Vykonání tohoto postupu zajišťuje funkční blok Modbus. Jako vstupní parametr je předána informace o tom, jaké relé je potřeba sepnout (celé číslo int). Další parametr je Enable, který aktivuje komunikaci. Funkční blok následně vykoná definovaný postup (Obr. 8.19).

Mezi výstupní hodnoty bloku patří pole Status, ve kterém je uložen aktuální stav všech relé. Dalším výstupem je signál OneStatus, který říká, jestli je sepnuté to jedno zadané relé. Posledním výstupem je NumberOfActive. Tato hodnota reprezentuje počet aktuálně sepnutých relé.



Obr. 8.20: Funkční blok Modbus

Zprávy musí mít definovaný formát zadaný výrobcem relé modulů (Obr. 8.21). Při požadavku o sepnutí relé zapisujeme do druhého registru modulu (tedy na adresu 40 003) 16bit informaci. V prvních osmi bitech (A) říkáme, které relé chceme sepnout (1), a které rozepnout (0). V následujících osmi bitech (B) říkáme, u kterého relé povolíme změnu stavu. V mém programu je všech těchto osm bitů permanentně nastaveno na 1, protože chci mít vždy možnost měnit stavy všech relé.



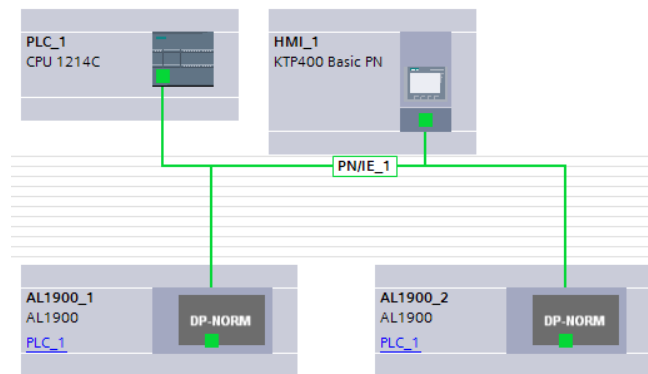
Obr. 8.21: Komunikační rámec pro zápis do relé modulu

Pokud chceme zjistit aktuální stav relé, musíme přečíst data z prvního registru modulu (tedy z adresy 40 002). Tato data jsou uložena do datového bloku DB6. Data z DB6 se následně zpracují a posílají se na výstup funkčního bloku.

Vstupní a výstupní hodnoty funkčního bloku jsou předávány pomocí UDT Pumps (Obr. 8.9).

8.6.2 IO-Link

Pro hlídání výšky hladiny se využívají kapacitní senzory KQ5100. Ty jsou připojeny do IO-Link Master modulu AL1900. Ten s PLC komunikuje pomocí Profinetu. Pro zprovoznění komunikace je potřeba provést konfiguraci v záložce Devide configuration.



Obr. 8.22: Připojení IO-Link modulů

TIA Portal si následně čtení dat ze senzorů zajistí automaticky. Tato data je však poté potřeba zpracovat. O to se stará funkce CapSensRaw, která ukládá zpracované hodnoty do pole KQ5100.Sensor.Raw.

Funkční blok CapSensControl se poté stará o další zpracování těchto dat. Zajišťuje kalibraci senzorů, přepočítá naměřené hodnoty na procenta zbývajícího objemu a vyvolává varování o nízké hladině tekutiny.

8.6.3 Komunikace s nadřazeným systémem

Komunikaci s nadřazeným systémem má za úkol zajistit funkce Comms, která je založená na protokolu S7 Protocol. Tato funkce zpracovává přijatá data, která jsou nadřazeným systémem zapisována do oblasti B datového bloku Comms_DB (Obr. 8.23). Tato data následně zpracuje a poté upraví hodnoty v oblasti A datového bloku, kterou nadřazený systém čte. Tento datový blok musí mít konzistentní strukturu pro všechny jednotky testbedu Self-Acting Barman.

Comms_DB									
	Name	Data ty...	Comment
1	Static								
2	Idle	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	Ready	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	Busy	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	Done	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	A
6	Error	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	Occupied	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	R7	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	R8	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	SetReady	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	Start	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	ResetDone	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	SetOccupied	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	ResetOccupied	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	B
15	W6	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16	W7	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17	W8	Bool	...	fa:l	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	DW1	DWord	...	16	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19	DW2	DWord	...	16	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
20	DW3	DWord	...	16	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
21	DW4	DWord	...	16	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
22	DW5	DWord	...	16	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	JobID

Obr. 8.23: Datový blok pro komunikaci s nadřazeným systémem

Pro vkládání dat do datového bloku a následný sběr dat nadřazený systém využívá příkazy PUT a GET. V HW konfiguraci PLC je potřeba tyto příkazy povolit, jinak by byl nadřazenému systému odepřen přístup.

8.7 Simulační režimy skladové jednotky

Skladová jednotka umožňuje dva typy simulace. Režim simulace je možné přepnout na dotykovém operátorském panelu.

Prvním režimem je simulace na digitálním dvojčeti. V tomto režimu je fyzická jednotka deaktivována a veškeré příkazy jsou z PLC odesílány do programu Siemens NX, ve kterém běží simulace digitálního dvojčete.

Druhý typ simulace je určen pro ladění komunikace s okolními jednotkami v rámci testbedu Self-Acting Barman. Během této simulace jsou pohony reálné jednotky deaktivovány a probíhá simulace komunikace popsané v kapitole 8.6.3.

8.8 Digitální dvojče a reálné nasazení

Řídicí logika nejprve vznikla s využitím digitálního dvojčete. Díky tomu bylo možné během vývoje využít všech výhod popsaných v kapitole 1.2.1.

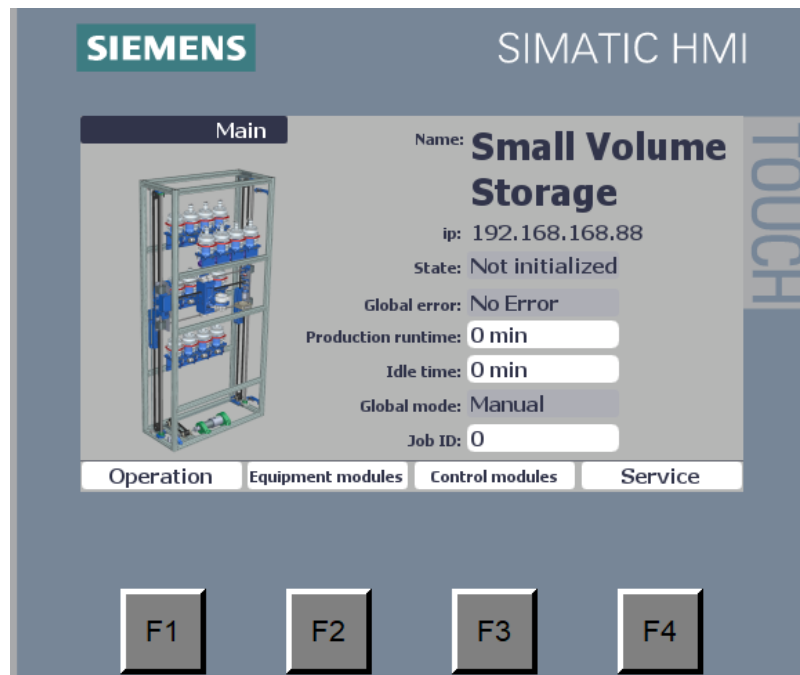
Po odladění programu a eliminaci všech hazardních stavů je možné implementaci aplikovat také na reálnou jednotku. Díky tomu, že je digitální dvojče velmi detailní, je následné reálné zprovoznění poměrně jednoduché a program vyžaduje minimální modifikace.

Řízení pohonů pracuje se standardními signály, tedy Enable (motor je aktivován), Direction (směr otáčení motoru) a Velocity (rychlost otáčení motoru). Tyto signály jsou odesílány do digitálního dvojčete a na základě nich dochází k pohybu motorů. Reálný manipulátor však používá krokové a DC motory. Signál Velocity, který je reálným číslem, je nutné transformovat na vhodný signál.

Krokový motor dostává informaci o požadované rychlosti signálem typu PTO (Pulse Train Output), tedy sledu impulzů, jejichž frekvence značí požadovanou rychlost. DC motor pro změnu rychlosti potřebuje signál PWM (Pulse Width Modulation). Tento signál je také sledem impulzů, tentokrát však rychlost motoru není zakódována ve frekvenci impulzů, ale v jejich střídě.

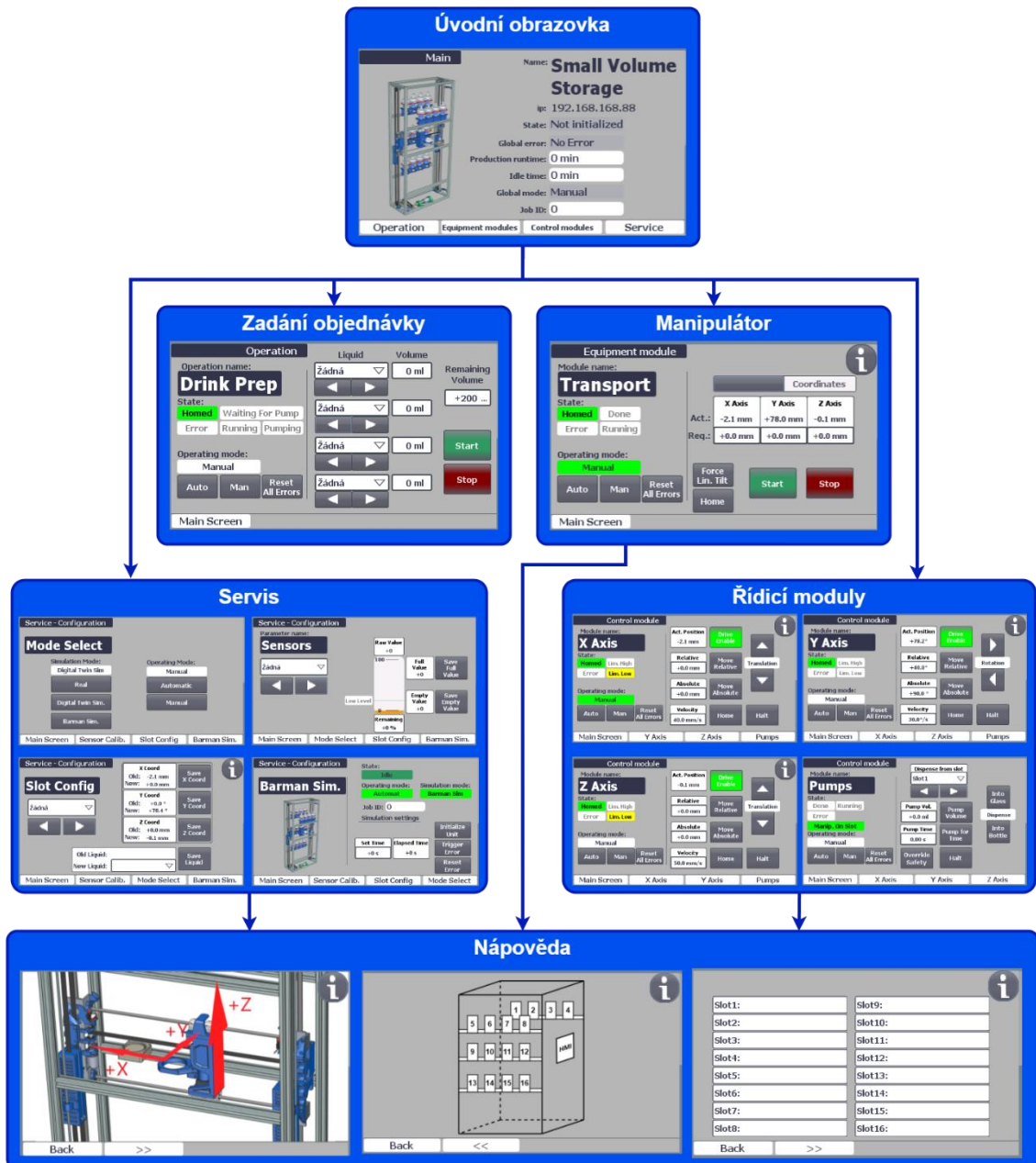
9. OPERÁTORSKÝ PANEL

Skladová jednotka je řízena dotykovým operátorským panelem Siemens KTP400 Basic. Ten disponuje 4“ displejem a čtyřmi fyzickými tlačítky. Tento panel označujeme jako HMI (Human – Machine Interface), tedy rozhraní mezi člověkem a strojem. Veškeré akce jsou řízeny dotykovými příkazy s výjimkou navigace mezi obrazovkami. Ta je realizována fyzickými tlačítky.



Obr. 9.1: Úvodní obrazovka HMI

Pomocí HMI může uživatel jednotku plně ovládat. Operátorský panel umožňuje v manuálním režimu řízení jednotlivých řídicích modulů (motory manipulátoru a čerpadla), je možné pohybovat s manipulátorem jako s celkem, ručně zadávat objednávky, či provádět servis skladové jednotky. Pomocí HMI se přepíná provozní režim jednotky. Manuál na obsluhu jednotky pomocí operátorského panelu je součástí přílohy (Příloha B - Provozní a servisní manuál skladové jednotky).



Obr. 9.2: Struktura obrazovek HMI

10. ZÁVĚR

V úvodní části práce jsou popsány principy Průmyslu 4.0, včetně prostředků potřebných k jejich realizaci. Definice Průmyslu 4.0 je nutná z toho důvodu, že skladová jednotka tekutin malého objemu, která je předmětem této práce, je součástí testbedu Self-Acting Barman, který je na těchto principech založen.

Následuje popis standardu ANSI/ISA-S88. Ten je nedílnou součástí této práce, protože jak skladová jednotka, tak celý testbed Barman je na základě tohoto standardu implementován.

Je popsán původní stav skladové jednotky společně s problémy, které s touto koncepcí souvisí. Využitím peristaltických čerpadel pro dávkování tekutin jsou uvedené problémy eliminovány. Pro realizaci tohoto způsobu dávkování však bylo zapotřebí modifikovat konstrukci a také elektrické zapojení jednotky. Vzniklo kompletní elektrické schéma, které je součástí přílohy (viz Příloha A - Elektrické schéma zapojení skladové jednotky). Díky těmto modifikacím je skladová jednotka výrazně spolehlivější a nehrozí poškození manipulátoru při dávkování tekutin. Dávkování zadaného objemu je navíc výrazně přesnější.

Dále je popsán vývoj komplexního digitálního dvojčete skladové jednotky, včetně jeho virtuálního zprovoznění. Řídicí logika, která je založena na standardu ANSI/ISA-S88 a je implementována na PLC S7-1200, nejprve vznikla na digitálním dvojčeti. Po odladění a eliminaci všech hazardních stavů byla následně aplikována také na reálnou jednotku. Díky tomu, že je digitální dvojče velmi detailní, bylo pro reálné nasazení nutné provést minimální modifikace programu.

LITERATURA

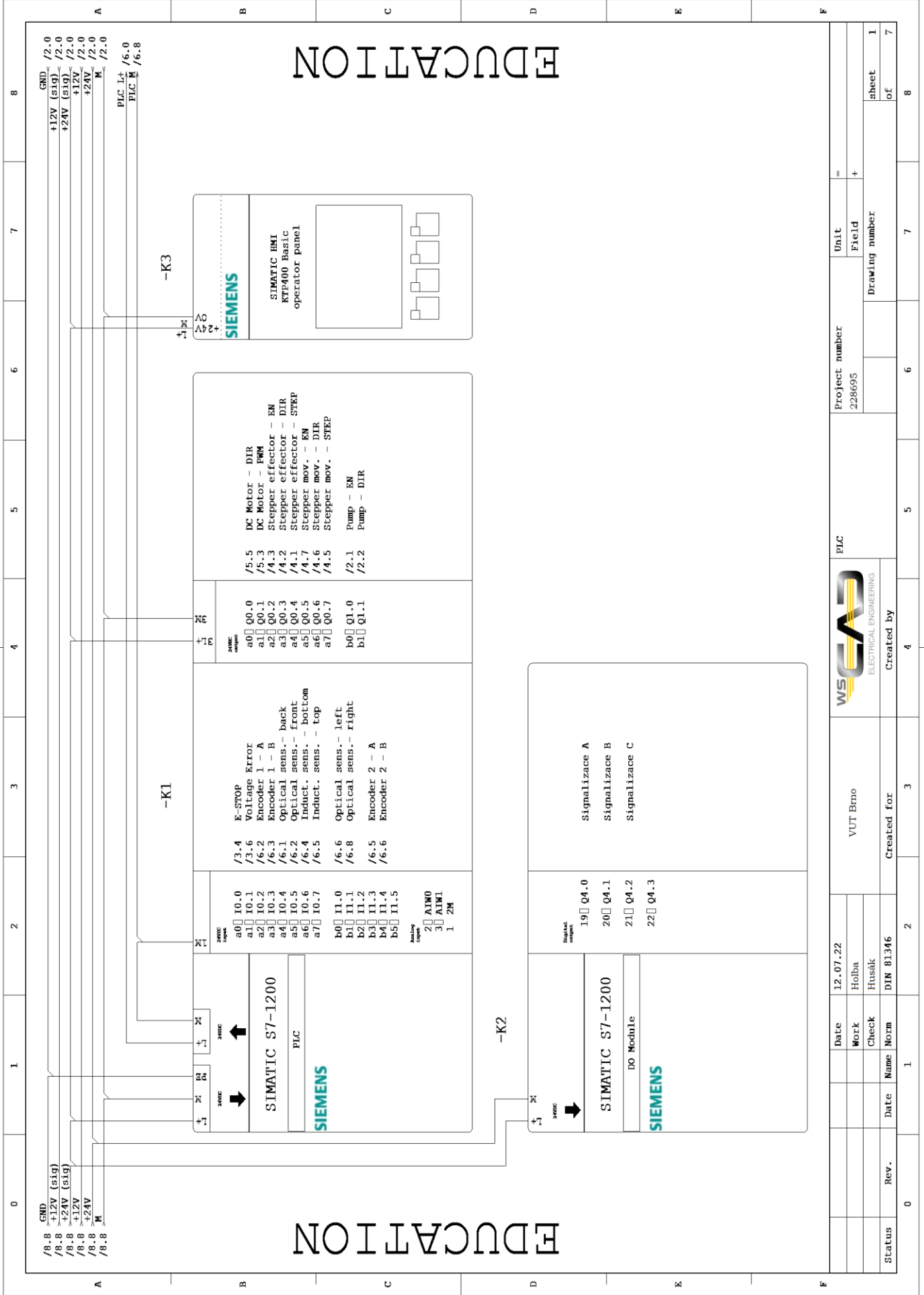
- [1] REJCHLÍK, Lukáš. Návrh, konstrukce a programové vybavení inteligentního skladu pro testbed Průmyslu 4.0 [online]. Brno, 2019, 85 s [cit. 2022-12-22]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/118406>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Václav Kaczmarczyk.
- [2] HERMANN, Mario a spol., Design Principles for Industrie 4.0 Scenarios: A Literature Review. ResearchGate [online]. Technická univerzita Dortmund, Fakulta strojní, 2015 [cit. 2022-12-25]. Dostupné z: https://www.researchgate.net/publication/307864150_Design_Principles_for_Industry_4.0_Scenarios_A_Literature_Review
- [3] Industry 4.0 and the fourth industrial revolution explained. In: I-scoop [online]. [cit. 2023-05-14]. Dostupné z: <https://www.i-scoop.eu/industry-4-0/>
- [4] ARM, Jakub. Prostředky průmyslové automatizace (BPC-PPA) - Úvodní přednáška. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2021.
- [5] Automa: Industrie 4.0 Asset Administration Shell: koncepce a využití v praxi [online]. 2019, 08. [cit. 2023-05-12]. Dostupné z: https://automa.cz/cz/casopis-clanky/industrie-4-0-asset-administration-shell-koncepce-a-vyuziti-v-praxi-2019_08_0_12444/
- [6] VDI Standards: Language for I4.0 Components – Structure of messages [online], 2020. VDI Verein Deutscher Ingenieure e.V. [cit. 2023-05-13]. Dostupné z: <https://www.vdi.de/en/home/vdi-standards/details/vdivde-2193-blatt-1-language-for-i40-components-structure-of-messages>
- [7] FORTIER, Paul a Howard MICHEL, 2003. Computer Systems Performance Evaluation and Prediction [online]. 2019. ScienceDirect [cit. 2022-12-25]. ISBN 9781555582609. Dostupné z: <https://www.sciencedirect.com/topics/computer-science/testbed>
- [8] BOCH, Jan. Autonomní sklad kapalin s chladicím systémem [online]. Brno, 2022, 63 s [cit. 2022-12-22]. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/142623>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce: Ing. Michal Husák
- [9] KACZMARCZYK, Václav a spol., 2018. An Industry 4.0 Testbed (Self-Acting Barman): Principles and Design [online]. ScienceDirect [cit. 2022-12-25]. ISSN 2405-8963. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S2405896318309108>

- [10] ISA-THE INSTRUMENTATION SYSTEMS, AND AUTOMATION SOCIETY. ANSI/ISA-88.01-1995, Batch Control, Part 1: Models and Terminology. 1995. ISBN 1-55617-562-0.
- [11] HUSÁK, Michal. Návrh a realizace laboratorní úlohy na téma "Standard ISA-S88" [online]. Brno, 2018 [cit. 2022-12-23]. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/111055>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Václav Kaczmarczyk,
- [12] Watson-Marlow Fluid Technology Solutions. How do peristaltic pumps work? [online]. [cit. 2022-12-28]. Dostupné z: <https://www.wmfts.com/en/support/pump-principles/how-do-peristaltic-pumps-work/>

SEZNAM PŘÍLOH

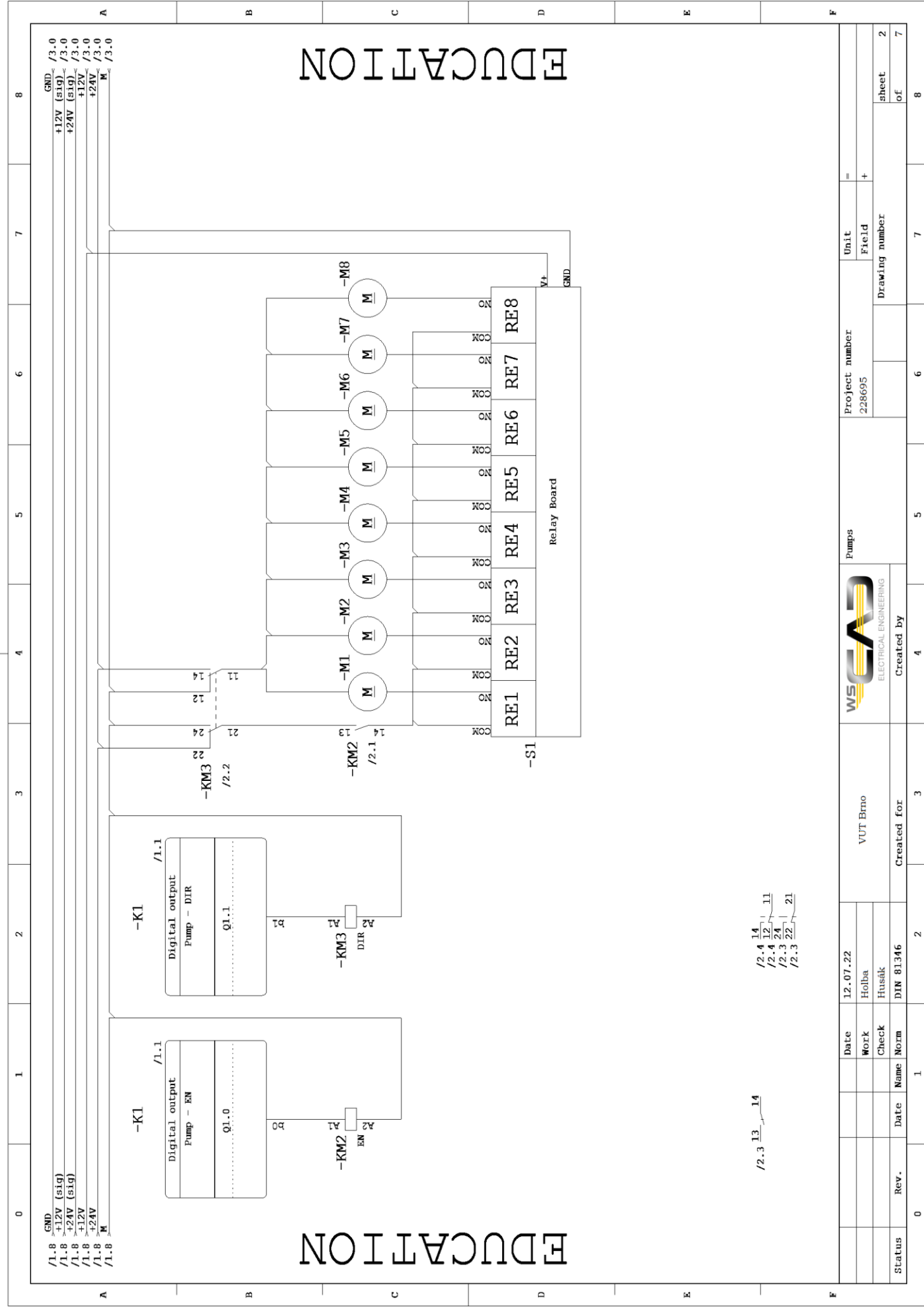
PŘÍLOHA A - ELEKTRICKÉ SCHÉMA ZAPOJENÍ SKLADOVÉ JEDNOTKY	62
PŘÍLOHA B - PROVOZNÍ A SERVISNÍ MANUÁL SKLADOVÉ.....	70
PŘÍLOHA C - PŘÍLOHY NA SD KARTĚ	81

Příloha A - Elektrické schéma zapojení skladové jednotky



Status	Rev.	Date	Name	Norm	Created for	Created by	Project number	Unit	Field	Drawing number	Sheet	of
	0	12.07.22	Holba		VUT Brno		228695	=	+		1	8
	1		Hutak								7	8





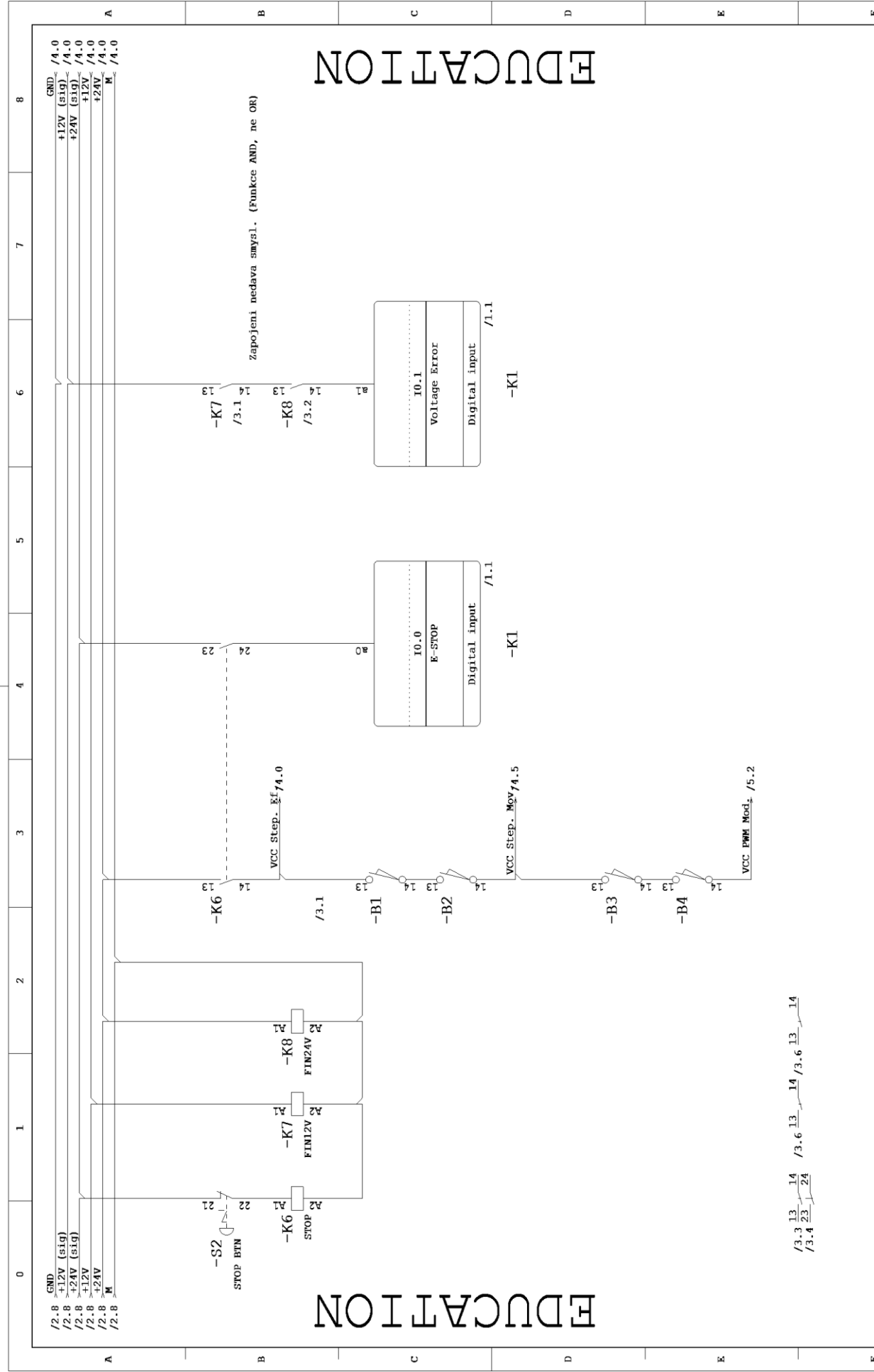
1.8	GND	/3.0
1.8	+12V (sig)	/3.0
1.8	+24V (sig)	/3.0
1.8	+12V	/3.0
1.8	+24V	/3.0
1.8	M	/3.0

/2.3 13 14
 /2.4 11 12 13
 /2.3 22 21

Status	Rev.	Date	Name	DIN 81346	Created for	VUT Brno	Pumps	Project number 228695	Unit	=
									Field	+
									Drawing number	
									sheet	2
									of	7
										8



Created by



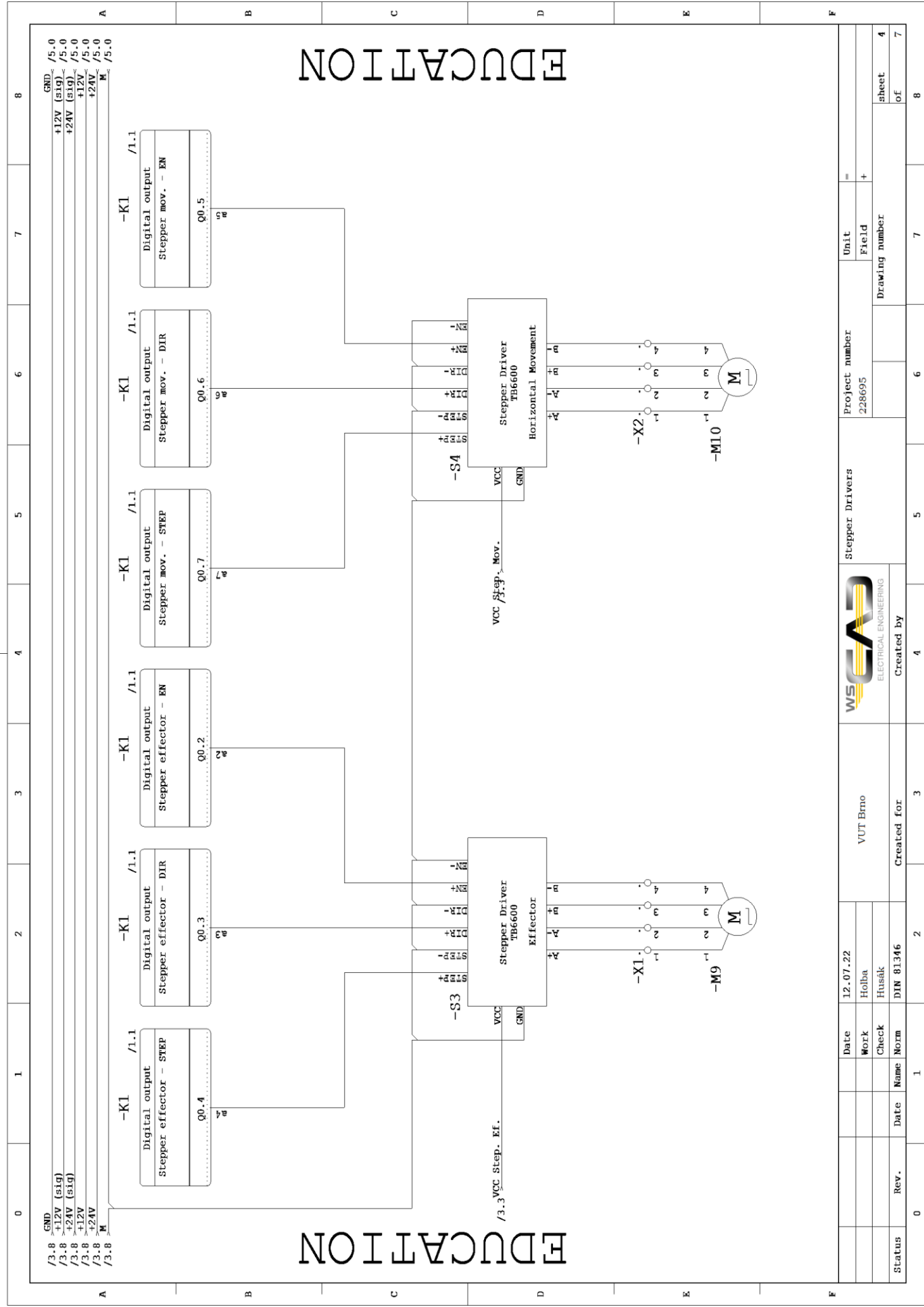
EDUCATION

EDUCATION

zapojeni nedava smysl. (Funkce AND, ne OR)

/3.3 13 14 /3.6 13 14
/3.4 23 24

Status	Rev.	Date	Name	Created for		Safety	Project number		Unit	Drawing number	sheet of
				VUT Brno			228695	=	Field		
0	1	12.07.22	Holba							7	8
			Husák								
			DIN 81346	Created by							



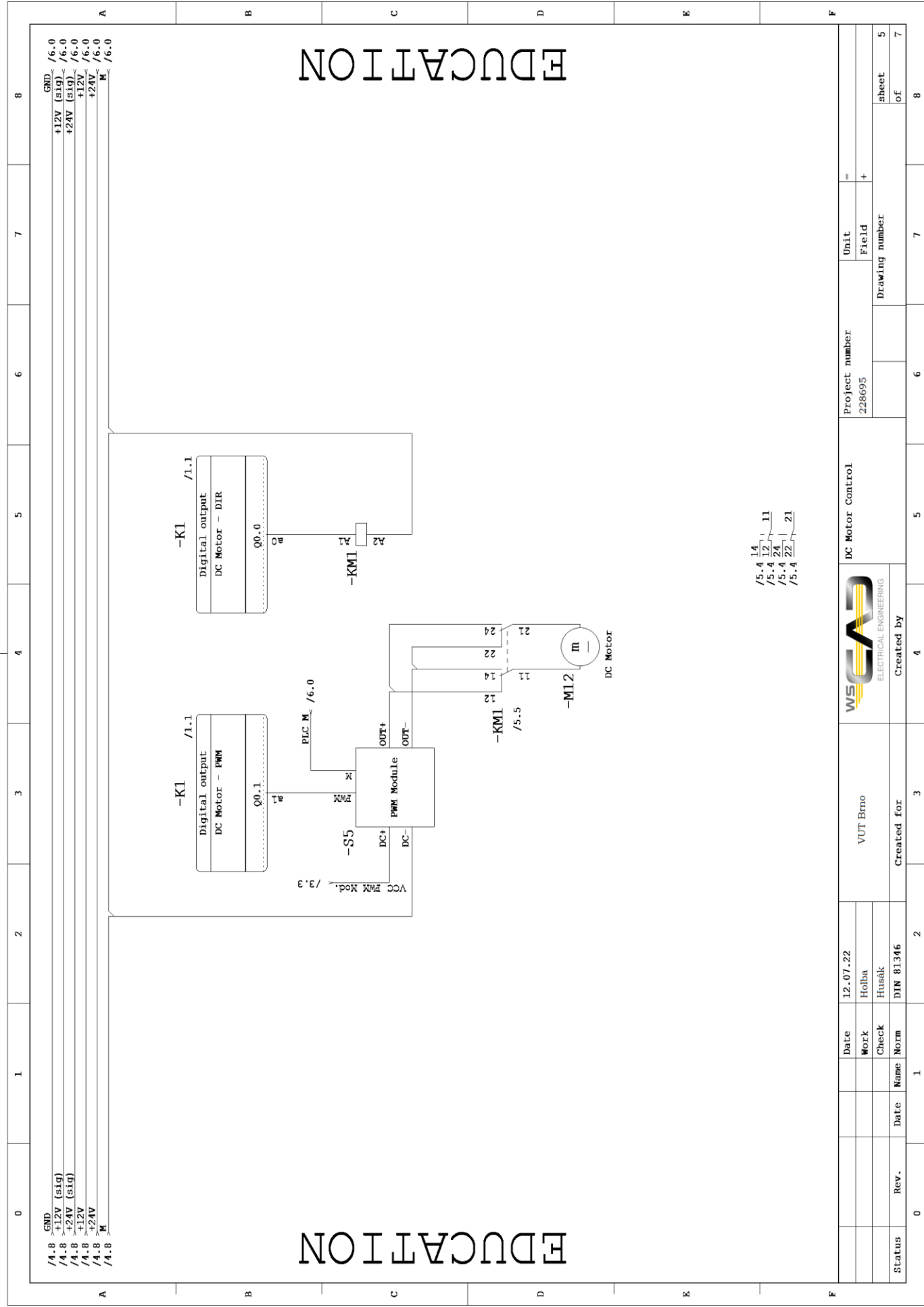
EDUCATION

EDUCATION

0	1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---	---

/3.8	GND	/3.8	+12V (sig)	/3.8	+24V (sig)	/3.8	+12V	/3.8	+24V	/3.8	M	/5.0
	+12V (sig)		+24V (sig)		+12V		+24V		M			

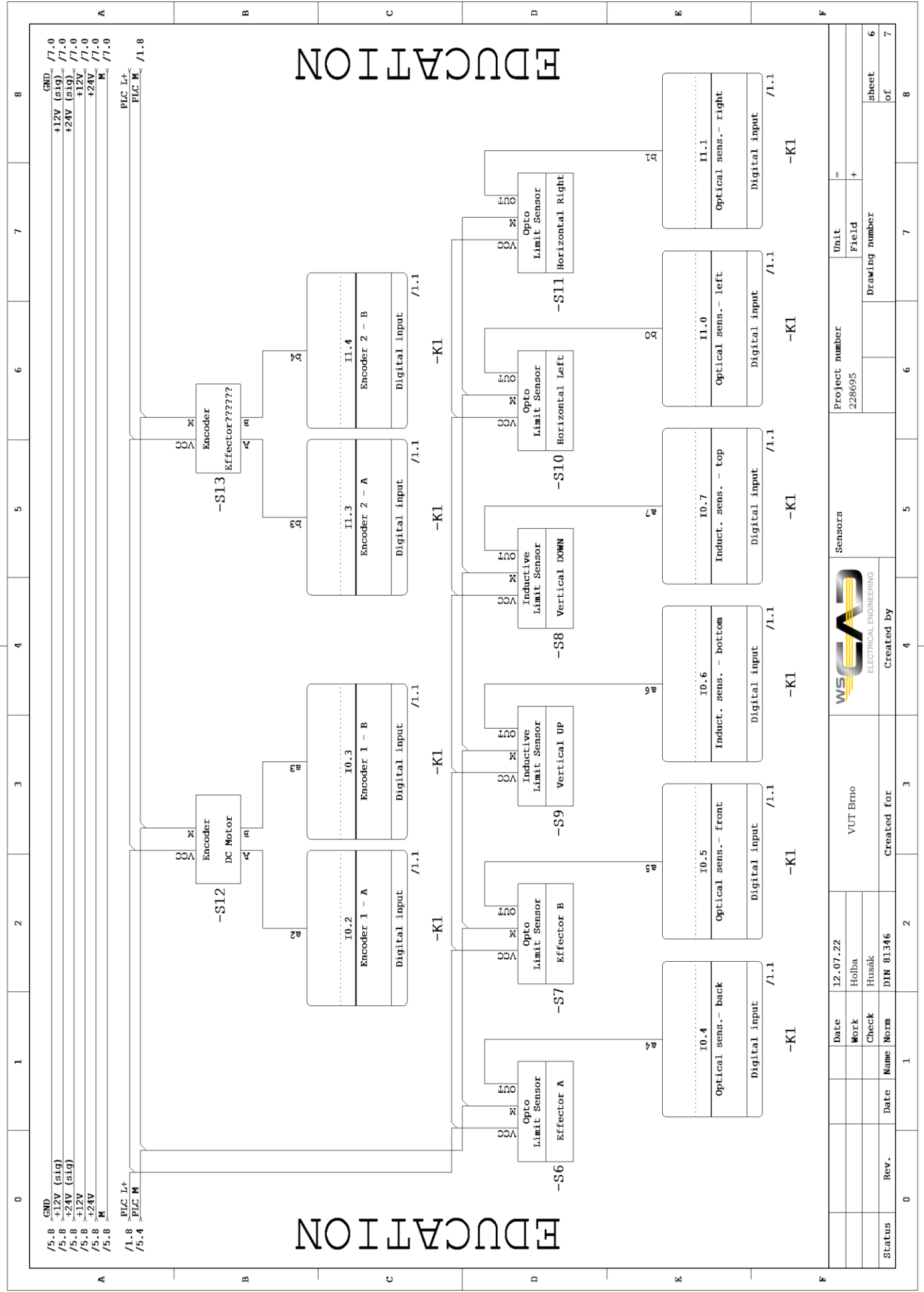
Date	12.07.22	VUT Brno		Project number	228695		Unit	=
Work	Holba			Stepper Drivers			Field	+
Check	Husák			Created by			Drawing number	
Name	DIN 81346	Created for					Sheet	4
Date							of	7
Rev.								8



4.8	GND	/6.0
4.8	+12V (sig)	/6.0
4.8	+24V (sig)	/6.0
4.8	+12V	/6.0
4.8	+24V	/6.0
4.8	M	/6.0

/5.4 14 11
 /5.4 12 11
 /5.4 24 21
 /5.4 22 21

Date	12.07.22	VUT Brno	DC Motor Control	Project number	228695	Unit	=
Work	Holba					Field	+
Check	Husák					Drawing number	
Norm	DIN 81346	Created for				Sheet	5
Rev.						of	7
							8

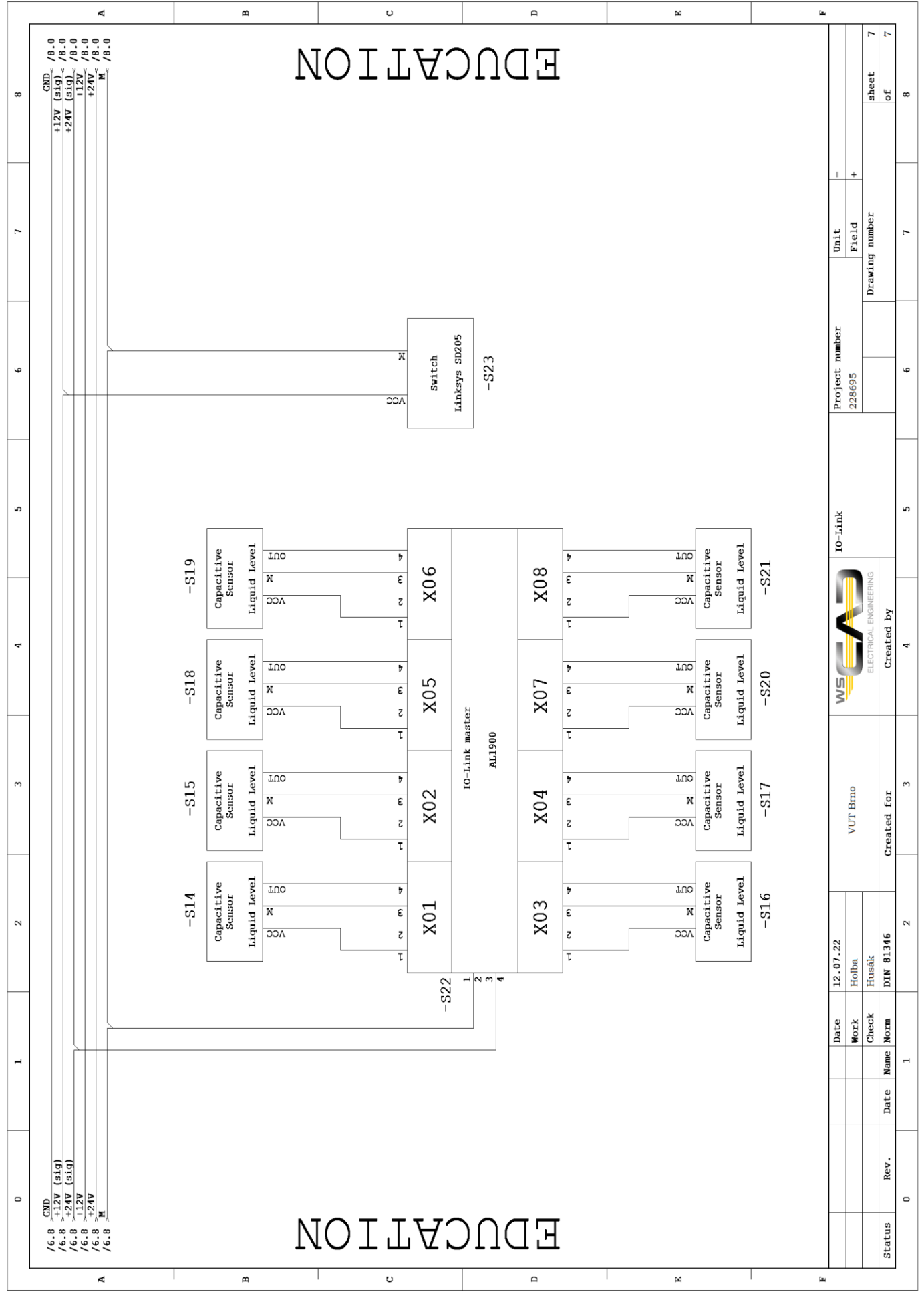


5.8	GND	/7.0
5.8	+12V (sig)	/7.0
5.8	+24V (sig)	/7.0
5.8	+12V	/7.0
5.8	+24V	/7.0
5.8	M	/7.0
1.8	PIC L+	
5.4	PIC M	/1.8

Date		12.07.22		Project number		228695	
Work		Holba		Unit		=	
Check		Husák		Field		+	
Name		DIN 81346		Created by		VUT Brno	
Date				Sensors			
Rev.				Created for			
Status				Drawing number			
				Sheet		6	
				of		7	

EDUCATION

EDUCATION



6.8	GND	/8.0
6.8	+12V (sig)	/8.0
6.8	+24V (sig)	/8.0
6.8	+12V	/8.0
6.8	+24V	/8.0
6.8	M	/8.0

EDUCATION

EDUCATION

Date	12.07.22	Project number	228695
Work	Holba	Unit	=
Check	Husák	Field	+
Norm	DIN 81346	Drawing number	
Date		Sheet	7
Name		of	7
Rev.			



VUT Brno

Created for

Created by

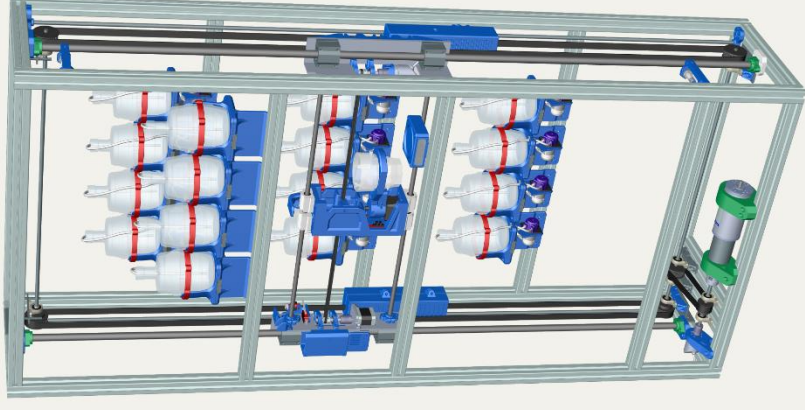
IO-Link

0	1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---	---

Příloha B - Provozní a servisní manuál skladové jednotky

SKLADOVÁ JEDNOTKA TEKUTIN MALÉHO OBJEMU


PROVOZNÍ MANUÁL



 Jan Holba
228695

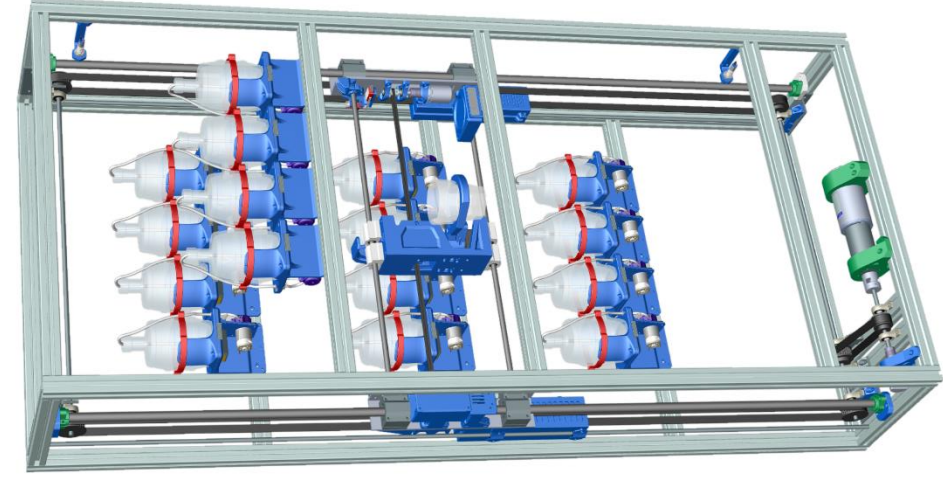
 xholba05@vutbr.cz
jholba1409@gmail.com

 FEKT VUT Brno

 2023



OBSAH



1	Úvodní informace
1	Provozní režimy
2	Navigace systémem
4	Nastavení objednávky
6	Řízení manipulátoru
8	Řízení modulů
11	Servis a konfigurace
15	Časté akce

ÚVODNÍ INFORMACE

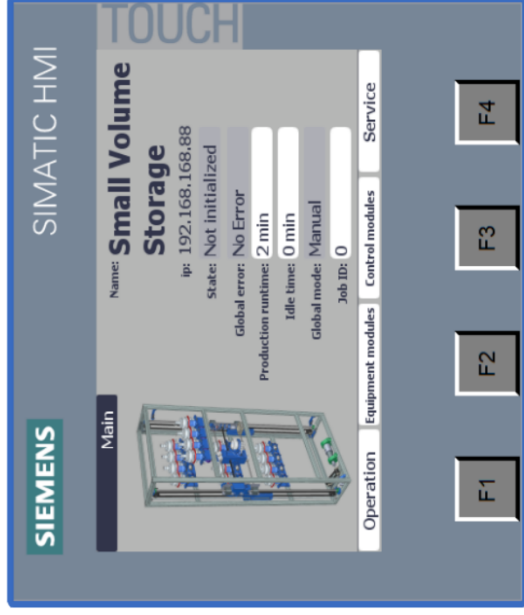
- **Úkol skladové jednotky:**
 - Naplnit skleničku požadovanou kombinací nápojů
- V kombinaci s dalšími jednotkami tvoří testbed Self-Acting Barman
- **Cíl testbedu:**
 - Vytvořit automatizovaný způsob kompletní přípravy míchaných nápojů s využitím principů Průmyslu 4.0
- **Zbylé jednotky:**
 - Sklad skleniček, Výrobník sycené vody, Sklad tekutin většího objemu, Shaker a Drtič ledu

PROVOZNÍ REŽIMY

- **Automatický**
 - Řízení jednotky přebírá nadřazený systém
 - Skladová jednotka plní svoji roli v rámci testbedu
- **Manuální**
 - Jednotka je řízena dotykovým panelem
 - Možnost samostatné funkce jednotky bez nutnosti připojení ke zbytku testbedu

NAVIGACE SYSTÉMEM

- Orientace mezi obrazovkami je zajištěna fyzickými tlačítky na panelu.



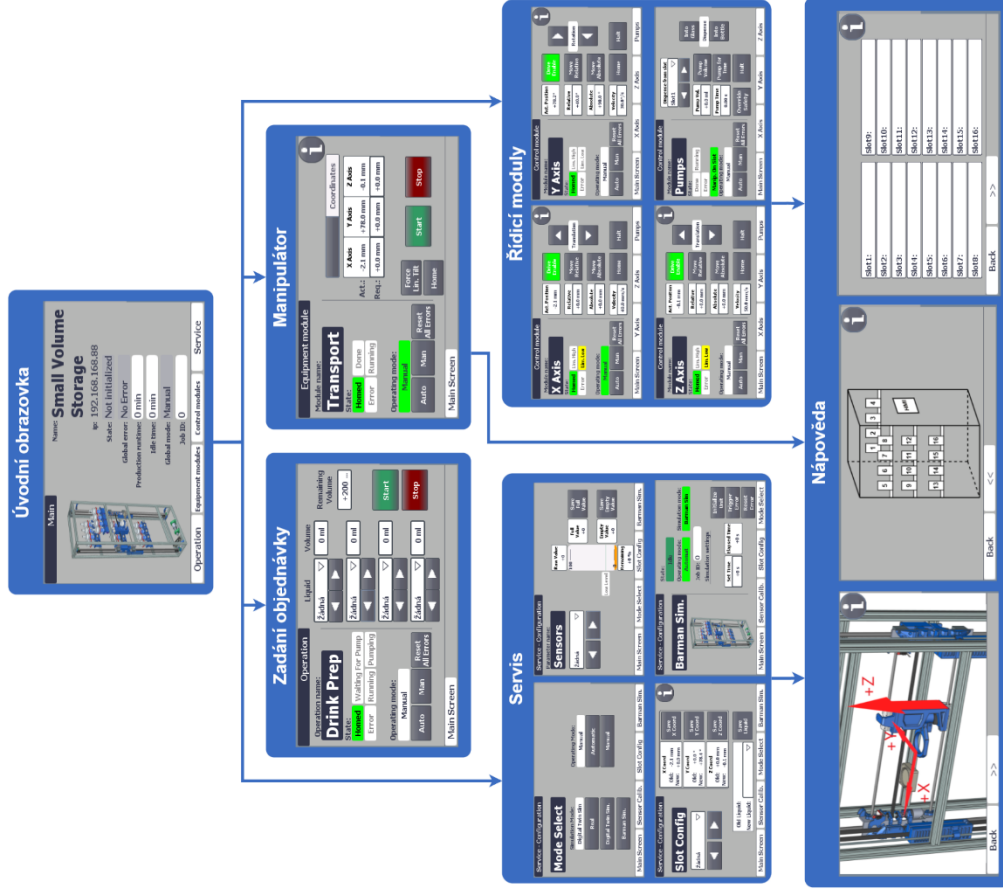
- Řídící obrazovky jsou rozděleny na dvě části:
 - V levé části je zobrazen aktuální stav skladové jednotky
 - Pravá část obrazovky slouží k zadávání příkazů a řízení jednotky
- Pole ohraničená tučnou čarou jsou určena k zadávání požadavků uživatelem.
- Pole, která nejsou ohraničená, slouží pro zobrazení informací.



POZOR

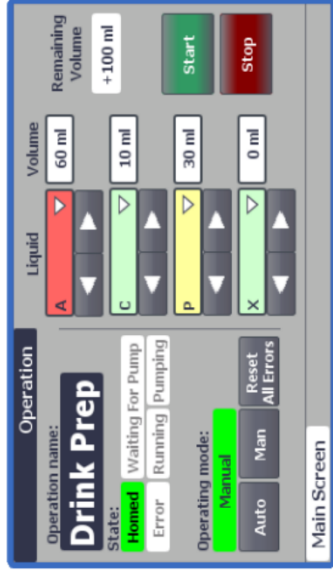
Při opuštění každé obrazovky (v manuálním režimu) se z bezpečnostních důvodů okamžitě přeruší všechny vykonávané akce.

Veškeré akce je možné přerušit také stisknutím tlačítka STOP nebo Halt. Po přerušení je možné stejnou akci opakovat bez nutnosti resetu systému.



NASTAVENÍ OBJEDNÁVKY

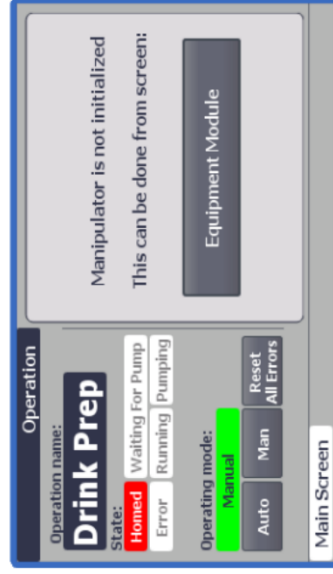
- Objedávka je vytvořena na obrazovce Operation, ke které je přístup z úvodní obrazovky.



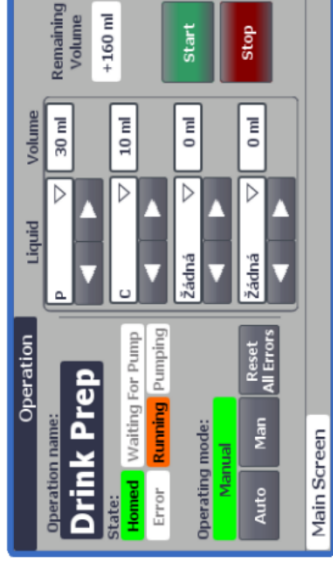
- Uživatel ze seznamu vybere požadovanou kombinaci tekutin společně s jejich objemem.
- Barva výběru tekutiny signalizuje její přítomnost ve skladu:
 - **Zelená** – Zvolená tekutina je přítomna ve skladu
 - **Oranžová** – Zvolené tekutiny je ve skladu méně než 20 %
 - **Červená** – Zvolená tekutina není uložena ve skladu a nemůže být načerpána

POZOR

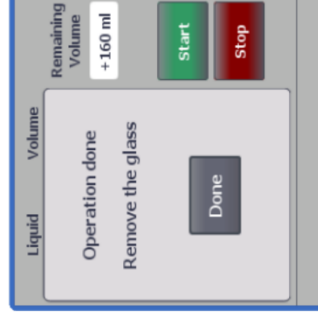
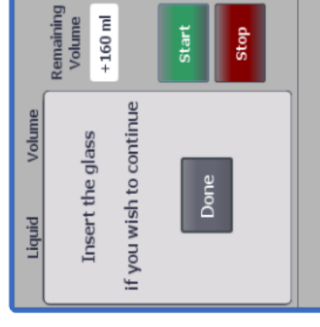
Objedávku je možné vykonat jen v případě, že je jednotka inicializována. Pokud tomu tak není, je uživatel odkázán na obrazovku, kde tak může učinit.



- Po spuštění operace jsou tekutiny přeskládány, aby vznikla optimální trasa manipulátoru a proces plnění skleničky byl tak co nejrychlejší.
- Tekutiny, které se ve skladu nenachází, jsou odstraněny z objednávky. Stejně tak ty, u kterých nebyl zadán objem.
- V tomto okamžiku už není možné objednávku upravovat.



- Manipulátor najíždí na pozici pro vyzvednutí skleničky, pokud tam ještě nestojí.
- Uživatel je poté vyzván, aby na platformu vložil prázdnou skleničku pro manipulátor. Po potvrzení se začne vykonávat operace plnění skleničky.



- Po dokončení operace je napplněná sklenička umístěna na stejnou platformu, ze které ji následně uživatel odebere.
- Po potvrzení odebrání skleničky je možné zadat další objednávku.

ŘÍZENÍ MANIPULÁTORU

- Manipulátor je možné řídit z obrazovky Equipment Module.
- Nejprve je ale nutné jej inicializovat.
- Proces inicializace se provádí po každém novém spuštění jednotky, protože manipulátor při odpojení ze sítě ztrácí informace o své poloze.
- Inicializace se aktivuje tlačítkem "Home".
- Uživatel je po stisku tlačítka dotázán, zda se aktuálně manipulátor nachází v bezpečné pozici.

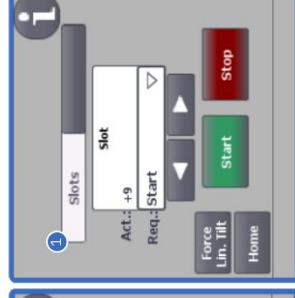
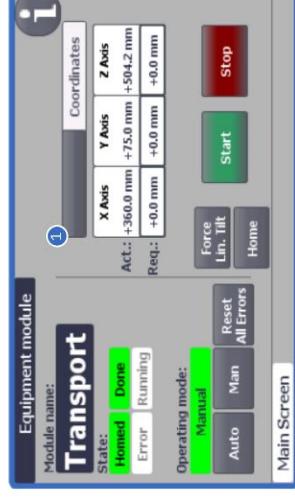


- 1 Po stisku tlačítka "Yes" započne proces inicializace manipulátoru
- 2 Po stisku tlačítka "No" je uživatel přesměrován na obrazovku "Control Modules", ze které může zajistit bezpečnou inicializaci manipulátoru
- 3 Stisk tlačítka "Back" zruší požadavek na inicializaci

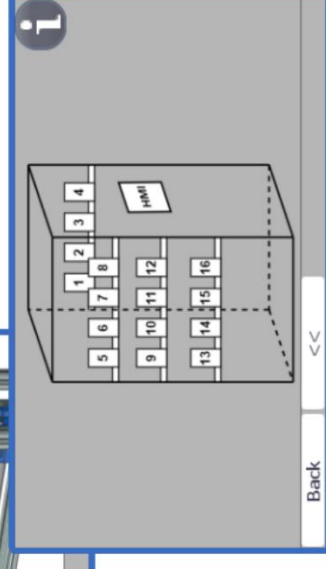
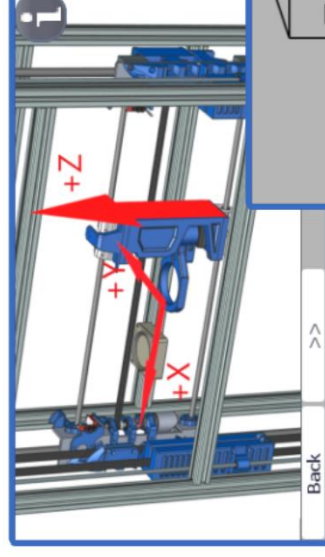
POZOR

- Nebezpečnou polohou je obecně myšlena poloha, kdy se rameno manipulátoru nenachází ve středové poloze.
- Pokud by manipulátor při aktivaci inicializace nebyl v bezpečné pozici, mohlo by dojít ke kolizi a poškození manipulátoru.

- Manipulátor zajišťuje transport na zadanou pozici. Tu je možné zadat buď jako souřadnice X, Y, Z, nebo jako číslo skladového slotu. Přepínání mezi způsoby zadání požadované polohy je možné tlačítkem 1.



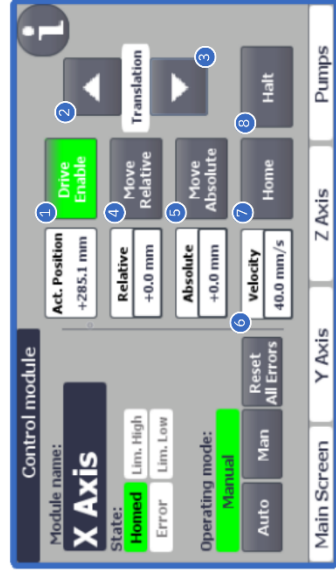
- V pravém horním rohu se nachází tlačítko 1. Po jeho stisku se aktivují obrazovky nápověd. První je nápověda souřadnicového systému. Druhá ukazuje rozložení slotů.



ŘÍZENÍ MODULŮ

- Jednotlivé moduly skladové jednotky je možné manuálně ovládat z obrazovky Control Modules.
- Tyto moduly jsou motory pro horizontální a vertikální pohyb manipulátoru (X Axis a Z Axis), rotační pohyb ramene manipulátoru (Y Axis) a čerpadla (Pumps).

ŘÍZENÍ MOTORŮ



- Pro ovládní motoru je nutné jej nejprve aktivovat.
- Všechny tři motory se ovládají totožným způsobem.

- 1 Drive Enable – Aktivuje motor
- 2 ↑ - Během držení tlačítka se manipulátor pohybuje v dané ose dopředu
- 3 ↓ - Během držení tlačítka se manipulátor pohybuje v dané ose dozadu
- 4 Move Relative – Manipulátor se posune v dané ose o zadanou vzdálenost
- 5 Move Absolute – Manipulátor se posune v dané ose na zadanou pozici
- 6 Velocity – Rychlost pohybu manipulátoru
- 7 Home – Inicializace manipulátoru v dané ose
- 8 Halt – Přeruší veškerý pohyb manipulátoru v dané ose

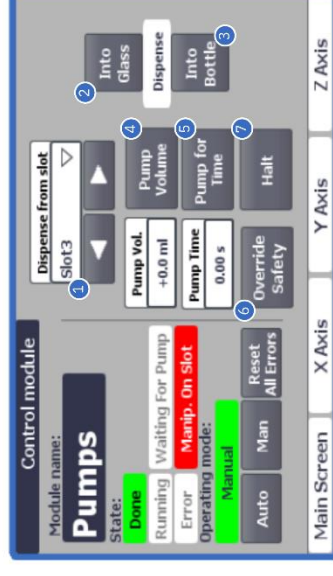
POZOR

Při neopatrném zacházení by mohlo dojít k poškození jednotky.

Bez inicializace motoru není možné používat příkaz "Move Absolute".

ŘÍZENÍ ČERPADEL

- Čerpadla jsou řízena podobně jako motory.



- 1 Volba skladového slotu, ze kterého se bude čerpat
- 2 Into Glass – Během držení tlačítka dochází k čerpání z láhve do skleničky
- 3 Into Bottle – Během držení tlačítka dochází ke zpětnému chodu – čerpání ze skleničky do láhve
- 4 Pump Volume – Dojde k načerpání požadovaného množství tekutiny
- 5 Pump for Time – Dojde k čerpání po zadanou dobu
- 6 Safety Override - Aktivuje režim popsaný na následující straně
- 7 Halt – Přeruší čerpání

POZOR

Po aktivaci určité akce je nutné vyčkat na dokončení komunikace s modulem pro čerpadla. Během tohoto čekání bliká kontrolka „Waiting for Pump“.

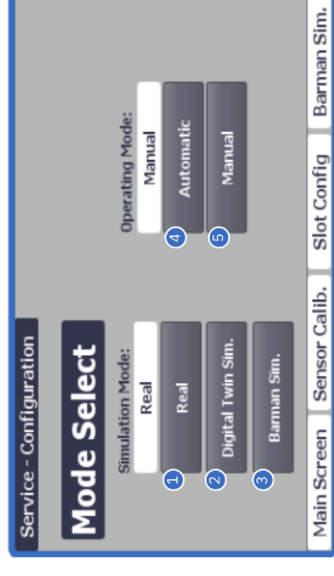
SERVIS A KONFIGURACE

PROVOZNÍ REŽIMY

- Skladová jednotka obsahuje kontrolu před aktivací nevhodného čerpadla.
- Pokud se pod zvoleným čerpadlem nenachází manipulátor se skleničkou, do které by se mohla napouštět tekutina, nedojde k aktivaci čerpadla.
- Tuto ochranu je možné deaktivovat stiskem tlačítka 6 "Override Safety".
- Poté bude možné čerpat z jakéhokoliv slotu.

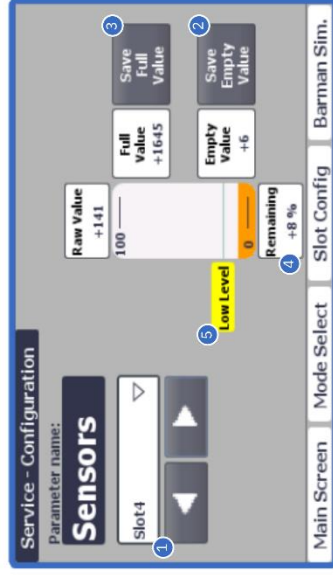


- Skladová jednotka se může nacházet v několika režimech. Mezi těmito režimy je možné přepínat na obrazovce Service → Mode Select.



- Režimy se dělí na dvě kategorie.
- První kategorií jsou režimy simulace.
 - 1 V režimu Real je řízena fyzická jednotka.
 - 2 V režimu Digital Twin Sim. je reálná jednotka deaktivována a je řízeno její digitální dvojče.
 - 3 Posledním simulačním režimem je Barman Sim. Tento režim je určen pro simulaci komunikace v rámci nadřazeného systému Testbed Self-Acting Barman.
- Další kategorií jsou provozní režimy. Zde je možné volit mezi manuálním a automatickým režimem.
 - 4 V manuálním režimu je skladová jednotka řízena připojeným dotykovým operátorským panelem.
 - 5 V automatickém režimu řízení jednotky přebírá nadřazený systém a operátorský panel je určen pouze pro monitoring.

KALIBRACE SENZORŮ HLADINY

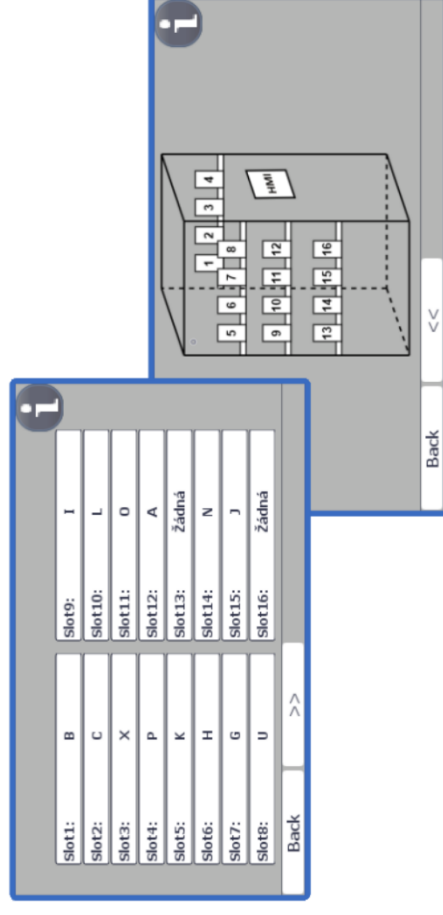


- Aktuální množství tekutiny v jednotlivých skladových pozicích je hlídáno kapacitními snímači hladiny. Tyto snímače je před použitím potřeba kalibrovat.
- Kalibrace probíhá ve třech krocích.
 - 1 Obsluha zvolí, jaký senzor bude kalibrovat
 - 2 Uloží se naměřená hodnota, když je láhev prázdná
 - 3 Uloží se naměřená hodnota, když je láhev plná
- Systém následně přepočítává naměřené hodnoty na procenta 4.
- Při poklesu hladiny pod 20 % dojde k vyvolání varování o nízké hladině 5.

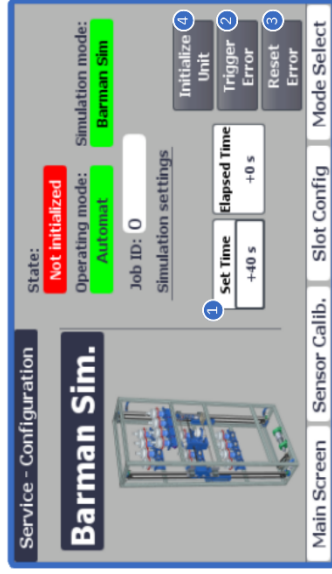
KONFIGURACE SKLADOVÝCH POZIC



- Skladová jednotka umožňuje dynamickou úpravu skladu.
- Je možné upravovat jak umístění skladových slotů 1, tak aktuální tekutiny, které obsahují 2.
- V pravém horním rohu se nachází tlačítko i. Po jeho stisku se aktivují obrazovky napovéd. První ukazuje aktuální stav skladu. Druhá ukazuje rozložení slotů.



- Skladová jednotka je jednou z mnoha jednotek testbedu Self-Acting Barman. Pokud je nutné ladit komunikaci s nadřazeným systémem a v rámci celého testbedu, využije obsluha obrazovku Service → Barman Sim.
- Obsluha volí čas simulace ①. Je to čas, po jehož uplynutí bude nastaven příznak "Done", se kterým pak pracuje nadřazený systém.
- Dále je možné vyvolat ② a resetovat ③ chybu jednotky, případně simulovat inicializaci jednotky ④.



POZOR

Je nutné, aby byl aktivován automatický provozní režim a simulační režim Barman Sim. Přepínání režimů zajišťuje obrazovka Service → Mode Select.

INICIALIZACE MANIPULÁTORU

- 1) Obrazovka Equipment Modules
- 2) Stisknutí tlačítka "Home"
 - a) Pokud je manipulátor v bezpečné poloze, stisknutí tlačítka "Yes - Continue". Započne proces automatické inicializace všech os. Po jeho dokončení se rozsvítí kontrolka stavu "Homed" zeleně a je možné opustit obrazovku.
 - b) Pokud manipulátor není v bezpečné poloze, stisknutí tlačítka "No - Control Module"

1. Dojde k přepnutí na obrazovku Control Modules.
2. Zde manuálně pomocí tlačítek ↑ ↓ →← nastaví obsluha manipulátor do bezpečné polohy.

3. Poté je možné jednotlivé osy postupně inicializovat samostatně, nebo opětovně zvolení obrazovky Equipment Module. Nyní už se ale manipulátor nachází v bezpečné poloze a můžeme jej inicializovat automaticky.

Nápovědu souřadnicového systému je možné zobrazit kliknutím na tlačítko ⓘ v pravém horním rohu.

POZOR

Nebezpečnou polohou je obecně myšlena poloha, kdy se rameno manipulátoru nenachází ve středové poloze.

Při pouštění obrazovky v průběhu inicializace dojde k jejímu přerušení.

KONFIGURACE POZICE SKLADOVÉHO SLOTU

1. Najedeme manipulátorem na požadovanou pozici jedním ze způsobů:
 - 1.a) Obrazovka Equipment Module -> Zadání X, Y, Z souřadnic -> Aktivace pohybu
 - 1.b) Obrazovka Control Module -> manuální najetí manipulátorem na požadovanou pozici v jednotlivých osách
2. Přepneme se na obrazovku Main Screen -> Service -> Slot Config
3. Zvolíme číslo slotu, který chceme přepsat
4. Vidíme aktuální pozici slotu a aktuální pozici manipulátoru, tzn. novou pozici slotu
5. Uložíme jednotlivé souřadnice tlačítky "Save"

ZMĚNA TYPU TEKUTINY VE SKLADOVÉM SLOTU

1. Provedeme kalibraci senzorů
 - 1.1. Main Screen -> Service -> Sensor Calib
 - 1.2. Výběr slotu
 - 1.3. Vložení prázdné láhve, uložení aktuální hodnoty = "Save Empty Value"
 - 1.4. Naplnění láhve, uložení aktuální hodnoty = "Save Full Value"
2. Přepíšeme seznam obsahu skladu
 - 2.1. Main Screen -> Service -> Slot Config, nebo přímo Slot Config
 - 2.2. Výběr slotu
 - 2.3. Výběr nové tekutiny ze seznamu a uložení

POZOR

Předpokládáme, že je manipulátor inicializován (týká se konfigurace pozice).

ČIŠTĚNÍ ČERPADEL

- Čerpadla je vhodné pravidelně čistit. Obzvlášť je to nutné při změně tekutiny ve skladovém slotu.
- Čištění je možné provést několika způsoby. Prvním krokem každého ze způsobů je vymytí láhve a její naplnění vodou, případně vhodným čisticím prostředkem.

A) Vytvoření uzavřeného okruhu

1. Na trubičku, vedoucí z čerpadla do skleničky, nasadíme gumový nástavec.
2. Druhý konec nástavce umístíme do láhve. Vytvoříme tak uzavřený okruh.
3. Na obrazovce Control Modules -> Pumps aktivujeme Override Safety a spustíme čerpadlo na daném slotu. To je možné udělat libovolným z dostupných způsobů.

B) Využitím manipulátoru

Provedeme klasické čerpání s využitím manipulátoru z požadovaného slotu. Můžeme využít například obrazovku pro vytvoření objednávky.

C) Spolupráce s kolegou a využití funkce "Into Glass"

Pokud máte k dispozici kolegu, můžete využít na obrazovce Control Modules -> Pumps funkce IntoGlass. Jeden člověk tak drží nádobu pod daným čerpadlem a druhý na dotykovém panelu drží tlačítko požadované funkce. Tento způsob je nejrychlejší, je však potřeba dvou osob a dobrý přístup do jednotky.

Příloha C - Přílohy na SD kartě

CAD	...	Virtuální dvojče skladové jednotky
Firmware\S7-1200	...	Řídicí logika pro PLC S7-1200 v TIA Portal V17
Firmware\S7-1500	...	Řídicí logika pro PLC S7-1500 v TIA Portal V17
xholba05_BP.pdf	...	Elektronická verze bakalářské práce
xholba05_manual.pdf	...	Elektronická verze provozního manuálu
xholba05_schematic.pdf	...	Elektronická verze schématu elektrického zapojení