



Bakalářská práce

Digitální dvojče reálného zařízení pro výuku programování PLC

Studijní program:

B0714A270001 Mechatronika

Autor práce:

Zdeněk Ort

Vedoucí práce:

Ing. Petr Školník, Ph.D.

Ústav mechatroniky a technické informatiky

Liberec 2024



Zadání bakalářské práce

Digitální dvojče reálného zařízení pro výuku programování PLC

Jméno a příjmení:

Zdeněk Ort

Osobní číslo:

M21000046

Studijní program:

B0714A270001 Mechatronika

Zadávací katedra:

Ústav mechatroniky a technické informatiky

Akademický rok:

2023/2024

Zásady pro vypracování:

1. Seznamte se s prostředím SIMIT a vytvářením vizualizace a simulačních modelů v něm.
2. Vyberte vhodný reálný technologický proces pro účely virtuálního prototypování.
3. Realizujte model zvoleného procesu v prostředí SIMIT. Model doplňte vhodnou vizualizací a signalizací chybových stavů.
4. Navrhněte řídicí algoritmus pro PLC Simatic a ten pak ověřte na modelu.
5. Pro řízení modelového procesu vypracujte přehledný návod.

Rozsah grafických prací: dle potřeby dokumentace
Rozsah pracovní zprávy: 30 až 40 stran
Forma zpracování práce: tištěná/elektronická
Jazyk práce: čeština

Seznam odborné literatury:

- [1] Úvod do virtuálního zprovoznění na platformě [online]. [cit. 2022-10-10]. Dostupné z: <https://cz.webinar.siemens.com/uvod-do-virtualniho-zprovozni-na/>
- [2] SIMIT/MCD/AMESIM – Virtual commissioning in machine building [online]. [cit. 2022-10-10]. Dostupné z: <https://support.industry.siemens.com/cs/document/109777165/>
- [3] KARNOPP, Dean, Donald L. MARGOLIS a Ronald C. ROSENBERG. *System dynamics: modeling and simulation of mechatronic systems*. 5th ed. Hoboken, NJ: Wiley, 2012. ISBN 978-0-470-88908-4. *Siemens Step 7 (TIA PORTAL) Programming, a Practical Approach*, 2nd Edition. Independently published, 2019. ISBN 978-1091474109.

Vedoucí práce: Ing. Petr Školník, Ph.D.
Ústav mechatroniky a technické informatiky

Datum zadání práce: 12. října 2023
Předpokládaný termín odevzdání: 14. května 2024

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.

doc. Ing. Josef Černoorský, Ph.D.
vedoucí ústavu

V Liberci dne 12. října 2023

Prohlášení

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

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

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

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

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

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

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

Poděkování

Hlavní poděkování patří mému vedoucímu práce Ing. Petrovi Školníkovi, Ph.D. za poskytnutí programů a rad k realizaci této bakalářské práce práce.

Také chci poděkovat doc. Ing. Josefovi Černoorskému, Ph.D. a Ing. Janu Koprnickému, Ph.D. za předání potřebných informací ohledně bakalářské práce a za podporu s prostředím LaTeX v průběhu bakalářského semináře.

Další poděkování je věnováno společnosti ORT Nový Bydžov spol. s r. o. za inspiraci technologického procesu použitého v této práci.

Důležité poděkování patří mé drahé polovičce Julii Kleinové za obrovskou podporu při realizaci této práce a za poskytnuté konzultace.

Nakonec bych chtěl poděkovat Bc. Davidovi Sládečkovi za pomoc s jazykovou strukturou celé práce.

Digitální dvojče reálného zařízení pro výuku programování PLC

Abstrakt

Bakalářská práce je zaměřena na tvorbu digitálního dvojčete vybraného technologického procesu. Práce představuje možnosti pro virtuální testování řídicích systémů bez potřeby vytváření fyzických prototypů. S pomocí programu SIMIT od společnosti Siemens je vytvořena simulace a vizualizace digitálního dvojčete. Simulace a vizualizace slouží jako zpětná vazba řídicímu programu vytvořeném v prostředí TIA Portal. Cílem práce je ukázat možnosti simulace a vizualizace digitálních dvojčat, které mohou posloužit jako nástroj pro vzdělávání a rozvoj praktických dovedností v oblasti mechatroniky a automatizace.

Digitální dvojče reálného zařízení je virtuální simulace reálného zařízení. Pro zařízení se vytvoří vizualizace, kde jsou zobrazeny hlavní komponenty. Následně se tyto komponenty rozpohybují a pomocí řídicího programu a simulovaného prostředí otestují jednotlivé funkce tohoto zařízení.

Práce zahrnuje analýzu procesu prostřihovacího zařízení, návrh řídicího algoritmu a jeho implementaci v simulačním prostředí. Detailně jsou popsány kroky od konceptu po realizaci celého digitálního dvojčete, s důrazem na řídicí systém a interaktivní vizualizaci. Tato práce by měla sloužit jako předloha pro výuku. K této práci je také příloha, kterou lze přímo použít jako studijní materiál v hodinách zaměřených na programování PLC a vytváření automatizace strojních zařízení.

Klíčová slova: Digitální dvojče, SIMIT, TIA Portal, PLC programování.

Digital twin of a real device for PLC programming training

Abstract

The bachelor's thesis is focused on the creation of a digital twin of a selected technological process. The work presents possibilities for virtual testing of control systems without the need to create physical prototypes. With the help of the SIMIT program from Siemens, a simulation and visualization of a digital twin is created. Simulation and visualization serve as feedback to the control program created in the TIA Portal environment. The aim of the work is to show the possibilities of simulation and visualization of digital twins, which can serve as a tool for education and development of practical skills in the field of mechatronics and automation.

A digital twin of a real device is a virtual simulation of a real device. A visualization is created for the device, showing the main components. Subsequently, these components are moved and with the help of a control program and a simulated environment, they test the individual functions of this device.

The work includes the analysis of the slitting device process, the design of the control algorithm and its implementation in a simulation environment. The steps from the concept to the realization of the entire digital twin are described in detail, with an emphasis on the control system and interactive visualization. This work should serve as a model for teaching. There is also an appendix to this work that can be directly used as study material in classes focused on PLC programming and creating automation of machinery.

Keywords: Digital twin, SIMIT, TIA Portal, PLC programming.

Obsah

Úvod	12
1 Technologický proces	14
1.1 Výběr zařízení	14
1.2 Základní komponenty	15
1.2.1 Hydraulický válec	16
1.2.2 Upínka	16
1.2.3 Optický a mechanický senzor	16
1.2.4 Signalizační maják a ovládací panel	16
1.3 Princip práce zařízení	17
1.3.1 Přidané funkce	17
1.4 Schéma logiky programu	18
2 Vizualizace	21
2.1 SIMIT	21
2.1.1 Couplings - propojovací oddíl	22
2.1.2 Visualization - vizualizační oddíl	22
2.1.3 Chart - simulační oddíl	22
2.1.4 Propojení prostředí SIMIT a TIA portal	23
2.2 Návrh vizualizace	24
2.2.1 Ovládací panel	25
2.2.2 Založení dílu	26
3 Simulace zařízení v prostředí SIMIT	27
3.1 Simulační prvky	27
3.1.1 Generátor signálu Ramp	27
3.1.2 Vstupy a výstupy simulace	27
3.1.3 Logické prvky	28
3.2 Výběr Pravého nebo levého režimu	28
3.3 Zpětná vazba	30
4 Návrh řídicího programu	31
4.1 Ladder diagram	31
4.1.1 Kontakty a cívky	31
4.1.2 SR a RS - Flip Flop	32

4.1.3	TON časovač	32
4.2	TIA Portal	33
4.2.1	Výběr PLC	34
4.2.2	Proměnné - Tagy	35
4.2.3	Rozdělení do podprogramů	36
4.3	Main Řídícího Programu	37
4.3.1	Main	37
4.3.2	Reset a Bezpečnost	38
4.3.3	Zastavení zařízení	39
4.3.4	Signalizace	40
4.4	Střížné cykly řídicího programu	42
4.4.1	Uzavření upínek	42
4.4.2	Vysunutí střížného válce	43
4.4.3	Zasunutí střížného válce	44
4.4.4	Otevření upínek	45
	Závěr	46
	Použitá literatura	47
	Přílohy	48

Seznam zkratk

EU	Evropská unie
PLC	Programmable Logic Controller
LAD	Ladder diagram
FBD	Function Block Diagram
STL	Structured Text
SCL	Structured Control Language
SFC	Sequential Function Chart
TIA Portal	Totally Integrated Automation Portal
NO	Normally open
NC	Normally closed
SR	SET-RESET
RS	RESET-SET
TON	Timer ON

Seznam tabulek

1.1	Tabulka s komponenty	16
1.2	Tabulka postupu řízení	20

Seznam obrázků

1.1	Prostřihovací zařízení z firmy ORT s.r.o.	14
1.2	Schéma řídicího programu	19
2.1	Prostředí programu SIMIT	21
2.2	Prostředí PLCSIM Advanced	23
2.3	Vizualizace prostřihového zařízení	24
2.4	Ovládací panel	25
2.5	Založení dílu	26
3.1	Generátor signálu Ramp se vstupy a výstupy	28
3.2	Logické prvky	28
3.3	Výběr režimu Levé / Pravé lože	29
3.4	Zpětná vazba a podklady pro vizualizaci	30
4.1	Kontakt NO, NC a cívka	31
4.2	Blok SR obvodu. Obrázek použit ze zdroje [1]	32
4.3	TON časovač	32
4.4	Prostředí TIA Portal	33
4.5	Příklad proměnných (TAGů)	35
4.6	Project tree v TIA Portal	36
4.7	Větev Main	37
4.8	Větev pro resetování pozice	38
4.9	Větev pro zastavení stroje	39
4.10	Větev Signalizace stavu stroje	41
4.11	Větev pro uzavření upínek	42
4.12	Větev pro vysunutí střižného válce	43
4.13	Větev pro zasunutí válce	44
4.14	Větev pro otevření upínek	45

Úvod

V rámci současných trendů digitalizace průmyslových procesů a významného rozvoje průmyslu 4.0 přichází stále větší důraz na využití digitálních technologií nejen ve výrobních, ale i vzdělávacích procesech. Tato bakalářská práce se zabývá vývojem digitálního dvojčete prostřihovacího zařízení, jehož primárním účelem je podpora výuky programování řídicích systémů PLC. Digitalizace a simulace procesů umožňuje výrazně snížit náklady na vývoj a zvýšit efektivitu a bezpečnost výuky a testování.

Cílem práce je demonstrovat, jak může simulace digitálního dvojčete v prostředí softwaru SIMIT a TIA Portal od společnosti Siemens efektivně nahradit fyzické prototypy a poskytnout studentům a odborníkům nástroj pro hlubší porozumění a experimentování bez rizika poškození zařízení nebo plýtvání materiály. Dále práce zkoumá, jak digitální dvojče umožňuje interaktivní vizualizaci a simulaci, což přispívá k lepšímu pochopení dynamiky a funkčnosti systému v reálném čase.

K realizaci celého digitálního dvojčete je zapotřebí rozklíčovat princip provozu vybraného zařízení, to zahrnuje i tvorbu logiky procesu a vymyšlení jeho řízení, programování, ověřování funkčnosti, simulaci a tvorbu grafické vizualizace. S pomocí takto zpracovaného digitálního dvojčete lze odladit řídicí program, odhadnout časový takt jednoho pracovního cyklu, a ověřit celkový princip stroje. Pro různá odvětví průmyslu mohou být tyto body klíčové.

První část bakalářské práce představuje vybraný technologický proces. Zahájena je výběrem vhodného zařízení a procesu, který bude realizován. Následně je popsán princip funkčnosti vybraného zařízení a jsou specifikovány základní komponenty nezbytné pro jeho provoz. Úvodní fáze práce obsahuje základní kroky předcházející praktickému testování, simulaci, vizualizaci, které jsou pospány v následujících částech práce.

Druhá část spočívá v detailním zhotovení vizualizace zařízení, která poskytuje uživateli příjemný pohled na dané operace a kroky. Vizualizace přináší podstatné poznatky pro efektivní tvorbu řízení a monitorování funkčnosti prostřihového zařízení. Pomocí této simulace lze zaškolit nové operátory pro práci na vybraném zařízení dříve, než se zařízení vyrobí a uvede do procesu výroby.

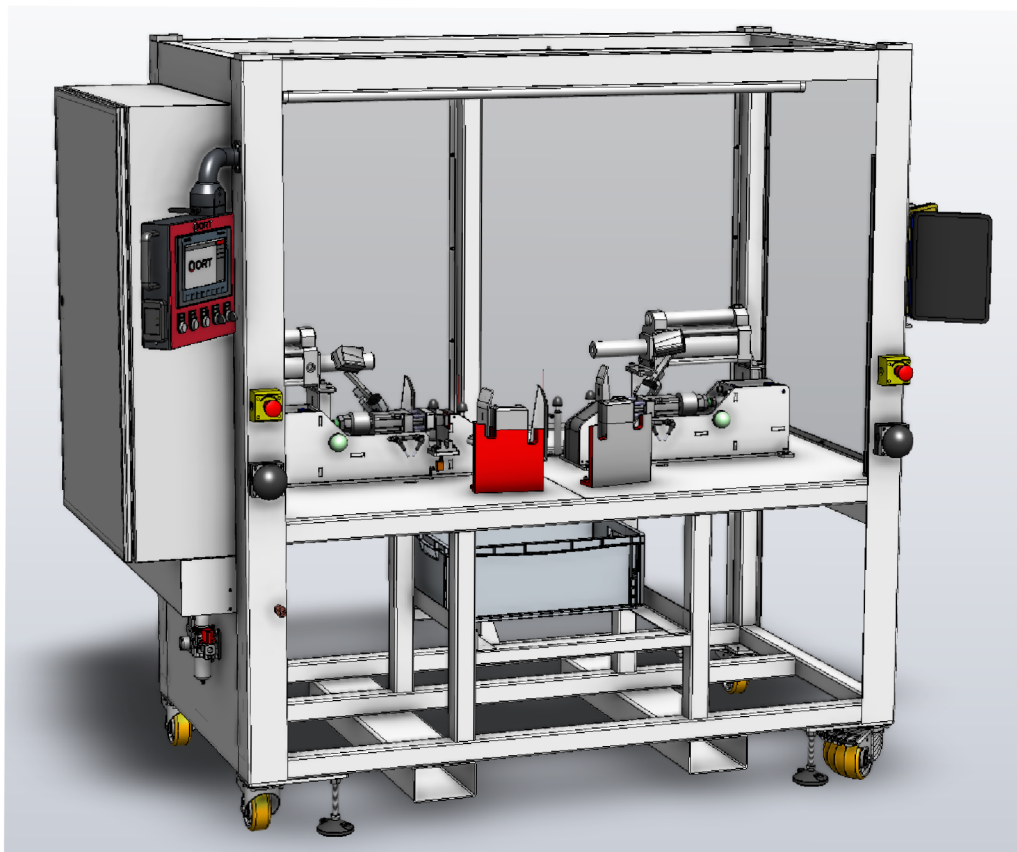
Třetí část práce je zaměřena na simulaci prostřihového zařízení. Tato část práce zahrnuje propojení simulačního prostředí SIMIT s programovacím prostředím TIA Portal. Zároveň je v této části vypracována zpětná vazba, umožňující testování a optimalizaci řídicího programu.

Závěrečná část bakalářské práce podrobně popisuje kroky potřebné pro realizaci řídicího programu prostřihového zařízení. Důležitou částí je vytvoření řídicího programu pro PLC za použití softwaru TIA Portal, který vytváří základ pro simulaci a testování funkčnosti zařízení. V této části je mimo jiné seznámení s prostředím TIA Portal a výběr konkrétního PLC. Také je zde vysvětlen jazyk Ladder Diagram, který je využit pro stavbu řídicího programu.

1 Technologický proces

1.1 Výběr zařízení

Úspěšná realizace digitálního dvojčete vyžaduje výběr zařízení a procesu, které budou simulovány. Výběr zařízení je zásadní pro dosažení požadovaných cílů práce, jelikož přímo ovlivňuje relevanci a aplikovatelnost výsledků v praxi. Pro tuto práci bylo vybráno prostřihovací zařízení určené k zpracování plastových komponent. Jedná se o stroj, který vystřihuje otvor do plastového krytu B sloupku automobilu (*Sloupek mezi předními a zadními dveřmi vozidla*). Vystřižený otvor slouží k následné instalaci krytky airbagu.



Obrázek 1.1: Prostřihovací zařízení z firmy ORT s.r.o.

Toto zařízení bylo vybráno, jelikož jeho princip je ideální pro výuku tvorby řídicího programu. Řídicí program tvořený v TIA Portalu je poměrně obsáhlý, proto tento proces může být využit například jako zadání semestrální práce, nebo čistě jako jedno z mnoha zadání ve výuce programování PLC automatů.

Inspirace pro volbu tohoto zařízení pochází z firmy ORT Nový Bydžov spol. s r.o. , ve které bylo zařízení vytvořeno. Autor se podílel na vývoji tohoto zařízení, tím pádem věděl jak celé zařízení funguje, a bylo mu tedy přívětivé, využít ho jako inspiraci. Bohužel se autor nepodílel na vývoji řídicího programu. Jakožto konstruktor se spíše podílel na vývoji rámu zařízení ze strojařského hlediska a funkčnosti jednotlivých komponent.

Digitální dvojče by při vývoji řídicího programu bylo vyhovující, jelikož termín dodání zařízení nebyl zrovna vyhovující. Programátor musel tedy poměrně dlouho čekat než se zařízení celé vyrobilo, sestavilo, zapojilo a následně až mohl testovat první verze programu. Software pro tvorbu virtuální variantu zařízení by byl rozhodně velkým přínosem. Programátor by měl možnost otestovat program ještě před tím, než se zařízení vyrobilo, tím pádem by měl více času na finální doladění, a odladil by menší části programu předčasně.

Zařízení musí splňovat několik podmínek pro úspěšný a efektivní chod. Samozřejmostí je kvalita vystřiženého otvoru a dobrá ergonomie pracoviště. To nejdůležitější je ovšem rychlý a bezchybný chod řídicího systému. Ve většině odvětví průmyslu je nutné, aby dodaná pracovní stanice byla maximálně optimalizována tak, aby se minimalizovaly jakékoliv prostoje ve výrobním procesu. Je nutné přizpůsobit operátorovi pracoviště a ovládání zařízení k co nejjednoduššímu používání, protože ideálně si zákazník přeje co nejvíce eliminovat možné chyby operátora a usnadnit mu pracovní proces. Operátor musí být schopen bezúhonně pracovat na stroji několik hodin denně v daném taktu. Je důležité si uvědomit, že při velkém počtu pracovních cyklů, může i sebemenší chyba operátorovi přinést velké obtíže.

1.2 Základní komponenty

Každé zařízení, a tím i prostřihovací stroj, se skládá z několika základních komponent, které jsou klíčové pro vykonávání operací. Pro vytvoření samotné logiky zařízení je potřeba vyhodnotit, jaké komponenty volit pro zvolený projekt. Následující sekce poskytuje podrobný přehled hlavních komponentů vybraného prostřihovacího zařízení a vysvětluje jejich úlohu v rámci celého technologického procesu.

Základní komponenty použité vytvoření digitálního dvojčete jsou uvedeny v tabulce 1.1:

Tabulka 1.1: Tabulka s komponenty

	Název	Počet	Vlastnosti
1.	Hydraulický Válec	2	senzory koncových poloh
2.	Upínka	2	válec s ramenem a senzory koncových poloh
3.	Optický senzor	2	Pro sledování přítomnosti dílu
4.	Mechanický senzor	2	Pro sledování uchycení dílu
5.	Optické brány	1+1	snímání bezpečnosti pracovního prostředí
6.	Signalizační maják	1	3-barevný pro signalizace stavu zařízení
7.	Ovládací panel	1	panel s tlačítky pro ovládání stroje

Jednotlivé komponenty lze vidět na obrázku 2.3 v kapitole 2.

1.2.1 Hydraulický válec

Hydraulické válce jsou zásadní pro pohon střížných součástí (střížníků), které vykonávají fyzické operace, jako je stříhání materiálu. V případě popisovaného zařízení jsou válce vybaveny senzory koncových poloh, které umožňují jednoduché a přesné ovládání jejich pohybu.

1.2.2 Upínka

Upínky jsou pneumatické komponenty, které zajišťují fixaci materiálu během procesu stříhání. Jsou vybaveny válcem s ramenem a senzory koncových poloh, což umožňuje automatické nastavení polohy prostřihovaného materiálu a zvyšují efektivitu a bezpečnost operace.

1.2.3 Optický a mechanický senzor

Optické senzory slouží k detekci přítomnosti materiálu v prostřihovacím stroji, zatímco mechanické senzory kontrolují jeho správné uchycení. Tato kombinace senzorů zajišťuje, že stroj pracuje pouze tehdy, když je materiál správně umístěn a uchycen, což minimalizuje riziko chyb.

1.2.4 Signalizační maják a ovládací panel

Signalizační maják je klíčový pro zobrazení stavu stroje operátorům. Jeho barvy signalizují různé provozní stavy. Ovládací panel, který obsahuje řadu tlačítek a přepínačů, umožňuje operátorům rychlou a intuitivní kontrolu nad funkcemi stroje.

- Stroj bezpečný k přístupu obsluhy (Zelená barva).
- Stroj je v chodu a hrozí nebezpečí (Oranžová barva).
- Stroj je v chybovém stavu (Červená barva).

Mimo výstražného majáku je stroj osazen podsvícenými tlačítky. Na střížných válcích jsou umístěny signalizační světla pro upozornění chodu cyklu. Taktéž jsou pro lepší diagnostiku chyb umístěna signalizační světla na senzorech. Jedno na optickém senzoru přítomnosti dílu a druhé na mechanickém senzoru dotlačení dílu do pozice.

1.3 Princip práce zařízení

Zařízení je řízené pomocí PLC. Nastavení stroje lze měnit pomocí ovládacího panelu. PLC zhodnotí všechny stavy uživatelského nastavení, zkontroluje bezpečnost a přítomnost dílu. Následně může prostříhnout dané součásti.

Na zařízení lze pracovat na levém nebo na pravém dílu separátně, nebo na obou stranách zároveň. Operátor si může volit, na které variantě dílu bude pracovat. Po odstartování cyklu nesmí být porušeny bezpečnostní optické brány.

V případě porušení je celý proces zastaven. Zastavený proces lze opětovně spustit pomocí potvrzení bezpečnosti. Zařízení také disponuje hlavním vypínačem a tlačítkem *TOTAL STOP*.

1.3.1 Přidané funkce

Pro optimálnější práci operátora bylo zkoumáno, jaké další funkce vybrané zařízení může mít. Vždy se dají najít způsoby jak optimalizovat danou pracovní operaci. Z tohoto důvodu autor implementoval možnost takzvaného *RESETU*. Tato funkce byla přidána pro více komplexní digitální dvojče. Díky této funkci lze simulované zařízení ovládat novými způsoby, které mohou být v budoucnu přínosné i pro reálné zařízení.

Na ovládacím panelu lze nalézt tlačítko *RESET*, které uvede všechny pohyblivé prvky zpět do výchozí polohy. Tlačítko *RESET* je zde umístěno pro potřebu otevření upínky nebo pro zasunutí střížného válce po zastavení stroje. *RESET* funguje za jakéhokoliv stavu až na moment stlačeného bezpečnostního tlačítka *TOTAL STOP*. V takové situaci je potřeba *TOTAL STOP* vytáhnout (vrátit do výchozí pozice).

V zastavení, ať už kvůli porušení bezpečnosti, nebo kvůli stlačení tlačítka *TOTAL STOP*, se zařízení zastaví v momentální pozici a nepohne se, dokud operátor neudá další úkon. Operátor takovou situaci řeší dvěma způsoby. Lze vrátit vše do výchozí pozice pomocí *RESET* a nebo pokračovat v procesu potvrzovacím tlačítkem.

1.4 Schéma logiky programu

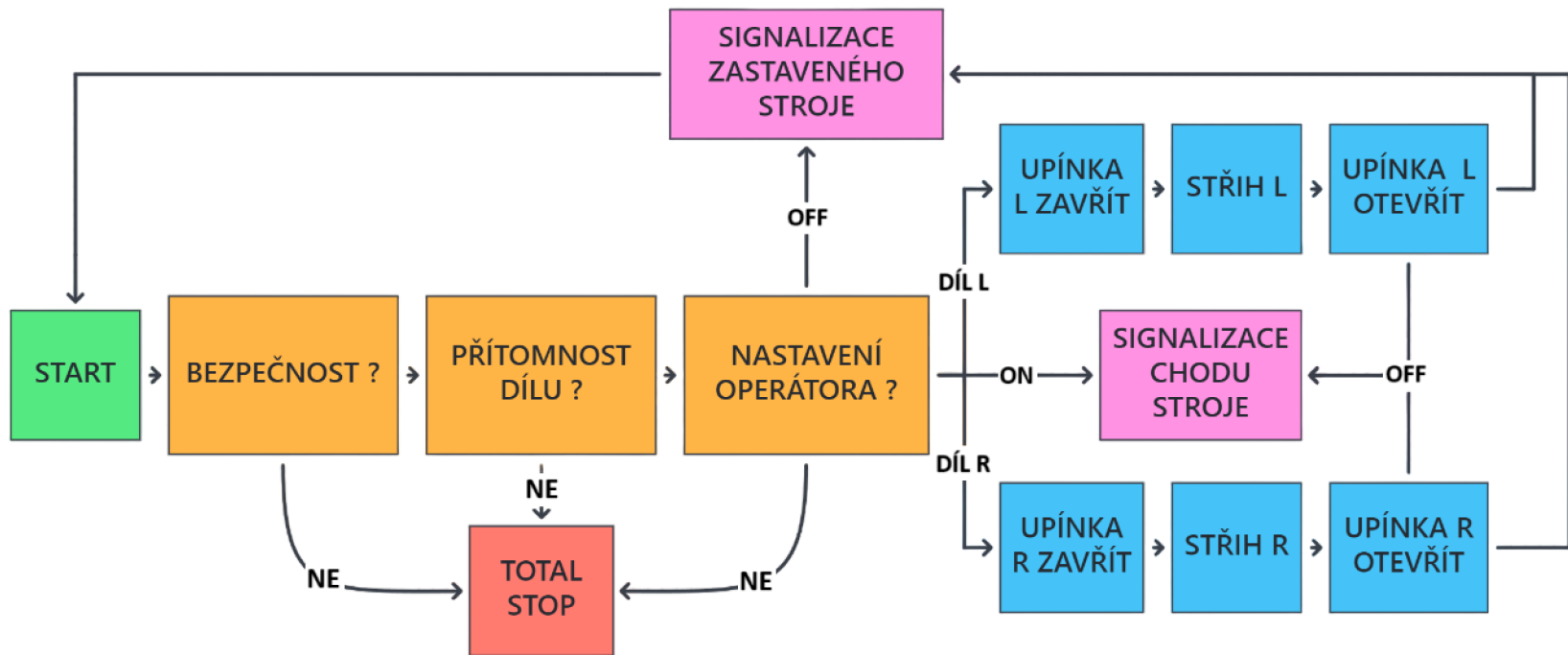
Schéma logiky programu představuje základní strukturu řídicího softwaru, který řídí celkovou funkčnost prostřihovacího zařízení. Toto schéma je zásadní pro pochopení operací stroje a pro efektivní tvorbu programu. V této podkapitole je popsáno, jak jsou jednotlivé základní komponenty programu propojeny a jakým způsobem řídí fyzické procesy stroje.

Řídicí program a vizualizace simulovaného zařízení je třeba nejdříve hrubě navrhnout. Je třeba vymyslet základní průběh práce zařízení a určit závislosti mezi jednotlivými akcemi. Tato rozvaha je základem pro tvorbu řídicího programu.

Pro rozvržení řídicího programu je klíčové definovat hlavní stavy, ve kterých se může zařízení nacházet a události, které tyto stavy mění. Tyto stavy zahrnují inicializaci systému, běžný provoz, zastavení systému a signalizaci zařízení. Události, které mění tyto stavy, jsou identifikovány jako signály od senzorů nebo uživatelské vstupy z ovládacího panelu.

Následující zjednodušené schéma je vytvořeno pro interpretaci postupu simulovaného řízení. Schéma vyzobrazené na obrázku 1.2 slouží jako výchozí předloha pro prvotní návrh řídicího programu.

Obrázek 1.2: Schéma řídicího programu



Struktura postupu řízení bude vypadat následovně:

Tabulka 1.2: Tabulka postupu řízení

	Akce	Podmínka	Výsledek
1.	Zapnutí stroje	Operátor zapne	Hlavní vypínač-OK
2.	Kontrola TOTAL STOP	Operátor zkontroluje	TOTAL STOP-OK
3.	Výběr Left/Right/L+R	Operátor nastaví	Informace, které lože bude stříhat
4.	Založení dílu operátorem	Operátor vloží	Přítomnost dílu
5.	Bezpečnost	Nepřerušené optické brány	Bezpečnost-OK
6.	START	Operátor stlačí	Zapnutí cyklu
7.	Uzavření upínek	-Cyklus zapnut -Bezpečnost OK -Přítomnost dílu	-Upínka v dolní pozici -Signalizační maják svítí žlutou barvou -mechanický senzor stlačen
8.	Vysunutí střížných válců	Stejně jako předchozí -Upínka dole -mechanický senzor stlačen	Vysunutí střížného válce -Válec v konečné pozici
9.	Zasunutí střížných válců	Stejně jako předchozí -Válec v konečné pozici	zasunutí střížného válce -Válec ve výchozí pozici
10.	Otevření upínek	Stejně jako předchozí -Válec ve výchozí pozici	Upínky otevřeny
11.	Vyndání dílu operátorem	Operátor Vyjme	Konec Cyklu

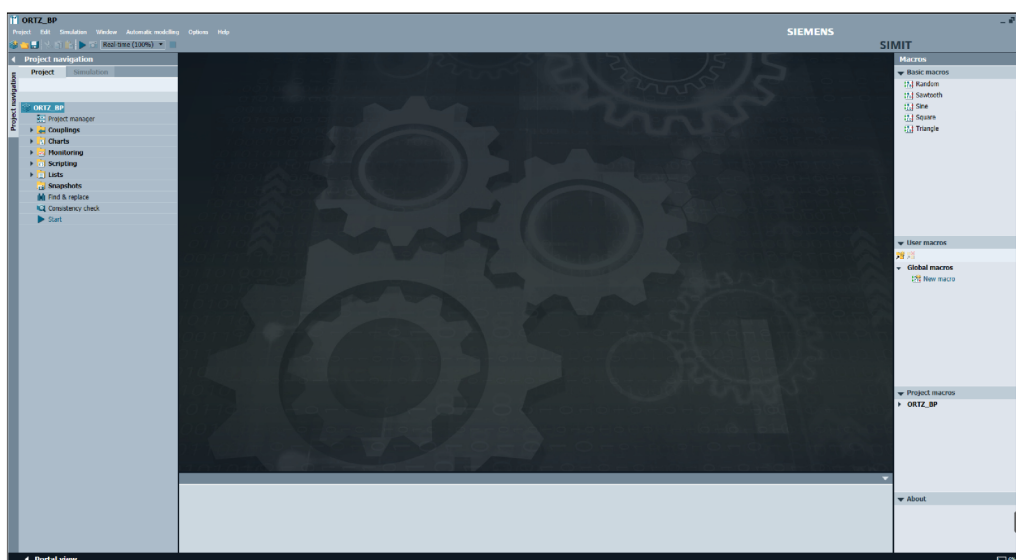
2 Vizualizace

Vizualizace je klíčovou součástí digitálního dvojčete, neboť poskytuje uživatelům grafické zobrazení procesů, které by jinak byly abstraktní a méně srozumitelné. Tato kapitola popisuje, jak jsou vizualizace vytvářeny, jaké postupy jsou využívány k její animaci a jaký význam mají pro efektivní ovládání prostřihovacího zařízení.

Ideální software pro vytvoření vizualizace a simulace je již zmiňované prostředí SIMIT. V této kapitole je také vysvětleno, jak s prostředím SIMIT pracovat a k čemu slouží. Kapitola také popisuje, jak propojit SIMIT s prostředím TIA Portal.

2.1 SIMIT

SIMIT Simulation Platform (SIMIT SP) je software od společnosti Siemens, který se používá pro simulaci a testování automatizačních systémů. Tento nástroj umožňuje uživatelům vytvářet realistické virtuální modely průmyslových systémů, což pomáhá v optimalizaci procesů, školení operátorů a vývoji software. SIMIT poskytuje prostředí, ve kterém lze testovat a ověřovat ovládání strojů a procesů před jejich skutečným nasazením v terénu, což může výrazně snížit riziko a náklady spojené s reálným spuštěním systémů [5].



Obrázek 2.1: Prostředí programu SIMIT

V tomto prostředí je možné také tvořit vizualizace simulací ve 3D formátu, avšak pro účely této práce není tato funkcionality nezbytná.

Základní prvky pro vytvoření projektu se dají rozdělit na hlavní tři části:

2.1.1 Couplings - propojovací oddíl

Couplings v softwaru SIMIT slouží k propojení simulačního prostředí s reálnými systémy, jako jsou PLC, HMI a dalšími zařízeními v automatizovaných systémech. Tyto spojení umožňují přenos signálů mezi simulačním modelem v SIMIT a skutečným zařízením v reálném světě [2].

Pomocí tohoto oddílu můžeme také komunikovat s virtuálním PLC. Tak se dá například propojit TIA Portal a SIMIT. Po propojení lze importovat všechny proměnné použité v řízení [2].

2.1.2 Visualization - vizualizační oddíl

Vizualizace je interpretována pomocí grafických, ovládacích a informačních prvků. Umožňují zobrazovat hodnoty signálů a jejich stavy v simulaci. Pro přehlednou vizualizaci se přednostně používají animace jednotlivých grafických objektů. Pomocí těchto objektů lze vytvořit 2D pohyblivé a interaktivní prostředí [2].

Pro zobrazení a interakci s prostředím se používají například komponenty jako čáry, čtverce, elipsy, křivky, tlačítka, posuvníky nebo přepínače. Animace mohou být využity ať už pro pohyb v prostoru tak i pro změnu vzhledu grafických objektů [2].

2.1.3 Chart - simulační oddíl

V této části se vytváří chování simulace a také například zpětná vazba do řídicího programu v prostředí TIA Portal. Pro vývoj simulace se kombinují komponenty dostupné v knihovnách. Vytváří se schéma s předdefinovanými komponentami a ovládacími prvky [2].

Můžeme zde vkládat prvky jako jsou:

- Vstupy a Výstupy
- Logické operátory (*AND*, *OR*, ...)
- Převodníky signálů analog/integer
- Generátory rampových signálů

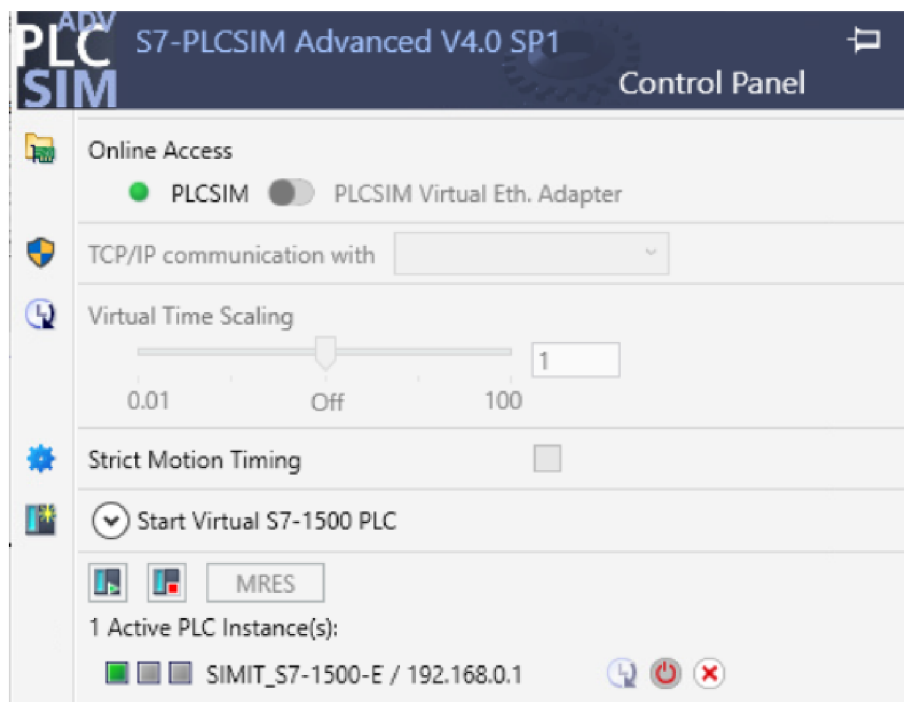
2.1.4 Propojení prostředí SIMIT a TIA portal

Pro vytvoření testovacího modelu je potřeba implementovat řídicí program do simulace v prostředí SIMIT. Toho lze docílit pomocí programu S7-PLCSIM Advanced V4.0 SP1.

PLCSIM Advanced dovoluje uživatelům vytvářet virtuální PLC a tím simulovat jakýkoliv proces bez nutnosti vlastnit fyzický hardware. Podporuje také simulaci více PLC systémů najednou [4].

Nejdříve je nutné propojit pomocí oddílu Couplings jednotlivé projekty. Tím získáme aktuální proměnné (*Tagy*) z řídicího programu. Po zapnutí simulace v prostředí SIMIT se vytvoří virtuální PLC a to se automaticky zobrazí v software PLCSIM. Následně se na toto PLC můžeme připojit z TIA Portalu.

Propojení je velmi spolehlivé. Díky tomuto spojení lze sledovat chování jednotlivých proměnných v průběhu programu. Dokud nevypneme simulaci v prostředí SIMIT, jsou vůči sobě programy propojeny. V momentě změny programu stačí pouze být v takzvaném *Go Online* stavu. Následně se dá do PLC nahrát nový řídicí program a otestovat ho v SIMITu. Prostředí SIMIT má také možnost sledovat průběh programu ve zpomaleném nebo zrychleném čase.



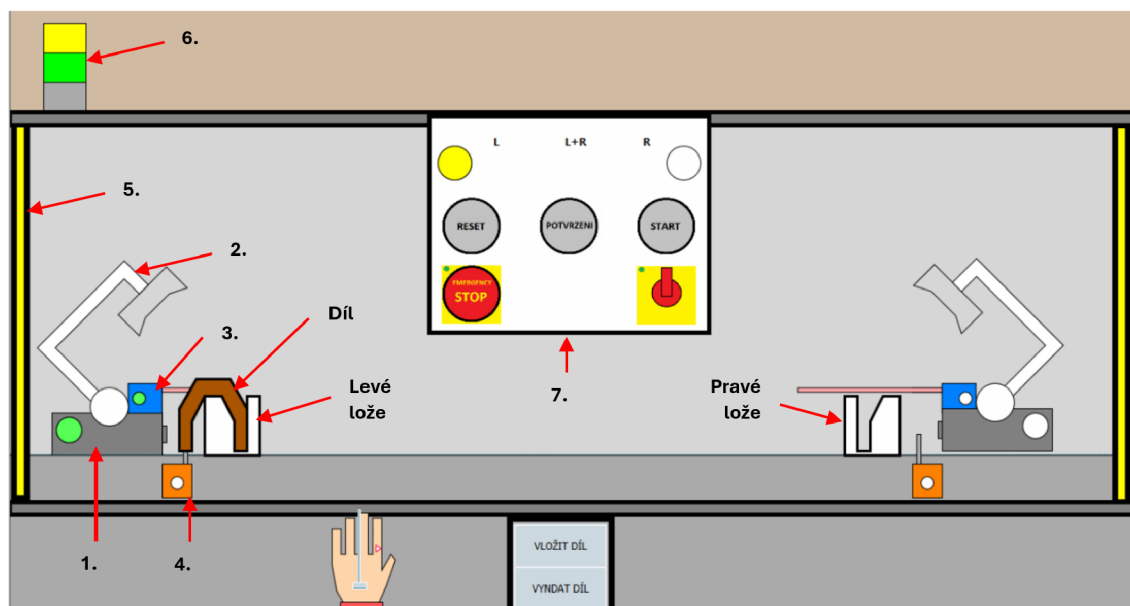
Obrázek 2.2: Prostředí PLCSIM Advanced

2.2 Návrh vizualizace

Vytvoření vizualizace má za cíl zobrazit pohyb a fungování stroje pomocí různých komponent, které byly již zmíněny v tabulce 1.1. Mezi tyto komponenty patří střížné válce s upínacími rameny, ovládací panel s tlačítky, signalizační maják, optické a mechanické senzory a další podobné komponenty.

Celá vizualizace je sestavena z jednoduchých základních tvarů, které jsou následně animovány, aby simulovaly pohyb a interakci jednotlivých komponent. Například střížné válce mohou být animovány tak, aby se vysouvaly a zasouvaly, ovládací tlačítka mohou měnit svůj stav podle interakce uživatele a signalizační maják se může rozsvěcet a zhasínat.

Na obrázku 2.3 lze vidět celé zařízení. Nachází se zde levé a pravé zakládací lože zobrazené bílou barvou. Na levém loži lze vidět založený hnědý díl, připravený na prostříhnutí. Těsně vedle zakládacích loží lze vidět střížné válce *POZ. 1.*, ty jsou zobrazeny šedou barvou, mají na sobě zelený maják a je na nich umístěné rameno upínky *POZ. 2.* a optický senzor *POZ. 3.*. Dole pod dílem je k vidění oranžový mechanický spínač *POZ. 4.*. Po obou stranách lze vidět žlutě zobrazené optické závory *POZ.5.*. Na vršku zařízení lze vidět trojbarevný signalizační maják *POZ. 6.* a přímo uprostřed zařízení lze vidět ovládací panel *POZ. 7.*.



Obrázek 2.3: Vizualizace prostříhového zařízení

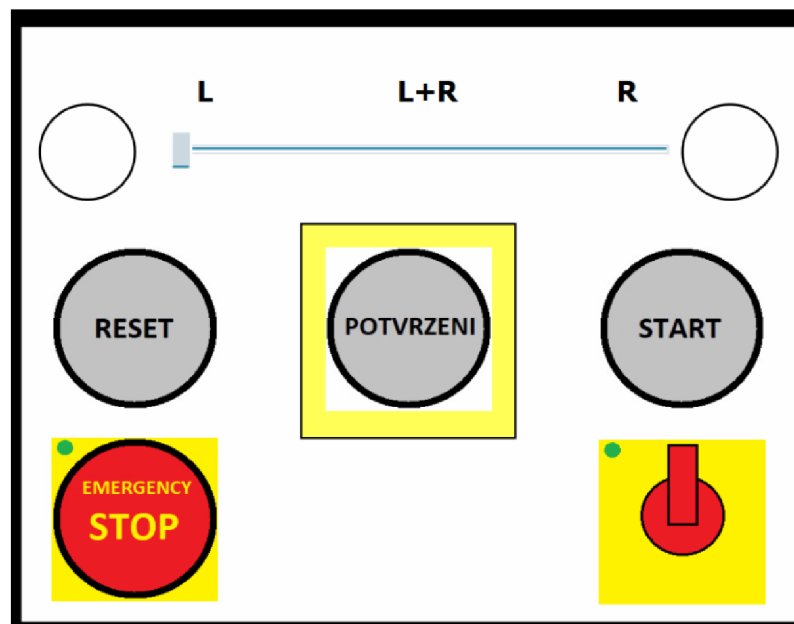
Kromě toho jsou v rámci vizualizace k dispozici také tlačítka, která mají vlastní grafický vzhled, aby co nejvíce odpovídala skutečným fyzickým tlačítkům. Pro efektivní zobrazování stavů a událostí jsou některé tlačítka ovládacího panelu nastaveny tak, že se rozsvěcí nebo zhasínají v závislosti na jejich stavu. Blikání je realizováno buď změnou zdrojového obrázku daného tlačítka nebo překrytím jiným objektem s odpovídající barvou. Blikání přispívá k realističnosti vizualizace a k její intuitivnosti používání.

2.2.1 Ovládací panel

Design ovládacího panelu je optimalizován pro pohodlnou ergonomii a intuitivní ovládání. Panel je navržen s ohledem na přehlednost a umístěn tak, aby byl snadno přístupný. Ovládací panel je umístěn uprostřed zařízení, aby byl vždy na dosah operátorovi.

Jsou zde umístěna ovládací tlačítka, kterými operátor řídí zařízení. V horní části obrázku 2.4 lze vidět posuvník pro vybírání levého nebo pravého dílu. Níže jsou umístěna tlačítka pro ovládání základních funkcí stroje (*RESET*, *POTVRZENÍ*, *START*). Pod nimi je hlavní vypínač a *EMERGENCY STOP*.

Tlačítko potvrzení má dokola žlutou signalizaci, která bliká za podmínky, že je potřeba jej stlačit pro pokračování v chodu stroje.



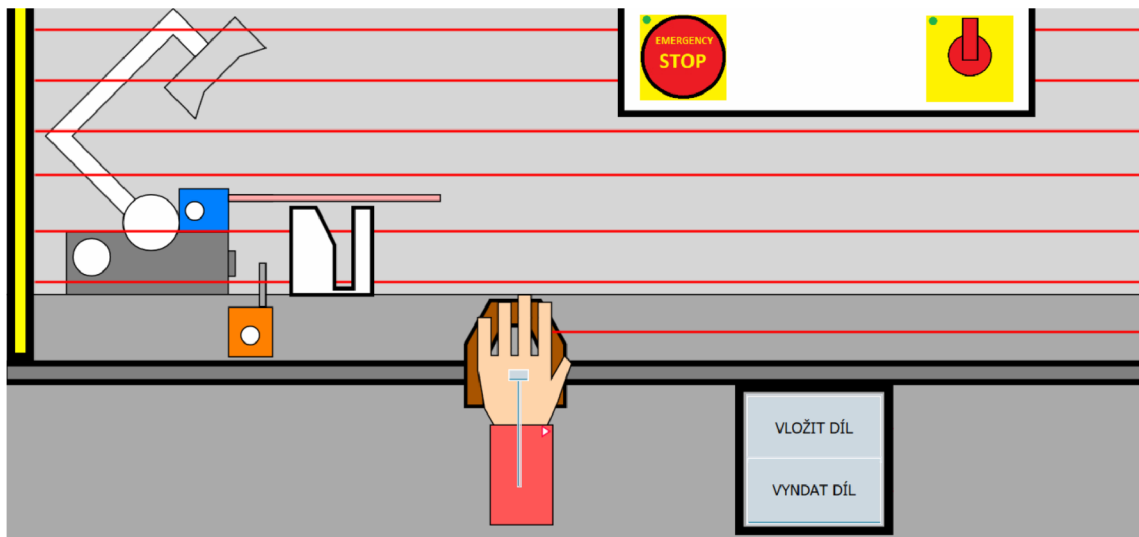
Obrázek 2.4: Ovládací panel

2.2.2 Založení dílu

Aby vizualizace a simulace byla plně autentická, je zde vytvořena funkce vlastnoručního zakládání dílů. Díky této funkci dokážeme ještě lépe simulovat proces operátora u zařízení. Na této funkci se také názorně zobrazuje narušení bezpečnosti.

Založení dílu se provede pomocí přetažení operátorovy ruky do pracovního prostoru. Následně lze vložit nebo vyjmout díl pomocí příslušného tlačítka na pravé straně operátorovy ruky. Společně s vybraným režimem L/L+R/R, který je zvolen na ovládacím panelu, se spustí animace, při které se díl přesune z ruky operátora do zakládacího lože. Při zakládání dílu je vizualizovaný paprsek od optického čidla zkrácen, aby bylo poznat, že přítomnost dílu byla zaznamenána.

Lze si také povšimnout zobrazených laserových paprsků optické brány. Ty jsou zde úmyslně znázorněny, aby bylo ze simulace zřejmé, že je narušena bezpečnost v pracovním prostoru. Při přetažení ruky z pracovního prostoru tyto paprsky zmizí.



Obrázek 2.5: Založení dílu

3 Simulace zařízení v prostředí SIMIT

Následující kapitola je zaměřena na vytvoření simulace prostřihového zařízení. Je zde rovněž vysvětlena logika zpětné vazby, kterou vytváříme jako odezvu zařízení pro řídicí systém realizovaný v prostředí TIA Portal.

Simulace je vytvořena především s cílem poskytnout zpětnou vazbu do TIA Portalu, ale zahrnuje i důležité prvky umožňující přehlednější a estetičtější animaci vizualizace.

3.1 Simulační prvky

3.1.1 Generátor signálu Ramp

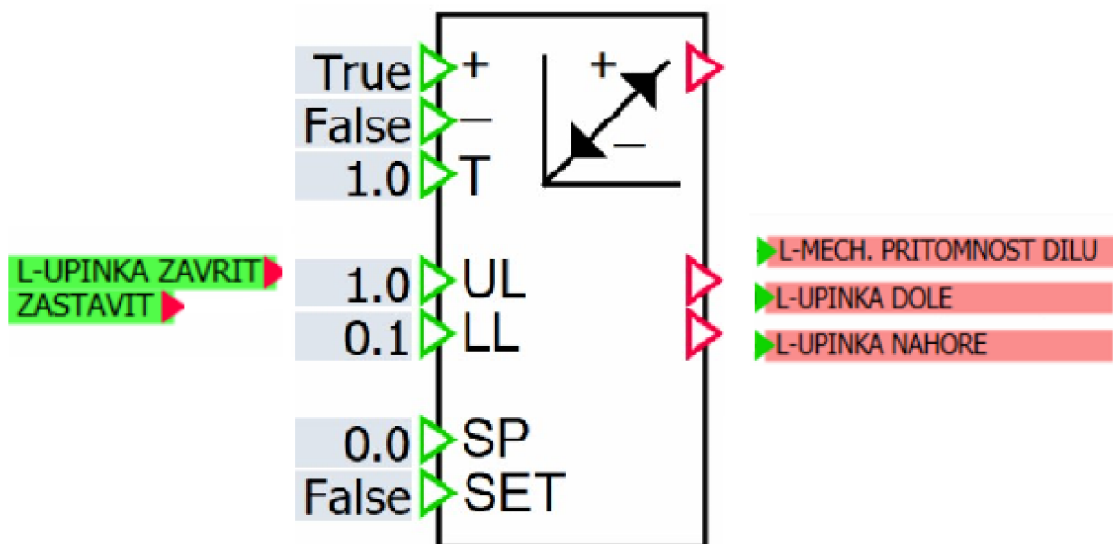
Jedním z často používaných prvků je takzvaná rampa, která slouží jako generátor ramповého signálu. Rampa postupně zvyšuje nebo snižuje svou funkční hodnotu v průběhu simulace. K tomu používá vstupní signály, které rozhodují, zda má zvýšit (*+UP*) nebo snížit (*-DOWN*) hodnotu rampy. Pokud jsou oba tyto signály aktivní současně, hodnota rampy zůstane nezměněna. Tyto signály jsou typu BOOL.

Je důležité poznamenat, že hodnota rampy je omezena na určený interval, který je definován horní a dolní mezí. Pokud hodnota rampy dosáhne některé z těchto mezních hodnot, je to signalizováno pomocí odpovídajících binárních výstupů.

Rampa má další nastavitelné parametry, jako je počáteční hodnota, časová konstanta a mezní hodnoty, které lze nastavit podle potřeby. Lze také přímo nastavit momentální hodnotu rampy na vlastní hodnotu pomocí signálu *SP* a potvrzovacího příkazu *SET* [2].

3.1.2 Vstupy a výstupy simulace

Pro ovládání prvků, jako je například *Ramp*, jsou zapotřebí hodnoty daných podmínek. Tyto hodnoty jsou zakomponovány do simulace pomocí vstupů. Výstupní signály jsou buď využity k animacím vizualizace nebo pro zpětnou vazbu do TIA Portalu. K výstupu zpět do řídicího programu jsou využívány výstupní prvky, které jsou popsány na obrázku.3.1

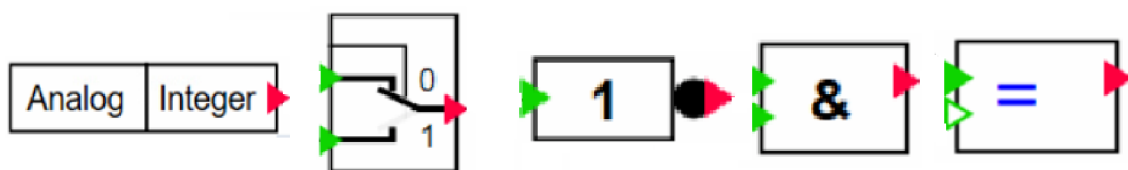


Obrázek 3.1: Generátor signálu Ramp se vstupy a výstupy

3.1.3 Logické prvky

Řídící signály jsou často kombinovány s dalšími typy signálů, aby bylo možné vytvořit specifické podmínky nezbytné pro správný chod zařízení. Kombinace různých signálů umožňuje realizaci komplexních logických operací a tím udávají podmínky výstupům. Signály taktéž mohou být upravovány a měněny na jiné typy signálů, pro správné ovládání různých bloků.

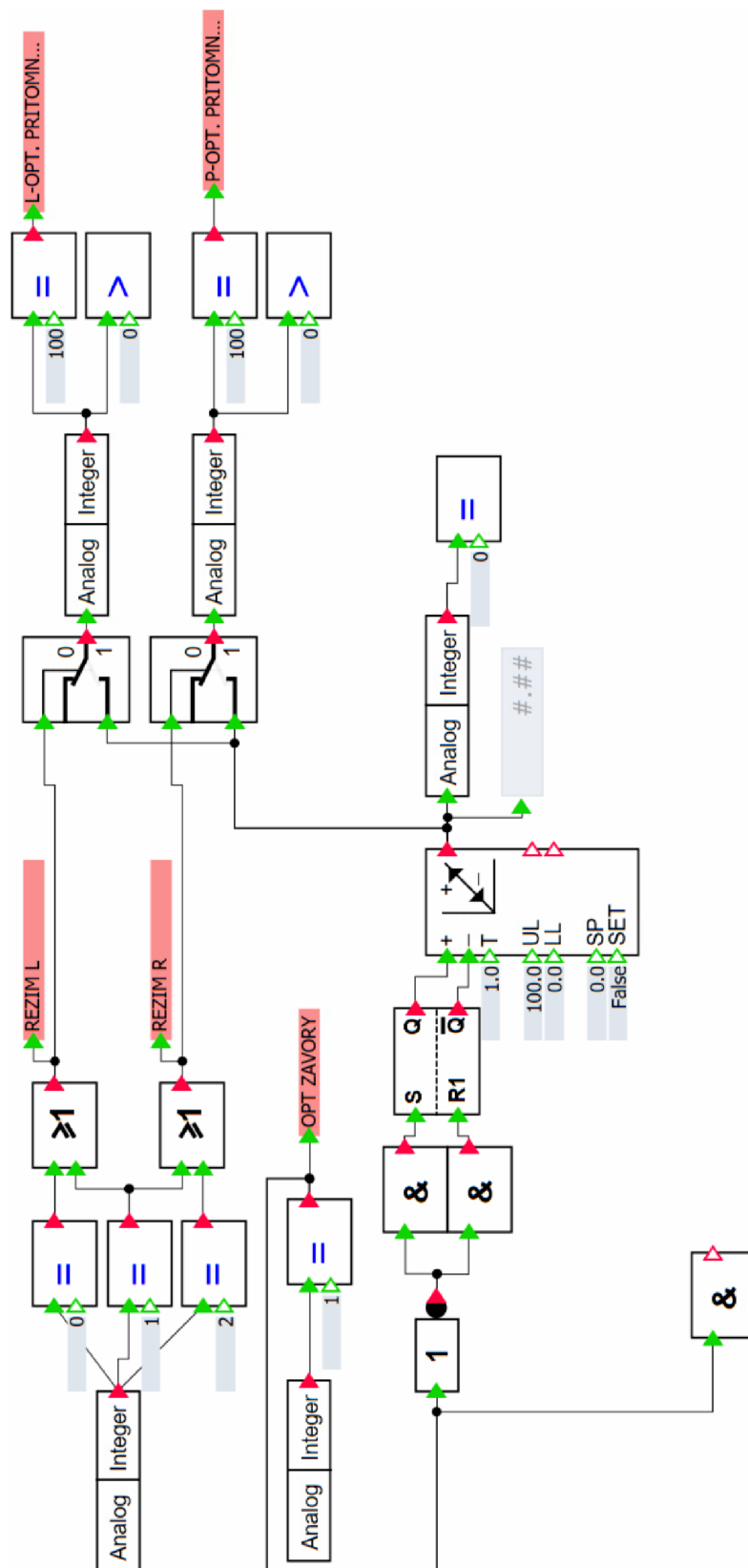
K těmto kombinacím a změnám slouží velká řada logických prvků. Tyto prvky se používají běžně ve velkém množství u logických operací. Mohou to být například Komparátory, převodníky typů hodnot, *AND*, *OR*, *SWITCH CASE*, *Negace*, a mnohé další. Příklady těchto prvků lze vidět na obrázku 3.2



Obrázek 3.2: Logické prvky

3.2 Výběr Pravého nebo levého režimu

Pomocí posuvníku na ovládacím panelu může operátor vybírat mezi třemi stavy. Levý díl, pravý nebo oba dva. Tento posuvník vrací do simulace hodnoty 0, 1, 2. podle těchto hodnot je v simulaci rozlišováno kam, bude založen díl. Taktéž se tato informace odesílá do řídicího programu. Následně umístění dílu aktivuje optický senzor. Tím je zajištěna přítomnost dílu.



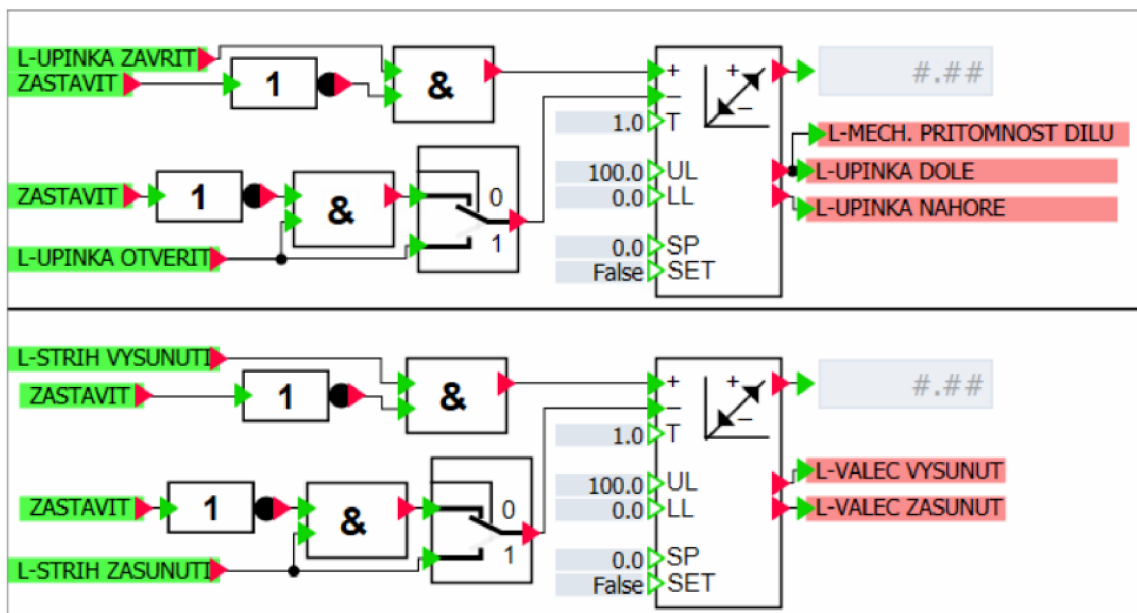
Obrázek 3.3: Výběr režimu Levé / Pravé lože

3.3 Zpětná vazba

Pro zpětnou vazbu byly výstupy z řídicího programu použity jako podmínky do RAMP bloků. Na obrázku 3.4 jsou znázorněny negace a kombinace signálů pro vytvoření správných podmínek. Ramp bloky zde vytvářejí postupně měnící se signál pro ladný průběh vizualizace. Pomocí koncových poloh Ramp bloků je vytvořena zpětná vazba. V momentě kdy *Ramp* blok narazí na jednu ze svých hranic, vyšle se signál o dosažení koncové polohy. To znamená například, že se upínka dostala do otevřené polohy. Dosažení koncové polohy je důležitá informace pro řídicí systém v TIA Portalu, protože jinak by řídicí program nevěděl v jaké pozici upínka je.

Na obrázku 3.4 jsou zobrazeny dvě sekce. Jedna sekce vypovídá o stavu a poloze levé upínky. Společně s upínkami přichází i informace ohledně upnutí dílu. Druhá sekce vypovídá o vysunutí a zasunutí levého střížného válce. Primárně se těmito zpětnými vazbami řídí střížné cykly. Logika sekce pro ovládání upínek a střížných válců pravého lože je totožná jako pro levé lože.

Výstup z Ramp bloků je zde použit jako podmínka animací. Postupná změna hodnoty vytváří plynulý pohyb střížníků a nebo rotaci upínacího ramene.



Obrázek 3.4: Zpětná vazba a podklady pro vizualizaci

4 Návrh řídicího programu

4.1 Ladder diagram

Řídicí program je napsaný v jazyce LAD (Ladder diagram), který simuluje schéma reléových obvodů. Volba jazyka LAD byla pro autora příhodnou i z důvodu, že je s ním dobře seznámen.

LAD je vizuální programovací jazyk, využívající symboly podobné elektrickým schémátům. Základ LAD jazyka je příčka spojení. Tu si lze představit jako vodič přenášející hodnotu zleva doprava. Na této lince jsou umístěny prvky které ovlivňují jestli signál projde dále nebo ne. [1].

4.1.1 Kontakty a cívky

Jeho základní prvky jsou takzvané kontakty a cívky. Kontakty jsou aktivovány vnějšími signály a slouží k otevírání nebo uzavírání obvodu. S pomocí těchto kontaktů dokážeme vytvářet podmínky pro výstupy[1]. Kontakty mohou být dvou základních variant:

- **NO** (*Normaly open*) - Kontakt se po aktivování podmínky sepne a propustí signál dál[1].
- **NC** (*Normaly closed*) - Kontakt se po aktivování podmínky rozepne a dál nic nepropustí[1].

Můžeme kombinovat signály sériově či paralelně. Sériově zapojené kontakty (*za sebou*) propustí signál dále, pouze pod podmínkou že budou všechny kontakty přepnuty do stavu propuštění. Funkce sériově zapojených kontaktů se dá přirovnat logickému operátoru *AND*. Zatím co paralelně zapojené kontakty fungují jako logický operátor *OR*, tudíž pokud je-li jeden z kontaktů sepnutý, signál se propustí dále, bez ohledu na další kontakt paralelně zapojen[1]. Kontakty a cívku lze vidět na obrázku 4.1

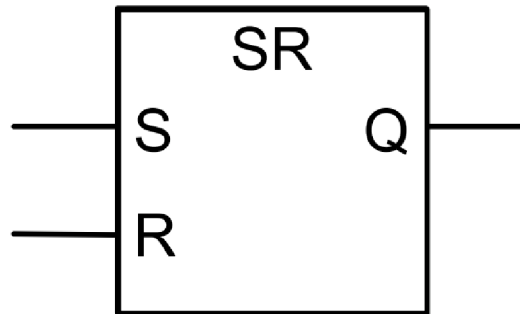


Obrázek 4.1: Kontakt NO, NC a cívka

4.1.2 SR a RS - Flip Flop

Ve velké míře je v LAD využíváno SR klopných obvodů. Funkce tohoto bloku je podobná jednoduchému přepínači.

Pokud na vstupním konektoru *S* (*SET*) nastane stav signálu logické *1*, obvod SR bude držet dále stav *1* dokud nebude vyresetován a to právě logickou *1* na konektoru *R* (*RESET*). Stav konektoru *S* má přednost před konektorem *R*. Výsledný signál obvodu je zobrazen na konektoru *Q* [1]. SR klopný obvod lze vidět na obrázku 4.2



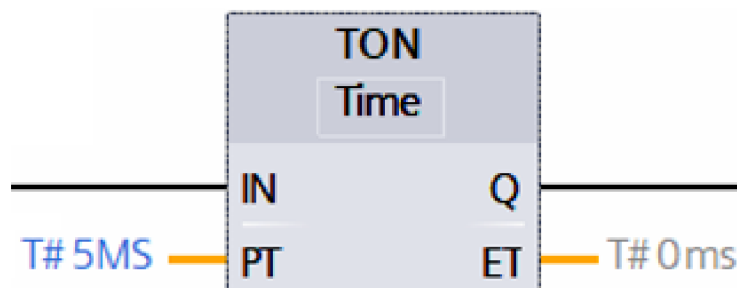
Obrázek 4.2: Blok SR obvodu. Obrázek použit ze zdroje [1]

Blok RS (RESET-SET Flip Flop) funguje přímo opačnou logikou oproti SR.

4.1.3 TON časovač

Bez časových rozdílů některých signálů by se většina programů neobešla. Proto existuje časovací blok TON (*Timer ON*) jinak řečeno zpožďovač signálu.

Blok má vstupní kontakty *IN*, *PT*, *Q* a *ET*. Na konektor *IN* se připojí signál který chceme zpozdít, na konektoru *PT* nastavíme hodnotu zpoždění s danou jednotkou (většinou milisekundy) a na výstupu *Q* se nám objeví signál zpožděný o čas hodnoty *PT* od momentu kdy se na *IN* objeví *1*. S vypnutím vstupního signálu se ihned vypne i signál výstupní. Konektor *ET* lze použít k detekci konce zpoždění časovače. TON časovač lze vidět na obrázku 4.3



Obrázek 4.3: TON časovač

4.2 TIA Portal

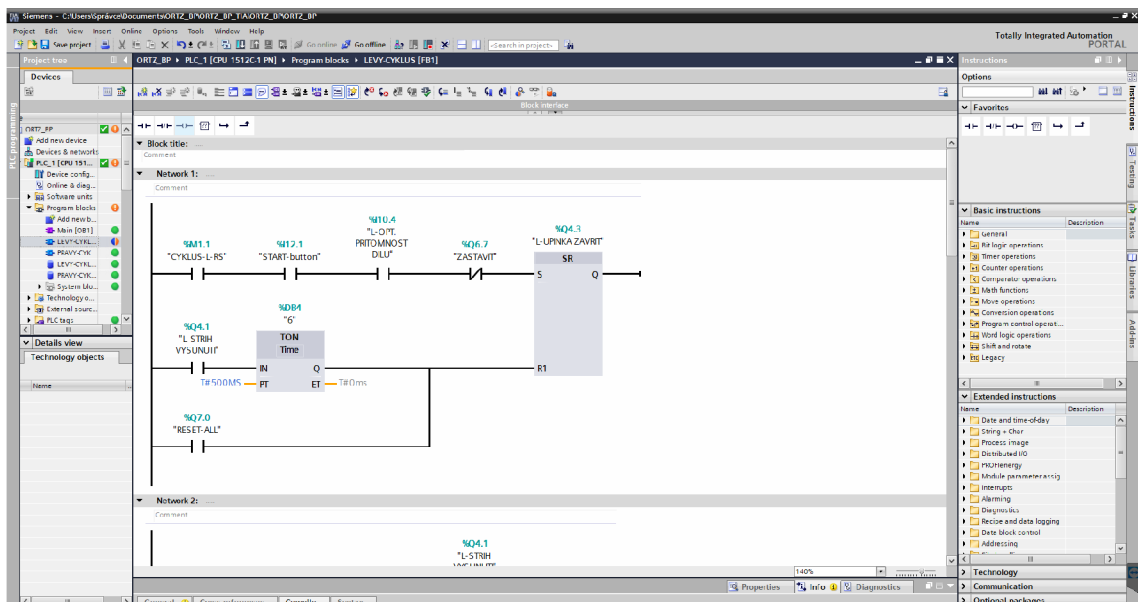
Tato kapitola seznamuje s prostředím TIA portal. TIA Portal V17 (Totally Integrated Automation Portal - verze 17) je inovativní softwarové řešení od společnosti Siemens, které slouží k vytváření projektů pro automatizační techniku. Je to univerzální nástroj, který umožňuje komplexní práci s PLC, HMI a dalšími komponenty automatizace. Díky jeho flexibilním možnostem a uživatelsky přívětivému rozhraní je ideálním nástrojem pro vývoj a simulaci řídicích systémů v rámci projektů digitálních dvojčat.

Tento software byl využit pro tvorbu programu pro řízení PLC, protože umožňuje programování, konfigurace různých zařízení a přímé sledování chodu programu řízení PLC. Tento software byl vybrán na doporučení vedoucího práce pana Ing. Petra Školníka, Ph.D. Výběr TIA Portalu je jednoznačná volba z mnoha důvodů:

- Jeden z nejpoužívanějších programů na programování PLC automatů.
- Klíčový prvek k programování PLC od společnosti Siemens.
- Jednoduchá komunikace mezi TIA Portalem a SIMATIC prostředím jelikož oba programy jsou vytvořeny společností Siemens.

Software TIA Portal nabízí také jiné programovací jazyky než již zmiňovaný LAD, jako např. SCL, SFC, FBD a STL. [6].

Náhled do tohoto prostředí lze vidět na obrázku 4.4.



Obrázek 4.4: Prostředí TIA Portal

4.2.1 Výběr PLC

Výběr PLC může být obtížné, z důvodu nutnosti zohlednění mnoha faktorů. Hlavními faktory vstupující do rozhodování jsou:

- **Počet vstupů, výstupů a modularita**

Počet a typ vstupních a výstupních modulů, které jsou potřebné pro správu signálů od senzorů, aktuátorů atd. U PLC lze většinou vyměnit celé moduly nebo karty pro jiné vstupy a výstupy.

- **Komunikace**

PLC mohou ovládat různé komunikační protokoly, díky kterým mohou nadále komunikovat s dalšími zařízeními.

- **Náklady**

Na většinu projektů stačí levnější verze PLC. Některé projekty nepotřebují ani speciální komunikace. Na těchto prvcích se dají ušetřit finanční prostředky.

Pro tuto implementaci je ideální volbou model S7-1500, konkrétně CPU 1512C-1 PN, protože poskytuje dostatek vstupních a výstupních portů. Tím je eliminována možnost, že by se v budoucnosti musely přidávat další moduly kvůli nedostatku portů pro nově přidané proměnné. Tato specifická verze je kompaktní a má pevně stanovený počet vstupů a výstupů [3].

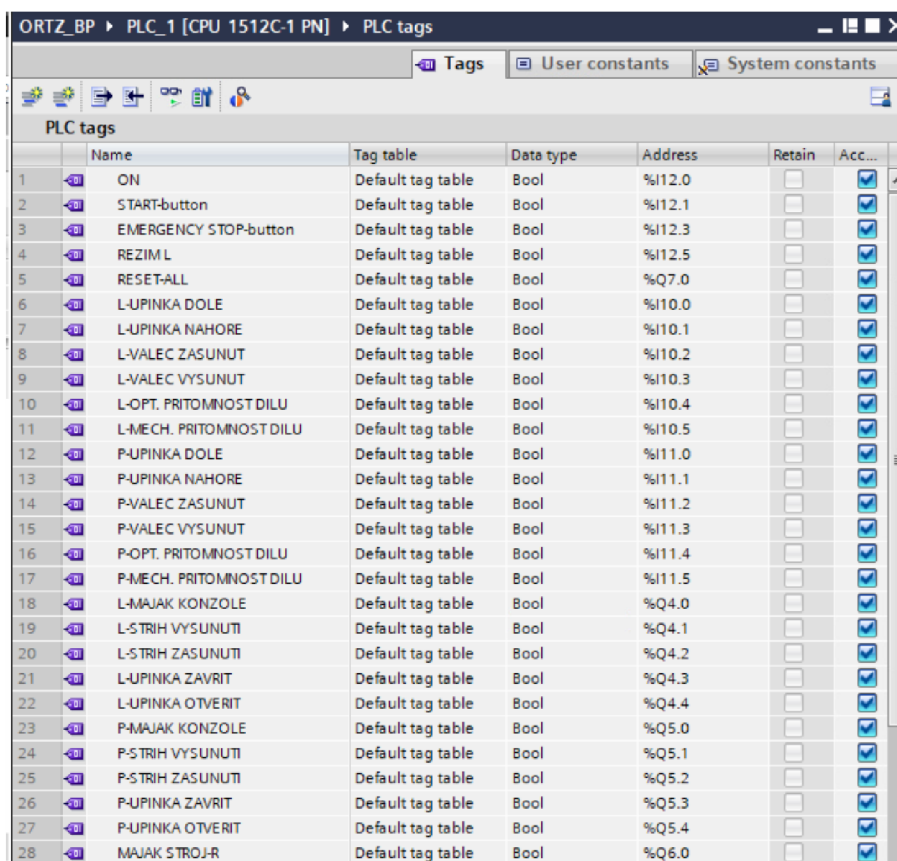
Volba modelu CPU 1512C-1 PN přináší řadu technických výhod. Jeho kompaktní design eliminuje potřebu dalších rozšiřujících modulů, čímž se snižují náklady a zjednodušuje instalace, jelikož veškeré klíčové komponenty jsou integrovány do jednoho zařízení.

Dále, model CPU 1512C-1 PN podporuje komunikaci přes PROFINET, čímž se otevírá možnost implementace tohoto zařízení do interního systému, společnosti ve které by takovéto zařízení mohlo působit. Prostředí TIA Portal je kompletně přispůsobené tomuto modelu PLC [3].

4.2.2 Proměnné - Tagy

Před samotnou skladbou programu v LAD je vhodné, vytvořit rozmístění vstupů a výstupů na PLC. Určit jaká skupina vstupů a výstupů bude sloužit k ovládání daného prvku a aby nebyly porty vedle sebe obsazené různými proměnnými. Po rozvržení portů jsou zadefinovány základní proměnné, které jsou v programu používány. Jelikož princip programu je již zhruba znám, lze si předdefinovat většinu základních Tagů. Díky této definici je skládání programu snazší a přehlednější.

Příklad přehledného rozmístění proměnných je vidět na obrázku 4.5.



	Name	Tag table	Data type	Address	Retain	Acc...
1	ON	Default tag table	Bool	%I2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	START-button	Default tag table	Bool	%I2.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3	EMERGENCY STOP-button	Default tag table	Bool	%I2.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4	REZIM L	Default tag table	Bool	%I2.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>
5	RESET-ALL	Default tag table	Bool	%Q7.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
6	L-UPINKA DOLE	Default tag table	Bool	%I10.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
7	L-UPINKA NAHORE	Default tag table	Bool	%I10.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
8	L-VALEC ZASUNUT	Default tag table	Bool	%I10.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
9	L-VALEC VYSUNUT	Default tag table	Bool	%I10.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>
10	L-OPT. PRITOMNOST DILU	Default tag table	Bool	%I10.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>
11	L-MECH. PRITOMNOST DILU	Default tag table	Bool	%I10.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>
12	P-UPINKA DOLE	Default tag table	Bool	%I11.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
13	P-UPINKA NAHORE	Default tag table	Bool	%I11.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
14	P-VALEC ZASUNUT	Default tag table	Bool	%I11.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
15	P-VALEC VYSUNUT	Default tag table	Bool	%I11.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>
16	P-OPT. PRITOMNOST DILU	Default tag table	Bool	%I11.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>
17	P-MECH. PRITOMNOST DILU	Default tag table	Bool	%I11.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>
18	L-MAJAK KONZOLE	Default tag table	Bool	%Q4.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
19	L-STRIH VYSUNUTI	Default tag table	Bool	%Q4.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
20	L-STRIH ZASUNUTI	Default tag table	Bool	%Q4.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
21	L-UPINKA ZAVRIT	Default tag table	Bool	%Q4.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>
22	L-UPINKA OTVERIT	Default tag table	Bool	%Q4.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>
23	P-MAJAK KONZOLE	Default tag table	Bool	%Q5.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
24	P-STRIH VYSUNUTI	Default tag table	Bool	%Q5.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
25	P-STRIH ZASUNUTI	Default tag table	Bool	%Q5.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
26	P-UPINKA ZAVRIT	Default tag table	Bool	%Q5.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>
27	P-UPINKA OTVERIT	Default tag table	Bool	%Q5.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>
28	MAJAK STROJ-R	Default tag table	Bool	%Q6.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Obrázek 4.5: Příklad proměnných (TAGů)

Pro dodržení jednoduchosti názvů proměnných jsou na adresách $I10.x$ pouze proměnné z levého lože. Na adrese $I10.0$ je definováno snímání spodní polohy levé upínky. Port s adresou $I10.1$ slouží k snímání horní polohy levé upínky. V případě, že by se jednalo o jinou část zařízení, budou proměnné zapsány do jiné skupiny adres. Například pravé lože s adresami $I11.x$.

Všechny proměnné jsou typu BOOL. Data typu BOOL jsou založena na logice pravda a nepravda (*TRUE/FALSE*). Tento typ lze také připodobnit jako 1 a 0 , kdy 1 je *TRUE* a 0 je *FALSE*.

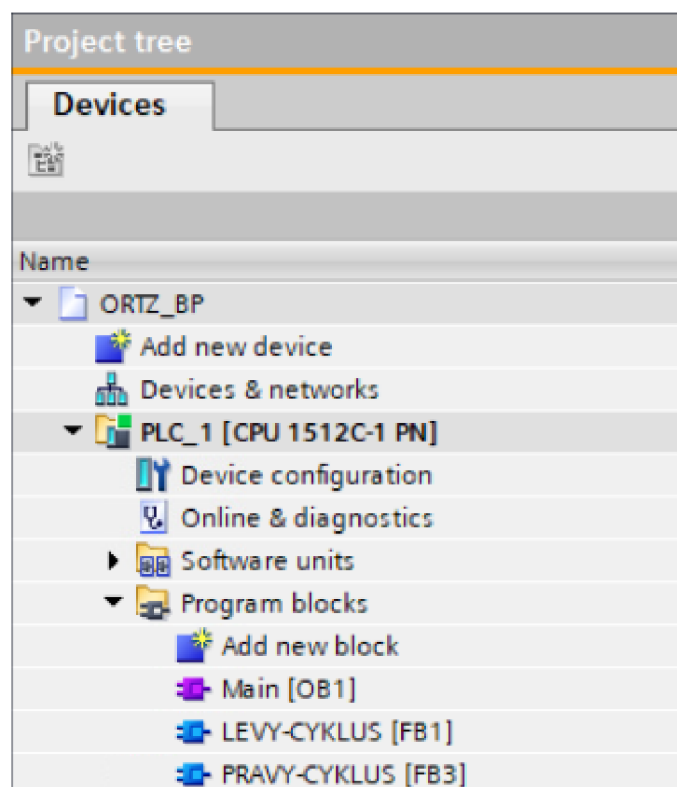
4.2.3 Rozdělení do podprogramů

Pokud by celý řídicí program zůstal v jednom jediném bloku, byl by program nepřehledný a zmatečný. Pro přehlednější práci s celým programem bylo celé řízení rozděleno do třech oddílů.

- Hlavní blok programu *MAIN*
Zde se odehrává nejdůležitější rozhodující situace.
- Funkce *LEVY-CYKLUS*
Řízení stříhání na levém loži.
- Funkce *PRAVY-CYKLUS*
Řízení stříhání na pravém loži.

V těchto podfunkcích jsou řešeny upínací a střížné kroky stroje pro levé a pravé lože zvlášť. Takto je program mnohem přehlednější a lépe upravitelný.

Rozdělení ovládání jednotlivých loží je vhodné, protože dovoluje upravovat jednotlivé strany zvlášť. V realitě by mohly být rozdílnosti mezi pravou a levou stranou. Tím pádem by mohlo být jednotlivé řízení rozdílné.



Obrázek 4.6: Project tree v TIA Portal

Na obrázku 4.6 lze vidět tzv. strom projektu. Slouží pro organizaci jednotlivých odvětví projektu v TIA Portalu.

4.3 Main Řídícího Programu

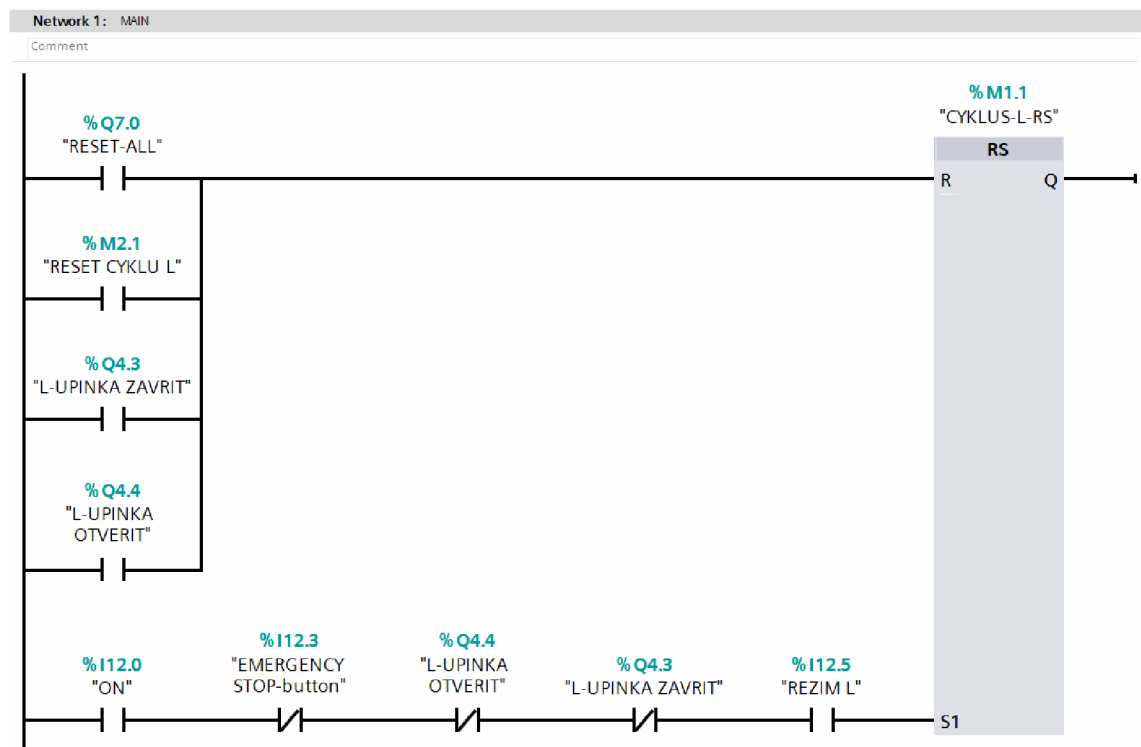
4.3.1 Main

V hlavním bloku programu lze najít větev nazvanou Main. Obsahuje základní řízení programu. Tato větev také spouští funkce *LEVY-CYKLUS* a *PRAVY-CYKLUS*.

Na obrázku 4.7 je zobrazená větev Main. Jsou na ní umístěny kontakty, které udávají podmínky, podle kterých se spínají bloky *CYKLUS-L-RS* a *CYKLUS-R-RS*. Tyto bloky slouží jako hlavní podmínka pro spuštění vedlejších funkcí. Podmínky berou ohled na stav napájení stroje, stav tlačítka *TOTAL STOP* (také nazýváno *EMERGENCY STOP* jako je na obrázku 4.7) a jaký režim vybral operátor.

Také se zde nachází podmínky *UPINKA OTEVRIT/ZAVRIT*. Tyto podmínky jsou umístěny tak, aby se cyklus nemohl znovu zapnout v průběhu již běžícího cyklu. fungují jako ochrana proti zacyklení systému. Bloky cyklů jsou resetovány vnitřní proměnnou *RESET CYKLU*, která se aktivuje když proběhne celý pracovní cyklus. Případně je lze nulovat pomocí *RESET-ALL*.

Pro pravé lože vypadá main totožně, pouze se mění proměnné z levých na pravé.

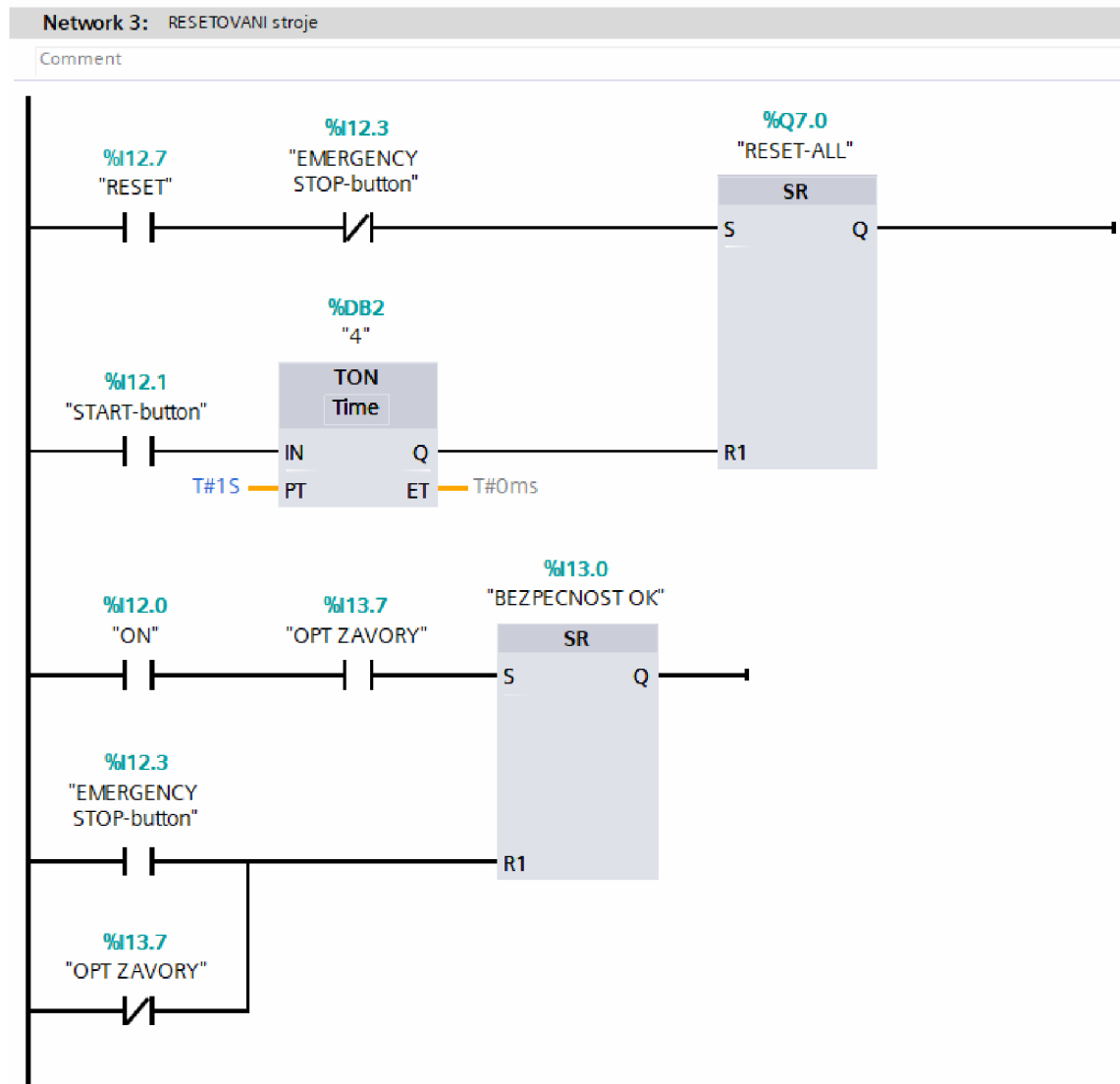


Obrázek 4.7: Větev Main

4.3.2 Reset a Bezpečnost

Funkce reset je klíčovou součástí programu, jelikož vytváří možnost navrácení zařízení do původního stavu. V reálném procesu by tlačítko reset bylo velmi užitečnou variantou pro řešení chybových stavů.

Ve stejné větvi byla i zakomponována bezpečnost. Ta se skládá z podmínek bezpečnostních optických bran které zabraňují operátorovi narušit pracovní prostor. Na obnovení bezpečnosti samozřejmě závisí i *TOTAL STOP* viz 4.8.



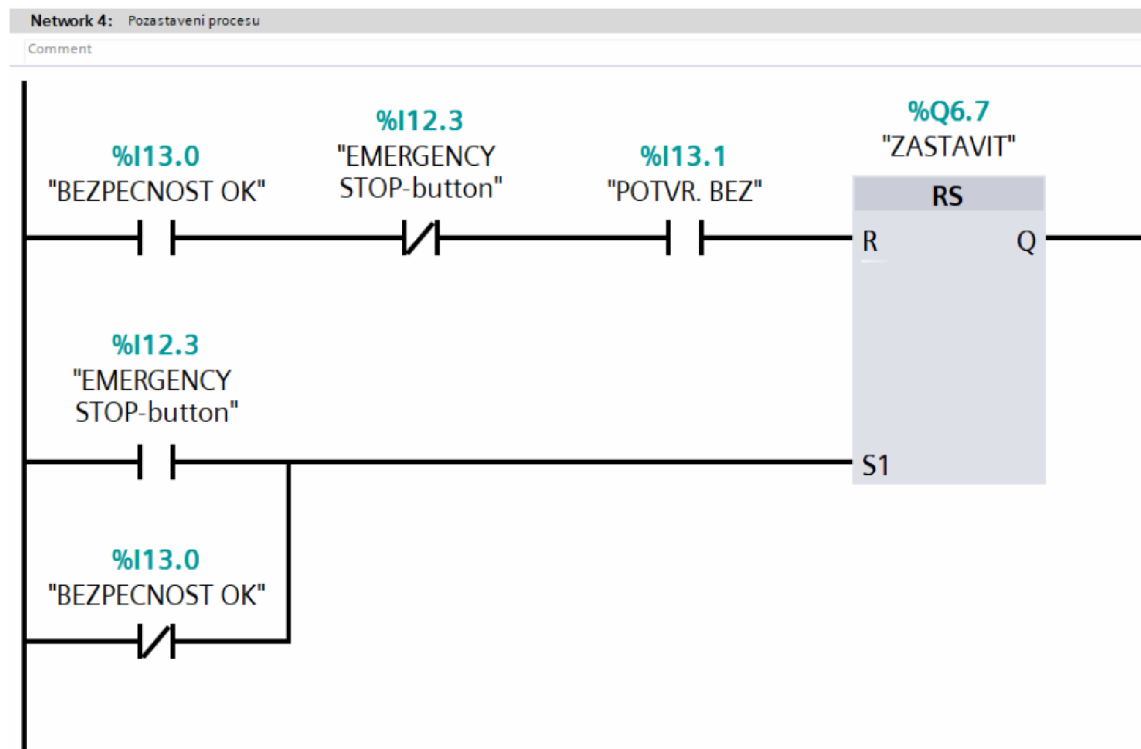
Obrázek 4.8: Větev pro resetování pozice

Na obrázku 4.8. lze také vidět časovač *TON*. Ten je zde použitý, aby operátor mohl spustit další proces, až pokud pokračování není z důvodu bezpečnosti ohroženo. Musí tedy tlačítko podržet po dobu 1s.

4.3.3 Zastavení zařízení

Následující oddíl programu existuje pro zjednodušení procesu zastavení všech pohyblivých částí. Jedná se o zastavující větev, která zvažuje všechny podmínky pro zastavení stroje.

Následně lze ve zbytku programu použít pouhého jednoho kontaktu *ZASTAVIT* bez nutnosti vkládat opakovaně celou logiku. Zmíněné zjednodušení programu ušetří velké množství zbytečných větví, které by jinak byly vkládány před každé zařízení. Vytváří tak přehlednější prostředí.



Obrázek 4.9: Větev pro zastavení stroje

Z obrázku 4.9 lze vidět jasné fungování tohoto zastavujícího programu. Pokud je narušena bezpečnost, nebo je stlačeno tlačítko *TOTAL STOP*, stroj se zastaví. Do pohybu ho lze dostat pokud je vše znovu v pořádku a potvrdí se potvrzovacím tlačítkem na ovládacím panelu.

4.3.4 Signalizace

Signalizace je důležitou součástí stroje. S pomocí kvalitně zpracované signalizace může většina jednoduchých zařízení pracovat bez toho aniž by byl přítomný například HMI panel, nebo obrazovka s chybovými informacemi.

V dnešní době je to klíčový prvek pro ideální fungování stroje a jeho bezchybný chod. U velkých výrobních procesů musí být stroj pro operátora přizpůsobený tak, aby všechny chybové stavy dokázal vyřešit sám. Toho lze docílit vhodným proškolením operátorů při seznamování se se strojem. Ovšem signalizace dané situace zařízení je taktéž klíčová. Bez ní by možná operátor možná věděl jak problém vyřešit, ale nevěděl by, který problém je třeba řešit.

Na tomto zařízení je informační maják usazen nad pracovní plochou. tento barevný maják nám dokáže říci velké množství informací ohledně stavu stroje. Každá barva značí danou situaci stroje.

- **Zelená barva**
Stroj je připraven na práci, nebo již pracuje správně.
- **Žlutá barva**
Stroj je v chodu, dbejte zvýšené opatrnosti.
- **Červená barva**
Stroj je v chybovém stavu, byla porušena bezpečnost nebo byl nouzově zastaven.

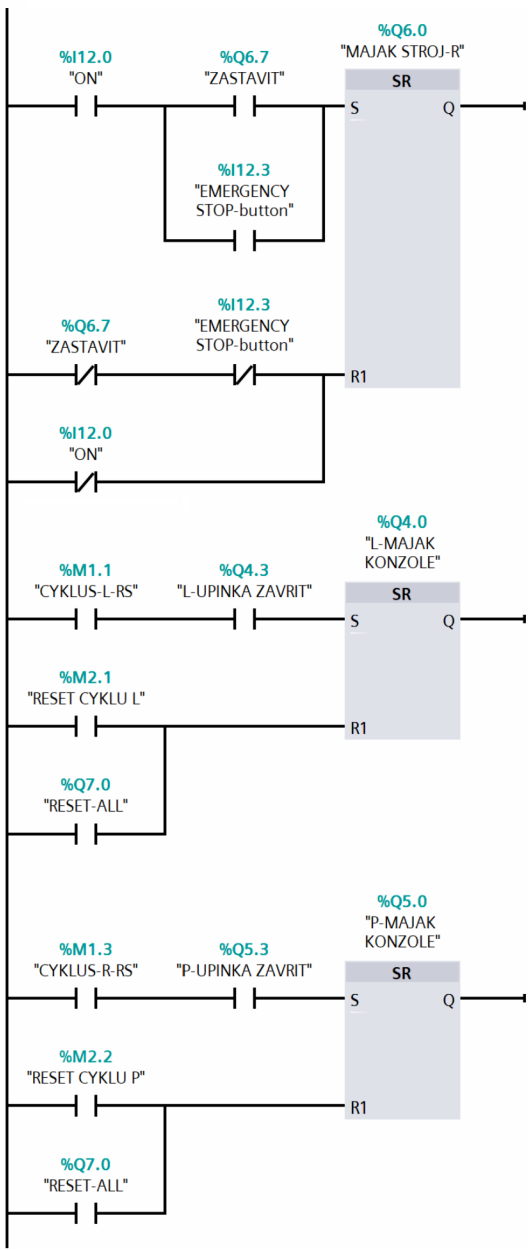
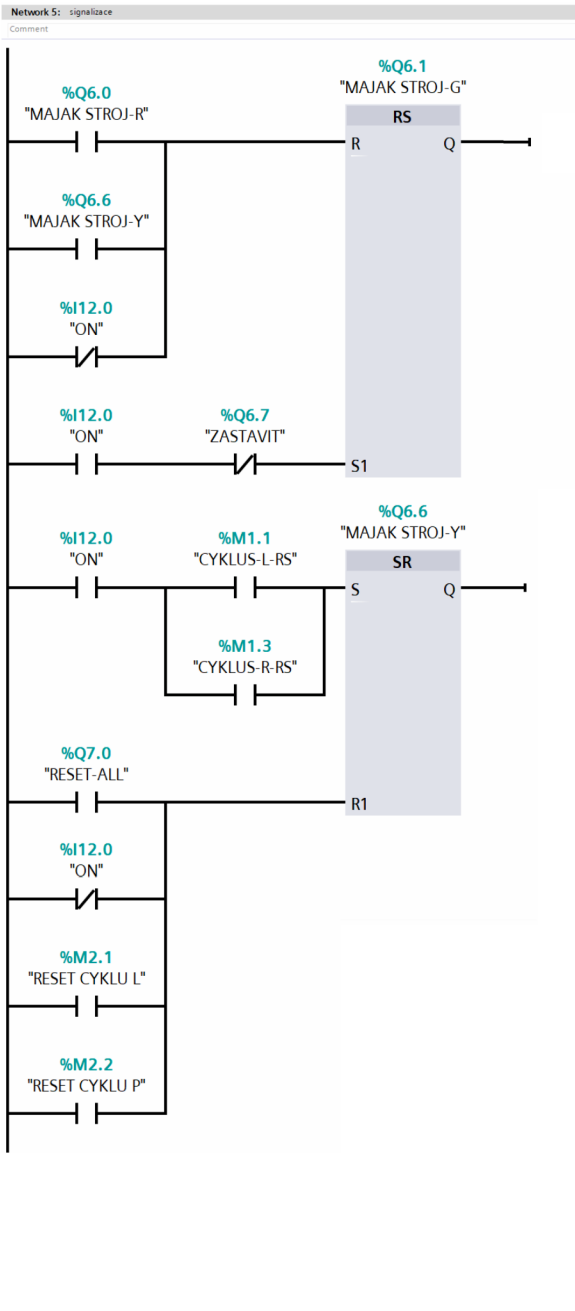
Na samotných střížných válcích se nachází také signalizační majáčky. Ty slouží k upozornění operátora, že zařízení je v pohybu, aby dbal zvýšené opatrnosti. Jsou řízeny samostatně, takže pokud se pracuje pouze na levém loži, pravý maják na válci nesvítí.

Pro snížení chybovosti lidským faktorem jsou na samotných senzorech světélka, která zvýrazňují zda bylo s dílem manipulováno správně. Pomocí dvou snímačů můžeme zajistit správné založení dílu do zakládacího lože.

Nachází se zde optický senzor sloužící k podání informace o přítomnosti dílu. Tato informace je potřebná k následujícímu uchycení dílu, ale nezajišťuje přesnou polohu.

K přesnému určení slouží mechanické čidlo na dně zakládacího lože, které lze aktivovat pouze po správném zatlačení dílu do lože upínkou. Zakládací lože je konstrukčně vytvořeno tak, aby čidlo nemohlo být aktivováno, aniž by díl nebyl správně založen.

Na obrázku níže je zobrazená logika, podle které jsou signalizační prvky řízeny. Jelikož větev signalizace je příliš dlouhá, byla rozdělena na dvě poloviny. Ty jsou vyobrazeny na obrázku [4.10](#).



Obrázek 4.10: Větev Signalizace stavu stroje

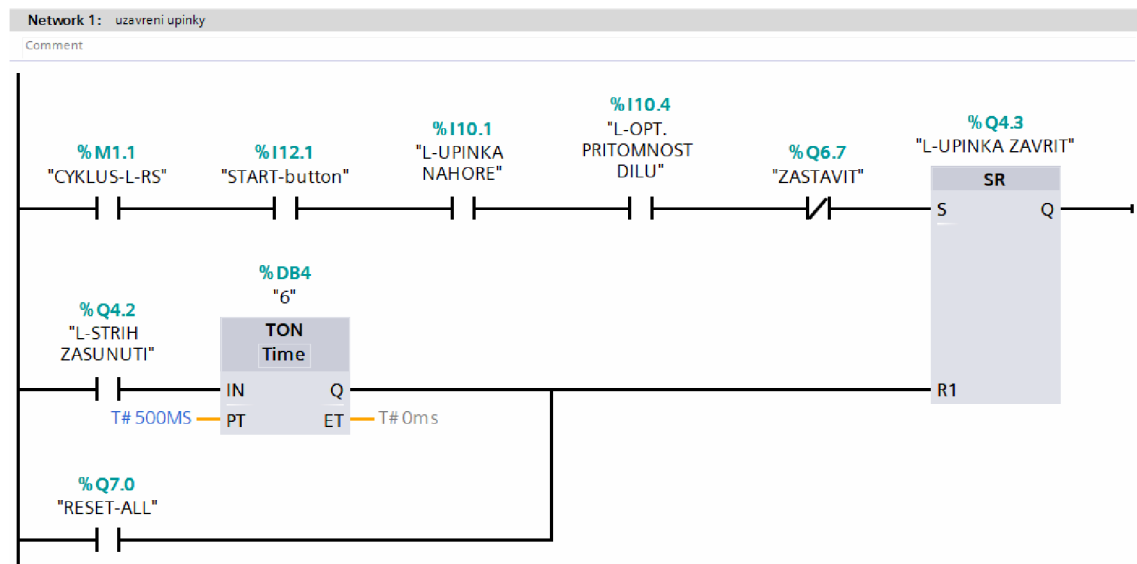
4.4 Střížné cykly řídicího programu

Hlavním produktem prostřihovacího stroje je vytvoření prostřížené díry do plastového dílu. Docílení tohoto produktu mají na starosti funkce cyklů. Jsou dvě a to pro levé a pravé lože. obě funkce jsou totožné, ale musí být oddělené. Princip obou cyklů je vysvětlen na levém loži.

4.4.1 Uzavření upínek

Poté, co operátor zapne přívod elektřiny do stroje pomocí hlavního vypínače, musí založit díl, opustit pracovní prostor a stlačit tlačítko start. Následně se zapne část cyklu, která uzavře upínací ramena. Tyto ramena zatlačí díl do správné pozice v loži a tím aktivují mechanický senzor.

Signál pro uzavření upínky zmizí až poté, co se díl prostřihne, protože následovně má být upínka otevřena. Autor zvolil logiku vypínání signálu až po prostřihnutí, aby zabránil možné ztrátě upínacího tlaku na díl.



Obrázek 4.11: Větev pro uzavření upínek

Pokud budeme brát v úvahu, že pneumatické obvody jsou ovládány elektromagnetickými ventily, které jsou řízeny výstupními hodnotami z PLC, tak ventil přepíná své stavy na základě výstupních hodnot z řídicího systému.

Autor se rozhodl ponechat ventil v otevřeném stavu po celou dobu stříhu, aby upínací rameno neustále vytvářelo tlak na díl a drželo díl ve správné poloze. Kdyby se tento signál přerušil ihned po dosažení koncové polohy, mohl by zde vzniknout nižší tlak na díl po přepnutí stavu ventilu.

Časovač *TON 6* se zde využívá pro zpoždění, díky kterému se ventil přepne do výchozího stavu až v momentě, kdy se střížník zasune do pozice, kde je mimo díl. To zapříčiní, že upínka stále drží díl v pozici i při zasouvání.

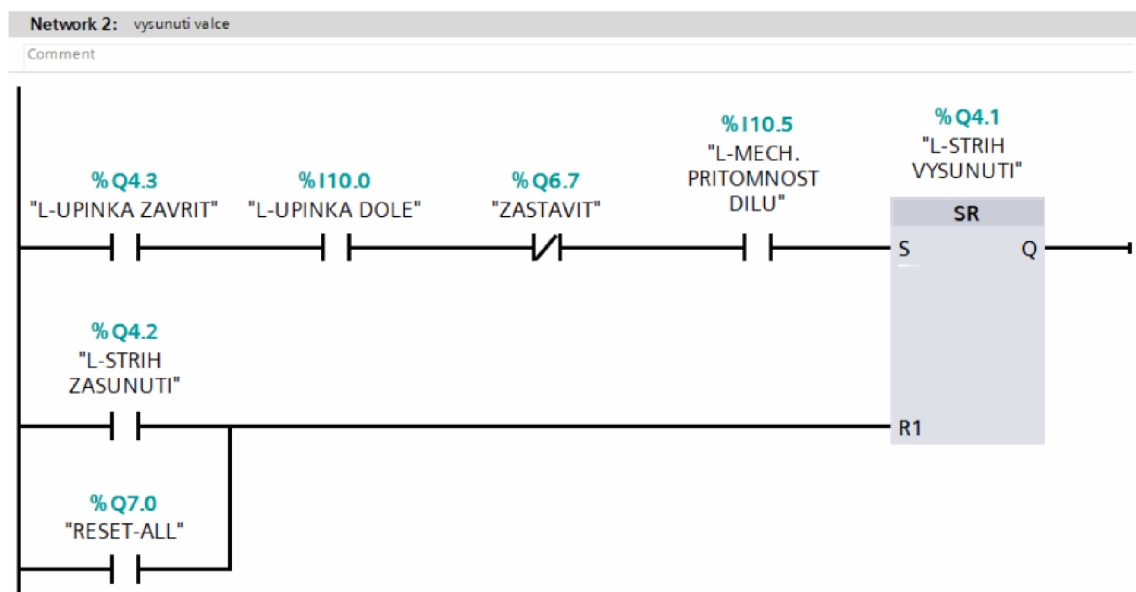
Na obrázku 4.11 lze vidět příslušný kontakt *RESET-ALL*. Ten se zde nachází, aby při resetování stroje nebránil následnému otevření upínky.

4.4.2 Vysunutí střížného válce

Hydraulický válec osazený střížným nástrojem je vysunut jakmile upínka dosáhne konečné pozice a mechanický senzor pro přítomnost dílu je aktivován. Také je zde závislost na předchozí větvi, aby kroky cyklu museli jít vždy jednotlivě po sobě, a nevzniklo zacyklení.

Válec se vysune a prostříhne díl. Signál pro vysunutí se přeruší až v závislosti na další větvi, pomocí které se válec zasouvá.

Lze zde opět vidět příslušný kontakt *RESET-ALL*. Kontakt je zde umístěn, aby při resetování stroje nebránil tento signál k navrácení do původní pozice válce.



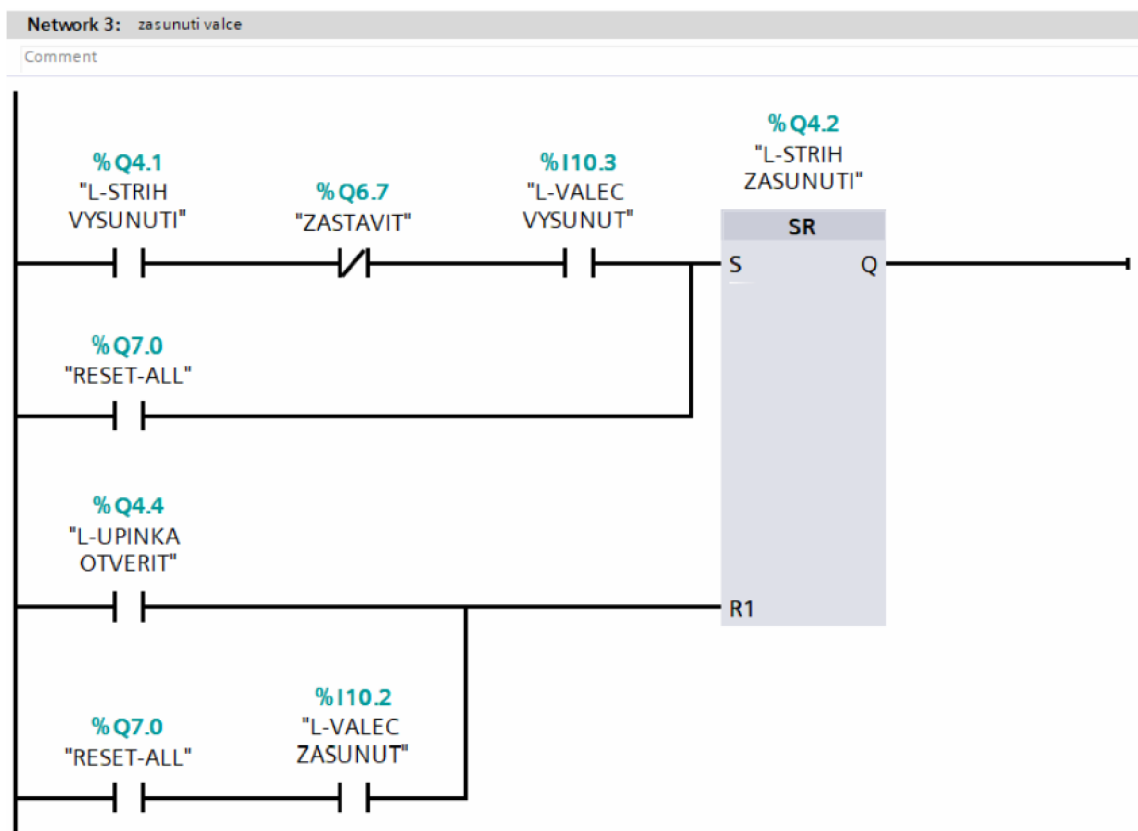
Obrázek 4.12: Větev pro vysunutí střížného válce

4.4.3 Zasunutí střížného válce

Po prostříhnutí dílu se válec se střížníkem může zasunout. Zasunutí válce nastane po dokončení předchozího kroku a zpětné vazby, že válec dosáhl koncové pozice. Příkaz na zasouvání válce bude ukončen s dalším krokem, a to jest otevření upínky.

Lze také vidět implementaci logiky zastavení a resetování stroje. Tato větev je jedna z klíčových větví pro funkci resetování zařízení. Zaručuje že po resetování zařízení se střížný válec zasune, a v momentě kdy dorazí do koncové pozice tak se zasouvat přestane.

Ovládání této větve je zobrazeno na obrázku 4.13.



Obrázek 4.13: Větev pro zasunutí válce

4.4.4 Otevření upínek

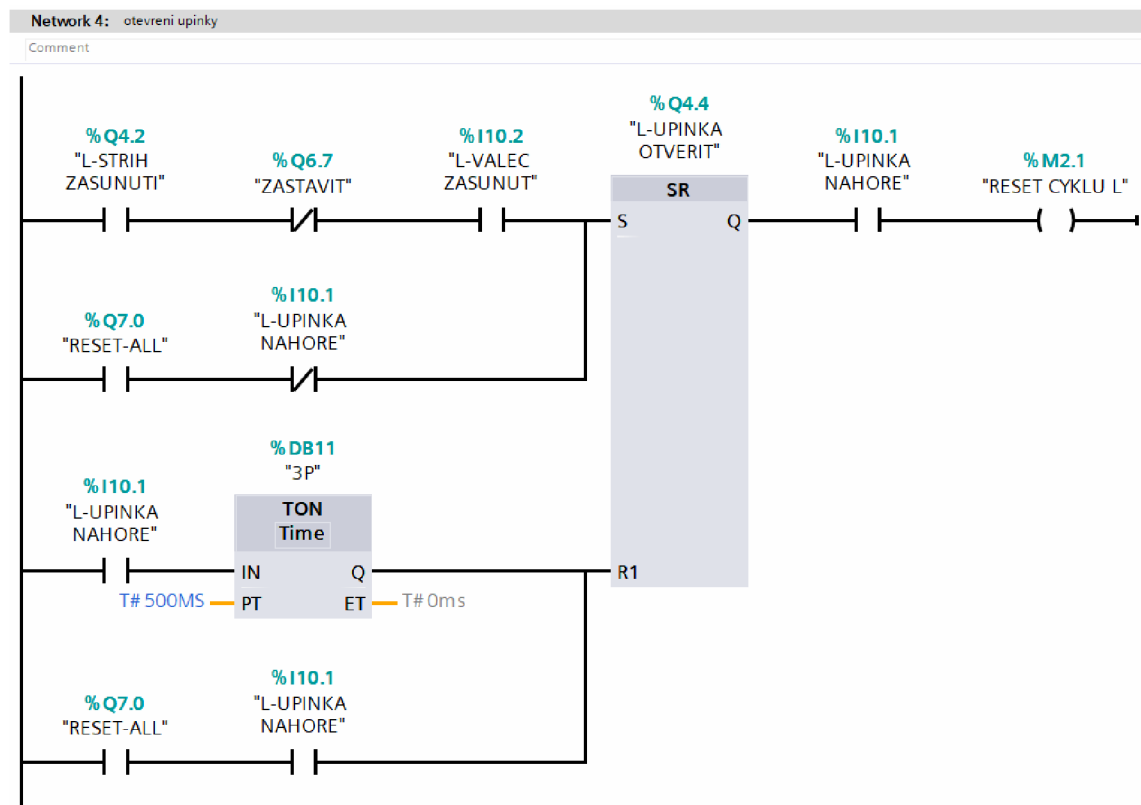
Otevření upínek je poslední akcí v celém cyklu stříhání. Po zasunutí střížného válce zpět do původní pozice se splní dvě hlavní požadované podmínky pro otevření upínek.

Otevírání těchto upínek je závislé na předchozím cyklu zasouvání válce a zároveň na koncové pozici zasunutého válce.

Na výstupu lze vidět cívku *RESET CYKLU*. S pomocí této cívky se celý průběh cyklu vyresetuje, a lze začít nový cyklus. Tento stav ale nastane, až v momentě, kdy se upínka otevře do své koncové polohy.

Dosažením otevřeného stavu se také vyresetuje samotný SR blok. Na větvi resetu tohoto bloku se také nachází TON časovač, aby se vždy nejdříve vyresetoval celý cyklus cívkou na výstupu, a až následně se vypnul SR blok.

Otevírání upínek je druhý klíčový bod pro vracení stroje do původního stavu. Na obrázku 4.14 lze vidět implementované kontakty *RESET-ALL*. Jak na vstup bloku SR (*S - SET*) tak na resetovací větvi bloku SR (*R1 - RESET*) jsou kontakty pro otevírání upínky do otevřeného stavu.



Obrázek 4.14: Větev pro otevření upínek

Závěr

V této bakalářské práci byla úspěšně realizováno digitální dvojčce prostřihovacího zařízení pro výuku programování PLC, s použitím softwarů SIMIT a TIA Portal od společnosti Siemens. Cílem práce bylo ověřit možnosti tvorby digitálního dvojčete ve zmíněných softwarech. Výsledkem práce je simulace, vizualizace a řízení v reálném čase, které umožňují testování a vývoj řídicích programů bez potřeby fyzického prototypu zařízení. S pomocí výsledku bakalářské práce lze využít digitální dvojčce pro následnou výuku v oblasti programování PLC.

Práce systematicky popisuje výběr a analýzu procesu prostřihování, návrh a realizaci řídicího algoritmu v prostředí TIA Portal a následnou simulaci zařízení v SIMIT s důrazem na interaktivní vizualizaci procesu a zpětnou vazbu pro řídicí program. Klíčovým přínosem digitálního dvojčete je redukce nákladů a času potřebného pro vývoj a testování PLC programů, což potvrzuje, že navrhovaný model může sloužit jako efektivní nástroj pro výuku a vývoj v oblasti mechatroniky.

Výsledky této práce jasně ukazují, že použití digitálních dvojčat ve vývoji a výuce umožňuje nejen hloubkové pochopení dynamiky a funkčnosti zařízení, ale také poskytuje bezpečné a ekonomicky efektivní prostředí pro experimentování a optimalizaci, což přináší významné výhody pro akademické i průmyslové aplikace. Zkušenosti a znalosti získané během této práce poskytují pevný základ pro další výzkum a vývoj v oblasti simulací a virtuálního prototypování.

Tato práce také podporuje další rozvoj digitálních dvojčat, včetně integrace s dalšími pokročilými technologiemi, což by mohlo dále zlepšit přesnost simulací a efektivitu vývoje. Doporučuje se proto pokračovat v rozvoji a aplikaci digitálních dvojčat v průmyslových aplikacích, s potenciálem pro inovace v automatizaci a vzdělávání v oboru mechatroniky.

Použitá literatura

1. SIEMENS. *Ladder Logic (LAD) for S7-300 and S7-400 Programming*. 2005. Dostupné také z: https://cache.industry.siemens.com/dl/files/395/18654395/att_33349/v1/KOP_e.pdf.
2. SIEMENS. *SIMATIC SIMIT Simulation Platform (V10.2)*. 2020. Dostupné také z: [https://support.industry.siemens.com/cs/document/109780242/simatic-simit-simulation-platform-\(v10-2\)?dti=0&lc=en-CZ](https://support.industry.siemens.com/cs/document/109780242/simatic-simit-simulation-platform-(v10-2)?dti=0&lc=en-CZ).
3. SIEMENS. *CPU 1512C-1 PN*. 2024. Dostupné také z: <https://support.industry.siemens.com/cs/pd/1093653?pti=td&dl=cs&lc=cs-CZ>.
4. SIEMENS. *SIMATIC S7-PLCSIM Advanced*. 2024. Dostupné také z: <https://www.siemens.com/cz/cs/products/automation/systems/industrial/plc/simatic-s7-plcsim.html>.
5. SIEMENS. *SIMIT Simulation Platform*. 2024. Dostupné také z: <https://www.siemens.com/global/en/products/automation/industry-software/simit.html>.
6. SIEMENS. *TIA Portal*. 2024. Dostupné také z: <https://www.siemens.com/cz/cs/products/automation/industry-software/automatizacni-software/tia-portal.html>.

Přílohy

ORTZ_BP_NAVOD

V této příloze lze nalézt kompletní podrobný návod pro zprovoznění digitálního dvojčete.

ORTZ_BP_TIA

Projekt v prostředí TIA Portal pro kompletní řízení digitálního dvojčete.

ORTZ_BP_SIMIT

Projekt v prostředí SIMIT pro simulaci a vizualizaci digitálního dvojčete.