



Střední průmyslová škola technická

Jablonec nad Nisou, Belgická 4852, příspěvková organizace



Středoškolská technika 2016

**Setkání a prezentace prací středoškolských studentů
na ČVUT**

NÁVRH A REALIZACE AUTOMATICKÉHO OVLÁDÁNÍ FESTO™ ROBOTINA

Milan Čácha

Střední průmyslová škola technická

Belgická 4852, Jablonec nad Nisou

Anotace

Tato práce pojednává o návrhu automatického ovládní **Festo™ Robotina** a jeho vizualizaci v příslušném programu, který je pro tuto činnost vhodný.

Klíčová slova: robot , robotino , vizualizace , programování.

Annotation

This work deals with designing automatic festo robotino control of movement and its visualization in the relevant programme, which is used in this work.

Keywords : Robot, robotino , visualization, programming.

Obsah

Obsah.....	5
Použité značky a symboly.....	6
1. Úvod.....	8, 9
2. Teoretická část.....	9
2.1. Konstrukční část Robotina	9
2.1.1. Podvozek.....	9
2.1.2. Pohonná jednotka.....	10
2.1.3. Řídicí jednotka	11
2.1.4. Rozšiřující porty.....	11
2.2. Senzory a jejich typy	12
2.2.1. Optické inkrementální senzory úhlu natočení.....	12
2.2.2. Triangulační dálkoměry SHARP GD2D120.....	13
2.2.3. Indukční senzory	14
2.3. Navigace.....	15
2.4. Webkamera.....	16
3. Praktická část	17
3.1. Programy pro realizaci úloh	17
3.2. Robotino® View3	18
3.3. Popis zadání praktické části	19
3.4. Realizace praktické části	20-27
4. Závěr	28
5. Použité zdroje	29
6. Seznam použitého software	30

7.	Seznam příloh	30
8.	Licenční ujednání.....	31

Použité značky a symboly

F – síla,	F	[N]
I – elektrický proud,	I	[A]
m – hmotnost,	m	[kg]
M – moment síly,	M	[Nm]
P – výkon,	P	[W]
R – elektrický odpor,	R	[Ω]
U – elektrické napětí,	U	[V]
ρ – rezistivita,	ρ	[Ωm]
ω – úhlová rychlost,	ω	[$\Omega\text{ s}^{-1}$]
atd.	a tak dále	
mj.	mimo jiné	
tj.	to jest	
IR	infra-red = infračervené	
CPU	central processing unit – procesor	
®	označení registrace	
™	trademark – označení obchodní značky	
tzn.	to znamená	

1. Úvod

Tato práce, která je zaměřená na problematiku Robotiky a všeobecných základních atributů by měla objasnit jak a z čeho se tyto věci skládají. V průběhu práce se vám pokusím přiblížit toto vysoce perspektivní odvětví, které se dnes využívá v podstatě všude, kde je potřeba buď z nějakého důvodu ušetřit pracovní sílu a nebo tam kde to ani jinak nelze z důvodu vysokého nebezpečí úrazu. Pro příklad bych mohl uvést Automobilový Průmysl, kde se ve výrobním cyklu využívá velký potenciál těchto robotů ať už to jsou automatické linky, které se pohybují a posouvají díly buď po příslušných pracovištích a nebo už i celé celky, jako jsou třeba karosérie. Pro přesný příklad plně automatizovaného pracoviště kde se používají plně automatické roboty můžeme lakovnu. V lakovně jsou plně automatizované robotické ruky které mají příslušný nástroj a nanaší rovnoměrně ve vrstvách lak na karoserii vozidla. Pro základní popsání této problematiky nám ovšem poslouží takzvané Robotino od firmy FESTO™, která vynalezla tento zmenšený typ robota, který slouží hlavně pro výukové účely na odborných školách zabývajících se touto problematikou. Tohoto robota jak jsem již zmínil zkonstruovala firma FESTO™, která byla založena roku 1925 Albertem Fezerem s Gottliebem Stollem. Původně šlo o společnost působící v dřevozpracujícím průmyslu, konkrétně se zabývala výrobou dřevozpracujících nástrojů používaných tehdejšími obyvateli i továrnami. Prvním významnějším produktem společnosti byla světově první přenosná, benzinem poháněná motorová pila. Později firma expandovala svým polem působnosti také do pole automatizace. Automatizace je pojem, který se využívá pro řízení a ovládní automatických procesů s hlavním účelem ušetřit pracovní sílu a nebo předejít jejímu nebezpečí zranění, jak jsem zmínil výše. Další velkou výhodou a hlavně část jejich využití je prvek opakovatelnosti s příslušnou přesností na požadované parametry. Roku 2000 byla firma rozdělena na dvě části, kde první se zabývala dál výrobou zemědělské techniky, ale pod jménem Festool, čímž se oddělila od mateřské společnosti Festo. V dnešní době festo zaměstnává přes 15000 lidí po celém světě. Hlavní filozofie firmy je rozdělena do 3 bloků.

A.) Automatizace Výroby: Tato část mapuje trh a na základě toho vyvíjí nové prvky pro zdokonalení pracovních procesů a ušetření ekologických nákladů a dopadů jakožto trendu dnešní doby.

B.) Automatizace procesů a postupů: Postupy pro konkrétní výrobu a manipulaci a logistiku pro specifické typy materiálů. Pro příklad můžeme uvést manipulaci s plyny, a kapalnými látkami. Jde o koncepci určenou do extrémně náročných podmínek, která musí odolat specifickým vlivům a okolnostem.

C.) Školení a vzdělávání : FESTO poskytuje konzultace a také výukové zdroje. Jedná se o takzvané vzdělávání FESTO Didactic a je firmou poskytováno po celém světě ve specifických školicích zařízeních a účastní se jich cca 42 000 lidí ročně. Zaměřuje se na průmyslovou a procesní automatizaci => Zefektivnění procesů v údržbě, výrobě a samotné konstrukci.

2. Teoretická část

2.1 Konstrukční část Robotina

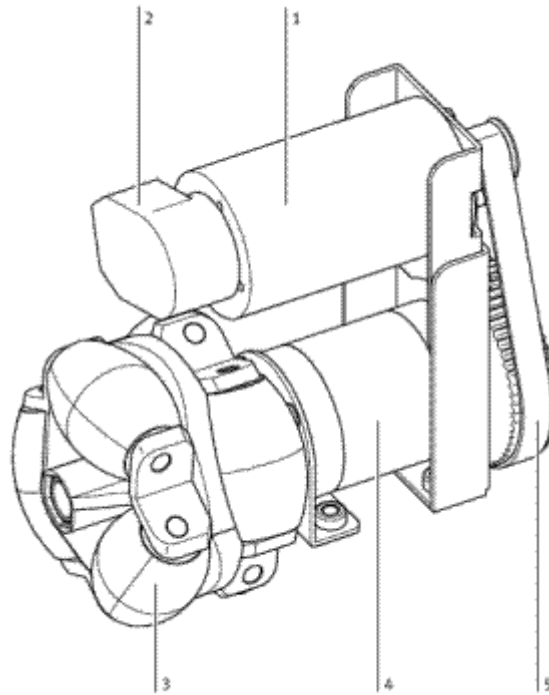
Konstrukci robotina můžeme rozdělit do zhruba 13ti částí. Jelikož se robot skládá z mnoha částí rozdělíme si je do kapitol, které nám popíší vždy příslušnou část a na konci shrneme do celku.

2.1.1 Podvozek

Podvozek robotina se skládá z kruhového nerezového půdorysu o průměru 370mm a je svařen. Na podvozku robotina se nacházejí tři výkonné nezávisle na sobě pracující pohonné jednotky které jsou pootočené o 120°, kde jejich největší výhodou je nezávislost na sobě a separátní řízení každého z nich. Tato funkce ovšem potřebuje velice přesné řízení každého z nich přes PID regulátory. Tyto regulátory získávají důležité podněty z motorů pomocí indukčních senzorů, které se nacházejí na každém motoru a vysílají údaje o otáčkách a natočení motorů. Dále se na podvozku nacházejí dva akumulátory, které se nacházejí mezi dvojicí motorů o celkovém napětí 24V, tudíž má každý z nich 12V. Robotino může být pro delší provozuschopnost vybaveno třetím akumulátorem, ale jeho aplikace je složitá a pro normální komerční použití na výukové zdroje postačují akumulátory dva.

2.1.2 Pohonná jednotka

Je tvořena třemi motory, jak jsem již zmínil o kapitole výše. Tyto motory jsou Stejnoseměrné-komutátorové, které přenášejí svůj potenciál přes Planetovou převodovku s převodovým poměrem 16:1 dále je motor vybaven inkrementální snímačem, který snímá aktuální parametry motoru a předává je řídicím regulátorům. Viz. Obrázek



1. Stejnoseměrný motor
2. Planetová převodovka
3. Gumové kolečko (roller)
4. Inkrementální snímač
5. Ozubený řemen

2.1.3 Řídící jednotka

Základní jednotka celého motoru, která řídí veškeré dění a shromažďuje data, ze všech senzorů. Pro tento typ Robotina je použita jednotka jednodeskového počítače standard PC104 s operačním systémem RT Linux. Rozhraní mezi ním a vlastním hardwarem robota zajišťuje speciální modul s jednočipovým mikrokontrolerem Atmel. Na tento modul lze připojit i další senzory a měřící přístroje. Dále po obvodu robotina je umístěn koncový spínač pro případ kontaktu s překážkou, kde tento spínač zastaví celý program indukčnostní snímač polohy. Dále po obvodu je rovnoměrně rozmístěno 9 infračervených triangulačních dálkoměrů Sharp s maximálním rozsahem 4-30cm. Dva limitní optické snímače a jeden analogový indukčnostní snímač polohy. Přímou k řídicí jednotce je připojena USB kamera pro získání obrazové informace. Komunikace s PC je možná přes Wifi modul. Robotino můžeme ovládat buď přes počítač, kde je robotino řízeno přes příslušný program z počítače a nebo přímo přes software nahraný v robotinu bez participace externí počítačové jednotky.

2.1.4 Rozšiřující porty

Jelikož je na robotinu spousta rozšiřujících individuálních možností jsou zde i rozšiřující rozhraní pro připojení dalších externích systému a periférií , jako jsou například senzory.

Rozšiřující deska obsahuje: 8. Analogových vstupů

8. Digitálních vstupů (může být i více v závislosti na napětí)

2. relé



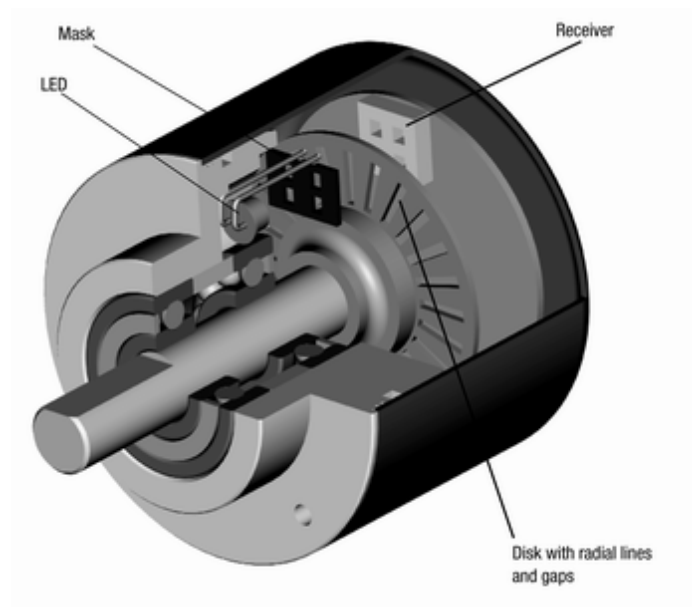
Rozšiřovací deska napojená na základní desku.

2.2 Senzory a jejich typy.

2.2.1 Optické inkrementální senzory úhlu natočení

Jde o důležitý senzor, který nám mapuje úhel natočení jednotlivých motorů.

Základem tohoto senzoru je tkz. Pulzní disk , který je vyroben převážně ze skla , ale může být vyroben také z plastu či jiných materiálu v závislostech na požadavcích a přesném využití. Tento disk je spojen mechanicky s hřídelí, která prochází osou senzoru. Disk obsahuje pole a to Světlá a Tmavá. Světlo generují diody, které většinou pracují v infračerveném spektru. Světlo prochází membránou , ale zároveň tímto pulzním diskem, které je zachyceno na druhé straně tkz. Fotodetektorem. Poté , když se hřídel otáčí, střídavě zacloňuje a odcloňuje.(Fotodetektor). Pulsy jsou následně zpracovány a převedeny na elektrický signál. Následně podle toho kolik má tmavých a světlých polí, zjistíme počet pulsů na jednu otáčku. Tento parametr je u těchto snímačů stěžejní. Opět tu máme dvojí dělení a to na jednobytové, které se v dnešní době příliš nevyužívají. (Mají jednu stopu na pulzním disku.) Dnes již využívané Tři kanálové pracují na takovém principu , že 2 signály jsou vzájemně posunuty o 90° ,díky čemuž dokážeme rozpoznat směr otáčení. Třetí puls tohoto typu slouží jako „Nulový Puls“. Tyto snímače mají jednu nevýhodu a to je ta , že když dojde ke ztrátě napájení , tak si nepamatují svoji poslední polohu.



2.2.2 Triangulační dálkoměry SHARP GD2D120

Toto Robotino® je osazeno devíti infračervenými dálkoměry , které využíváme pro měření určitých vzdáleností od předmětů. Tyto senzory fungují tak, že senzor vyšle paprsek , který se následně odrazí od případné překážky zpět do přijímače. Tyto senzory jsou namontovány po obvodu a jak jsem již říkal v devíti kusech pod úhlem 40° a je s nimi možno detokovat na vzdálenost 4 až 30cm. Senzory jsou specifické a individuálně nastavitelné v závislosti na použití, protože každý může být v programátorském rozhraní naprogramován na různou reakční vzdálenost a docílit tím plně simultálního ovládání každého z nich. Následně díky této funkci dokáže Robotino® reagovat a předejít tak nárazu.

Ukázka takového senzoru Sharp.



2.2.3 Indukční senzory

Indukční snímač jenž je nainstalován na spodu Robotina nám umožňují snímat kovové předměty jenž se nacházejí pod samotným Robotem. Jak je již vidno z názvu tyto senzory využívají pro svoji funkci magnetických vlastností prostředí, které obklopují cívku. Můžeme hovořit o dvou typech indukčnosti a to je indukčnost vlastní, když máme pouze jednu cívku a pak indukčnost vzájemnou, která je vyvolána přítomností dvou cívek. Indukční senzory mají spíše doplňkovou roli a mají tu nevýhodu, že detekují pouze kovové předměty.



2.3. Navigace

Automatická navigace je jedním ze základních stavebních kamenů pro dosažení plné inteligence a tím čím dál více se rozrůstající autonomity robotů jako takových. Jednou z možností, která se nám naskytá je tkz. "Dead Reckoning". Tato metoda je definována matematicky a používá se pro určování současné polohy objektu pomocí výpočtu dráhy a rychlosti v čase. Aplikace této metody se nazývá Odometrie. Odometrie nám odhaduje aktuální polohu pohybujícího se objektu v závislosti k jeho počáteční poloze , ve kterém se nacházel a k určení dráhy potažmo vzdálenosti , kterou sledovaný objekt urazil od počátku pohybu. Používá se právě u Robotů, které jsou postaveny na podvozku který je vybaven koly. Provedené bývá většinou pomocí Inkrementálních snímačů, které jsem předesílal o několik stránek výše v příslušném bloku Senzorů Robotina. S touto technologií ovšem ruku v ruce přicházejí také chyby, které tato technologie nese v závislosti na odhadované poloze, protože otáčky jednotlivých kol mohou být převedeny na lineární posuv. Ovšem v některých případech je právě toto určování polohy jediná možnost samotné navigace Robota jakožto celku. Faktory , které nám mohou vytvářet chyby jsou:

1. Rozdílné průměry kol.

2. Výchylka kol.

3. Aktuální průměr , který se liší od nominálního.

4. Jízda po nerovném povrchu nebo přes neočekávané překážky.

Dalším navigačním modulem , který můžeme použít je navigační modul NorthStar® jenž využívá optické lokalizace robota , který koná pohyb v uzavřeném prostoru (místnosti). Ovšem nejvyužívanější metodou je Odometrie , která není tak přesná ale v kombinaci s dalšími externími pomocnými prvky jako jsou senzory dokáže vyhovět.

2.4. Webkamera

Robotino může být osazeno také kamerovým systémem , který dokáže snímat okolí a přenášet obraz do vhodného rozhraní. (RoboView). Tato kamera může být také využita při realizaci některých úloh , kde nám nahrazuje snímače, nebo je to jejich záložní varianta.

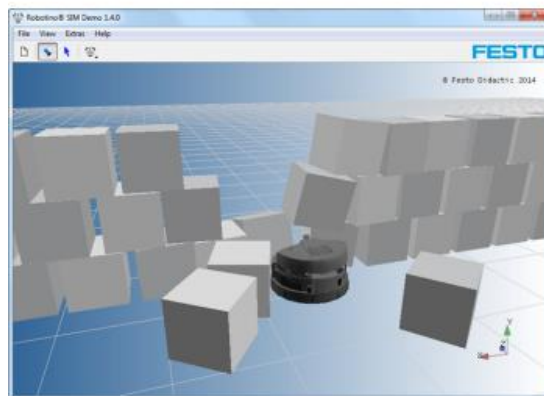
Konkrétně v úlohách, kde kamera má snímat libovolný barevný pásek na zemi popř. stolu, kde se robotino nachází a následně podle vyhodnocení obrazu z kamery také následovat zadanou pásku a udržovat se v její trajektorii. V závislosti na ochotě investovat do příslušné kamery se nám v dnešní době naskytá celá řada různých kamer od různých výrobců v různých rozlišeních.



3. Praktická část

3.1. Programy pro realizaci úloh.

Jak to již v dnešním individualistickém a různorodém světě chodí i zde se můžeme setkat s rozdílnými softwary programátorského rozhraní Robotina ®. Já osobně ovšem vyznávám originální licencované produkty od firem nebo institucí , které danou věc vytvořili. Proto jsem i v tomto případě volil oficiální software v kterém programujeme a také simulační software ve kterém provádíme simulaci a následnou kontrolu rozpracované úlohy. Simulační program nám v tomto případě slouží jako zpětná vazba. Software pro programování úloh se nazývá Robotino® View3. Simulační software se nazývá Robotino® SIM DEMO. Oba programy komunikují mezi sebou pomocí Wi-Fi modulu, ať už domnělého k propojení Robotino® View s Robotino® SIM DEMO, kde využíváme IP adresy daného robota spolu s kombinací vygenerovaného portu z programu Robotino® SIM DEMO, který zadáme jako párovací pro virtuální vizualizaci. V praxi při operacích s fyzickým Robotem využíváme skutečného Wi-Fi připojení, kde propojíme program Robotino® View3 s robotem a poté můžeme následně začít pracovat. Robot je velice náchylný na Wi-Fi a může docházet v určitých prostorech , kde je přítomných více Wi-Fi signálů k samotnému kolísání signálu a následným výpadkům komunikace mezi programem a Robotem samotným. Robotino® View 3 využívá programovacího jazyku FBD neboli tkz. funkčních bloků pro přístup k hardwaru robota a zpracování informací a pro jejich samotné zobrazení. Dále ovšem můžeme programovat i v jazycích C++ a Java. V programu Robotino® SIM DEMO poté můžeme vidět naprogramované simulace toho , jak se robot chová. V tomto programu máme na výběr i řadu překážek a různých vjemů na, které robot dokáže reagovat svými vlastními snímači.



3.2. Program Robotino® View 3

Po zapnutí tohoto programu a případného následného párování, které v případě programování není zcela nezbytné se můžeme rovnou pustit do programování, protože programování robotina je založeno na vkládání funkčních bloků a jejich pomocných prvků do určitých zapojení mezi sebou. Tyto vazby jsou důležité pro následnou správnou funkci Robotina. Jedná se o například o bloky motorů a jejich hlavního řízení (OMNIDRIVE), které je následně propojeno s každým motorem individuálně. Dále se v programu mohou nacházet různé konstanty, které nám mohou ovládat rychlost robota. Motory jsou díky systému řízení OMNIDRIVE na sobě nezávislé, a tak pomocí vhodných propojení můžeme ovládat každý zvlášť. Toto se využívá, když chceme s robotem jet "šejdrem". Dále musíme mít tzv. Timery neboli časovače , které nám zajistí , že kroky na sebe budou plynule navazovat. Dále můžeme samozřejmě využívat celou škálu senzorů, které nám mohou mapovat okolí a následně vyhodnocovat pozici Robota. Příkladem těchto senzorů mohou být Dálkoměry , které má robotino po svém obvodu a kryjí celý radius okolo samotného robotina. Tyto senzory jsou umístěny ve 40° radiusech a je jich celkem 9, aby kryly celý obvod okolo Robotina. Tyto dálkoměry se hojně využívají v senzometrických úlohách , které se zaměřují hlavně na snímání okolí a jejich následným reakcím samozřejmě ruku v ruce s ovládáním. Příkladem mohu uvést úlohu, při které robot kopíruje vytyčenou ohraničenou plochu pomocí dálkoměrů, kde plocha mezi robotinem a bariérou může být naprogramována na libovolnou hodnotu mezi 4-30cm a následnou reakci každého senzoru na vytyčenou překážku v trajektorii robota. Tento program nabízí opravdu širokou škálu možností vizualizace a tím odvozené individualizaci každého programátora. Po celém světě se koná mnoho akcí, na kterých programátoři prezentují svá díla a závodí mezi sebou.

3.3. Popis zadání praktické části.

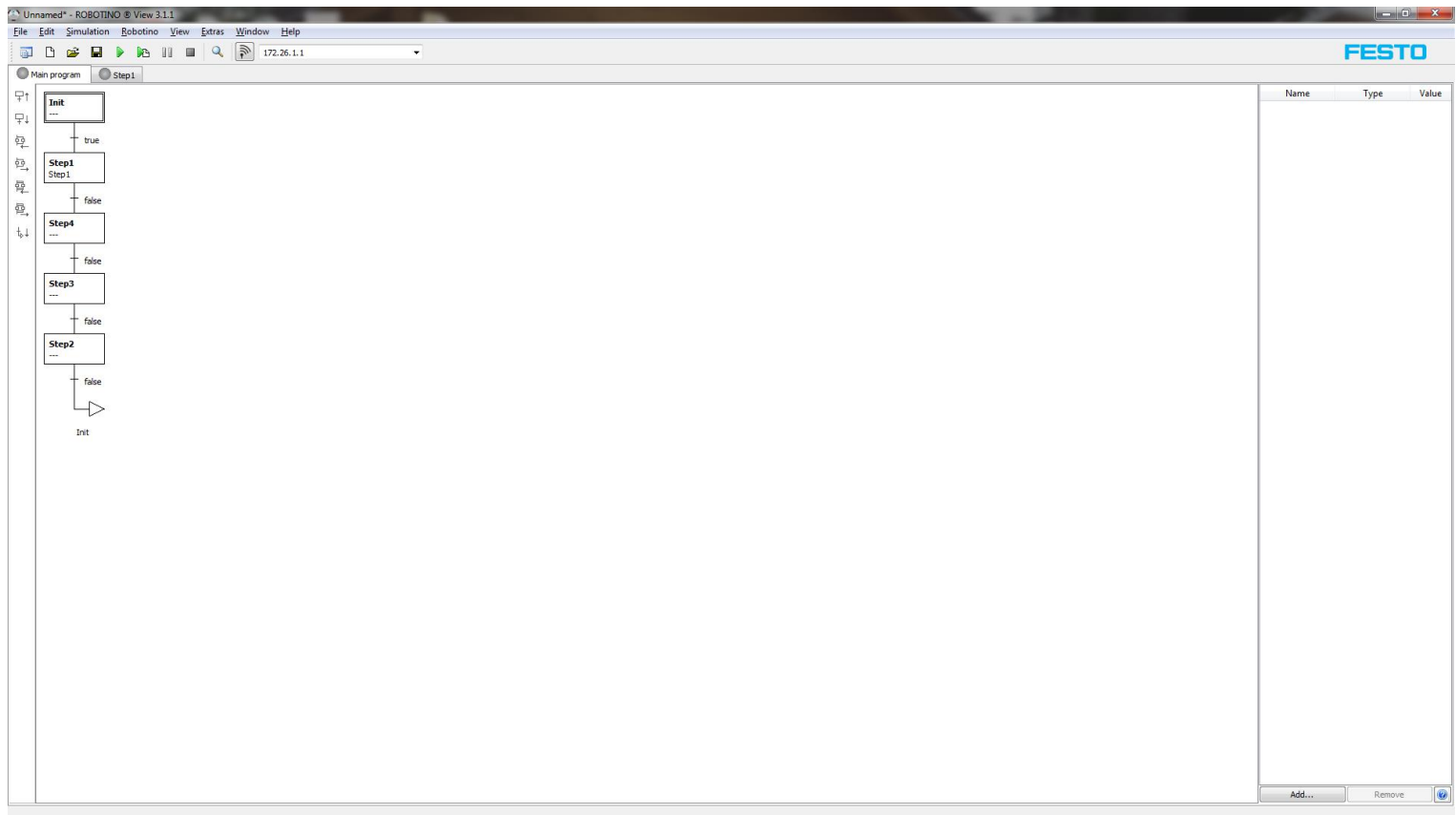
Moje praktická část jenž mi byla zadána se zabývá základní tematikou řízeného ovládání motorů a jejich následnému pohybu v libovolném směru. Úloha, kterou jsem vypravoval se zabývá řízením všech tří příslušných motorů, kterými je Robotino osazeno. Cílem úlohy je, aby robotino plně automaticky po zapnutí programu vykonávalo pohyb po přesném čtverci v nekonečné smičce, aniž by vytvářelo chyby, které by ovlivňovali trajektorii, po které se Robotino pohybuje. Program se bude skládat celkem ze čtyř fází plus dalších pomocných fází. Pojmenovaných X1, Y1, X2, Y2. Kde X1 nám bude zobrazovat pohyb doprava po směru osy X. Y1 nám bude zobrazovat pohyb po ose Y směrem vzhůru. X2 bude zobrazovat pohyb doleva opět po ose X. Y2 bude zobrazovat pohyb po ose Y směrem dolů pro dokončení čtverce.

Prezentace této úlohy v této závěrečné práci bude realizována pomocí obrázků z příslušného programu Robotino® View 3 a následných postupů v prezentované úloze, na kterou budou navazovat vysvětlivky co jaký obrázek znamená a v jaké fázi se celá úloha nachází.

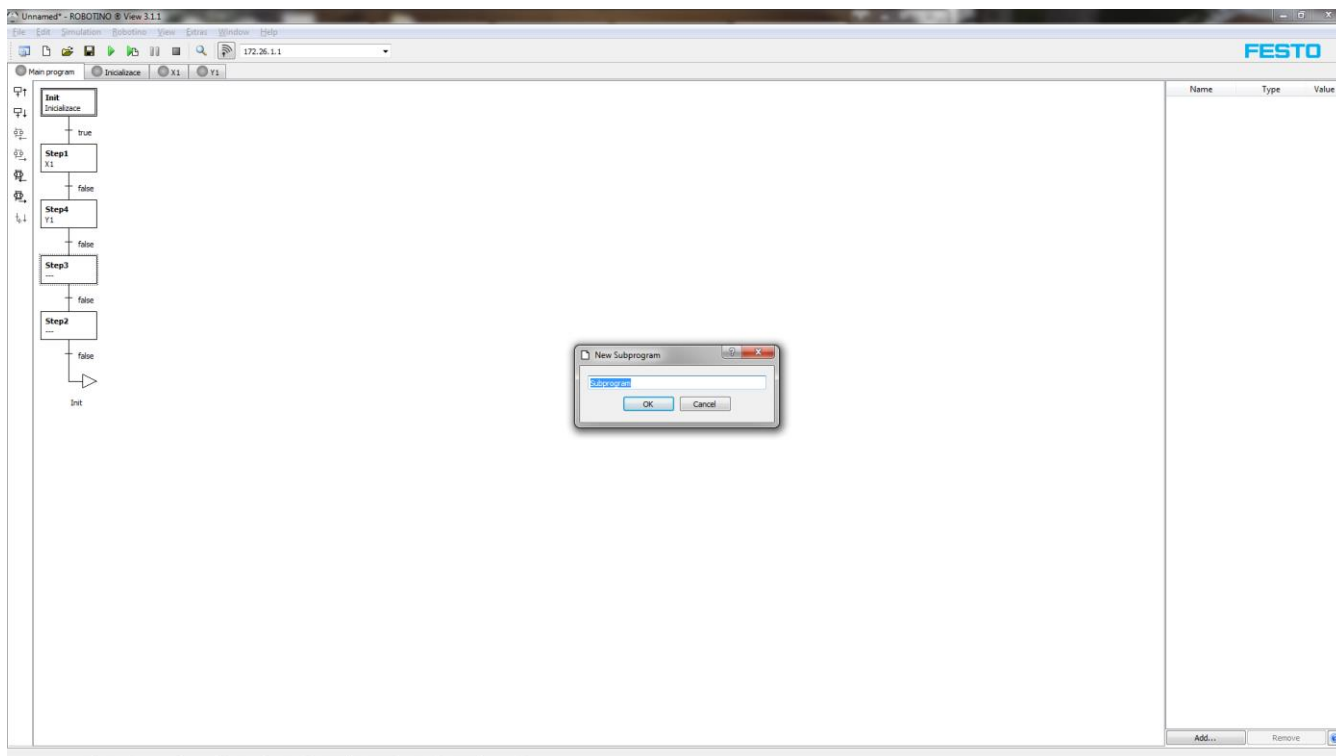
3.4. Realizace praktické části.

Zde na tomto obrázku můžeme vidět úplný základ, kde my musíme vytvořit hlavní segmenty inicializace , která je takovým spouštěčem celé úlohy. Vytvoříme tolik základních bloků kolik v podstatě bude robot vykonávat kroků (posuv doprava , doleva, nahorů, dolů atd.) Jednotlivé bloky si pojmenujeme podle příslušných kroků které reprezentují. Tím máme vytvořen prozatím hlavní program, na který navazuje vše ostatní. K tomuto "hlavnímu programu" se budeme muset vrátit posléze kvůli nastavení Timerů pro každou navazující úlohu. (slouží ke stabilizaci pohybu při napojení dalších kroků v realizaci)

KROK 1



shrnutí kroku 1: Vytvořit akční členy.

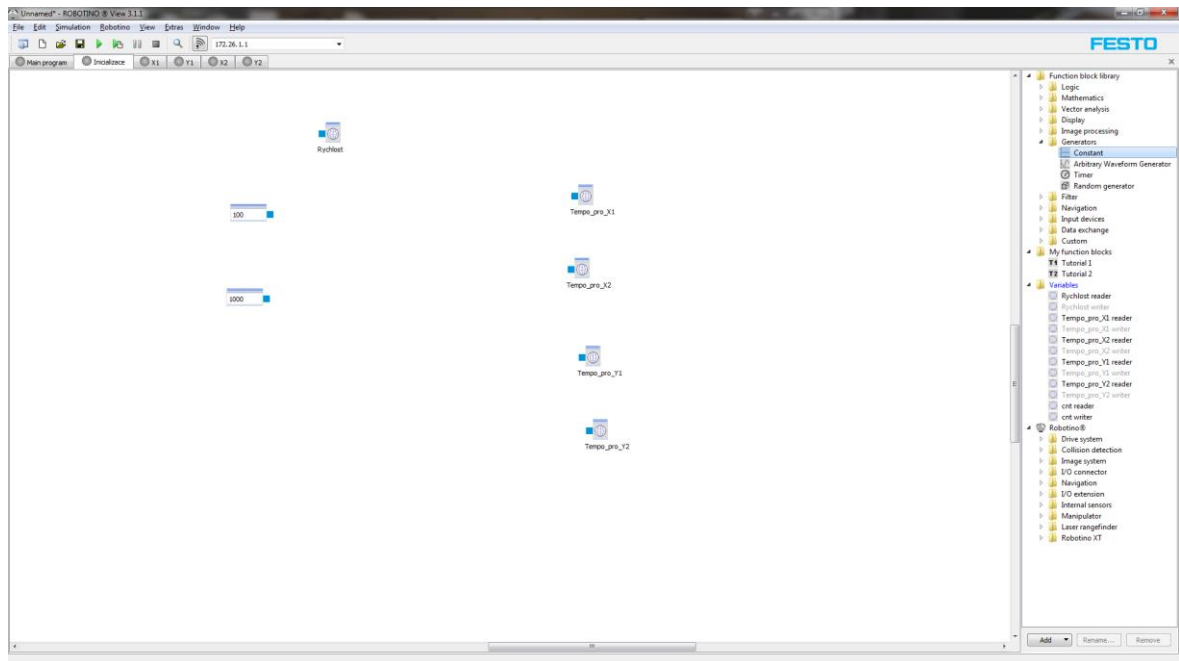


KROK 2

Zde můžeme vidět krok 2 a to již zmíněné pojmenovávání, po kterém se nám na notifikační liště zobrazují další odrážky, což jsou vytvořené kroky, do kterých vložíme jednotlivé motory a ovládání. Můžeme si to představit jako takové podprogramy. Jak je na tomto obrázku vidno STEP1 je pojmenován jako X1 a přiřazen do notifikační lišty. Pojmenování X a Y je vybráno pro větší přehlednost, protože jednotlivé záložky na notifikační liště vzniklé tímto pojmenováním provádí své pohyby po těchto osách a pohyb se dá tak lépe představit.

Shrnutí krok 2: Pojmenování akčních členů tím následně automatické vytvoření jednotlivých podprogramů X a Y.

Krok 3

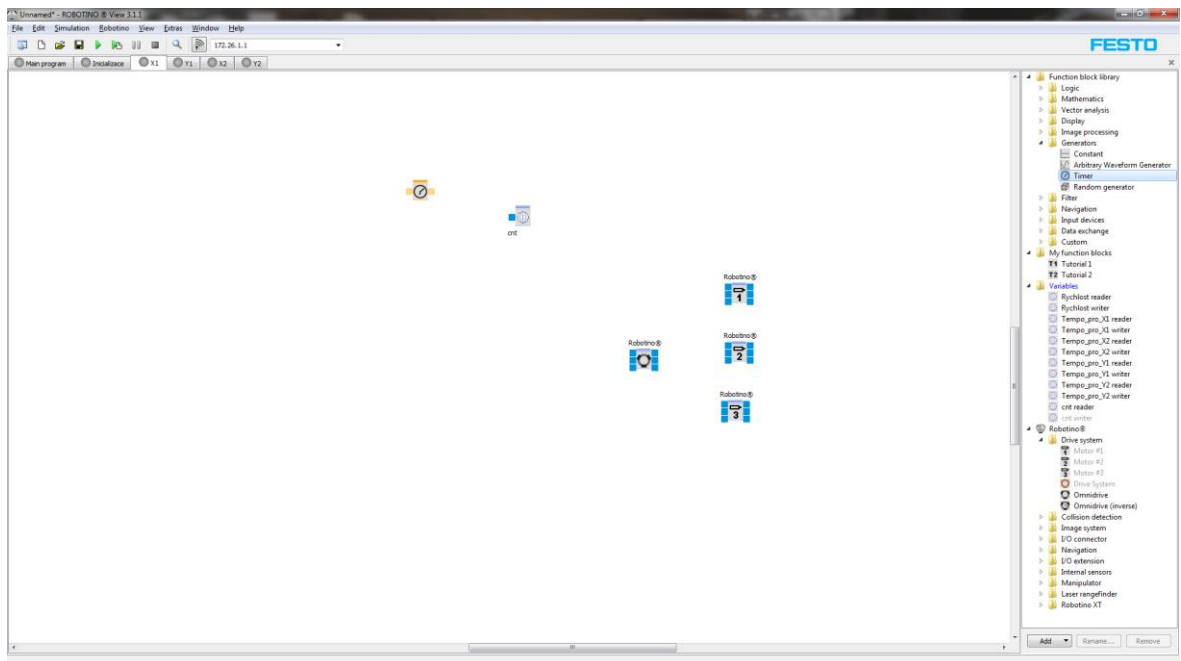


V tomto kroku můžeme vidět realizaci tkz inicializace, která nám vytváří nadřezené příkazy pro rychlost jednotlivých bloku (X1, X2...) tzn. Každá blok = změna pohybu musí mít definovanou rychlost jak rychle se má robot na danou souřadnici přesunout a to nám realizuje tento krok. Tyto bloky vezmeme z pomocného panelu , který se nachází na pravé straně. Na tomto panelu se nachází veškeré věci, které implementujeme do obvodů. Jak je vidět na obrázku v panelu je otevřená záložka "Variables" do které se nám uložily ty bloky, které jsme vytvořily a následně pojmenovali v předešlých krocích. Tempo_pro X1 reader a další.

Shrnutí kroku 3: Vytvořit inicializaci do které nadefinujeme rychlosti jednotlivých kroků. Následně propojíme mezi sebou. Blok Rychlost se propojí s konstantou "100" a Tempa se propojí s konstantou "1000". Poté pokaždé, když někde vložíme blok rychlost automaticky bude mít hodnotu rychlosti "100".

KROK 4

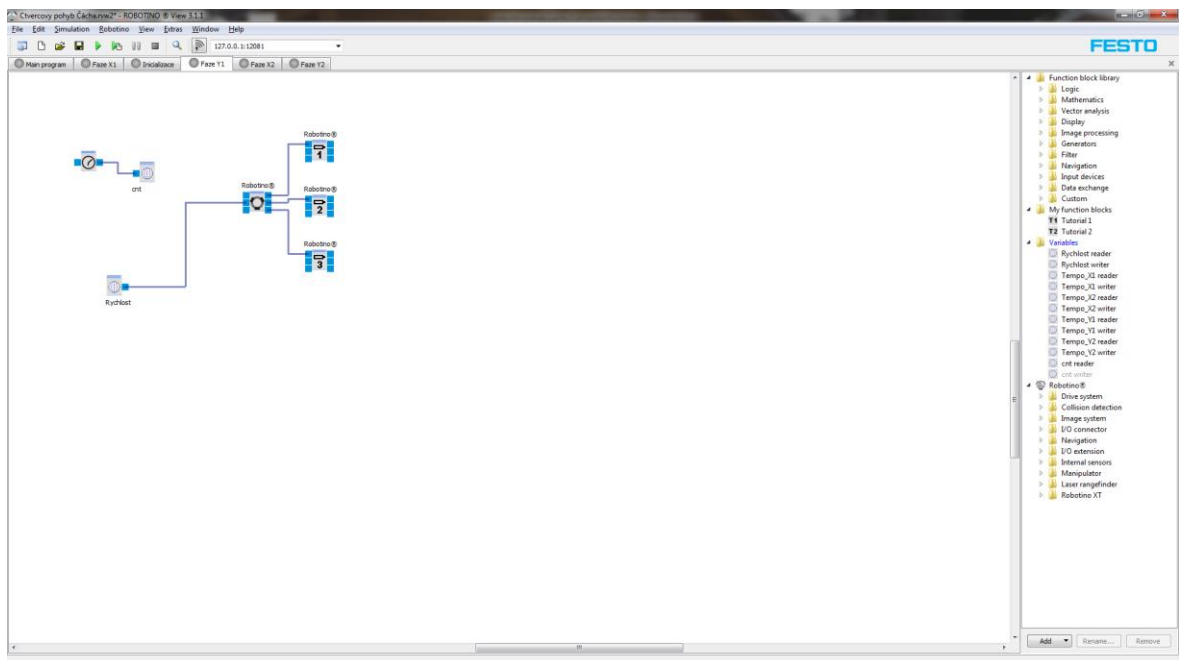
V tomto kroku se zaměříme už na programování jednotlivých podprogramů (X1, Y1 apod.) Jako první si otevřeme na notifikační liště záložku X1 do níž umístíme z pomocného panelu nástrojů na pravé straně motory a to všechny 3. Motory se nachází v záložce "Drive System". Potom přetahneme ze stejné složky ještě řídicí ovládání motorů "OMNIDRIVE". Následně motory propojíme s Omnidrivem tím způsobem, že každý motor zapojíme do Omnidrivu z prvního portu. Každý motor má tři porty. Využijeme ten první, ten slouží pro pohon. Potom co budeme mít toto úspěšně propojené přetáhneme opět z pomocného panelu ze složky "Generator" tkz. "Timer" , který nám zajistí plynulý přechod mezi navazujícími kroky. To ovšem není všechno. Ze složky "Variables" , který se nachází na pomocném panelu napravo, tkz. "Count reader" , který propojíme s "Timerem" a spolu nám budou zajišťovat plynulé přechody na další kroky.



Shrnutí kroku 4: Složit podprogram X1, což znamená pospojovat ovládání s motory a na libovolné místo přetáhnout "Timer" a "Count" , které nám zajistí plynulý přechod na další kroky. (X , Y)

KROK 5

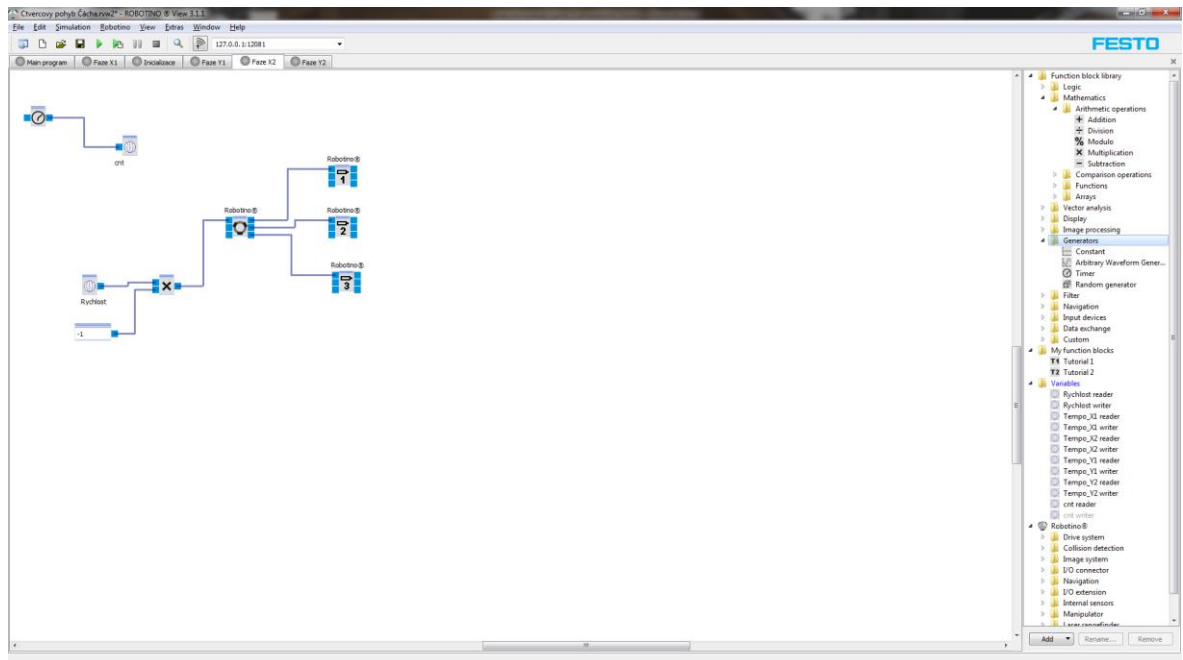
Tímto krokem naprogramujeme podprogram Y1 , kde budeme postupovat úplně stejně jako v předchozím kroku , kde jsme programovali podprogram X1. Jediným rozdílem bude to, že rychlost (konstanta vytvořena v kroku 3) nebude napojena na první vstupní port do Omnidrivu (ovládací segment motorů) , ale do vstupního portu 2, protože port 2 nám zajišťuje rychlost ve směru Y zatímco port 1 ve směru X na souřadnicové ose. Tudiž nám robot pojede nahoru.



Shrnutí kroku 5: V tomto kroku jsme provedli to samé zapojení jako v kroku 4 s tím rozdílem , že konstantu rychlosti zapojíme do portu 2 místo 1 (vstup)

Krok 6

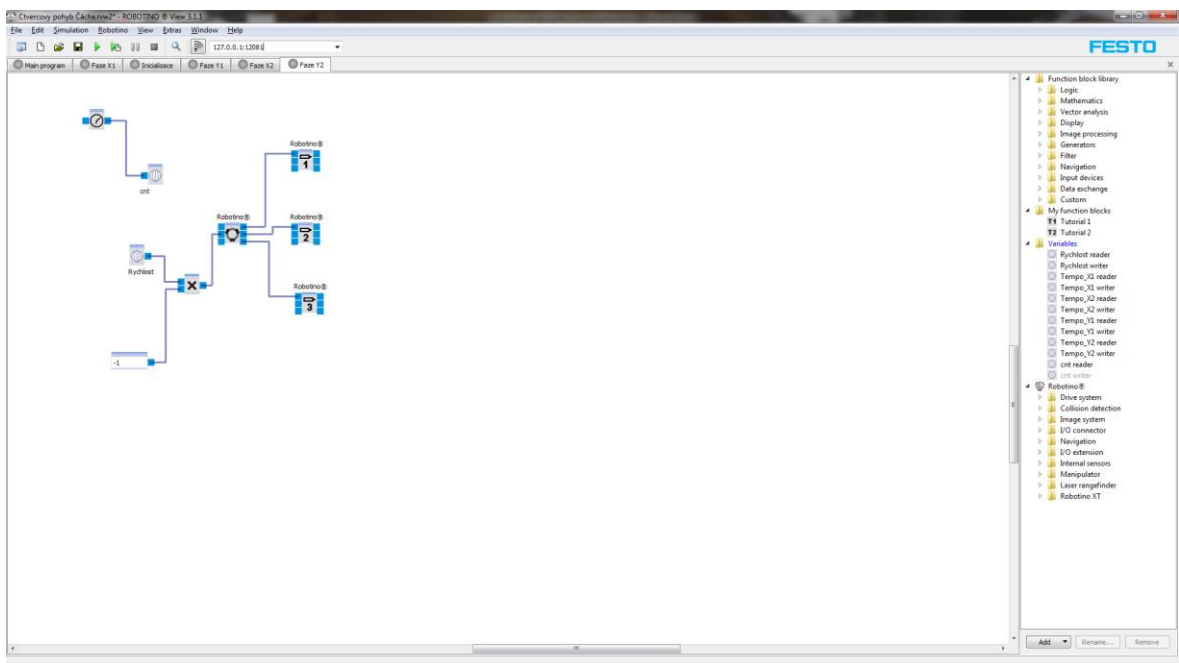
Tento Krok nám bude realizovat podprogram X2 , kde opět vše uděláme tak, jako v předchozích krocích. Opět vložíme motory, které propojíme s Omnidrivem (opět vstup PORT1). Dále vložíme znovu "Timer" spolu s "Count" , které propojíme. Rozdíl nastává až nyní, protože se pohybuje v ose X , ale v pohybu doleva tudíž do záporných hodnot ,a proto musíme použít přídavný matematický úkon tkz. "Multiplication" neboli násobení. Tento blok najdeme opět v pomocném panelu ve složce "Mathematics" a dále v podsložce Arithmetic Operations. Přetáhneme na plochu a pak musíme vložit konstantu Rychlosti (definována Krok 3) tu propojíme s "Multiplication".Následně vložíme ještě jednu konstantu, kterou nalezneme ve složce "Generators".Opět přetáhneme a do políčka napíšeme -1 a spojíme s "Multiplication" (port 2 vstup).



Shrnutí Kroku 6: Opět sestavit podprogram stejně jako v předeslých případech s tím rozdílem , že je potřeba vložit pomocný aritmetický prvek, který nám zajistí převedení rychlosti do mínusu a tím i správnost pohybu doleva po směru osy X.

Krok 7

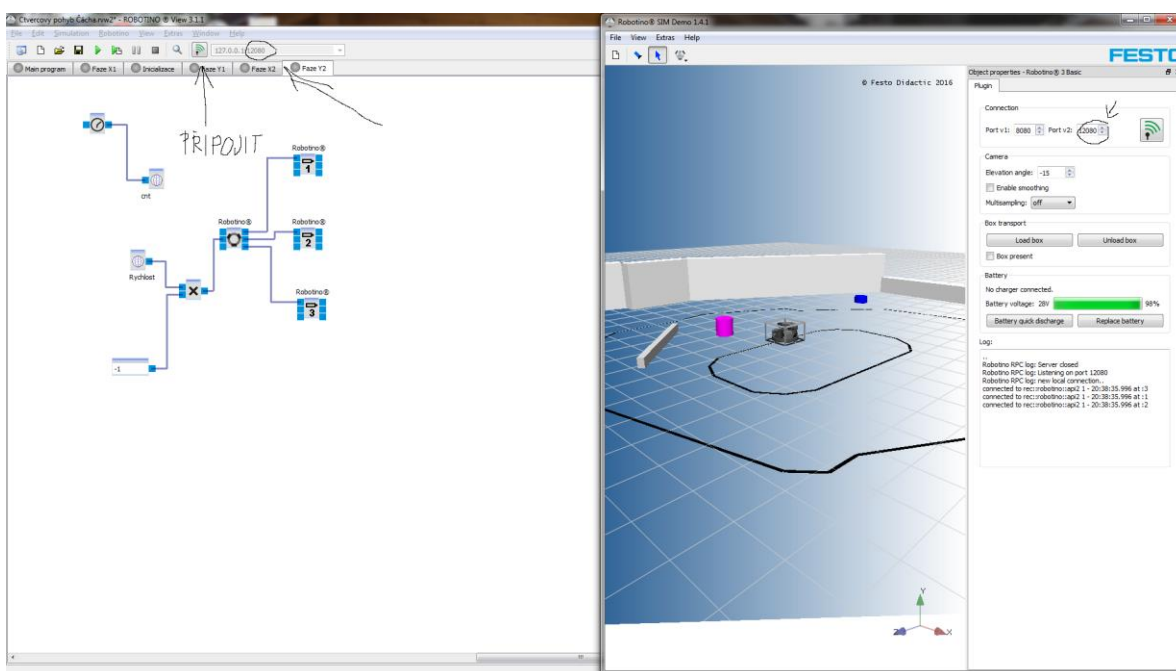
V tomto kroku budeme programovat podprogram Y2 , který vypadá na první pohled stejně jako u předchozího kroku s tím rozdílem, že napojení té aritmetické části (vysvětlena v kroku 6) se nepřipojuje na port 1 vstupu nýbrž na port 2 vstupu , protože opět programuje pohyb na ose Y a ta má definován port 2.



Shrnutí Kroku 7: Naprogramujeme podprogram Y2 s tím rozdílem, že opět musíme dát pozor na to , že toto je osa Y, tudíž napojit Aritmetiku do portu 2 vstupu Omnidrivu.

Krok 8

Dostáváme se k finálnímu kroku našeho vytvořeného programu a to k simulaci, která nám ukáže zda jsme byli úspěšní a úloha funguje podle představ. Začneme tím, že si otevřeme Robotino® SIM DEMO, které nám slouží jako prostředí pro simulaci. Jakmile otevřeme program uvíme prostranství s překážkami a robota. Pokud chceme převést program do simulace musíme první kliknout na modrou šipku vlevo nahoře, tím se dostaneme do výběru. Potom co zmáčkneme šipku, klikneme na robota, který se nám virtuálně jakoby zavře do "klece" tím ho označíme. Objeví se vpravo hodnoty v pomocném panelu. V tomto panelu nahoře se nachází odstavec, který se nazývá "Connection" ten nás zajímá nejvíc, protože tam bude tkz. Port V2 , který nám říká port pro Robota. Tento port přepíšeme do IP adresy robota. Viz obrázek. Pokud vše souhlasí bude ikonka připojení svítit zeleně. Pak už jen stačí zapnout v Robotino®View 3 Simulaci a na Robotino® SIM DEMO uvidíme naši úlohu



Shrnutí Kroku 8: Finální propojení úlohy se simulačním programem viz. foto.

4. Závěr

Tato práce je zaměřena na ovládání a pohyb Robotina® a je rozdělena na dvě části.

První část je teoretická a je zaměřena na rozbor dané tematiky více do detailů s názorným popisem částí a principu funkce Robotina® jakožto celku.

Ve druhé části je zpracována praktická část a názorná úloha na ovládání Robotina®. Jelikož jsem se dříve s programováním robotů nikdy nesetkal a toto byla moje první větší zkušenost, musím ocenit přínos této učební pomůcky, která je dobře využitelná při školské praxi, ale i jakožto nástroj pro ukázky a programování jednotlivých segmentu z kterých je složena. Protože jsou segmenty z kterých je složena na vysoké úrovni a jsou průmyslově aplikovány do takovéhoho modulárního celku vidím širokou perspektivu v používání těchto výukových pomůcek jakožto přípravu a seznámení s průmyslovou automatizací a programováním.

5. Seznam použité literatury a zdrojů informací

Zdroje informací

[1.] <https://cs.wikipedia.org/wiki/Festo>

[2.] <https://en.wikipedia.org/wiki/Robotino>

[3.] <http://www.festo-didactic.com/int-en/learning-systems/education-and-research-robots-robotino/robotino-for-research-and-education-premium-edition-and-basic-edition.htm>

[4.] konzultace s vedoucím práce.

Literatura

Jelikož je toto téma vcelku nové v dnešní době není jednoduché získat tištěné materiály v českém jazyce. Proto pro tuto práci bylo využito výhradně Internetových Stránek, které se zabývají danou tematikou.

6. Seznam použitého software

- Robotino® View 3
- Robotino® SIM DEMO
- Microsoft Word 2007
- Malování-v počítači

7. Seznam příloh

Veškeré přílohy jsou přiloženy na příslušném CD, které se nachází na zadní straně této práce. CD obsahuje: Příslušné programy, Vypracovanou úlohu, Maturitní práci.

8. Licenční ujednání

Ve smyslu zákona č. 121/2000 Sb., O právu autorském, o právech souvisejících s právem autorským, ve znění pozdějších předpisů (dále jen autorský zákon) jsou práva k maturitním nebo ročníkovým pracím následující:

Zadavatel má výhradní práva k využití práce, a to včetně komerčních účelů.

Autor práce bez svolení zadavatele nesmí využít práci ke komerčním účelům.

Škola má právo využít práci k nekomerčním a výukovým účelům i bez svolení zadavatele a autora práce.

