



Pedagogická  
fakulta  
Faculty  
of Education

Jihočeská univerzita  
v Českých Budějovicích  
University of South Bohemia  
in České Budějovice

Jihočeská univerzita v Českých Budějovicích  
Pedagogická fakulta  
Katedra informatiky

Bakalářská práce

# Použití paralelního robota CoreXY pro výuku v zájmových kroužcích

*Parallel robot CoreXY in free time activities*

Autor: Patrik Janda  
Vedoucí práce: Ing. Michal Šerý, PhD.

České Budějovice 2019

JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH  
Fakulta pedagogická  
Akademický rok: 2016/2017

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Patrik JANDA**  
Osobní číslo: **P15663**  
Studijní program: **B7507 Specializace v pedagogice**  
Studijní obor: **Informační technologie a e-learning**  
Název tématu: **Použití paralelního robota CoreXY pro výuku v zájmových kroužcích**  
Zadávající katedra: **Katedra informatiky**

### Z á s a d y p r o v y p r a c o v á n í :

1. Úvod
2. Základní pojmy, uvedení do problematiky robotů
3. Konstrukce CoreXY (mechanika a elektronika), sestavení a oživení CoreXY
4. Použití Arduina a Raspberry Pi pro řízení
5. Vypracování modelových úloh
  - konstrukce: plán robota (1)
  - elektronika: zapojení motorů, čidla (2)
  - počítače HW: komunikace mezi Arduinem a RPi, s PC (3)
  - počítače SW: programování Arduina, RPi (2)
6. Vypracování ukázkových řešení jednotlivých úloh
7. Zhodnocení a závěr

Rozsah grafických prací: **CD ROM**  
Rozsah pracovní zprávy: **40**  
Forma zpracování bakalářské práce: **tištěná**  
Seznam odborné literatury: **viz příloha**

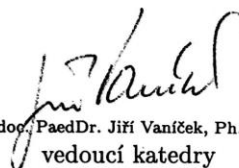
Vedoucí bakalářské práce: **Ing. Michal Šerý, Ph.D.**  
Katedra aplikované fyziky a techniky

Datum zadání bakalářské práce: **24. dubna 2017**

Termín odevzdání bakalářské práce: **30. dubna 2018**



Mgr. Michal Vančura, Ph.D.  
děkan



doc. PaedDr. Jiří Vaníček, Ph.D.  
vedoucí katedry

V Českých Budějovicích dne 24. dubna 2017

## **Prohlášení**

Prohlašuji, že svoji bakalářskou práci na téma Použití paralelního robota CoreXY pro výuku v zájmových kroužcích jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě, elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Českých Budějovicích 12. 07. 2019

.....

**Patrik Janda**

## **Abstrakt**

Bakalářské práce je zaměřena na použití paralelního robota CoreXY pro výuku v zájmových kroužcích, poté se bude zabývat návrhem vhodných modelových úloh pro robota CoreXY. Posledním bodem bude otestování vylepšené konstrukce robota CoreXY a modelových úloh. V rámci práce bude vypracován konstrukční návod na vylepšení robota CoreXY a použitelné modelové úlohy v zájmových kroužcích.

## **Klíčová slova**

CoreXY, 3D tiskárna, Arduino, Raspberry Pi, zájmový kroužek, modelové úlohy

## **Abstract**

The bachelor's thesis is focused on the use of CoreXY Parallel Robot for the teaching of interest circles, then it will deal with the design of suitable model tasks for the CoreXY robot. The last point will be testing the improved CoreXY robot design and model tasks. In the course of the thesis, the design instructions for the CoreXY robot will be developed and usable model tasks in the interest rings.

## **Keywords**

CoreXY, 3D printer, Arduino, Raspberry pi, free time activities, Model examples

## **Poděkování**

Děkuji vedoucímu mé bakalářské práce Ing. Michalu Šerému, Ph.D. za odborné vedení, udílení cenných inspirativních rad, trpělivost a veškerý čas, který mi věnoval. Poděkování patří také všem mým blízkým, kteří mi byli po celou dobu studia velkou oporou.

# Obsah

<b>1</b>	<b>Úvod.....</b>	<b>11</b>
1.1	Cíle práce .....	11
1.2	Metody práce .....	12
1.3	Východiska práce .....	12
<b>2</b>	<b>Základní pojmy, uvedení do problematiky robotů .....</b>	<b>13</b>
2.1	G-code.....	13
2.1.1	Rozbor základních kódů .....	14
2.2	OctoPrint.....	14
2.3	Raspberry Pi.....	15
2.4	Arduino .....	16
2.5	CoreXY.....	17
<b>3</b>	<b>Konstrukce CoreXY.....</b>	<b>19</b>
3.1	Mechanika.....	20
3.2	Elektronika .....	21
3.3	Sestavení a oživení CoreXY.....	21
<b>4</b>	<b>Použití Arduina a Raspberry Pi pro řízení.....</b>	<b>25</b>
<b>5</b>	<b>Vypracování modelových úloh.....</b>	<b>26</b>
5.1	Konstrukce: plán robota.....	26
5.1.1	Model CoreXY z kartonu .....	26
5.2	Elektronika: zapojení motorů, čidla.....	28
5.2.1	Zapojení motorů .....	28
5.2.2	Zapojení čidel do Raspberry Pi .....	28
5.3	Počítače HW: komunikace mezi Arduinem a RPi, s PC.....	29
5.3.1	Připojení Arduina .....	29
5.3.2	Nahrání GRBL.....	29
5.3.3	Kalibrace CoreXY .....	31
5.3.4	Instalace OctoPrint do Raspberry Pi.....	34
5.4	počítače SW: programování Arduina, RPi .....	36
5.4.1	Úloha Start.....	36
5.4.2	Úloha Úsečka.....	37
5.4.3	Úloha Čtverec.....	39
5.4.4	Úloha Obdélník .....	41



5.4.5	Úloha Trojúhelník 1 .....	44
5.4.6	Úloha Trojúhelník 2 .....	46
5.4.7	Úloha Čtverec ve čtverci .....	48
5.4.8	Úloha kosočtverec .....	51
<b>5.5</b>	<b>Práce se SW a podpůrnými prostředky .....</b>	<b>54</b>
5.5.1	Práce s OctoPrint .....	54
5.5.2	Odeslání souboru G-code přes GrblController .....	54
5.5.3	Přizpůsobení Inkscape pro CoreXY .....	55
5.5.4	Kreslení v Inkscape .....	55
5.5.5	Převod obrázku pomocí Inkscape .....	56
5.5.6	Práce s pomocnou šablonou pro určení souřadnic X a Y .....	56
<b>6</b>	<b>Vypracování ukázkových řešení jednotlivých úloh .....</b>	<b>57</b>
<b>6.1</b>	<b>Řešení úlohy konstrukce .....</b>	<b>57</b>
6.1.1	Model CoreXY z kartonu .....	57
<b>6.2</b>	<b>Řešení úloh elektronika .....</b>	<b>57</b>
6.2.1	Zapojení motorů .....	57
6.2.2	Zapojení čidel .....	58
<b>6.3</b>	<b>Řešení úloh HW .....</b>	<b>58</b>
6.3.1	Připojení Arduina .....	58
6.3.2	Nahrání GRBL .....	59
6.3.3	Kalibrace CoreXY .....	59
6.3.4	Instalace OctoPrint do Raspberry Pi .....	60
<b>6.4</b>	<b>Řešení úloh SW .....</b>	<b>61</b>
6.4.1	Úloha Start .....	61
6.4.2	Úloha Úsečka .....	62
6.4.3	Úloha Čtverec .....	63
6.4.4	Úloha Obdélník .....	64
6.4.5	Úloha Trojúhelník 1 .....	65
6.4.6	Úloha Trojúhelník 2 .....	66
6.4.7	Úloha Čtverec ve čtverci .....	67
6.4.8	Úloha Kosočtverec .....	68
<b>7</b>	<b>Zhodnocení a závěr .....</b>	<b>69</b>
	<b>Seznam použité literatury a zdrojů .....</b>	<b>70</b>
	<b>Seznam obrázků .....</b>	<b>72</b>

<b>Seznam tabulek.....</b>	<b>74</b>
<b>Seznam příkladů.....</b>	<b>75</b>
<b>Přílohy .....</b>	<b>76</b>
<b>Příloha A (DVD) .....</b>	<b>77</b>
<b>Příloha B (tabulka).....</b>	<b>78</b>
<b>Příloha C (tabulka).....</b>	<b>80</b>
<b>Příloha D (šablona).....</b>	<b>81</b>
<b>Příloha E (šablona).....</b>	<b>83</b>
<b>Příloha F (šablona) .....</b>	<b>85</b>

# 1 Úvod

S používáním robotů v praxi se setkáváme stále častěji. Myslím, že robotizace v budoucnu však ovlivní i veškerou lidskou činnost. Většinou lidem se líbí, když se dívají na roboty vykonávající těžkou, nebo náročnou činnost v extrémních podmínkách, např. svařovny, nebo lakovny automobilek. Robot je vlastně stroj, vhodný pro většinu obslužných činností s velkou opakovatelností úkonů, náročností na přesnost i fyzickou námahu. Nejen, že roboti značně ulehčují lidem práci, ale také jim na druhé straně práci ubírají. Tak jako všechny vynálezy, lze však i roboty zneužít, např. vyroběním robotů, kteří jsou schopni zabít a ničit lidské životy.

Jen málokdo si však dovede představit, co úsilí musí být vynaloženo k hladkému a přesnému chodu robotů.

Samostatnou kapitolou je programování robotů, neboť každý robot pracuje dle vlastního programu, který mu sestavil programátor. Budoucnost však patří robotům, kteří se sami budou učit a optimalizovat. Programování je speciální činnost, které se věnuje jen málokdo, vzhledem k její poměrně obtížnosti. S ochotou a trpělivostí se lze ale naučit vše, a to i programování a používání robotů. K přípravě a porozumění programování je však nutné mít alespoň základní znalosti elektrotechniky a několika příbuzných oborů. [1]

Mým úkolem není rozvíjet znalosti elektrotechniky, nýbrž poskytnout praktický návod k sestavení a použití paralelního robota CoreXY pro výuku především v zájmových kroužcích. Vzhledem k tomu, že v současné době je omezený příliv investic ve školách pro nákup vhodných výukových pomůcek, mohl bych po sestavení a odzkoušení paralelního robota CoreXY věnovat tohoto robota některému z kroužků zabývajícím se výukou a programováním robotů, aby si řada studentů mohla osvojit tuto problematiku.

## 1.1 Cíle práce

Cílem mé bakalářské práce je navrhnutí a vytvoření několika modelových úloh pro užití ve výuce v zájmových kroužcích na robotovi CoreXY pro studenty se základy programování a elektroniky. Návrh vhodných modelových úloh bude realizován s využitím Arduino a Raspberry Pi. Účastníci zájmových kroužků by si měli pomoci

sbírky vhodných modelových úloh osvojit stavbu konstrukce, zapojení čidel a motorů. Na vytvořeném modelu konstrukce by se mohli seznámit, jak probíhá pohyb v osách X a Y. Dále by se měli naučit zvládat komunikaci mezi Arduinem, Raspberry Pi a PC. Také si osvojí programování a práci s Raspberry Pi a Arduinem.

## **1.2 Metody práce**

Nejdříve prostuduji různé modely CoreXY, zvolím nejvhodnější sestavu a následně se pokusím ji přizpůsobit pro volnočasový kroužek. Vybranou konstrukci CoreXY sestavím. Po sestavení a zprovoznění CoreXY, Arduina a Raspberry Pi navrhnu několik modelových úloh a následně otestuji, zda lze úlohy realizovat na vybrané konstrukci.

## **1.3 Východiska práce**

Rozvoj 3D tisku přinesl mnoho skvělých myšlenek do paralelních robotů. V současné době se používá mnoho různých konstrukcí a technik, které jsou pro finální kvalitu velmi důležité. Výhody použití konstrukce CoreXY jsou: přesnost, rychlost, jednoduchost a flexibilita. Podrobnější popis a rozepsané výhody CoreXY naleznete v kapitole CoreXY.

Počáteční nedostatky byly odstraněny a v současnosti se již přešlo k lehčím a zejména levnějším konstrukcím, které umožní vyšší využití v 3D tisku. Tyto konstrukce jsou již vhodné i pro výuku v zájmových kroužcích. Základní problém doposud ale shledávám v nedostačujících modelových úlohách pro využití CoreXY robota v zájmových kroužcích.

## **2 Základní pojmy, uvedení do problematiky robotů**

Abyste snáze porozuměli problematice robotů, seznámím Vás s některými základními pojmy, které úzce souvisí s problematikou robotů. Jsou to například: G-code, Arduino, Raspberry PI, CoreXY, OctoPrint. Tyto pojmy naleznete níže v podkapitolách G-code, Arduino, Raspberry Pi, CoreXY a OctoPrint.

V dnešní době je problematika robotů stále aktuálnější téma. Roboti nás obklopují v našem životě dennodenně, aniž bychom jim věnovali nějakou zvláštní pozornost. Většina lidí si vůbec neuvědomuje, jak nebezpečná může být manipulace s roboty, ať už se jedná o ty běžné, které využívá každý v domácnosti, nebo roboty, kteří jsou určeny přímo k výrobě ve velkých podnicích. Je zásadní dodržování bezpečnostních pokynů při manipulaci s roboty a jejich údržbě. Ve výrobních linkách mají roboti svá bezpečnostní opatření, která je nutná aplikovat vždy s harmonizovanou normou ČSN EN ISO 10218-2:2011. Jde především o to, že v přítomnosti člověka dojde ke snížení výkonu a síly robota, nebo dokonce i k vypnutí robota. Při manipulaci s roboty existuje speciální nouzové tlačítko pro okamžité přerušení vykonávané práce robotem nazývané „Tlačítko mrtvého muže“. K dalším bezpečnostním opatřením patří například masivní ochranná klec, ve které roboti vykonávají práci, nebo použití optoelektronických prvků, které hlídají nebezpečný prostor robota. [2]

V praktické části je popsána konstrukce a mimo jiné i problémy při jejím sestavování a testování.

### **2.1 G-code**

Na konci 50. let vznikla v laboratoři MIT Servomechanism Laboratory první myšlenka programovacího jazyka pro numerické řízení, kde se používá G-code. Hlavní standardizovaná verze byla vytvořena ve Spojených státech v průběhu 60. let. Poslední úpravy byly schváleny v únoru 1980 jako RS-274-D. Evropské země používají jiné normy. Výrobci řídicích systémů a výrobci obráběcích strojů přidali svoje rozšířené varianty a jsou mezi nimi různé rozdíly. Během 70. až 90. let se na CNC strojích pokoušela standardizovat řídicí nástroje firma Fanuc. Dalším subjektem v Evropě ovlivňující řízení CNC byla firma Siemens. Po roce 2010 již nejsou rozdíly a nekompatibilita takovým problémem, protože jsou obráběcí operace vyvíjeny pomocí aplikace CAD / CAM, která umožňuje použít G-code.

G-code je využíván především u CNC obráběcích strojů. G-code slouží jako příkaz znamenající nějakou předdefinovanou instrukci v CNC stroji, který začíná písmenem G. Jde obecně o kód, který stroji určí, jaká operace a jak má být provedena. Kromě CNC je velmi vhodné použít G-code i u 3D tiskáren. [3]

### 2.1.1 Rozbor základních kódů

Základem jsou G a M kódy. G-code ovládají pohyby, které by měly být u všech strojů stejné, zatímco M kódy pracují se součástmi tiskárny (např. chlazení, motory atd.). M-code mohou být na každém stroji jinak nadefinované, a proto je vhodné zjistit, co na zvoleném stroji dělají. V Příloze B a C naleznete podporované příkazy pro systém GRBL, ve formě tabulky.

## 2.2 OctoPrint

Při tisknutí na 3D tiskárně a práci na CNC strojích mohou vznikat různé problémy. Když si jich včas nevšimneme, nebo je objevíme pozdě, může dojít k poměrně velkým ztrátám a dalším nákladům.

K zabránění vzniku nekvalitního výsledku, ztrátám a dalším nákladům, je zapotřebí mít neustálý dohled nad 3D tiskárnou a CNC strojem, a proto je ideální použít OctoPrint.

OctoPrint je bezplatný open-source software vytvořený a spravovaný společností Gina Haeusse, pomocí něhož provádíme řadu užitečných úloh, které souvisí s 3D tiskem a prací s CNC stroji, včetně dálkového ovládání stroje, živého videa a sběru dat.

Open-source software OctoPrint může pomocí připojené webové kamery přenášet video z chodu robota, a to nám umožňuje vidět právě probíhající činnost robota nebo chyby, které vyžadují přerušení nebo restart. Na časosběrné video můžeme i nahrát celý průběh práce.

OctoPrint zaznamenává data v reálném čase jako je např. teplota, díky čemuž lze sledovat a zajišťovat hladký a nepřerušovaný proces. Záznam teplotních dat navíc může pomoci i při řešení problémů.

Díky open-source software OctoPrint můžeme robota ovládat vzdáleně za pomoci internetu, i když se nenacházíme ve stejné místnosti s robotem, a nahrávat na něj G-code soubory, nebo jen pohybovat jeho rameny. [4]

### 2.3 Raspberry Pi

Raspberry Pi je malý jednodeskový počítač přibližně o velikosti platební karty. Byl vyvinut v roce 2012 britskou nadací Raspberry Pi Foundation. Cílem nadace byla podpora výuky informatiky na školách a seznámení studentů s řízením domácích spotřebičů počítači. Jako primární operační systém byl použit Raspbian. Cenové rozpětí bylo v roce 2019 za různé modely od 200 Kč do 1 850 Kč.

Raspberry Pi je malý jednočipový počítač (mikrokontroler), který můžeme srovnat se slabším stolním počítačem. Jeho součástí je vývod pro monitor podporující rozhraní HDMI a je možné také připojit přes USB klávesnici a myš. Raspberry Pi dále obsahuje i několik druhů sběrnic pro čidla a různá rozšíření. Bylo již vyvinuto několik generací Raspberry Pi, které se liší použitím a výkonem. Vzhledem k použitému mikroprocesoru ARM je skoro srovnatelný s běžným chytrým telefonem. Raspberry Pi funguje na různé distribuce Linuxu, dokonce i na speciální verzi Microsoft Windows 10 IoT Core.

Raspberry Pi lze, na rozdíl od počítače a Arduina, použít nejen k ovládní různých zařízení, ale i přímo k vývoji aplikací. Může se použít i jako multimediální přehrávač videa nebo hudby, nebo jen pro přístup k internetu. [5] [6]

Model	Datum uvedení
Původní Raspberry Pi	2012
Model „A“	2/2013
Model „A+“	11/2014
Model „B“	04/2012
Model „B+“	07/2014
Raspberry Pi 2	02/2015
Raspberry Pi 3	02/2016
Raspberry Pi 4	06/2019

Tabulka 1: Historie modelů Raspberry Pi [6]

## 2.4 Arduino

Arduino je open source platforma založená na uživatelsky jednoduchém hardware a software. Je vhodné pro výuku programování, nebo i jen tak pro pochopení funkcí moderních technologií. Je určeno především pro tvorbu rychlých, jednoduchých, interaktivních a zábavných projektů.

Arduino je vlastně malý počítač, který dokáže vnímat vnější svět pomocí různých senzorů, na které pak reaguje a uvede v činnost různá zařízení, například rozsvěcuje LED diody nebo se ozve zvukový signál či se spustí motory atd.

Základní Arduino deska je velmi jednoduchá a můžeme si ji sami vyrobit, nebo můžeme součástky pospojovat na desce plošných spojů. Arduino tvoří v podstatě mikrokontroler, krystal, napájecí 5 V zdroj a převodník pro komunikaci s počítačem.

Podle požadované funkce (např. ovládání motorů, displeje, nebo bezdrátového modulu) můžeme využít celou škálu rozšiřujících desek pro Arduino. Tyto desky označujeme jako Arduino Shildy. Dnes jich již existuje velké množství a ke každému bývá volně k dispozici knihovna pro okamžité použití.

Pro práci na projektech a komunikaci s Arduinem můžeme použít software Arduino IDE, který je možný stáhnout zdarma z webových stránek výrobce <https://arduino.cc>. Programovací jazyk, který lze použít v rozhraní softwaru Arduino IDE se nazývá Wiring.

Velkou výhodou je, že Arduino desky a příslušenství si můžeme také zakoupit. Jsou k dispozici jak originální Arduino desky, tak i jejich klony od dalších výrobců. V každém případě je spolehlivější koupit originální desky Arduino, které budou vždy fungovat. Mají i potřebné certifikáty (CE, FCC, RoHS) pro zabudování do profesionálních produktů. Jsou však pochopitelně dražší. Klony Arduino desek vykazují někdy problémy, jako je nekompatibilita hardware a ovladačů, a jsou také často i nespolehlivé. To platí hlavně u Arduino klonů od bezejmenných výrobců z Číny. Jejich nespornou výhodou je však cena a jsou vhodnější pro zkušenější uživatele, kteří si dovedou s případnými problémy poradit. [7]



Výhody	Popis
Nízkonákladovost	Proti ostatním platformám jsou Arduino desky podstatně levnější. Osazené desky Arduino seženeme již za 50 USD. Nejlevnější jsou však při ručním sestavování, když si je sami vyrobíme.
Multiplatformnost	Software Arduino je možno využít na řadě operačních systémů jako Windows, Macintosh OSX a Linux. U většiny ostatních lze použít pouze Windows.
Jednoduché a čisté programovací prostředí	Programovací prostředí Arduino je jednoduché a vhodné jak pro začátečníky, tak i pro zkušené uživatele vzhledem k svojí flexibilitě.
Open source a rozšiřitelný software	Software Arduino je otevřený systém umožňující zkušenějším uživatelům další rozšíření. Rozšíření jazyka může být provedeno přes knihovny C++ a pokročilejší mohou přejít z Arduina na AVR C.
Open source a rozšiřitelný hardware	Základem Arduina jsou mikrokontroléry firmy Atmel. Desky návrhů Arduina jsou vydávány pod licencí Creative Commons. Začínající uživatelé si mohou sami vytvořit Arduino s použitím desky plošných spojů. Úplnou vlastní verzi desky si však mohou vytvořit jen zkušení vývojáři.

*Tabulka 2: Výhody Arduina [7]*

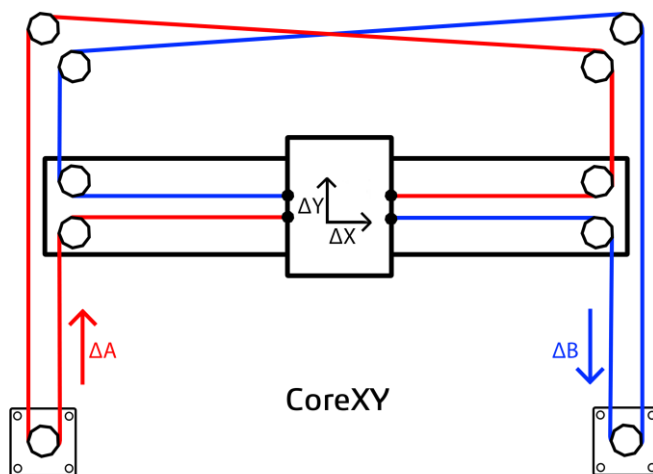
### 2.5 CoreXY

CoreXY je často využívaný způsob techniky pohybu v tiskárnách CORE 3D. Proti běžným metodám pohybu (kartézský a delta) má svá specifika.

Výhody	Popis
Přesnost	Tím, že jsou oba krokové motory stacionární, nevzniká nepřesnost vlivem setrvačných sil.
Rychlost	Paralelní kinematika u CoreXY se stacionárními motory umožňuje dosahovat vysokého zrychlení.
Jednoduchost	Realizace CoreXY je poměrně jednoduchá, stačí pouze 3 konstrukční desky.
Flexibilita	Na konstrukci CoreXY lze uchytit různá příslušenství (laser, fréza, pero, hlava pro 3D tisk).

*Tabulka 3: Výhody CoreXY [8]*

Princip CoreXY je zjednodušeně uveden na obrázku „Princip CoreXY“.



Obrázek 1: Princip CoreXY

Pohyby X a Y jsou řízeny dvěma stacionárními krokovými motory. Ani jeden motor není určen pro jednu osu, ale firmware použije motory v tandemu k dosažení různých souřadnic X a Y. Starost s výpočty odpadá kvůli firmwaru GRBL, který se o ně postará. [8]

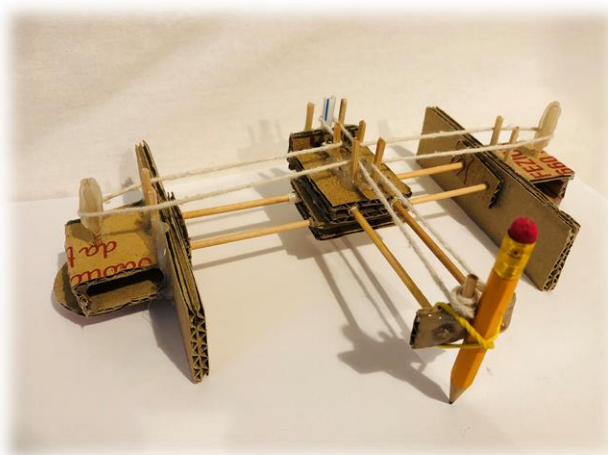
$$\Delta X = 0,5(\Delta A + \Delta B), \Delta Y = 0,5(\Delta A - \Delta B), \Delta A = \Delta X + \Delta Y, \Delta B = \Delta X - \Delta Y$$

Příklad 1: Vzorec výpočtu souřadnic CoreXY [8]

### 3 Konstrukce CoreXY

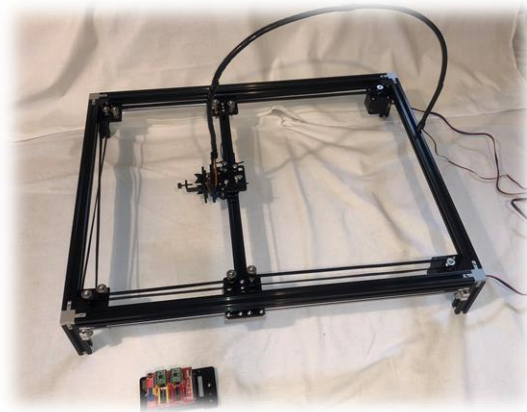
Funkci a vlastnosti CoreXY jsem již popsal v předchozí kapitole CoreXY. Nyní se podíváme na potřebné součástky a řídicí elektroniku. Dále bude podrobněji popsán postup sestavení robota CoreXY a jeho následné oživení.

Rozhodoval jsem se mezi dvěma možnostmi konstrukce pro CoreXY. Konstrukce DrawBot posouvá celé rameno Y po ose X. Na konci ramena Y může být uchycen laser, frézovací či řezací nástroj nebo pero. Konstrukce je poháněna dvěma krokovými motory. Pro ovládání pera, řezacího či frézovacího nástroje je zapotřebí servomotor. Na konstrukci může nastat problém se stabilitou v ose Y. Při využití v zájmovém kroužku by mohlo také dojít lehce k narušení pracovní plochy, neboť na této konstrukci není zcela viditelně vymezen pracovní prostor, proto může být konstrukce nevhodně umístěna, nebo bude do pracovního prostoru umístěna překážka (penál, mobilní telefon a jiné předměty).



Obrázek 2: Model konstrukce DrawBot

Konstrukce CoreXY má obdélníkový tvar. Konstrukce se skládá ze čtyř hliníkových profilů na nožičkách s kříženým ozubeným řemenem, který je natažený kolem tří stěn robota. Uvnitř robota se pohybuje platforma po osách X a Y. U robota je na první pohled vidět, že se jeho pracovní plocha nachází uvnitř. Pracovní plocha je chráněna ze všech stran. Pro toto viditelné ohraničení pracovní plochy a možnou větší stabilitu shledávám tuto variantu jako nejvhodnější pro použití v zájmovém kroužku.



Obrázek 3: Konstrukce CoreXY

Dále jsem pro robota CoreXY zajistil profesionální díly v prodejní hodnotě přibližně 50 000 Kč. Tyto díly by bylo nutné ještě pro tento projekt přizpůsobit, ale bylo by to časově a finančně náročnější, a proto jsem se rozhodl najít levnější a časově méně náročnější variantu. Rozhodl jsem se pro variantu částečně předpřipravené stavebnice. Stavebnice cenově vyšla přibližně na 6 000 Kč. Sestavení a úprava nepasujících dílů stavebnice zabere přibližně 5 hodin. Tato varianta je, dle mého názoru, nejideálnější pro užití v zájmovém kroužku také vzhledem k její poměrně malé finanční a časové náročnosti.

### 3.1 Mechanika

Mechanickou část robota CoreXY tvoří osa X a Y skládající se převážně z hliníkového profilu 20 x 20 mm, polykarbonátové desky, M2 šroubů, matek, lineárních a kuličkových ložisek, vodicích tyčí.



Obrázek 4: Součástky robota CoreXY

## 3.2 Elektronika

Jako řídicí elektronickou část robota jsem se rozhodl použít Arduino Uno s rozšiřitelnou deskou s názvem CNC Shield V3 a dvěma drivery A4988. Komunikaci s Arduinem Uno může zajišťovat Raspberry Pi 3B+.

Vyzkoušel jsem levnější variantu Arduina Una vyrobenou v Číně s obvodem CH340G. Tato varianta mi nevyhovovala ve složitosti zprovoznění komunikace mezi PC a Arduinem Unem a stabilitě komunikace. Tuto variantu Arduina Una s obvodem CH340G shledávám jako nevhodnou pro užití v zájmovém kroužku. Jako další jsem objednal klon Arduina Una, který je od originálu k nerozeznání. Tento klon je cenově levnější o 500 Kč než originál Arduino Uno. Při práci s klonem došlo k menší komplikaci, kdy se poškodila paměť Arduina Una. Rozhodl jsem se vyzkoušet originál Arduino Uno v pořizovací hodnotě 700 Kč. Toto Arduino Uno funguje bez problému, a proto jsem se rozhodl, že by bylo nejvhodnější pro užití v kombinaci s robotem CoreXY v zájmovém kroužku jako řídicí jednotka.

K rozšíření komunikace s řídicí elektronikou robota CoreXY jsem vyzkoušel Raspberry Pi Zero v hodnotě přibližně 250 Kč. Na webových stránkách octoprint.org užití Raspberry Pi Zero nedoporučují. Raspberry Pi Zero vykazovalo známky nestability, před kterými jsem byl varován. Na webových stránkách octoprint.org doporučili Raspberry Pi 3B+. Toto Raspberry Pi 3B+ jsem použil a fungovalo zcela bez problému. Pořizovací cena Raspberry Pi 3B+ byla 1 100 Kč.

## 3.3 Sestavení a oživení CoreXY

Stavebnici robota CoreXY jsem sestavoval přibližně necelé 3 dny. Nejprve jsem musel zkontrolovat, jestli stavebnice obsahuje všechny díly. Jednotlivé díly jsem nafotil. Pořídil jsem také fotodokumentaci celého sestavení. Některé díly do sebe nepasovaly a bylo je nutné dodatečně zabrousit. Pro zabroušení jsem použil pilník s diamantovou úpravou a ruční vrtací brusku.

**Postup montáže CoreXY:**

1) Sestavení rámu



*Obrázek 5: Nesestavený rám*



*Obrázek 6: Sestavený rám*

2) Připevnění krokových motorů



*Obrázek 7: Nesestavené krokové motory*



*Obrázek 8: Uchycené krokové motory*

3) Připevnění vodící tyče s lineárními motory k rámu



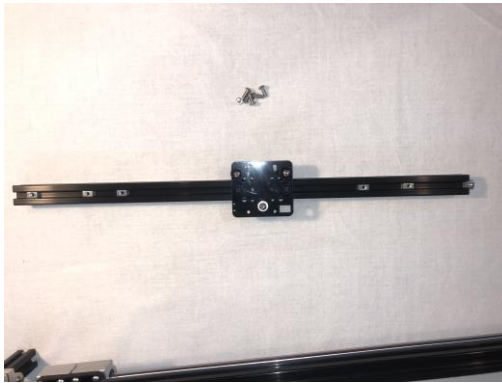
*Obrázek 9: Vodící tyče s lineárními ložisky*



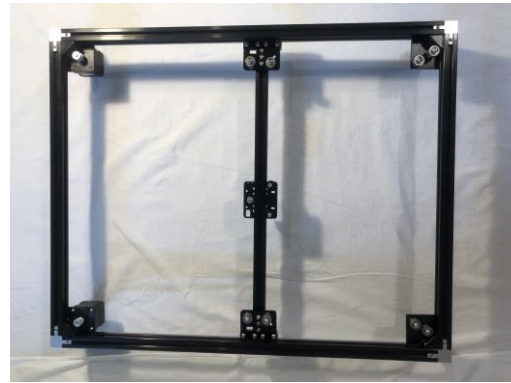
*Obrázek 10: Namontované vodící tyče*



4) Připevnění pohyblivého ramena k vodicím tyčím

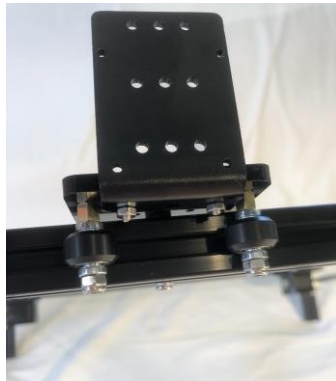


Obrázek 11: Pohyblivé rameno



Obrázek 12: Připevněné rameno ke konstrukci

5) Připevnění desky pro uchycení pera k platformě a napnutí ozubeného řetězu



Obrázek 13: Platforma s úchytem pro pero

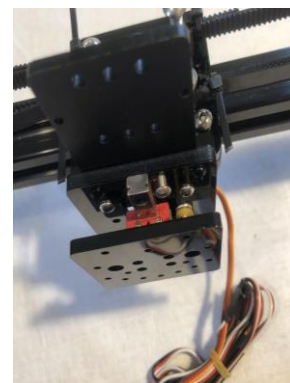


Obrázek 14: Natažený ozubený řemen

6) Nainstalování mechanismu se servomotorem



Obrázek 15: Servomotor s mechanismem



Obrázek 16: Připevněný servomotor s mechanismem

7) Připevnění držáku pera

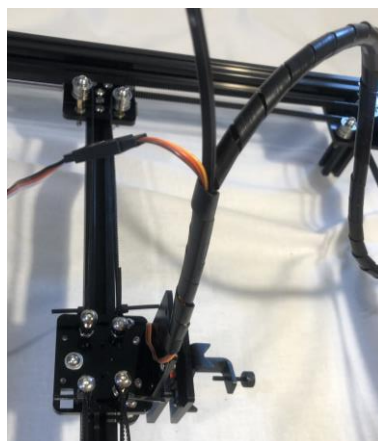


Obrázek 17: Držák na pero

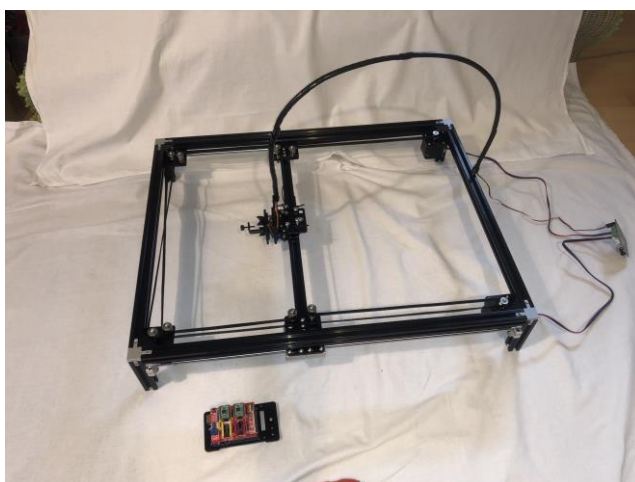


Obrázek 18: Připevněný držák na pero

8) Vývod kabelů k elektronice



Obrázek 19: Vyvedení kabelů



Obrázek 20: Sestavený robot CoreXY



## 4 Použití Arduina a Raspberry Pi pro řízení

Jako řídicí jednotku jsem zvolil Arduino Uno s nainstalovaným systémem GRBL. Na Arduino je připojena rozšiřující deska CNC Shield v3. Motor 1 a motor 2 jsou připojeny k rozšiřující desce Arduino Uno. U desky je nejprve nutné aktivovat mikro krokování pomocí několika jumperů, které se umístí na piny mezi sběrnice pro drivery. Každý motor má potom svůj vlastní driver A4988. Dále je k rozšiřující desce připojen i servomotor, který slouží pro ovládání pera. Aby krokové motory fungovaly, je potřeba do rozšiřující desky přivést dostačující externí napájení od 12–36 V z adaptéru.



Obrázek 21: Arduino Uno

Raspberry Pi slouží pro komunikaci s Arduinem Unem místo na přímo připojeného počítače. Raspberry Pi má na paměťové kartě nainstalovaný systém OctoPrint. K Raspberry Pi je dále připojena webová kamera. Aby Raspberry Pi fungovalo, musí být alespoň napájeno 5 V a 2 A pomocí Micro USB konektoru. Raspberry Pi by mělo být připojeno na přímo k Arduinu Uno se systémem GRBL.



Obrázek 22: Raspberry Pi

## 5 Vypracování modelových úloh

### 5.1 Konstrukce: plán robota

#### 5.1.1 Model CoreXY z kartonu

Jako modelovou úlohu jsem navrhl, aby si účastníci kurzu vyzkoušeli vytvořit mechanicky pohyblivý model robota CoreXY, na kterém by měli pochopit princip pohybu osy X a osy Y. Za tímto účelem jsem navrhl a vytvořil šablonu, podle které se bude model moci vystříhnout. Šablona je přiložena v příloze „Příloha D (šablona)“.

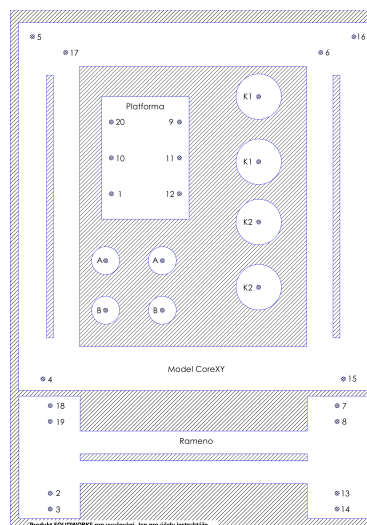
Pro vytvoření modelu účastníci kurzu budou potřebovat: karton o velikosti minimálně A4, vytištěnou šablonu, lepidlo, tavnou pistoli s náplní, odlamovací nůž, nůžky, pravítko, provázek a špejle.

#### **Pracovní postup:**

- 1) Nalepíme vytištěnou šablonu na karton.
- 2) Vyřízneme nebo vystříhneme části potřebné k sestavení modelu ze šablony.
- 3) Připravíme špejle do velikosti: 4 x 5,5 cm, 16 x 4 cm.
- 4) Sestavíme levý ovládací mechanismus.
  - a. Vezmeme část s označením K1 a přilepíme k ní část s označením A.
  - b. K části A přilepíme část s označením B.
  - c. K části B přilepíme část s označením K2.
  - d. Nyní zasuneme do otvoru slepené části 4 cm špejli.
  - e. Vložíme sestavený mechanismus do otvoru s číslem 4.
- 5) Sestavíme pravý ovládací mechanismus.
  - a. Vezmeme část s označením K1 a přilepíme k ní část s označením A.
  - b. K části A přilepíme část s označením B.
  - c. K části B přilepíme část s označením K2.
  - d. Nyní zasuneme do otvoru slepené části 4 cm špejli.
  - e. Vložíme sestavený mechanismus do otvoru s číslem 15.
- 6) Umístíme špejle do vyznačených míst na šabloně.
  - a. Špejle o velikosti 4 cm umístíme do otvorů s čísly: 1, 2, 3, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19 a 20.
  - b. Špejle o velikosti 5,5 cm umístíme do otvorů s čísly: 5, 6, 16 a 17.

- 7) Umístěné špejle v polovině přilepíme tavnou pistolí.
- 8) Složíme části modelu do sebe.
  - a. Díl s označením „Rameno“ vložíme do drážek v části s názvem „Model CoreXY“.
  - b. Platformu vsadíme do držáky v části s názvem „Rameno“, tak aby špejle s číslem 10 a 11 byly umístěny v drážce.
- 9) Natáhneme 1. provázek.
  - a. Začátek provázku uvážeme k vyznačené špejli s číslem 1.
  - b. Provázek natáhneme kolem špejle s číslem 2.
  - c. Provázek jednou obtočíme kolem levého ovládacího mechanismu.
  - d. Postupně pokračujeme s natažením kolem špejlí s čísly: 5, 6, 7 a 8.
  - e. Konec provázku uvážeme k vyznačené špejli s číslem 9, tak aby byl provázek lehce napnutý.
- 10) Natáhneme 2. provázek.
  - a. Začátek provázku uvážeme k vyznačené špejli s číslem 12.
  - b. Provázek natáhneme kolem špejle s číslem 13.
  - c. Provázek jednou obtočíme kolem pravého ovládacího mechanismu.
  - d. Postupně pokračujeme s natažením kolem špejlí s čísly: 16, 17, 18 a 19.
  - e. Konec provázku uvážeme k vyznačené špejli s číslem 20, tak aby byl provázek lehce napnutý.

Sestavený model robota CoreXY je ovladatelný otáčením levého a pravého ovládacího mechanismu.



Obrázek 23: Ukázka šablony pro model CoreXY

## 5.2 Elektronika: zapojení motorů, čidla

### 5.2.1 Zapojení motorů

Jako modelovou úlohu jsem navrhl, aby si účastníci kurzu vyzkoušeli zapojit krokové motory a servomotor do Arduina.

U krokového motoru jsou vyvedené kabely s konektorem. Konektor u krokového motoru pro osu X se musí zapojit do Arduina Uno, které je osazené CNC v3 shieldem. Piny pro tento konektor se nachází vedle driveru A4988 nad označením „X“. Konektor je natočen směrem, aby červený kabel byl zapojen na pin, který se nachází jako první nad označením „X“. Konektor u krokového motoru pro osu Y se musí také zapojit do Arduina Uno, které je osazené CNC v3 Shieldem. Piny pro tento konektor se nachází vedle driveru A4988 nad označením „Y“. Konektor je natočen směrem, aby červený kabel byl zapojen na pin, který se nachází jako první nad označením „Y“. U servomotoru jsou vyvedené kabely s konektorem. Konektor u servomotoru se musí zapojit do pinů na CNC v3 shieldu. Potřebné piny se nachází napravo od driveru pro osu Y s označením „5V“, „GND“ a „Z+“. Červený kabel vedoucí ze servomotoru, který by měl být kladný, zapojíme do pinu s označením „5V“. Černý kabel, který by měl být záporný, zapojíme do pinu s označením „GND“. Bílý kabel ze servomotoru zapojíme do pinu s označením „Z+“, který se nachází u seskupení „END STOPS“.

### 5.2.2 Zapojení čidel do Raspberry Pi

Jako modelovou úlohu jsem navrhl, aby si účastníci kurzu vyzkoušeli zapojit rozšiřující čidla do Raspberry Pi.

Účastník kurzu si může například vyzkoušet zapojit vhodnou webovou kameru do Raspberry Pi. Zda-li je webová kamera, kterou by chtěl účastník zapojit do Raspberry Pi, vhodná, zjistí v seznamu podporovaných a doporučených webových kamer na webových stránkách: <https://github.com/foosel/OctoPrint/wiki/Webcams-known-to-work>. Účastník kurzu vezme podporovanou webovou kameru a pomocí USB konektoru jí zapojí do Raspberry Pi.

## 5.3 Počítače HW: komunikace mezi Arduinem a RPi, s PC

### 5.3.1 Připojení Arduina

Jako modelovou úlohu jsem navrhl, aby si účastníci kurzu vyzkoušeli propojit Arduino s počítačem.

#### **Prostředky**

**HW:** Arduino Uno, USB kabel, PC

#### **Postup**

Účastníci kurzu umístí Arduino na dosah kabelu. Musí dbát na to, aby Arduino nebylo položené na kovovou nebo jinak vodící plochu. Arduino zapojí pomocí USB kabelu do počítače. Když je Arduino správně připojené, rozsvítí se na něm diody. Pokud připojení proběhlo v pořádku, zkontrolují ještě jeho nastavení. Stiskne se kombinace kláves Windows a písmena R. Do kolonky Otevřít se napíše „devmgmt.msc“ a stiskne se OK. Zobrazí se nám „Správce zařízení“. Ve správci zařízení rozkliknou „Porty (COM a LPT)“. Za názvem „Sériové zařízení USB“ uvidí v kulatých závorkách, pod jakým portem je Arduino připojené. Dále 2x kliknou na „Sériové zařízení USB“ a otevřou se vlastnosti. Ve vlastnostech vyberou „Nastavení portu“. Je nutno dbát na to, aby byly „Bity za sekundu“ nastaveny na hodnotu 115200. Pokud jsou bity nastaveny, mohou kliknout na tlačítko „OK“. Nyní je Arduino připraveno pro práci.

### 5.3.2 Nahrání GRBL

Do řídicí jednotky, která se skládá z Arduina uno, CNC Shield v3 a 2 driverů, studenti nahrají a přizpůsobí GRBL firmware pro CoreXY.

#### **Prostředky**

**HW:** Arduino Uno, CNC Shield v3, drivery, krokové motory, USB kabel, PC

**SW:** Arduino IDE, GrblController, GRBL firmware

#### **Postup**

Na webové stránce arduino.cc si stáhnou aktuální Arduino IDE, které budou potřebovat pro komunikaci a nahrání GRBL firmwaru do Arduina. Stažené Arduino IDE nainstalují. Jestliže nejsou práva pro instalaci, požádají správce. Dále si stáhnou

GRBL firmware. Do webového prohlížeče zadají webovou adresu: <https://github.com/gnea/grbl>. Na webové stránce kliknou na zelené tlačítko označené "Clone or download", zobrazí se jim nabídka a zvolí "Download Zip". Po stažení a rozbalení souboru extrahují jeho obsah, například do vytvořené složky na ploše. Otevrou si složku, ve které mají extrahovaný GRBL zip. Otevrou složku "grbl", dále složku "examples", a potom složku "grblUpload". Ve složce "grblUpload" spustí Arduino soubor "grblUpload.ino". Když se načte Arduino IDE, musí přidat grbl knihovnu. V horní liště najedou na záložku „Projekt“, dále pak na „Přidat knihovnu“ a vyberou „Přidat ZIP knihovnu...“. Nyní musí najít složku s extrahovaným GRBL. Otevrou ji a vyberou podsložku "grbl". Arduino IDE si minimalizují na lištu. Pokud již někdy v minulosti přidávali knihovnu grbl, budou ji nejprve muset vymazat. Knihovna by se měla nacházet v Documents\Arduino\libraries a složku grbl odstraní. Když je knihovna vytvořena, je nutné změnit soubor "config.h" nacházející se v Documents\Arduino\libraries\grbl. Soubor lze otevřít v libovolném textovém editoru. Doporučuji použít software PSPad. Když mají soubor config.h otevřený, hledají řádek 112 a 113, na kterém se nachází:

```
// #define HOMING_CYCLE_0 (1<<X_AXIS) // COREXY
    COMPATIBLE: First home X
// #define HOMING_CYCLE_1 (1<<Y_AXIS) // COREXY
    COMPATIBLE: Then home Y
```

Odstraní „//“ dále pak odstraní „//“ na řádku 189, na kterém je:

```
// #define COREXY // Default disabled. Uncomment to enable.
```

Přidají „//“ na začátek řádku 105 a 106, na kterém se nachází:

```
#define HOMING_CYCLE_0 (1<<Z_AXIS) // REQUIRED: First
    move Z to clear workspace.
#define HOMING_CYCLE_1 ((1<<X_AXIS)|(1<<Y_AXIS)) //
    OPTIONAL: Then move X,Y at the same time.
```

Soubor uloží.

Otevrou znovu Arduino IDE, kde mají otevřené grblUpload. Připojí Arduino. Pro kontrolu, zda je Arduino správně připojené, najedou do horní lišty, kde se nachází záložka „Nástroje“. Ve „Vývojová deska:“ zvolí možnost Arduino Uno. Dále

v nástrojích zvolí port připojeného Arduina. Jak zjistit port, pod kterým je připojené Arduino je popsáno v kapitole „Připojení Arduina“. Dále pak kliknou v „Nástrojích“ na „Základní informace o desce“. Pokud je vyplněn VID, PID a SW, mohou nahrát GRBL do Arduina. V horní liště kliknou na projekt a zvolí „Kontrola/Kompilace“. Pokud se po dokončení kontroly a kompilace v terminálu zobrazí pouze varovná hláška o malé paměti a problémech se stabilitou, mohou projekt nahrát do Arduina. Nahrání se provedeme, klikne-li se v horní liště na „Projekt“ a dále na „Nahrát“, nebo lze kliknout na šipku v kolečku pod horní lištou. Po dokončení zkontrolují, jestli GRBL s nimi správně komunikuje. V horní liště Arduino IDE kliknou na „Nástroje“, dále pak na „Sériový monitor“. Otevře se okno. V pravém dolním rohu si vyberou možnost „Nová řádka (NL)“ a „115200 baudů“. Do řádky napíše symbol „\$“. Jestli se zobrazí:

```
[HLP:$ $ # $G $I $N $x=val $Nx=line $J=line $SLP $C $X $H  
~ ! ? ctrl-x]  
Ok
```

Arduino s GRBL komunikuje s PC správně. Mohou ještě vyzkoušet napsat „\$\$“. Tato varianta zobrazí nastavené parametry GRBL. [9]

### 5.3.3 Kalibrace CoreXY

Může se stát, že CoreXY vykonává jinou vzdálenost, než byla zadána. Například máte zadáno, aby se platforma s perem posunula o 20 mm, ale platforma s perem se posune o 50 mm.

#### **Prostředky**

**HW:** robota CoreXY, PC

**SW:** PSPad, GrblController

#### **Postup**

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Do textového editoru napíše G-code, který se nachází níže v „Příklad 2: G-code Testovací čtverec“.

```

M05 S0
G90
G21
G1 F1000.00000
G1 X0 Y20
G1 X20 Y20
G1 X20 Y0
G1 X0 Y0

```

Příklad 2: G-code Testovací čtverec

Tento G-code spustí pero, nastaví jednotky na milimetry, posunovou rychlost na 1000 a pak vykoná souřadnice pro namalování 2 cm čtverce. Tento čtverec slouží jako testovací čtverec. Napsaný G-code uloží. V horní liště PSPadu vybere záložku soubor a zvolí „Uložit jako..“. Zobrazí se mu okno. Vybere, kam chce svůj G-code uložit, např. na „Plochu“. Do kolonky „Název souboru“ napíše jméno souboru např. „TestovacíCtverec“ a za jméno napíše „.gcode“. Dále pak student vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“. Nyní může stisknout tlačítko „Uložit“. Jako první musí připojit robota CoreXY k počítači. Spustí nainstalovaný GrblController. Když se program spustí, musí nastavit port, na kterém má připojený CoreXY. Dále musí nastavit přenosovou rychlost na 115200. Stiskne tlačítko „Otevřít“. Když je robot CoreXY připojený, tlačítko zčervená a změní se na „Zavřít/Restartovat“.

#### Nahrání souboru C-code s testovacím čtvercem:

Pod tlačítkem „Zavřít/Restartovat“ je kolonka pro zadání kompletní cesty k G-code souboru, nebo může student stisknout tlačítko „Vybrat soubor“. Student vybere testovací čtverec, který chce načíst a stiskne „Open“. Vizualizace nám zobrazí nahraný G-code. Na pravé straně pod „Strojní souřadnice“ a „Pracovní souřadnice XYZ“ jsou 3 záložky. Student klikne na „Vizualizace“. Ve vizualizaci vidí, jakou cestu bude robot CoreXY vykonávat. Dále je možno vidět „šířku-x“ a „výšku-y“, v našem případě bychom měli mít X 20 mm a Y 20 mm. Pokud vše souhlasí, přichytí student pero tak, aby se dotýkalo papíru, a nyní může spustit nahraný G-code pomocí tlačítka „Start“, které se nachází na levé straně pod cestou, kde je nahraný G-code. Když se čtverec nakreslí, přeměří ho student pomocí pravítka. Pokud jsou strany různě velké, je



zapotřebí změnit nastavení GRBL. Do kolonky pro příkaz se zadá „\$\$“. Vypíše se nastavení GRBL. Student zájmového kroužku musí najít, na jaké hodnoty jsou nastavené osy X a Y. Hodnoty pro osu X jsou pod „\$100“. Hodnoty pro osu Y jsou pod „\$101“. [10]

### **Kalibrační čtverec je větší, než má být**

Když je strana čtverce na ose X delší než 2 cm, musí se číselná hodnota za znaménkem „=“ zmenšovat do doby, dokud nebude strana čtverce 2 cm.

#### **Například:**

Ve výpisu je, že osa X má hodnotu 250:

```
$100=250.000
```

Napíše se tedy příkaz:

```
$100=101.000
```

Mohlo by se také napsat:

```
$100=101
```

Když je strana čtverce na ose Y delší než 2 cm, musí se číselná hodnota za znaménkem „=“ zmenšovat do doby, dokud nebude strana čtverce 2 cm.

#### **Například:**

Ve výpisu je, že osa Y má hodnotu 250:

```
$101=250.000
```

Napíše se tedy příkaz:

```
$101=101.000
```

Mohlo by se také napsat:

```
$101=101
```

### **Kalibrační čtverec je menší, než má být**

Když je strana čtverce na ose X menší než 2 cm, musí se číselná hodnota za znaménkem „=“ zvětšovat do doby, dokud nebude strana čtverce 2 cm.

**Například:**

Ve výpisu je, že osa X má hodnotu 250:

```
$100=250.000
```

Napiše se tedy příkaz:

```
$100=325.000
```

Mohlo by se také napsat:

```
$100=325
```

Když je strana čtverce na ose Y menší než 2 cm, musí se číselná hodnota za znaménkem „=“ zvětšovat do doby, dokud nebude strana čtverce 2 cm.

**Například:**

Ve výpisu je, že osa Y má hodnotu 250:

```
$101=250.000
```

Napiše se tedy příkaz:

```
$101=325.000
```

Mohlo by se také napsat:

```
$101=325
```

### 5.3.4 Instalace OctoPrint do Raspberry Pi

Pokusíme se zprovoznit Raspberry Pi s webovou kamerou.

**Prostředky**

**HW:** Raspberry Pi 3B+, PC, USB kabel, microSD kartu (4-32GB), microSD adapter do PC

**SW:** OctoPrint, Etcher, PUTTy

**Postup**

Na webových stránkách octoprint.org si stáhneme systém. Stiskneme zelené tlačítko „Download“ a po přesměrování dáme stahovat komprimovaný soubor OctoPrint pomocí tlačítka „Download OctoPi 0.16.0“. Když se stáhne komprimovaný

soubor, rozbálíme ho a extrahujeme obsah například na „Plochu“. Dále bude zapotřebí stáhnout program, například Etcher, který využijeme na zavedení OctoPrint na SD kartu. Otevřeme si webovou stránku [balena.io/etcher](https://balena.io/etcher). Rozklikneme zelené tlačítko „Download for Windows (x86/x64)“ pomocí malé šipky v pravém okraji a vybereme „Etcher for Windows (x86/x64) (Portable)“. Jedná se o program, který se nebude muset instalovat a jen se spustí. Do počítače vložíme microSD kartu. Pokud nemáme slot v PC pro microSD kartu, budeme potřebovat USB adaptér. Spustíme program Etcher. Klikneme na modré tlačítko „Select image“ a vybereme extrahovaný soubor OctoPrint (2018-11-13-octopi-stretch-lite0.16.0). Dále vybereme SD kartu, na kterou chceme nahrát systém OctoPrint. Klikneme na „Flash!“. Po nahrání systému kartu odpojíme a znovu připojíme. Načtenou kartu otevřeme a soubor „octopi-wpa-supPLICANT.txt“ otevřeme v textovém editoru PSpad. V souboru budeme vybírat způsob připojení do lokální sítě pomocí Wi-Fi. Podle lokální sítě vybereme vhodné nastavení. Když máme zabezpečení Wi-Fi WPS/WPA2, musíme odstranit „#“ na začátku řádků 27, 28, 29, 30 a poté doplnit místo „put SSID here“ název Wi-Fi sítě a „put password here“ heslo k Wi-Fi síti. Když je Wi-Fi síť nezabezpečená, odstraníme „#“ ze začátku řádků 33, 34, 35, 36 a doplníme místo „put SSID here“ název Wi-Fi sítě. Když máme zabezpečení Wi-Fi sítě pomocí WEP, odstraníme „#“ ze začátku řádků 43, 44, 45, 46, 47 a doplníme místo „put SSID here“ název Wi-Fi sítě a místo „put password here“ heslo k Wi-Fi síti. Soubor uložíme a zavřeme. Bezpečně odpojíme microSD kartu. MicroSD kartu vložíme do Raspberry Pi. K Raspberry Pi připojíme webovou kameru. Pomocí microUSB propojíme Raspberry Pi s PC. Chvilku počkáme a vyzkoušíme, jestli je Raspberry Pi připojené k lokální síti. Stiskneme kombinaci tlačítka Win+R. Zobrazí se nám okno. Napíšeme „cmd“ a klikneme na OK. Spustí se nám rozhraní příkazového řádku. Do konzole napíšeme „ping octopi.local“. Když „Lost=0“ znamená to, že Raspberry Pi je viditelné a správně připojené do lokální sítě. Nyní si stáhneme program PUTTy ze stránek: <https://www.putty.org>. Spustíme program PUTTy. Do „Host Name (or IP address)“ zadáme „octopi.local“ nebo IP adresu Raspberry Pi, port necháme na 22 a „Connection type“ dáme na SSH. Klikneme „Open“. Zobrazí se černé okno a také okno s informací o bezpečnosti. Pokud máme jistotu, že se připojujeme k zařízení, ke kterému chceme, můžeme u bezpečnostního okna souhlasit a zvolit „Yes“. PUTTy si uloží informace o zařízení, a kdyby nastala nějaká změna, bude nás varovat, abychom nechtěně nedali své informace podvodnému

zařízení. Přihlašovací jméno do OctoPrint je „pi” a heslo je „raspberry”. Po přihlášení zadáme příkaz „sudo raspi-config”. V configu můžeme změnit heslo, časovou zónu nebo název zařízení. Do webového prohlížeče zadáme „http://octopi.local” nebo IP adresu Raspberry Pi s OctoPrint. Zobrazí se „Setup Wizard” (průvodce nastavení) a klikneme na „next”. V „access control” si zvolíme přihlašovací jméno a heslo k OctoPrint. Stiskneme tlačítko „Keep access control enabled”. V „Anonymous usage tracking” zvolíme, že souhlasíme. V „online connectivity check” stiskneme tlačítko „Enable connectivity check”. Povolíme „Configure plugin blacklist processing”. V „set up your printer profile” pojmenujeme robota a model. Zakončíme tlačítkem „finish”. Nyní nainstalujeme rozšiřující plugin GRBL do OctoPrint. Otevřeme si nastavení. Najdeme „Plugin Manager” a klikneme na tlačítko „Get More...”. Otevře se nám okno „Install new Plugins...”. Do kolonky s „...from URL” vložíme odkaz pro přidání rozhraní GRBL:

<https://github.com/synman/OctoPrint-Bettergrblsupport/archive/master.zip>

a klikneme na tlačítko „Install”. Po instalaci restaurujeme Raspberry Pi. Nyní máme OctoPrint připravený pro práci. [4]

## 5.4 počítače SW: programování Arduina, RPi

### 5.4.1 Úloha Start

Student dostane za úkol napsat G-code, kterým robot CoreXY vykoná pohyb po ose X, následně po ose Y, dále pak s robotem CoreXY vykoná diagonální pohyb a pohyb nahoru a dolů se servomotorem, kterým se ovládá pero. Student vše zakončí návratem na začátek.

#### **Prostředky**

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G21, M03, G4, M05

#### **Postup**

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první musí student nastavit rychlost, kterou se bude pohybovat hlava: **G1 F3000** Rychlost se mění pomocí změny hodnoty za písmenem F. Dále musí nastavit, v jakých jednotkách se

bude pohybovat: pro milimetry napíše **G21**, pro palce napíše **G20**. Pro posun platformy po ose X musí student napsat: **G1 X50 Y0** nebo **G1 X50**. Za X může dosadit libovolnou velikost, ale nesmí překročit velikost pracovní plochy. Pro posun platformy po ose Y musí student napsat: **G1 X50 Y50** nebo **G1 Y50**. Za Y může dosadit libovolnou velikost, ale nesmí překročit velikost pracovní plochy. Pro diagonální posun platformy, musí student napsat například: **G1 X25 Y25** Pro aktivaci (zvednutí pera) bude muset student napsat: **M03 S255**. Tento příkaz aktivuje servomotor. Pero musí vykonat cestu nahoru, a protože tento úkon trvá určitý čas, je proto vhodné nastavit pauzu cca 2 vteřiny pomocí příkazu: **G4 P2**. Hodnota za písmenem P určuje, kolik vteřin bude robot čekat. Pro položení pera musí student napsat: **M05 S0**. Pro návrat platformy na začátek, musí student napsat: **G1 X0 Y0**. Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může například uložit na „Plochu“. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „gcode“. Dále pak vybere v poli „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

### 5.4.2 Úloha Úsečka

Student dostane za úkol napsat G-code, kterým robot CoreXY vykoná úsečku o délce 10 cm kdekoliv v pracovní ploše.

Pro ulehčení získání souřadnic X a Y může použít mnou vytvořenou šablonu „Pomocná šablona pro určení souřadnic X a Y“. Jak pracovat se šablonou lze nalézt v odstavci „5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y“.

Robot CoreXY bude muset dojet na libovolné místo v pracovní ploše, spustit pero, vytvořit úsečku, zvednout pero a dostat se zpět na začátek.

#### Prostředky

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G4, G90, G21, M05, M03

## Postup

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první příkaz sloužící pro zvednutí pera napíše:

```
M03 S255
```

Absolutní režim odměřování:

```
G90
```

Nastavení jednotek souřadného systému na milimetry:

```
G21
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, na které se má pero přesunout:

```
G1 X45 Y113
```

Příkaz pro položení pera a začátku kreslení:

```
M05 S0
```

Pero musí vykonat cestu dolů a tento úkon trvá určitý čas, proto je vhodné nastavit pauzu cca 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout:

```
G1 X17 Y18
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení vrácení do výchozí polohy:

```
G1 X0 Y0
```

Položení pera:

```
M05 S0
```

Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může uložit například na Plochu. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „gcode“. Dále pak vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

### 5.4.3 Úloha Čtverec

Student dostane za úkol napsat G-code, kterým robot CoreXY nakreslí čtverec o rozměrech 5 cm x 5 cm kdekoliv v pracovní ploše.

Pro ulehčení získání souřadnic X a Y může použít mnou vytvořenou šablonu „Pomocná šablona pro určení souřadnic X a Y“. Jak pracovat se šablonou lze nalézt v odstavci „5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y“.

Platforma s perem bude muset dojet na libovolné místo v pracovní ploše, spustit pero, vytvořit čtverec, zvednout pero a dostat se zpět na začátek.

#### **Prostředky**

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G4, G90, G21, M05, M03

## Postup

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první příkaz sloužící pro zvednutí pera napíše:

```
M03 S255
```

Pro absolutní režim odměřování musí napsat:

```
G90
```

Nastavení jednotek souřadného systému na milimetry:

```
G21
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, na které se má pero přesunout:

```
G1 X50 Y50
```

Příkaz pro položení pera, aby začalo kreslit:

```
M05 S0
```

Pero musí vykonat cestu dolů a tento úkon trvá určitý čas, proto je vhodné nastavit pauzu cca 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení první strany čtverce:

```
G1 X50 Y100
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení druhé strany čtverce:

```
G1 X100 Y100
```



Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení třetí strany čtverce:

```
G1 X100 Y50
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení čtvrté strany čtverce:

```
G1 X50 Y50
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení vrácení do výchozí polohy:

```
G1 X0 Y0
```

Položení pera:

```
M05 S0M05 S0
```

Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může uložit například na Plochu. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „,gcode“. Dále pak vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

### 5.4.4 Úloha Obdélník

Student dostane za úkol napsat G-code, kterým robot CoreXY nakreslí obdélník o rozměrech 5 cm x 10 cm kdekoliv v pracovní ploše.

Pro ulehčení získání souřadnic X a Y může použít mnou vytvořenou šablonu „Pomocná šablona pro určení souřadnic X a Y“. Jak pracovat se šablonou lze nalézt v odstavci „5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y“.

Platforma s perem bude muset dojet na libovolné místo v pracovní ploše, spustit pero, vytvořit obdélník, zvednout pero a dostat se zpět na začátek.

### Prostředky

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G4, G90, G21, M05, M03

### Postup

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první příkaz sloužící pro zvednutí pera napíše:

```
M03 S255
```

Pro absolutní režim odměřování musí napsat:

```
G90
```

Nastavení jednotek souřadného systému na milimetry:

```
G21
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, na které se má pero přesunout:

```
G1 X50 Y50
```

Příkaz pro položení pera, aby začalo kreslit:

```
M05 S0
```

Pero musí vykonat cestu dolů a tento úkon trvá určitý čas, proto je vhodné nastavit pauzu cca 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení krátké strany obdélníku:

```
G1 X50 Y100
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení dlouhé strany obdélníku:

```
G1 X150 Y100
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení krátké strany obdélníku:

```
G1 X150 Y50
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení dlouhé strany obdélníku:

```
G1 X50 Y50
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení vrácení do výchozí polohy:

```
G1 X0 Y0
```

Položení pera:

```
M05 S0
```

Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může uložit například na Plochu. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „,gcode“. Dále pak vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější

informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

### 5.4.5 Úloha Trojúhelník 1

Student dostane za úkol napsat G-code, kterým robot CoreXY nakreslí velký rovnoramenný trojúhelník o základně délky 10 cm a výšce 15 cm kdekoli v pracovní ploše.

Pro ulehčení získání souřadnic X a Y může použít mnou vytvořenou šablonu „Pomocná šablona pro určení souřadnic X a Y“. Jak pracovat se šablonou lze nalézt v odstavci „5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y“.

Platforma s perem bude muset dojet na libovolné místo v pracovní ploše, spustit pero, vytvořit trojúhelník, zvednout pero a dostat se zpět na začátek.

#### Prostředky

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G4, G90, G21, M05, M03

#### Postup

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první příkaz sloužící pro zvednutí pera napíše:

```
M03 S255
```

Pro absolutní režim odměřování musí napsat:

```
G90
```

Nastavení jednotek souřadného systému na milimetry:

```
G21
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, na které se má pero přesunout:

```
G1 X50 Y50
```

Příkaz pro položení pera, aby začalo kreslit:

```
M05 S0
```

Pero musí vykonat cestu dolů a tento úkon trvá určitý čas, proto je vhodné nastavit pauzu cca 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení základny trojúhelníku:

```
G1 X150 Y50
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení prvního ramena trojúhelníku:

```
G1 X100 Y200
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení druhého ramena trojúhelníku:

```
G1 X50 Y50
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení vrácení do výchozí polohy:

```
G1 X0 Y0
```

Položení pera:

```
M05 S0
```

Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může uložit například na Plochu. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „gcode“. Dále pak vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

### 5.4.6 Úloha Trojúhelník 2

Student dostane za úkol napsat G-code, kterým robot CoreXY nakreslí pravoúhlý trojúhelník o základně 10 cm a výšce 3 cm kdekoliv v pracovní ploše.

Pro ulehčení získání souřadnic X a Y může použít mnou vytvořenou šablonu „Pomocná šablona pro určení souřadnic X a Y“. Jak pracovat se šablonou lze nalézt v odstavci „5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y“.

Platforma s perem bude muset dojet na libovolné místo v pracovní ploše, spustit pero, vytvořit trojúhelník, zvednout pero a dostat se zpět na začátek.

#### Prostředky

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G4, G90, G21, M05, M03

#### Postup

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první příkaz sloužící pro zvednutí pera napíše:

```
M03 S255
```

Pro absolutní režim odměřování musí napsat:

```
G90
```

Nastavení jednotek souřadného systému na milimetry:

```
G21
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, na které se má pero přesunout:

```
G1 X50 Y50
```

Příkaz pro položení pera, aby začalo kreslit:

```
M05 S0
```

Pero musí vykonat cestu dolů a tento úkon trvá určitý čas, proto je vhodné nastavit pauzu cca 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení odvěsny trojúhelníku:

```
G1 X150 Y50
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení přepony trojúhelníku:

```
G1 X50 Y80
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení odvěsny trojúhelníku:

```
G1 X50 Y50
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení vrácení do výchozí polohy:

```
G1 X0 Y0
```

Položení pera:

```
M05 S0
```

Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může uložit například na Plochu. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „.gcode“. Dále pak vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

### 5.4.7 Úloha Čtverec ve čtverci

Student dostane za úkol napsat G-code, kterým robot CoreXY nakreslí čtverec o straně 13 cm, ve kterém se bude nacházet další čtverec o straně 9 cm kdekoliv v pracovní ploše.

Pro ulehčení získání souřadnic X a Y může použít mnou vytvořenou šablonu „Pomocná šablona pro určení souřadnic X a Y“. Jak pracovat se šablonou lze nalézt v odstavci „5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y“.

Platforma s perem bude muset dojet na libovolné místo v pracovní ploše, spustit pero, vytvořit čtverec, zvednout pero, přesunout se do čtverce, nakreslit čtverec a dostat se zpět na začátek.

#### **Prostředky**

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G4, G90, G21, M05, M03

#### **Postup**

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první příkaz sloužící pro zvednutí pera napíše:

```
M03 S255
```



Pro absolutní režim odměřování musí napsat:

```
G90
```

Nastavení jednotek souřadného systému na milimetry:

```
G21
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, na které se má pero přesunout:

```
G1 X20 Y20
```

Příkaz pro položení pera, aby začalo kreslit:

```
M05 S0
```

Pero musí vykonat cestu dolů a tento úkon trvá určitý čas, proto je vhodné nastavit pauzu cca 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení první strany většího čtverce:

```
G1 X20 Y150
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení druhé strany většího čtverce:

```
G1 X150 Y150
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení třetí strany většího čtverce:

```
G1 X150 Y20
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení čtvrté strany většího čtverce:

```
G1 X20 Y20
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, pro přesun pera do vnitřku většího čtverce:

```
G1 X40 Y40
```

Příkaz pro položení pera, aby začalo kreslit:

```
M05 S0
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení první strany menšího čtverce:

```
G1 X40 Y130
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení druhé strany menšího čtverce:

```
G1 X130 Y130
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení třetí strany menšího čtverce:

```
G1 X130 Y40
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení čtvrté strany menšího čtverce:

```
G1 X40 Y40
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení vrácení do výchozí polohy:

```
G1 X0 Y0
```

Položení pera:

```
M05 S0
```

Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může uložit například na Plochu. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „gcode“. Dále pak vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

### 5.4.8 Úloha kosočtverec

Student dostane za úkol napsat G-code, kterým robot CoreXY nakreslí libovolný kosočtverec o straně 5,5 cm kdekoliv v pracovní ploše.

Pro ulehčení získání souřadnic X a Y může použít mnou vytvořenou šablonu „Pomocná šablona pro určení souřadnic X a Y“. Jak pracovat se šablonou lze nalézt v odstavci „5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y“.

Platforma s perem bude muset dojet na libovolné místo v pracovní ploše, spustit pero, vytvořit čtverec, zvednout pero a dostat se zpět na začátek.

### Prostředky

**HW:** robot CoreXY

**SW:** PSPad, GrblController nebo OctoPrint

**Použité kódy:** G1, G4, G90, G21, M05, M03

### Postup

Student si otevře PSPad nebo jakýkoliv libovolný textový editor. Jako první příkaz sloužící pro zvednutí pera napíše:

```
M03 S255
```

Pro absolutní režim odměřování musí napsat:

```
G90
```

Nastavení jednotek souřadného systému na milimetry:

```
G21
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení souřadnice X a Y, na které se má pero přesunout:

```
G1 X50 Y50
```

Příkaz pro položení pera, aby začalo kreslit:

```
M05 S0
```

Pero musí vykonat cestu dolů a tento úkon trvá určitý čas, proto je vhodné nastavit pauzu cca 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu při kreslení:

```
G1 F1000
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení první strany kosočtverce:

```
G1 X30 Y100
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení druhé strany kosočtverce:

```
G1 X50 Y150
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení třetí strany kosočtverce:

```
G1 X70 Y100
```

Nastavení souřadnice X a Y, na které se má kreslicí pero přesunout pro nakreslení čtvrté strany kosočtverce:

```
G1 X50 Y50
```

Příkaz pro zvednutí pera:

```
M03 S255
```

Pauza 2 vteřiny:

```
G4 P2
```

Nastavení rychlosti přesunu:

```
G1 F3000
```

Nastavení vrácení do výchozí polohy:

```
G1 X0 Y0
```

Položení pera:

```
M05 S0
```

Nyní student uloží napsaný G-code. V horní liště PSPadu vybere záložku „Soubor“ a zvolí „Uložit jako...“. Zobrazí se mu okno. Vybere, kam chce svůj napsaný G-code uložit. G-code může uložit například na Plochu. Do kolonky „Název souboru“ napíše název, jak se soubor bude jmenovat a za název napíše „.gcode“. Dále pak vybere v „Uložit jako typ“ možnost „Všechny soubory (\*.\*)“ Nyní může stisknout tlačítko „Uložit“. Student vyzkouší svůj uložený G-code na robotovi CoreXY. Podrobnější informace, jak odeslat svůj G-code do robota CoreXY, lze nalézt v kapitolách: „5.5.1 Práce s OctoPrint“ nebo „5.5.2 Odeslání G-code přes GrblController“.

## 5.5 Práce se SW a podpůrnými prostředky

### 5.5.1 Práce s OctoPrint

Nejprve musíme připojit CoreXY k Raspberry Pi. V „Connection“ zvolíme port CoreXY, „Baudrate“ na 115200 a stiskneme tlačítko pro připojení. V „State“ vidíme, jestli je robot správně připojen, a tudíž online. V sekci „Files“ můžeme nahrát několik G-code souborů, které jdou třídit do adresářů. V záložce „Grbl Control“ máme zobrazení z kamery. V „Starting Position“ můžeme zvolit, odkud bude platforma startovat. Ve stejné záložce lze také nastavit velikost pracovní plochy, nebo zde můžeme využít i ruční ovládání pomocí šipek. V záložce „Terminal“ můžeme ručně zadávat G-code. V záložce „Timelapse“ lze získat video z průběhu práce.

### 5.5.2 Odeslání souboru G-code přes GrblController

Student bude komunikovat s robotem CoreXY skrze software GrblController a G-code soubor.

#### Postup

Jako první musí student připojit robota CoreXY k počítači pomocí USB kabelu. Spustí nainstalovaný GrblController. Když se program spustí, musí student nastavit port, na kterém je připojený CoreXY. Dále nastaví přenosovou rychlost na 115200. Stiskne tlačítko „Otevřít“. Když je robot CoreXY připojený, tlačítko zčervená a změní se na „Zavřít/Restartovat“.

#### Nahrání souboru G-code

Pod tlačítkem „Zavřít/Restartovat“ je kolonka pro zadání kompletní cesty k G-code souboru. Pokud student nezná kompletní cestu, kde se nachází G-code, může stisknout tlačítko „Vybrat soubor“. Vybere G-code soubor, který chce načíst a stiskne „Open“. Na pravé straně pod „Strojní souřadnice“ a „Pracovní souřadnice XYZ“ jsou 3 záložky. Klikne na „Vizualizace“. Ve vizualizaci vidí, jakou cestu bude robot CoreXY vykonávat. Dále může vidět „šířku-x“ a „výšku-y“. Pokud vše souhlasí a šířka s výškou nepřekračují pracovní plochu, může spustit nahraný G-code pomocí tlačítka „Start“, které se nachází na levé straně pod cestou, kde je nahraný G-code. Když by chtěl předčasně zastavit stroj, musí stisknout tlačítko „Stop“. Může se stát, že se robot

CoreXY nezastaví, potom musí student vyčkat, až dokončí robot CoreXY nahrany úkon.

### 5.5.3 Přizpůsobení Inkscape pro CoreXY

Nahrajeme plugin do Inkscape a nastavíme ho pro práci s perem.

Inkscape je open source vektorový grafický editor. Lze stáhnout z oficiálních stránek <https://inkscape.org/cs/>. Po stažení a nainstalování Inkscape musíme ještě stáhnout rozšíření „J Tech Laser Tool Plugin“. Plugin pro verzi Inkscape 0.92 a 0.91 lze nalézt na webové stránce [https://jtechphotonics.com/?page\\_id=1980](https://jtechphotonics.com/?page_id=1980). Otevřeme adresář, ve kterém je nainstalovaný Inkscape, dále otevřeme složku „share“ a do složky „extensions“ nahrajeme obsah staženého pluginu. Nyní spustíme software Inkscape. Nyní nastavíme vlastnosti dokumentu. V horní liště vybereme „Soubor“ a klikneme na „Vlastnosti dokumentu...“. Otevře se okno „Vlastnosti dokumentu“, zde změním zobrazovací jednotky na „mm“ a velikost stránky zvolíme podle plochy, kterou můžeme použít, například A4. Okno „Vlastnosti dokumentu“ můžeme zavřít pomocí červeného křížku. V horní liště „Rozšíření“ vybereme „Generate Laser Gcode“ a zvolíme „J Tech Photonics Laser Tool...“. Otevře se nám okno. Nyní změním pár parametrů. U „Laser ON Command:“ nastavíme hodnotu na „M05“, dále u „Laser OFF Command:“ nastavíme hodnotu na „M03 S255(“. Následně nastavíme 0 u „Laser Power S# (0-255 or 0-12000):“. V poli „Složka:“ si nastavíme cestu, kam se nám bude ukládat vygenerovaný G-code. V „Název souboru“ si můžeme zvolit, jak se soubor bude jmenovat. Nyní máme přednastavený Inkscape pro další použití.

### 5.5.4 Kreslení v Inkscape

Otevřeme software Inkscape. Pomocí lišty nástrojů v levé části nakreslí obrázek do pracovní plochy, který budeme chtít, aby nakreslil robot CoreXY. Po nakreslení vyznačíme všechny objekty. V horní liště „Křivka“ vybereme možnost „Objekt na křivku“. Nyní se všechny objekty převedou na křivky. Pro převedení křivek do G-code musíme použít rozšíření „J Tech Photonics Laser Tool...“, které se nachází v „Rozšíření“ pod „Generate Laser Gcode“. Nastavení „J Tech Photonics Laser Tool...“ je popsáno v kapitole „5.5.3 Přizpůsobení Inkscape pro CoreXY“.

### 5.5.5 Převod obrázku pomocí Inkscape

Otevřeme software Inkscape. Nahrajeme vlastní nebo stažený obrázek do pracovní plochy, který budeme chtít, aby namaloval robot CoreXY. Abychom obrázek nahráli musíme v horní záložce „Soubor“ vybrat „Import...“. Nyní najdeme obrázek, který chceme nahrát a dáme „Otevřít“. V otevřeném okně nemusíme nic měnit a klikneme na tlačítko „Budiž“. Po importu vyznačíme obrázek. V horní liště „Křivka“ vybereme možnost „Vektorizovat bitmapu...“. Zde máme několik možností. Vybereme tu nejvhodnější podle nahraného obrázku, když si zaškrtneme živý náhled, tak uvidíme, jak výsledný obrázek bude přibližně vypadat. Když jsme s nastavením spokojeni, klikneme na tlačítko „Budiž“. Nyní převedený obrázek posuneme a klikneme na původní, abychom ho odstranili. Vektorový obrázek nyní můžeme zvětšovat, zmenšovat, přesouvat nebo deformovat. Pro převedení vektorového obrázku do G-code musíme použít rozšíření „J Tech Photonics Laser Tool...“, který se nachází v „Rozšíření“ pod „Generate Laser Gcode“. Nastavení „J Tech Photonics Laser Tool...“ je popsáno v kapitole „5.5.3 Přizpůsobení Inkscape pro CoreXY“.

### 5.5.6 Práce s pomocnou šablonou pro určení souřadnic X a Y

Pro tento kurz jsem navrhl a vytvořil pomocnou šablonu pro určení souřadnic X a Y. Pomocí této šablony může student jednoduše zjistit souřadnice bodů X a Y z obrázku nakresleného na šabloně a zaznamenat si je. G-code lze pomocí těchto zjištěných souřadnic bodů X a Y snadněji napsat.

Student nakreslí obrazec z úseček. Zvolí si počáteční bod, odkud chce začít obrazec malovat. Zjistí na ose označené X souřadnici pro vybraný bod. Zaznamená ho do připravené tabulky pod písmeno X. Dále na ose označené Y zjistí souřadnici pro tentýž bod a zaznamená ho do tabulky pod písmeno Y. Následně se vydá po namalované úsečce, až do té doby, dokud nenarazí na další bod. Tento bod zaznamená jako další do připravené tabulky. Takto bude student dále pokračovat, dokud nebude mít všechny body zaznamenané.

Obrazec by neměl mít více jak 24 bodů, jinak by student musel souřadnice psát mimo připravenou tabulku.

Pomocná šablona je přiložena v „Příloha E (šablona)“ a „Příloha F (šablona)“.

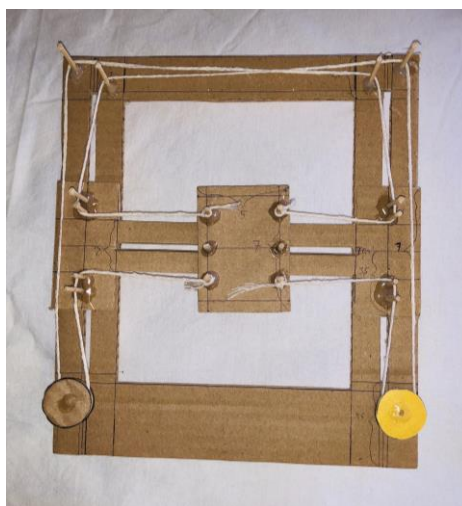


## 6 Vypracování ukázkových řešení jednotlivých úloh

### 6.1 Řešení úlohy konstrukce

#### 6.1.1 Model CoreXY z kartonu

Student má k dispozici vlastnoručně vyrobený funkční model CoreXY dle návodu v kapitole „5.1.1 Model CoreXY z kartonu“.

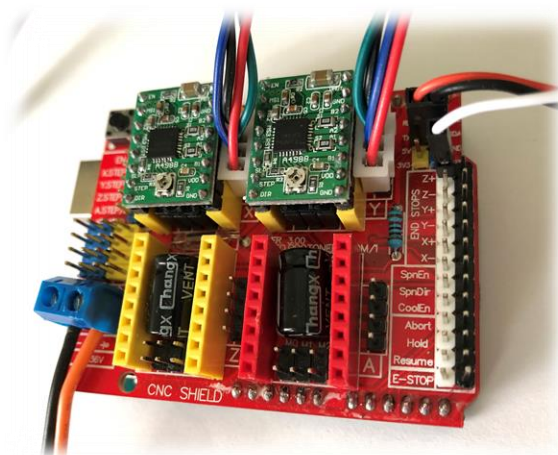


Obrázek 24: Vytvořený model z kartonu robota CoreXY

### 6.2 Řešení úloh elektronika

#### 6.2.1 Zapojení motorů

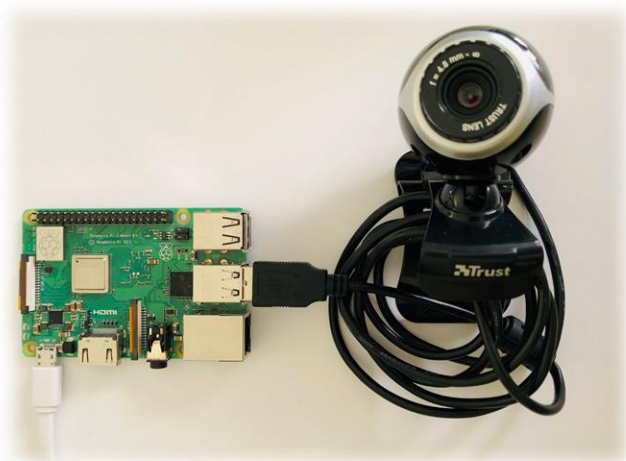
Student má správně zapojené krokové motory a servomotor do Arduina dle návodu v kapitole „5.2.1 Zapojení motorů“.



Obrázek 25: Zapojení motorů do Arduina

## 6.2.2 Zapojení čidel

Student má zapojenou podporovanou kameru do Raspberry Pi jako čidlo dle návodu v kapitole „5.2.2 Zapojení čidel do Raspberry Pi“.

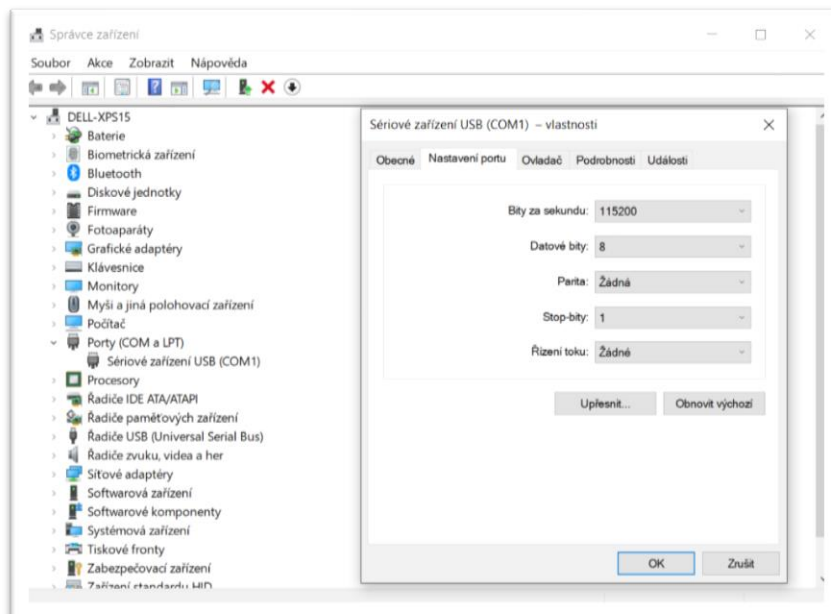


Obrázek 26: Zapojení kamery do Raspberry Pi

## 6.3 Řešení úloh HW

### 6.3.1 Připojení Arduina

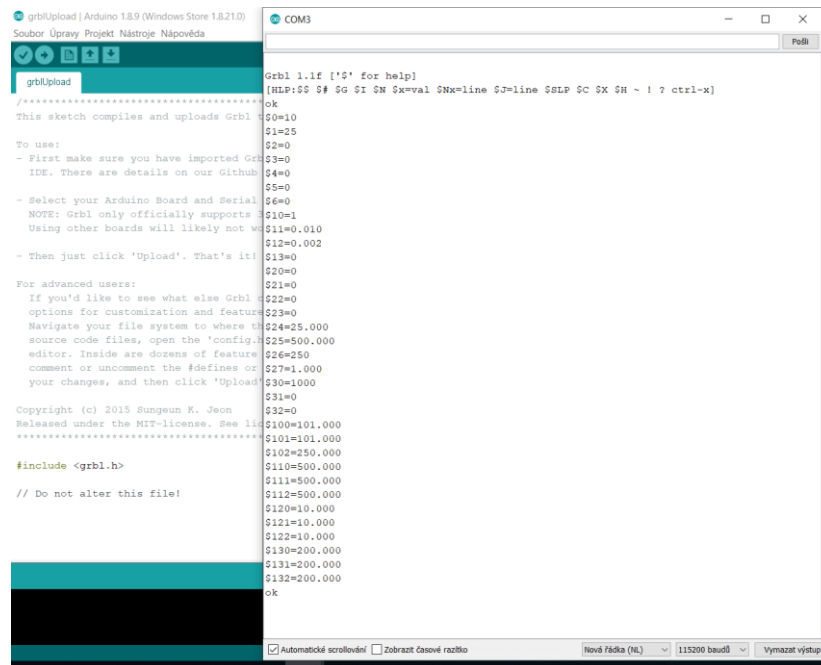
Student má správně připojené a nastavené Arduino k počítači dle návodu v kapitole „5.3.1 Připojení Arduina“.



Obrázek 27: Nastavené připojení Arduina

### 6.3.2 Nahrání GRBL

Student má správně nainstalovaný a zprovozněný systém GRBL dle návodu v kapitole „5.3.2 Nahrání GRBL“.



```
grblUpload | Arduino 1.8.9 (Windows Store 1.8.21.0)
Soubor Úpravy Projekt Nástroje nápověda

grblUpload
This sketch compiles and uploads Grbl to your Arduino board.

To use:
- First make sure you have imported Grbl to your Arduino IDE. There are details on our Github page.
- Select your Arduino Board and Serial Port.
NOTE: Grbl only officially supports Arduino Uno. Using other boards will likely not work.
- Then just click 'Upload'. That's it!

For advanced users:
If you'd like to see what else Grbl can do, check the options for customization and features.
Navigate your file system to where the source code files are, open the 'config.h' file in a text editor. Inside are dozens of features that can be enabled, commented or uncommented. The #defines control your changes, and then click 'Upload'.

Copyright (c) 2015 Sungeun K. Jeon
Released under the MIT-license. See LICENSE for details.

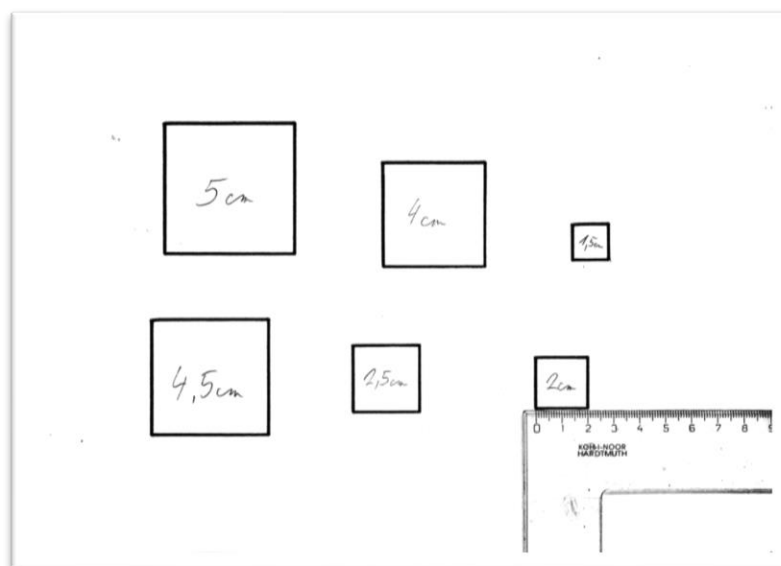
#include <grbl.h>
// Do not alter this file!

Grbl 1.1f ['?' for help]
[HELP:?? $# SG SI SM Sx=val SNx=line GJ=line SSLP SC SX SH ~ ! ? ctrl-x]
ok
G0=10
G1=25
G2=0
G3=0
G4=0
G5=0
G6=0
G10=1
G11=0.010
G12=0.002
G13=0
G20=0
G21=0
G22=0
G23=0
G24=25.000
G25=500.000
G26=250
G27=1.000
G30=1000
G31=0
G32=0
G100=101.000
G101=101.000
G102=250.000
G110=500.000
G111=500.000
G112=500.000
G120=10.000
G121=10.000
G122=10.000
G130=200.000
G131=200.000
G132=200.000
ok
 Automatické scrollování  Zobrazit časové razítko Nová řádka (NL) 115200 baudů Vymazat výstup
```

Obrázek 28: Komunikace Arduina přes Sériový monitor

### 6.3.3 Kalibrace CoreXY

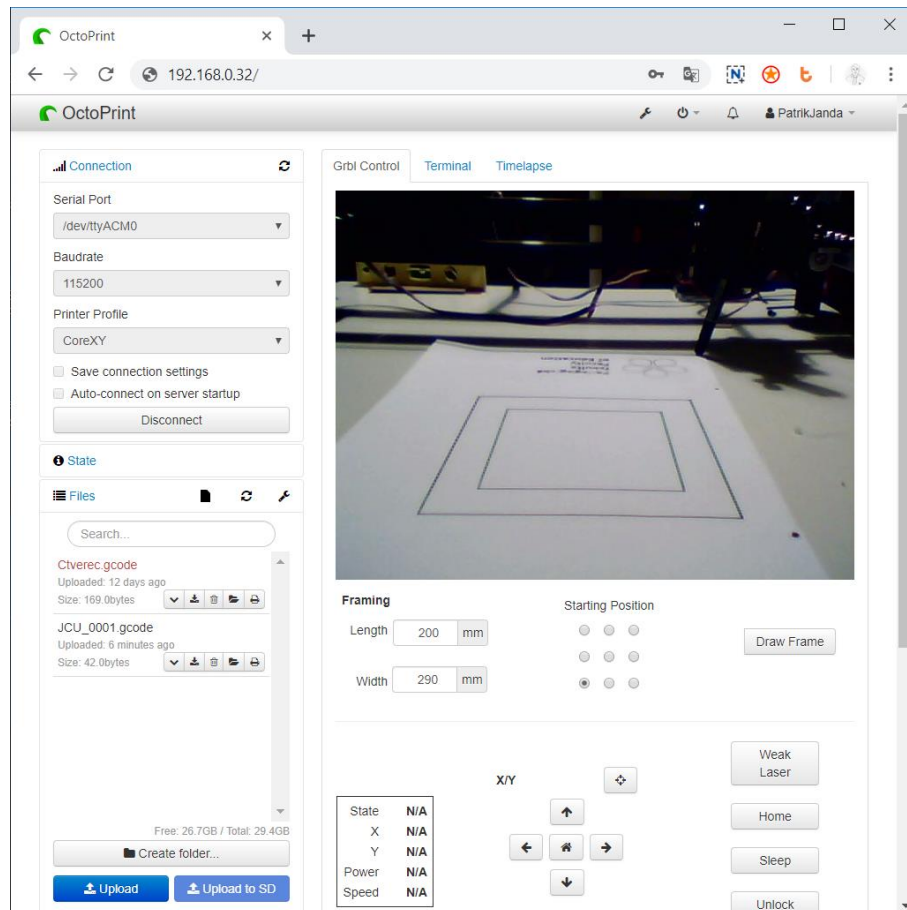
Student má úspěšně zkalibrován systém robota CoreXY s GRBL dle návodu v kapitole „5.3.3 Kalibrace CoreXY“.



Obrázek 29: Čtverce nakreslené při kalibraci

### 6.3.4 Instalace OctoPrint do Raspberry Pi

Student má zprovozněný systém OctoPrint na Raspberry Pi s doinstalovaným pluginem GRBL dle návodu v kapitole „5.3.4 Instalace OctoPrint do Raspberry Pi“.



Obrázek 30: Octoprint s pluginem GRBL

## 6.4 Řešení úloh SW

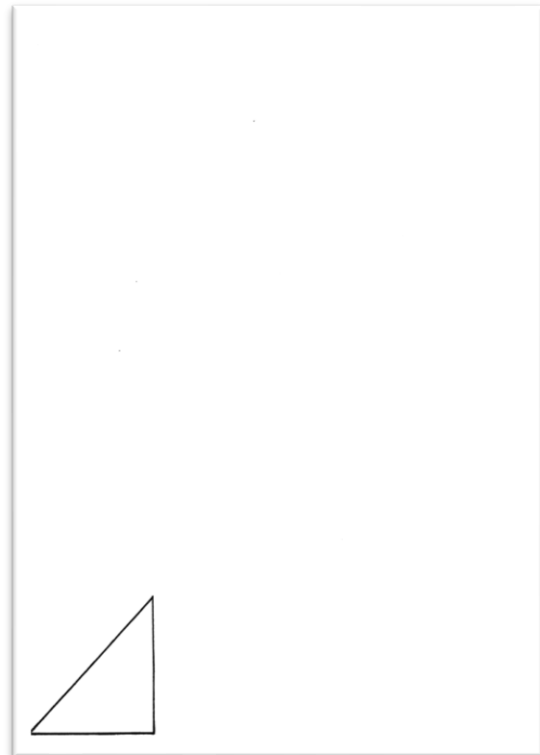
### 6.4.1 Úloha Start

Výsledné řešení úlohy Start z kapitoly „5.4.1 Úloha Start“ je uvedeno níže v „Příklad 3: G-code úlohy Start“.

Výstup z úlohy Start je zobrazen na „Obrázek 31: Nakreslená úloha Start“.

```
G1 F3000  
G21  
G1 X50 Y0  
G1 X0 Y50  
G1 X25 Y25  
M03 S255  
G4 P2  
M05 S0  
G1 X0 Y0
```

*Příklad 3: G-code úlohy start*



*Obrázek 31: Nakreslená úloha start*

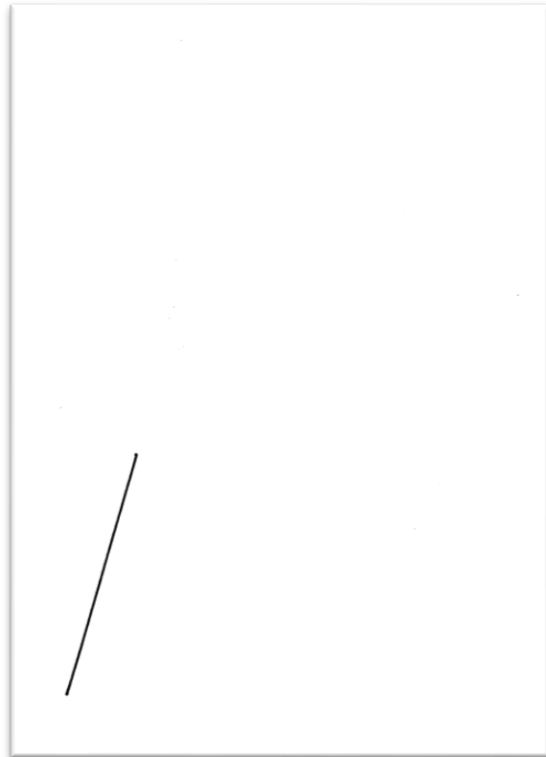
### 6.4.2 Úloha Úsečka

Výsledné řešení úlohy Úsečka z kapitoly „5.4.2 Úloha úsečka“ je uvedeno níže v „Příklad 4: G-code úlohy Úsečka“.

Výstup z úlohy Úsečka je zobrazen na „Obrázek 32: Nakreslená úloha Úsečka“.

```
M03 S255
G90
G21
G1 F3000
G1 X45 Y113
M05 S0
G4 P2
G1 F1000
G1 X17 Y18
M03 S255
G4 P2
G1 F3000
G1 X0 Y0
M05 S0
```

Příklad 4: G-code úlohy Úsečka



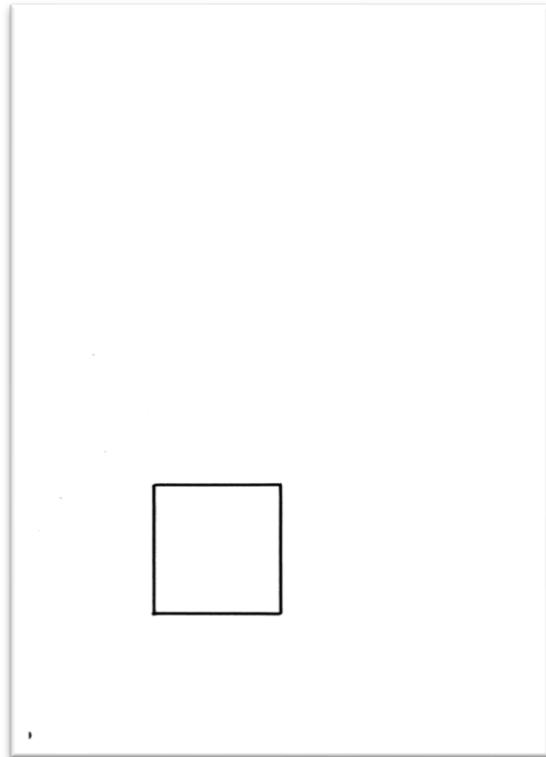
Obrázek 32: Nakreslená úloha Úsečka

### 6.4.3 Úloha Čtverec

Výsledné řešení úlohy Čtverec z kapitoly „5.4.3 Úloha Čtverec“ je uvedeno níže v „Příklad 5: G-code úlohy Čtverec“.

Výstup z úlohy Čtverec je zobrazen na „Obrázek 33: Nakreslená úloha Čtverec“.

```
M03 S255
G90
G21
G1 F3000
G1 X50 Y50
M05 S0
G4 P2
G1 F1000
G1 X50 Y100
G1 X100 Y100
G1 X100 Y50
G1 X50 Y50
M03 S255
G4 P2
G1 F3000
G1 X0 Y0
M05 S0
```



Obrázek 33: Nakreslená úloha Čtverec

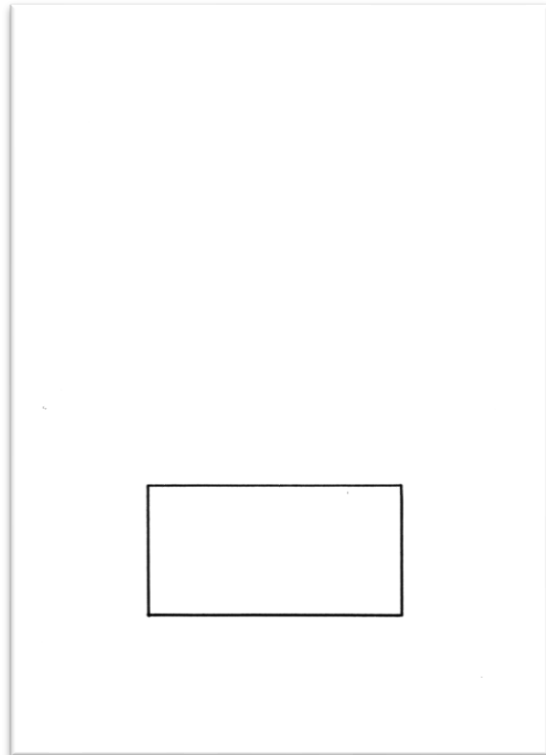
Příklad 5: G-code úlohy Čtverec

#### 6.4.4 Úloha Obdélník

Výsledné řešení úlohy Obdélník z kapitoly „5.4.4 Úloha obdélník“ je uvedeno níže v „Příklad 6: G-code úlohy Obdélník“.

Výstup z úlohy Obdélník je zobrazen na „Obrázek 34: Nakreslená úloha Obdélník“.

```
M03 S255
G90
G21
G1 F3000
G1 X50 Y50
M05 S0
G4 P2
G1 F1000
G1 X50 Y100
G1 X150 Y100
G1 X150 Y50
G1 X50 Y50
M03 S255
G4 P2
G1 F3000
G1 X0 Y0
M05 S0
```



Obrázek 34: Nakreslená úloha Obdélník

Příklad 6: G-code úlohy Obdélník

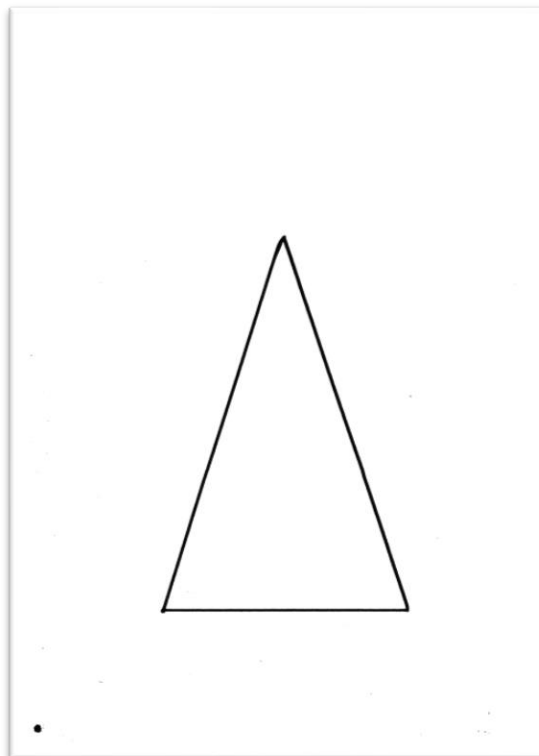


### 6.4.5 Úloha Trojúhelník 1

Výsledné řešení úlohy Trojúhelník 1 z kapitoly „5.4.5 Úloha Trojúhelník 1“ je uvedeno níže v „Příklad 7: G-code úlohy Trojúhelník 1“.

Výstup z úlohy Trojúhelník 1 je zobrazen na „Obrázek 35: Nakreslená úloha Trojúhelník 1“.

```
M03 S255
G90
G21
G1 F3000
G1 X50 Y50
M05 S0
G4 P2
G1 F1000
G1 X150 Y50
G1 X100 Y200
G1 X50 Y50
M03 S255
G4 P2
G1 F3000
G1 X0 Y0
M05 S0
```



Obrázek 35: Nakreslená úloha Trojúhelník 1

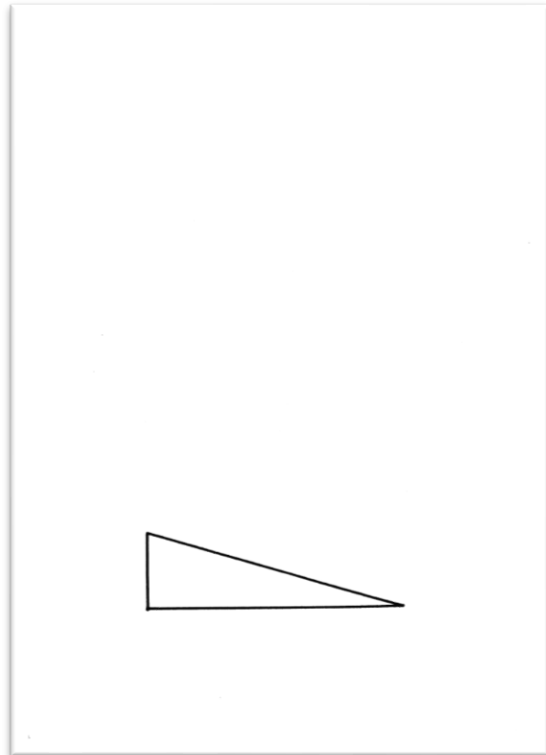
Příklad 7: G-code úlohy Trojúhelník 1

### 6.4.6 Úloha Trojúhelník 2

Výsledné řešení úlohy Trojúhelník 2 z kapitoly „5.4.6 Úloha Trojúhelník 2“ je uvedeno níže v „Příklad 8: G-code úlohy Trojúhelník 2“.

Výstup z úlohy Trojúhelník 2 je zobrazen na „Obrázek 36: Nakreslená úloha Trojúhelník 2“.

```
M03 S255  
G90  
G21  
G1 F3000  
G1 X50 Y50  
M05 S0  
G4 P2  
G1 F1000  
G1 X150 Y50  
G1 X50 Y80  
G1 X50 Y50  
M03 S255  
G4 P2  
G1 F3000  
G1 X0 Y0  
M05 S0
```



Obrázek 36: Nakreslená úloha Trojúhelník 2

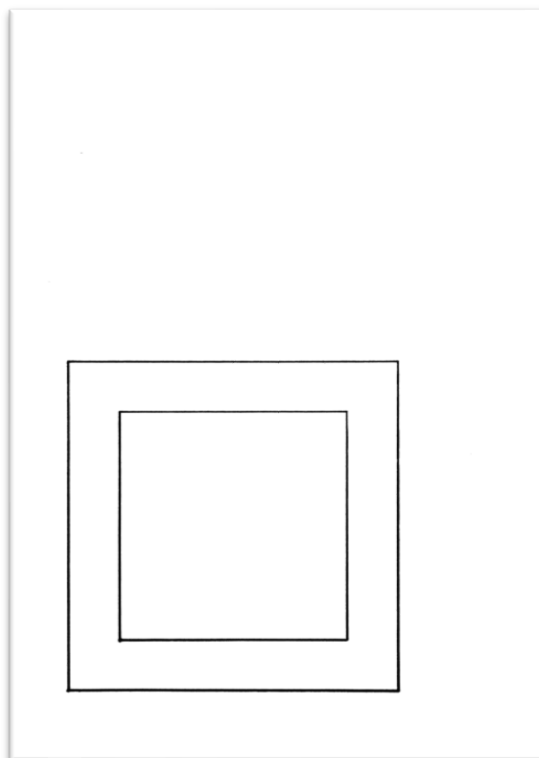
Příklad 8: G-code úlohy Trojúhelník 2

### 6.4.7 Úloha Čtverec ve čtverci

Výsledné řešení úlohy Čtverec ve čtverci z kapitoly „5.4.7 Úloha Čtverec ve čtverci“ je uvedeno níže v „Příklad 9: G-code úlohy Čtverec ve čtverci“.

Výstup z úlohy Čtverec ve čtverci je zobrazen na „Obrázek 37: Nakreslená úloha Čtverec ve čtverci“.

```
M03 S255
G90
G21
G1 F3000
G1 X20 Y20
M05 S0
G4 P2
G1 F1000
G1 X20 Y150
G1 X150 Y150
G1 X150 Y20
G1 X20 Y20
M03 S255
G4 P2
G1 F3000
G1 X40 Y40
M05 S0
G4 P2
G1 F1000
G1 X40 Y130
G1 X130 Y130
G1 X130 Y40
G1 X40 Y40
M03 S255
G4 P2
G1 F3000
G1 X0 Y0
M05 S0
```



Obrázek 37: Nakreslená úloha Čtverec ve čtverci

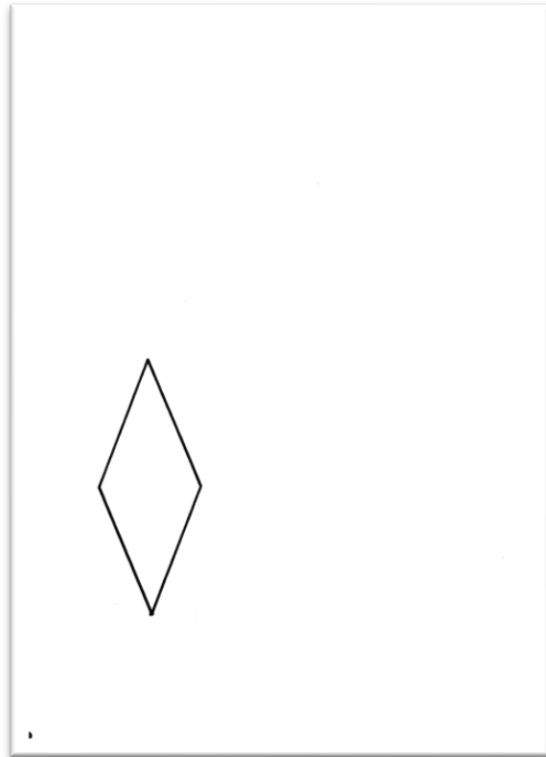
Příklad 9: G-code úlohy Čtverec ve čtverci

### 6.4.8 Úloha Kosočtverec

Výsledné řešení úlohy Kosočtverec z kapitoly „5.4.8 Úloha Kosočtverec“ je uvedeno níže v „Příklad 10: G-code úlohy Kosočtverec“.

Výstup z úlohy Kosočtverec je zobrazen na „Obrázek 38: Nakreslená úloha Kosočtverec“.

```
M03 S255
G90
G21
G1 F3000
G1 X50 Y50
M05 S0
G4 P2
G1 F1000
G1 X30 Y100
G1 X50 Y150
G1 X70 Y100
G1 X50 Y50
M03 S255
G4 P2
G1 F3000
G1 X0 Y0
M05 S0
```



Obrázek 38: Nakreslená úloha Kosočtverec

Příklad 10: G-code úlohy Kosočtverec

## 7 Zhodnocení a závěr

Ve své práci jsem se snažil zúročit všechny mé teoretické i praktické znalosti. Těch jsem musel využít zejména při sestavování a zprovoznění robota CoreXY, abych dosáhl stanoveného cíle.

Předpokládám, že cíl mé práce – navrnutí a vytvoření modelových úloh pro využití ve výuce v zájmových kroužcích na robotovi CoreXY se mi podařilo splnit. Vyrobil jsem několik modelových úloh, které mohou být vzorem pro výuku programování v těchto kroužcích.

Pro praktické ověření těchto úloh jsem musel nejdříve sestavit a zprovoznit kompletního robota CoreXY, což se mi podařilo. Tento robot může sloužit při výuce programování v některém z kroužků.

Modelové úlohy jsem navrhl a ověřil přímo na robotovi CoreXY. Úlohy jsou využitelné pro studenty se základy programování a elektroniky.

V současné době funguje robot CoreXY pouze pro kreslení. V dalším vývojové etapě bych ho rád doplnil o laser a o osu Z a tím získal možnost frézování. Tím by se rozšířila i možnost výuky. Pro tyto účely bych vytvořil pro výuku programování další modelové i zábavné úlohy.

## Seznam použité literatury a zdrojů

- [1] RAJZL, Lukáš. *Paralelní robot CoreXY [online]*. České Budějovice, 2017 [cit. 2019-07-08]. Dostupné z: <http://www.nusl.cz/ntk/nusl-263276>. *Bakalářská práce (Bc.)*. JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH. Vedoucí práce Ing. Michal Šerý, PhD.
- [2] PELIKÁN, Filip. *Pracuje robot s člověkem nebo člověk s robotem?*. *VoltWorld [online]*. 2018, 26. 10. 2018 [cit. 2019-07-11]. Dostupné z: <http://www.elektroprumysl.cz/automatizace/pracuje-robot-s-clovekem-nebo-clovek-s-robotem>
- [3] WIKIPEDIA CONTRIBUTORS. *G-code*. In: *Wikipedia: the free encyclopedia [online]*. San Francisco (CA): Wikimedia Foundation, 2001-, 5 July 2019 15:33 UTC [cit. 2019-07-11]. Dostupné z: <https://en.wikipedia.org/wiki/G-code>
- [4] HÄUßGE, Gina. *OctoPrint*. *OctoPrint [online]*. 2019 [cit. 2019-07-11]. Dostupné z: <https://octoprint.org/>
- [5] UPTON, Eben a Gareth HALFACREE. *Raspberry Pi: uživatelská příručka*. Brno: Computer Press, 2013. ISBN 978-80-251-4116-8.
- [6] *Wikipedie Příspěvatelé, Raspberry Pi [online]*, *Wikipedie: Otevřená encyklopedie, c2019, Datum poslední revize 29. 06. 2019 [cit. 2019-07-11]* Dostupný z WWW:  
<[https://cs.wikipedia.org/w/index.php?title=Raspberry\\_Pi&oldid=17442385](https://cs.wikipedia.org/w/index.php?title=Raspberry_Pi&oldid=17442385)>
- [7] *Arduino: What is Arduino? [online]*. *Arduino - Introduction, 2019 [cit. 2019-07-11]*. Dostupné z: <https://www.arduino.cc/en/Guide/Introduction>

- [8] *LIER, Maarten, WHY THE CORE3D PRINTER USES COREXY. [Online], CORE3D CUSTOM PRINTERS, 2017[cit. 2019-07-11]. Dostupný z WWW: <https://core3d.tech/2017/06/10/why-core3d-printer-uses-corexy/>*
- [9] *GNEA. Grbl [online]. 2018 [cit. 2019-07-11]. Dostupné z: <https://github.com/gnea/grbl>*
- [10] *How to make GRBL+arduino based CNC plotter PCB ink plotter, in: Youtube [online].2017, 14. 11. 2017 [cit. 2019-07-11]. Dostupné z: <https://youtu.be/klvzT-T8LJA>. Kanál uživatele Mr Innovative.*
- [11] *JeDe Robot s.r.o, GRBL PRO ARDUINO A GENUINO. [Online], PROFIROBOT, 2019 [cit. 2019-07-11]. Dostupný z WWW: <http://profirobot.cz/grbl-pro-arduino-a-genuino/>*

## Seznam obrázků

Obrázek 1: Princip CoreXY .....	18
Obrázek 2: Model konstrukce DrawBot.....	19
Obrázek 3: Konstrukce CoreXY .....	20
Obrázek 4: Součástky robota CoreXY .....	20
Obrázek 5: Nesestavený rám .....	22
Obrázek 6: Sestavený rám .....	22
Obrázek 7: Nesestavené krokové motory .....	22
Obrázek 8: Uchycené krokové motory.....	22
Obrázek 9: Vodící tyče s lineárními ložisky .....	22
Obrázek 10: Namontované vodící tyče .....	22
Obrázek 11: Pohyblivé rameno .....	23
Obrázek 12: Připevněné rameno ke konstrukci.....	23
Obrázek 13: Platforma s úchytem pro pero .....	23
Obrázek 14: Natažený ozubený řemen.....	23
Obrázek 15: Servomotor s mechanismem.....	23
Obrázek 16: Připevněný servomotor s mechanismem .....	23
Obrázek 17: Držák na pero.....	24
Obrázek 18: Připevněný držák na pero .....	24
Obrázek 19: Vyvedení kabelů .....	24
Obrázek 20: Sestavený robot CoreXY .....	24
Obrázek 21: Arduino Uno .....	25
Obrázek 22: Raspberry Pi.....	25
Obrázek 23: Ukázka šablony pro model CoreXY .....	27
Obrázek 24: Vytvořený model z kartonu robota CoreXY .....	57



Obrázek 25: Zapojení motorů do Arduina .....	57
Obrázek 26: Zapojení kamery do Raspberry Pi .....	58
Obrázek 27: Nastavené připojení Arduina .....	58
Obrázek 28: Komunikace Arduina přes Sériový monitor .....	59
Obrázek 29: Čtverce nakreslené při kalibraci .....	59
Obrázek 30: Octoprint s pluginem GRBL.....	60
Obrázek 31: Nakreslená úloha start.....	61
Obrázek 32: Nakreslená úloha Úsečka.....	62
Obrázek 33: Nakreslená úloha Čtverec .....	63
Obrázek 34: Nakreslená úloha Obdélník.....	64
Obrázek 35: Nakreslená úloha Trojúhelník 1.....	65
Obrázek 36: Nakreslená úloha Trojúhelník 2.....	66
Obrázek 37: Nakreslená úloha Čtverec ve čtverci .....	67
Obrázek 38: Nakreslená úloha Kosočtverec.....	68

Jsem autorem všech obrázků uvedených v této práci.

## Seznam tabulek

Tabulka 1: Historie modelů Raspberry Pi [6].....	15
Tabulka 2: Výhody Arduina [7] .....	17
Tabulka 3: Výhody CoreXY [8].....	17
Tabulka 4: G-code podporované v GRBL [11].....	79
Tabulka 5: M-code podporované v GRBL [11] .....	80

## Seznam příkladů

Příklad 1: Vzorec výpočtu souřadnic CoreXY [8] .....	18
Příklad 2: G-code Testovací čtverec.....	32
Příklad 3: G-code úlohy start.....	61
Příklad 4: G-code úlohy Úsečka.....	62
Příklad 5: G-code úlohy Čtverec .....	63
Příklad 6: G-code úlohy Obdélník.....	64
Příklad 7: G-code úlohy Trojúhelník 1.....	65
Příklad 8: G-code úlohy Trojúhelník 2.....	66
Příklad 9: G-code úlohy Čtverec ve čtverci .....	67
Příklad 10: G-code úlohy Kosočtverec .....	68

## **Přílohy**

- A. DVD
- B. Tabulka G-code
- C. Tabulka M-code
- D. Šablony pro model CoreXY
- E. Pomocná šablona pro určení souřadnic X a Y (v1)
- F. Pomocná šablona pro určení souřadnic X a Y (v2)

## **Příloha A (DVD)**

### DVD

Na disku se nachází digitální verze mé bakalářské práce „Použití paralelního robota CoreXY pro výuku v zájmových kroužcích“. Dále se na DVD nachází šablona pro kartonový model CoreXY a pomocné šablony pro určení souřadnic X a Y. DVD obsahuje zálohu SW, který byl použit při psaní mé bakalářské práce.

## **Příloha B (tabulka)**

Tabulka s podporovanými G-code v systému GRBL.

<b>Kódy</b>	<b>Funkce</b>
G00	Rychlý přesun
G01	Lineární interpolace
G02	Kruhová interpolace ve směru hodinových ručiček
G03	Kruhová interpolace proti směru hodinových ručiček
G04	Pauza
G10	Změna offsetu souřadného systému
G17	Volba pracovní roviny X-Y
G18	Volba pracovní roviny X-Z
G19	Volba pracovní roviny Y-Z
G20	Nastavení jednotek souřadného systému na palce (inch)
G21	Nastavení jednotek souřadného systému na milimetry
G28	Nájezd do referenční polohy
G28.1	Nastavení souřadnic referenční polohy
G30	Nájezd do další referenční polohy
G30.1	Nastavení souřadnic další referenční polohy
G38.2 až G38.5	sonda
G40	Kompenzace poloměru nástroje
G43.1	Dočasná změna délkového offsetu nástroje
G49	Vypnutí offsetu nástrojů
G53	Nastavení souřadného systému stroje na absolutní pozicování
G54 až G59	Volba souřadného systému obrobku
G61	Režim řízení pohybu
G80	Zrušení polohování včetně pevných cyklů
G90	Absolutní režim odměřování
G91	Inkrementální (přírůstkový) režim odměřování
G91.1	Parametry IJK definují polohu středu oblouku relativně k počátečnímu bodu.
G92	Nastavení lokálního počátku souřadného systému (od aktuální pozice nástroje)
G92.1	Zrušení lokálního počátku souřadného systému
G93	Posuv v jednotkách času
G94	Posuv v délkových jednotkách

Tabulka 4: G-code podporované v GRBL [11]

## Příloha C (tabulka)

Tabulka s podporovanými M-code v systému GRBL.

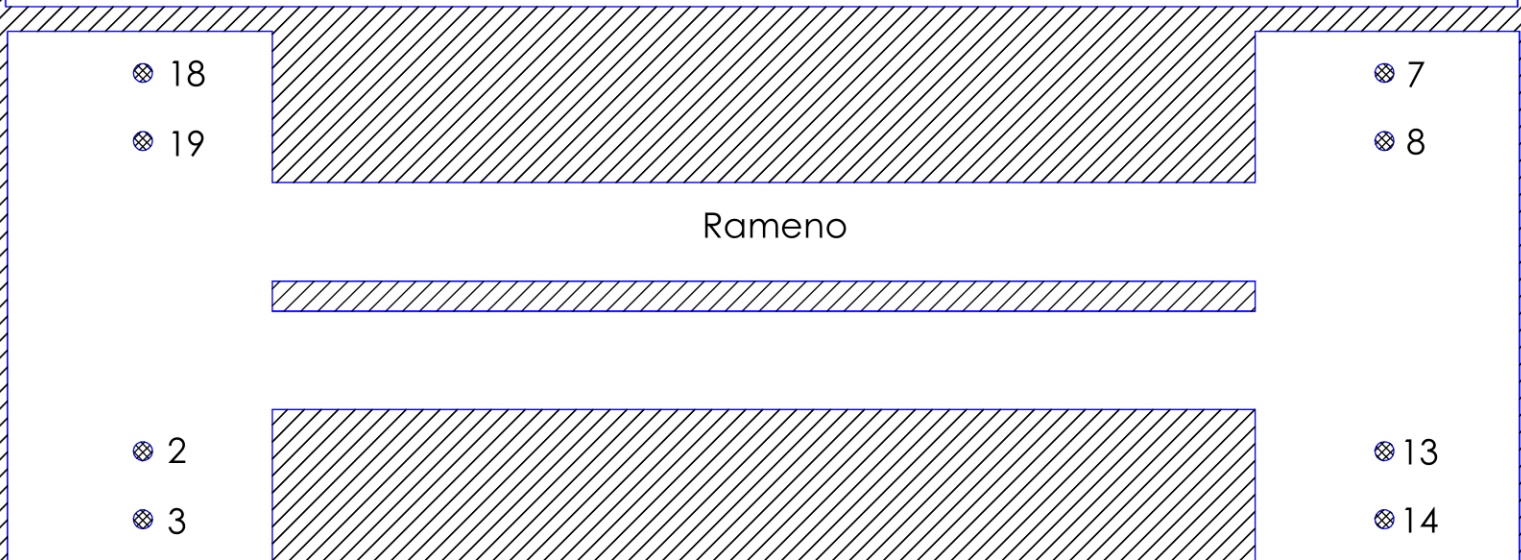
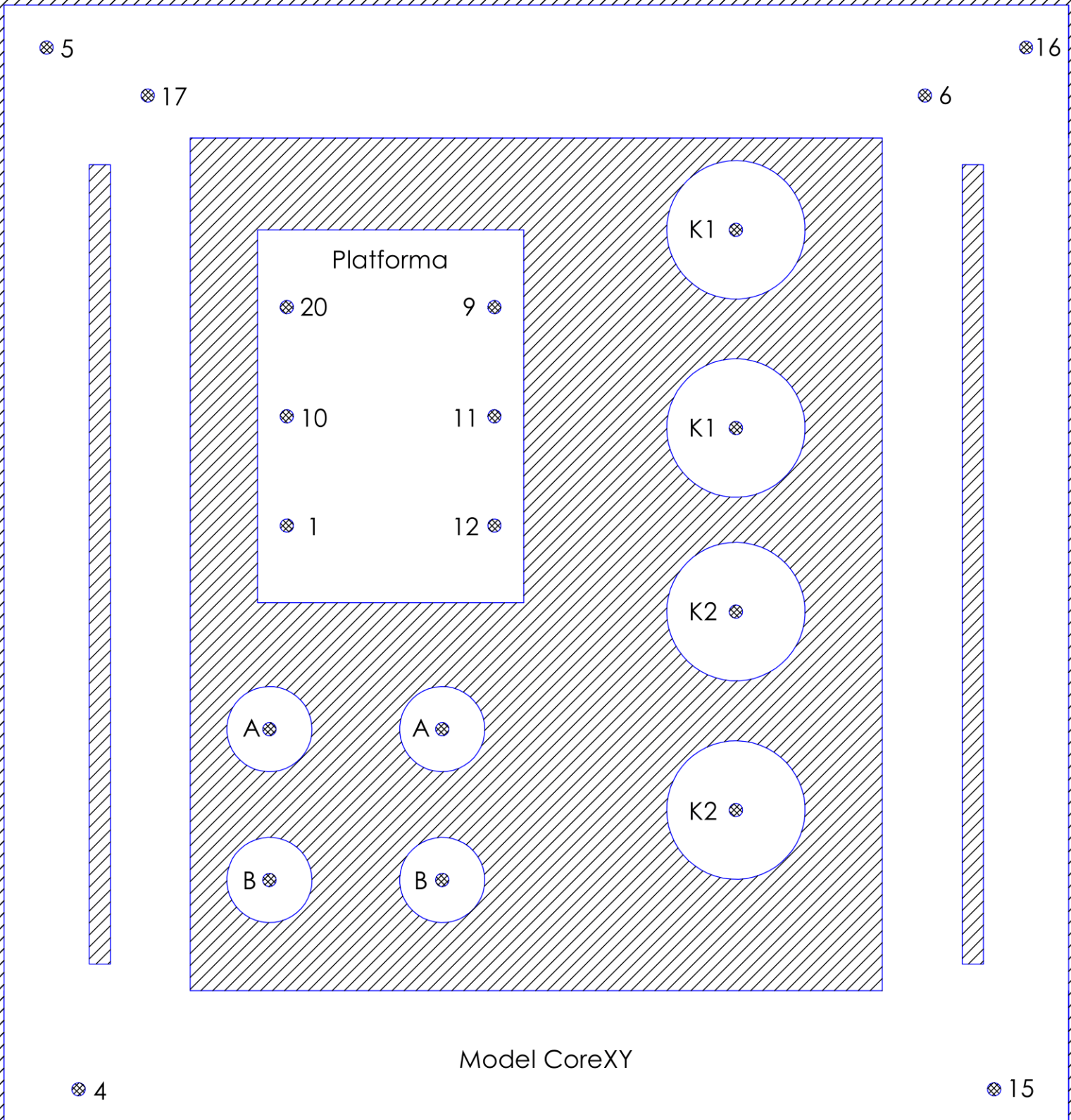
<b>Kódy</b>	<b>Funkce</b>
M0	zastavení
M2	konec programu
M3	pravé otáčky vřetena
M4	levé otáčky vřetena
M5	stop vřetena
M8	spuštění chlazení
M9	zastavení chlazení
M30	konec programu (reset)

*Tabulka 5: M-code podporované v GRBL [11]*



## **Příloha D (šablona)**

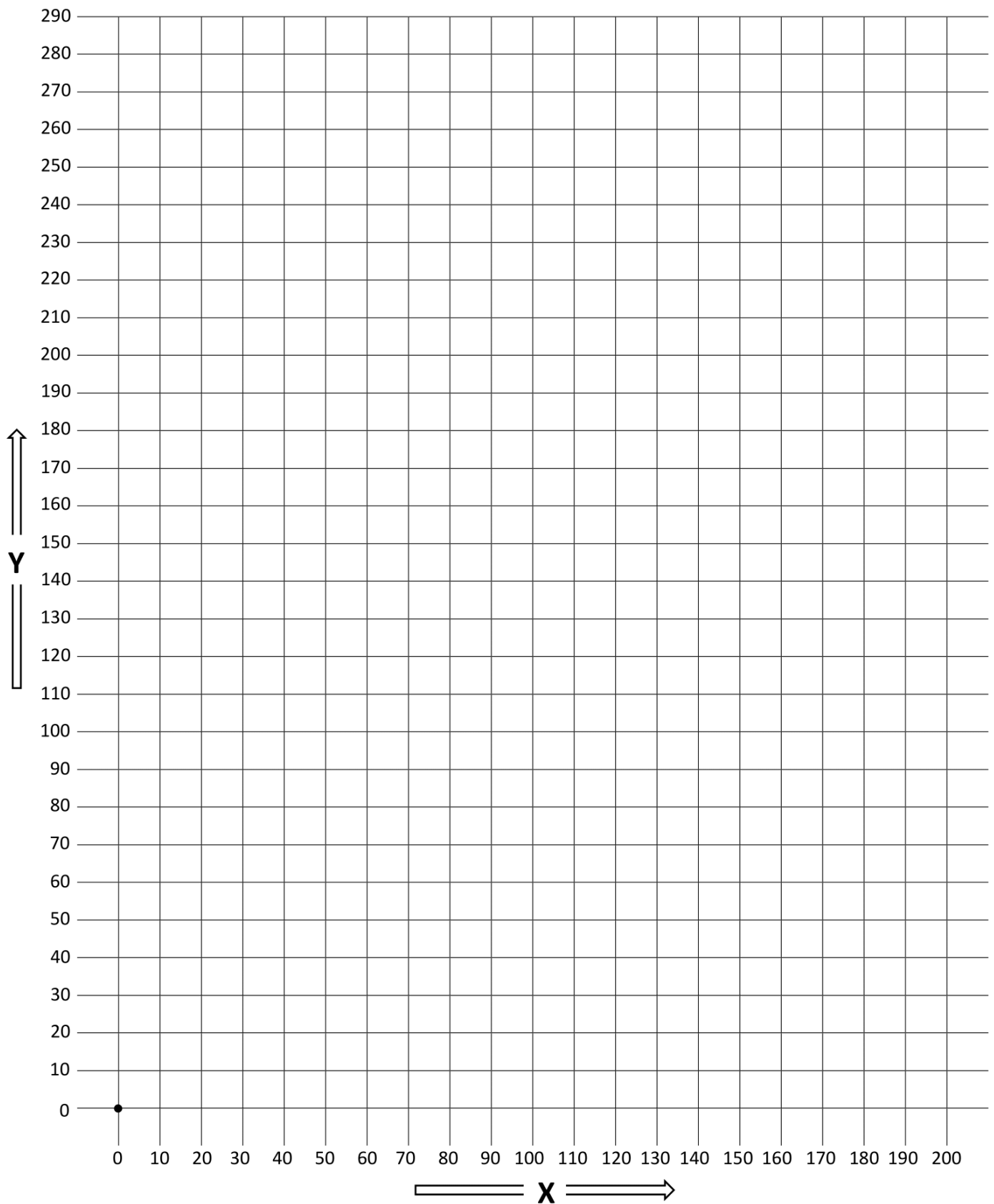
Šablona pro model CoreXY



## **Příloha E (šablona)**

Pomocná šablona pro určení souřadnic X a Y (v1)

# Pomocná šablona pro určení souřadnic X a Y

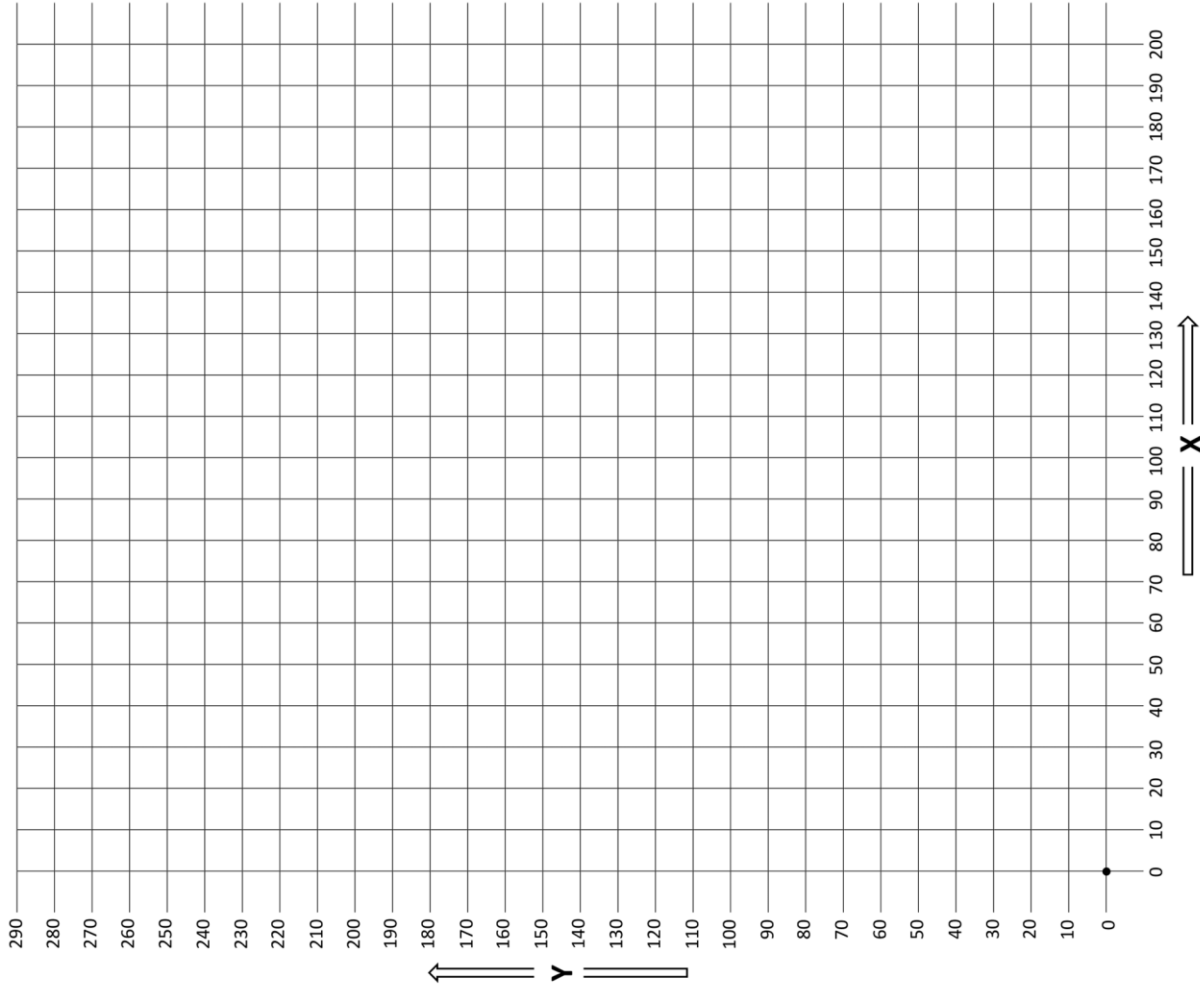


1) X..... Y.....	7) X..... Y.....	13) X..... Y.....	19) X..... Y.....
2) X..... Y.....	8) X..... Y.....	14) X..... Y.....	20) X..... Y.....
3) X..... Y.....	9) X..... Y.....	15) X..... Y.....	21) X..... Y.....
4) X..... Y.....	10) X..... Y.....	16) X..... Y.....	22) X..... Y.....
5) X..... Y.....	11) X..... Y.....	17) X..... Y.....	23) X..... Y.....
6) X..... Y.....	12) X..... Y.....	18) X..... Y.....	24) X..... Y.....

## **Příloha F (šablona)**

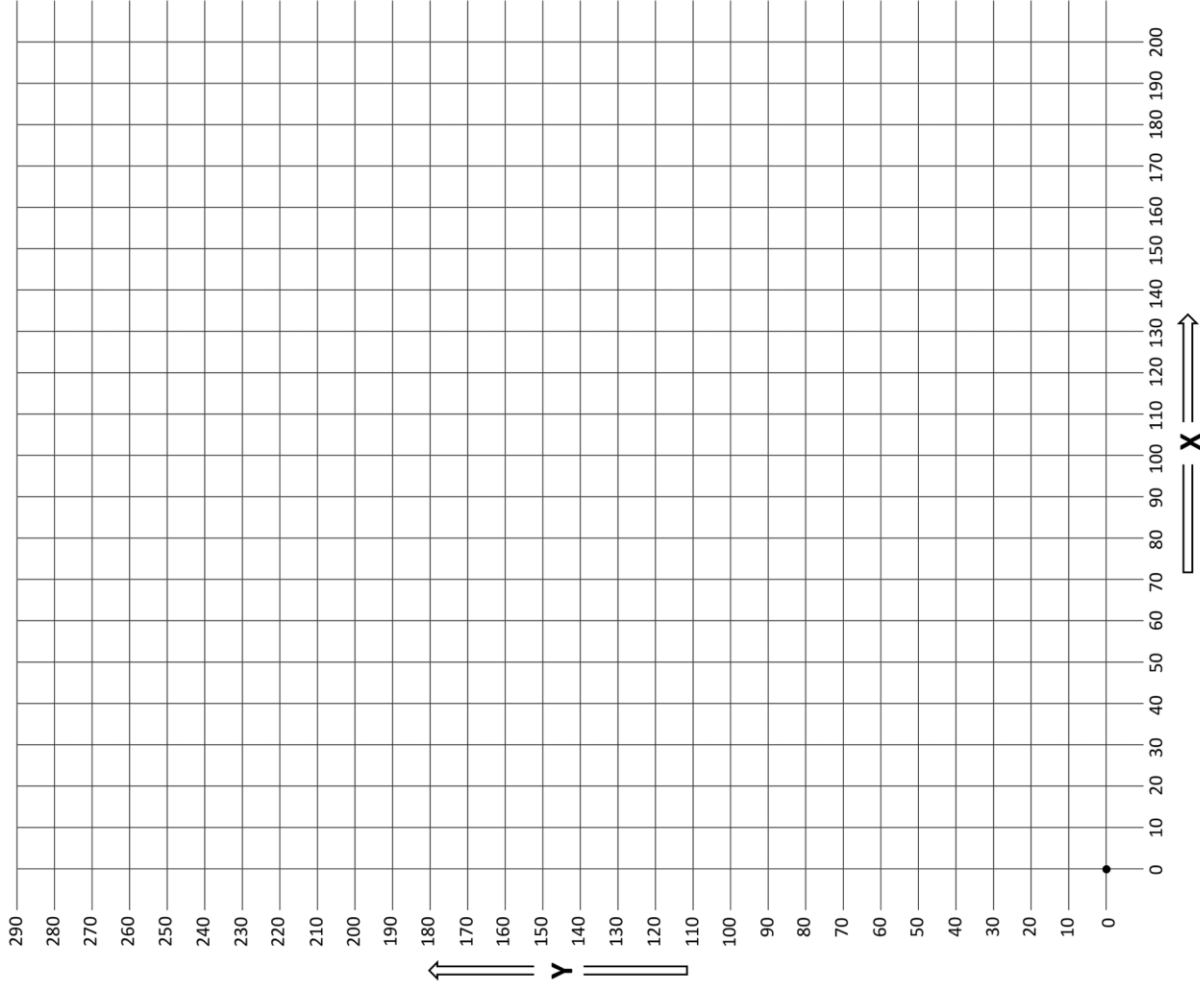
Pomocná šablona pro určení souřadnic X a Y (v2)

### Pomocná šablona pro určení souřadnic X a Y



1) X..... Y.....	7) X..... Y.....	13) X..... Y.....	19) X..... Y.....
2) X..... Y.....	8) X..... Y.....	14) X..... Y.....	20) X..... Y.....
3) X..... Y.....	9) X..... Y.....	15) X..... Y.....	21) X..... Y.....
4) X..... Y.....	10) X..... Y.....	16) X..... Y.....	22) X..... Y.....
5) X..... Y.....	11) X..... Y.....	17) X..... Y.....	23) X..... Y.....
6) X..... Y.....	12) X..... Y.....	18) X..... Y.....	24) X..... Y.....

### Pomocná šablona pro určení souřadnic X a Y



1) X..... Y.....	7) X..... Y.....	13) X..... Y.....	19) X..... Y.....
2) X..... Y.....	8) X..... Y.....	14) X..... Y.....	20) X..... Y.....
3) X..... Y.....	9) X..... Y.....	15) X..... Y.....	21) X..... Y.....
4) X..... Y.....	10) X..... Y.....	16) X..... Y.....	22) X..... Y.....
5) X..... Y.....	11) X..... Y.....	17) X..... Y.....	23) X..... Y.....
6) X..... Y.....	12) X..... Y.....	18) X..... Y.....	24) X..... Y.....