



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV VÝROBNÍCH STROJŮ, SYSTÉMŮ A ROBOTIKY

INSTITUTE OF PRODUCTION MACHINES, SYSTEMS AND ROBOTICS

# VIRTUÁLNÍ ZPROVOZNĚNÍ ROBOTIZOVANÉHO PRACOVNÍŠTĚ PRO STUDENÝ NÁSTRÍK

VIRTUAL COMMISSIONING OF ROBOTIZED WORKPLACE FOR COLD SPRAYING

## DIPLOMOVÁ PRÁCE

MASTER'S THESIS

## AUTOR PRÁCE

AUTHOR

**Bc. Peter Miklis**

## VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Jan Vetiška, Ph.D.**

**BRNO 2021**



## Zadání diplomové práce

Ústav:	Ústav výrobních strojů, systémů a robotiky
Student:	<b>Bc. Peter Miklis</b>
Studijní program:	Strojní inženýrství
Studijní obor:	Výrobní stroje, systémy a roboty
Vedoucí práce:	<b>Ing. Jan Vetiška, Ph.D.</b>
Akademický rok:	2020/21

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

### **Virtuální zprovoznění robotizovaného pracoviště pro studený nástřik**

#### **Stručná charakteristika problematiky úkolu:**

Současná doba klade stále větší důraz na rychlé zprovoznění nových výrobních systémů a jejich flexibilitu. Jednou z cest jak tohoto cíle dosáhnout je využití tzv. virtuálního zprovoznění nově navrhovaného výrobního systému. Cílem práce bude virtuální zprovoznění robotizovaného pracoviště pro studený nástřik. Práce je založena na syntéze znalostí z mechaniky, elektřiny, automatizace, robotiky a programování.

#### **Cíle diplomové práce:**

Rešerše dané problematiky.  
Návrh uživatelského sw.  
Návrh HMI.  
Digitální zprovoznění.  
Testování na reálném zařízení.

#### **Seznam doporučené literatury:**

SICILIANO, Bruno a Oussama. KHATIB. Springer handbook of robotics. Berlin: Springer, 2008. ISBN 978-3-540-23957-4.

KOLÍBAL, Zdeněk. Roboty a robotizované výrobní technologie: 1. Brno: Vysoké učení technické v Brně - nakladatelství VUTIUM, 2016. ISBN 978-802-1448-285.

NOF, Shimon Y. Springer handbook of automation. 1. New York: Springer, 2009. ISBN 978-354-07-8-300.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2020/21

V Brně, dne

L. S.

---

doc. Ing. Petr Blecha, Ph.D.  
ředitel ústavu

---

doc. Ing. Jaroslav Katolický, Ph.D.  
děkan fakulty

## **ABSTRAKT**

Diplomová práca sa zaoberá virtuálnym sprevádzkovaním robotizovaného pracoviska pre studený nástrek. Teoretická časť práce obsahuje prehľad súčasného stavu poznania v oblasti virtuálneho uvedenia do prevádzky, teórie tvorby užívateľských prostredí HMI, zaoberá sa princípmi programovacieho jazyka Rapid ako nástrojom pre ovládanie priemyselných robotov a analýzou komponentov robotizovaného pracoviska. V praktickej časti je definovaná štruktúra navrhnutého užívateľského programu a užívateľského prostredia, ktoré sú vytvorené pomocou softvérových nástrojov Robotstudio a Screenmaker. Posledná časť podrobne rozoberá kroky počas implementovania vytvoreného programu spolu s HMI do prevádzky spojeného s testovaním funkcionalít na reálnom zariadení pre studený nástrek. Súčasťou diplomovej práce je tiež užívateľský manuál vytvorený pre obsluhu priemyselného robota.

## **ABSTRACT**

The diploma thesis deals with the virtual commissioning of robotized workplace for cold spraying. The theoretical part of the thesis contains an overview of the current state of knowledge in the field of virtual commissioning, the theory of crating HMI user environments, deals principles of the Rapid programming language as a tool for controlling industrial robots and analysis of robotic workplace components. The practical part defines the structure of the proposed user program and user environment, which are created using software tools Robotstudio and Screenmaker. The last part discusses in detail the steps during the implementation of the created program together with the HMI into commission, which is associated with the testing of functionalities on a real device for cold spraying. The diploma thesis also includes a user manual created for operating an industrial robot.

## **KLÚČOVÉ SLOVÁ**

Virtuálne sprevádzkovanie, ABB, RAPID, Studený nástrek, PLC, HMI, RobotStudio, Screenmaker

## **KEYWORDS**

Virtual commissioning, ABB, RAPID, Cold spray, PLC, HMI, RobotStudio, Screenmaker



## **BIBLIOGRAFICKÁ CITÁCIA**

MIKLIS, Peter. *Virtuální zprovoznění robotizovaného pracoviště pro studený nástřik* [online]. Brno, 2021 [cit. 2021-05-21]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/131696>. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav výrobních strojů, systémů a robotiky. Vedoucí práce Jan Vetiška.





## **POĎAKOVANIE**

V prvom rade by som sa chcel poďakovať vedúcemu mojej diplomovej práce Ing. Janovi Vetiškovi, PhD. za odbornú pomoc a potrebné rady pri vypracovávaní diplomovej práce. Hlavné poďakovanie však patrí mojej najbližšej rodine a priateľom za ich nekonečnú podporu a pomoc pri akademických a životných udalostiach.



## **ČESTNÉ PREHLÁSENIE**

Prehlasujem, že táto práca je mojím pôvodným dielom, spracoval som ju samostatne pod vedením Ing. Jana Vetišky, Ph.D. a s použitím literatúry uvedenej v zozname.

V Brne dňa 21.5.2021

.....

Peter Miklis



# OBSAH

<b>1</b>	<b>ÚVOD</b> .....	<b>15</b>
<b>2</b>	<b>PREHLAD SÚČASNÉHO STAVU POZNANIA</b> .....	<b>17</b>
2.1	Virtuálne uvedenie do prevádzky .....	17
2.2	Technológia studeného nástreku .....	20
2.3	Programovací jazyk RAPID .....	22
2.4	Zásady programovania.....	23
2.5	HMI – Human machine interface .....	23
<b>3</b>	<b>ANALÝZA ROBOTICKÉHO PRACOVISKA</b> .....	<b>27</b>
3.1	Priemyselný robot IRB 4600 .....	27
3.2	Polohovadlo IRBP A-250 .....	29
3.3	Priemyselný ovládač robotov IRC 5 .....	30
3.4	Cold Spray System 5/11 .....	31
<b>4</b>	<b>VIRTUÁLNE SPREVÁDZKOVANIE</b> .....	<b>35</b>
4.1	Rozbor 3D virtuálneho pracoviska .....	35
4.2	Vstupné požiadavky na program .....	36
4.3	Definovanie súradnicových systémov .....	36
4.4	Programová štruktúra.....	37
4.5	Základné pohybové procedúry.....	38
4.5.1	Podprogramy horizontálneho a vertikálneho pohybu.....	40
4.5.2	Podprogramy diagonálneho pohybu .....	41
4.6	Použitie I/O signály.....	44
4.7	Externý modul všeobecného programu .....	45
4.8	Hlavný modul MAIN.....	46
4.9	Návrh užívateľského prostredia .....	51
<b>5</b>	<b>REÁLNE SPREVÁDZKOVANIE</b> .....	<b>57</b>
<b>6</b>	<b>ZHODNOTENIE A DISKUSIA</b> .....	<b>61</b>
<b>7</b>	<b>ZÁVER</b> .....	<b>63</b>
<b>8</b>	<b>ZOZNAM POUŽITÝCH ZDROJOV</b> .....	<b>65</b>
<b>9</b>	<b>ZOZNAM SKRATIEK, SYMBOLOV, OBRÁZKOV A TABULIEK</b> .....	<b>69</b>
9.1	Zoznam skratiek a symbolov .....	69
9.2	Zoznam tabuliek .....	69
9.3	Zoznam obrázkov .....	69
<b>10</b>	<b>ZOZNAM PRÍLOH</b> .....	<b>71</b>



# 1 ÚVOD

Človek a jeho prirodzená potreba napredovať zanecháva za sebou bohatú históriu technologického pokroku. Postupnými krokmi sa ľudstvo učilo využívať prírodu a okolité zdroje pre dosiahnutie svojich cieľov a uľahčenie si každodenného života. V súčasnosti je technologický pokrok ľudstva aj vďaka globalizácii nezastaviteľný, pričom jedným zo základných pilierov zdokonaľujúcich technologické a výrobné procesy je automatizácia. Priemyselná automatizácia predstavuje syntézu mechaniky, elektrotechniky, informatiky a strojárstva, ktorej výsledný produkt je schopnosť automatického riadenia strojných systémov a robotizovaných pracovísk. Využitie priemyselných robotov, automatizačných systémov a manipulatorov má za dôsledok zvýšenie efektivity práce a zvýšenie finančného zisku firiem, nakoľko tieto komponenty dokážu vykonávať isté druhy prác rýchlejšie, presnejšie a lacnejšie, než ľudská obsluha. Neoddeliteľnou súčasťou automatizácie je však samotný proces návrhu a prípravy automatizovaných pracovísk a liniek, ktorá predstavuje nemalú výzvu spojenú so sériou konštrukčných riešení a rozsiahleho nastavovania. Z dôvodu zjednodušenia a zefektívnenia prípravných procesov automatizovaných pracovísk sa v posledných rokoch vyvinula technológia virtuálneho uvedenia do prevádzky. Pomocou tejto technológie je možné výrazne zredukovať prípravný čas uvedenia robotizovaných pracovísk do prevádzky a odhaliť chyby výrobných procesov, ešte predtým, ako mohli vzniknúť na reálnom pracovisku.

Diplomová práca sa venuje riešeniu problematiky virtuálneho uvedenia do prevádzky pracoviska pre studený nástrek. Pracovisko pozostáva z viacerých konkrétnych komponentov, ktoré sú podrobne opísané v kapitole analýzy pracoviska. Hlavnou úlohou diplomového projektu je vytvorenie softvéru a užívateľského rozhrania umožňujúceho plne automatizované ovládanie robotickej bunky. V prvej časti sú vysvetlené pojmy ako virtuálne uvedenie do prevádzky, studený nástrek a priblíženie základnej teórie tvorby zdrojového kódu a užívateľských obrazoviek. Nosná časť diplomovej práce sa zaoberá tvorbou a štruktúrou vytvoreného programu a vysvetľuje postupy a riešenia, ktoré boli vytvorené tak, aby program nielen spĺňal všetky požiadavky definované zákazníkom ale ich aj aktívne rozšíril. V záverečnej časti je zdokumentovaný priebeh aplikovania virtuálneho uvedenia do prevádzky konkrétneho pracoviska pre studený nástrek, ktorý bol spojený s testovaním vytvoreného programu na skutočnom zariadení.



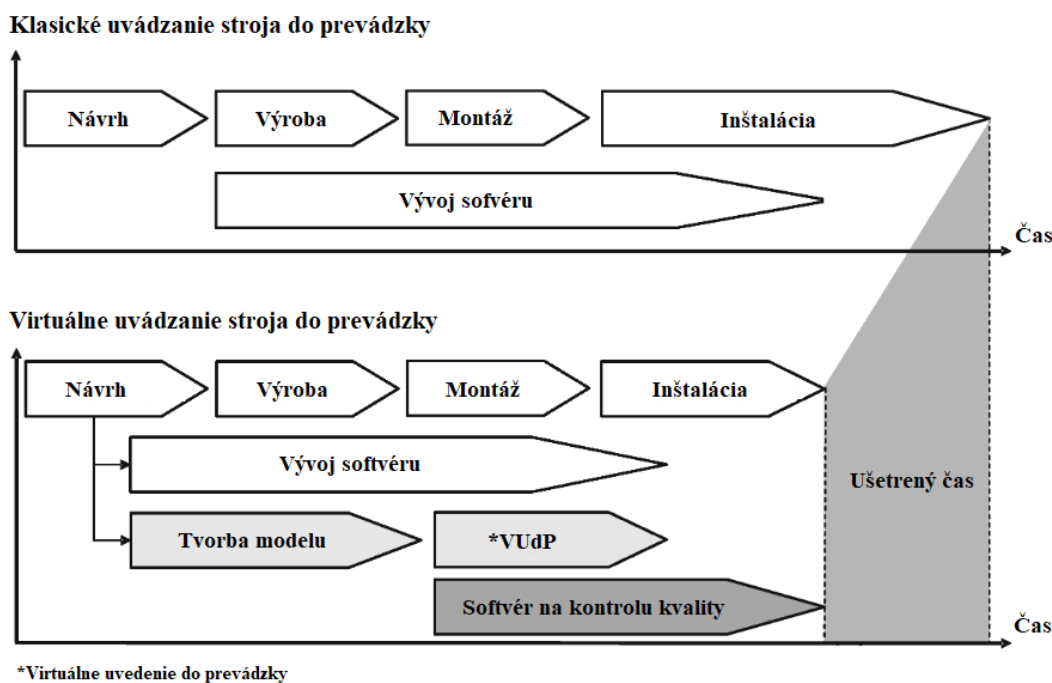


## 2 PREHĽAD SÚČASNÉHO STAVU POZNANIA

### 2.1 Virtuálne uvedenie do prevádzky

Pod pojmom virtuálne sprevádzkovanie sa rozumie proces aplikovania moderných technológií na vytvorenie a testovanie „virtuálnych“ simulácií ovládacích systémov predtým, než sú pripojené do skutočných obrábacích a manipulačných systémov.

Virtuálne uvedenie do prevádzky bolo prvýkrát predstavené v roku 1999 profesorom Franzom Auingerom a kolektív vo vedeckom článku *Interface driven domain-independent modeling architecture for “soft-commissioning” and “reality in the loop”*, avšak v tom čase trh neposkytoval potrebné hardvérové vybavenie za prijateľnú cenu. Najmä ak mali byť splnené požiadavky na vytvorenie simulácií v reálnom čase pri použití hardvérových programovateľných logických radičov (PLC). Postupným vývinom technológií prof. Záh, Michael F. a kolektív v roku 2006 predstavili štúdiu, ktorá dokazuje, že virtuálne uvedenie do prevádzky môže ušetriť až 50-75% času potrebného na reálne uvedenie projektu do prevádzky. [1] [2]



Obr. 1) Časové porovnanie klasického a virtuálneho uvádzania do prevádzky [3]

Súčasný stav technologických vymožeností ponúka automatizačnému priemyslu možnosti, ktorých využiteľnosť je prístupnejšia než kedykoľvek predtým. V rámci rastúcej potreby urýchliť vývoj a minimalizovať riziko pri vývoji nových produktov sa virtuálne uvedenie do prevádzky a digitálne dvojčatá stali bazálnymi technológiami. [4]

Rozdiel medzi reálnym - fyzickým uvedením do prevádzky a virtuálnym uvedením do prevádzky spočíva v tom, že reálne uvedenie do prevádzky zahŕňa testovanie priamo na skutočnom výrobnom systéme, so skutočnými strojmi, reálnym radičom PLC a ďalšími

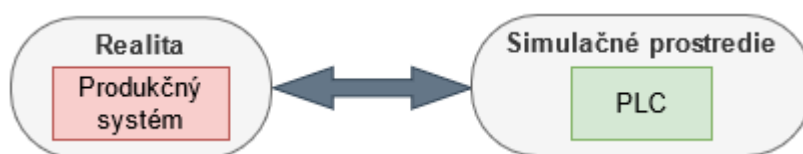
periférnymi zariadeniami a senzormi. Na druhej strane, virtuálne uvedenie do prevádzky využíva virtuálne vytvorené ekvivalenty výrobného systému, pričom kompletná konfigurácia systému prebieha v rámci virtuálneho prostredia. [4]

Virtuálne prostredie sa správa úplne rovnako ako reálny hardvér, vrátane využívania všetkých vstupno-výstupných signálov a údajov. Vďaka tomu poskytuje možnosti testovania naprogramovaných výrobných procesov, pričom je možné odstrániť chyby v počiatočných fázach projektu, čiže ešte predtým, než by sa prejavili vo výrobe. Zároveň virtuálne prostredie poskytuje možnosti zdokonaľovania a ladenia procesov presne podľa požiadaviek užívateľa tak, aby sa dosiahla maximálna využiteľnosť a efektivita celého systému. V neposlednom rade veľkou výhodou virtuálneho uvádzania do prevádzky je jeho využitie na tréningové účely, kedy neskúsení programátori majú možnosť získať praktické skúsenosti s nastavovaním robotických pracovísk, bez rizika poškodenia výrobného linky. [5] [6]

Virtuálne uvedenie do prevádzky pozostáva z troch hlavných častí, pričom tento koncept ponúka efektívne riešenie odstraňovania chýb vzniknutých pri uvádzaní strojov do prevádzky, nakoľko programátor/inžinier sa môže zamerať na konkrétny okruh problémov:

- Model in the Loop (MiL)
- Software in the Loop (SiL)
- Hardware in the Loop (HiL)

Prvou fázou je Model in the Loop (MiL), v ktorej sa vytvorí kontrola ovládania a správania logického modelu, používaného v regulátore PLC a kontrola užívateľských prostredí HMI. Tento logický model sa pripája k simulácií robotického bunky, v rámci ktorej sa overuje jeho správna funkčnosť. Fáza MiL poskytuje možnosti testovania a overovania logického modelu virtuálneho PLC a HMI, kde sa ich správanie porovnáva so správaním skutočnej robotického bunky. [5] [7]



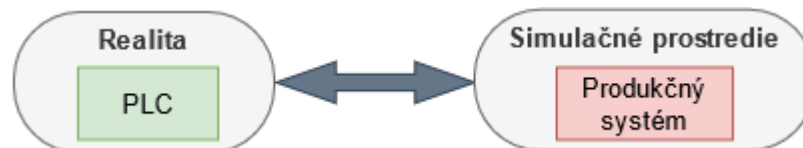
Obr. 2) Princíp fázy Model in the Loop [7]

Ďalšou fázou je Software in the Loop (SiL), v ktorej sa testuje správnosť skompilovanej logiky regulátora PLC do softwaru. Znamená to, že v tejto časti je možné sa zamerať na kontrolu kompilovaného kódu programu, ktorý má byť vykonávaný výrobnou linkou, vo virtuálnom emulátore PLC. Kontrola prebieha v rámci simulovaného virtuálneho prostredia a teda celý proces je možné uskutočniť na jednom PC zariadení. [5] [7]



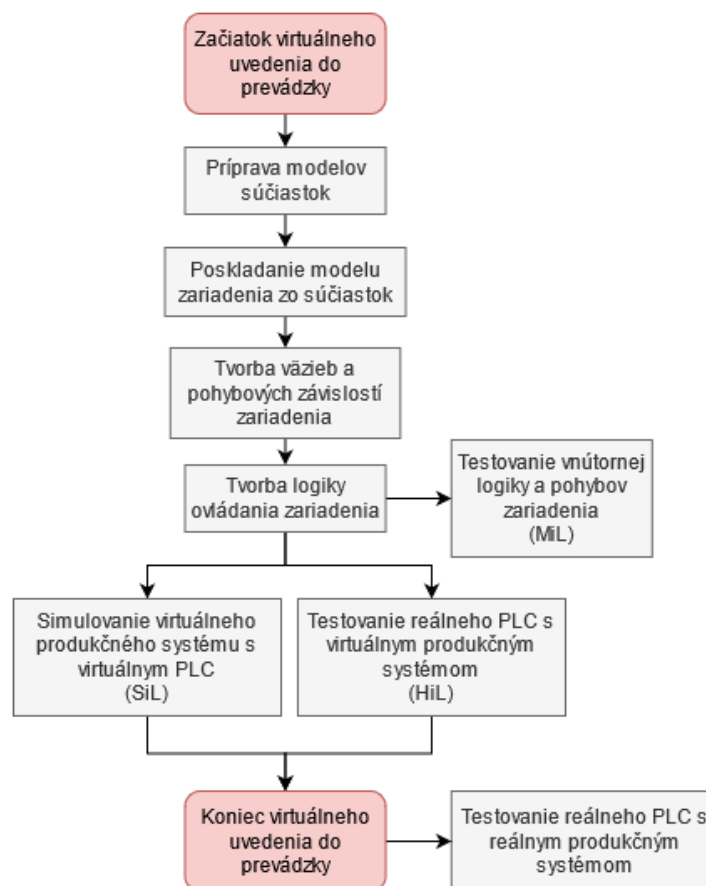
Obr. 3) Princíp fázy Software in the Loop [7]

Poslednou treťou fázou je Hardware in the Loop (HiL). Po kontrole konfigurácie logického modelu a po jeho úspešnom skompilovaní prichádza na rad záverečné overenie skompilovaného programu priamo na skutočnom PLC. Toto overovanie prebieha na skutočnom hardvéri, avšak samotné simulovanie naprogramovaných procesov je možné pripojiť k virtuálnemu simulačnému prostrediu výrobnjej bunky. Tento spôsob zapojenia (reálne PLC – virtuálne pracovisko) poskytuje ďalšie možnosti odhalenia chýb vzniknutých počas prípravy logických modelov a výrobných programov pred finálnym zapojením naprogramovaného reálneho PLC k výrobnjej bunke. [5] [7]



Obr. 4) Princíp fázy Hardware in the Loop [7]

Nasledujúci obrázok opisuje postup a jednotlivé časti, ktoré sa vykonávajú pri virtuálnom uvedení robotizovaného pracoviska do prevádzky.

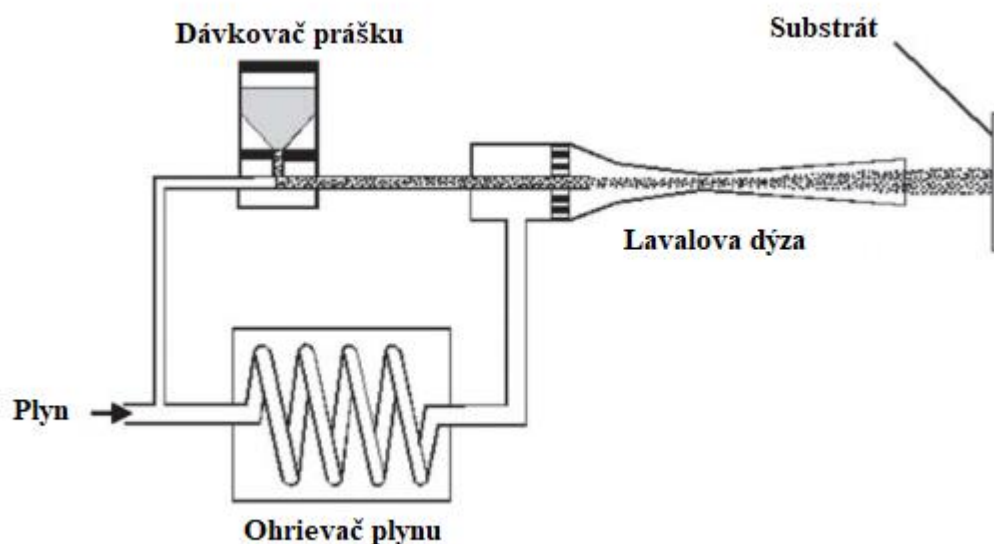


Obr. 5) Postup virtuálneho uvedenia do prevádzky [8]

## 2.2 Technológia studeného nástreku

Prvotná technológia studeného nástreku, z anglického prekladu „Cold Spray“, bola prvýkrát objavená na začiatku 80-tych rokov 20. storočia na Inštitúte Teoretickej a Aplikovanej Mechaniky Ruskej Akadémie Vied v meste Novosibirsk počas realizácie štúdií zameraných na pozorovanie nadzvukového prúdenia pevných častíc a plynu v aerodynamickom tuneli. Počas experimentov si vedci všimli, že pevné častice hliníka poháňané héliom rýchlosťou 400 - 450 m/s vytvárajú na pozorovanom objekte rovnomernú povrchovú štruktúru a to už pri teplote 300 K. Na základe týchto a ďalších zistení si v priebehu niekoľkých rokov vedúci tohto projektu A. P. Alchimov, Anatoli Papyrin, V.F. Kossarev a ich tím nechali patentovať metódu nanášania povlaku na povrch nazývanú „Cold Spray“, rovnako ako aj nástroj na to vyvinutý. [9] [10]

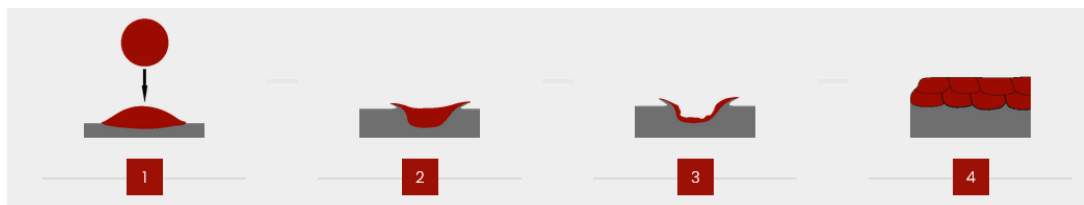
V priebehu vývoja technológie studeného nástreku vzniklo viacero riešení a patentov, ktorých cieľom bolo vytvoriť efektívnu metódu nanášania urýchlených pevných častíc na povrch obrobkov. Všeobecný princíp technológie Cold Spray aj v súčasnosti spočíva v predhrievaní vysokotlakového plynu, ktorý je vedený cez konvergentnú (zbiehavú) časť Lavalovej dýzy do divergentnej (rozbiehavej) časti, kde prichádza k jeho expanzii. Táto expanzia zapríčiňuje radikálne zníženie tlaku spojené s urýchľovaním plynových molekúl na nadzvukovú rýchlosť, čím sa aj samotná teplota plynu výrazne znižuje a to až na teplotu okolia. Častice nanášaného materiálu sú počas procesu privádzané axiálne na začiatok Lavalovej dýzy tak, aby boli urýchľované spolu s ohriatym vysokotlakovým plynom, s ktorým sú doslova vystreľované z dýzy von a to rýchlosťou dosahujúcou až 1200 m/s. Takto vystrelené častice nanášaného materiálu dopadajú na povrch obrobku s veľkou kinetickou energiou, ktorá zapríčiňuje extrémne rýchlu plastickú deformáciu dopadajúcich mikročastíc a naruší povrchovú vrstvu obrobku. V nej sa pevne zachytávajú zdeformované mikročastice, ktoré postupne vytvárajú hrubé vrstvy usadeného materiálu. Základný princíp technológie Cold Spray je možné vidieť na obrázku 6. [9] [11] [12]



Obr. 6) Schéma princípu technológie studeného nástreku [11]

Existuje mnoho štúdií zaoberajúcich sa rozborom technológie studeného nástreku, s cieľom optimálne nastaviť „Cold Spray Gun“, aby sa dosiahla maximálna efektívnosť obrábania a najvyššia kvalita nanášaných vrstiev materiálu. Výsledná efektívnosť a kvalita povrchu je priamo závislá na mnohých faktoroch. Medzi najdôležitejšie patrí typ plynu používaný na urýchľovanie nanášaného prachu, veľkosť tlaku tohto plynu vstupujúceho do systému, teplota, na ktorú sa vstupný plyn ohreje pred vstupom do Lavalovej dýzy, typ nanášaného materiálu a veľkosť jeho mikročastíc, konštrukcia Lavalovej dýzy a ďalšie. Všetky tieto aspekty ovplyvňujú výslednú rýchlosť mikročastíc na ústí Lavalovej dýzy. Je to práve úst'ová rýchlosť, ktorá definuje mieru kvality nanášaných vrstiev. Pre dosiahnutie optimálne hustého a pevného vrstvenia pridávaného materiálu je nevyhnutné, aby dopadová rýchlosť mikročastíc ležala medzi hodnotami tzv. „kritickej rýchlosti“ a „erozívnej rýchlosti“. [11] [12]

Hodnota kritickej rýchlosti predstavuje hlavný faktor ovplyvňujúci vytvorenie úspešného spojenia nanášaných mikročastíc na obrobok, pričom táto hodnota je odlišná pre každý druh materiálu a ovplyvňuje ju aj samotná veľkosť mikročastice. Ako je možné vidieť na obrázku 7 v prípade, že nanášaná častica dopadá na obrobok kriticou rýchlosťou, plasticky sa deformuje a pevne sa spojí s povrchom obrobku. Podľa druhu použitého materiálu, ktorý sa má nanášať technológiou Cold Spray na substrát, je nutné dosiahnuť minimálnu dopadovú rýchlosť častíc rovnú kritickej rýchlosti. Pokiaľ by táto hodnota bola nižšia, častica prášku nevytvorí väzbu so substrátom a povrchová úprava obrobku nebude možná. Pokiaľ však dopadová rýchlosť bude príliš vysoká a dosiahne hodnotu erozívnej rýchlosti, mikročastice namiesto spojenia so substrátom začnú odoberať materiál z povrchu, čím dochádza k jeho znehodnoteniu. [12]



Obr. 7) Porovnanie dopadových rýchlostí materiálu na substrát [12]

Veľkou výhodou je možnosť použitia rozličných druhov materiálov vo forme prášku, počnúc rôznymi druhmi ocele, hliníkom, horčíkom, titánom, meďou, niklom, cínom, zinkom, chrómom, zirkóniom, molybdénom, tantalom, nióbovom a inými. Rovnako sa dajú využiť práškové varianty rozličných zliatin, ako napríklad bronz, nikel – chróm, zliatiny hliníku alebo zliatiny titánu a iných. Možnosťou je taktiež používanie vzácnych kovov ako striebro, zlato a platina. Veľkou výhodou tejto technológie je možnosť vytvorenia a používania materiálových zmesí, čím sa dajú dosiahnuť presné požadované povrchové vlastnosti obrobkov. Pre predstavu, takýmto zmiešaním materiálových práškov je možné pomocou studeného nástreku naniesť na povrch obrobku kompozitnú vrstvu, zloženú z dvoch alebo viacerých materiálov, čím sa vytvorí jednotný materiál s fyzikálnymi a chemickými vlastnosťami, ktoré nemá sama o sebe žiadna z jeho súčastí. [9] [12]

## 2.3 Programovací jazyk RAPID

Programovací jazyk RAPID je jazyk vytvorený spoločnosťou ABB pre obsluhu a ovládanie priemyselných robotov tejto firmy, ktorého prvá verzia bola prvýkrát predstavená už v roku 1994. Program je tvorený z rutín, v rámci ktorých naprogramované inštrukcie dokážu riadiť pohyb priemyselného robota, spínať signálové výstupy a mnoho ďalších činností. Používané sú 3 druhy rutín: [13] [14]

- Procedúry – predstavujú podprogramy využívané v rámci hlavného programu.
- Funkcie – slúžia ako časť programu vracajúca hodnoty špecifických typov, ktoré sú následne možné použiť ako vstupné argumenty v programových inštrukciách.
- Trap rutiny – predstavujú podprogramy reagujúce na vzniknuté prerušenie v rámci programu. Trap rutiny je možné prepojiť napríklad na signál prerušenia, pri ktorom zopnutí sa programový ukazovateľ presúva do trap rutiny.

Programovací jazyk RAPID, tak ako iné jazyky, pracuje s dátami, v ktorých sú uložené riadiace informácie. RAPID tieto dáta kategorizuje do celkov, nazývaných dáta typy. Dohromady sa v rámci jazyka používa 103 rôznych dáta typov, pomocou ktorých je možné pracovať s informáciami ako rozmery a konfigurácia nástroja, nastavenie pozícií, numerické a stringové hodnoty a mnohé ďalšie. Tieto dáta sa v rámci programu používajú v 3 triedach: [14]

- CONST – konštanta predstavuje pevne stanovenú hodnotu dát, ktorá môže byť zmenená jedine manuálnou zmenou v rámci zdrojového kódu.
- VAR – premenná predstavuje typ dát, ktorých hodnotu je možné meniť počas vykonávania programu
- PERS – perzistentná premenná predstavuje typ dát, ktorých hodnotu je možné meniť počas vykonávania programu a zároveň sa táto hodnota ukladá do pamäte programu.

Pamäť programu tvoreného jazykom RAPID je delená do programových modulov samotného programu a systémových modulov. Programové moduly môžu obsahovať definované dáta a používané procedúry, pričom jeden z modulov musí obsahovať hlavnú procedúru Main, v rámci ktorej sú v programe vykonávané inštrukcie a volané všetky ostatné procedúry. Systémové moduly sa v RAPIDE používajú pre definovanie špecifických dát a rutín, ktoré sú samostatne definované v systéme a sú rovnaké pre všetky využívané programy. [14]

## 2.4 Zásady programovania

Správne zásady písania zdrojového kódu programu predstavujú dôležitú súčasť tvorby softvéru. Zdrojový kód každého programovacieho jazyka by mal byť jasný a pochopiteľný nielen pre samotného autora programu ale aj pre bežného vývojára. Takýto kód sa v programátorskej praxi označuje názvom „Clean Code“, resp. čistý kód. Vo všeobecnosti platí, že zdrojový kód vytváraného programu by mal spĺňať 4 základné princípy: [15] [16]

1. Kód by mal byť čo najjednoduchší a jasne čitateľný.
2. Kód by mal byť udržiavateľný a stabilne štruktúrovaný.
3. Kód by mal poskytovať možnosť tvorby jednoduchých zmien.
4. Kód by malo byť možné ďalej jednoducho rozširovať.

## 2.5 HMI – Human machine interface

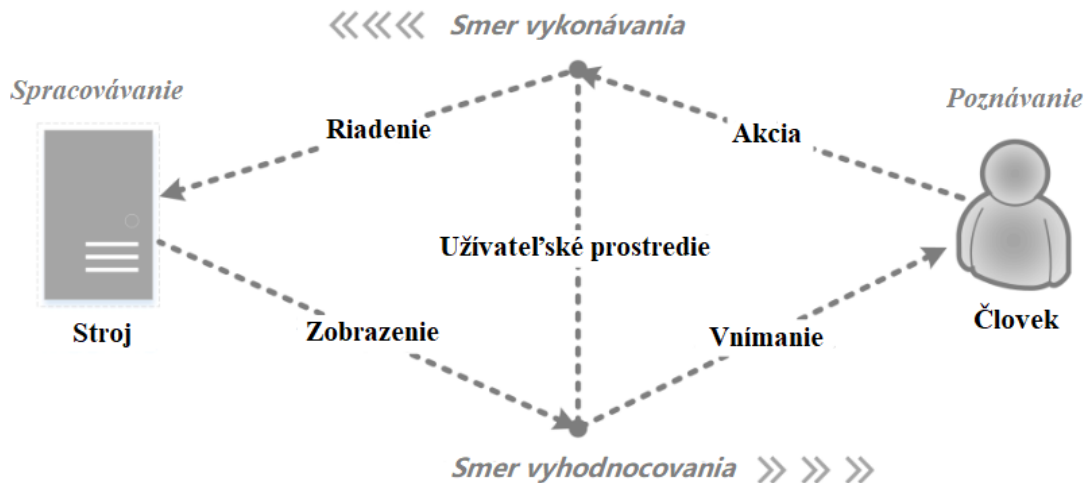
HMI z anglického slova Human machine interface, čo v preklade znamená Rozhranie človek – stroj, predstavuje užívateľské prostredie umožňujúce interakciu človeka so strojom alebo systémom. V súčasnosti sa toto označenie používa primárne v spojitosti s priemyslom. [17]

Programovateľné logické radiče (PLC) boli vytvorené na prijímanie vstupných signálov, ich spracovanie a následné odosielanie výstupných signálových inštrukcií. Samotné PLC však nedokáže zobrazit' aktuálne vykonávané procesy a hodnoty signálov tak, aby ich človek, resp. operátor mohol sledovať a upravovať. Z toho dôvodu bolo vyvinuté špecializované zariadenie s programovateľným displejom, ktoré dokáže zobrazit' aktuálne nastavenia a signály PLC a umožniť tak operátorovi s nimi pracovať podľa vopred naprogramovaných kritérií. [17] [18]

V priemysle sa obrazovky HMI môžu používať ako jednoúčelové monitorovacie zariadenia na sledovanie vykonávaných aktuálnych procesov, alebo môžu mať rozšírenú funkciu umožňujúcu operátorovi zadávať inštrukcie do PLC cez dotykovú obrazovku, alebo hardvérové tlačidlá umiestnené v okolí obrazovky. [17] [18]

V posledných desaťročiach vznikli viaceré výskumy zaoberajúce sa HMI užívateľským prostredím, ktorých cieľom bolo analyzovať a vytvoriť efektívne a bezpečné riešenie tejto problematiky. Ako už bolo načrtnuté, HMI predstavuje hmatateľnú časť vizualizovaného softvéru, ktorá spája dve základné zložky priemyslu (človek a stroj). Na obrázku 8 je znázornený komunikačný model, ktorého stredová línia reprezentuje užívateľské prostredie. Tento pomyselný most spája poznanie človeka, ktoré sa skladá z prijímaných vnemov a vykonávaných činov so strojným spracovaním riadiacich signálov a ich vizuálnej prezentácie. V prípade, ak navrhnuté užívateľské prostredie je chybné, resp. nedostatočne naprojektované, HMI bude mať za následok negatívne ovplyvnený výsledok pracovnej úlohy. [19] [20] [21]





Obr. 8) Model komunikačného cyklu medzi človekom a strojom [20]

Správne zvolený a navrhnutý dizajn užívateľského prostredia je kľúčovým faktorom ovplyvňujúci ergonómiu komunikácie človeka so strojom. Obrázok 9 prezentuje, z akých rôznych častí sa táto ergonómia skladá a čo všetko treba pri navrhovaní užívateľského prostredia brať do úvahy. Hlavný zámer ergonomicky správne navrhnutého HMI je zvýšenie kvality vzájomnej interakcie a produktivity celého systému a zároveň znižovanie nebezpečenstva vzniku chýb, únavy alebo zranení. Kvôli neustále rozširujúcej sa komplexnosti používaných zariadení sa zistilo, že užívatelia strácajú prehľad v ovládaní zložitých produktov a tým pádom nevyužívajú všetky funkcie, ktoré majú k dispozícii. V rámci výskumného projektu HÜMNOS z roku 1995 sa zistilo, že operátori zvyčajne využívajú iba polovicu funkcií, ktoré zariadenie ponúka. Dôvodom tejto skutočnosti bolo navrhovanie užívateľského prostredia a softvéru, ktorého základ bol založený na funkčno-technologickom princípe, zatiaľ čo užívateľ sa prirodzene sústreďuje na princíp riešenia jednotlivých úloh. [19] [20]


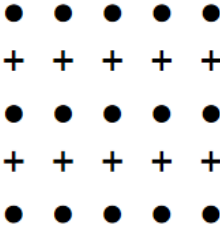
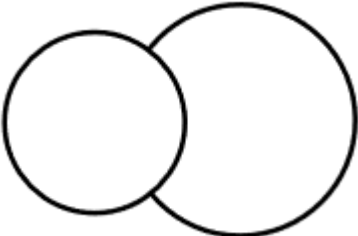
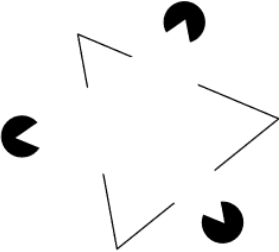
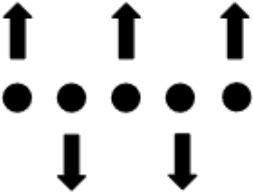


Obr. 9) Ergonómia užívateľského prostredia [21]



Po preukázaní faktov dôležitosti správneho prístupu pri navrhovaní užívateľského prostredia sa upriamuje pozornosť na konkrétne potreby a preferencie užívateľa. Namiesto toho, aby užívateľ bol nútený prispôbovať štýl práce funkcionalite zariadenia, sú užívateľské prostredia zariadení navrhované tak, aby zodpovedali štýlu práce. Tento prístup využíva princípy psychológie a spôsobu vnímania, ktoré boli zhrnuté do tzv. „Gestaltových zákonov“ (Tab 1). [19]

Tab 1) Výber pravidiel z Gestaltových zákonov [19]

	<p style="text-align: center;"><b>Zákon blízkosti</b></p> <p>Priestorová alebo časová blízkosť vedie k vnímaniu ucelených celkov. Prvky, ktoré sú bližšie pri sebe budú vnímané ako súvislý objekt.</p> <p style="text-align: center;">Vnímate skôr štyri páry čiar než osem samostatných čiar</p>
	<p style="text-align: center;"><b>Zákon podobnosti</b></p> <p>Podobné prvky sú vnímané ako súčasť rovnakej skupiny. Podobnosť môže závisieť od vzťahov skupín, vzorov, farieb, veľkosti a iných.</p> <p style="text-align: center;">Celkový vzor vnímame skôr ako zoskupenie radov kruhov, striedajúcich sa s radmi krížov.</p>
	<p style="text-align: center;"><b>Zákon pekných tvarov</b></p> <p>Objekt sa bude zdať byť usporiadaný do symetrického, jednoduchého a pravidelného tvaru.</p> <p style="text-align: center;">Objekt pripomína skôr dva prekrývajúce sa kruhy, než kombináciu niekoľkých komplikovaných tvarov.</p>
	<p style="text-align: center;"><b>Zákon uzatvárania</b></p> <p>V prípade medzier medzi pravidelnými obrysami máme tendenciu tieto objekty uzatvárať. Nekompletné objekty tak vnímame ako prepojené celky.</p>
	<p style="text-align: center;"><b>Zákon spoločného osudu</b></p> <p>Ak sa objekty pohybujú rovnakým smerom, máme tendenciu ich vnímať ako jeden celok. Ak sa na obrázku bodky 1, 3, 5 hýbu nahor a bodky 2, 4 nadol, bodky sú vnímané ako separátne skupiny.</p>

Dôležitú úlohu pri navrhovaní užívateľských prostredí HMI zohrávajú vytvorené štandardy pre ergonómiu a interakciu človek-systém ISO 9241 a procesy návrhu interaktívnych systémov zameraných na človeka ISO 13407. Tieto štandardy obsahujú princípy, akým spôsobom by sa tvorba užívateľských prostredí mala riadiť. Nakoľko však tempo, akým technológie a prístup k technológiám napreduje, tieto štandardy môžu prísť v istých ohľadoch zastaralé. Vo všeobecnosti však platí päť základných princípov, ktoré by sa mali dodržiavať pri vytváraní užívateľských prostredí HMI: [19]

- 1) Vhodnosť pre danú úlohu – HMI by malo byť prispôsobované schopnostiam užívateľa pre vykonanie konkrétnej úlohy.
- 2) Jednoduchosť – HMI by malo ponúkať jasné a pochopiteľné možnosti komunikácie medzi strojom a človekom.
- 3) Kontrolovateľnosť – užívateľ by mal byť schopný nastaviť požadované kroky a rýchlosti interakcie.
- 4) Tolerancia chýb – HMI by malo predvídať, akých chýb sa užívateľ môže dopustiť a prispôbiť tomu adekvátnu reakciu programu.
- 5) Podpora učenia – HMI by malo byť podporovať resp. uľahčovať proces učenia práce s programom.

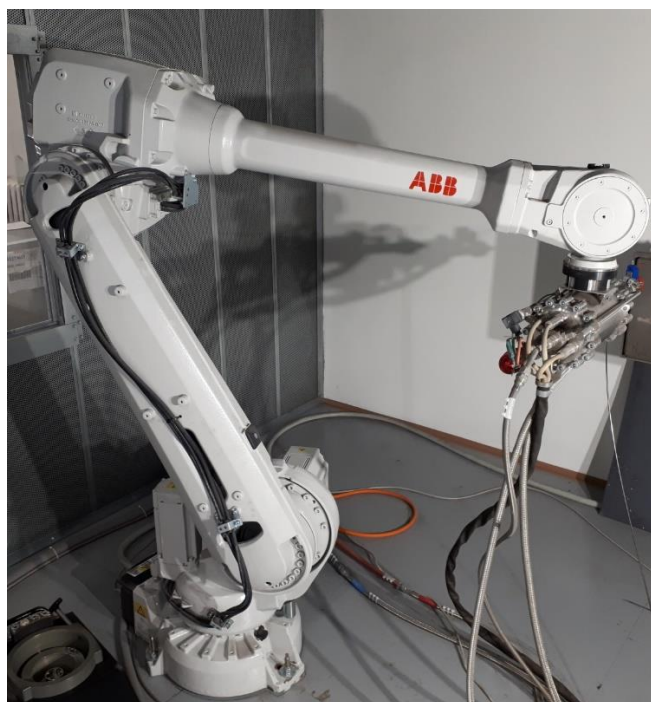
## 3 ANALÝZA ROBOTICKÉHO PRACOVISKA

Cieľom tejto kapitoly je vytvorenie kompletnej analýzy skutočného robotického pracoviska pre technológiu studeného nástreku, nachádzajúceho sa na ústave materiálového inžinierstva Vysokého učení technického v Brne. Robotické pracovisko je zložené z:

1. Priemyselného robota IRB 4600
2. Priemyselného polohovadla IRBP A-250
3. Priemyselného ovládača robotov IRC 5
4. Cold Spray System 5/11

### 3.1 Priemyselný robot IRB 4600

Hlavný prvok celej robotickej bunky predstavuje 6 osí priemyselný robot IRB 4600 od spoločnosti ABB zobrazený na obrázku 10. IRB 4600 patrí medzi rýchle a kompaktné roboty s nízkou hmotnosťou a dlhými ramenami, umožňujúce umiestnenie robota bližšie k polohovadlu, čím sa znižuje veľkosť pracoviska a zvyšuje produktivita. Robot má stupeň ochrany IP 67, ktorý poskytuje kompletnú prachotesnosť, vode-odolnosť, ochranu proti korózií a ochranu nepohyblivej kabeláže robota. Táto ochrana je dostačujúca pre použitie robota v spojitosti s technológiou studeného nástreku. Z dôvodu jednoduchosti upevnenia a dostatočného miesta v hale je podstava robota ukotvená v podlahe približne 1,2 metra od polohovadla. V tabuľkách 2 a 3 sú uvedené technické vlastnosti robota IRB 4600, zatiaľ čo obrázok 11 zobrazuje jeho maximálny pracovný dosah. [22]



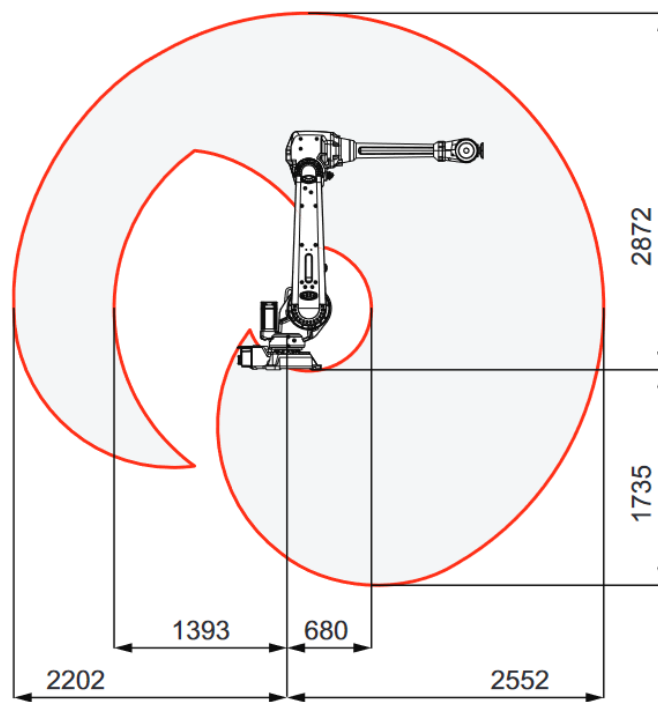
Obr. 10) Priemyselný robot IRB 4600 používaný na pracovisku

Tab 2) Tabuľka technických vlastností priemyselného robota IRB 4600 [22]

Dosah [m]	Hmotnosť [kg]	Užitočné zaťaženie [kg]	Zaťaženie ramena [kg]	Opakovateľnosť dojazdu do pozície [mm]	Opakovateľnosť dráhy [mm]
2,55	435	40	20	0,06	0,28

Tab 3) Tabuľka rozmedzí uhlového natočenia a rýchlosti osí robota IRB 4600 [22]

Os 1	Os 2	Os 3	Os 4	Os 5	Os 6
+180° do -180°	+150° do -90°	+75° do -80°	+400° do -400°	+120° do -125°	+600° do -600°
175 °/s	175 °/s	175 °/s	250 °/s	250 °/s	360 °/s



Obr. 11) Pracovný dosah robota IRB 4600 v [mm] [22]

### 3.2 Polohovadlo IRBP A-250

Priemyselné robotické polohovadlo IRBP A-250 od spoločnosti ABB je vhodné pre použitie v robotických bunkách, kde je predpokladaná rotačná manipulácia obrobkom v dvoch osiach. Ide o polohovadlo s robustnou konštrukciou určený do náročných priemyselných podmienok, ktorého veľkou výhodou je jeho kompatibilita so šesť osími priemyselnými robotmi od firmy ABB. Polohovadlo je vďaka tomu možné ovládať a programovať spolu s priemyselným robotom IRB 4600 v rámci jednotného ovládača PLC. V nižšie uvedených tabuľkách 4 a 5 sú uvedené hlavné technické vlastnosti a rozmery polohovadla. [23]



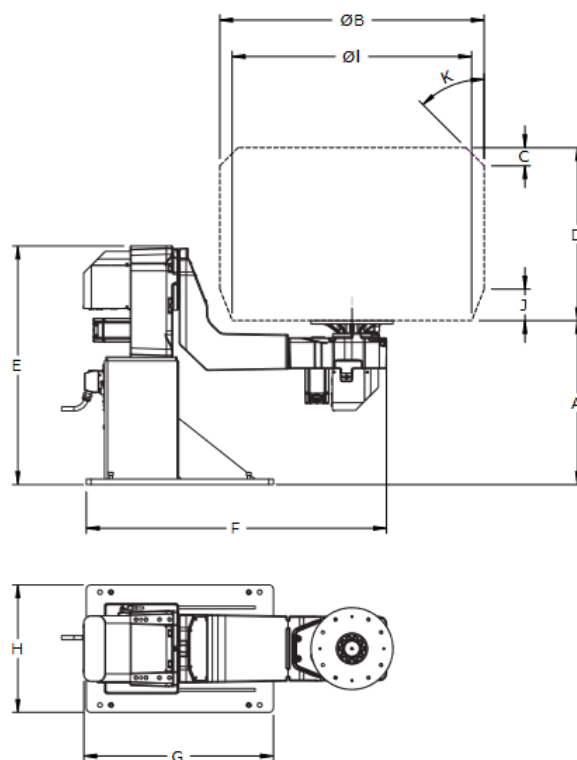
Obr. 12) Polohovadlo IRBP A-250 používaný na pracovisku

Tab 4) Tabuľka technických vlastností polohovadla IRBP A-250 [23]

Manipulačná kapacita [kg]	Maximálny nepretržitý moment [Nm]	Opakovateľná presnosť [r = 500]	Maximálna uhlová rýchlosť osy 1 [°/s]	Maximálna uhlová rýchlosť osy 2 [°/s]
250	350	+/- 0,05	150	180

Tab 5) Tabuľka rozmerových hodnôt polohovadla na Obr. 13, hodnoty v [mm] [23]

A	øB	C	D	E	F	G	H	øI	J	K
900	1180	0	900	1210	1293	910	440	1000	173	0°



Obr. 13) Rozmerová schéma polohovadla IRBP A-250 [23]

### 3.3 Priemyselný ovládač robotov IRC 5

Priemyselný ovládač IRC 5 predstavuje najnovšiu generáciu priemyselných ovládačov od firmy ABB. IRC 5 slúži ako základný bezpečnostný a komunikačný prvok medzi operátorom a robotickou bunkou v rámci priemyselného prostredia, ktorý riadi všetky pohyby priemyselného robota a polohovadla. Pomocou jednotného programovacieho jazyka RAPID, ktorým komunikujú robotické systémy ABB, je operátorovi umožnené vytvárať komplexné automatizované riadiace programy ovládajúce všetky časti robotických buniek súčasne. Existujú dva možné spôsoby tvorby riadiacich programov, ktoré má operátor k dispozícii. [24]

Prvou možnosťou je využitie doplnkového ovládacieho rozhrania FlexPendant, ktorý predstavuje prenosný ovládač zložený z hardvérových tlačidiel a dotykovej obrazovky slúžiacich na kompletne riadenie všetkých robotických súčastí pripojených k priemyselnému ovládaču IRC 5. Pomocou tohto užívateľského ovládača má operátor na obrazovke neustále k dispozícii prehľad kompletneho nastavenia systémov v robotической bunke, dokáže vytvárať programy automatických cyklov robota, manuálne riadiť všetky robotické súčasti podľa potreby a mnohé ďalšie. [25]

Druhou možnosťou ako vytvárať riadiace programy robotického systému je využitie softvéru Robotstudio na akomkoľvek PC, ktoré ponúka spoločnosť ABB pre offline tvorbu programových a logických štruktúr ovládania robotической bunky. Softvér Robotstudio predstavuje zjednodušený spôsob prípravy programu na počítači bez nutnosti zastavenia výroby. Jadro Robotstudia je postavené na virtuálnom riadiacom systéme, ktorý je identický so skutočným riadiacim systémom robotической bunky. To umožňuje vytvárať realistické simulácie,

v rámci ktorých je možné vyladiť konfigurácie reálnych robotických programov ešte predtým, než sú nahrané do priemyselného ovládača IRC 5. [26]



Obr. 14) Priemyselný ovládač IRC 5 používaný na pracovisku

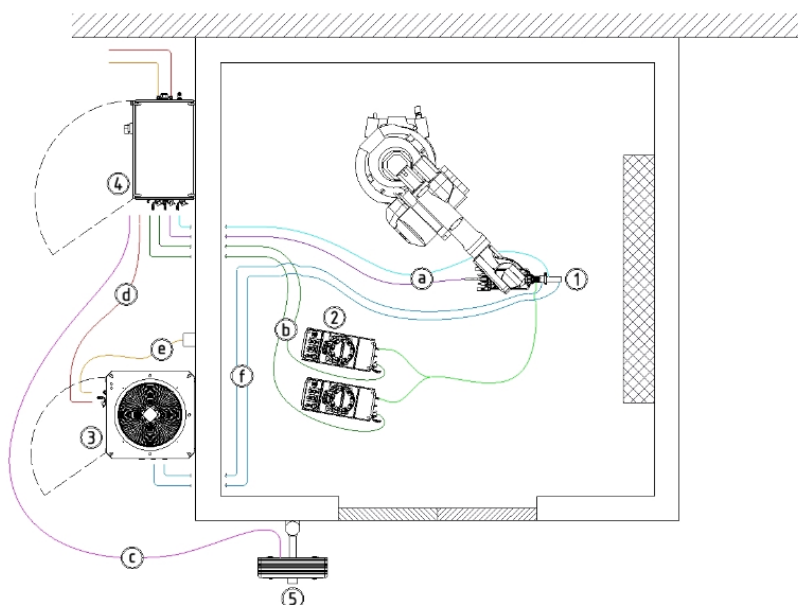
### 3.4 Cold Spray System 5/11

Neodmysliteľnou súčasťou robotického pracoviska studeného nástreku je samotný systém využívanej technológie studeného nástreku. Použitý je systém spoločnosti Impact Innovations, ktorý je zložený z dohromady piatich samostatných jednotiek:

- 1) Impact Gun 5/11
- 2) Impact Powder Feeder 5/11
- 3) Impact Water Cooling 5/11
- 4) Impact Control Unit 5/11 Control Cabinet
- 5) Impact Control Unit 5/11 Control Panel

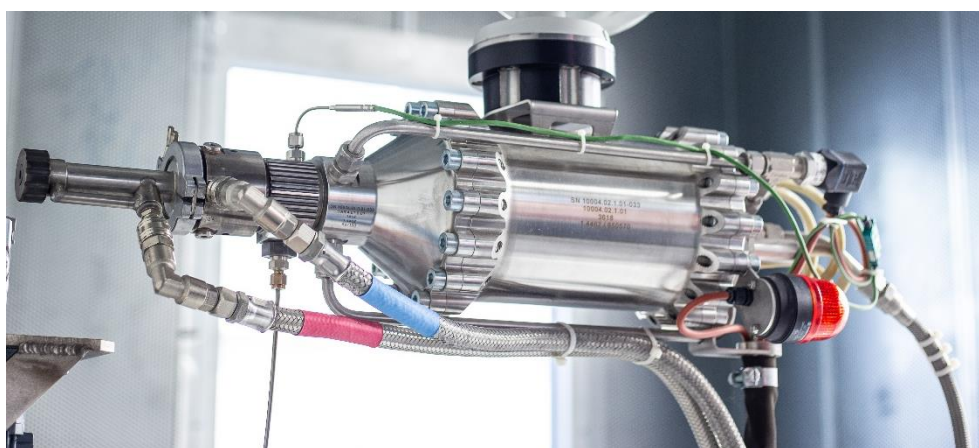
Zoradenie jednotiek použitých v systéme studeného nástreku zodpovedá číselnému označeniu súčastí v schéme robotického pracoviska na obrázku 15.





Obr. 15) Schéma robotického pracoviska využívajúce technológiu studeného nástreku [27]

Prvou a hlavnou jednotkou je Impact Gun 5/11 (Obr. 16), čo označuje hlavicu s dýzou upevnenou na konci priemyselného robota. V tejto hlavici sa ohrievaný plyn zlučuje s prachovými časticami nanášaného materiálu, ktoré sú cez dýzu vystreľované von. Celkové rozmery hlavice predstavujú 784 mm x 209 mm x 223 mm, zatiaľ čo hmotnosť hlavice upevnenej na priemyselnom robote je 27 kg. V zadnej časti hlavice je pripojená elektrická kabeláž, ako aj hadice privádzajúce pracovný plyn (dusík). V prednej časti sú viditeľné pripojenia chladiaceho média a zo spodnej časti je privádzaný práškový materiál. [27]



Obr. 16) Hlavica studeného nástreku používaná na pracovisku

Druhou v poradí je jednotka dávkovacieho zariadenia Impact Powder Feeder 5/11 (Obr. 17), ktoré zodpovedá za kontrolovaný prívod prášku nanášaného materiálu do dýzy hlavice. Zlatistá nádoba s rúčkou, ktorá je upevnená z vrchnej časti dávkovacieho zariadenia obsahuje až 3,7 litra pripraveného práškového materiálu určeného na striekanie, ktorý sa následne pod tlakom



privádza priamo do hlavice Impact Gun. Maximálna objem dávkovania predstavuje 1,5 litra prášku za hodinu. Zariadenie sa nachádza uložené na zemi v primeranej vzdialenosti od priemyselného robota, pričom výstupná trubica na prívod materiálu je pripojená k hlavici tak, aby neboli obmedzené pohybové trajektórie robota. [28]



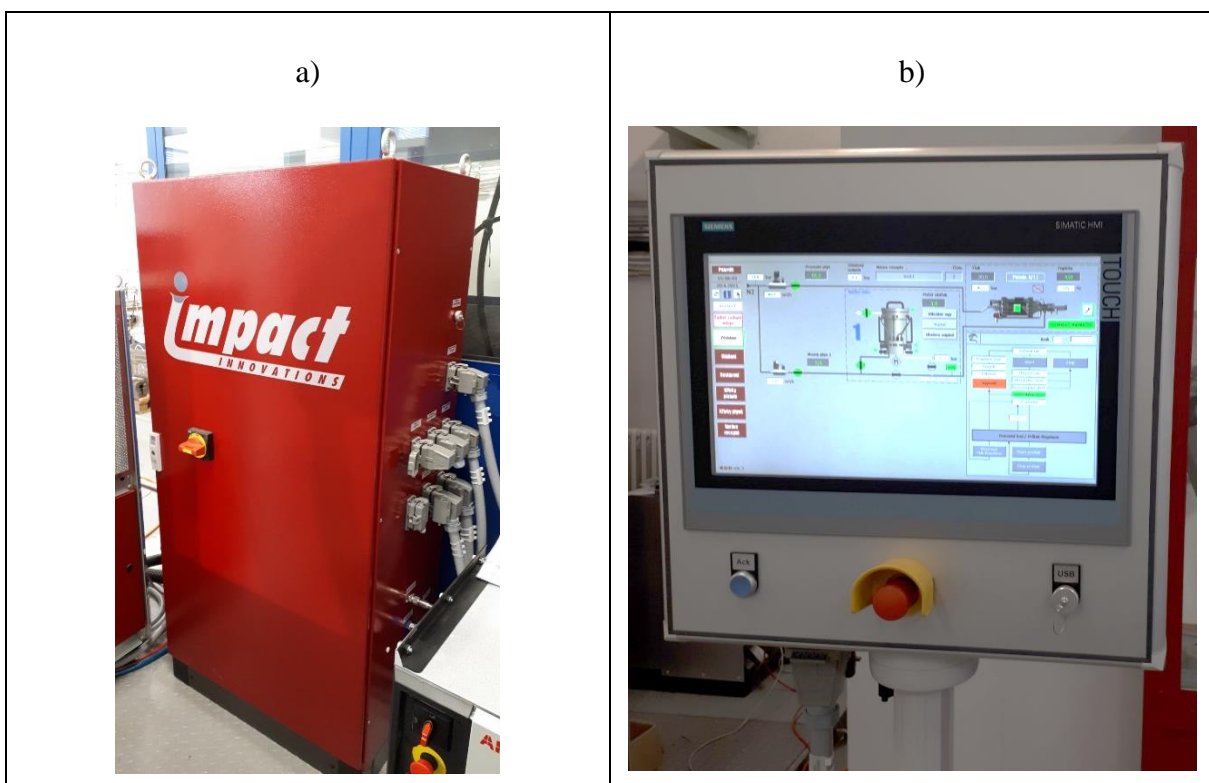
Obr. 17) Dávkovacia jednotka prášku používaná na pracovisku

Ďalšou v poradí je chladiaca jednotka Impact Water Cooling 5/11 (Obr. 18), ktorej plniaca kapacita predstavuje 120 litrov. Jej úlohou je zabezpečiť dostatočnú cirkuláciu destilovanej chladiacej vody v oblasti dýzy hlavice studeného nástreku. V tejto oblasti hlavice sa dôsledkom pôsobenia vysokého tlaku a ohrievaného plynu prehrieva. Nepriaznivé zvýšenie teploty je limitované prívodom ochladzovanej vody skrz prvú chladiacu hadicu do miesta ohrevu, odkiaľ sa ohriata voda odčerpáva skrz druhú hadicu do chladiacej jednotky. [27]



Obr. 18) Chladiaca jednotka vody používaná na pracovisku

Centrálneou súčasťou celého systému využívanej technológie studeného nástreku je hlavná ovládacia jednotka Impact Control Unit 5/11 Control Cabinet (Obr. 19). Okrem toho, že táto jednotka je pripojená ku každej zo spomenutých samostatných jednotiek, s ktorými komunikuje a riadi ich činnosť, predstavuje aj hlavný prívod stlačeného pracovného plynu do hlavice studeného nástreku. Konkrétne ovládacia tlaková jednotka na pracovisku je prispôbena na prácu s čistým dusíkom, pričom pri maximálnom tlaku 50 barov je možné dosiahnuť prietok plynu až 190 m<sup>3</sup>/h. Neoddeliteľnou súčasťou ovládacej jednotky je ovládací panel, ponúkajúci HMI užívateľské prostredie, umožňujúce operátorovi nastavovať a kontrolovať všetky procesy spojené s technológiou studeného nástreku. 19-palcový displej ovládacieho panelu poskytuje intuitívny softvér na správu všetkých procesných parametrov, aktuátorov a senzorov, ktoré sú obsiahnuté v systéme. [29] [30]



Obr. 19) a) Hlavná ovládacia tlaková jednotka a b) ovládací panel používaná na pracovisku

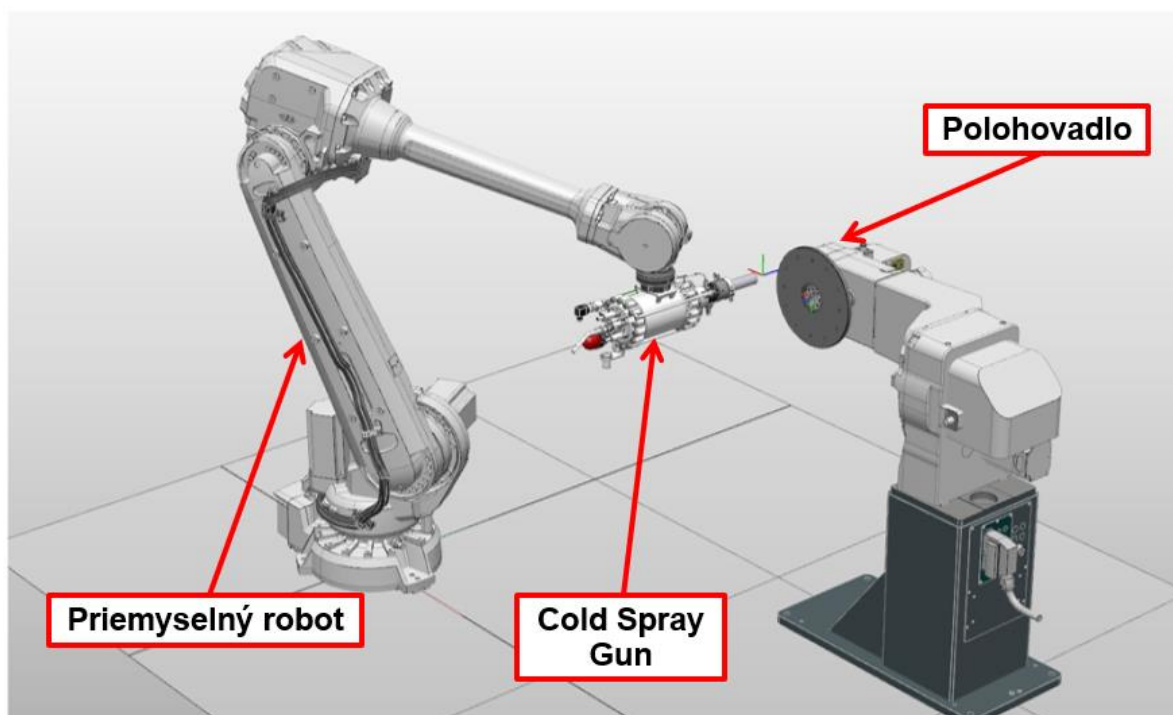
## 4 VIRTUÁLNE SPREVÁDZKOVANIE

Táto kapitola sa venuje vypracovaniu návrhu užívateľského softwaru a užívateľského ovládacieho rozhrania robotického pracoviska a jeho digitálne uvedenie do prevádzky. V podkapitolách sú vysvetlené jednotlivé časti tvorby programu, od primárnych modulov až po ich vzájomné prepojenie. Každá z častí predstavuje neoddeliteľnú súčasť návrhu, ktorého výsledkom je virtuálne uvedenie pracoviska do prevádzky. Dodatkou k tejto práci je užívateľský manuál ovládania vytvoreného programu, ktorý je prílohou diplomovej práce. Cieľom tejto práce je vytvoriť bezpečné, jednoduché a spoľahlivé ovládanie priemyselného robota používaného na skúmanie metód technológie studeného nástreku na Ústave materiálových vied a inžinierstva fakulty strojného inžinierstva VUT (ďalej už iba ako zákazník). Užívateľský software ako aj celá logika ovládania sú vytvárané v programovom prostredí Robotstudio od spoločnosti ABB, keďže obsluhované pracovisko je zložené z konkrétnych zariadení tejto firmy. Na začiatku mi bol poskytnutý virtuálny model pracoviska, v ktorom umiestnenie zariadení zodpovedá tomu skutočnému. 3D virtuálne dvojča robotického pracoviska bolo získané z Ústavu výrobných strojov systémov a robotiky fakulty strojného inžinierstva VUT.

### 4.1 Rozbor 3D virtuálneho pracoviska

Virtuálne pracovisko pozostáva z troch hlavných častí:

- Priemyselný robot IRB 4600
- Priemyselné polohovadlo IRBP A-250
- Spray Gun 5/11 Cold Spray System



Obr. 20) Rozbor pracoviska v prostredí programu Robotstudio

## 4.2 Vstupné požiadavky na program

Program a jeho digitálne uvedenie do prevádzky musí spĺňať viacero požiadaviek, ktoré boli predložené zákazníkom. Jednou z najdôležitejších požiadaviek je bezpečnosť. Program musí byť navrhovaný tak, aby nemohol nastať nekontrolovateľný pohyb robota, ktorý by mohol spôsobiť zranenie obsluhy alebo kolíziu robota s jeho okolím. Program by mal brať do úvahy veľkosť zariadenia Spray Gun a jeho kabeláž, nakoľko tieto rozmery ovplyvňujú v značnej miere pohyblivosť celej sústavy. Z tohto dôvodu by sa maximálna rýchlosť Tool Center Point (ďalej ako TCP) mala pohybovať v rozmedzí 350-450 mm/s, pričom pracovnú rýchlosť TCP počas nanášania materiálu si užívateľ zvolí sám.

Ďalšou požiadavkou zákazníka bola možnosť výberu vlastného pracovného cyklu a nastavenia počtu pracovných cyklov. Pracovné cykly by mali obsahovať horizontálne, vertikálne a diagonálne prechody dýzy nad plochou nanášania. Obsluha by si mala možnosť vybrať konkrétny pracovný cyklus, respektíve ich kombinácie, čím by bolo možné doceliť optimálny a rovnomerný nástrek materiálu na povrch. Ďalšou požiadavkou bolo umiestnenie konca dýzy hlavice Cold Spray Gun vo vzdialenosti 30 mm nad plochou nanášania z dôvodu dosiahnutia najvyššej efektivity pôsobenia nanášaného materiálu na obrábaný povrch. Zákazník taktiež požadoval možnosť natáčania dýzy v dvoch osiach rotácie.

Súčasťou programu by malo byť aj užívateľské prostredie Human Machine Interface (HMI), umožňujúce obsluhu rýchle a presné ovládanie priemyselného robota a manipulátora. V rámci tohto užívateľského prostredia majú byť k dispozícii ovládacie prvky na:

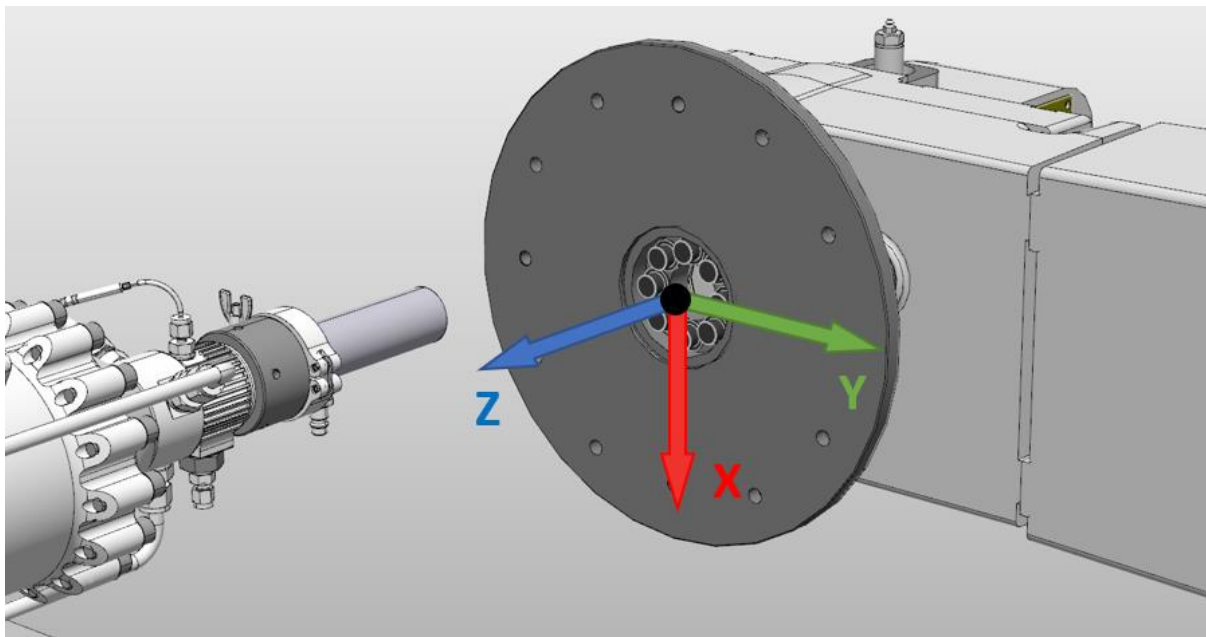
1. Spustenie automatického pracovného cyklu.
2. Zastavenie pracovného cyklu s okamžitým vrátením robota do bezpečnej pozície.
3. Pozastavenie pracovného cyklu na konci pohybu robota a možnosť pokračovať z miesta, kde bol pohyb prerušený.
4. Vytvorenie ukazovateľov aktuálneho pracovného času operácií.
5. Vytvorenie ukazovateľov aktuálnej rýchlosti pohybu robota.
6. Grafický prehľad, v ktorej fáze procesu sa robot nachádza.

Ďalšou súčasťou, ktorú požadovaný program má obsahovať, je možnosť vloženia separátnych modulov do programovej štruktúry a umožnenie ich spustenie v rámci automatického cyklu programu. Užívateľ tak bude môcť vytvárať vlastné programové moduly, ktoré mu umožnia aplikovať studený nástrek na predmety rôznych tvarov, čím sa zabezpečí univerzálnosť užívateľského programu.

## 4.3 Definovanie súradnicových systémov

Virtuálne pracovisko vo svojom pôvodnom stave bolo rozložené v súradnicovom systéme *World*, ktorého počiatok sa nachádzal uprostred podstavy priemyselného robota IRB 4600, kde rovina XY bola orientovaná paralelne s podlahou a os Z smerovala nahor. Tento súradnicový systém sa v rámci programu Robotstudio označuje ako základný pracovný objekt „wobj0“. V rámci riešenia projektu sa však tento pracovný objekt nevyužíva a nahrádza ho pracovný objekt, ktorého počiatok sa nachádza uprostred upínacej dosky polohovadla, kde rovina XY je

rovnobežná s doskou a osa Z smeruje z dosky von. Výhodou takto novovytvoreného pracovného objektu „Workobject\_1“ (Obr. 21) je, že predstavuje možnosť jednoduchšieho orientovania TCP nástroja počas vytvárania programových cyklov.

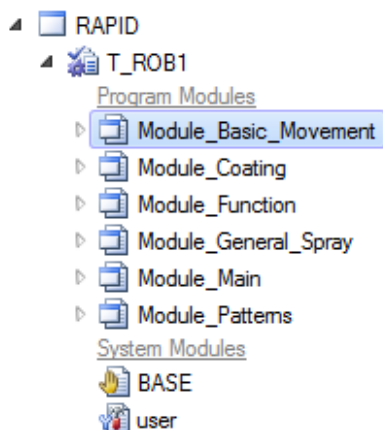


Obr. 21) Súradnicový systém pracovného objektu „Workobject\_1“

#### 4.4 Programová štruktúra

Ako už bolo spomínané vyššie, program a jeho súčasti boli tvorené v rámci softvéru Robotstudio, ktorý využíva programovací jazyk RAPID. Hlavná programová štruktúra je zložená zo šiestich programových modulov a dvoch systémových modulov. Programové moduly rozdeľujú program do prehľadných celkov, pričom každý z nich je zložený z podprogramov a deklarovaných dát. Primárnym modulom je *Module\_Main*, ktorý obsahuje hlavnú programovú rutinu *main()*, v rámci ktorej sú volané všetky podprogramy z ostatných modulov. Tento modul taktiež obsahuje dve Trap rutiny, resp. podprogramy reagujúce na vzniknuté prerušenie v rámci prebiehajúceho programového cyklu. Funkcia Trap rutín bude vysvetlená v kapitole 5.8 – Hlavný modul Main. V module *Module\_Basic\_Movement* sa nachádzajú všetky dáta a podprogramy definujúce základné pohybové inštrukcie, pričom moduly *Module\_Coating* a *Module\_Patterns* tieto pohybové inštrukcie spájajú do súvislých pohybových celkov zodpovedajúcich zadaným požiadavkám obsluhy robota. V rámci programovej štruktúry sa nachádzajú taktiež podprogramy modulu *Module\_General\_Spray*, ktoré umožňujú spravovanie externých modulov, ktorých funkcia je vysvetlená v kapitole 5.7 – Externý modul všeobecného programu. Posledným programovým modulom programovej štruktúry je *Module\_Function*, v ktorom sú definované doplnkové funkcie používané v užívateľskom softvéri.

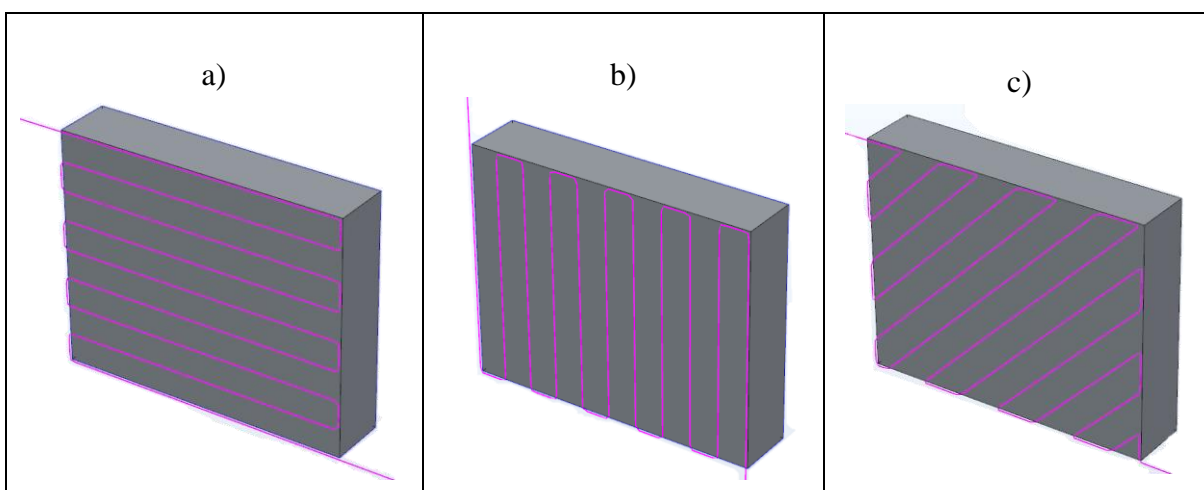




Obr. 22) Zobrazenie programovej štruktúry v programe Robotstudio

#### 4.5 Základné pohybové procedúry

Základné pohybové procedúry (rutiny/podprogramy) programu sú definované v rámci modulu *Module\_Basic\_Movement*. Základnou pohybovou procedúrou sa rozumie kontinuálny pohyb TCP priemyselného robota po vopred nadefinovanej trajektórii, nachádzajúcej sa v určitej vzdialenosti nad povrchom súčiastky. V rámci programu sú vytvorené tri základné pohybové procedúry: horizontálna, vertikálna a diagonálna. Každá z týchto procedúr vytvára svojím pohybom TCP riadkovú trajektóriu, ktorej tvar zodpovedá pravouhlému štvorhranu zobrazenom na obrázku 23.

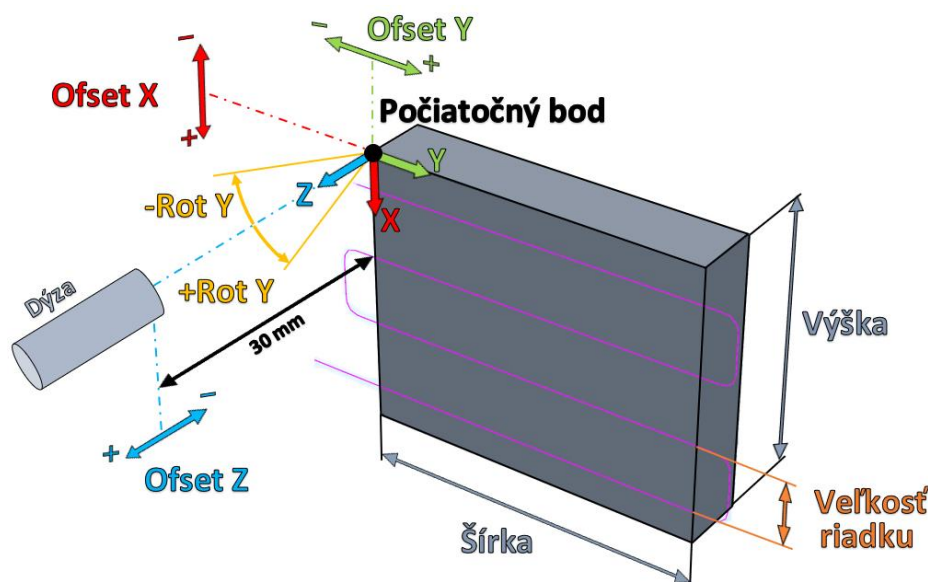


Obr. 23) Príklad a) horizontálnej, b) vertikálnej, c) diagonálnej procedúry

Pohyb horizontálnej procedúry je charakterizovaný vodorovnými prechodmi dýzy Cold Spray Gun ponad obrábaný objekt. Obdobne vertikálny pohyb je tvorený zo zvislých prechodových trajektórií ponad obrábanú súčiastku. Procedúra diagonálneho pohybu vykonáva priame prechody nad súčiastkou pri uhlovom natočení každého prechodu o 45 stupňov v rovine XY.

Každá z pohybových procedúr potrebuje na začiatku užívateľom nastavené vstupné parametre, na základe ktorých program automaticky vypočíta body, ktorými bude TCP prechádzať. Tieto vstupné parametre sú:

- 1) počiatkový bod,
- 2) šírka súčiastky, ktorá bude striekaná,
- 3) výška súčiastky, ktorá bude striekaná,
- 4) riadkovanie – vzdialenosť medzi jednotlivými prechodmi TCP,
- 5) ofset v ose X,
- 6) ofset v ose Y,
- 7) vzdialenosti konca dýzy pištole studeného nástreku od povrchu súčiastky,
- 8) natočenie konca dýzy okolo osi X a Y.



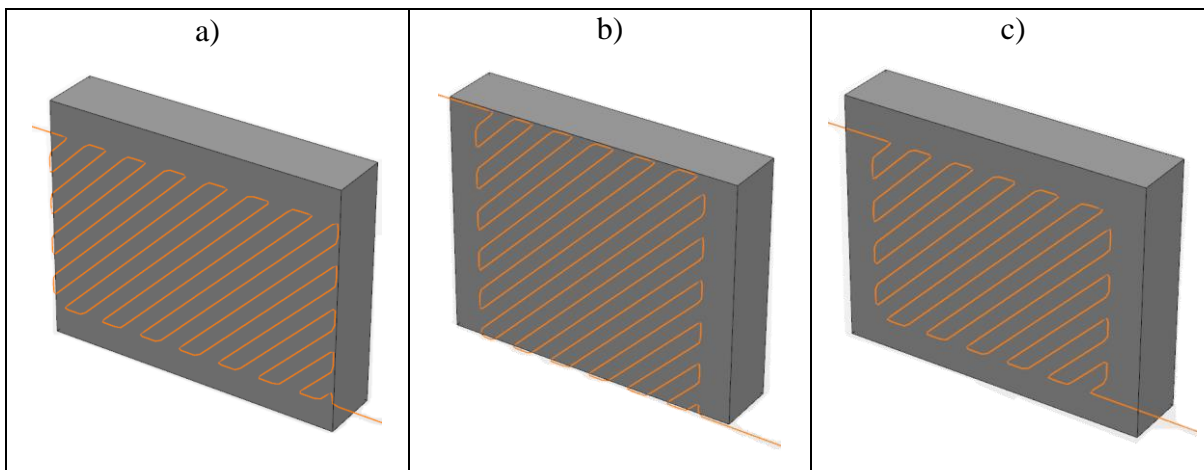
Obr. 24) Zobrazenie vstupných parametrov nastaviteľné obsluhou

Počiatkový bod je bod, ktorý si obsluha robota volí sama manuálnym navigovaním robota do požadovanej pozície. Tento bod sa musí (!) vždy nachádzať v ľavom hornom rohu súčiastky, ktorá bude striekaná. Programová logika totiž odvodzuje všetky nasledujúce body z tohoto počiatkového bodu. V programe sa tento bod označuje názvom „START\_POINT“ a predstavuje počiatok lokálneho súradnicového systému s hodnotami TCP (0,0,0).

V rámci programu je umožnené nastavovanie ofsetov vo všetkých troch osiach pracovného objektu. Ofsety slúžia na parametrické upravenie trajektórie TCP, pričom zadaná hodnota sa pripočítava k hodnote danej osy počiatkového bodu. Grafické zobrazenie vlastností ofsetov je na obrázku 25, kde:

- a) Ofset X = 10 mm a ofset Y = 0 mm.
- b) Ofset X = 0 mm a ofset Y = 10 mm
- c) Ofset X = 10 mm a ofset Y = 10 mm.

Ako je možné vidieť, pomocou ofsetov si obsluha robota dokáže pozmeniť tvar dráhy TCP dýzy podľa potreby. V prípade použitia negatívnych ofsetov sa TCP bude presúvať vždy za hranice striekanej súčiastky, čo je výhodné v prípade, keď nad striekanou súčiastkou má dýza prechádzať iba priamymi prechodmi, bez viditeľných koncových zaoblení riadkov. Rovnako je možné nastavovať aj ofset v osi Z, čo obsluha poskytuje možnosť upraviť vzdialenosť medzi koncom dýzy pištole a povrchom súčiastky.



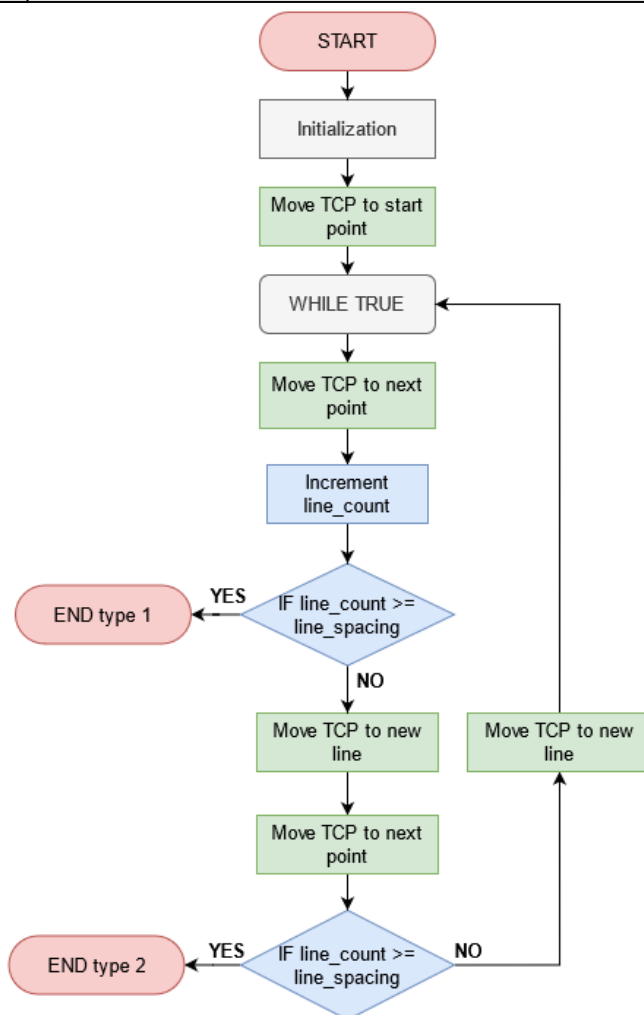
Obr. 25) Porovnanie nastavenia ofsetov pri diagonálnom nanášaní

#### 4.5.1 Podprogramy horizontálneho a vertikálneho pohybu

V rámci modulu *Module\_Basic\_Movement* sú okrem iného vytvorené štyri podprogramy vykonávajúce horizontálny pohyb TCP a štyri podprogramy vykonávajúce vertikálny pohyb TCP. Nakoľko základný program je univerzálny a vytvára nad striekaným povrchom trajektóriu opísanú štvorhranom, vytvoril som štyri procedúry pre každý z pohybov tak, aby pohyb dýzy mohol začínať v ktoromkoľvek z jeho rohov. Vďaka tomu sa zvyšuje adaptabilita celého systému a znižuje sa pracovný čas. Logická štruktúra všetkých procedúr horizontálnych a vertikálnych pohybov je navrhnutá rovnakým spôsobom, pričom rozdiel je v orientácii pohybu a začiatočnom bode. Zjednodušený vývojový diagram vytvárania vertikálnych a horizontálnych pohybov je znázornený na obr. 26, pričom každá pohybová inštrukcia je spojená s výpočtom nasledujúceho pohybu TCP.

Pri spustení každej pohybovej procedúry prichádza najprv k inicializácii premenných. V tejto fáze sa vynulujú počítadlá riadkov, vykonáva sa výpočet koľko riadkov sa bude nanášať a výpočet dĺžky prejazdu TCP ponad striekanú súčiastku. Následne sa spúšťa pohyb do počiatočného bodu, odkiaľ je spustená funkcia WHILE, ktorej ukončenie nastáva po splnení podmienky jednej z dvoch hlavných IF funkcií. Po jej splnení sa TCP presúva do bezpečnej vzdialenosti *Safe Zone* a program pokračuje vo vykonávaní ďalšej inštrukcie v poradí.



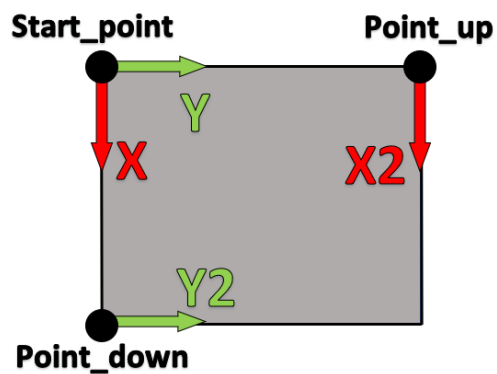


Obr. 26) Vývojový diagram pre horizontálny a vertikálny pohyb

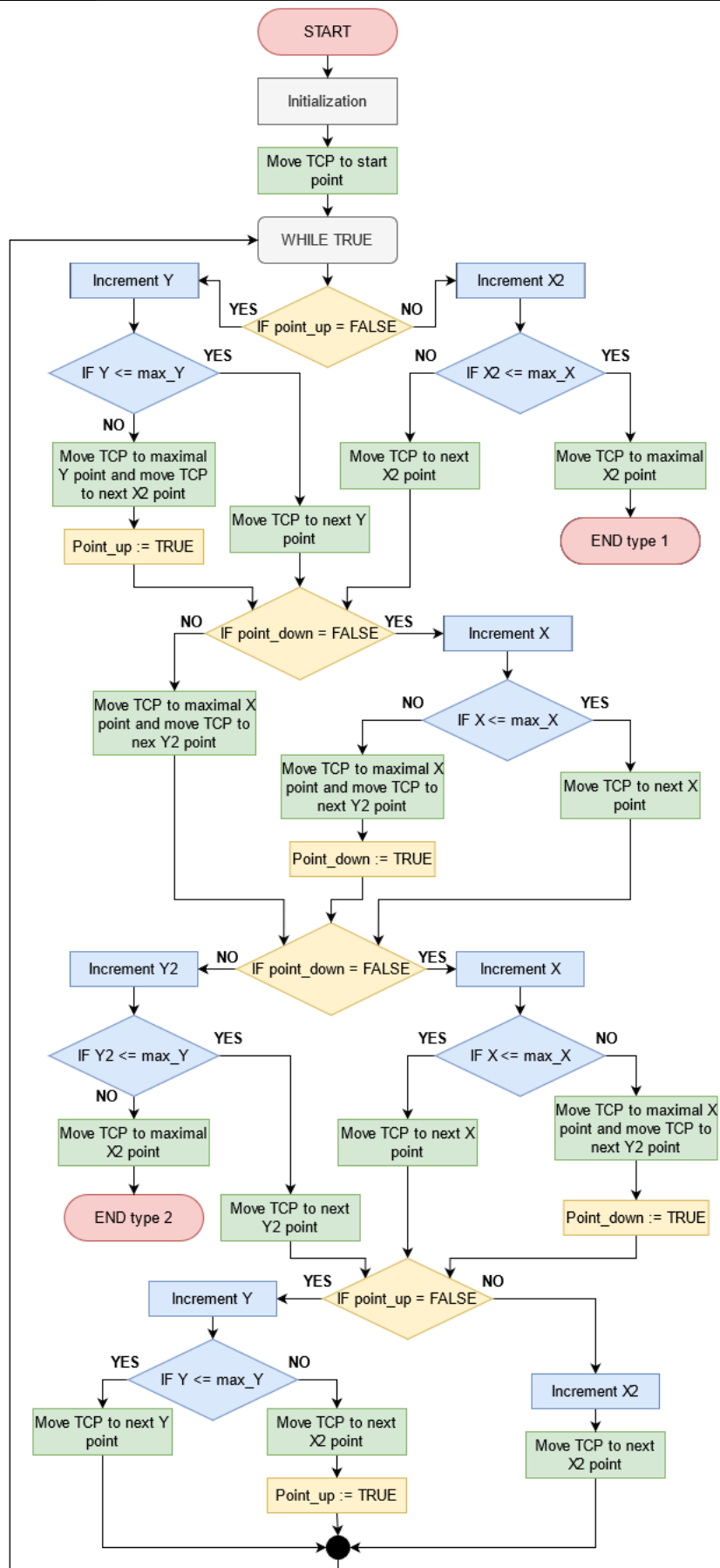
#### 4.5.2 Podprogramy diagonálneho pohybu

V zdrojovom kóde sa nachádzajú štyri podprogramy pre diagonálny pohyb začínajúci v každom rohu pomyselného štvorhranu. Procedúra diagonálneho pohybu je však podstatne komplexnejšia, než klasický podprogram pre horizontálny pohyb. Táto procedúra priraduje každej strane štvorhranu vlastnú os, čím je možné separovať pohybové inštrukcie pre jednotlivé osy zvlášť. Rozdelenie orientácie os je zobrazené na obr. 27. Po inicializácii programu sa TCP presúva do počiatočného bodu nad súčiastku, od ktorej sú vypočítavané všetky prechodové body. V tejto fáze sa spúšťa funkcia WHILE, ktorá prvotne vykoná presun TCP pozdĺž osy Y a vypočíta súradnice nasledujúceho bodu na osi X. TCP sa priamočiarno presunie do presne vygenerovanej vzdialenosti na ose X a vykoná ďalší pohyb pozdĺž tejto osy, čím sa TCP posunie o riadok nižšie. Robot sa následne premiestni opäť na presne vypočítanú pozíciu na osi Y, čím sa ukončil jeden cyklus a začína ďalší. V rámci programového cyklu sa pravidelne kontroluje pozícia TCP a vyhodnocuje sa, či sa nasledujúca pozícia TCP nachádza za bodmi *Point\_up* alebo *Point\_down*. V prípade, že áno, vykonajú sa trigonometrické výpočty, ktoré posunú TCP do jedného z týchto bodov a zvyšok pohybu presmerujú pozdĺž nasledujúcej osi. To znamená, že ak pohyb robota pozdĺž osi Y by mal siahať za bod *Point\_up*, program posunie

robot maximálne do bodu *Point\_up* a zvyšok pohybu sa prenesie na os X2 tak, aby sa zachovalo konštantné riadkovanie. Procedúra sa ukončuje až v prípade, keď sa TCP dostane do konečného bodu, ktorý sa nachádza na priesečníku osí X2 a Y2. Zjednodušený vývojový diagram podprogramu pre diagonálny pohyb je zobrazený na obr. 28. V rámci podprogramu sa nachádzajú ďalšie doplňujúce funkcie súvisiace s ovládaním celého programu pre studený nástrek, ktoré nie sú vo vývojom diagrame uvedené.



Obr. 27) Rozdelenie osí pri generovaní bodov trajektórie



Obr. 28) Vývojový diagram pre diagonálny pohyb

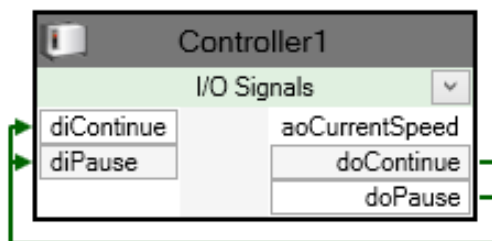
## 4.6 Použité I/O signály

Riadiace vstupno-výstupné signály sú neoddeliteľnou súčasťou celého projektu. Pomocou nich sú ovládané a spúšťané podprogramy v rámci celej programovej štruktúry. Zoznam vytvorených signálov je zobrazený na obr. 29.

Name	Type of Signal	Name	Type of Signal
aiCurrentSpeed	Analog Input	doEnablePattern21	Digital Output
aoCurrentSpeed	Analog Output	doEnablePattern22	Digital Output
diContinue	Digital Input	doEnablePattern31	Digital Output
diK1_ACT	Digital Input	doEnablePattern32	Digital Output
diK3_ACT	Digital Input	doEnableRotation	Digital Output
diMotorsON	Digital Input	doGeneralSpray	Digital Output
diPattern2	Digital Input	doGoHome	Digital Output
diPattern3	Digital Input	doGoToStartPoint	Digital Output
diPause	Digital Input	dolsHome	Digital Output
diPPToMain	Digital Input	doLoadProgram	Digital Output
doACT_K1	Digital Output	doPause	Digital Output
doACT_K3	Digital Output	doPauseButton	Digital Output
doACT_K11	Digital Output	doPPToMain	Digital Output
doACT_K13	Digital Output	doSearchDir	Digital Output
doAngularSpray	Digital Output	doStartGeneralModule	Digital Output
doContinue	Digital Output	doStartProgram	Digital Output
doContinueButton	Digital Output	doSTOPbutton	Digital Output
doEnableFlexpendant	Digital Output	doSwitch2	Digital Output
doEnableFlexpendant2	Digital Output	doSwitch3	Digital Output
doEnableFlexpendant3	Digital Output	doTeachHome	Digital Output
doEnableFlexpendant4	Digital Output	doTeachStartPoint	Digital Output
doEnableFlexpendantPause	Digital Output	doUnLoadProgram	Digital Output
doEnableFlexpendantStart	Digital Output	doWorldZone	Digital Output

Obr. 29) Vytvorené riadiace signály

Kontroler používa digitálne ako aj analógové signály. Digitálne signály sa využívajú najmä v rámci programového kódu, kde je potrebná komunikácia medzi kontrolerom a Flexpendantom, zatiaľ čo analógové signály slúžia na sledovanie aktuálnej rýchlosti pohybu TCP. Obr. 30 znázorňuje prepojenie časti signálov, ktorých logiku je potrebné oddeliť od logiky v *Main* procedúre programu. Ako je možné vidieť na obrázku, analógový systémový výstup *aoCurrentSpeed*, je vedený do bloku *Expression*, kde sa hodnota tohto signálu násobí číslom 1000 a výsledok tejto hodnoty vchádza do PLC ako analógový vstup *aiCurrentSpeed*. Vďaka tomu sa hodnota systémového výstupu rýchlosti TCP, ktorá predstavuje hodnotu s jednotkou m/s, mení na hodnotu s jednotkou mm/s a výsledná hodnota sa zobrazuje operátorovi na obrazovke Flexpendantu. Ďalej sú na priamo prepojené digitálne výstupy so systémovými vstupmi *doContinue* – *diContinue* a *doPause* – *diPause*.



Obr. 30) Prepojenie vstupných a výstupných signálov radiča PLC

Tieto prepojenia (Obr. 31) sú vytvorené preto, aby sa pri zmene hodnoty digitálneho výstupu z 0 na 1 spustila systémová procedúra nadradená užívateľskému softvéru. Digitálny vstup *diContinue* je v rámci radiča PLC nastavený ako systémový vstup spúšťajúci program. Digitálny vstup *diPause* systémovo zastavuje ukazovateľ programu na konci aktuálnej pohybovej inštrukcie robota, čo zabezpečí, aby TCP robota nezastavil nad substrátom, ale dokončil svoj aktuálne vykonávaný pohyb. Ako posledný využívaný systémový vstup je signál *diPPToMain*, ktorý pri zopnutí premiesti programový ukazovateľ na začiatok hlavnej procedúry *Main*.

Signal Name	Action	Argument 1
diPPToMain	PP to Main	N/A
diPause	Stop at End of Instruction	N/A
diContinue	Start	Continuous

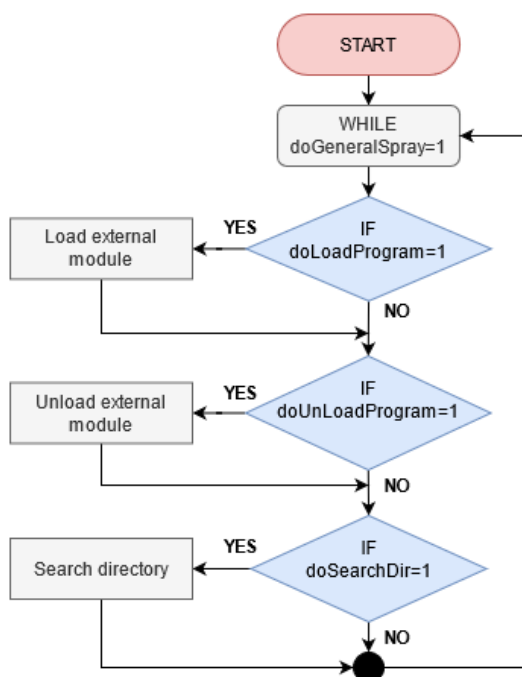
Obr. 31) Systémové vstupy radiča PLC

#### 4.7 Externý modul všeobecného programu

Súčasťou požiadavky zákazníka bolo, aby v používateľskom softvéri bola možnosť nahrania externého programového modulu do aktuálneho programu. Takáto funkcia by užívateľovi umožňovala externe vytvoriť program s presnými pohybovými inštrukciami pre konkrétny objekt a spustiť tento program v rámci tohto univerzálneho programu určený pre studený nástrek nerotačných súčastí. V rámci modulu *Module\_General\_Spray* sú obsiahnuté všetky podprogramy umožňujúce obsluhu úplnú kontrolu nad správou externých programových modulov. Pomocou HMI aplikácie na flexpendante obsluha robota dokáže pár klikmi nahrat' a aj odstrániť potrebný modul do a z programovej štruktúry aktuálnej úlohy. Vďaka vytvorenému podprogramu *Search\_Directory* obsluha priamo vo Flexpendante dokáže načítať všetky zložky a súbory nachádzajúce sa v kontroléri v zložke Home a zvoliť si ten, s ktorým potrebuje pracovať.

Hlavná programová štruktúra je koncipovaná tak, že v prípade zmeny hodnoty digitálneho signálu *doGeneralSpray* na 1 sa presunie programový ukazovateľ („program pointer“) do procedúry *General\_Spray* (Obr.32), kde čaká na ďalší signál. V rámci tejto procedúry sa následne na základe ďalšieho konkrétneho digitálneho signálu spustí príslušný podprogram pre nahratie alebo odstránenie externého modulu alebo prehládanie zložky Home.

V prípade využívania externého modulu, ako hlavnej pohybovej procedúry robota, je možné v rámci užívateľských obrazoviek využívať presne vymedzené funkcie, ktoré sú automaticky aktivované, aby užívateľ presne vedel, čo môže v rámci nastavení meniť. Bližšie inštrukcie na ovládanie všeobecného programu budú vysvetlené v kapitole 5.9 – Návrh užívateľského prostredia a v prílohe 1 – Užívateľský manuál ovládania programu.

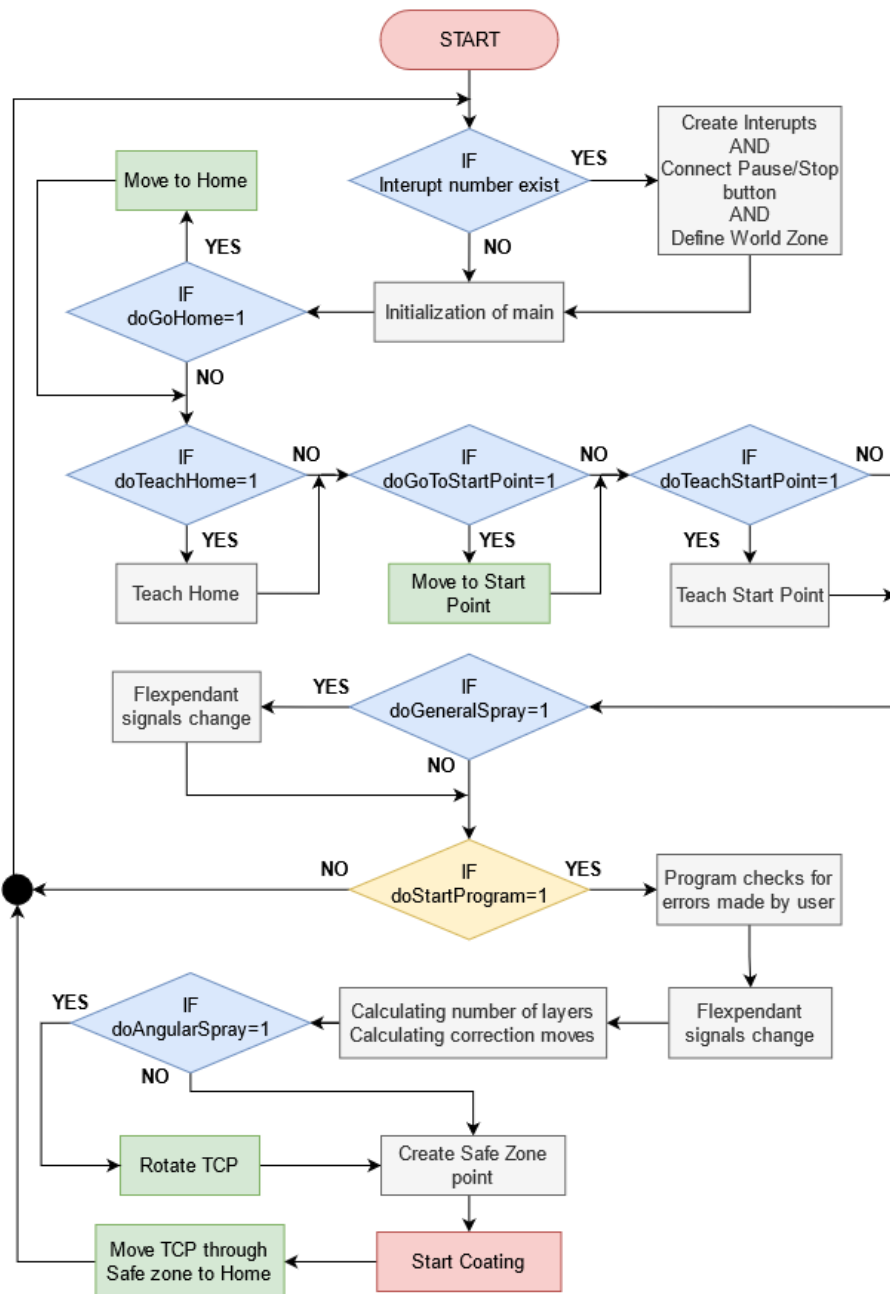


Obr. 32) Vývojový diagram procedúry General Spray

## 4.8 Hlavný modul MAIN

Ako už bolo spomenuté vyššie, hlavný modul Main sa skladá z troch Trap procedúr reagujúcich na vzniknuté prerušenie programu, jednej základnej procedúry Main a deklarovaných dát používaných v tomto module. Základná procedúra Main predstavuje podprogram, ktorý je automaticky systémom spustený v nekonečnej slučke, pričom po vykonaní posledného riadku príkazu sa programový ukazovateľ presúva na začiatok procedúry a cyklus sa opakuje. Prvou funkciou procedúry Main je funkcia IF kontrolujúca, či v rámci programu má každé prerušenie pridelenú variabilnú hodnotu, ktorá je následne spojená s konkrétnou Trap funkciou. Táto kontrola spojenia je zásadná, nakoľko ak by bola vytvorená chyba v spojení, tak by v rámci programu nebolo možné spúšťať Trap procedúry na pozastavenia a zastavenie automatického cyklu. Následne vo fáze inicializácie dochádza k nastavovaniu signálov do požadovaných hodnôt, resetovaniu časového meradla procesu a výpočtu a nastaveniu niektorých vstupných hodnôt premenných. Ďalej nasleduje séria IF funkcií, pri ktorých sa overuje hodnota signálov *doGoHome*, *doTeachHome*, *doToLoad*, a *doStartProgram* prepojených s ovládacími tlačidlami na Flexpendante. Pokiaľ by hodnota akéhokoľvek signálu bola rovná 1, bola by spustená konkrétna procedúra pridelená k danej funkcii IF. V rámci tejto série IF funkcií sa kontrolujú aj hodnoty signálu *doGeneralSpray*, ktorého hodnota je rovná 1, keď je aktívna obrazovka pre správu externých modulov všeobecného programu a signál *doSwitch2*, ktorého nulová hodnota nastaví nulovú hodnotu na spínači tretieho vzoru na

obrazovke 2 Flexpendantu. Ďalšia IF funkcia z tejto série vykonáva kontrolu, či existuje v rámci obrazovky 2 Flexpendantu zvolená možnosť na spustenie všeobecného programu, ktorá bude vysvetlená bližšie v 5.9 – Návrh užívateľského prostredia.



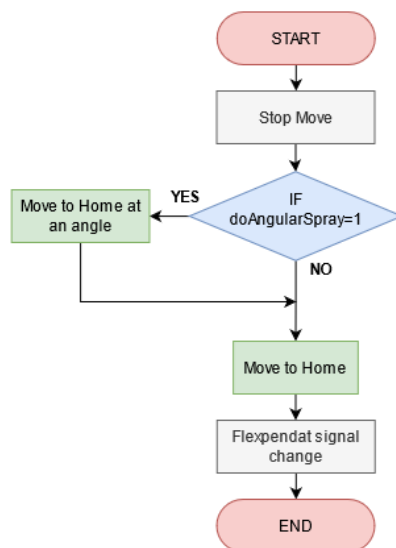
Obr. 33) Vývojový diagram základného programu Main

Signál *doStarProgram* predstavuje najdôležitejší signál spojený s tlačidlom *START*, ktorý spúšťa hlavný programový tok. Ihneď po jeho zopnutí prichádza na rad séria kontrolných IF funkcií, ktoré prekontrolujú zadávané vstupné hodnoty operátorom cez Flexpendant. Tieto kontroly majú za cieľ odhaliť chyby, ktorých sa operátor dopustil pri nastavovaní robota a v prípade odhalenej chyby vypíšu na displej Flexpendatu chybovú hlášku opisujúcu vzniknutý problém a poskytnú stručný návod, ako je možné túto chybu odstrániť. Nasleduje zmena hodnôt signálov medzi PLC a Flexpendantom a sčítavajú sa počty vrstiev, ktoré sa budú



nanášať a generujú hodnoty posuvov, vykonávané pri začatí a ukončení základných pohybov. Pokračuje kontrola požiadavky pre nanášanie mikročastíc pod uhlovým natočením a v prípade, že v rámci nanášania prvých vrstiev bude vyžadované uhlové natočenie, robot začne rotovať okolo nadefinovanej osi. Následne program začne výpočet bezpečnostných bodov, ktoré sa nachádzajú 40 mm od každého rohového bodu nadefinovaného substrátu. Tieto body sú v programe nadefinované ako prejazdové body, ktorými TCP musí prechádzať v prípade pohybov mimo substrát. Chronologicky na to sa začína fáza *Coating*, resp. nanášanie, čo predstavuje celý hlavný programový pohyb robota. Po úspešnom vykonaní všetkých pohybových procedúr sa TCP robota presúva po vopred definovanej trase do bezpečnostnej pozície *Home*.

Spomínané Trap rutiny *Stop\_Interrupt* (Obr. 34), *Pause\_Interrupt* (Obr. 35) a *World\_Zone\_Interrupt* (Obr. 36) sú esenciálnymi podprogramami celého programu. Každá z týchto procedúr je privolávaná po tom, čo systém zaznamená konkrétne prerušenie, ktoré bolo vopred zadefinované v hlavnej procedúre *Main*. V prípade, že nastane prerušenie spojené s procedúrou *Stop\_Interrupt*, programový ukazovateľ sa okamžite, bez ohľadu na to, aký pohyb alebo inštrukciu robot vykonáva, presúva do tejto procedúry a začína vykonávať inštrukcie v nej naprogramované. Ako je znázornené vo vývojovom diagrame procedúry *Stop\_Interrupt* (Obr. 34), po spustení tohto podprogramu nastane okamžité zastavenie pohybu a robot sa presúva najkratšou trajektóriou do bezpečnostnej pozície. Tesne predtým sa vyhodnotí, či je robot v aktuálnej pozícii pod uhlovým natočením a ak áno, robot sa s týmto uhlovým natočením presunie do bezpečnostného bodu, kde sa vyrovná TCP do počiatočného stavu a programový ukazovateľ sa vracia na začiatok procedúry *main()*.



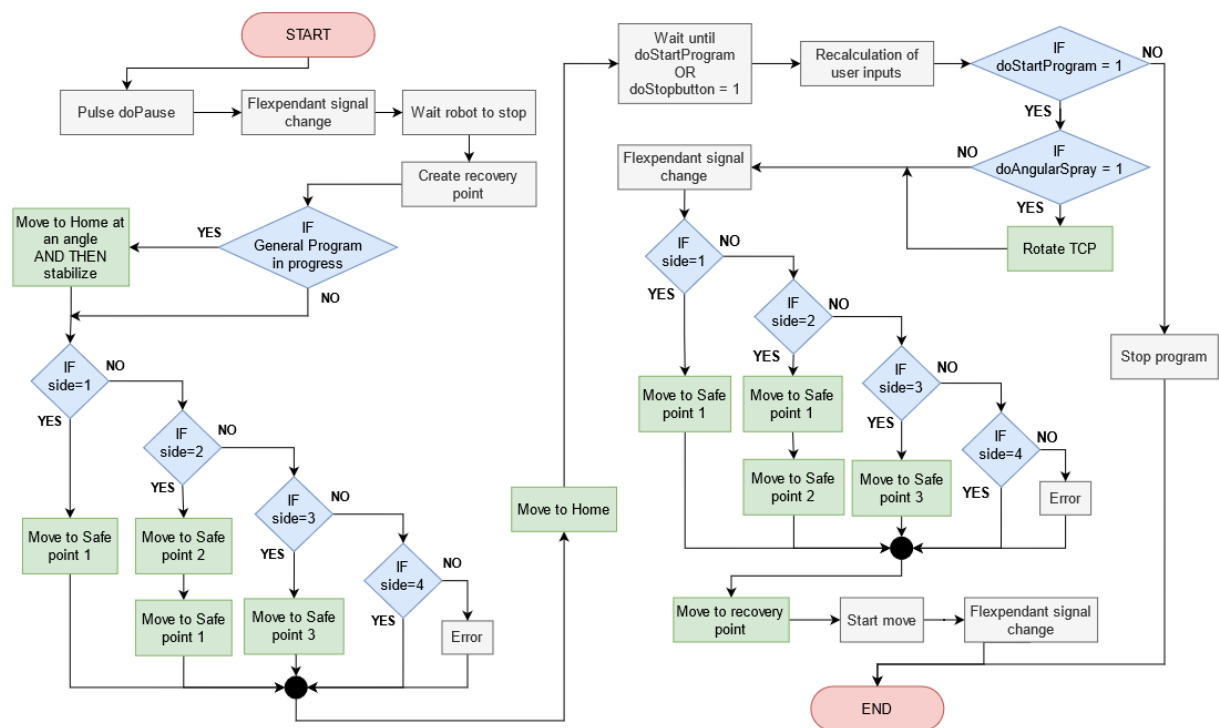
Obr. 34) Vývojový diagram Trap rutiny *Stop*

Trap rutina *Pause\_Interrupt*, ktorej vývojový diagram je zobrazený na obr. 35, je podstatne komplexnejšia. Procedúra je privolaná stlačením tlačidla *PAUSE* na Flexpendante. Prvými vykonávanými inštrukciami sú zmeny hodnôt signálov medzi Flexpendantom a kontrolerom a aktivovanie systémového vstupu, ktorý preruší pohyb robota na konci vykonávanej pohybovej inštrukcie. Po úspešnom zastavení pohybu sa v bode, v ktorom sa robot aktuálne



nachádza, vytvorí takzvaný *Recovery\_point*. Ďalšia inštrukcia skontroluje, či bol prerušený proces vykonávania externého modulu, nahrávaného do programu užívateľom, alebo bol prerušený základný pohybový program.

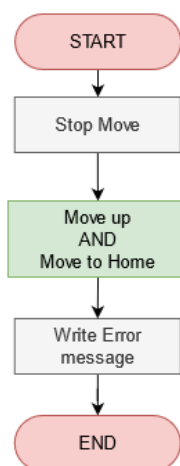
Ak sa prerušil externý modul všeobecného programu, robot sa presúva najkratšou trasou do bezpečnostnej pozície *Home*. V prípade, že bol prerušený základný pohybový program, nasleduje séria ELSEIF funkcií, ktoré podľa aktuálnej pozície zastavenia robota vedú TCP skrz bezpečnostné body až do pozície *Home*. Táto inštrukcia má minimalizovať prebytočné aplikovanie mikročastíc studeného nástreku na substrát tým, že TCP bude vedené v dostatočnej vzdialenosti mimo povrch súčiastky. Po presunutí robota do pozície *Home* sa programový ukazovateľ pozastaví a program čaká kým sa hodnota jedného z dvoch signálov *doStartProgram* alebo *doStopbutton* bude rovnať 1. V prípade, že PLC zaznamená signál *doStopbutton*, program sa ihneď ukončí a programový ukazovateľ sa presunie do procedúry *Main*. Na druhej strane, ak PLC zaznamená signál *doStartProgram*, spúšťajú sa inštrukcie na vrátenie TCP do bodu, kde bol jeho pohyb prerušený. V prvom rade sa kontroluje, či bolo TCP v tomto bode pod uhlovým natočením a ak áno, TCP hneď v bezpečnostnej pozícii začne rotovať, aby dosiahol konkrétnu uhlovú konfiguráciu, odkiaľ sa presúva cez bezpečnostné body okolo substrátu do bodu *Recovery\_point*. Akonáhle sa robot bude nachádzať v pozícii, v ktorej bol jeho pohyb pôvodne prerušený, spúšťa sa jeho pohyb a programový ukazovateľ pokračuje bez zmeny v automatizovanom cykle ďalej.



Obr. 35) Vývojový diagram Trap rutiny Pause

V hlavnom module *Main* sa taktiež nachádza Trap rutina *World\_Zone\_Interrupt*, ktorej vývojový diagram je zobrazený na obr. 36. Táto procedúra predstavuje prerušenie, ktoré sa spúšťa ihneď, akonáhle sa nástroj robota ocitne v nepovolenej oblasti. Povolená oblasť

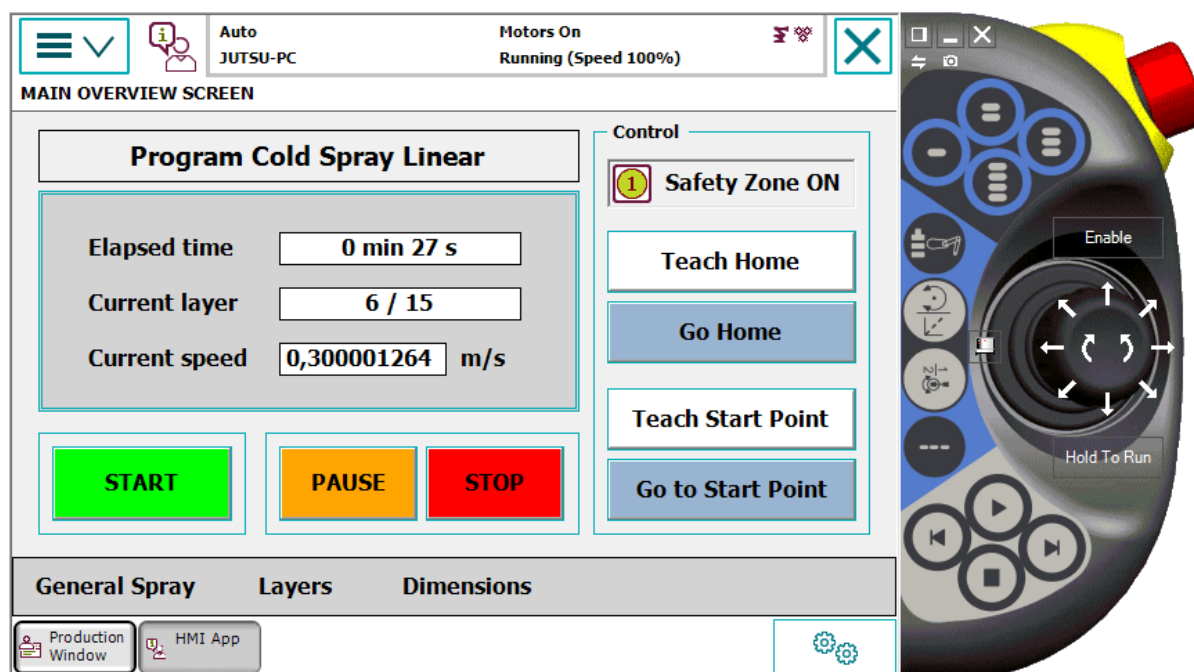
reprezentuje pomyselný kváder s počiatkom v báze robota s veľkosťou 1155 x 1110 x 1250 mm. Tieto hodnoty boli presne zvolené pri reálnom sprevádzkovaní, aby zabránili robotu naraziť do upínacej časti polohovadla v automatickom pohybovom cykle. Aj napriek tomu, že kompletne pohybové inštrukcie programu by nemali dovoliť vznik náhodných nekoordinovaných pohybov robota, použitie tejto bezpečnostnej procedúry zamedzuje vzniku akýchkoľvek kolízií nástroja s polohovadlom alebo okolím robota. V prípade, že sa TCP robota ocitne mimo pomyselný bezpečnostný kváder, prichádza k okamžitému pozastaveniu pohybu robota. Následne robot oddiali koniec nástroja v smere od polohovadla a presunie sa do bezpečnostnej *Home* pozície. Nakoľko aby bol užívateľ informovaný o situácii, ktorá ukončením procesu nastala, zobrazí sa na obrazovke Flexpendantu informatívna správa o vzniknutej skutočnosti, pričom sú mu ponúknuté možnosti riešenia problému. Programový ukazovateľ sa následne presúva do hlavnej programovej procedúry *main()*.



Obr. 36) Vývojový diagram Trap rutiny World\_Zone

## 4.9 Návrh uživatelského prostředí

Program pre pracovisko na studený nástrek je ovládaný pomocou uživatelského rozhrania v rámci obslužnej jednotky Flexpendant. Uživatelské prostredie, resp. HMI aplikácia, bola vytvorená pomocou softvérovej súčasti Robotstudia nazývanej ScreenMaker. HMI aplikácia sa skladá zo štyroch uživatelských obrazoviek, ktoré obsluhu poskytujú všetky potrebné informácie a možnosti nastavenia parametrov na automatizované ovládanie priemyselného robota. Aplikácia je na požiadanie zákazníka kompletne vytvorená v anglickom jazyku, aby spĺňala požiadavku univerzálnosti a mohla byť ovládaná ako českým, tak aj zahraničným operátorom.

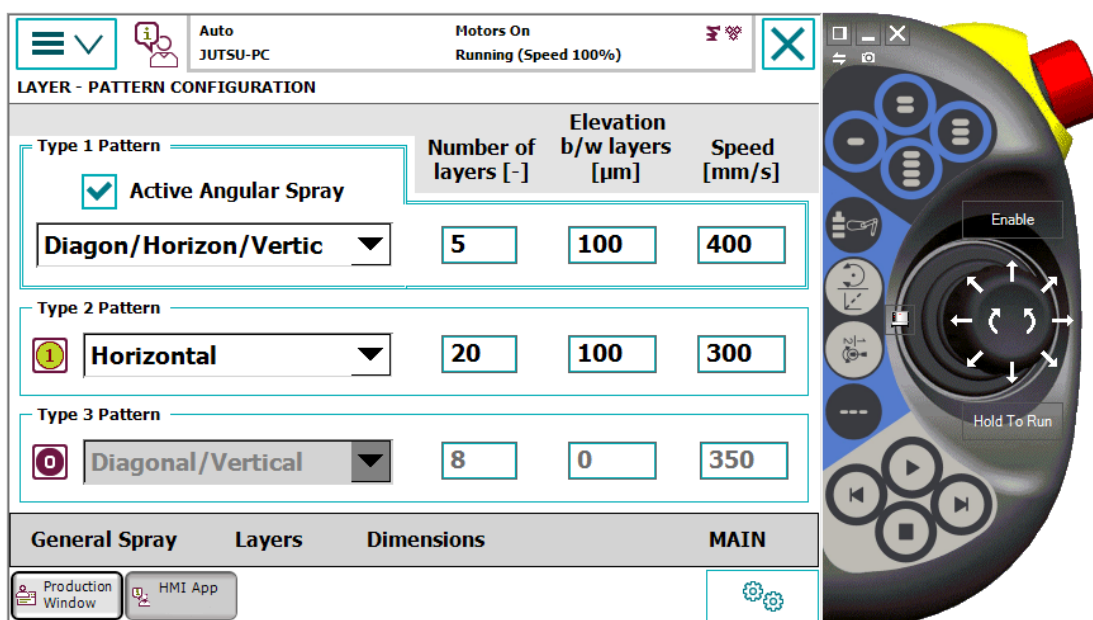


Obr. 37) Hlavná obrazovka HMI aplikácie

Hlavná obrazovka HMI aplikácie (Obr. 37) obsahuje hlavné ovládacie prvky celého programu. V strednej časti obrazovky sú umiestnené informačné okienka informujúce obsluhu o dĺžke trvania aktuálneho pracovného cyklu, aktuálnej vrstve, ktorá je nanášaná z celkového počtu vrstiev a aktuálnej rýchlosti pohybu TCP robota. Aktuálny čas pracovného cyklu sa aktualizuje v pravidelných pohybových intervaloch a je zobrazovaný vo formáte kombinácie minút a sekúnd. Na pravej strane obrazovky, v kolónke *Control*, sa nachádzajú tlačidlá na vybrané pohybové inštrukcie a možnosti práce s programom. Tieto tlačidlá významným spôsobom zjednodušujú prácu operátora s programom a eliminujú akúkoľvek potrebu užívateľa meniť údaje primárnych bodov v rámci komplexnejších nastavení Flexpendantu. Kolónka *Control* obsahuje jeden spínač a štyri tlačidlá. Spínač *Safety Zone* slúži pre aktiváciu/deaktiváciu bezpečnostnej funkcie na zastavenie pohybu robota v prípade, že by malo prísť ku kolízií s polohovadlom, alebo by nástrek mohol nechcene pôsobiť na rameno polohovadla. Dve biele tlačidlá slúžia na spustenie procedúry na zapamätanie aktuálnej polohy robota a dve modré tlačidlá sa používajú na presunutie TCP robota do pozície, ktorú má aktuálne zapísanú v programe. Užívateľ tak manuálnym navigovaním robota príde do požadovanej pozície

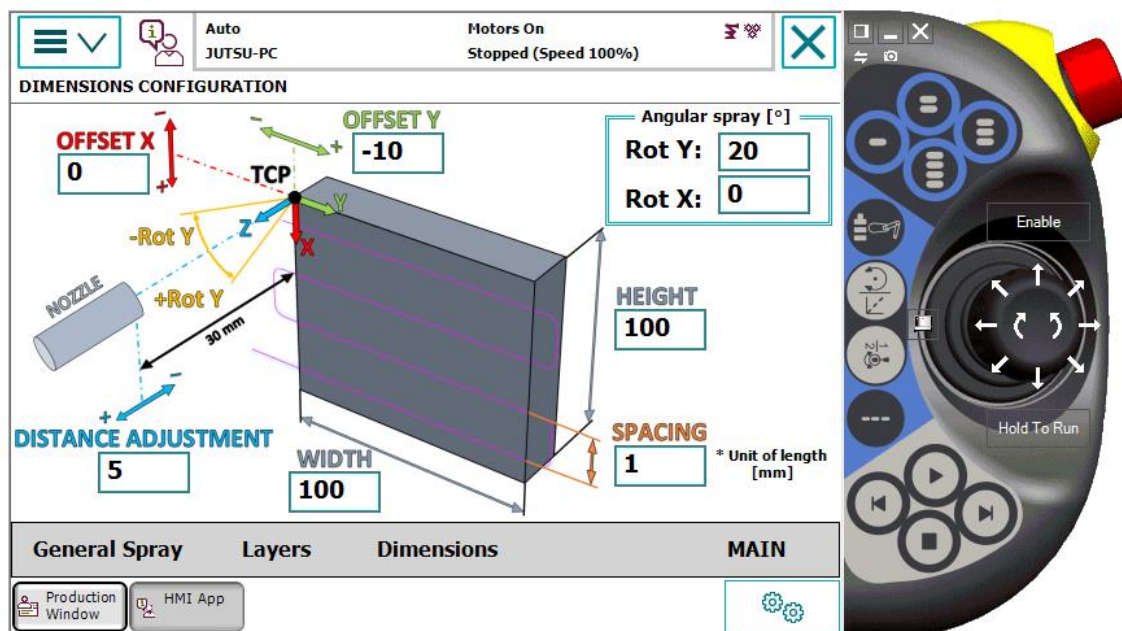
a pomocou jednoduchého stlačenia tlačidla *Teach* zaznamená aktuálnu pozíciu, ako bezpečnostnú pozíciu *Home* alebo počiatočný bod *Start Point*. Následne užívateľ môže preveriť správnosť zapísaných bodov tým, že stlačením modrého tlačidla *Go Home* alebo *Go to Start Point* sa premiestni do týchto bodov. V spodnej časti obrazovky je umiestnené menu, pomocou ktorého obsluha vie preklikávať medzi jednotlivými obrazovkami aplikácie, aby správne nastavila vstupné parametre.

Neoddeliteľnou súčasťou základnej obrazovky sú tlačidlá na spúšťanie a zastavovanie automatizovaného cyklu programu. Tieto tlačidlá sú viditeľne umiestnené a farebne odlišené od ostatných tlačidiel. Tlačidlo *START* je prepojené so signálom, ktorý v rámci procedúry *main()* spúšťa hlavný automatizovaný pohybový cyklus priemyselného robota. Po stlačení tohto tlačidla sa však ako prvé v programe vykonajú kontrolné funkcie, ktorých cieľom je prekontrolovať vstupné parametre zadané obsluhou. V prípade, že by program odhalil nejakú chybu, alebo hodnotu nekompatibilnú s ostatnými parametrami, systém vypíše chybu na obrazovku Flexpendantu s inštrukciami na jej odstránenie a vráti ukazovateľ programu na začiatok procedúry *main()*. Ďalším tlačidlom je tlačidlo *STOP*. Po jeho stlačení dochádza k prerušeniu programu, spusteniu TRAP rutiny *Stop Interrupt*, k urýchlenému zastaveniu všetkých pohybov robota a následne k presunutiu TCP najkratšou trasou do bezpečnostnej pozície *HOME*. Posledným spomenutým je tlačidlo *PAUSE*. Ako už názov napovedá, slúži na dočasné pozastavenie pohybu robota. Pri stlačení tohto tlačidla sa pomocou systémového vstupu zastaví pohyb na konci vykonávanej inštrukcie a v programe nastane prerušenie, čo presmeruje programový ukazovateľ do rutiny TRAP *Pause Interrupt*. Pokiaľ je program pozastavený, na hlavnej obrazovke dôjde ku zmenám tlačidla *START* na tlačidlo *CONTINUE* a tlačidlo *PAUSE* sa stane neaktívnym. Táto zmena zamerne limituje možnosti obsluhy a ponúka jej dva možné varianty na pokračovanie. Prvým je stlačenie zeleného tlačidla *CONTINUE*, pri ktorom sa TCP robota presunie na miesto kde bol program prerušený a robot bude pokračovať ďalej v programovom cykle. Druhým variantom je stlačenie tlačidla *STOP*, čo ukončí pokračovanie programu a vráti programový ukazovateľ do rutiny *main()*.



Obr. 38) Obrázok 2 – Layers pre nastavenie vrstiev HMI aplikácie

Druhá užívateľská obrazovka s názvom *Layer – Pattern Configuration* (Obr. 39) ponúka užívateľovi prehľadne roztriedené možnosti nastavenia pohybových inštrukcií, ktoré má priemyselný robot vykonávať počas aplikovania materiálu studeným nástrekom. Obsluha si môže zvoliť tri rôzne vzory pohybov, ktoré sa budú vykonávať postupne za sebou. V každom z týchto troch vzorov sa nachádzajú rolovacie okná, kde sa nastaví druh pohybu TCP (horizontálny, vertikálny, diagonálny), resp. vzájomná kombinácia dvoch (horizontálne – vertikálny, diagonálne – horizontálny, diagonálne – vertikálny) alebo kombinácia troch pohybov (horizontálne – vertikálne – diagonálny). Pre každý vzor zvlášť si obsluha vie nastaviť separátne počet nanášaných vrstiev, posunutie v smere osi Z medzi každou vrstvou a rýchlosť pohybu TCP pre každú skupinu vzorov. Na tejto obrazovke sa taktiež volí, či má byť aktívne nanášanie na substrát pod uhlom. Táto možnosť je povolená iba pre prvý vzor nanášania. V rámci obrazovky sa ešte vľavo nachádzajú dva spínače, slúžiace na aktiváciu a deaktiváciu druhého a tretieho vzoru nanášania. V prípade, že by užívateľ chcel spustiť program iba s dvoma vzormi nanášania, vypne tretí spínač pri treťom vzore, čo deaktivuje tento vzor a program bude pracovať iba s parametrami, ktoré boli zvolené v prvom a druhom vzore nanášania. V prípade, že by účelom bolo nanášať iba jednu vrstvu ale na obrazovke svietia kontrolky spínačov pre vzor 2 a vzor 3, operátor môže jedným stlačením spínaču vzoru 2 vypnúť oba spínače naraz.

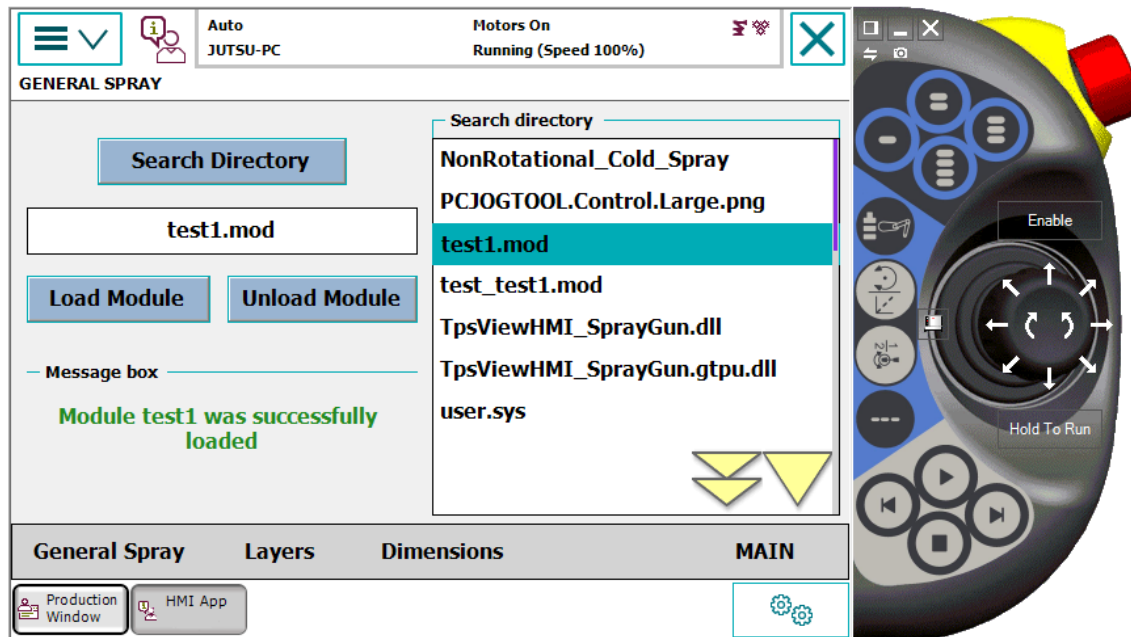


Obr. 39) Obrázok 3 – Dimensions pre nastavenie hodnôt striekaného objektu a orientácie TCP aplikácie HMI

Tretia užívateľská obrazovka s názvom *Dimensions Configuration* (Obr. 40) je tvorená z jedného hlavného obrázka s popismi nastaviteľných parametrov, do ktorého sú pridané kontextové okná s možnosťou vpisovania numerických hodnôt. Obrázok zobrazuje univerzálny tvar substrátu, na ktorý má byť nanášaný povlak. Sú tu prehľadne označené orientácie jednotlivých osí, veľkosti objektu a nastavenie ofsetov v osiach. Operátor si tak dokáže rýchlo



asociovať zadávané hodnoty parametrov ku konkrétnym označeniam. V pravom hornom rohu sa nachádza kolónka pre zadanie presného uhla natočenia TCP okolo osi X a Y. Aj keď je v obrázku ukážka orientácie natočenia iba pre os Y, natočenie okolo osi X je smerované podľa pravidla pravej ruky, rovnako ako je tomu pri osi Y.



Obr. 40) Obrázok 4 – General Spray pre nahranie a odstránenie externých modulov všeobecného programu aplikácie HMI

Poslednou obrazovkou je obrazovka s názvom *General Spray*, ktorej účelom je poskytnúť operátorovi nástroje pre efektívnu správu externých modulov programu. Hneď prvé tlačidlo *Search Directory* nesmierne urýchľuje proces vyhľadávania externých modulov. Pri stlačení tohto tlačidla sa v programe spúšťa vytvorená procedúra na prehľadanie základnej zložky kontrolera, nazývanej *Home*. Táto procedúra vypíše na pravej strane obrazovky všetky nájdené zložky a súbory, ktoré sa v *Home* nachádzajú. Operátor si následne môže priamo zvoliť konkrétny modulový súbor s koncovkou „.mod“ a po stlačení tlačidla *Load Module* sa tento modul nahrá do programovej štruktúry spusteného programu. Podobne tlačidlo *Unload Module* zvolený modul odstráni z programovej štruktúry. Pre lepšiu prehľadnosť, pokiaľ operátor klikne na nejaký zo súborov, názov tohto súboru sa napíše do kontextového okna pod tlačidlo *Search Directory*, aby bolo jasné, s akým súborom sa bude ďalej pracovať. Súčasťou tejto obrazovky je aj takzvaný *Message box*, v ktorom sú generované upozornenia na informovanie operátora o stave nahrávania a odstraňovania externých modulov. Program dokáže vypísať upozornenia pri úspešnom a neúspešnom nahratí modulu do programu, pokuse o nahranie modulu, ktorý sa už v programe nachádza, pokuse o nahranie priečinku namiesto súboru, alebo aj pokus o nahranie súboru, ktorý nemá koncovku „.mod“.

Užívateľské rozhranie aplikácie HMI je, podľa možnosti, vytvárané tak, aby bolo pre užívateľa čo najjednoduchšie ovládateľné ale zároveň, aby poskytovalo maximálny výber možných nastavení. Aplikácia je koncipovaná tak, aby eliminovala chybu vytvorenú na strane operátora a preto jednou z funkcií je automatické zablokovanie všetkých kontextových okien

pre zadávanie vstupných parametrov a spínačov ihneď po spustení automatického pracovného cyklu tlačidlom *START*. Toto blokovanie zamedzí, aby sa počas prebiehajúceho programu zmenila akákoľvek vstupná hodnota nechceným zavinením operátora. Operátor však bude môcť počas spusteného automatického cyklu preklikávať medzi jednotlivými obrazovkami, ak bude potrebovať skontrolovať správnosť zadaných vstupných parametrov. V prípade, že operátor po spustení automatického pracovného cyklu robota zistí, že jedna z jeho zadaných vstupných hodnôt nevyhovuje jeho požiadavkám, má dve možnosti, ako postupovať:

1. Stlačením tlačidla *STOP* zastaví celý proces a počká, kým sa robot vráti do bezpečnostnej pozície. Následne sa mu aktivujú na Flexpendante tlačidlá a bude môcť zmeniť vstupnú hodnotu. Automatický cyklus bude možné však začať jedine od začiatku.
2. Stlačením tlačidla *PAUSE* pozastaví proces a počká, kým sa robot vráti do bezpečnostnej pozície. Následne sa mu aktivuje vybraná časť tlačidiel na Flexpendante, ktorými bude môcť meniť nastavenia na obrazovke 2 – Layers a obrazovke 3 - Dimensions. Výhodou tohto postupu je možnosť pokračovania automatického cyklu robota z miesta, kde sa pohyb prerušil.

Ďalšími bezpečnostnými prvkami eliminujúcimi chybu užívateľa je celkové obmedzenie minimálnych a maximálnych hodnôt vstupných parametrov. Každé kontextové okno, určené na vloženie hodnoty zadanej užívateľom, je limitované v kladnom aj zápornom smere do určitej hodnoty, čím sa opäť znižuje riziko zadania chybných hodnôt obsluhou. V neposlednom rade jedným z ďalších bezpečnostných nastavení je aplikovanie pulzového ovládania všetkých signálov napojených na tlačidlá Flexpendantu. Takáto konfigurácia poskytuje pri stlačení akéhokoľvek tlačidla časovo presne definovanú dĺžku vysielaného signálu, čo limituje v rámci programovej štruktúry vznik náhodných signálových konfliktov.

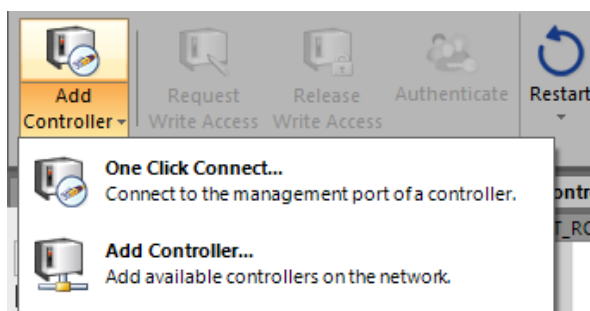
Ďalšie nezmiernené vlastnosti užívateľského prostredia HMI ako aj kompletný návod na obsluhu programu je uvedený v prílohe 1 – Užívateľský manuál ovládania programu.





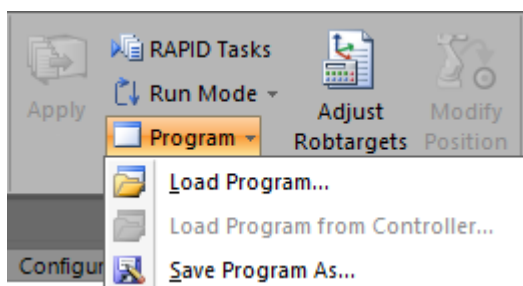
## 5 REÁLNE SPREVÁDZKOVANIE

Záverečnou úlohou celého diplomového projektu bolo nahrať vytvoreného programu spolu s užívateľským prostredím a všetkými potrebnými komponentami do skutočného priemyselného ovládača robota, testovať funkcie vytvoreného programu a uviesť do prevádzky robotické pracovisko na studený nástrek. Uvedenie do prevádzky sa vykonávalo prepojením osobného laptopu s priemyselným ovládačom IRC 5 pomocou ethernetového káblu. V rámci programu Robotstudio sa v nastaveniach ovládača toto spojenie vytvorilo pomocou možnosti „One Click Connect“ z obrázka 41.



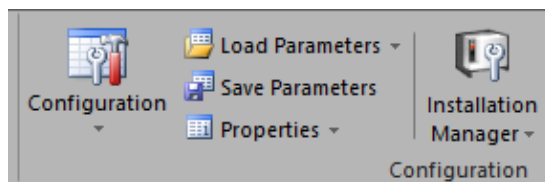
Obr. 41) Vytvorenie spojenia medzi Robotstudiom a priemyselným ovládačom IRC 5

Následne mohlo prísť na rad kopírovanie programu, vytvorených signálov a všetkých potrebných komponentov do skutočného ovládača. Kopírovanie programu prebieha vytvorením kópie všetkých modulov virtuálneho ovládača v záložke RAPID pomocou tlačidla „Save Program As...“ a nahrať týchto modulov do skutočného ovládača IRC 5 tlačidlom „Load Program...“ zobrazených na obrázku 42.

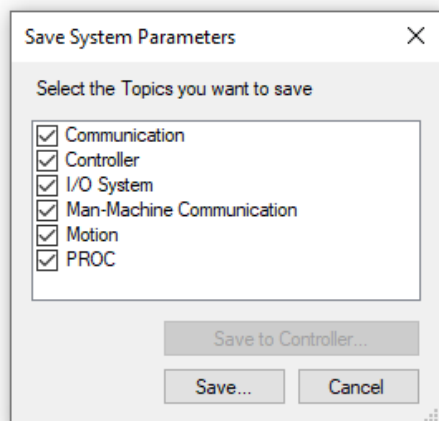


Obr. 42) Uloženie a nahrať programu

Ďalej sa v nastaveniach virtuálneho ovládača (Obr. 43) zvolí možnosť „Save Parameters“, čo uloží systémové parametre virtuálneho programu a vytvorí 6 konfiguračných súborov (Obr. 44). Tieto súbory sa následne nahrávajú do reálneho ovládača na pracovisku pomocou tlačidla „Load Parameters“.



Obr. 43) Možnosti na správu systémových parametrov



Obr. 44) Kontextové okno pre uloženie systémových parametrov ovládača

Pokiaľ sú všetky systémové a programové moduly nahrané, prekopírujú sa ešte dva .dll súbory vytvorenej užívateľskej HMI aplikácie cez „File Transfer“ do priečinku HOME reálneho ovládača. Ovládač IRC 5 týmto obsahuje všetky potrebné súbory a prichádza na rad kontrola správnosti prepísaných súborov a testovanie kompatibility softvéru s robotom.

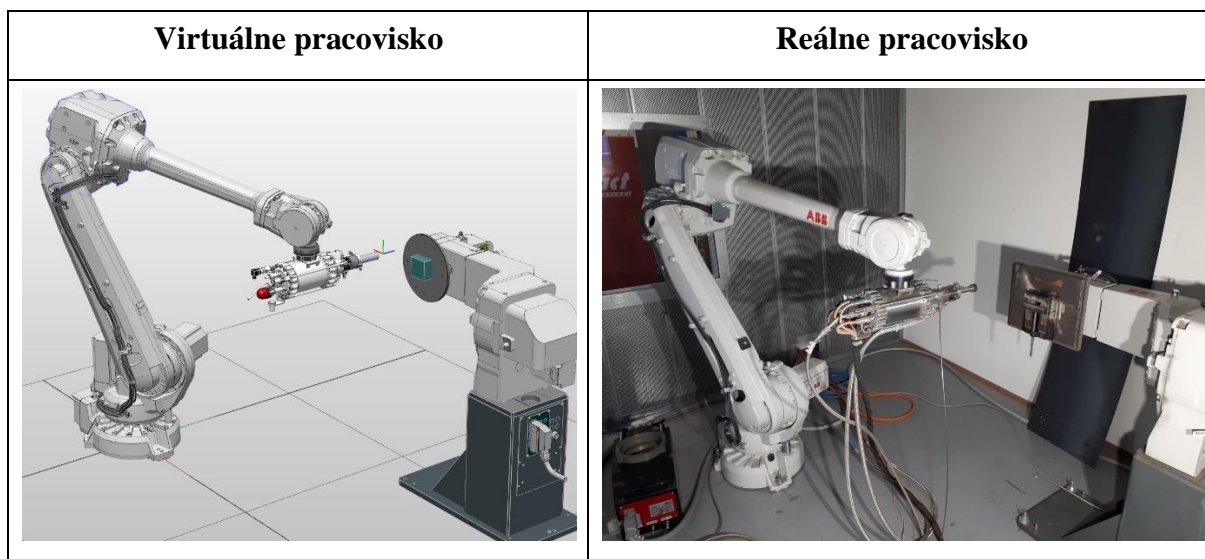
Prvotným testom bola kontrola kompatibility súradnicových systémov. V nastavení Flexpendantu „Jogging“ sa menili súradnicové systémy a manuálnym ovládaním robota sa jeden po druhom prekontrolovalo, či orientácie jednotlivých osí pracoviska sú totožné s orientáciou osí virtuálneho modelu, v ktorom bol program vytváraný. Vďaka kontrole sa zistila chyba nesprávne orientovaného TCP v rámci programu. Táto chyba bola spôsobená nesprávne nakonfigurovanou orientáciou nástroja vo virtuálnom prostredí, ktoré mi bolo poskytnuté v rámci prvotného virtuálneho modelu pracoviska. Chyba bola následne odstránená a test kompatibility súradnicových systémov prebehol v poriadku.

Ďalším testom bola kontrola funkčnosti užívateľského HMI prostredia. Na reálnom Flexpendante sa testovalo, či vytvorená HMI aplikácia reaguje správne na vstupy operátora, či fungujú všetky tlačidlá a signály na všetkých obrazovkách a či sa zadávané parametre zapisujú do pamäte ovládača. Kontrola bola úspešná.

Pretože vytvorený program využíva nastavenie „World Zone“, ktorého funkcia je znemožnenie pohybu robota mimo povolený priestor, súčasťou reálneho sprevádzkovania bolo definovať tento priestor. Priestor bol definovaný bodom v počiatku bázy robota a presne zvoleného bodu, ktorý sa nachádzal 10 mm od zveráka upevneného na upínacej doske polohovadla a pravého okraja upínacej dosky. Tieto hodnoty sa následne vložili do zdrojového kódu programu.

Nasledovalo testovanie funkčnosti a správnosti pohybov robota. V rámci HMI sa volili rôzne vstupné inštrukcie a náhodné vstupné parametre, ktoré mal program spracovať a vykonať generovaný pohyb. Všetky vykonávané pohyby robota zodpovedali pohybu, ktoré mali byť vykonané. Testovanie pokračovalo skúškou funkčnosti nahrávania a spúšťania externých modulov. Do programu sa pomocou HMI nahral vytvorený skúšobný modul, reprezentujúci externý pohybový modul špecifického nástreku, ktorý bol následne robotom vykonaný. Test prebehol úspešne a kontrola správy a spúšťania externých modulov v rámci užívateľského programu bola úspešná.

Záverečným testom reálneho sprevádzkovania pracoviska na studený nástrek bolo odskúšanie automatizovaného ovládania pracoviska počas reálne vykonávaného studeného nástreku. Testovanie sa uskutočnilo za účasti kolegov z Ústavu materiálového inžinierstva, ktorí toto naprogramované pracovisko využili na aplikovanie medeného práškového nástreku na oceľové platničky, na ktorých v rámci výskumu bude skúmaná materiálová štruktúra nanesej medi. Celkovo sa vykonávalo päť nástrekov na piatich oceľových platničkách. Každý nástrek pozostával z horizontálnej pohybovej inštrukcie robota. Na prvú oceľovú platničku sa naniesla jedna vrstva, na každú ďalšiu platničku sa počet nanášaných vrstiev zvyšoval vždy o jednu navyše. Platničky boli striekané separátne, pričom výmenu a uchytenie platničiek vykonávala obsluha pracoviska. Studený nástrek medi prebehol úspešne. Všetky automatizované pohybové cykly priemyselného robota boli vykonané presne podľa zadaných vstupných parametrov, čo preukázalo správnosť vytvoreného užívateľského softvéru a užívateľského rozhrania priamo v priemyselnej praxi.



Obr. 45) Virtuálne a reálne pracovisko



## 6 ZHODNOTENIE A DISKUSIA

Virtuálne sprevádzkovanie predstavuje komplexný ale aj nesmierne efektívny spôsob uvedenia robotického pracoviska do prevádzky v skrátenej dobe a s možnosťou odstránenia chýb, ešte pred zahájením skutočného výrobného procesu.

Pre realizáciu zámeru diplomovej práce – návrhu a realizácie virtuálneho uvedenia do prevádzky robotizovaného pracoviska na studený nástrek, bolo najdôležitejšou časťou vytvorenie softvéru pre automatizované pohybové inštrukcie priemyselného robota. Benefitom navrhnutého a v praxi úspešne otestovaného užívateľského programu je ponuka širokého výberu nastavení, umožňujúcich obsluhu pracoviska bezpečne a efektívne konfigurovať univerzálne pohybové inštrukcie robota a zároveň program poskytuje možnosť nahratia a vykonávania externých pohybových modulov pre obrobky špecifických tvarov. Užívateľský program obsahuje všetky zákaznikom požadované ovládacie prvky, ale súčasne disponuje širokými možnosťami ďalšieho rozširovania jeho funkcií a v neposlednom rade aj možnosťou vyriešenia identifikovaných drobných nepresností spôsobených vývojárskym prostredím Robotstudio a Screenmaker.

Súčasný stav užívateľského programu ponúka jednotnú bezpečnostnú funkciu „World zones“ pre všetky typy pohybov. Túto funkciu by bolo možné napríklad rozšíriť o možnosť adaptabilného prispôsobovania veľkosti bezpečnostných zón generovaných v závislosti na veľkosti striekaného obrobku a pozícií začiatočného bodu *Start point*.

Priestorom pre inováciu užívateľského programu môže byť pridanie možnosti úpravy, resp. prispôsobenia prejazdov TCP, ktoré robot vykonáva medzi každou nanášanou vrstvou. Momentálne je veľkosť prejazdu fixná s hodnotou 40 mm od konca hrany substrátu. Možnosť úpravy pohybu robota pri týchto prejazdoch medzi vrstvami by mohla prispieť k zníženiu celkového času procesu nanášania.

V priebehu tvorby užívateľského programu a užívateľského prostredia sa vyskytla nezrovnalosť v zaznamenávaní času a zobrazovaním času na obrazovku Flexpendantu v rámci virtuálnej simulácie. Vzniknutá chyba spôsobovala rozdiel medzi skutočným časom trvania pohybového cyklu meraným stopkami a časom zobrazovaným na virtuálnej obrazovke Flexpendantu. Túto nepresnosť sa našťastie nepodarilo preukázať pri reálnom uvedení robota do prevádzky ale mala by byť do budúcnosti viac preverená a zdokumentovaná.

V neposlednom rade je potrebné upozorniť na chybu spôsobovanú grafickým zobrazovaním užívateľských obrazoviek na Flexpendante pri zadávaní hodnôt s desatinnou čiarkou. V prípade že obsluha pracoviska zadá akúkoľvek hodnotu s desatinnou čiarkou (napr. riadkovanie = 12.5 mm), program túto hodnotu zapíše do svojej pamäti, avšak Flexpendant tento parameter, z neznámeho dôvodu, zobrazí rovný nule. Táto chyba nemá vplyv na presnosť pohybu robota, nakoľko hodnota s desatinnou čiarkou, ktorú obsluha zadala, sa skutočne zapíše do programu a program s ňou ďalej pracuje, avšak táto chyba môže negatívne ovplyvniť interakciu človeka so strojom pri nastavovaní vstupných parametrov programu.

Priemyselný ovládač robotického pracoviska pre studený nástrek je navyše vybavený opciou „Multitasking“, ktorá umožňuje vykonávanie dvoch a viacerých programových úloh

naraz. Túto opciu by bolo možné využiť napríklad pri zjednodušení ovládania programu pomocou systémových vstupov „PP To Main“ alebo iných. Taktiež by bolo možné navrhnúť úpravu zobrazovania aktuálnej rýchlosti na hlavnej obrazovke v jednotke mm/s namiesto zobrazovanej jednotky m/s.

Vytvorený program diplomovej práce ponúka univerzálne riešenie ovládania robotického pracoviska pre studený nástrek nachádzajúci sa v technologickej hale ústavu materiálového inžinierstva Vysokého učení technického v Brne. Cieľom bolo dosiahnuť v maximálnej možnej miere spokojnosť zákazníka s návrhom programového riešenia a jeho implementovaním tak, aby program mohol byť využívaný v priemyselnej praxi. Splnenie cieľa bolo dosiahnuté, užívateľský program zohľadnil a rozvinul všetky požiadavky vymedzené zákazníkom. Program a jeho užívateľské prostredie boli testované v praxi priamo na ústave materiálového inžinierstva a sú tak k dispozícii pre využívanie pri nanášaní studeného nástreku pre ďalšiu výskumnú činnosť.

## 7 ZÁVER

Cieľom diplomovej práce bolo virtuálne sprevádzkovať robotické pracovisko pre studený nástrek pomocou softvéru Robotstudio a Screenmaker. Práca obsahuje vypracovanie všetkých vytyčených bodov stanovených v zadaní, ktoré viedli k finalizácii digitálneho sprevádzkovania pracoviska nachádzajúceho sa v technologickej hale ústavu materiálového inžinierstva univerzity VUT v Brne.

Prvá časť práce je zameraná na oboznámenie sa s problematikou, v rámci ktorej je vytvorený všeobecný prehľad súčasného stavu poznania v oblasti virtuálneho sprevádzkovania. Vysvetlená je teória tvorby HMI, písanie zdrojového kódu programu, predstavenie programovacieho jazyku RAPID, ktorý bol použitý pre programovanie a vysvetlenie technológie studeného nástreku, pre ktorú je pracovisko vytvorené. Ďalšia časť všeobecného prehľadu je analýza robotického pracoviska, v rámci ktorej sú predstavené všetky hlavné súčasti a komponenty systému používané na pracovisku.

Hlavnú časť práce tvorí kapitola samotného virtuálneho uvedenia pracoviska do prevádzky. V tejto kapitole sú rozoberané vstupné požiadavky zákazníka, čo by mal program obsahovať, nasledované sériou podkapitol venovaných vysvetleniu najdôležitejších súčastí vytvoreného programu. Je tu vysvetlená hlavná štruktúra programu, z čoho sa skladá a aké rôzne podprogramy sú v rámci automatického režimu robota vykonávané. Súčasťou je taktiež podkapitola prezentujúca vytvorené užívateľské prostredie HMI, ktoré poskytuje obsluhu efektívnu kontrolu nad robotom.

Záverečná časť pojednáva o reálnom uvedení priemyselného robota na studený nástrek do prevádzky. Vysvetľuje sa v nej postup nahrávania vytvoreného užívateľského softvéru a užívateľského prostredia HMI do ovládača robota. Nahraný program je následne podrobený viacerým testom, ktoré mali za cieľ odhaliť prípadné nepresnosti, aby následne mohol byť program doladený.

Primárnym cieľom užívateľského programu bolo vytvoriť možnosť ovládania robota s veľkým dôrazom na bezpečnosť a pri použití jednoduchých ovládacích prvkov, ktorých finálna forma rozšírila možnosti ovládania nad požiadavky zákazníka. Program tak predstavuje efektívny a univerzálny nástroj na využitie technológie studeného nástreku, ktorého riadenie sa dokáže obsluha veľmi rýchlo naučiť aj za pomoci priloženého užívateľského manuálu.





## 8 ZOZNAM POUŽITÝCH ZDROJOV

- [1] AUINGER, F., M. VORDERWINKLER a G. BUCHTELA. Interface driven domain-independent modeling architecture for "soft-commissioning" and "reality in the loop". *WSC'99. 1999 Winter Simulation Conference Proceedings. 'Simulation - A Bridge to the Future' (Cat. No.99CH37038)*. IEEE, 1999, , 798-805. ISBN 0-7803-5780-9. Dostupné z: doi:10.1109/WSC.1999.823265
- [2] LIU, Zheng, Nico SUCHOLD a Christian DIEDRICH. Virtual Commissioning of Automated Systems. *Automation* [online]. InTech, 2012 [cit. 2021-05-10]. ISBN 978-953-51-0685-2. Dostupné z: doi:10.5772/45730
- [3] ZÄH, Michael, Georg WÜNSCH, Thomas HENSEL a Alexander LINDWORSKY. Nutzen der virtuellen Inbetriebnahme: Ein Experiment. *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb*. 2006, 101(10), 595-599. ISSN 0947-0085. Dostupné z: doi:10.3139/104.101070
- [4] TURNBULL, Charlotte, Graham JACKSON, ed. What is Virtual Commissioning. *Virtual Commissioning.com* [online]. Maplesoft, 2019 [cit. 2021-05-10]. Dostupné z: <https://virtualcommissioning.com/what-is-virtual-commissioning/>
- [5] JACKSON, Chad. Hybrid Digital Twin for Virtual Commissioning. In: *Virtual Commissioning.com* [online]. Maplesoft, 2020 [cit. 2021-05-10]. Dostupné z: <https://virtualcommissioning.com/hybrid-digital-twin-for-virtual-commissioning/>
- [6] LEE, Chi a Sang PARK. Survey on the virtual commissioning of manufacturing systems. *Journal of Computational Design and Engineering* [online]. 2014, 1(3), 213-222 [cit. 2021-05-10]. ISSN 2288-5048. Dostupné z: doi:10.7315/JCDE.2014.021
- [7] LECHLER, Tobias, Eva FISCHER, Maximilian METZNER, Andreas MAYR a Jörg FRANKE. Virtual Commissioning – Scientific review and exploratory use cases in advanced production systems. *Procedia CIRP*. 2019, 81, 1125-1130. ISSN 22128271. Dostupné z: doi:10.1016/j.procir.2019.03.278
- [8] VERMAAK, Herman a Johan NIEMANN. Virtual commissioning: A tool to ensure effective system integration. *2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM)*. IEEE, 2017, , 1-6. ISBN 978-1-5090-5582-1. Dostupné z: doi:10.1109/ECMSM.2017.7945899
- [9] PAPYRIN, Anatolii, Vladimir KOSAREV, Sergey KLINKOV, Anatolii ALKHIMOV a Vasily FOMIN. *Cold Spray Technology*. Oxford: Elsevier, 2007. Prvá edícia. ISBN 978-0-08-045155-8.
- [10] Cold Spraying - Cold Spray Coating Deposition. *twi-global.com* [online]. Cambridge, UK: TWI Ltd, 2021 [cit. 2021-05-10]. Dostupné z: <https://www.twi-global.com/what-we-do/research-and-technology/technologies/coating-and-surface-engineering/cold-spraying>

- [11] GÄRTNER, Frank, Thorsten STOLTENHOFF, Tobias SCHMIDT a Heinrich KREYE. The Cold Spray *Process and Its Potential for Industrial Applications*. Journal of Thermal Spray Technology [online]. 2006, **15**(2), 223-232 [cit. 2021-05-10]. ISSN 10599630. Dostupné z: doi:10.1361/105996306X108110
- [12] Cold Spray Technology [online]. Rattenkirchen, Nemecko: Impact Innovations GmbH, 2020 [cit. 2021-05-10]. Dostupné z: <https://impact-innovations.com/en/technology/>
- [13] LUNDQVIST, Rasmus a Tobias SÖRELING. New Interface for Rapid Feedback Control on ABB-robots. In: Diva-Portal [online]. Linköping: Linköping Universit t, 2005 [cit. 2021-05-10]. Dostupn  z: <http://www.diva-portal.org/smash/get/diva2:20104/FULLTEXT01.pdf>
- [14] Technical reference manual: RAPID overview [online]. In: . V ster s,  v dsko: ABB AB, Robotics, 2018 [cit. 2021-05-10]. Dostupn  z: <https://abb.sluzba.cz/Pages/Public/IRC5UserDocumentationRW6/en/3HAC050947%20TRM%20RAPID%20Overview%20RW%206-en.pdf>
- [15] GOLDBERG, Zachary. The Six Commandments of Good Code: Write Code That Stands *the Test of Time*. Toptal.com [online]. Toptal, LLC, 2016 [cit. 2021-05-10]. Dostupn  z: <https://www.toptal.com/software/six-commandments-of-good-code>
- [16] Clean code: what makes programming code clean. Ionos.com [online]. 1&I IONOS Inc., 2020 [cit. 2021-05-10]. Dostupn  z: <https://www.ionos.com/digitalguide/websites/web-development/clean-code-principles-advantages-and-examples/>
- [17] REDDY, Bharadwaj. What is Human Machine Interface (HMI). Instrumentationtools.com [online]. India, 2019 [cit. 2021-05-10]. Dostupn  z: <https://instrumentationtools.com/what-is-a-human-machine-interface-hmi/>
- [18] What is HMI?. In: Inductive Automation.com [online]. Folsom, Kalifornia: *Inductive Automation*, 2018 [cit. 2021-05-10]. Dostupn  z: <https://www.inductiveautomation.com/resources/article/what-is-hmi#>
- [19] FLASP LER, Eva, Angelika HAUKE, Preethy PAPPACHAN et al. LITERATURE REVIEW THE HUMAN-MACHINE *INTERFACE AS AN EMERGING RISK*. European Agency for Safety and Health at *Work* [online]. 2006 [cit. 2021-05-10]. Dostupn  z: doi:10.2802/21813
- [20] MAN, Yemao. Human-Machine Interface Considerations for Design and Testing in Distributed Sociotechnical Systems [online]. Gothenburg,  v dsko, 2015 [cit. 2021-05-10]. Dostupn  z: [https://research.chalmers.se/publication/231914/file/231914\\_Fulltext.pdf](https://research.chalmers.se/publication/231914/file/231914_Fulltext.pdf). Diplomov  pr ca.
- [21] CANNAN, James a Huosheng HU. Human-Machine Interaction (HMI): A Survey. In: *Citeseerx.ist.psu.edu* [online]. Essex, UK: University of Essex [cit. 2021-05-10]. Dostupn  z: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.186.1644&rep=rep1&type=pdf>

- [22] IRB 4600: Industrial Robot. Abb.com [online]. ABB AB, Robotics, 2020 [cit. 2021-05-10]. Dostupné z: [https://search.abb.com/library/Download.aspx?DocumentID=ROB0109EN\\_G&LanguageCode=en&DocumentPartId=&Action=Launch](https://search.abb.com/library/Download.aspx?DocumentID=ROB0109EN_G&LanguageCode=en&DocumentPartId=&Action=Launch)
- [23] IRBP A: Industrial robot positioner. In: Abb.com [online]. ABB AB, Robotics, 2020 [cit. 2021-05-10]. Dostupné z: [https://search.abb.com/library/Download.aspx?DocumentID=ROB10080EN\\_R3&LanguageCode=en&DocumentPartId=&Action=Launch](https://search.abb.com/library/Download.aspx?DocumentID=ROB10080EN_R3&LanguageCode=en&DocumentPartId=&Action=Launch)
- [24] IRC5: Industrial Robot Controller. In: Abb.com [online]. ABB AB, Robotics, 2020 [cit. 2021-05-10]. Dostupné z: <https://search.abb.com/library/Download.aspx?DocumentID=ROB0295EN&LanguageCode=en&DocumentPartId=&Action=Launch>
- [25] Production Screen: Application-independent FlexPendant interface. In: Abb.com [online]. ABB AB, Robotics, 2013 [cit. 2021-05-10]. Dostupné z: <https://new.abb.com/products/robotics/application-software/production-screen>
- [26] RobotStudio. Abb.com [online]. ABB AB, Robotics, 2021 [cit. 2021-05-10]. Dostupné z: <https://new.abb.com/products/robotics/sk/robotstudio>
- [27] Customers Information and Questionnaire: Customers performance for installation and start-up of the „Impact Spray System“. Rattenkirchen, Nemecko, 2020.
- [28] Powder Feeder: Impact Cold Spray Systems. Impact-innovations.com [online]. Rattenkirchen, Nemecko: *Impact Innovations GmbH*, 2020 [cit. 2021-05-10]. Dostupné z: <https://impact-innovations.com/en/portfolio-item/impact-powder-feeder-5-11/>
- [29] Control Unit: Impact Cold Spray Systems. Impact-innovations.com [online]. Rattenkirchen, Nemecko: *Impact Innovations GmbH*, 2020 [cit. 2021-05-10]. Dostupné z: <https://impact-innovations.com/en/portfolio-item/impact-control-unit-5-11/>
- [30] Control Panel: Impact Cold Spray Systems. Impact-innovations.com [online]. Rattenkirchen, Nemecko: *Impact Innovations GmbH*, 2020 [cit. 2021-05-10]. Dostupné z: <https://impact-innovations.com/en/portfolio-item/impact-control-panel-5-11/>



## 9 ZOZNAM SKRATIEK, SYMBOLOV, OBRÁZKOV A TABULIEK

### 9.1 Zoznam skratiek a symbolov

HiL	Hardware in the Loop
HMI	Human Machine Interface
ISO	International Organization for Standardization
MiL	Model in the Loop
PC	Personal Computer
PLC	Programmable Logic Controller
PP	Program Pointer
SiL	Software in the Loop
TCP	Tool Center Point

### 9.2 Zoznam tabuliek

Tab 1)	Výber pravidiel z Gestaltových zákonov [19].....	25
Tab 2)	Tabuľka technických vlastností priemyselného robota IRB 4600 [22].....	28
Tab 3)	Tabuľka rozmedzí uhlového natočenia a rýchlosti osí robota IRB 4600 [22].	28
Tab 4)	Tabuľka technických vlastností polohovadla IRBP A-250 [23] .....	29
Tab 5)	Tabuľka rozmerových hodnôt polohovadla na Obr. 13, hodnoty v [mm] [23]	29

### 9.3 Zoznam obrázkov

Obr. 1)	Časové porovnanie klasického a virtuálneho uvádzania do prevádzky [3].....	17
Obr. 2)	Princíp fázy Model in the Loop [7] .....	18
Obr. 3)	Princíp fázy Software in the Loop [7] .....	18
Obr. 4)	Princíp fázy Hardware in the Loop [7].....	19
Obr. 5)	Postup virtuálneho uvedenia do prevádzky [8] .....	19
Obr. 6)	Schéma princípu technológie studeného nástreku [11] .....	20
Obr. 7)	Porovnanie dopadových rýchlostí materiálu na substrát [12].....	21
Obr. 8)	Model komunikačného cyklu medzi človekom a strojom [20].....	24
Obr. 9)	Ergonómia užívateľského prostredia [21] .....	24
Obr. 10)	Priemyselný robot IRB 4600 používaný na pracovisku .....	27
Obr. 11)	Pracovný dosah robota IRB 4600 v [mm] [22] .....	28
Obr. 12)	Polohovadlo IRBP A-250 používaný na pracovisku.....	29

Obr. 13) Rozmerová schéma polohovadla IRBP A-250 [23] .....	30
Obr. 14) Priemyselný ovládač IRC 5 používaný na pracovisku .....	31
Obr. 15) Schéma robotického pracoviska využívajúce technológiu studeného nástreku	32
Obr. 16) Hlavica studeného nástreku používaná na pracovisku .....	32
Obr. 17) Dávkovacia jednotka prášku používaná na pracovisku.....	33
Obr. 18) Chladiaca jednotka vody používaná na pracovisku .....	33
Obr. 19) a) Hlavná ovládacia tlaková jednotka a b) ovládací panel používaná na pracovisku .....	34
Obr. 20) Rozbor pracoviska v prostredí programu Robotstudio.....	35
Obr. 21) Súradnicový systém pracovného objektu „Workobject_1“.....	37
Obr. 22) Zobrazenie programovej štruktúry v programe Robotstudio .....	38
Obr. 23) Príklad a) horizontálnej, b) vertikálnej, c) diagonálnej procedúry .....	38
Obr. 24) Zobrazenie vstupných parametrov nastaviteľné obsluhou .....	39
Obr. 25) Porovnanie nastavenia ofsetov pri diagonálnom nanášaní.....	40
Obr. 26) Vývojový diagram pre horizontálny a vertikálny pohyb.....	41
Obr. 27) Rozdelenie osí pri generovaní bodov trajektórie.....	42
Obr. 28) Vývojový diagram pre diagonálny pohyb.....	43
Obr. 29) Vytvorené riadiace signály .....	44
Obr. 30) Prepojenie vstupných a výstupných signálov radiča PLC .....	45
Obr. 31) Systémové vstupy radiča PLC.....	45
Obr. 32) Vývojový diagram procedúry General Spray.....	46
Obr. 33) Vývojový diagram základného programu Main.....	47
Obr. 34) Vývojový diagram Trap rutiny Stop .....	48
Obr. 35) Vývojový diagram Trap rutiny Pause .....	49
Obr. 36) Vývojový diagram Trap rutiny World_Zone .....	50
Obr. 37) Hlavná obrazovka HMI aplikácie.....	51
Obr. 38) Obrazovka 2 – Layers pre nastavenie vrstiev HMI aplikácie .....	53
Obr. 39) Obrazovka 3 – Dimensions pre nastavenie hodnôt striekaného objektu a orientácie TCP aplikácie HMI.....	53
Obr. 40) Obrazovka 4 – General Spray pre nahratie a odstránenie externých modulov všeobecného programu aplikácie HMI .....	54
Obr. 41) Vytvorenie spojenia medzi Robotstudiom a priemyselným ovládačom IRC 557	
Obr. 42) Uloženie a nahratie programu .....	57
Obr. 43) Možnosti na správu systémových parametrov .....	58
Obr. 44) Kontextové okno pre uloženie systémových parametrov ovládača .....	58
Obr. 45) Virtuálne a reálne pracovisko.....	59

## 10 ZOZNAM PRÍLOH

Príloha 1 – Užívateľský manuál ovládania programu Cold Spray Linear

Príloha 2 – Prezentačné video programu pre ovládanie pracoviska na studený nástrek





# Užívateľský manuál ovládania programu Cold Spray Linear

## Obsah:

I. Predstavenie a spustenie HMI aplikácie.....	1
II. Rozdelenie obrazoviek.....	2
III. Konfigurácia primárnych bodov .....	6
IV. Parametre univerzálneho pohybového programu.....	7
V. Práca s externými modulmi .....	8
VI. Ovládanie robota .....	9
VII. Tvorba externých modulov.....	10
VIII. Odstránenie bežných chýb .....	13

## I. Predstavenie a spustenie HMI aplikácie

**HMI SprayGun App** je aplikácia určená na ovládanie ABB priemyselného robota riadiaceho nanášanie studeného nástreku na nerotujúce obrobky upevnené na ABB polohovadle. Aplikácia ponúka široké možnosti nastavení automatizovaných pohybových inštrukcií pre presné nanášanie studeného nástreku. Aplikácia je zložená zo štyroch užívateľských obrazoviek, medzi ktorými sa preklikáva pomocou jednotnej lišty umiestnenej v spodnej časti každej z obrazoviek:

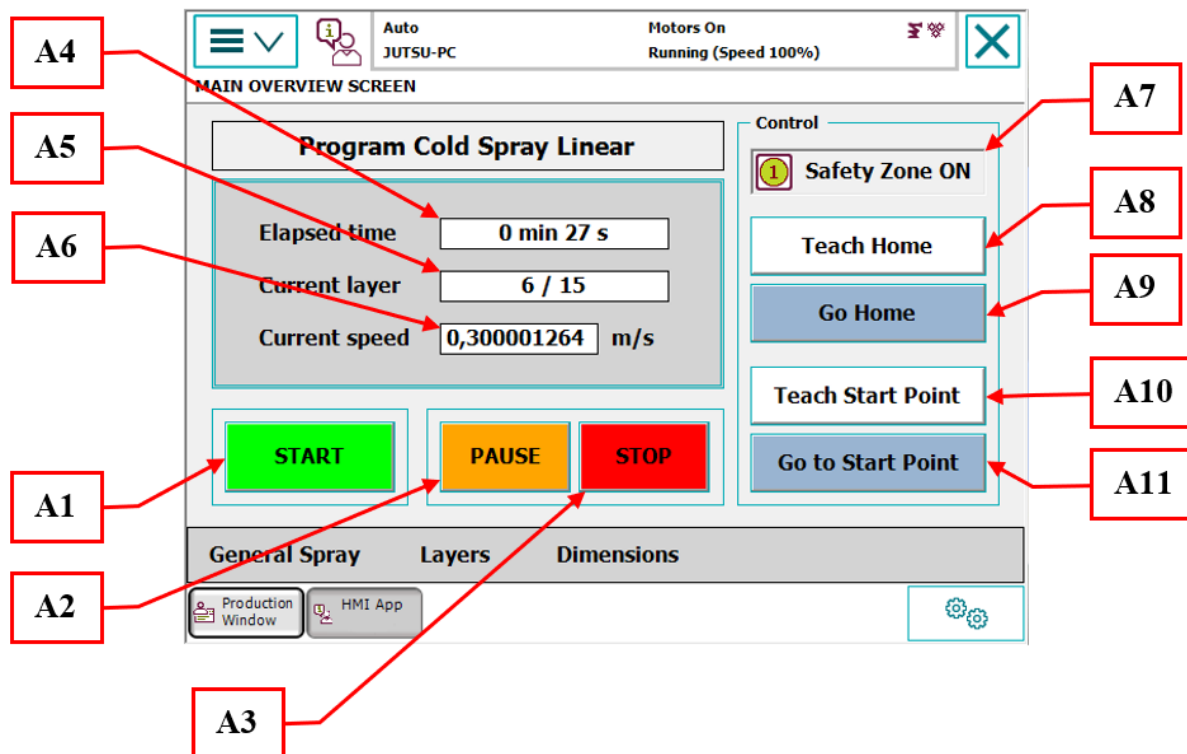
- Obrazovka A – MAIN
  - Obrazovka B – General Spray
  - Obrazovka C – Layers
  - Obrazovka D – Dimensions
- 
- Pred spustením HMI aplikácie je nutné v rámci „Production Window“ vo Flexpendante, nahráť program „ **ColdSpray\_Linear** “ pomocou tlačidla „Load Program“.
  - HMI aplikácia sa nachádza v rolovacom menu Flexpendantu vľavo hore. Po kliknutí na názov aplikácie sa spustí jej hlavná základná obrazovka – *Obrazovka A - Main*.
  - Pre ovládanie HMI aplikácie je nutné ovládač prepnúť do režimu „AUTO“ pomocou kľúča na skrini ovládača.
  - Vždy (!) po prepnutí do režimu „AUTO“ je nutné stlačiť hardvérové tlačidlo *PLAY* v pravom dolnom rohu Flexpendantu, aby sa program aktivoval.



## II. Rozdelenie obrazoviek

### Ponuka Obrazovky A – Main

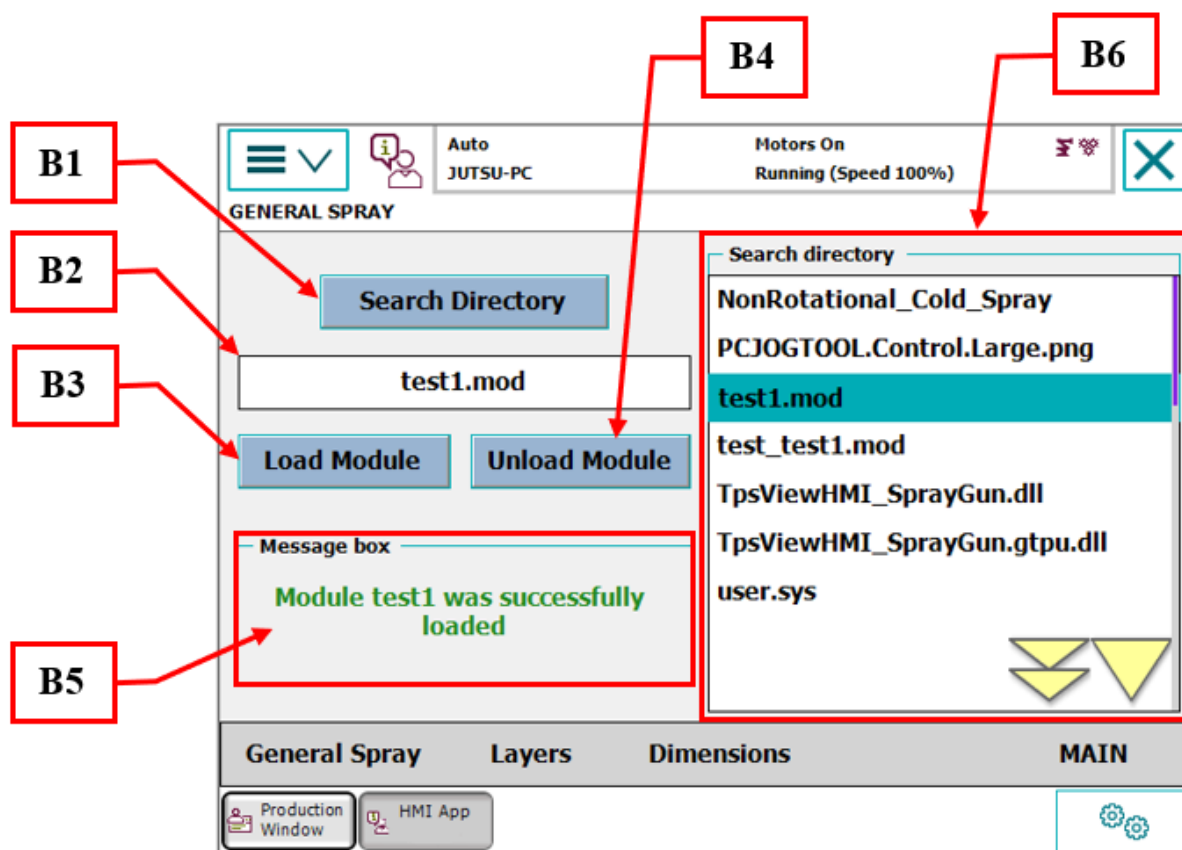
<b>A1</b>	Start	Tlačidlo na spustenie automatizovaného pohybu.
<b>A2</b>	Pause	Tlačidlo na zastavenie pohybu na konci inštrukcie a presun do pozície HOME.
<b>A3</b>	Stop	Tlačidlo na okamžité zastavenie pohybu a presun do pozície HOME.
<b>A4</b>	Elapsed time	Ukazovateľ uplynulého času aktuálneho studeného nástreku.
<b>A5</b>	Current layer	Ukazovateľ aktuálne striekanej vrstvy a celkového počtu vrstiev.
<b>A6</b>	Current speed	Ukazovateľ aktuálnej rýchlosti posuvu TCP.
<b>A7</b>	Safety Zone	Spínač aktivácie/deaktivácie bezpečnostnej zóny <i>Safety Zone</i> . *
<b>A8</b>	Teach Home	Tlačidlo na zapísanie aktuálnej pozície robota ako pozícia HOME.
<b>A9</b>	Go Home	Tlačidlo na presunutie robota do pozície HOME.
<b>A10</b>	Teach Start Point	Tlačidlo na zapísanie aktuálnej pozície robota ako pozícia START POINT.
<b>A11</b>	Go to Start Point	Tlačidlo na presunutie robota do pozície START POINT.



\* *Safety Zone* je zóna, ktorá predstavuje hranicu, kde sa robot môže pohybovať. Ak robot túto hranicu prekročí, zastaví sa jeho pohyb a presunie sa do pozície HOME. Zóna povoľuje pohyb robota max. 10 mm pred zverákom uchytenom na polohovadle a max. po pravý okraj upínacej dosky polohovadla. *Safety Zone* zamedzuje kolízií robota s polohovadlom, resp. nechcenému nástreku na rameno polohovadla.

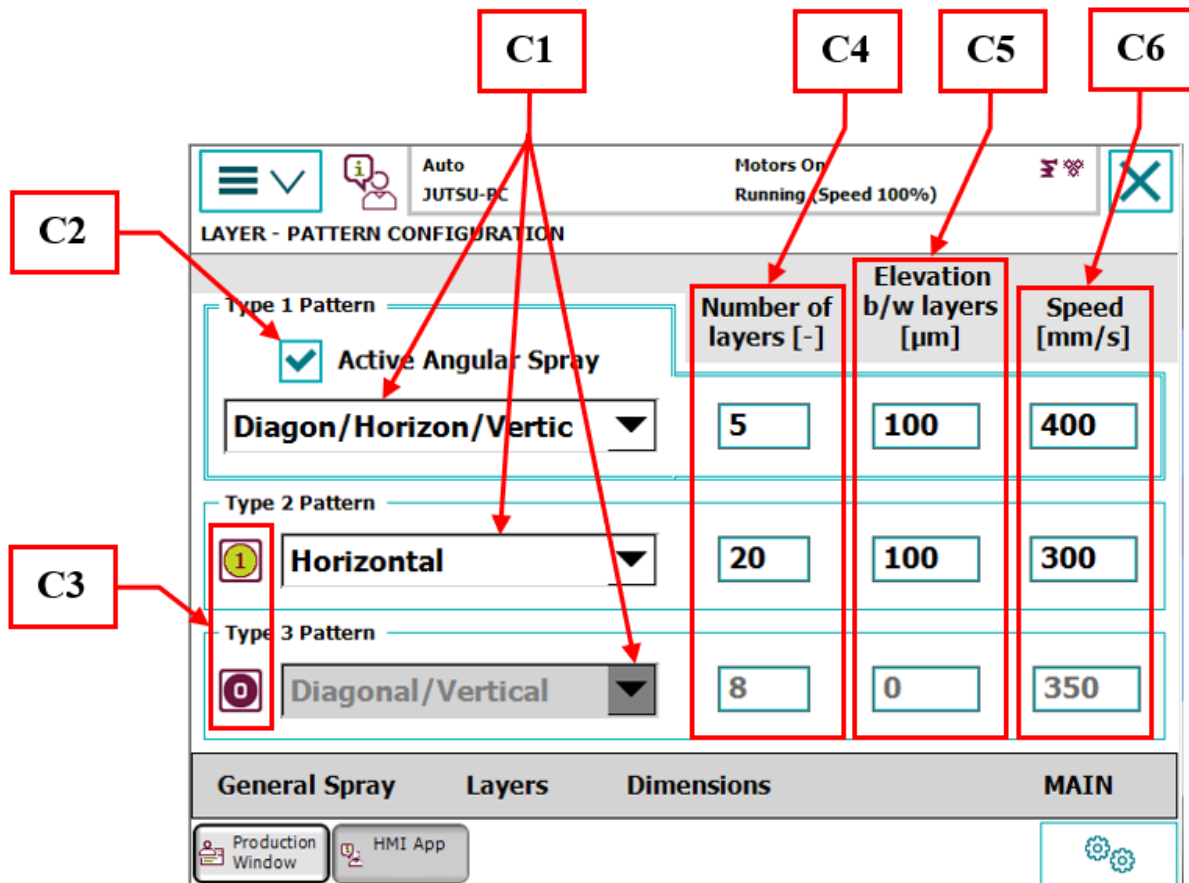
## Ponuka Obrazovky B – General Spray

<b>B1</b>	Search Directory	Tlačidlo na prehľadanie/aktualizovanie zobrazovaných súborov.
<b>B2</b>	Selected file	Zobrazenie názvu aktuálne zvoleného súboru.
<b>B3</b>	Load Module	Tlačidlo na nahranie vybraného externého modulu do programu.
<b>B4</b>	Unload Module	Tlačidlo na vyradenie vybraného externého modulu z programu.
<b>B5</b>	Message box	Informačné okno poskytuje spätnú väzbu pri práci s externými modulmi.
<b>B6</b>	Directory	Ponuka súborov z priečinku HOME.



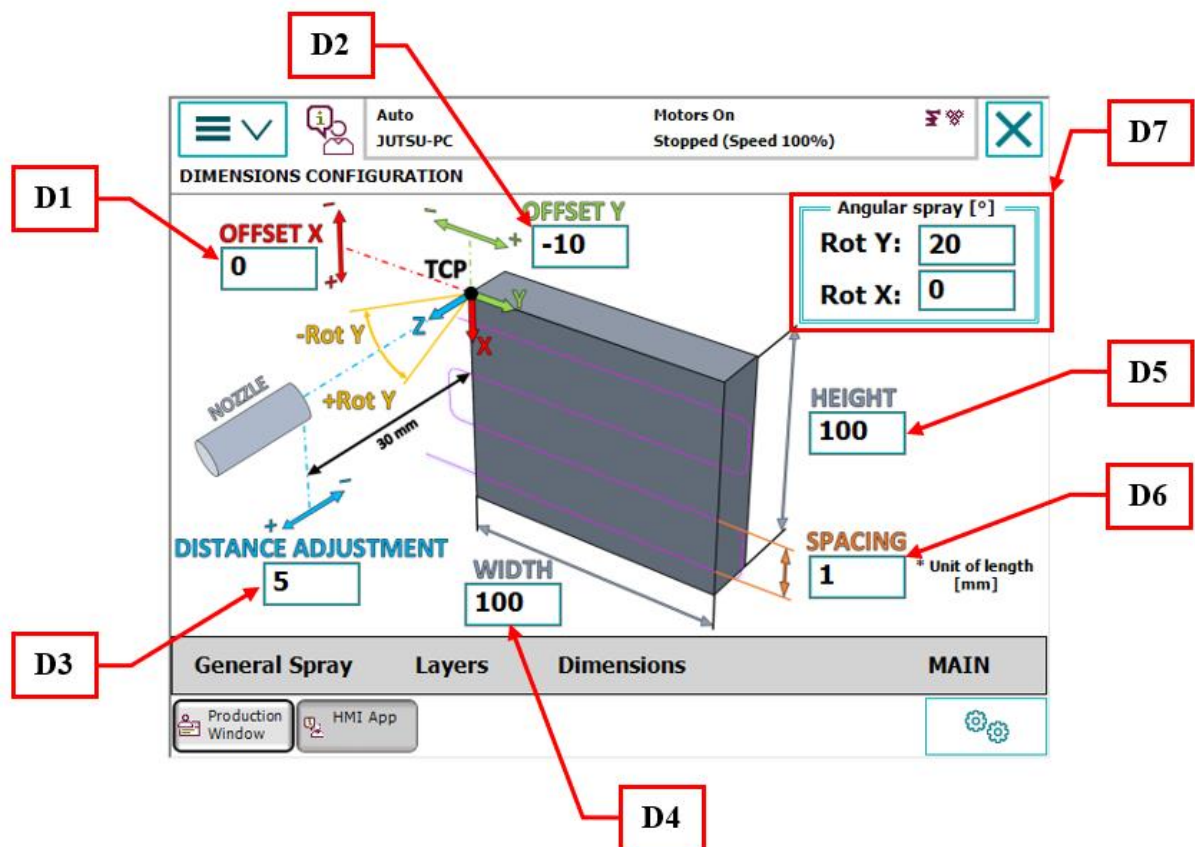
## Ponuka Obrazovky C – Layers

<b>C1</b>	Pattern 1-3	Nastavenie typu pohybových procedúr TCP nad substrátom.
<b>C2</b>	Active Angular Spray switch	Spínač pre aktivovanie/deaktivovanie uhlového natočenia hlavice pre prvú pohybovú procedúru.
<b>C3</b>	Pattern switch	Spínače pre aktivovanie/deaktivovanie druhej a tretej pohybovej procedúry.
<b>C4</b>	Number of layers	Nastavenie počtu nanášaných vrstiev konkrétnej procedúry.
<b>C5</b>	Elevation b/w layers	Nastavenie vzdialenosti posunutia hlavice od substrátu medzi každou nanášanou vrstvou. <b>Upozornenie:</b> jednotka $\mu\text{m}$ (mikrometer).
<b>C6</b>	Speed	Nastavenie rýchlosti pohybu TCP pri nanášaní vrstvy na substrát.



## Ponuka Obrazovky 4 – Dimensions

<b>D1</b>	Offset X	Nastavenie veľkosti prejazdu TCP v ose X. Kladná hodnota presunie trajektóriu pohybu v smere od okraja substrátu <b>dovnútra</b> . Záporná hodnota presunie trajektóriu pohybu v smere od okraja <b>von</b> .
<b>D2</b>	Offset Y	Nastavenie veľkosti prejazdu TCP v ose Y. Kladná hodnota presunie trajektóriu pohybu v smere od okraja substrátu <b>dovnútra</b> . Záporná hodnota presunie trajektóriu pohybu v smere od okraja <b>von</b> .
<b>D3</b>	Distance Adjustment	Dodatočná úprava vzdialenosti konca dýzy hlavice od povrchu substrátu. Ak hodnota je „0“, dýza je vzdialená 30 mm od povrchu substrátu (30 mm je veľkosť distančného TCP nástavca).
<b>D4</b>	Width	Nastavenie šírky upevneného obrobku.
<b>D5</b>	Height	Nastavenie výšky upevneného obrobku.
<b>D6</b>	Spacing	Nastavenie veľkosti riadkovania v rámci vrstvy.
<b>D7</b>	Angular spray	Nastavenie uhlového natočenia hlavice okolo osy X a osy Y. Natočenie je okolo bodu TCP (30 mm od konca dýzy). Orientácia natočenia je podľa pravidla pravej ruky.



### III. Konfigurácia primárnych bodov

Pred spustením automatizovaného pohybu je nutné zvoliť a nakonfigurovať 2 základné body:

- Bezpečnostný bod HOME
- Začiatkový bod START POINT

**Bezpečnostný bod HOME** predstavuje pozíciu robota, v ktorej robot čaká pred spustením studeného nástreku a do ktorého sa robot vracia po vykonaní všetkých pohybových procedúr. Do tohto bodu sa robot taktiež presúva pri každom prerušení programu po stlačení tlačidla STOP (A3) alebo PAUSE (A2).

- **Bod HOME musí (!) byť vždy nakonfigurovaný vľavo od substrátu.**

**Začiatkový bod START POINT** predstavuje ľavý horný rožný bod substrátu, upevneného na polohovadle od ktorého budú generované a vykonávané všetky automatizované pohyby programu. Bod sa konfiguruje pomocou 30 mm dlhého TCP nástavca upevneného na konci dýzy hlavice studeného nástreku.

- **Bod START POINT musí (!) byť vždy nakonfigurovaný v ľavom hornom rohu upevneného substrátu.**

#### Postup konfigurácie bodu HOME:

1. Prepnete ovládač robota IRC 5 do módu „MANUAL“ pomocou kľúča priamo na skrini ovládača (kontroleru).
2. Manuálnym ovládaním cez Flexpendant nastavte robot do požadovanej polohy vľavo od substrátu.
3. Prepnete ovládač robota do módu „AUTO“ pomocou kľúča na skrini ovládača.
4. Zapnite motory robota tlačidlom na skrini ovládača.
5. Stlačte hardvérové tlačidlo *PLAY* pre spustenie užívateľského programu.
6. Na *Obrazovke A - MAIN* stlačte tlačidlo *Teach Home* (A8) a potvrdzte kontrolnú hlášku.

#### Postup konfigurácie bodu START POINT:

1. Upevnite distančný TCP nástavec na koniec hlavice studeného nástreku.
2. Prepnete ovládač robota IRC 5 do módu „MANUAL“ pomocou kľúča priamo na skrini ovládača IRC 5.
3. Manuálnym ovládaním cez Flexpendant nastavte robot tak, aby sa koniec TCP nástavca dotýkal ľavého horného rohu substrátu.
4. Prepnete ovládač robota do módu „AUTO“ pomocou kľúča na skrini ovládača.
5. Zapnite motory robota tlačidlom na skrini ovládača.
6. Stlačte hardvérové tlačidlo *PLAY* pre aktivovanie užívateľského programu.
7. Na *Obrazovke A - MAIN* stlačte tlačidlo *Teach Start Point* (A10) a potvrdzte kontrolnú hlášku.

Pre overenie správnosti konfigurácie bodov je možnosť na *Obrazovke A – MAIN* využiť tlačidlá *Go Home* a *Go To Start Point*. Stlačením jedného z týchto tlačidiel sa robot presunie do konkrétnej pozície a operátor overí správnosť zapísaného bodu v programe.

## IV. Parametre univerzálneho pohybového programu

Univerzálny pohybový program je program vykonávajúci automatizovaný pohyb hlavice studeného nástreku nad substrátom podľa parametrov, ktoré mu boli zadané v rámci Flexpendantu. Parametre pohybového programu sa nastavujú na *Obrazovke C – Layers* a *Obrazovke D – Dimensions*.

### Obrazovka C – Layers

*Obrazovka C – Layers* poskytuje možnosť nastavenia parametrov pre 3 separátne pohybové procedúry, ktoré sú robotom vykonávané v poradí jeden za druhým. Procedúry sú viditeľne oddelené a pomenované „Type 1 – 3 Pattern“. *Type 1 Pattern* je prvá, vždy vykonávaná procedúra, v rámci ktorej je taktiež možné nastaviť uhlové natočenie hlavice studeného nástreku. *Type 2 Pattern* a *Type 3 Pattern* povoľujú iba normálový nástrek na povrch substrátu.

Každá z troch procedúr má možnosť voľby z 8 typov pohybových procedúr TCP (C1):

- Diagonálny pohyb
  - Horizontálny pohyb
  - Vertikálny pohyb
  - Striedanie horizontálneho a vertikálneho pohybu
  - Striedanie diagonálneho a horizontálneho pohybu
  - Striedanie diagonálneho a vertikálneho pohybu
  - Striedanie diagonálneho, horizontálneho a vertikálneho pohybu
  - Pohyb definovaný z externého modulu
- 
- Ak je zvolený typ pohybu definovaný externým modulom, na *Obrazovke D – Dimensions* sa deaktivuje možnosť nastavenia niektorých parametrov.
  - Spínačmi (C3) sa nastavuje, či v programe bude využívaná druhá a tretia pohybová procedúra.
  - Pre každú aktívnu procedúru je nutné zadať parametre (C4), (C5), (C6).

### Obrazovka D – Dimensions

*Obrazovka D – Dimensions* poskytuje možnosť nastavenia rozmerových parametrov substrátu a pohybov hlavice studeného nástreku nad ním. Po zadaní veľkosti substrátu je možnosť využiť ofsetové nastavenie pohybovej trajektórie. Zadaná hodnota sa automaticky aplikuje na oba protiľahlé okraje substrátu.

- Jednotka rozmerových parametrov je milimeter [mm], uhlové natočenie je v stupňoch [°].
- V prípade použitia hodnoty v *Angular spray* (D7) inej než „0“, je potrebné zaškrtnúť spínač *Active Angular Spray* (C2) na *Obrazovke C – Layers*.
- Nie je možné používať *Angular spray* (D7) v kombinácií s hodnotou *Distance Adjustment* (D3) inou než „0“.

## V. Práca s externými modulmi

Práca s externými modulmi je možná na *Obrazovke B – General Spray*. Pomocou tejto obrazovky je možné nahrat' externý modul do aktuálneho programu. Takto nahratý externý modul môže obsahovať presne nadefinovaný pohyb, aký má robot vykonávať nad špeciálne tvarovanou súčiastkou.

### Postup nastavenia externého modulu na nástrek:

1. Súbor „. MOD“ prekopírujeme do priečinku „HOME“ ovládača IRC 5 pomocou USB alebo ethernetového káblu.
  2. Prepne ovládač robota do módu „AUTO“ pomocou kľúča na skrini ovládača.
  3. Na Flexpendante stlačíme hardvérové tlačidlo *PLAY* pre aktivovanie programu.
  4. V aplikácii otvoríme *Obrazovku B – General Spray*.
  5. Stlačíme tlačidlo *Search Directory* (B1) pre prehľadanie priečinku HOME.
  6. Zvolíme požadovaný súbor s príponou „. MOD“ z ponuky súborov (B6)
  7. Stlačíme tlačidlo *Load Module* (B3) pre nahranie modulu do aktuálneho programu.
  8. Na *Obrazovke C – Layers* zvolíme v type pohybu (C1) možnosť „General Program“ a nastavíme zvyšné parametre na *Obrazovke C a D*.
- Možnosť „General Program“ v type pohybu TCP (C1) nie je možné kombinovať s ostatnými typmi pohybov.
  - Pri využívaní možnosti „General Program“ v type pohybu TCP (C1) je na *Obrazovke D – Dimensions* k dispozícii iba nastavenie parametrov uhlového natočenia (D7). Ostatné parametre sú deaktivované, nakoľko „General Program“ používa vlastné parametre.
  - Pre vyradenie vybraného externého modulu z programu stačí v ponuke súborov (B6) zvoliť súbor s rovnakým názvom, tak aby sa jeho názov ukázal v okne aktuálne zvoleného súboru (B2) a stlačiť tlačidlo *Unload Module* (B4).



## VI. Ovládanie robota

### Postup spustenia programu:

1. Nakonfigurujte primárne body podľa kapitoly III.
2. Nastavte parametre programu podľa toho, čo chcete striekať studeným nástrekom.
  - 2a. Pre striekanie univerzálnym programom nastavte parametre na *Obrazovke C* a *Obrazovke D*.
  - 2b. Pre striekanie s využitím externých modulov postupujte podľa kapitoly V.
3. Spínačom (A7) zvolte či chcete nanášať s aktívnou funkciou *Safety Zone*.
4. Automatický pohyb spustíte pomocou tlačidla *Start* (A1).

Vykonávané pohybové procedúry robota je možné prerušiť 2 spôsobmi:

- **Tlačidlo STOP** (A3) ihneď zastaví pohyb a najkratšou cestou TCP presunie robot do pozície HOME. Program je resetovaný a je možné ho spustiť jedine od začiatku.
- **Tlačidlo PAUSE** (A2) počká, kým robot príde k najbližšej hrane obrobku, presunie sa do pozície HOME a čaká na ďalšiu inštrukciu:
  - Inštrukcia *Start / Continue* (A1) vráti robot na posledné miesto, kde bol pohyb prerušený a program pokračuje ďalej.
  - Inštrukcia *Stop* (A3) ukončí a resetuje program.

## VII. Tvorba externých modulov

Pre správnu kompatibilitu externých modulov s užívateľským softvérom je nutné pri vytváraní týchto modulov dodržiavať nasledujúce pravidlá:

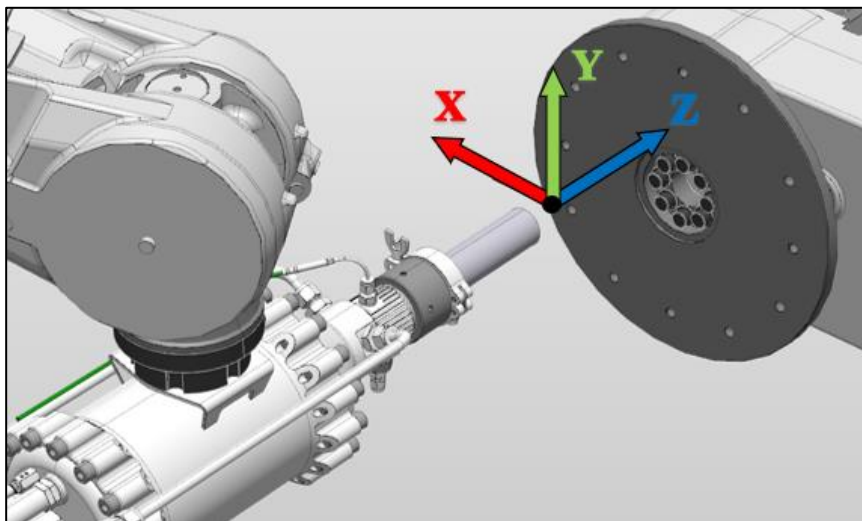
- Názov hlavnej procedúry sa musí (!) nazývať „**General\_Procedure()**“.
- Hlavná procedúra predstavuje lokálny MAIN podprogram a z nej sú volané všetky ostatné procedúry.
- Jeden cyklus hlavnej procedúry = Jedna nanosená vrstva studeného nástreku
  - Hlavná procedúra „General\_Procedure“ musí obsahovať všetky inštrukcie pre nanosenie jednej vrstvy nástreku.
- Začiatok prvej vrstvy nástreku musí začínať v bode odvodenom od počiatočného bodu START POINT.
- Bod START POINT je definovaný v užívateľskom programe a nesmie byť definovaný v externom module.
- Od počiatočného bodu START POINT musia byť parametricky vytvorené všetky ostatné body, ktorými TCP robota bude prechádzať.
  - Pre tvorbu bodov treba použiť funkciu „RelTool“ presne podľa nasledujúceho príkladu.

Offset X

Offset Y

```
Point3:=RelTool(START_POINT, -100, -100, -move_Z\Rx:=Rotation_Y\Ry:=Rotation_X);
```

- Z uvedenej funkcie je možné meniť iba Offset X a Offset Y, zvyšné parametre aj s rotáciami a „Ofsetom Z = - move\_Z“ treba kopírovať presne podľa príkladu.
- Nepoužívať (!) funkciu „Offs“.
- Nasledujúci obrázok zobrazuje súradnicový systém používaný funkciou „RelTool“.



- Rovina XY je rovnobežná s plochou polohovadla a os Z smeruje kolmo do polohovadla.

- Pohybové inštrukcie „MoveL“, „MoveJ“ a i. musia obsahovať nasledujúce parametre:
  - Parameter rýchlosti:
    - vMove – rýchlosť pri pohybe mimo substrátu (400 mm/s)
    - vSpray – rýchlosť pri pohybe nad substrátom (mení operátor na Flexpendante).
  - Parameter zóny:
    - zMove – zóna pri pohybe mimo substrát (z10)
    - zSpray – zóna pri pohybe nad substrátom (z1)
- V pohybových inštrukciách treba definovať nástroj – „ **toolImpactGun** “.
- V pohybových inštrukciách treba definovať pracovný súradnicový systém – „ **Workobject\_1** “.
- Pre využitie funkcie počítania času na *Obrazovke A – Main* je nutné podľa potreby pridávať do kódu nasledujúce dva riadky.

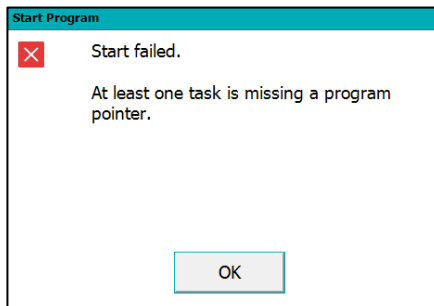
```
time:=Trunc(ClkRead(clock1));  
time_text:=SecondsToTime(time);
```

## Vzorový příklad externého modulu:

```
1
2 MODULE Example_module
3
4 !*****
5 ! Definovanie dát pre General Program
6 !*****
7 VAR robtarget Point1;
8 VAR robtarget Point2;
9 VAR robtarget Point3;
10 VAR robtarget Point4;
11
12 !*****
13 ! Hlavná MAIN procedúra General Program
14 !*****
15 PROC General_Procedure()
16     Proc1;
17     Proc2;
18 ENDPROC
19
20 !*****
21 ! SUB-procedúry General Program
22 !*****
23 PROC Proc1()
24     Point1:=RelTool(START_POINT,0,0,-move_Z\Rx:=Rotation_Y\Ry:=Rotation_X);
25     Point2:=RelTool(START_POINT,-100,0,-move_Z\Rx:=Rotation_Y\Ry:=Rotation_X);
26     Point3:=RelTool(START_POINT,-100,-100,-move_Z\Rx:=Rotation_Y\Ry:=Rotation_X);
27     Point4:=RelTool(START_POINT,0,-100,-move_Z\Rx:=Rotation_Y\Ry:=Rotation_X);
28 ENDPROC
29
30 PROC Proc2()
31     MoveL Point1,vSpray,zSpray,toolImpactGun\WObj:=Workobject_1;
32     MoveL Point2,vSpray,zSpray,toolImpactGun\WObj:=Workobject_1;
33     time:=Trunc(ClkRead(clock1));
34     time_text:=SecondsToTime(time);
35     MoveL Point3,vSpray,zSpray,toolImpactGun\WObj:=Workobject_1;
36     MoveL Point4,vSpray,zSpray,toolImpactGun\WObj:=Workobject_1;
37     time:=Trunc(ClkRead(clock1));
38     time_text:=SecondsToTime(time);
39 ENDPROC
40
41 ENDMODULE
```

## VIII. Odstránenie bežných chýb

### Chyba 1:



### Riešenie:

Hore uvedená chybová hláška oznamuje absenciu programového ukazovateľa. Pre odstránenie chyby postupujte nasledovne:

1. V ponuke Flexpendantu kliknite na „Production Window“.
2. Na spodnej lište kliknite na tlačidlo „PP To Main“.
3. Aktivujte program pomocou hardvérového tlačidla PLAY.

### Chyba 2:

Tlačidlá na aktuálnej obrazovke v aplikácií HMI SprayGun App sú neaktívne.

### Riešenie:

Postupujte podľa nasledujúcich krokov od č.1 pokým chybu neodstránite.

1. Skontrolujte, či je robot nastavený v režime AUTO a zapnite motory na skrini ovládača IRC 5.
2. Na spodnej lište aktuálnej obrazovky kliknite na tlačidlo zobrazenia inej obrazovky aplikácie HMI SprayGun App.
3. Stlačte hardvérové tlačidlo PLAY pre aktivovanie programu.
4. Prepnete robot do režimu MANUAL a späť do režimu AUTO, zapnite motory na skrini ovládača IRC 5 a stlačte hardvérové tlačidlo PLAY.