



# **DLOUHODOBÁ MATURITNÍ PRÁCE**

## **2021/2022**

**Automatizovaný vysokozdvíhový sklad**  
Práce s modelem, připojení k PLC, programová obsluha

Vedoucí práce: Ing. Zdeněk Vondra

Jakub Adamec

A4

## **Prohlášení**

Prohlašuji, že jsem maturitní práci vypracoval samostatně a použil k tomu úplný výčet citací použitých pramenů, které uvádím v seznamu přiloženém k maturitní práci.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu § 60 zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů.

V Praze dne .....

.....  
Jakub Adamec

## **Poděkování**

Rád bych poděkoval svému konzultantovi a vedoucímu práce Ing. Zdeňkovi Vondrovi za nápad na zpracování maturitní práce, pomoc s její realizací a vstřícný přístup během celého studia.

Další poděkování patří mému spolužákovi a kolegovi Martinu Erbenovi za podporu při tvorbě práce a pomoc s detekcí problémů v programu.



Střední průmyslová škola elektrotechnická, Praha 2, Ječná 30  
121 36 PRAHA 2, Ječná 30  
tel./fax 224 941 469\*, 224 942 064\*, 224 942 066\*  
tel. 224 941 146 (ŘŠ), 224 942 065 (ZŘ), 224 941 470 (VP)  
e-mail: spsejecna@spsejecna.cz, mandik@spsejecna.cz  
www: http://www.spsejecna.cz



## Zadání odborné práce

Školní rok:	2021/2022
Zadáno dne:	
Termín 1. kontroly	do 1. 2.
Termín 2. kontroly	do 1. 3.
Termín odevzdání:	do 31. 3.
Termín obhajoby:	v rámci ústních zkoušek profilové části maturitní zkoušky před maturitní komisí
Minimální rozsah práce:	15 stránek formátu A4 (text včetně schémat, tabulek a grafů)
Počet vyhotovení:	2
Kritéria hodnocení:	práce bude hodnocena po stránce obsahu – přínosu zpracovávaného tématu, po stránce jejího písemného a grafického zpracování a po stránce vytvořené prezentace. V případě vytvoření funkčního vzorku se hodnotí i jeho provedení, konstrukce a hlavně jeho funkčnost.

Jméno a příjmení žáka: **Jakub Adamec**

Třída / katalogové číslo: A4/1

Název (téma): Práce s modelem „Automatizovaný vysoko zdvižný sklad“ – připojení k PLC, programová obsluha

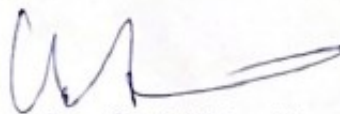
Vedoucí práce: **Ing. Zdeněk Vondra**

Práce bude obsahovat:

- 1) Toto zadání
- 2) Teoretický rozbor dané problematiky
- 3) Popis použitých součástí a vysvětlení jejich funkce (DC motor, kompresor, pneumatický válec, elektromagnetický ventil, světelná závora, koncový spínač)
- 4) Vlastní řešení a jeho popis
- 5) Schéma zapojení, pokud se jedná současně o praktickou realizaci obvodu, včetně případných tabulek a grafů
- 6) Závěr
- 7) Další body zadání určené vedoucím práce:  
Cílem práce je realizovat připojení sestavené stavebnice automatizovaný vysoko zdvižný sklad Fischertechnik 24 V k PLC Tecomat Foxtrot a naprogramovat obsluhu pracoviště.  
Sklad je tvořen devíti skladovacími boxy uspořádanými do tří řad a tří sloupců. Úkolem je umístit do jednotlivých skladovacích boxů barevné obrobky (bílé, červené, modré) tak, aby stejné barvy byly v jedné řadě. V nejvyšší řadě budou obrobky bílé, v další červené a v nejspodnější modré. Obrobky jsou umístěny ve

obrobky bílé, v další červené a v nejspodnější modré. Obrobky jsou umístěny ve speciálních nosičích (paletách), sloužících k jejich přemístění. Na paletách jsou pruhy pro kódování barvy obrobku, který do dané palety patří. Na začátku programu jsou všechny palety prázdné a jsou umístěny v boxech. Pro pohyb nosiče palety slouží tři pohybové šrouby a tři stejnosměrné motory, které mohou měnit směr otáčení. Dva pohyby jsou horizontální ve dvou na sebe kolmých směrech a třetí pohyb je vertikální.

- a) První horizontální pohyb je ve směru od komory s IR čidly ke skladu a naopak. Pro snímání polohy zde slouží koncový spínač, který je umístěn v krajní poloze u komory. Od tohoto výchozího stavu se poloha odměřuje pomocí inkrementálního snímače polohy. Nulování inkrementálních snímačů provádějte softwarově.
- b) Druhý horizontální pohyb je ve směru od nosiče palet k boxům skladu a naopak. Obě koncové polohy jsou snímány pomocí dvou koncových spínačů.
- c) U vertikálního pohybu je koncový spínač v horní krajní poloze a od něj se poloha odměřuje pomocí dalšího inkrementálního snímače polohy.
- d) Paleta je pomocí dopravníku posouvána do komory, ve které se do ní vkládá obrobek. Na začátku a na konci komory jsou světelné závory, které indikují přítomnost palety. V komoře jsou dva IR senzory, které snímají kód na paletách a určují tak, barvu obrobku, který se má do palety vložit. Tyto senzory jsou připojeny na analogové vstupy PLC.
- e) K vstupní jednotce IB – 1301 připojte ještě čtyři vstupní bity, na nich budou tlačítka, která použijete k ovládání chodu programu.
- f) Po spuštění programu pohony najedou do inicializační polohy – sepnut koncový spínač horizontálního pohybu z bodu a) a z bodu c). Od této výchozí polohy se budou odměřovat polohy jednotlivých boxů. Dopravník se nepohybuje. Nosič palet vyjede do krajní polohy vně boxů.
- g) Potom program čeká na stisknutí tlačítka „start“ – viz bod 7) f) zadání.
- h) Pak nosič palet najede pod první paletu v první řadě a přemístí ji dovnitř komory. Zde se otestuje, jakou barvu má mít vkládaný obrobek a na displeji PLC se tato barva vypíše. Obrobek této barvy se ručně vloží do palety.
- i) Dále program čeká na stisknutí tlačítka „umístnit“ – viz bod 7) f) zadání.
- j) Potom se obrobek s paletou přemístí do boxu skladu. Následuje další obrobek z první řady atd. Pak se plní druhá a nakonec třetí řada boxů. Na závěr všechny pohony najedou do inicializační polohy.
- k) Další cyklus se zahájí stisknutím tlačítka „start“.
- l) Vysvětlíte princip použitých IR senzorů.



Ing. Ondřej Mandík  
ředitel školy

Potvrzuji, že jsem si přečetl směrnici k vypracování odborné práce a převzal jsem zadání

datum:

Podpis žáka:

# Obsah

Úvod.....	4
1 Teoretický rozbor problematiky práce.....	5
1.1 Úvodní popis modelu.....	5
1.2 Zamyšlení nad tématem.....	6
2 Popis použitých součástí.....	7
2.1 Programovatelný automat.....	7
2.2 Stejnosměrné motory.....	8
2.2.1 Motory se snímači.....	8
2.2.2 Motory bez snímačů.....	9
2.3 Inkrementální snímače.....	9
2.4 Koncové spínače.....	10
2.5 Infračervené senzory.....	10
2.6 Světelné závory.....	11
3 Software.....	11
4 Závěr.....	14
5 Použitá literatura a prameny.....	15
6 Příloha – Celý program.....	16

## Úvod

Už z dob dávno minulých si lidé chtěli ulehčovat práci různými vynálezy a inovacemi. Svědčit o tom může už zmínka z dávné Alexandrie, kdy se za pomoci ohně a vzniklých par z vroucí vody automaticky otevíraly a zavíraly chrámové dveře. Od té doby se lidský přístup k automatizaci velmi změnil a odlišil. Další významný postup v sestrojování automatických přístrojů nastal v roce 1801, kdy francouzský vynálezce a tkadlec Joseph Jacquard představil svůj vynález, který nahrazoval práci pomocníka. Od té doby se více a více ve společnosti objevují stroje, které ulehčují práci a zvyšují kvalitu výrobků i života lidí.

Dělník, který je plně nasazený 24 hodin každý den v roce. Bez přestávky a bez jakéhokoliv pocitu únavy pracuje tak, jak se mu přikáže. Nezná bolest nebo životní potřeby. Pracuje efektivně, rychle, precizně. To je sen každého podnikatele a inženýra. Jak je ale možné, že to stroj dokáže? Co stojí za tak vysokou kvalitou práce? Tato práce má sice být především o popsání fungování jednoho typického modelu zautomatizování rutinní lidské práce, tedy práce ve skladu, ale zároveň je i ukázkovým příkladem toho, proč jsou stroje mnohem spolehlivější a lepší pracovní síla, než je člověk. Má maturitní práce má popsat řešení problematiky automatizovaného skladu pomocí programovatelného automatu, ale v podstatě popíše i pomocí čeho lidstvo docílí dokonalé strojové práce.

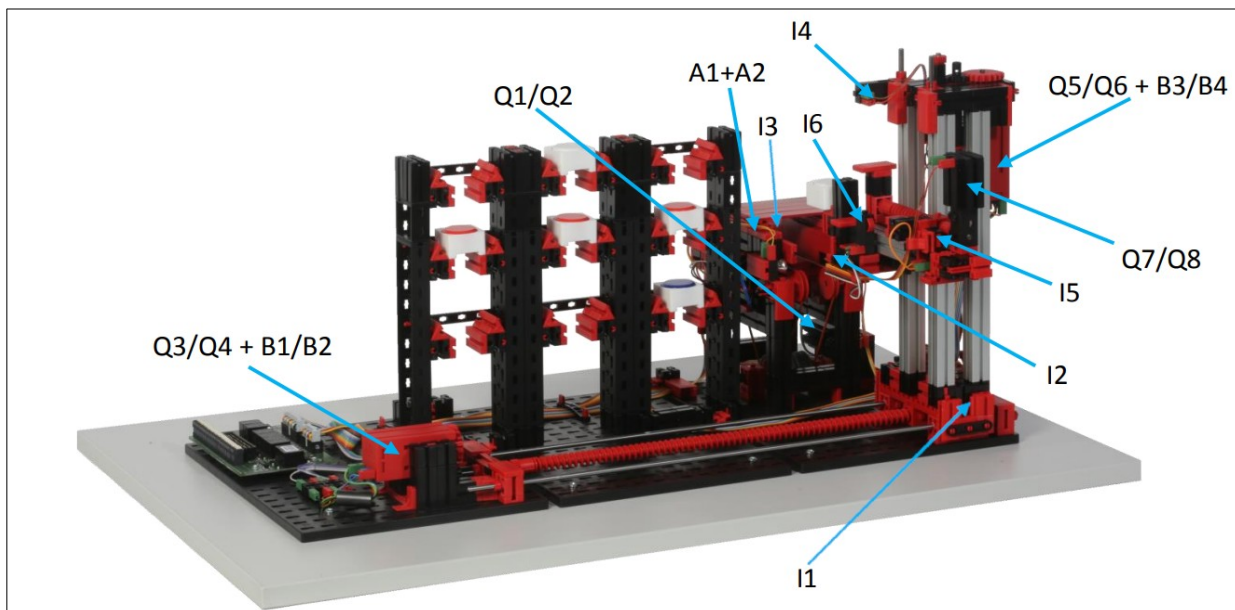
Může vyvstat otázka, zdali časem nenahradí automatické stroje práci všech lidí na světě, tedy jestli nenastane situace jak v dramatu R.U.R. od českého autora Karla Čapka. Osobně se této situace nebojím, neboť i ten nejlepší stroj a vynález nedokáže být stejně kreativní jako člověk. Rozhodně časem nahradí (a už i nahrazuje) práci spousty dělníků, skladníků, pokladníků a jiných profesí, které vyžadují rychlost a přesnost, ale naopak nepotřebují kreativitu.

Motivací a důvodem výběru tématu této práce bylo pro mě především ukázat na jednoduchém modelu krásy automatizace širšímu publiku, které se nemuselo úplně setkat s praktickou automatizací v reálném světě. Dalším důvodem byl také fakt, že i já jsem se chtěl odborněji věnovat práci s programovatelnými automaty a zajímat se o problematiku hlouběji.



# 1 Teoretický rozbor problematiky práce

## 1.1 Úvodní popis modelu



Obr. 1 Vyobrazení modelu s popsánymi prvky

Tab. 1 Zapojení jednotlivých periférií skladu na programovatelný automat

číslo pinu	vstup/výstup	jednotka	bit číslo	D9 konektor pin č.	popis
1	"+"		hnědá	4, 5	Zdroj + 24 V pro akční členy
2	"+"		zelená	4, 5	Zdroj + 24 V pro senzory
3	"_"		modrá	1, 2	0 V
4	"_"		modrá	1, 2	0 V
5	I1	IB1301	4 – šedá	DI3/9	koncový spínač horizontální osy
6	I2	IB1301	5 – fialová	DI3/8	světelná brána "dovnitř"
7	I3	IB1301	6 – modrá	DI3/7	světelná brána "ven"
8	I4	IB1301	7 – zelená	DI3/6	koncový spínač vertikální osy
9	A1	CP1014	4 – šedá	DI1/9	IR senzor pohybu dolní – pulzy 1...na analog. vstup
10	A2	CP1014	5 – fialová	DI1/8	IR senzor pohybu horní – pulzy 2...na analog. vstup
11	B1	IB1301	0 – šedá	DI2/9	kodér – horizontální osa ... pulzy 1
12	B2	IB1301	1 – fialová	DI2/8	kodér – horizontální osa ... pulzy 2
13	B3	IB1301	2 – modrá	DI2/7	kodér – vertikální osa ... pulzy 1
14	B4	IB1301	3 – zelená	DI2/6	kodér – vertikální osa ... pulzy 2
15	I5	CP1014	6 – modrá	DI1/7	koncový spínač nosný článek (krakorec) dopředu
16	I6	CP1014	7 – zelená	DI1/6	koncový spínač nosný článek (krakorec) zpět



Tab. 2 Zapojení akčních členů skladu na programovatelný automat

číslo pinu	vstup/výstup	jednotka	bit číslo	D9 konektor pin č.	popis
17	Q1 (M1)	OS1401	4 – žlutá	8 datových bitů, „+“ zdroje pro periferie, „+“ zdroje systémového a zem	motor dopravníku vpřed
18	Q2 (M1)	OS1401	5 – růžová		motor dopravníku vzad
19	Q3 (M2)	OS1401	6 – červená		motor horizontálního posuvu směrem k policím
20	Q4 (M2)	OS1401	7 – hnědá		motor horizontálního posuvu směrem k dopravníku
21	Q5 (M3)	OS1401	8 – černá		motor vertikálního posuvu dolů
22	Q6 (M3)	OS1401	9 – bílá		motor vertikálního posuvu nahoru
23	Q7 (M4)	OS1401	10 – šedivá		motor nosný článek dopředu
24	Q8 (M4)	OS1401	11 – fialová		motor nosný článek zpět

Zdroj +24 V pro akční členy je použit HDR-100-24 (dává 3,83 A). Zdroj pro CPU a senzory HDR-60-24 (dává 2,5 A).

V modelu se používá černých palet, kdy některé z nich mají nalepený bílý pruh buď na horní hraně, nebo na dolní hraně. Tyto bílé pruhy mají symbolizovat čárový kód, který se následně rozeznává pomocí infračervených senzorů. Dopravník přijede pro první paletu, doveze ji do nakládací oblasti. V ní se načte požadovaná barva obroku, který se do ní má vložit. Tato barva se vypíše na displej PLC. Po vložení obroku uživatel stiskne tlačítko, čímž dá pokyn k pokračování programu. Paleta se vrátí na dopravník, jenž ji vrátí do regálu. Poté se dopravník přesune k další paletě v řadě a proces se opakuje.

## 1.2 Zamyšlení nad tématem

Tématem je práce s modelem „Automatizovaný vysokozdvizný sklad“ a jeho připojení k PLC a programová obsluha, což vyžaduje několik úvodních úvah, jak k práci správně přistupovat.

První úvaha byla ale naprosto prostá, tedy proč chtít automatizovaný sklad? Proč by někdo chtěl drahý a prostorově nákladný sklad, a tedy, jestli má práce může mít užitkovou hodnotu. Kvůli takovému skladu by se musela postavit nová hala a vybudovat nová logistika, a když tu je už zavedená desítky let ověřená lidská práce ve skladech, která je už vyšperkovaná do nejmenšího detailu a jsou pro ni uzpůsobeny všechny sklady, má vůbec smysl taková investice? Tyto otázky jsou naprosto validní a klíčové.

Odpověděl jsem již v úvodu, jednoduše je takový sklad je efektivnější. Zprvu si sice žádá mnohonásobně vyšší kapitálovou investici, ale v horizontu několika let se začne projevovat jeho výhoda oproti „starému“ provedení. Někdo by mohl namítat, že se tím ve výsledku nezmenší lidský vliv, neboť bude stále potřeba člověka při kontrolách, opravách a úpravách jednotlivých strojů. Myslím, že v automatizaci nejde o snižování počtu lidí v procesu, spíše o usměrnění jejich práce. Lidé jsou chybový faktor kdekoli, a proto bude ve výsledku lepší usměrnit jejich pozornost na jednu konkrétní věc. V takovém skladu by měli lidé za úkol jednu jedinou věc, kontrolovat, zdali stroj funguje tak, jak má. Pracovníci již nebudou muset věnovat pozornost různým dalším problémům, jako je například hlídání si vlastního zdraví i zdraví lidí okolo.

Po zvážení myšlenky o pořízení automatizovaného skladu, je důležité se rozhodnout nad jeho provedením. Momentálně existuje už poměrně široká škála různých provedení. Jsou firmy, které k tomu přistupují tak, že využívají práce dronů, které balíčky přenášejí mezi jednotlivými stanovišti. Pak tu jsou řešení, kdy se vytvoří určitá síť pásových přepravníků a vertikálních skladovacích výtahových systémů. Jako třetí příklad se může užít například automatizovaný zakladač, který pomocí jeřábu umístí příslušné krabice a palety na přidělené místo. Jako můj osobní favorit je kombinace 2. a 3. příkladu, tedy vytvořená síť dopravníků, pomocí kterých se příslušné boxy anebo palety dopraví na jeřáb s vysouvacím ramenem. Prostorově je toto řešení sice větší, naopak je ale velmi účinné a rychlé. a díky využití jeřábu je možné vybudovat mnohem vyšší sklady, ve kterých mohou být mnohem těžší náklady, než je tomu u dronového skladu. a právě o tomto provedení je také tato práce.

Třetím důležitým faktorem je celkové řízení celého systému. Řešení je taky mnoho, některá se liší více, některá méně. Jako nejvhodnější se zdá využití programovatelných automatů (dále PLC<sup>1</sup>). Mezi výhody jejich použití se řadí například flexibilita a servis, kdy může jakýkoliv servisní technik vyměnit samotné části PLC na místě. Další výhodou je velmi vysoký výpočetní výkon a velký výběr komunikačních rozhraní. Na jeden programovatelný automat mohou být připojeny desítky vstupů i výstupů. Jako poslední majoritní výhoda PLC oproti jiným řídicím systémům je kratší doba vývoje řídicího programu. Programy jsou tedy jednoduché, leč snadno upravitelné, což se kladně projeví i na jejich případných úpravách postupem času. Naopak značnou nevýhodou programovatelných automatů je jejich vysoká pořizovací cena nebo jejich velké rozměry. PLC jsou například mnohem mohutnější jak konkurenční jednodeskové počítače. Díky jednoduchosti programovacích jazyků, které PLC používají, jsou i komplexnější programy mnohem méně efektivní, a tedy pomalejší než počítačové programy. Jako poslední nevýhoda PLC je postrádání běžných konektorů, jako jsou například USB nebo výstupy na externí displej.

Poslední myšlenka je, jaký jazyk zvolit a jakých funkcí využít pro programování PLC. PLC se programují různými jazyky, za zmínku stojí například LD – žebříčkový diagram, ST – strukturovaný text a například ještě FBD – funkční bloky. Výběr jazyka je už na každém technikovi, který chce s PLC pracovat. Já sám jsem pro tuto práci zvolil asi nejobecnější a nejvariabilnější jazyk, strukturovaný text. ST má jasnou syntaxi, ve které se vyzná opravdu každý, což je u takto komplexního projektu důležité. Rád bych také využil funkcí a funkčních bloků, pomocí nichž docílím ještě větší přehlednosti celého programu a zjednoduším jeho přípravné úpravy.

## **2 Popis použitých součástí**

### **2.1 Programovatelný automat**

PLC je číslicový elektronický systém, který řídí pracovní nástroje převážně v průmyslovém prostředí. Pro předávání informace používá číslicové anebo analogové vstupy a výstupy. Program je uložený přímo v automatu a opakuje se v cyklu. Tento projekt bude řešen pomocí PLC modulární řady Foxtrot od firmy Tecomat. Automat je vybaven základním modulem – procesorem CP-1014 a rozšiřující moduly IB-1301 a OS-1401.

---

1 Programmable Logic Controllers

Procesor má 8 binárních vstupů 24 V, z nichž maximálně 4 vstupy lze využít jako analogové s rozsahem 0÷10 V a maximálně 4 jako vstupy se speciálními funkcemi (rychlý čítač, připojení inkrementálního snímače apod.) se vstupním proudem 5 mA. Je také vybaven 6 reléovými výstupy, každý s maximálním proudem 3 A, výstupy celkově však 10 A.

Rozšiřující IB modul má celkem 12 vstupů, kdy vstupy DI0÷DI3 umožňují používat stejné speciální funkce jako CP-1014. Vstupy DI4÷DI11 jsou standardní binární vstupy se vstupním filtrem 5 ms. Umožňují zachytit pulsy s minimální šířkou 50 ms.

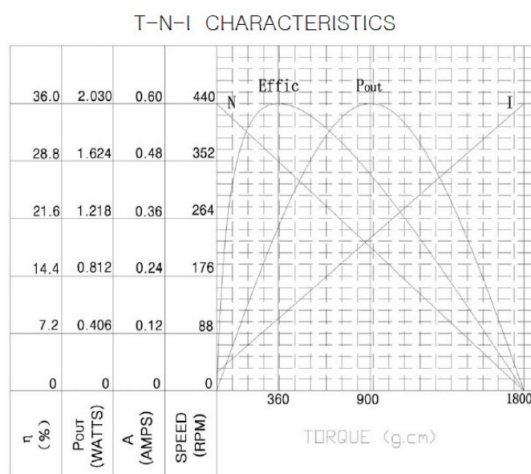
Modul OS-1401 obsahuje 12 tranzistorových výstupů se společnou svorkou "+" umožňující spínat max. 28,8 V DC. Každý výstup DO0÷DO3 umožňuje spínat proud 2 A (součet proudů zátěží všech čtyř výstupů DO0÷DO3 nesmí překročit 2,5 A). Každý z výstupů DO4÷DO11 umožňuje spínat proud až 0,5 A. Celkem však 9 A pro DO0÷DO11, max. 4,4 A pro DO0÷DO3. Zdroje jsou 2,5 A (pro CPU) a 3,83 A.

## 2.2 Stejnosměrné motory

V projektu jsou použity dva různé typy stejnosměrných motorů. Pro pohyb podavače s ramenem po horizontální a vertikální ose jsou použity dva motory se snímači (každý na jednu osu). Další dva motory jsou určeny pro výsun ramena podavače a pro pohon dopravníku palet v nakládací oblasti. Tyto další dva motory již snímač nemají. Všechny motory jsou spínány dvojicí relé  $Re_1$  a  $Re_2$ .

### 2.2.1 Motory se snímači

Tyto použité stejnosměrné motory jsou vybaveny permanentními magnety a kvadrturním enkodérem, pomocí něhož je umožněno měření úhlu natočení – inkrementální měření úhlů pomocí snímačů s Hallovým jevem. V bodě 2.3 jsou tyto enkodéry popsány podrobněji. Motory mají jmenovité napětí 24 V a maximální výkon 2,03 W při 214 otáčkách za minutu. Příkon proudu při maximálním výkonu je 320 mA. Maximální točivý moment je 5 mNm. Převodový poměr integrované převodovky je 25:1. To znamená, že snímač produkuje tři impulsy na jednu otáčku hřídele motoru nebo 75 impulsů na jednu otáčku výstupní hřídele převodovky. Vzhledem k tomu, že jsou indexovány dva fázově posunuté impulsy, je snímač schopen rozlišit směr, kterým se motor otáčí. Připojení se provádí čtyřžilovým kabelem s červeným vodičem pro 24 V a zeleným vodičem pro připojení země. Černý a žlutý vodič přenáší impulsy (push-pull výstup, max. 1 kHz, max. 10 mA).



Pro hodnoty na charakteristice Obr. 2 platí, že  $P_{out}$  zachycuje výkon na hřídeli,  $Effic$  efektivní výkon motoru,  $N$  počet otáček a  $I$  je proud motorem.

Obr. 2 Zatěžovací charakteristika použitých DC motorů se snímači

### 2.2.2 Motory bez snímačů

Jedná se o kompaktní stejnosměrné motory s permanentními magnety, který lze použít společně s přípojitelnou redukční převodovkou motoru. Motor je provozován při jmenovitém 24 V DC a maximální příkon proudu je 300 mA. Výsledkem je maximální točivý moment 5 mNm a volnoběžné otáčky 10 700 ot/min. Redukční převodovka motoru má převodový poměr 64,8 : 1 a boční výstup.

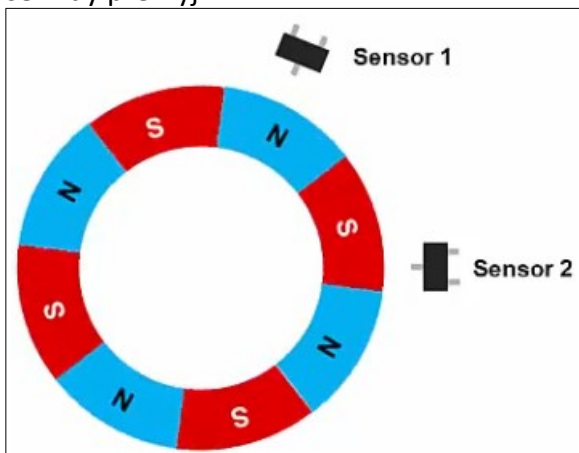
### 2.3 Inkrementální snímače

Inkrementální snímače, také přírůstkové, patří pod snímače na měření polohy. Jedná se buď o snímače s Hallova sondou, které využívají Hallova jevu, nebo optické snímače, které pracují se světlem.

Tyto snímače jsou zdrojem impulsů, které vytváří systém při přechodech mezi kladnými a zápornými póly na disku, který se otáčí na hřídeli, jež je poháněná snímaným akčním členem. Princip těchto snímačů je, že dráha je rozdělena na stejně velké úseky, které mají určenou velikost. Při překonání jednoho tohoto úseku se pomocí Hallova jevu detekuje přechod mezi póly, tím senzor vyšle 1 impuls. Změřená poloha je tedy dána počtem inkrementů (přírůstků) od počáteční polohy. Měření se provádí přičítáním nebo odečítáním impulsů.

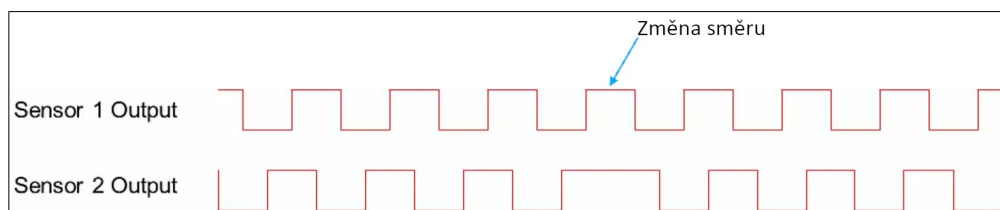
Základem přírůstkových snímačů je disk, jenž má na sobě kovové zmagnetované plošky, od kterých se odvíjí vyhodnocování. Postavení těchto plošek/dílků je snímáno vzhledem k nepohyblivé části pomocí Hallova senzoru. Díky této vlastnosti nezáleží, jestli se jedná o kruhový tvar (pro úhlovou dráhu), nebo o rovinný tvar (lineární dráha).

V této práci jsou použity inkrementální snímače úhlu. Ty fungují tak, že je disk s kladně a záporně zmagnetovanými ploškami spojen s otočnou náhonnou hřídelí, která je připevněna k otočné hřídeli motoru. Na nepohyblivou část je umístěn Hallův senzor, který při následném pohybu hřídele zaznamená změnu otáčení, díky změně magnetického pole, a převede ji na impulsy nebo obdélníkový signál. Aby bylo u snímače zajištěno generování obdélníkových impulsů bez rušení, musí být elektrický signál zesílen a elektronicky zpracován. Pro zaručení, že výsledný signál nebude zašumělý, se výstupních signálů může snímat v diferenciálním režimu, kdy se porovnávají dva téměř totožné signály, jen s opačnou fází. Tím se odstraní souhlasné rušení, protože stejné signály libovolného tvaru se vždy překryjí.



Obr. 3 Princip fungování přírůstkového enkodéru

V mém použití je ale ještě zapotřebí rozeznávat směr pohybu a nastavit počátek. Toho se docílí tak, že enkodér generuje dva obdélníkové signály (*senzor 1*, *senzor 2*), které jsou vzájemně posunuté o 90°. Zpracováním samotné *senzor 1* se získá informace o rychlosti otáčení (počet dílků) a zpracováním *senzor 2* lze rozeznat směr otáčení.



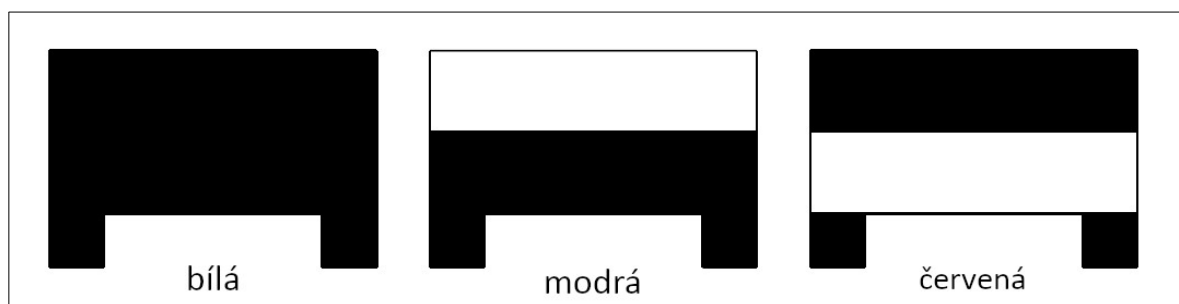
## 2.4 Koncové spínače

Koncové spínače jsou na modelu použity hned pro několik os. První koncový spínač se nachází na vertikální ose a druhý na horizontální. Ty jsou určeny převážně pro kalibrační potřeby. Při kalibrační funkci jsou určeny jako výchozí bod pro vyresetování a následné spuštění funkce inkrementálních snímačů. Další dva spínače jsou použity na výsuvném rameni podavače. Ty jsou určeny pro určení, jestli je rameno plně vysunuté, nebo plně zatažené.

V projektu se vyskytuje pouze jeden typ spínače, tím je Fischertechnik 37783. Spínají při maximálním napětí 50 V a maximálním proudu 2 A. Jsou konstruovány tak, aby mohly být použity jako detektory sepnutí i nesepnutí. Při nesepnutí jsou spojeny kontakty 1 a 2, při sepnutí 1 a 3 (1 je střední).

## 2.5 Infračervené senzory

Snímač infračervené stopy je digitální infračervený snímač pro identifikaci černé stopy na bílém pozadí ve vzdálenosti 5 až 30 mm. Skládá se ze dvou vysílacích a dvou přijímacích prvků. Signály jsou vyvedeny jako push-pull výstupy. Propojení se provádí pomocí čtyř kabelů. Červený vodič musí být připojen k 9 V DC a zelený kabel musí být připojen k zemi. Černý a žlutý kabel přenášejí signály. Deska adaptéru převádí napětí a upravuje úroveň z 24 V DC na 9 V DC.



Obr. 5 Kódování na jednotlivých paletách a barva obroku, který do nich patří

## 2.6 Světelné závory

U tohoto vysokozdvizného skladu byla použita dvojice fototranzistorů jako jednocestné světelné závory s oddělenými zdroji a přijímači záření. Jedná se o model Fischertechnik 36134, s kolektorovým proudem 15 mA, nárazově 75 mA a kolektorovým napětím maximálně 35 V.

Na takovýto fototranzistor přímo svítí zdroj světla, pokud je překročena určitá úroveň jasu, tak se tranzistor sepne a vede elektrický proud. A naopak, pokud je něco mezi světelným zdroje a fototranzistorem, tak se tranzistor uzavře. Díky těmto vlastnostem jej lze použít jako světelnou závoru. Ke snížení vlivu okolního světla byla umístěna krycí stříška. Při připojování fototranzistoru k napájení je důležité dbát na správnou polaritu. Kladný pól se připojuje na červenou značku na fototranzistoru.

### 3 Software

V této kapitole bych rád ukázal části svého programu, které podle mě stojí za zmínku a jsou svým způsobem zajímavé. Celý program, i s okomentováním a viditelnou historií úprav, dostupný veřejně na mém GitHubu<sup>2</sup> a zároveň je přiložen v příloze práce (kapitola 6 Příloha – Celý program), jakožto kompletně kopírovatelný text.

V celém programu se nachází celkem 3 funkce. Funkce kalibrační, která sama o sobě není nijak zajímavá, protože slouží pouze k úvodní kalibraci jednotky a zresetování inkrementálních snímačů. Vždy se spouští jako první věc po spuštění cyklu celého programu skladu. Další funkce objevující se v kódu je nazvaná *manualni()*. Název vypovídá za sebe, ta je určená jen pro manuální obsluhu skladu, kdy pomocí switchu je technik schopný ovládat jakýkoliv akční člen skladu. Poslední funkcí, která je už zajímavá, je funkce *ridici()*. Tato funkce se zacyklí a spouští zbylé části programu ve správném pořadí tak, aby byly dodrženy všechny body zadání. Má v sobě také zakomponovaný „pohyb v polích“, ve kterých jsou uloženy horizontální a vertikální polohy palet v regálu.

```
1 FUNCTION ridici:bool
2   IF not blokace then
3     IF not nabirani.probehlo then
4       nabirani(i:=vertikal, j:=horizontal);
5     ELSE
6       IF not hybej.odvezl then
7         hybej();
8       ELSE
9         IF not vraceni.vraceno then
10          //hlavni_barva:=3;
11          vraceni(i:=vertikal, j:=horizontal);
12        ELSE
13          IF horizontal<2 then //podminka pro posun v polich
14            horizontal:=horizontal+1; //posun v horizontalnim poli
15            nabirani.probehlo:=0;
16            hybej.odvezl:=0;
17            vraceni.vraceno:=0;
18          END_IF;
19          IF horizontal=2 AND vraceni.vraceno then
20            IF vertikal<2 then
21              vertikal:=vertikal+1;
22              horizontal:=-1;
23            ELSE
24              blokace:=1;
25            END_IF;
26          END_IF;
27        END_IF;
28      END_IF;
29    END_IF;
30  ELSE
31    kalibrace();
32  END_IF;
33 END_FUNCTION
```

Obr. 6 Funkce ridici()

<sup>2</sup> <https://github.com/knedl1k/skladplc>

Spíše, než funkcí jsem využil vlastních funkčních bloků, protože po jejich zavolání proběhnou právě jednou a je možné z nich vracet hodnotu proměnných. To je klíčové pro funkčnost dalších závislých podmínek. Funkční bloky (dále v textu i programu odkazováno jako FCB<sup>3</sup>) jsem použil celkem čtyři. Celé automatizované ovládání jsem rozdělil do 3 částí. Regál÷nakládací oblast, nakládací oblast, nakládací oblast÷regál.

První část obstarává FCB *nabirani\_FCB*, na který dále odkazuji jako *nabirani()*. Po úvodní kalibraci se sklad začne hned přesouvat na první regálovou polohu, tedy úplně nejvyšší řadu, nejvíce vpravo. Jakmile dorazí na horizontální a vertikální souřadnice, rameno se začne vysouvat. Dopravník je trochu níže na vertikálních hodnotách, to je kvůli nabírání prázdné palety. Jakmile se rameno vysune, dopravník ještě trochu vyjede nahoru, čímž paletu umístí na rameno, které následně zasune. Tím role *nabirani()* končí.

Dále nastupuje funkční blok *hybej\_FCB*, též jako *hybej()*. S již naloženou paletou rozpohybuje celý ramenový dopravník a doveze ho k nakládací oblasti. Po té se v nakládací oblasti rozpohybuje dopravník, který paletu proveze skrz komoru, ve které jsou umístěny dva IR senzory. Ty naskenují „čárový kód“ na paletě. Podle jejich výstupu se následně na displej PLC vypíše požadovaná barva obroku. Tuto část rozpoznávání a vrácení přidělených hodnot jednotlivým barvám zajišťuje vnořený funkční blok *barva\_FCB*, tedy *barva()*.

```
1 FUNCTION_BLOCK barva_FCB //FCB na poznávání barevného kodování na boku palet
2   VAR_INPUT
3     a0:int;
4     a1:int;
5   END_VAR
6   VAR
7     h:int:=6; //buffer pro color
8   END_VAR
9   VAR_OUTPUT
10    color:int:=6;
11  END_VAR
12  IF (h=2 or h=6) and (a0>=5000 or a1>=5000) then //neco vjelo
13    IF a0<5000 then //bily pruh dole, CERVENA
14      h:=0;
15    ELSIF a1<5000 then //bily pruh nahore, MODRA
16      h:=1;
17    ELSE //zadny bily pruh, BILA
18      h:=2;
19    END_IF;
20  END_IF;
21  IF not h=6 AND a0<5000 AND a1<5000 then //ochrana proti falesnym skenum
22    color:=h;
23    h:=6;
24  END_IF;
25  IF hybej.odvezl then
26    color:=6;
27  END_IF;
28 END_FUNCTION_BLOCK
```

Obr. 7 Funkční blok *barva\_FCB*

Toto řešení funkčního bloku se může na první pohled zdát jako velmi zvláštní, ale v průběhu vývoje jsem přišel na to, že IR senzory načítají černou hranu palet a až po té kódování, proto program špatně vyhodnocoval požadovanou barvu. Kvůli tomu bylo nutné pojistit danou chybu tak, že FCB čeká, dokud nenačte hodnoty černé barvy a až poté zkoumá, jestli po ní následuje nahoře nebo dole bílý pruh. V podstatě se jedná o vylepšený funkční blok hlídající náběžnou hranu.



Jakmile paleta dorazí na konec nakládací oblasti, program čeká na zmáčknutí tlačítka od uživatele, což se považuje jako pokyn, kdy byl vložen požadovaný obrobek. Až je tlačítko zmáčknuto, naplněná paleta je opačným pohybem dopravníku komory vrácena na rameno skladového dopravníku. To se následně s paletou zvedne a zatáhne.

Posledním FCB je *vraceni\_FCB*, tedy *vraceni()*, které vrací již naplněnou paletu zpět na své původní místo do regálu. Jakmile naplněnou paletu vyloží, řídicí funkce *ridici()* posune v polích na další inkrementální souřadnice a celý cyklus se znovu opakuje. Po poslední odevzdané naplněné paletě do regálu se celý ramenový dopravník vrátí do kalibrační polohy.

## 4 Závěr

Po intenzivním úsilí a vývoji, kdy jsem si musel zopakovat spoustu různých dovedností, co se týká psaní programů pro programovatelné automaty, se mi zdárně podařilo vše zprovoznit tak, jak to bylo původně zamýšleno.

Největším problémem, kterému jsem čelil, byl funkční blok *hybej\_FCB*. Původně se totiž mělo jednat o obyčejnou funkci, vůbec to neměl být FCB. Upřímně ani nevím proč mě to napadlo. Když jsem program začal kompletovat, začalo mi postupně docházet, že je něco špatně. Strukturovaný text přistupuje k funkcím tak, že kdykoliv funkce probíhá, tak se všechny proměnné v ní vyresetují. Což mi ovlivňovalo jak „pevné“, tak i pomocné proměnné. Po nalezení a opravě tohoto problému se téměř všechny mé útrapy vyřešily.

Další problém jsem řešil se zobrazováním textu na displej PLC. To jsem ale rychle vyřešil pomocí PanelMakeru přímo ve vývojovém prostředí MosaicPLC.

Závěrem bych rád dodal, že jsem velmi rád za tuto zkušenost, kdy jsem mohl řešit reálný problém. Naučil jsem se toho mnoho, jak teoreticky o fungování jednotlivých součástí, tak i prakticky o psaní programu pro PLC.

## 5 Použitá literatura a prameny

### Literatura:

1. LACKO, B., BENEŠ, P., MAIXNER, L., ŠMEJKAL, L. *Automatizace a automatizační technika 1* . 1. vydání. Praha: Computer Press, 2000, 107 s. ISBN 80-7226-246-7
2. BENEŠ, P., CHLEBNÝ, J., KRÁL, J., LANGER, J., MARTINÁSKOVÁ, M. *Automatizace a automatizační technika 3* . 2. vydání. Brno: CP Books, a. s., 2005, 287 s. ISBN 80-251-0795-7
3. HRÁZSKÝ, J. *Skripta pro výuku AuR na SPŠE, kapitola 6. – Snímače* . 1. vydání. Praha, 49 s.

### Internetové zdroje:

4. CP-1014. Online. Datum publikace neuvedeno. Dostupné na:  
<https://catalog.tecomat.cz/produkt/cp-1014>
5. IB-1301. Online. Datum publikace neuvedeno. Dostupné na:  
<https://catalog.tecomat.cz/produkt/ib-1301>
6. OS-1401. Online. Datum publikace neuvedeno. Dostupné na:  
<https://catalog.tecomat.cz/produkt/os-1401>
7. Světelné závory – fototranzistor. Online. Datum publikace neuvedeno. Dostupné na:  
<https://www.fischertechnik.de/en/products/spare-parts/electronics/36134-photo-transistor-assembled-yellow>
8. Koncové spínače. Online. Datum publikace neuvedeno. Dostupné na:  
<https://www.fischertechnik.de/en/products/spare-parts/electronics/37783-mini-switch-black>
9. Automatizovaný sklad. Online. Datum publikace neuvedeno. Dostupné na:  
<https://www.fischertechnik.de/en/products/learning/training-models/536631-edu-automated-high-bay-warehouse-24v-education>

## 6 Příloha – Celý program

```
VAR_GLOBAL
cidloAnalog0 at %XW34:int;
//dolni cidlo na dopravniku- oznava barevny kod na palete
cidloAnalog1 at %XW46:int;
//horni cidlo na dopravniku-poznava barevny kod na palete
cidlo_horiz at %X100.4:bool; //mechanicke cidlo na horizontalni ose prepravniku
cidlo_verti at %X100.7:bool; //mechanicke cidlo na vertikalni ose prepravniku
cidlo_rameno_zad at %X10.7:bool;
//mechanicke cidlo v zadni casti ramena prepravniku
cidlo_rameno_pre at %X10.6:bool;
//mechanicke cidlo v predni casti ramena prepravniku
doprav_cidlo_zad at %X100.6:bool; //svetelna brana dopravniku vzadu
doprav_cidlo_pre at %X100.5:bool; // svetelna brana dopravniku vpredu
horiz_ink at %XL106:udint;
verti_ink at %XL116:udint;
ridSlovo at %YW20:uint;
ridSlovo1 at %YW26:uint;
//zk0 at %X101.0:bool; //pro manualni ovladani
//zk1 at %X101.1:bool;
//zk2 at %X101.2:bool;
//zk3 at %X101.3:bool;
doprav_do at %Y32.4:bool; //pohon dopravniku - dovnitr
doprav_ve at %Y32.5:bool; //pohon dopravniku - ven
horiz_vl at %Y32.6:bool; //horizontalni, vlevo:ccw
horiz_vp at %Y32.7:bool; //horizontalni, vpravo:cw
verti_do at %Y33.0:bool; //vertikalni, dolu
verti_na at %Y33.1:bool; //vertikalni, nahoru
ram_vy at %Y33.2:bool; //rameno, vysun
ram_vs at %Y33.3:bool; //rameno, vsun
PolohaVerti: array[0..3] of udint:=[310,2050,4030];
//ulozene vertikalni hodnoty regalu
PolohaHoriz: array[0..3] of udint:=[3629,6462,9300];
//ulozene horizontalni hodnoty regalu
nalozeno at %X101.0:bool;
kalibruj:bool:=0;
casovac:TON;
hlavni_barva:byte:=3;
END_VAR
FUNCTION kalibrace:bool //funkce na kalibraci jednotky
  IF not cidlo_rameno_zad then //zatahne rameno
    ram_vs:=1;
  ELSE
    ram_vs:=0;
    IF not cidlo_horiz then //narazi na horizontalni snimac
      horiz_vp:=1;
    ELSE
      horiz_vp:=0;
    END_IF;
    IF not cidlo_verti then //narazi na vertikalni snimac
      verti_na:=1;
    ELSE
      verti_na:=0;
    END_IF;
  END_IF;
  IF cidlo_rameno_zad AND cidlo_horiz AND cidlo_verti then
//vrati, ze probehla funkce
    kalibrace:=1;
  END_IF;
END_FUNCTION
```

```

(*)
FUNCTION manualni:bool //funkce na manualni ovladani skladu
    horiz_vp:=zk0 and not zk3;
    verti_na:=zk1 and not zk3;
    ram_vs:=zk2 and not zk3;
    horiz_vl:=zk0 and zk3;
    verti_do:=zk1 and zk3;
    ram_vy:=zk2 and zk3;
END_FUNCTION
*)
FUNCTION_BLOCK barva_FCB //FCB na poznavani barevneho kodovani na boku palet
    VAR_INPUT
        a0:int;
        a1:int;
    END_VAR
    VAR
        h:int:=6; //buffer pro color
    END_VAR
    VAR_OUTPUT
        color:int:=6;
    END_VAR
    IF (h=2 or h=6) and (a0>=5000 or a1>=5000) then //neco vjelo
        IF a0<5000 then //bily pruh dole, CERVENA
            h:=0;
        ELSIF a1<5000 then //bily pruh nahore, MODRA
            h:=1;
        ELSE //zadny bily pruh, BILA
            h:=2;
        END_IF;
    END_IF;
    IF not h=6 AND a0<5000 AND a1<5000 then //ochrana proti falesnym skenum
        color:=h;
        h:=6;
    END_IF;
END_FUNCTION_BLOCK

VAR_GLOBAL
    barva:barva_FCB;
END_VAR

FUNCTION_BLOCK hybej_FCB //funkce pro rozvoz palet
    VAR_INPUT
    END_VAR
    VAR
        vertikalne:bool:=0;
        horizontalne:bool:=0;
        zmacknute:bool:=0;
        plna:bool:=0;
        dole:bool:=0;
        nahore:bool:=0;
    END_VAR
    VAR_OUTPUT
        odvezl:bool:=0;
    END_VAR
    IF not zmacknute then
        IF verti_ink < 3350 AND not nahore then
//dojede na vertikalni polohu nakladaci oblasti smerem dolu
            verti_do:=1;
            dole:=1;
        ELSIF verti_ink > 3350 AND not dole then
//dojede na vertikalni polohu nakladaci oblasti smerem nahoru

```

```

        verti_na:=1;
        nahore:=1;
    ELSE
        verti_do:=0;
        verti_na:=0;
        vertikalne:=1;
    END_IF;
    IF horiz_ink>75 then //dojede na horizontalni polohu nakladaci oblasti
        horiz_vp:=1;
    ELSE
        horiz_vp:=0;
        horizontalne:=1;
    END_IF;
    ram_vy:=not cidlo_rameno_pre AND vertikalne AND horizontalne;
    //paleta se posadi na dopravnik v nakl. oblasti
    doprav_do:=doprav_cidlo_zad AND not doprav_ve AND vertikalne AND
horizontalne; //dopravnik posune paletu dovnitr
    barva(a0:=cidloAnalog0,a1:=cidloAnalog1); //sken barvy
    CASE barva.color OF //vypise pozadovanou barvu na displej PLC pres PanelMaker
        0:
            hlavni_barva:=0;
        1:
            hlavni_barva:=1;
        2:
            hlavni_barva:=2;
        6:
            hlavni_barva:=3;
    END_CASE;
    END_IF;
    IF nalozeno then //pokud je zmacklo tlacitko, nastavi pomocnou
        zmacknute:=TRUE;
    END_IF;
    IF zmacknute AND not plna then //pokud uzivatel stiskne tlacitko, ze barevny
//obrok vložil, tak se paleta vrati zpet na rameno a to se zatahne
        IF verti_ink<3400 then //rameno sjede trochu dolu
            verti_do:=1;
        ELSE
            verti_do:=0;
        END_IF;
    END_IF;
    IF zmacknute AND doprav_cidlo_pre AND not plna then
        doprav_ve:=1;
    ELSIF zmacknute AND not doprav_cidlo_pre AND doprav_cidlo_zad AND not plna then
        doprav_ve:=0;
        plna:=TRUE;
    END_IF;
    IF plna then
        IF verti_ink>3150 then //rameno jiz s paletou vyjede vyse a zatahne se
            verti_na:=1;
        ELSE
            verti_na:=0;
            ram_vs:=not cidlo_rameno_zad;
            odvezl:=cidlo_rameno_zad;
            IF odvezl then //reset vseh pomocnych pro dalsi pouziti
                vertikalne:=0;
                horizontalne:=0;
                zmacknute:=0;
                plna:=0;
                dole:=0;
                nahore:=0;
            END_IF;
        END_IF;
    END_IF;

```

```

        END_IF;
    END_IF;
END_FUNCTION_BLOCK

VAR_GLOBAL
    hybej:hybej_FCB;
END_VAR

FUNCTION_BLOCK nabirani_FCB
//FCB pro nabrani prazdnych palet a dovezeni je do nakladaci oblasti
    VAR_INPUT
        i:int;
        j:int;
    END_VAR
    VAR
        verti:udint;
        vertikalne:bool:=0;
        horizontalne:bool:=0;
        jednou:bool:=0;
        vysunute:bool:=0;
        nadzvednuto:bool:=0;
        nahoru:bool:=0;
        dolu:bool:=0;
        vpravo:bool:=0;
        vlevo:bool:=0;
    END_VAR
    VAR_OUTPUT
        probehlo:bool:=0;
    END_VAR
    IF verti_ink<PolohaVerti[i] AND not nahoru AND not vertikalne then
//posune prepravnik na vertikalni uroven
        verti_do:=1;
        dolu:=1;
    ELSIF verti_ink>PolohaVerti[i] AND not dolu AND not vertikalne then
        verti_na:=1;
        nahoru:=1;
    ELSE
        verti_do:=0;
        verti_na:=0;
        vertikalne:=1;
        IF not jednou then
            verti:=verti_ink;
            jednou:=1;
        END_IF;
    END_IF;
    IF horiz_ink<PolohaHoriz[j] AND not vpravo AND not horizontalne then
//posune prepravnik na horizontalni uroven
        horiz_vl:=1;
        vlevo:=1;
    ELSIF horiz_ink>PolohaHoriz[j] AND not vlevo AND not horizontalne then
        horiz_vp:=1;
        vpravo:=1;
    ELSE
        horiz_vl:=0;
        horiz_vp:=0;
        horizontalne:=1;
    END_IF;
    IF horizontalne AND vertikalne AND not cidlo_rameno_pre AND not vysunute then
        ram_vy:=1;
    ELSIF horizontalne AND vertikalne then
        ram_vy:=0;

```



```

vysunute:=1;
IF verti_ink>(verti-90) then //nadzvedne rameno uz s paletou
    verti_na:=1;
ELSE
    verti_na:=0;
    nadzvednuto:=1;
END_IF;
IF nadzvednuto then
    IF not cidlo_rameno_zad then
        ram_vs:=1;
    ELSE
        ram_vs:=0;
        probehlo:=1; //vrati, ze FCB probehl
        jednou:=0; //reset vseh pomocnych pro dalsi pouziti
        nadzvednuto:=0;
        vysunute:=0;
        nahoru:=0;
        dolu:=0;
        vpravo:=0;
        vlevo:=0;
        vertikalne:=0;
        horizontalne:=0;
    END_IF;
END_IF;
END_FUNCTION_BLOCK
VAR_GLOBAL
    nabirani:nabirani_FCB;
END_VAR
FUNCTION_BLOCK vraceni_FCB
//FCB pro odvoz a odkladani plnych palet na puvodni pozici
VAR_INPUT
    i:int;
    j:int;
END_VAR
VAR
    verti:udint;
    jednou:bool:=0;
    vysunute:bool:=0;
    nahoru:bool:=0;
    dolu:bool:=0;
    vertikalne:bool:=0;
    horizontalne:bool:=0;
END_VAR
VAR_OUTPUT
    vraceno:bool:=0;
END_VAR
IF verti_ink>PolohaVerti[i] AND not dolu AND not vertikalne then
//posune prepravnik na vertikalni uroven, nahoru
    verti_na:=1;
    nahoru:=1;
ELSIF verti_ink<PolohaVerti[i] AND not nahoru AND not vertikalne then
//posune prepravnik na vertikalni uroven, dolu
    verti_do:=1;
    dolu:=1;
ELSE
    verti_na:=0;
    verti_do:=0;
    vertikalne:=1;
    IF not jednou then
        verti:=verti_ink;
    
```

```

        jednou:=1;
    END_IF;
END_IF;
IF horiz_ink<PolohaHoriz[j] AND not horizontalne then
//posune prepravnik na horizontalni uroven
    horiz_vl:=1;
ELSE
    horiz_vl:=0;
    horizontalne:=1;
END_IF;
IF not cidlo_rameno_pre AND not vysunute AND horizontalne AND vertikalne
then //vysune rameno
    ram_vy:=1;
ELSIF cidlo_rameno_pre AND not vysunute AND horizontalne AND vertikalne then
    ram_vy:=0;
    vysunute:=1;
END_IF;
IF verti_ink<(verti+50) AND vysunute then
//vysunute rameno posune dolu, paleta zustane v regalu
    verti_do:=1;
ELSIF vysunute then
    verti_do:=0;
    IF not cidlo_rameno_zad then
        ram_vs:=1;
    ELSE
        ram_vs:=0; //zasune rameno
        vraceno:=1; //vrati, ze FCB probehl
        jednou:=0; //reset vseh pomocnych pro dalsi pouziti
        vysunute:=0;
        nahoru:=0;
        dolu:=0;
        vertikalne:=0;
        horizontalne:=0;
    END_IF;
END_IF;
END_FUNCTION_BLOCK
VAR_GLOBAL
    vraceni:vraceni_FCB;
END_VAR
VAR_GLOBAL
    horizontal:int:=0;
    vertikal:int:=0;
    blokace:bool:=0;
END_VAR
FUNCTION ridici:bool
    IF not blokace then
        IF not nabirani.probehlo then
            nabirani(i:=vertikal, j:=horizontal);
        ELSE
            IF not hybej.odvezl then
                hybej();
            ELSE
                IF not vraceni.vraceno then
                    hlavni_barva:=3;
                    vraceni(i:=vertikal, j:=horizontal);
                ELSE
                    IF horizontal<2 then //podminka pro posun v polich
                        horizontal:=horizontal+1; //posun v horizontalnim poli
                        nabirani.probehlo:=0;
                        hybej.odvezl:=0;
                        vraceni.vraceno:=0;
                    END_IF;
                END_IF;
            END_IF;
        END_IF;
    END_IF;
END_FUNCTION

```

```

        END_IF;
        IF horizontal=2 AND vraceni.vraceno then
            IF vertikal<2 then
                vertikal:=vertikal+1;
                horizontal:=-1;
            ELSE
                blokace:=1;
            END_IF;
        END_IF;
    END_IF;
END_IF;
END_IF;
ELSE
    kalibrace();
END_IF;
END_FUNCTION
PROGRAM prgMain
IF not kalibruj then //uvodni kalibrace, zajisti, ze po spusteni se system vzdy
nejdrive zkalibruje
    kalibruj:=kalibrace();
ELSE
    IF casovac.Q then //seridi nejdrive inkrementalni snimace, po te program dale
pokracuje
        ridSlovo:=1; //spusti pocitani inkr. snimacu
        ridSlovo1:=1;
        //manualni();
        ridici(); //hlavni ridici funkce
    ELSE
        ridSlovo:=2; //vyresetuje hodnotu inkr. snimacu
        ridSlovo1:=2;
        casovac(IN:=1, PT:=T#1s); //prodleva pro korektni serizeni snimacu
    END_IF;
END_IF;
END_PROGRAM

```