



SOŠ STROJNÍCKA
Partizánska cesta 76
957 01 Bánovce nad Bebravou

Robotické rameno

Ročníkový projekt

Bánovce nad Bebravou

2021

Samuel Adamus

Ročník štúdia: tretí



SOŠ STROJNÍCKA
Partizánska cesta 76
957 01 Bánovce nad Bebravou

Robotické rameno

Ročníkový projekt

Riešiteľ: Samuel Adamus
Študijný odbor: Technické lýceum
Katalógové číslo: 3918 M
Ročník štúdia: tretí
Konzultant: Ing. Zuzana Grešová, PhD.

Bánovce nad Bebravou

2024

Čestné vyhlásenie

Vyhlasujem, že som ročníkovú prácu vypracoval samostatne a použil som len odbornú literatúru, ktorú uvádzam v zozname použitej literatúry. Moja dokumentácia neobsahuje chránené údaje podniku alebo zákazníka a neporušuje autorské práva.

V Bánovciach nad Bebravou, dňa _____

Vlastnoručný podpis

Obsah

1. Úvod.....	5
2. Problematika a prehľad literatúry.....	7
3. Ciele práce.....	10
4. Robotické rameno	11
4.1 Modelovanie a tlač.....	11
4.2 Skladanie	11
4.3 Programovanie.....	12
5. Závěry práce a zhrnutie	14
6. Zoznam použitej literatúry	15
7. Prílohy	16

1. Úvod

Slovo „robot“ bolo prvý krát použité v roku 1920 v diele R.U.R., ktoré napísal Karel Čapek. Odvtedy ubehlo už viac ako 100 rokov, a teraz sa to slovo používa viac než kedykoľvek predtým. Roboty ovládli priemyselný priemysel a mnoho ďalších. Ich hlavnou úlohou je spríjemniť život ľuďom. Či už nosením ťažkých želez z nebezpečných prostredíach alebo vysávaním bytu podľa senzorov a čidiel. Celý život ma zaujímali roboty ale vždy som si myslel, že to je moc komplikované pre jednotlivca aby to dokázal spraviť. Koncom druhého ročníka som sa začal zaujímať o elektriku a elektrické komponenty, tak som si objednal elektrický krokový motor spolu s nijakým microcontrollerom a driverom. Zvolil som arduino lebo som sa predtým už s tým stretol a je to asi najznámejšia voľba pre malé projekty. Keď to prišlo, mal som problém to všetko zapojiť aby to fungovalo. Dlhú som rozmýšľal jak vôbec spravím aby sa motor začal točiť v jednom smere. Keď sa mi to konečne podarilo tak som začal experimentovať aby som mohol meniť otáčky a smer podľa predtým napísaného programu v C++. Potom čo som aj toto rozbehal som chcel zakomponovať menenie týchto hodnôt v reálnom čase cez python. To som docielil cez sériovú komunikáciu medzi počítačom a arduinom. Začal som uvažovať, že aký väčší projekt by som s týmito schopnosťami vedel docieľiť. Mal som doma stavebnicu, dlho zabudnutú, čo je vlastne robotické rameno ovládané cez ovládač podobný rc autíčkam akurát cez kábel. Rameno som rozobral, ponapájal servá na arduino a snažil sa to ovládať cez program. Nakoniec sa mi to aj čiastočne podarilo, jediná chyba čo tam bola, že kvôli kúpeniu lacných driverov iba niektoré nefungovali, tak som mohol ovládať iba niektoré servá. Pri tomto pokuse som zistil, že robotické ramená nie sú až také komplikované ako sa zdajú. V ten deň som sa rozhodol spraviť robotické rameno, kde všetky diely budú vytlačené na 3D tlačiarňi, motory použité budú krokové a ovládanie bude cez arduino, ktoré dostáva pokyny od počítaču. Doma som mal zakúpenú tlačiareň značky Prusa, na ktorej som dovtedy nerobil žiadne extra veľké veci. Ako materiál som použil PETG, kvôli pevnosti materiálu a jednoduchom zaobchádzaní pri tlačení. Krokové motory som vybral kvôli pomeru ceny a spoľahlivosti. Arduino bola voľba pri tomto ramene lebo som sa chcel do hĺbky naučiť pracovať s týmto programovateľným microcontrollerom. Na modelovanie som použil Blender, lebo som sa už predtým s programom stretol. Rameno som konštruoval celé letné prázdniny popri brigáde v automobilovom priemysle kde je vidieť jak moc sú potrební roboti pre prácu. Šetria peniaze zamestnávateľovi a zamestnancov šetria od namáhavých prác. Ďalší problém bol aké silné motory kúpiť. Rokovanie s fyzikálnymi zákonmi mi tiež pomohlo pri výbere vhodných motorov, ktoré sa ukázali ako kľúčové pre úspech môjho projektu. Všetky motory majú rovnakú silu akurát niektoré majú prevodovku

kde zamieňajú rýchlosť za silu v pomere v akom je prevodovka. Po mnohých neúspechoch pri tlači, najmä kvôli zlému meraniu alebo chybe pri modelovaní, som si uvedomil, že tvorba robotov vyžaduje trpezlivosť a systematický prístup k riešeniu problémov. Čím viac som robota dokončoval tým viac problémov nastávalo. Boli problémy napríklad s prevažovaním robota pri veľkom natočení ramena alebo ako spraviť spoľahlivý koncový afektor. Prevažovanie bolo vyriešené platničkami vyrezanými na cnc laseri a koncový afektor som vyriešil cez remeň a ozubené koleso. Týmto bol hardware vyriešený ostával už len software, ktorý bol pri niektorých častiach komplikovanejší ako samotný robot. Finálny program som vymyslel aby posielal arduinu zoznam inštrukcií pre každý motor, arduino následne vykoná inštrukcie tým, že si vypočíta rozdiel medzi hodnotou kde ma ísť a kde je. Táto hodnota je potom počet, koľkokrát sa pošle na driver motoru informácia aby sa pohol do nijakého smeru. A týmto sa zhotovil celý robot. Výsledkom môjho úsilia a vytrvalosti bol funkčný robot, ktorý som s hrdosťou uvítal ako symbol svojho technického rozvoja a tvorivosti. Tento projekt mi otvoril dvere do sveta robotiky a automatizácie, kde vidím nekonečné možnosti pre ďalší rozvoj a inovácie.

2. Problematika a prehľad literatúry

Pri skúmaní fyzikálnych vzorcov a zákonov pre výpočet potrebnej sily motorov som využíval hlavne wikipédiu a odborné dokumenty. Pre program som využíval rôzne fóra kde ľudia riešili podobné problémy ako som mal ja. Kód som si neskôr upravil aby vyhovoval mojim problémom. Taktiež som sledoval náučné videá o programovaní ako aj stránky určené na učenie sa programovacích jazykov. Na navrhovanie častí robota som sa inšpiroval rôznymi inakšími ramenami na internete alebo aké som videl na mojej brigáde.

2.1 Trojrozmerné modelovanie

Trojrozmerné modelovanie alebo 3D modelovanie umožňuje analýzy priestorových vzťahov 3D objektov. Každý element 3D objektu umožňuje pripojenie negrafických atribútov. Na základe požiadavky na zameranie detailov a analytického spracovania dát je možné ponúknuť tri základné typy 3D modelov budov:

- 1. generácia: Blokový model

Jednotlivé budovy sú nahradené blokom, ktorý je daný priemetom obvodu ich strešných plášťov stavieb na terén a charakteristickou výškou. Bloky budov sú umiestnené na zameraný digitálny model terénu.

- 2. generácia: Urbanistický model

Jednotlivé budovy sú nahradené blokom, ktorý je daný priemetom obvodu ich strešných plášťov na terén, výškovou kostrou a je doplnený o automatizované generované tvary striech z meraných výškových a polohových údajov. Bloky budov sú umiestnené na zameraný digitálny model terénu a môžu byť doplnené jednoduchým priestorovým modelom vegetácie.

- 3. generácia: Podrobný model

3D model blokov budov je daný priemetom obvodu ich strešných plášťov na terén a rozčlenením na jednotlivé budovy podľa stavebných parciel. Tvary striech sú vytvárané 3D modelovaním z meraných výškových a polohových údajov. Budovy sú umiestnené na zameraný digitálny model terénu a môžu byť doplnené priestorovým modelom vegetácie.

2.2 Blender

Blender je open source program, ktorý je zameraný na modelovanie a vykresľovanie 3D počítačovej grafiky, animácií a filmov s využitím rôznych techník. Program Blender bol vytvorený v C, C++ a Python programovacích jazykoch. Blender bol vyvinutý Tonom Roosendaalom, ktorý potom za pomociu Blendera a jeho tímu vytvoril film Elephants dream ktorý bol vydaný v máji 2005. Blender bol prvý krát vydaný 13. októbra 2002 pod licenciou GNU GPL.

Blender ponúka pomocnú ruku pre normálneho človeka do 3D scény. To že je Blender zadarmo umožňuje ľuďom stretnúť veľa ďalších, ktorí sa tak tiež rozhodli naučiť sa 3D grafiku. Blender má obrovskú komunitu rôzneho typu. (<https://sk.wikipedia.org/wiki/Blender>)

2.3 Arduino

Arduino je open-source platforma, založená pôvodne na mikrokontroléri ATmega od firmy Atmel a grafickom vývojovom prostredí, ktoré vychádza z prostredia Wiring a Processing. Arduino môže byť použité k vytváraniu samostatných interaktívnych zapojení alebo môže byť pripojené k softvéru na počítači. Momentálne možno kúpiť verzie, ktoré sú už skompletované; schéma a návrh plošného spoja je dostupná pre tých, ktorí si chcú postaviť Arduino sami.

Projekt Arduino získal ocenenie v kategórii digitálnych komunit na Prix Ars Electronica 2006.

(<https://sk.wikipedia.org/wiki/Arduino>)

2.4 Python

Python je interpretovaný, interaktívny programovací jazyk, ktorý vytvoril Guido van Rossum, pôvodne ako skriptovací jazyk pre Amoeba OS schopný systémových volaní. Python je často porovnávaný s jazykmi Tcl, Perl, Scheme, Java a Ruby. Python je vyvíjaný ako open source projekt, a je v súčasnosti pri verzii 3.12. Python je multi-paradigmaticý jazyk podobne ako Perl, na rozdiel od Smalltalku alebo Haskellu. To znamená, že namiesto toho aby nútil programátora používať určitý štýl programovania, umožňuje používanie viacerých. Python podporuje objektovo orientované, štruktúrované aj funkcionálne programovanie. Je to dynamicky typový jazyk, podporuje veľké množstvo vysokoúrovňových dátových typov a na správu pamäte používa garbage collection.

([https://sk.wikipedia.org/wiki/Python_\(programovac%C3%AD_jazyk\)](https://sk.wikipedia.org/wiki/Python_(programovac%C3%AD_jazyk)))

2.5 Visual Studio Code

Visual Studio Code je editor zdrojového kódu vyvíjaný spoločnosťou Microsoft pre operačné systémy Windows, Linux a macOS. Obsahuje podporu pre Git, zvýraznenie syntaxu, kontextový napovedač a podporu pre ladenie a refaktorizáciu. Zdrojový kód je slobodný software pod licenciou MIT. Zostavené verzie ponúkané priamo Microsoftom sú freeware obsahujúcim telemetriu, ale existuje aj komunitné zostavovaný variant VSCodium. Editor je naprogramovaný v JavaScripte a TypeScripte.

(https://cs.wikipedia.org/wiki/Visual_Studio_Code)

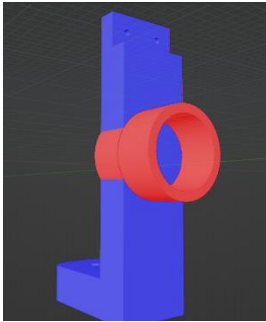
3. Ciele práce

V dnešnej dobe neustále rastie význam automatizácie a robotizácie v priemyselných odvetviach. Možnosti, ktoré prinášajú robotické technológie, sú nevyčerpatel'né, a práve v tomto kontexte je dôležité preskúmať ich potenciál využitia v priemyselnom prostredí. Táto práca sa zameriava na špecifický aspekt robotizácie, a to využitie robotickej ramene v priemyselných aplikáciách. Robotické rameno predstavuje flexibilný nástroj s veľkým potenciálom v rôznych odvetviach, od automobilového priemyslu po farmaceutický sektor. Jeho schopnosť vykonávať opakujúce sa úlohy s vysokou presnosťou a rýchlosťou ho robí ideálnym kandidátom pre aplikácie, kde je potrebná konzistentná a efektívna výroba. Práca sa bude zaoberať programovateľnosťou a možnosťami prispôsobenia robotického ramena konkrétnym potrebám výroby. Cieľom tejto práce je poskytnúť prehľad o tvorbe robotického ramena a identifikovať oblasti, kde by mohlo byť jeho využitie najvýhodnejšie. Zároveň sa bude diskutovať o výhodách a obmedzeniach tohto prístupu a navrhne sa možný smer ďalšieho výskumu v tejto oblasti.

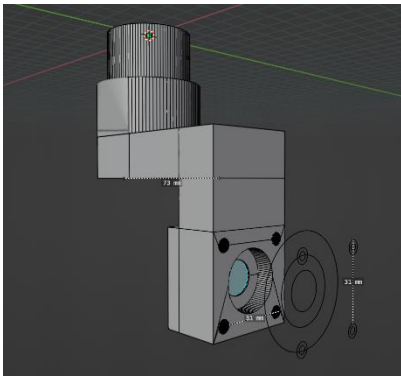
4. Robotické rameno

4.1 Modelovanie a tlač

Ako som už spomínal modelované to bolo v programe Blender, kvôli predošlým skúsenostiam v programe. Program je skôr stavaný na modelovanie vecí len na okrasu, nie presných ale stále sa tam aj to dá robiť. Musel som si zmeniť nastavenie jednotiek z metrov na

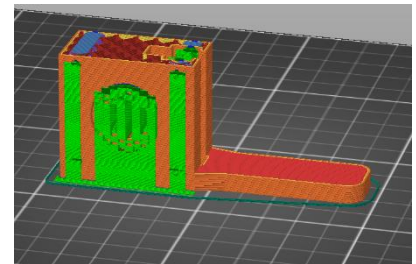


milimetre. Taktiež som si musel dávať pozor na orientáciu normálov lebo keď sú zle orientované (červená farba), pri väčších modeloch to môže spôsobiť problémy pri slicovaní modelu pre 3D tlač. Modeloval som to zo základných tvarov ako sú kocka, valec, guľa. Pri práci s telesom som využíval nástroje ako „extrude“, „knife project“, „scale“ a „grab“. Sú to všetko základne nástroje na prácu s geometriou



v Blenderi. Pre použitie knife projectu som si musel spraviť 2D teleso, ktoré by som vyrezal do hlavného modelu. Snažil som sa používať veľa bodov aby keď sa objekt vytlačí nebol na dotyk hranatý. Na tlačenie som použil Prusa Slicer kde som na menej zaťažených modeloch dával menšiu výplň pre ľahkosť ramena.

Podpery som nemusel modelovať lebo program si ich dokáže vygenerovať sám (zelená farba na obrázku su podpery).

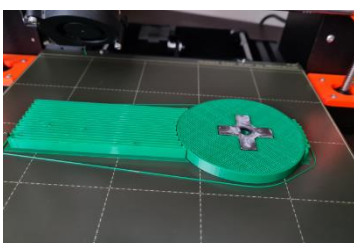


4.2 Skladanie

Najprv som začínal spodkom, ktorý otáča celého robota. Je to spravené cez remeň a ozubené



koleso na motore. V strede je priskrutkované ozubené koleso, vďaka ktorému sa to otáča. Po obvode modelu sú ešte pridané ložiská aby sa to ľahko pohybovalo po platni. Ďalšie na skládanie je prvý kĺb ramena. Tento kĺb je pod najväčším

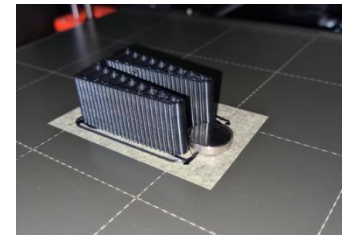


zaťažením z celého robota tak som tam dal najvyššiu prevodovku v pomere 1:50. Bál som sa že by plast pri takej sile mohol začať preskakovať tak som namodeloval miesto pre kovové teliesko

v tvare kríža a v strede výrezom pre motor. Toto teleso som pridal cez tlač, bolo dôležité aby je otočené dobrým smerom aby diery sedeli do seba. Na vrch telesa som ešte natrel vrstvu sekundového lepidla aby neskoršie vrstvy plastu dobre sedeli.



Na model som pridal ešte bočné rameno aby sa lepšie vyvážil model. Toto rameno sa priskrutkovalo na model a vložilo do ložiska. Keď som išiel poskladať koncový afektor chcel som ho spraviť mierne ohybným aby ľahšie dokázal chytat' veci. Tak som nato použil FLEX materiál. Celý afektor je dutý, je tam iba pár pomocných výplní aby to aspoň



trochu bolo pevné.

Nezabudni priložiť Protokol o priebehu realizácie projektu.

4.3 Programovanie

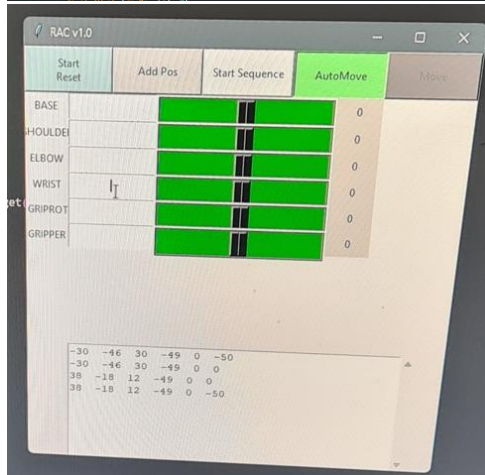
Program som písal v editore Visual Studio Code a v Arduine. V arduine som spravil program v C++, ktorý prijíma cez Sériovú komunikáciu bitové reťazce inštrukcií, ktoré si následne preloží do čísiel a znakov, s ktorými sa už dá pracovať. Ako prvé sa program pozerá aký je

```

void setup() {
  Serial.begin(9600); // Check first char
  Serial.read(); // Remove identification character
  int millisArray = 0;
  int speedArray[] = {0,0,0,0,0,0};
  bool isZerobArray[] = {false,false,false,false,false,false};
  int speedFunction[] = {0,0, 100, 100, 200,200,200};
  int acceleration[] = {0,0, 1000, 1000, 1000, 1000, 1000};
  int deceleration[] = {0,0,0,0,0,0};
  // int numofArrays = 8;
  // Read the data into the array
  for (int i = 0; i < arraySize; i++) {
    myArray[i] = Serial.parseInt(); // Read next int from serial buffer
    Serial.read(); // Remove comma
    if (i == 0) {
      myArray[i] += 2.05;
    } else if (i == 1) {
      myArray[i] += 0.8;
    } else if (i == 2) {
      myArray[i] += 0.2;
    } else if (i == 3) {
      myArray[i] += 5.18;
    } if (myArray[i] == 0) {
      millisArray = 0;
      isZerobArray[i] = true;
    }
  }
  while (millisArray != arraySize) {
    delayMicroseconds(5);
    for (int i = 0; i < arraySize; i++) {
      if (myArray[i] == 0.88 || isZerobArray[i]) {
        isZerobArray[i] = true;
        millisArray++;
      } else if (myArray[i] == 0 || !isZerobArray[i]) {
        continue;
      }
      speedArray[i] += 2;
      if (speedArray[i] == speedFunction[i] + acceleration[i] + deceleration[i]) {
        if (myArray[i] > 0) {

```

znak na začiatku a podľa toho zistí, s ktorým motorom ma hýbať. Ak sa spustil program kde sa využívajú všetky motory, program dostáva pole šiestich čísiel (koľko je motorov) a cez tie následne prechádza. Vytvorí si dočasné premenné, ktoré majú hodnotu rovnakú ako počet otočení. V tejto premennej sa to každým opakovaním odčíta až dokým nie sú všade 0, vtedy program pokračuje na ďalší set inštrukcií. Teraz som musel ešte spraviť program, ktorý bude tieto informácie posielat'. Rozhodol som sa pre program napísaný v pythone a s použitím Serial knižnice pre komunikáciu a knižnice Tkinter pre užívateľské rozhranie.



Najviac času mi zabralo definovanie objektov v a hranie sa s celkovým výzorom programu aby vyzeral

A screenshot of a code editor showing C++ code for controlling a robot arm. The code includes headers for `Arduino.h`, `Wire.h`, and `Adafruit_BMP280.h`. It defines constants for motor pins and a temperature sensor. The `setup` function initializes the sensor and pins. The `loop` function contains logic for moving the arm based on user input and sensor data. Comments in the code describe the movement logic, such as moving the arm to a specific position and then back to its initial position.

moderne. Keď sa program otvorí môžete si hýbať s ramenom po jednom a nastaviť si rameno na pozíciu, ktorú chcete za pomoci jazdcov alebo poli, ktoré keď stlačíte enter sa odošlú. Túto pozíciu tlačítkom alebo ručne môžete napísať do textového poľa, ktoré sa bude posielat' v bitovej forme do arduina. Je tam aj funkcia „AutoMove“, ktorá vám mení či sa robot bude hýbať keď si píšete pozície do textového poľa alebo až keď stlačíte štart programu. Tlačítko „Start/Reset“ vyresetuje robota do začiatkovej polohy. Oba dva programy majú dokopy približne 700 riadkov kódu.

5. Závery práce a zhrnutie

Nezamestnanosť je problém, ktorý sa v súvislosti s používaním robotických rúk stáva čoraz zreteľnejším. Ale v skutočnosti môže tento technologický pokrok prinášať pozitívne vplyvy. Aj keď vytvára potenciálne riziko pre pracovné miesta, umožňuje tiež rozvoj nových odvetví a pracovných príležitostí. Roboty nám umožňujú prenechať monotónne a nebezpečné práce na stroje, čím sa oslobodia ľudské zdroje pre kreatívnejšie a významnejšie úlohy. V dôsledku toho môžu ľudia investovať svoj čas a energiu do tvorby a inovácie nových robotických technológií, čím sa posilňuje celková ekonomika a spoločenský pokrok. Takže hoci nezamestnanosť kvôli robotickým rúkam je reálnym problémom, môže to viesť k novým možnostiam a lepšiemu využitiu ľudského potenciálu v oblastiach, ktoré prinášajú väčšiu spokojnosť a úžitok pre spoločnosť ako celok.

6. Zoznam použitej literatúry

[1]STECOVÁ, Eva a ŠVICKÝ, Emil. 1999: Asanácia. 1. vyd. Bratislava: Proxima Press, 1999. s. 72-82. ISBN 80-85454-26-2

[3]ISO 690-2: 1997, Information and documentation – Bibliographic references Part 2: Electronic documents or parts thereof.

[4]MINISTERSTVO FINANCIÍ SR: Metodický pokyn na použitie odborných výrazov pre oblasť informatizácie spoločnosti. Bratislava: MF SR, [cit. 2009-02-05].

Dostupné na internete: http://www.informatizacia.sk/ext_dok-metodicky_pokyn_glosar_pojmov/3482c.

O programe Blender. <https://sk.wikipedia.org/wiki/Blender>

O Arduine <https://sk.wikipedia.org/wiki/Arduino>

O Pythone [https://sk.wikipedia.org/wiki/Python_\(programovac%C3%AD_jazyk\)](https://sk.wikipedia.org/wiki/Python_(programovac%C3%AD_jazyk))

O programe Visual Studio Code https://cs.wikipedia.org/wiki/Visual_Studio_Code

7. Prílohy

Výpočty potrebných síl motorov.

Arm 5:	1.00 kg	Arm 4:	1.00 kg	Arm 3:	1.00 kg	Arm 2:	1.00 kg	Z-AXE 1:	5.10 kg
Dĺžka ramena	0.15 m	Dĺžka ramena	0.30 m	Dĺžka ramena	0.40 m	Dĺžka ramena	0.60 m	Dĺžka ramena	0.45 m
Fg	9.81 N	Fg	9.81 N	Fg	9.81 N	Fg	9.81 N	Fg	50.03 N
Fg		Fg		Fg		Fg		Fg	
suma	20.60 N	suma	30.41 N	suma	40.22 N	suma	50.03 N	suma	100.06 N
Mk	3.09 Nm	Mk	9.12 Nm	Mk	16.09 Nm	Mk	30.02 Nm	Mk	45.03 Nm

Model	Step Angle	Current	Torque	Length	Wires
42HS3413A4	1.8°	1.3A	28Ncm	34mm	4线
42HS4013A4	1.8°	1.3A	40Ncm	40mm	4线
42HS4813A4	1.8°	1.3A	52Ncm	48mm	4线
42HS6015A4	1.8°	1.5A	70Ncm	60mm	4线

Model	Step Angle	Current	Torque	Length	Ratio
42HS3413A4-XG3.71	1.8°	1.3A	28N.cm	34mm+L	3.71
42HS3413A4-XG5.18	1.8°	1.3A	28N.cm	34mm+L	5.18
42HS3413A4-XG14	1.8°	1.3A	28N.cm	34mm+L	14
42HS3413A4-XG19	1.8°	1.3A	28N.cm	34mm+L	19
42HS3413A4-XG27	1.8°	1.3A	28N.cm	34mm+L	27
42HS3413A4-XG50	1.8°	1.3A	28N.cm	34mm+L	50
42HS3413A4-XG100	1.8°	1.3A	28N.cm	34mm+L	100
42HS3413A4-XG139	1.8°	1.3A	28N.cm	34mm+L	139

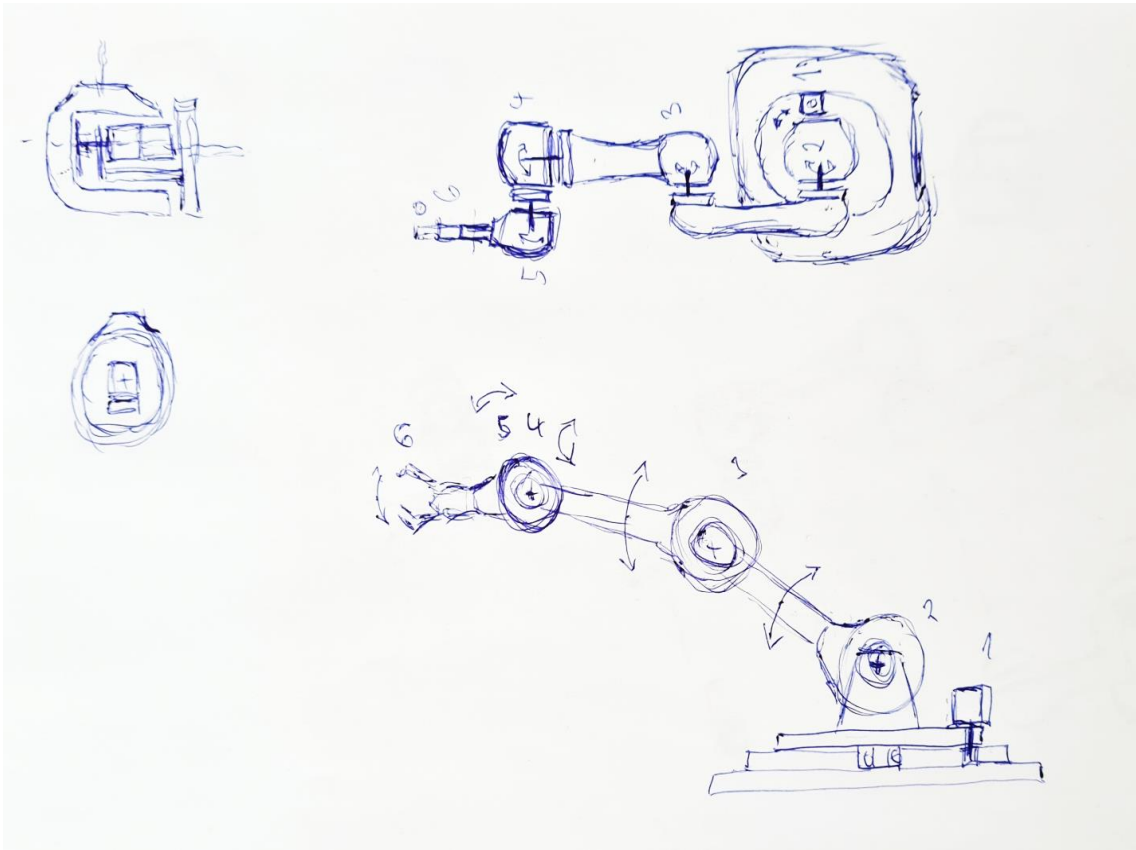
Model	Step Angle	Current	Torque	Length	Ratio
42HS4013A4-XG3.71	1.8°	1.3A	40N.cm	40mm+L	3.71
42HS4013A4-XG5.18	1.8°	1.3A	40N.cm	40mm+L	5.18
42HS4013A4-XG14	1.8°	1.3A	40N.cm	40mm+L	14
42HS4013A4-XG19	1.8°	1.3A	40N.cm	40mm+L	19
42HS4013A4-XG27	1.8°	1.3A	40N.cm	40mm+L	27
42HS4013A4-XG50	1.8°	1.3A	40N.cm	40mm+L	50
42HS4013A4-XG100	1.8°	1.3A	40N.cm	40mm+L	100
42HS4013A4-XG139	1.8°	1.3A	40N.cm	40mm+L	139

Model	Step Angle	Current	Torque	Length	Ratio
42HS4813A4-XG3.71	1.8°	1.3A	52N.cm	48mm+L	3.71
42HS4813A4-XG5.18	1.8°	1.3A	52N.cm	48mm+L	5.18
42HS4813A4-XG14	1.8°	1.3A	52N.cm	48mm+L	14
42HS4813A4-XG19	1.8°	1.3A	52N.cm	48mm+L	19
42HS4813A4-XG27	1.8°	1.3A	52N.cm	48mm+L	27
42HS4813A4-XG50	1.8°	1.3A	52N.cm	48mm+L	50
42HS4813A4-XG100	1.8°	1.3A	52N.cm	48mm+L	100
42HS4813A4-XG139	1.8°	1.3A	52N.cm	48mm+L	139

0,7 NM						
otacky 60/min, 1/sec strata 20% vykon				teoret krutis	Nosnost KG n	Nosnost KG na 600mm
Pomer	otacak za Ml	cas	Sec na 4	Moment	teoreticky	teoreticky
3.71	0.27	0.93	2.08		0.71	0.12
5.18	0.19	1.30	2.90		0.99	0.17
14	0.07	3.50	7.84		2.66	0.45
19	0.05	4.75	10.64		3.62	0.61
27	0.04	6.75	15.12		5.14	0.87
50	0.02	12.50	28.00		9.51	1.62
71	0.01	17.75	39.76		13.51	2.30
100	0.01	25.00	56.00		19.03	3.23
139	0.01	34.75	77.84		26.45	4.49
Gripper 6	bez prevodovky			0.17		
Z-AXE 1	bez prevodovky			0.40		

0,7 NM						
otacky 300/min, 5/sec strata 50% vyko				teoret krutis	Nosnost KG n	Nosnost KG na 600mm
Pomer	otacak za Ml	cas	Sec na 4	Moment	teoreticky	teoreticky
3.71	1.35	0.19	1.04		0.35	0.06
5.18	0.97	0.26	1.81		0.62	0.10
14	0.36	0.70	4.90		1.66	0.28
19	0.26	0.95	6.65		2.26	0.38
27	0.19	1.35	9.45		3.21	0.55
50	0.10	2.50	17.50		5.95	1.01
71	0.07	3.55	24.85		8.44	1.43
100	0.05	5.00	35.00		11.89	2.02
139	0.04	6.95	48.65		16.53	2.81

Skoré náčrty



Torque Formula

ScienceFacts.net

Example: Closing a door

