TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Department of Electrical Power Engineering and Mechatronics

HUMANOID ROBOTI KÄE-LAADNE MANIPULAATOR

DESIGN OF HUMANOID ROBOT ARM MANIPULATOR

MAGISTRITÖÖ

Üliõpilane: Aleksei Meshkov

Üliõpilaskood: 174822MAHM/177948EV

Juhendaja: Mart Tamre, EE - Department of Electrical Power Engineering and Mechatronics

Tallinn, 2018

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Department of Electrical Power Engineering and Mechatronics

DESIGN OF HUMANOID ROBOT ARM MANIPULATOR

HUMANOID ROBOTI KÄE-LAADNE MANIPULAATOR

MASTER THESIS

Student: Aleksei Meshkov

/name/

Student code: 174822MAHM/177948EV

Supervisor: Mart Tamre, EE - Department of Electrical Power Engineering and Mechatronics

/name, position/

Tallinn, 2018

AUTHOR'S DECLARATION

Hereby I declare, that I have written this thesis independently.

No academic degree has been applied for based on this material. All works, major viewpoints and data of the other authors used in this thesis have been referenced.

"……." …… 2018

Author: Aleksei Meshkov

Thesis is in accordance with terms and requirements

"……." …… 2018

Supervisor: Mart Tamre

Accepted for defence

"……." …… 2018

Chairman of theses defence commission: ……

Department of Electrical Power Engineering and Mechatronics

THESIS TASK

Student: Aleksei Meshkov 174822MAHM/177948EV

Study programme, main speciality: MAHM02/13 - Mechatronics

Supervisor(s): Professor Mart Tamre, 620 3202

Consultant: ……..

Thesis topic:

(in English) DESIGN OF HUMANOID ROBOT ARM MANIPULATOR

(in Estonian) HUMANOID ROBOTI KÄE-LAADNE MANIPULAATOR

Thesis main objectives:

Control and structural design of robotic arm manipulator for humanoid robot.

Thesis tasks and time schedule:

№	Task description	Deadline
1.	Theory analysis	18.03.2018
2.	SolidWorks model composition	15.04.2018
3.	Components selection	20.04.2018
4.	Control design	5.05.2018
5.	Final report presentation	15.05.2018

Language: English. **Deadline for submission of thesis**: "25"may 2018

Student: Aleksei Meshkov …….. "……."……2018

Supervisor: Mart Tamre ..... "....."....2018

Consultant: …….. ..... "....."....2018

TABLE OF CONTENTS

LIST OF ABBREVIATIONS 3

PREFACE 4

1. INTRODUCTION 5

2. LITERATURE REVIEW 6

3. THEORY ANALYSIS 9

3.1 Human arm motion 9

3.2 Material selection 11

3.3 Torque calculations 11

3.4 Geometrical gear calculations 13

4. ARM DESIGNING PROCESS 15

4.1 Humanoid robotic arm concept design 15

4.2 The first robotic arm prototype 16

4.3 The second robotic arm prototype 17

4.3.1 The second robotic arm prototype parts description 17

5. ELECTRICAL COMPONENTS SELECTION 23

5.1 Actuation mechanism selection 23

5.1.1 Finger actuation 23

5.1.2 Wrist actuation 24

5.1.3 Elbow actuation 25

5.1.3 Arm-Elbow actuation 26

5.1.3 Shoulder-arm actuation 27

5.2 Sensor selection 30

5.2.1 Angular positon sensor 30

5.2.2 Pressure sensor 34

5.3 Control device selection 35

5.3.1 Actuators and sensors controller 35

5.4 Power supply selection 37

6. CONTROL DESIGN 40
6.1 SolidWorks into SimScape model conversion 40
6.2 Forward kinematics 42
6.3 Inverse kinematics problem statement 43
6.4 Neural network-based solution for inverse kinematics task 44
6.4.1 Neural networks description 44
6.4.2 Neural network training data sets composition and parameters selection 45
6.4.3 Neural network training process description 46
6.4.4 Trained neural network testing 47
6.5 ARM CONTROL FLOW CHART 49
7. FURTHER WORK 50
8. SUMMARY 51
REFERENCES 52
APPENDICES 56

LIST OF ABBREVIATIONS

DoF – degree of freedom

FDM - fused deposition modeling

3D printing – three dimensional printing

NN – neural network

PLA – polylactide

ABS - acrylonitrile butadiene styrene

NEMA - National Electrical Manufacturers Association

DC - direct current

AC - alternating current

CAD - Computer-aided design

PREFACE

I would like to thank ITMO University's mechatronics department and TTU mechatronics department for giving me the opportunity to participate in double-degree educational program. In addition, I would like to thank my supervisors Svetlana Perepelkina and Mart Tamre, for providing guidance during this thesis composition, my friends and family, who supports me through my studying.

1. INTRODUCTION

With the rapid growth of technological progress, robotics nowadays can be found in wide application fields range: from industry to household usage. Humanoid robots are vivid example of robotics application. Humanoid robots nowadays are used to fulfill different tasks such as tool handling, social activities simulation, robotic competitions, research purposes, etc.

This paper is dedicated to robotic arm for humanoid robot mechanical design process description and forward and inverse kinematics task solution description, with the use of MatLab neural network toolbox.

The humanoid robot, for which robotic arm is being developed, represents an upper body part with two human arm-like manipulators and mounting for cameras, which was created to resemble a human head. This robot was a project of TTU bachelor students, that was based on open-source InMoov 3D-printed robot project. The InMoov project is still in development, but basic principles of arm design remained the same.

After analysis of the existing robot arm, several tasks were determined, that new design should solve:

1) Weight, that arm should be able to carry, should be increased to 4,905 N.

2) Additional rotation of the wrist should be added to position hand more precisely.

3) Elbow rotation actuation proved to be unreliable, so actuation method should be changed.

4) Weight of the arm should be reduced.

5) Positioning control should be designed

Most of this work have been done using SolidWorks 2015, MatLab 2016b and SimScape MatLab toolbox software.

2. LITERATURE REVIEW

In order to design a robotic arm there is a need to study existing solutions, their pros and cons. In this work, following humanoid robots were studied:

1) SoftBank Robotics Pepper

Pepper Is an autonomous humanoid robot, which is designed to be human assistant and guide. It have human-like form, as can be seen on Figure 2.1.

Figure 2.1 SoftBank robotics Pepper robot [1]

As stated on manufacturers web-site, Pepper robot is used by several organizations, such as banks and shops, to provide customer service support [1]. Using cameras, mounted on the head of the robot, pepper can recognize and react to peoples facial expressions. The robot have two 5 DoF arms with 5 fingers each, which can be used to reproduce complex human-like arm motion. Since pepper was built as a human guide, the arm cannot lift any additional weight. This solution was probably made in order to preserve compact shape of the robot (1,20 x 0,425 x 0,485 m) and to keep the robot light-weighted (mass – 28 kg) [2] . The cost of the Pepper robot is 1407 euro.

2) InMoov open source 3D-printed life sized robot

InMoov is a project of French designer Gael Langevin, which was created as a development platform, on which InMoov community could build their own projects. It is presented on Figure 2.2.

Figure 2.2 InMoov open source robot [3]

This project was designed to be easily reproducible. All parts, apart from electrical components, are designed to be manufactured, using 3D printer. InMoov robot consists of two 5 DoF arms with tendon-driven fingers, head with sound sensors and using Kinect, in order to capture gestures and react accordingly [3]. This project is still in development and constantly improving itself with the help of InMoov community, which is rather wide, thanks to the open-source distribution model of the project. The maximum lifting weight is not stated, but creator published a video on YouTube, where built InMoov robot holding automatic screwdriver, which in general have mass around 1 kg.

3) TTU mechatronics department humanoid robot

This project was built by group of TTU mechatronics department's bachelor students, and this is the robot, for which robotic arm was being developed and described in this work. It's appearance is presented on Figure 2.3.

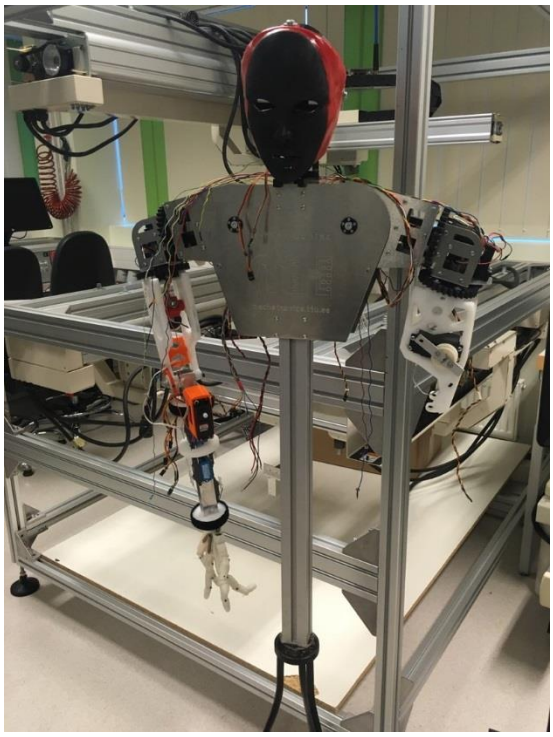


Figure 2.3 TTU mechatronics department humanoid robot

Although this project was based on InMoov robot, there are several differences. In this project some parts were made, using CNC-cutter. Robot consists of 4 DoF arms, on which different actuation methods were tested. Hand of this robot uses one-tendon finger actuation with springs, which are used to return fingers into initial position. Possible lifting weight is not stated. In order to define, what should be improved in the arm design, several problems of existing arm were stated. Elbow actuation method was made through worm-like gearbox, with servo drive connected to the helical shaft, which was screwing into the nut, placed on the stopper. This method of actuation was allowing 40-60 degrees rotation only, which is not enough to reproduce human arm-like motion, and proved to be unreliable. Finger actuation drives were consuming large amount of space in the wrist part of the robot, which made it impossible to implement additional degree of freedom to the wrist, which other reviewed project possess. Mass of the arm could be reduced, to increase possible lifting weight. Cable management could be improved. Elbow angle measurement method should be changed, since 3D-printed gear, that was used to drive potentiometer, was printed poorly and didn't gave required gear coupling.

Based on reviewed projects, described in this chapter, and stated problems of current arm design main designing objectives were set.

3. THEORY ANALYSIS

3.1 Human arm motion

In order to design a human arm-based manipulator, human arm kinematics is needed to be considered. The human arm motion can be imagined as a 7 DoF manipulator. Its kinematics scheme is shown on Figure 3.1.

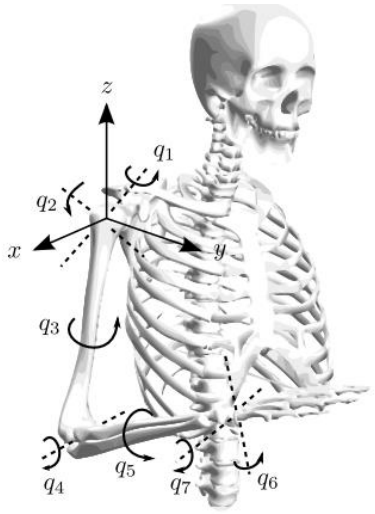


Figure 3.1 Human arm motion kinematics scheme representation [4]

Human arm is a complex mechanism, in which each joint might have several degrees of freedom, and several joints can move simultaneously [5].This type of mechanism is hard to reproduce by technical means, several joints must be used, in order to be able to create human arm-like motion. Keeping that in mind, kinematics scheme, as shown on Figure 3.2 was presented.

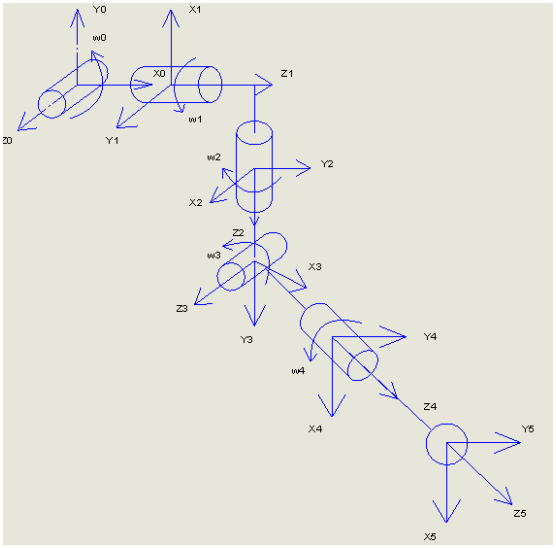


Figure 3.2 Presented humanoid arm kinematics scheme:

x,y,z – joint's rotation coordinate axis, *w* – rotation around z axis

Previous version of the robotic arm had provided 4-DoF. In the designed robotic arm, it was decided to include wrist rotation movement, so the resulting arm should be represented as 5-DoF rotation manipulator: two degrees of freedom for the shoulder movement, one for arm rotation, one for elbow rotation and one for wrist rotation, which schematic representation is shown on Figure 3.3.

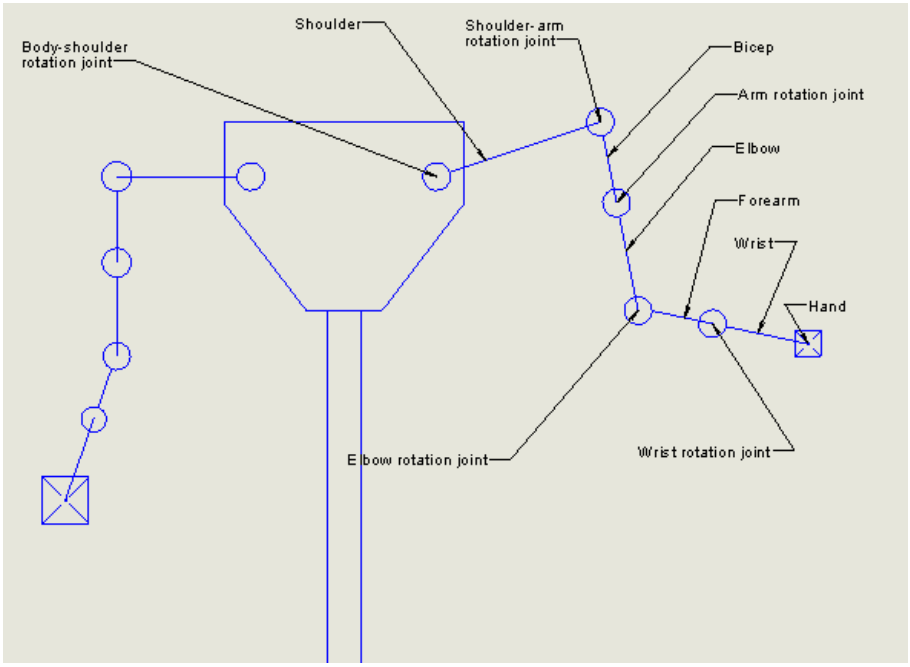


Figure 3.3 Humanoid robot composition scheme

3.2 Material selection

Material selection process was conducted, based on technical capabilities of TTU mechatronics department's laboratory. Parts of the designed robotic arm should be able to be produced and assembled by the means of the laboratory. Materials should be light weighted and durable, since allowed payload, that arm could handle, strongly depends on the mass of the manipulator. Most parts of the robotic arm, excluding drives, controllers, wires, gears and power supplies should be manufactured in the laboratory, using CNC machine and FDM 3D printing technology.

For FDM 3D printing there are several solutions, presented on the market nowadays. The most widely spread materials for 3D printing are PLA and ABS plastics. After reviewing information about these materials [6] [7] following comparison chart was composed, that presented on Table 3.1.

Table 3.1 PLA and ABS plastic thermal capabilities comparison

Material	PLA	ABS
Extrusion temperature	170..180 °C	≈180 °C
Glass transition temperature	≈50 °C	≈100 °C
Temperature shrinkage	Low	≈0,8 % of resulting volume

Since in designed robotic arm, small 3D-printed parts are planned to use, temperature shrinkage is decisive factor of choosing material, based on the fact, that PLA is more resistive to the thermal change of volume, it was chosen as a material for 3D-printed parts.

For CNC-manufactured parts, as a material aluminium alloy was chosen.

3.3 Torque calculations

In order to choose a motor, that could hold required stale torque, critical torque for each joint was calculated, starting from the last joint. The calculation method is relied on moments equality. For each joint the critical mass was calculated and the center of the masses was found. Also moment

of friction forces was taken into account. It is fair to say, that in critical point moment of masses, coming from the masses center should be equal or less, than a stale torque a drive can provide. Relied on this statement, following equation was composed:

$$M_d = M_{Ft} + M_{Ff} \tag{1.1}$$

In which M_{ft} – force, which is created by the weight of the part, M_{Ff} – the friction forces moment. This parameters were calculated as presented below:

$$M_{Ft} = m * g * l \tag{1.2}$$

Where m – mass of the part, g - gravitational acceleration, l – length to the parts center of masses. In order to reduce friction forces between parts, ballbearings were used, so the friction forces momentum can be calculated as shown below:

$$M_{Ff} = f_{fr} * F_{bb} * \frac{d}{2} \tag{1.3}$$

Here, f_{fr} – friction coefficient in ball-bearings, d – inner diameter of ball-bearings, F_{bb} – forces, that applied in both radial and axial directions, which can be defined as shown below:

$$F_{bb} = \sqrt{{F_r}^2 + {F_a}^2} \tag{1.4}$$

Each component mass were defined in SolidWorks mass analysis, and for each part, center of the masses was found. Then points, where the influence of rotating parts mass was maximal, were determined. Example of mass calculation for wrist rotation joint is shown on Figure 3.4.

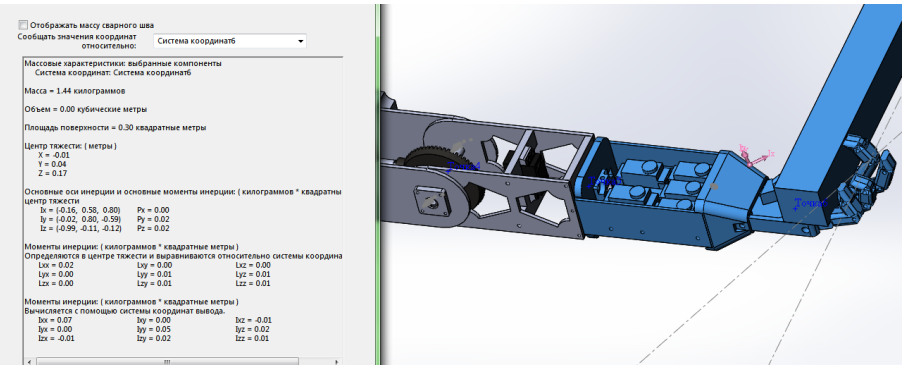


Figure 3.4 – Wrist-part mass and center of masses position calculation

After these parameters were determined, formula (1.1) is transferred into:

$$M_d = M_{Ft} + M_{Ff} = m * g * l + f_{fr} * \sqrt{{F_r}^2 + {F_a}^2} * \frac{d}{2} \quad (1.5)$$

As an example, the parameters for wrist rotation calculations are: m = 1,44+0,4 kg; g = 9,81 $\frac{m}{s^2}$;

$l = \sqrt{0{,}04^2 + 0{,}01^2} = 0{,}041\ m$; d = 15 mm = 0,015 m; f_{fr} = 0,002; $\sqrt{{F_r}^2 + {F_a}^2}$ = 86,7 N.

Then the required torque for wrist rotation was calculated as:

$$M_d = 1{,}84 * 9{,}81 * 0{,}041 + 0{,}002 * 86{,}7 * \frac{0{,}015}{2}\left[kg * \frac{m}{s^2} * m + N * m\right] = 0{,}74\ [N * m] \quad (1.6)$$

For every other joint similar calculations have been made.

3.4 Geometrical gear calculations

During designing process a need in usage of gear boxes has occurred, to avoid implementation of expensive drives directly to the rotation shaft. In this work, several types of gearboxes were used.

In case of standard torque multiplication, spur gearbox was used. Spur gears are the simplest gears, they made out of various materials such as steel, brass, nylon e.t.c.

In case, where rotation had to be translated by 90 °, worm-drives gearboxes and bevel gearsets were considered as a possible solution. It was decided to use bevel gears over worm-drive gearbox, because in order to create a required torque on the wheel of the worm-drive gearbox, high speed of rotation on the drive is needed, which is impossible requirement to reach, since servo motors with defined angle of rotation were planned to use as a drives. Another big issue of worm-drive gearsets is heat dissipation and lubrication. Bevel gearboxes are more reliable and it is easier to align them.

Other gearbox type that was used - planetary gearheads, but in case of these types of gearboxes, its parameters were not calculated. They were chosen from manufacturer instead, based on a gear ratio and compatibility with the drive. Planetary gearboxes are viable solution for high torque multiplication, with low space consumption, low backlash and input and output shaft alignment, but usually weight, cost and complicity of the planetary gearboxes makes their usage in simple mechanisms unreasonable. In designed robot torque requirements and space restrictions, lead to choosing planetary gearhead as a step motor gearbox.

The algorithm of choosing the spur gearboxes was conducted, according to the guides and articles [8] and [9]. Geometrical calculations for the gears implies gear selection, according to the gear set geometrical parameters. It this work, gear sets were chosen by number of teeth, gear ratio and gear module parameters from gear manufacturers web site. Designing task didn't imply designing and manufacturing of gear sets, since gears should be well made, in order to prevent unnecessary problems, that could occur due to poorly made gearboxes.

4. ARM DESIGNING PROCESS

4.1 Humanoid robotic arm concept design

After reviewing theory and solutions on the robotic arms field, the basic concept of the arm was proposed. Its appearance presented on Figure 4.1.

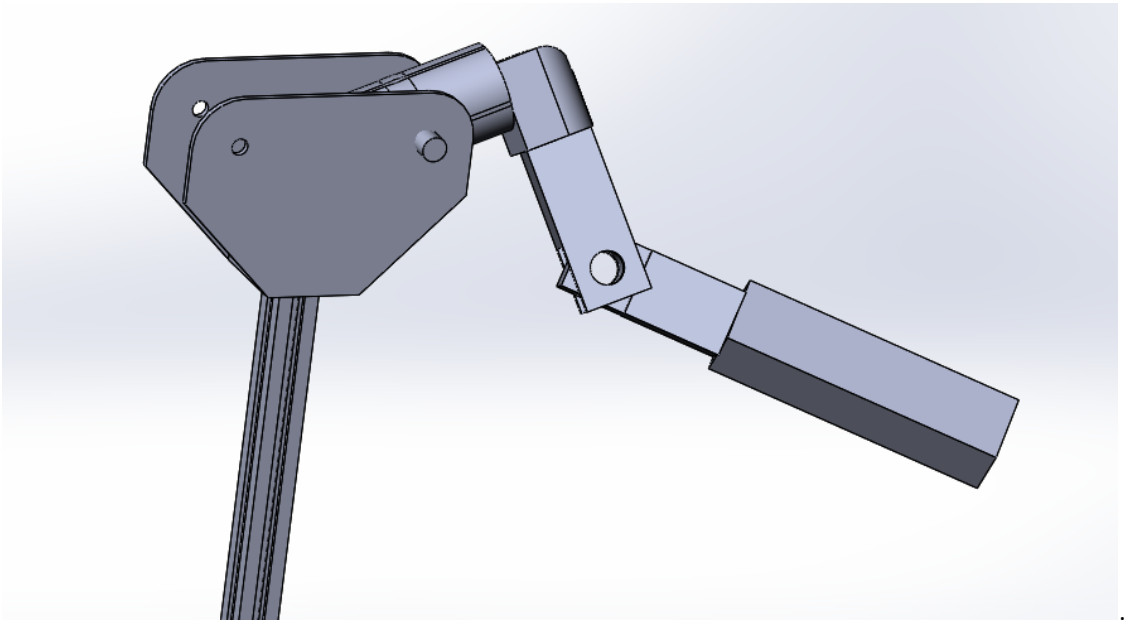


Figure 4.1: Basic concept of the robotic arm

Robotic arm concept scheme consists of 5 links: shoulder link, shoulder-arm connection link, elbow link, forearm and wrist link, - and of 5 joints: body-shoulder rotation joint, shoulder-arm rotation joint, arm rotation joint, elbow rotation joint and wrist rotation joint. This basic concept was created in order to compose first prototype of the arm.

4.2 The first robotic arm prototype

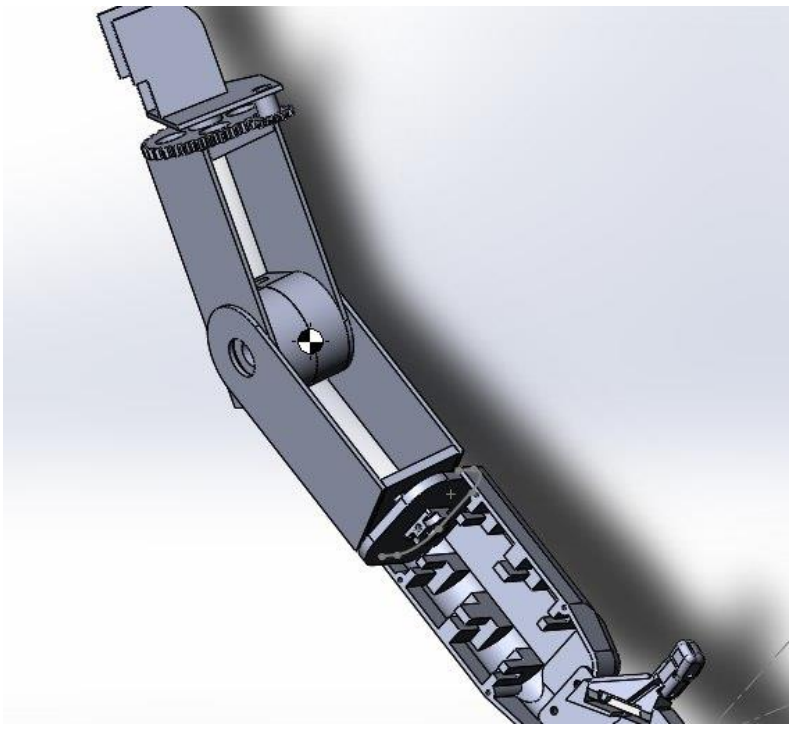


Figure 4.2: First prototype of the robotic arm design

The first prototype appearance is presented on Figure 4.2, it represents 5DoF manipulator, with hand rigidly fixed on the wrist. It consists of 5 rotary joints and 5 links. The actuation of rotary joints were planned to conduct, using servomotors with different gearboxes. For arm, shoulder and wrist rotation, was planned to use standard spur gearbox systems, for elbow rotation it was expected to implement worm-drive gearbox. After resulting torques and mass calculations, several issues were determined:

1) Worm-drive gearboxes are unreliable under heavy stall torque;

2) Presented wrist construction, with finger actuators located on the closed wrist hull, was heavy-weighted, which was resulting in decreasing mass of allowed payload;

3) For the resulting torques on the shoulder rotation joint, it was required to use drives and gearboxes, which geometrical parameter exceeded capabilities of the concept, which could lead to increased size of the robotic hull.

4.3 The second robotic arm prototype

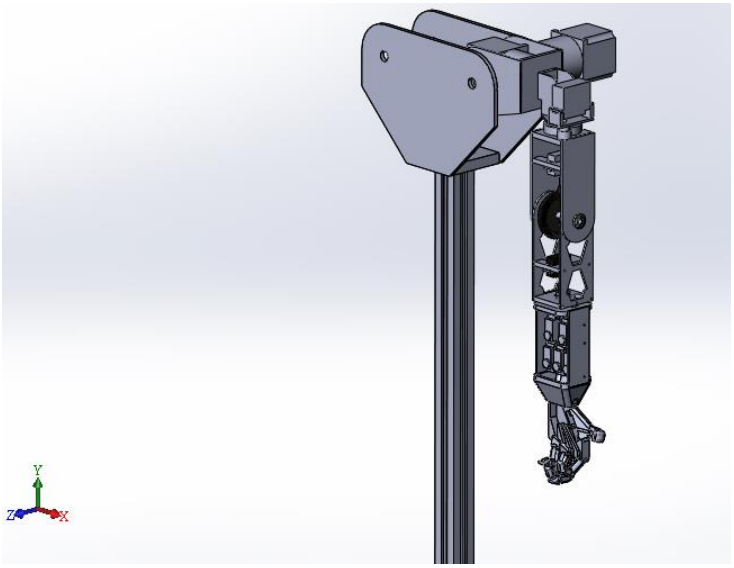


Figure 4.3: Second prototype of the robotic arm design

The second prototype, which is shown on Figure 4.3, design's basic concept remained the same. The arm consists of the 5 rotary joints and 5 links with the hand, rigidly connected to a wrist. Elbow joint rotational worm-drive gearbox was changed to the bevel gearbox, also different placement of the drive inside the arm hull was considered, in order to use standard spur gearbox, but due to cable management and drive positioning issues, it was decided to use bevel gearbox instead. The arm can be logically divided into five different parts: the hand, the wrist, the forearm, the elbow, the shoulder-arm part, the body-shoulder part and the robotic stand.

4.3.1 The second robotic arm prototype parts description

The hand model of the robot is presented on Figure 4.4.

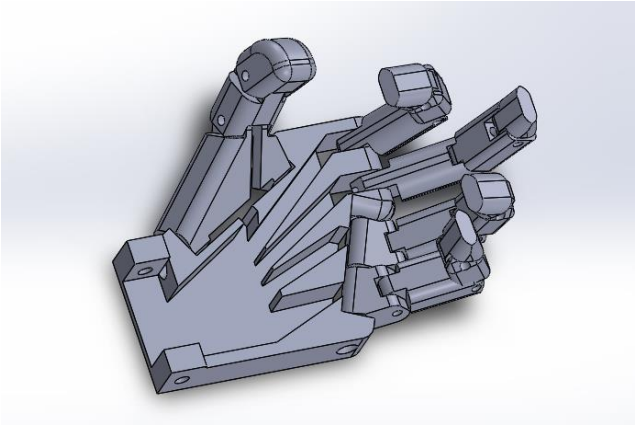


Figure 4.4: Second prototype robotic arm hand

It consists of hand base and five fingers. Each finger is composed out of three parts. Each part of the finger have a straight cut inside for actuation wires and for pressure sensor cables.

Actuation of the fingers was designed to me made through two separate tendons, that connected separately to the tip of the finger, and attached to the actuation servo motors, that located on the wrist part of the arm. The basic idea of this type of actuation robotic fingers is described as under-actuation method [10]. Servo motor rolls the wire on a coil, while simultaneously uncoiling the second wire. Wire pulls the finger tip, locking its movement by the following finger part, until the finger completely bent.

The hand is designed to be manufactured by 3D-printing from PLA plastic, in order ease the maintenance process and for easier parts replacement, if needed.

The mass of this part is expected to be 160 g.

The wrist model is presented on Figure 4.5.

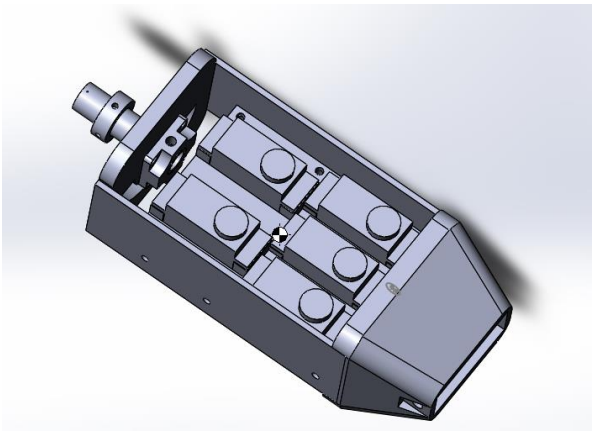


Figure 4.5: Second prototype robotic arm wrist

It holds five finger actuation servo motors on a 3D-printed servo motors holder, which attached to the wrist's hull, using six M4x12 stainless steel screws. The hand of the robot is designed to be attached to the wrist's hull, using single M6x70 bolt and nut. The hull of the wrist is made of two aluminium plates, which holds all parts of the wrist and attached to the rotation base pad, which is designed to be made of PLA plastic. The rotation base have wire handling holes, and firmly connected to the wrist rotation shaft, using a designed connector, that is attached to the rotation base pad with four M3x8 screws. The connection between shaft and connector is ensured by M5x35 bolt and a nut.

The expected mass for this part is 840 g.

The forearm model is shown on Figure 4.6.

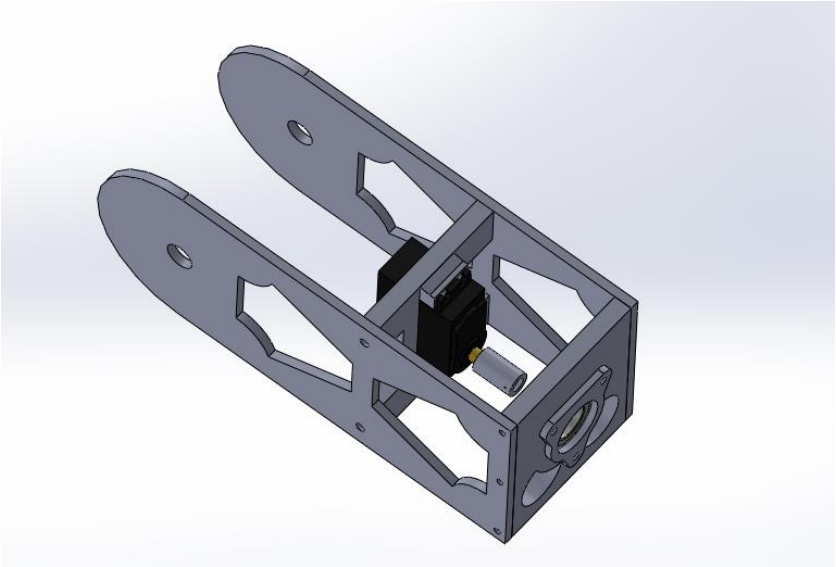


Figure 4.6: Second prototype robotic arm forearm

It contains wrist rotation servo motor, that is mounted between two aluminium plates, using four M4x12 screws. Shaft of the servo motor is connected to the wrist rotation shaft, using a designed coupling with breakable pin, which should break, if the torque, that is applied to a drive shaft, exceeds servo motor by 1,25 of allowed torque. This part firmly connected to an elbow rotation shaft.

The expected mass for this part is 415 g.

The elbow model is shown on Figure 4.7.

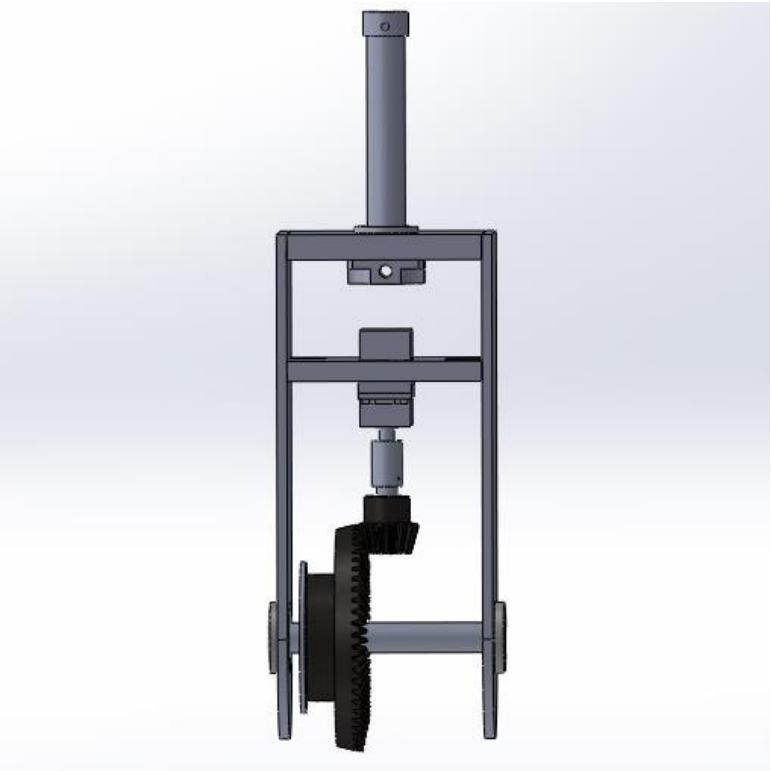


Figure 4.7: Second prototype robotic arm elbow

The elbow contains elbow joint rotation servo motor, that attached using the same type of mounting, as was stated in forearm part description, bevel gearset with gear ratio of 4 to create the required rotary torque. The hull of the elbow is composed, using two cnc-machined aluminium parts and a rotary base with shaft connector, that is firmly connected to an arm rotary shaft using specified connector, as described in the wrist part description. The connection between bevel gear and servo motor shaft Is insured by the coupling, the same way as stated in forearm description. The amount of torque, that the locking pin could endure is limited by 1,25 of allowed motor torque. The elbow shaft rotation is ensured by two ball bearings, installed on the hull, closed by a ball bearings cap on one side, and locked by the fluoroplastic sealing on the other. Ball bearings and sealing were used in order to reduce friction forces between mechanism's parts.

The calculated coupling breakable pin diameter for this drive was 1 mm, and, according to calculations, it can hold amount of torque equal to 1,83 N*m.

The mass of this part is expected to be 1369 g, it differs from other parts due to the specifications of the bevel gears material, that were used to rotate the elbow joint.

The shoulder-arm connector model is presented on Figure 4.8.

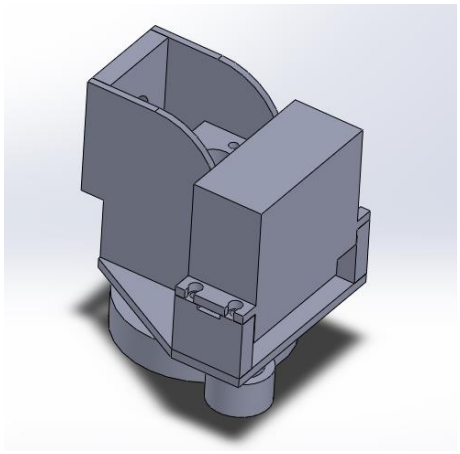


Figure 4.8: Second prototype robotic arm shoulder-arm connector

This part is responsible for arm rotation and rotates around shoulder itself. It consists of aluminium servo drive mounting plate, two aluminium hull plates, two PLA shaft holding pads, spur gearset and servo drive, that is used for arm rotation.

The calculated mass for this part is 486 g.

The shoulder model is shown on Figure 4.9.

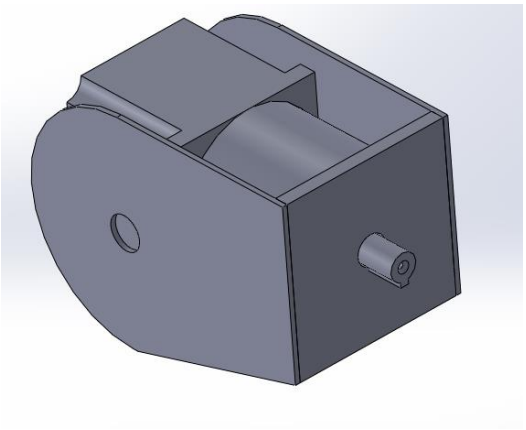


Figure 4.9: Second prototype robotic arm shoulder

This part consists of two hull aluminium plates, attached to a drive mounting pad, and a step-motor with planetary gearhead.

The expected mass for this part is 4600 g, due to heavy step motor with planetary gearhead.

The robotic stand is presented on the Figure 4.10.

Figure 4.10: Second prototype robotic stand

The robotic stand construction was remained the same as in the previous robot, in order to be compatible with the previous versions of the arm and the head. Shoulder rotation drive installed on the back of the robot. Connection between robotic stand and shoulder part was designed to be made with the use of ball bearings and sealing, as stated in elbow part description.

The resulting calculated mass for the whole arm is 8,35 kg,

All torque calculation were made with 400 g. above arm's weight, in order to remain available space for installing different components, such as additional pressure sensors, encoders and other light-weighted devices.

5. ELECTRICAL COMPONENTS SELECTION

The designed robotic arm consists of several electronic components: actuators, sensors, controllers and power supplies. In this chapter selection process for each component were described.

5.1 Actuation mechanism selection

In actuators selection process, in order to reduce manufacturing costs of the designed robotic arm, actuators, which were used in the previous version of the arm, are preferred selection option.

5.1.1 Finger actuation

To actuate fingers servo motor HK 15298B was chosen. Its parameters are presented below, and the drive itself is shown on Figure 5.1 [11]:

Torque: 0,9 N*m (6 V), 1,13 N*m (7.4 V)

Mass: 66.4 g

Speed: 0,21 s/ 60° (6 V), 0,17 s/ 60° (7.4 V)

Operating voltage: 4.8-7.4 V

Motor type: Coreless

Gear: Metal

Type: Digital

Figure 5.1: HK15298B coreless servo motor [11]

This motor was used in previous version of robotic arm to actuate tendon-driven fingers. In the new hand design, it was decided to change springs, that returns the finger into initial position, with parallel tendon, attached to the same drive. The main principle of such actuation is when one tendon is being coiled into drive shaft, other simultaneously uncoil, releasing holding force from the finger. For each tendon separate conduit is going to be used, in order to reduce friction forces between tendons, and to prevent tendons from collision.

5.1.2 Wrist actuation

To actuate wrist, the calculated required torque was found as 0,707 N*m. According to this torque, it was possible to install servo drive straight to the wrist rotation shaft, using specific coupling. The wrist should be able to rotate up to 180 °, according to the designing task. Following servo drives were considered, during the wrist actuation selection process: Towerpro MG996R [12], Firelap LS-RS150M [13] and JX Servo PDI-5521MG [14]. Their appearance presented on Appendix 1. The possible drive options comparison chart is presented on Table 5.1

Table 5.1 Possible wrist actuation solutions comparison chart

Name	Towerpro MG996R	Firelap LS-RS150M	JX Servo PDI-5521MG
Operating voltage, V	4.8 - 7.2	4.8 ~ 6.0	4.8-6.6
Maximum torque, N*m	1,27 (4.8 V) 1,47 (6 V)	1,27 (4.8 V) 1,47 (6.0 V)	1,69 (4.8 V) 1,99 (6 V)
Maximum speed	0,17 s / 60 ° (4.8 V) 0,13 s / 60 ° (6.0 V)	0,17 s /60° (4.8 V) 0,15 s /60 ° (6.0 V)	0,18 s /85 ° (4.8 V) 0,16 s /85 ° (6 V)
Maximum rotation angle, degrees	180	180	180
Gear material	metal	metal	metal
Dimensions, mm	40 x 19 x 43	40 x 20 x 38.5	40.5 x 20.2 x 44.2
Mass, g	55	60	55,6
Price, euro	5,4	10,5	11,02

Although JX Servo PDI-5521MG shows better results than TowerPro MG996R, it was decided to choose the latter option as a wrist rotation actuator. Considered drives parameters were relatively same, as can be seen from the comparison chart, so decisive factor for choosing the TowerPro MG996R servo was relatively low price.

5.1.3 Elbow actuation

For elbow actuation the calculated torque that is needed to rotate the elbow joint is 5.86 N*m. The elbow joint should be able to change its position up to 90 degrees. According to this calculations it was decided to use servo drive with the bevel gearset, which gear ratio is 1:4. With this gear ratio, drive required torque should be more than 1,467 N*m, and, because of additional gear set usage is needed, the drive should be either continuous rotation, or should be able to travel up to 360 degrees. These requirements narrowed the possible actuator options to the following drives: Feetech FR5317M [15], DS3115 [16], AXADD S1500MD [17], which are shown on Appendix 2. Their comparison chart is presented in Table 5.2.

Table 5.2 Selected servo drives comparison chart

Name	Feetech FR5317M	DS3115	AXADD S1500MD
Operating voltage, V	4,8 – 8,4	4,8 ~ 8,4	4,8-6,0
Maximum torque , N*m	1,39 (4,8 V) 1,67 (7,4V)	1,37 (4,8 V) 1,47 (6,0 V)	1,39 (4,8 V) 1,48 (6 V)
Maximum speed	0,18 s / 60 ° (4,8 V) 0,14 s / 60 ° (6,0 V)	0,14 s /60 ° (4,8 V) 0,16 s /60 ° (7,2 V)	0,18 s /85 ° (4,8 V) 0,16 s /85 ° (6 V)
Maximum rotation angle, degrees	According to the specification, can work in continuous and standard rotation modes	Continuous rotation	Continuous rotation
Gear material	metal	metal	metal
Dimensions, mm	40,2 x 20 x 38	40 x 20 x 40,5	40 x 20 x 37
Mass, g	58	60	55
Price, euro	7,76	37,64	Available to order directly from manufacturer

Continuous rotation servo market is quite narrow, for required torque only a few servo drives were possible to implement as an elbow rotation actuator. The prices range is differs highly, as can be seen in comparison chart. After consideration, it was decided to use FeeTech FR5317M continuous rotation servo in elbow joint.

5.1.3 Arm-Elbow actuation

For arm-elbow joint, the required torque for arm rotation was calculated as 5,49 N*m. For this joint was decided to use standard spur gear set with gear ratio of 1:3, so the required motor torque had to exceed the 1,82 N*m torque. For this joint, continuous rotation servo should be used. Other way of implementing servo drive to this joint – is to modify it, by removing mechanical stoppers and placing angular sensor, directly to the rotation shaft, as was made in previous robotic arm system, to actuate most of the joints. Based on these statements, it was decided to use modified HiTech 805BB servomotor [18], which presented on Figure 5.2.

Figure 5.2 HiTech 805BB servomotor [12]

Parameters of HiTech 805 BB presented on the Table 5.3 below:

Table 5.3 – HiTech 805BB parameters

Dimensions	66 x 30 x 57,6 mm
Product mass	152 g
Voltage Range	4,8 – 6,0 V
No-Load Speed (4,8 V)	0,19 s /60 °

No-Load Speed (6,0 V)	0,14 s /60 °
Stall Torque (4,8 V)	1,94 N*m
Stall Torque (6,0 V)	2,42 N*m
Pulse Amplitude	3 - 5 V
Current Drain - idle (4,8 V)	8 mA
Current Drain - idle (6,0 V)	8,7 mA
Current Drain - no-load (4,8 V)	700 mA
Current Drain - no-load (6 V)	830 mA
Continuous Rotation Modifiable	Yes

Since previous version of the arm used this servomotor for almost all joints actuation, it is proved to be easy modifiable, this was the decisive factor of using this drive, instead of the others.

5.1.3 Shoulder-arm actuation

For shoulder-arm joint, calculated torque to rotate the shoulder was 13,7 N*m. For this rotation It wasn't possible to implement standard servo drive. The available space restrictions and high demanded torque, made it hard to use standard spur gearboxes. After considering available options, it was decided to use high stale torque step motor with planetary gearhead. NEMA 34 frame size step drive with 1:4 gear ratio planetary gear head was chosen as shoulder rotation actuator, because NEMA 34 standard step drives comes in wide variety of configurations and parameters. As the most suitable step drive, EM Motion Modules 3,43 N*m Stepper Motor [19] was chosen, it is presented on Figure 5.3.

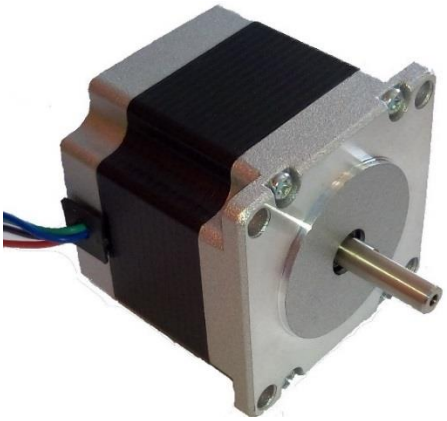


Figure 5.3 EM motion NEMA 34 step motor [19]

Parameters of this drive are presented below:

Operating voltage: 5,5 V, Dc

Operating current: 2 Amps

Step Angle – 1,8 degrees per step

Motor Power – 3,43 N*m

Holding Torque -– 3,43 N*m

Mass 1,7 kg

Estimated price: 40 euro

The planetary gearhead for this step motor appearance is presented on Figure 5.4 [20].

Figure 5.4 NEMA 34 1:4 planetary gearhead [20]

According to the gearhead manufacturer, it is possible to choose input shaft sizes, which ease the step drive installation process. The mass of this gearhead is 2,2 kg and estimated price – around 54,8 euro.

The main disadvantages of using step motors with planetary gearhead – weight of the actuation mechanism and usually step motors requires step motor driver in order to control it. According to the manufacturer of the step motor driver, as shown on Figure 5.5 can be used for control purposes.

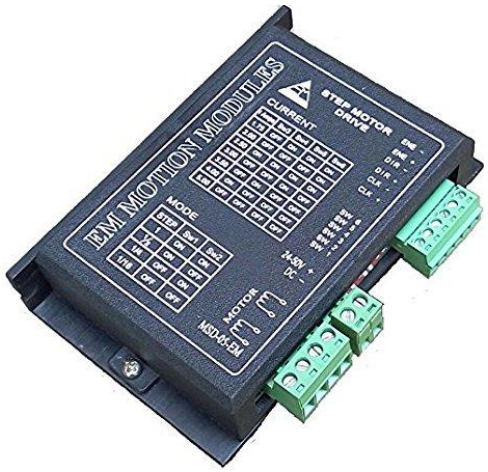


Figure 5.5 NEMA 34 step motor driver [21]

The driver's specifications are presented below [21]:

Operating Voltage: 24 - 48 VDC

Operating Current: 1,75 - 5 A

Operating Power: 240 W

Control Signal Voltage: 3-35 V

Control Signal Current Limit: 5-10 mA

Control Signal Isolation: 800 V Peak To Peak.

Price: around 38,4 euro

Required torque for shoulder-body rotation was defined as 12,87 N*m. The amount of required torque was nearly same, as for the shoulder rotation, so it was decided to use the same actuation mechanism, as was described in previous actuator selection process.

5.2 Sensor selection

In designed robotic arm two main parameters, that needs to be determined, by using different sensors, are angular position of joints and amount of pressure, that needs to be applied in order to hold an object. In described robotic arm, it was needed to implement rotary position sensor to the following joints: elbow rotation, arm rotation, shoulder rotation and body-shoulder rotation. Elbow rotation encoder should be able to detect rotation from up to 90 degrees, arm rotation – 180 degrees rotation, shoulder rotation – 310 degrees and for the body-shoulder rotation – 90 degrees. As for the pressure sensors, it was decided to attach them to the first, third and fifth fingers.

5.2.1 Angular positon sensor

In order to be able control positioning of the robotic arm, to every joint, where positioning control is not possible by the means of selected drive, additional position sensor should be installed. There are many different types of angular position sensor are presented on market nowadays, such as potentiometers, magnetic and optical encoders. Position encoder could be absolute or incremental, absolute encoder can detect its position straight at the moment, in which robot turns on, the incremental encoders defines the difference in position from the start moment to the actual position. In order to be able to find exact position, at each start of the system additional incremental encoder calibration should be made.

Potentiometers are the most common and inexpensive solution of all angular position sensors. They come in wide variety of shapes and sizes, which makes potentiometer-based angular sensor selection process easier. Basic potentiometer structure is shown on Figure 5.6 [22].

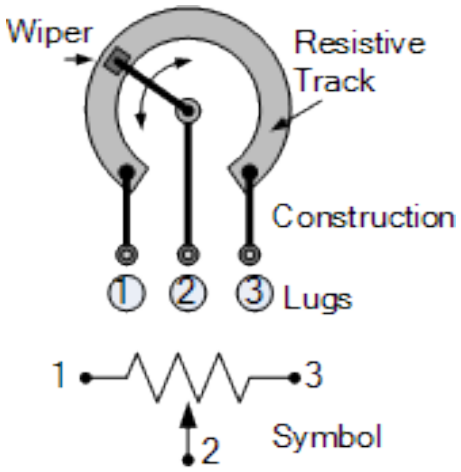


Figure 5.6 – Basic potentiometer structure [23]

It can detect angular position of the shaft that is fixed to the wiper, which, by moving on resistive track, changes resistance between first and third lugs, and by the difference in voltage, position can be determined. Since measured angle depends only on position of wiper, potentiometers can be used as absolute angle measurement sensors. The main disadvantage of absolute potentiometer-based angular sensors is fixed operating range.

Optical encoders uses light from an LED or infra-red light source, which is passed through a rotating high-resolution encoded disk that contains the required code patterns, either binary, grey code or BCD. Photo detectors scan the disk as it rotates and an electronic circuit processes the information into a digital form as a stream of binary output pulses that are fed to counters or controllers which determine the actual angular position of the shaft. The appearance of such encoded disk is presented on Figure 5.7 [22].

Figure 5.7 – Optical position sensor encoded disk appearance [22]

Optical encoders can be absolute or incremental. In absolute encoders, each position encoded by unique pattern on the disk. Optical encoders are quite expensive and very sensitive to the environmental influence.

In magnetic encoders a large magnetized wheel spins over a plate of magneto-resistive sensors, the wheel causes predictable responses in the sensor, based on the strength of the magnetic field. The magnetic response is fed through a signal conditioning electrical circuit, through which encoder position can be defined [24]. The working principle is shown on Figure 5.8.

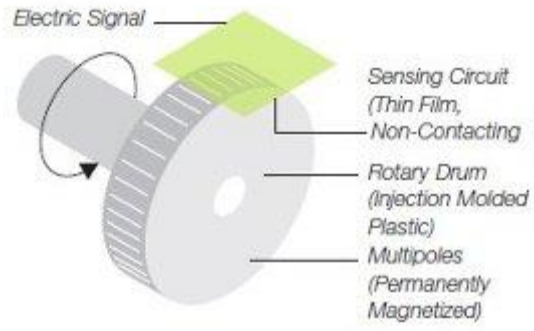


Figure 5.8 – Magnetic encoder working principal [24]

The main advantage of the magnetic encoders over optical, is that environmental influence is minimized. Magnetic encoders can be absolute and incremental, depending on the wheel construction.

The selection process of the angular sensor can be conducted by the main parameters: supply voltage, output type, sensing range, operating temperature range, packaging type and supply current [25]. After considering pros and cons of each sensor type, for the elbow, arm and shoulder-body rotation it was decided to use a rotation potentiometer as angular position sensors. For the shoulder-rotation, due to complicity of the regular shaft potentiometer installation, it was decided to use potentiometric angular sensor with hollow shaft. The potentiometers were chosen also because of the simple data requisition, since they required only two analog ports from the microcontroller board. The main parameter, by which potentiometers were selected was operating range, since the requirements for desired rotation were set by the designing task.

As the regular shaft potentiometer following options were considered:

1) Seeed Studio Grove - Rotary Angle Sensor [26]

2) SPARKFUN Rotary Potentiometer - 10k Ohm [27]

3) Conductive Plastic Potentiometer MP20/21 [28]

Their appearance presented on Appendix 4, and their comparison chart is presented on Table 5.4.

Table 5.4 – Potentiometer comparison chart

Parameters	SEEED rotary potentiometer	SPARKFUN rotary potentiometer	Megatron conductive plastic potentiometer MP 20/21
Maximum rotation angle	300 ± 10 °	300 ± 10 °	320 ± 5 °
Resistance value	10 kOhms	10 kOhms	100 kOhms
Shaft diameter	6 mm	6 mm	6 / 6.35 mm
Rotation stopper maximum torque	Not mentioned in the data sheet	0,3 N*m	0,6 N*m
Price	2,4 euro	1 euro	≈5 euro

Based on this comparison chart, MP 20 potentiometer from Megatron-Munich was chosen.

As for the hollow shaft potentiometer, suitable option were found from novotechnik manufacturer. The selection process were conducted based on the manufacturer data sheet [29]. As a result, novotechnik GL200 potentiometer was chosen, its appearance is presented on Figure 5.9.

Figure 5.9 – Novotechnik GL-series hollow shaft potentiometer [29]

The decisive factor in choosing this potentiometer manufacturer was possibility to order potentiometer with needed inner diameter, which is going to ease the assembly process of the sholder. GL200 main parameters are presented below:

Maximum angle: 340 degrees

Electrical resistance: 10 kOhm

Recommended wiper current: ≤ 1 μA

5.2.2 Pressure sensor

For the pressure sensing task**,** due to specific shape of installation place, it was decided to use thin analog pressure sensors, which appearance presented on Figure 5.10. [30]

Figure 5.10 Resistive pressure sensor [30]

This types of sensors changes its resistance, according to the pressure, applied to a tip, and returns analog value, by which amount of applied force could be define.

5.3 Control device selection

The control of the manipulator can be divided into two main parts: positioning control and direct actuators and sensors control. For positioning control task, it was decided to use a PC, since most of the positioning control were made, using MatLab software. It is also possible to replace the PC for single-board computer, in order to create autonomous robot, but in arm designing task creation of autonomous system were not required.

To control actuators and sensors, there are several options: build system with one controller, that would perform all controlling tasks, or use multiple microcontrollers instead. The first approach demands high-perfomance control system. Usually such control devices are expensive and would be hard to replace in case of malfunction. The second approach is to use multiple controllers, each responsible for their own tasks, connected to the main controller, which is more suitable approach, in case of complex system designing process.

5.3.1 Actuators and sensors controller

After possible option for microcontroller boards selection, Arduino MEGA controller was chosen as motors and sensors controller, which appearance presented on Figure 5.11 [31].

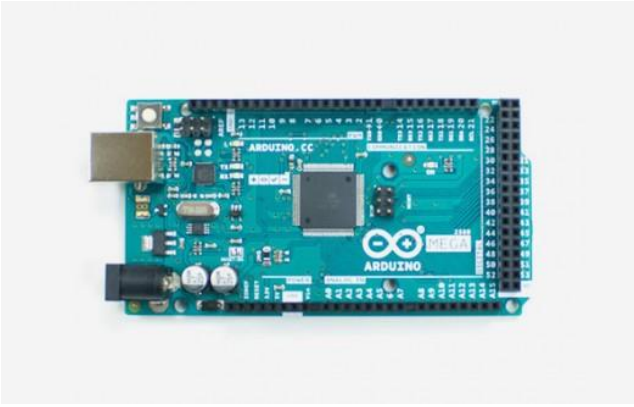


Figure 5.11 Arduino mega 2560 rev3 microcontroller board [31]

Technical specification of chosen microcontroller board are presented in Table 5.5 below:

Table 5.5 Arduino mega 2560 rev3 specifications

Microcontroller	ATmega2560
Operating Voltage	5 V
Input Voltage (recommended)	7-12 V
Input Voltage (limit)	6-20 V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED_BUILTIN	13
Length	101,52 mm
Width	53,3 mm
Mass	37 g

The amount of digital and analog inputs makes this controller viable solution for control task. Arduino microcontroller boards are widely used in different applications: robotics, RC vehicles, research purposes e.t.c, which are usually open-sourced, that provides vast amount of useful libraries and solutions, that can be implemented in designed robotic arm. Arduino microcontroller board can be paired with MatLab, using specific MatLab library, which would help in controlling the built arm, according to MatLab calculations and simulation.

Since analog pressure sensors were planned to use, the additional controller for transferring analog signal to digital had to be installed on the wrist, to avoid noise occurrences during analog data transmission. For analog to digital conversion, the microcontroller does not need to be high performance. In order to ease the process of controllers communication, the controller form Arduino series were chosen – Arduino Nano, which is presented on Figure 5.12 [32].

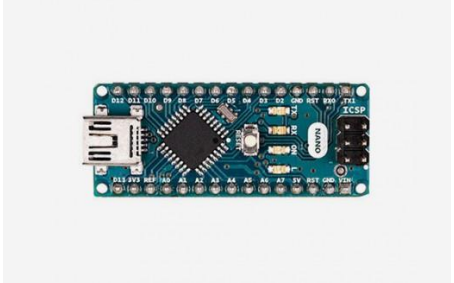


Figure 5.12 Arduino NANO microcontroller [32]

Arduino NANO specifications are presented below in Table 5.6:

Table 5.6 Arduino NANO specifications

Microcontroller	ATmega328
Architecture	AVR
Operating Voltage	5 V
Flash Memory	32 KB of which 2 KB used by bootloader
SRAM	2 KB
Clock Speed	16 MHz
Analog IN Pins	8
EEPROM	1 KB
DC Current per I/O Pins	40 mA (I/O Pins)
Input Voltage	7-12 V
Digital I/O Pins	22 (6 of which are PWM)
PWM Output	6
Power Consumption	19 mA
PCB Size	18 x 45 mm
Mass	7 g

5.4 Power supply selection

After all electrical component selection was conducted, it was needed to calculate the amount of power, that is needed to supply whole robotic arm system. Electric power can be found as stated below:

$$P = V * I \text{ [3.1]}$$

Where V – operating voltage, I – operating current. In order to find the resulting power consumption of the whole system, electric power of each component should be summed. Since

sensors receive their power supply straight from the microcontroller boards, their power consumption can be calculated as a power, which microcontroller board consumes. Power consumption chart was composed, in order to calculate required power supply, as presented on Table 5.7:

Table 5.7 Robotic arm's components required power supply chart

Component's name	Operating voltage, V	Operating current, A	Electrical power, W
5 x HK15298B	7,4	1,5	11,1
TowerPro MG996R	6	2,5	15
FeeTech FR5317M	7,4	1,5	11,1
HiTech 805BB	6	0,9	5,4
2 x NEMA 34 step driver	24	2	48
Arduino MEGA	7 - 12	0,9	10,8
Arduino Nano	5	0,5	2,5

According to composed table, the selected power supply should provide more than 196.3 W of electrical power. 24V 17A 360W power supply was chosen as suitable for the task, it is shown on Figure 5.13 [33].

Figure 5.13 AC 110-240V Input To DC 24V 17A 360W power supply [33]

In order to be able to supply less consuming components, additional 24V into 7,5V convertor is needed, the suitable option is presented on Figure 5.14 [34].

Figure 5.14 24 VDC to 7,5 VDC converter [34]

6. CONTROL DESIGN

6.1 SolidWorks into SimScape model conversion

Final version of the designed robotic arm was imported into SimScape, using SMimprot toolbox for MatLab. After import process, Simulink model was created, as shown on Figure 6.1. Using this model, movement of the robotic arm can be simulated. Each link connected to each other by revolute joint, that can only rotates around one of the axis. In order to be able to control movement of each joint, angle control subsystem, that consists of Joint Actuators and simulation parameters , were attached to each revolute joint. Joint actuator controls joint parameters, which type was defined in the revolute joint block. For this task, control of angle, angular velocity and acceleration were chosen as control parameters. Velocity and acceleration were set manually, since limits for these two parameters, were not determined by the designing task. To control angle of rotation at each joint, by knowing coordinates of the desired point, which represents arm desired destination, inverse kinematics task for humanoid arm manipulator should be solved.

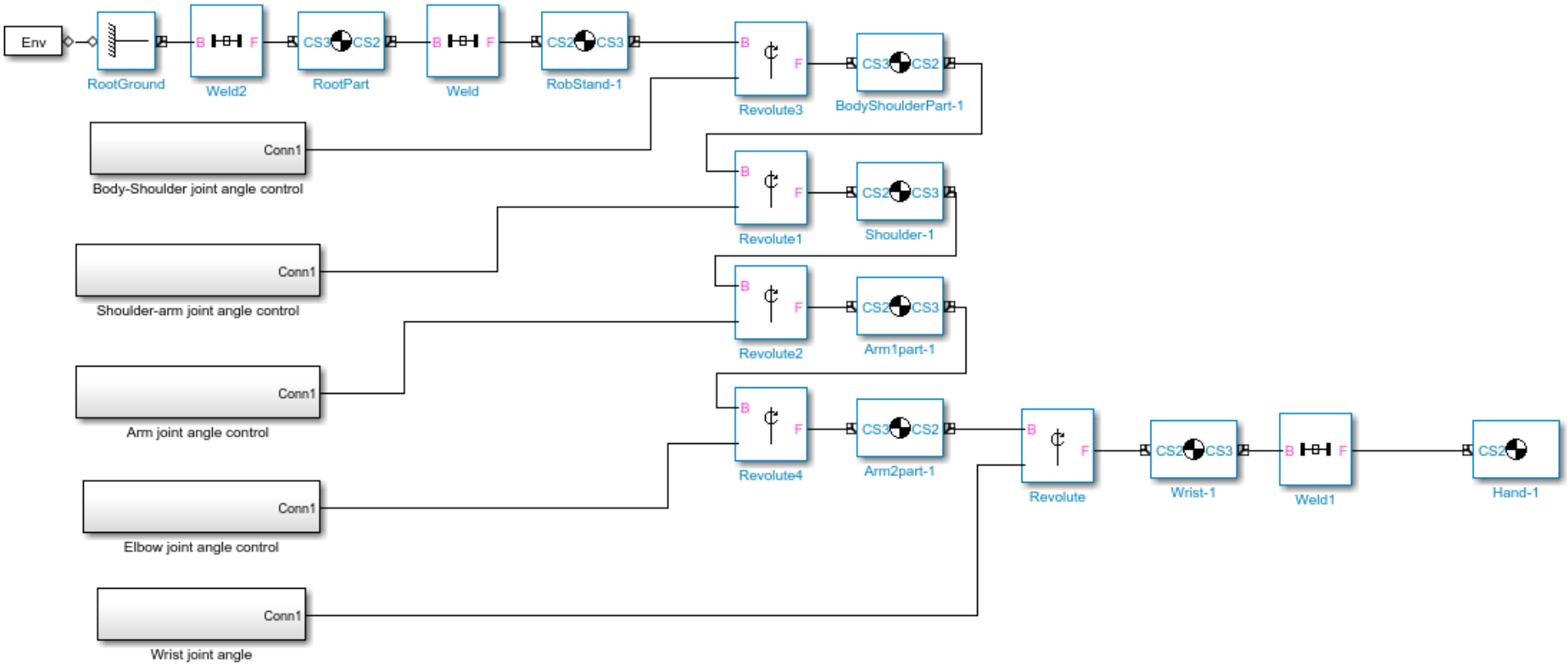


Figure 6.1 Imported SolidWorks model into Simulink

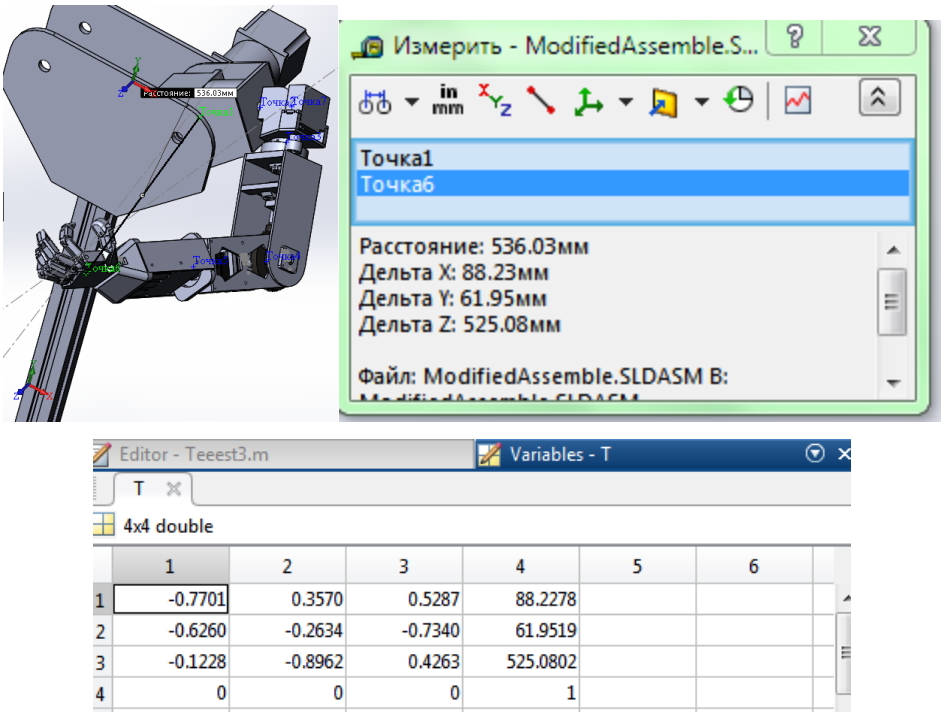
6.2 Forward kinematics

In order to solve mechanism inverse kinematics, forward kinematics task should be solved first. There are several available ways to do so: geometrical, through translation matrices, by finding Denavit-Hartenberg parameters, mathematical way and so on. For this task, it was decided to find end coordinates, using translation matrices in MatLab software. For each rotary joint translation matrix in relation to the previous joint was composed. as presented on Figure 6.2.

```
t1=[cos(alp1) -sin(alp1) 0 a0;
    sin(alp1) cos(alp1) 0 0;
    0 0 1 0;
    0 0 0 1];

t2=[1 0 0 a1;
    0 cos(alp2) -sin(alp2) d1;
    0 sin(alp2) cos(alp2) 0;
    0 0 0 1];

t3=[cos(alp3) 0 sin(alp3) a2;
    0 1 0 -d2;
    -sin(alp3) 0 cos(alp3) 0;
    0 0 0 1];

t4=[1 0 0 0;
    0 cos(alp4) -sin(alp4) -d3;
    0 sin(alp4) cos(alp4) 0;
    0 0 0 1];

t5=[cosd(alp5) 0 sind(alp5) 0;
    0 1 0 -d4;
    -sind(alp5) 0 cosd(alp5) 0;
    0 0 0 1];
t6=[1 0 0 0;
    0 1 0 -d5;
    0 0 1 0;
    0 0 0 1];

T=t1*t2*t3*t4*t5*t6;
```

Figure 6.2 Translation matrices for solving forward kinematics task

The resulting coordinates, based the given angles, were proved to be correct, using SolidWorks measurement tool. For example: for determined angles of rotation alp1, alp2, alp3, alp4, alp5, comparison of the results are presented on Figure 6.3.



	1	2	3	4	5	6
1	-0.7701	0.3570	0.5287	88.2278		
2	-0.6260	-0.2634	-0.7340	61.9519		
3	-0.1228	-0.8962	0.4263	525.0802		
4	0	0	0	1		

Figure 6.3 – Proof of correct forward kinematics solution

As can be seen from Figure 6.3, the coordinates of the end effector with respect to the base coordinate axis, were determined correctly.

6.3 Inverse kinematics problem statement

The inverse kinematics task was solved, based on the translation matrices, composed as forward kinematics solution. In MatTab software, angles, that had to be determined, were defined as simulated character objects. In order to be able to solve resulted translation matrix for 5 DoF robotic arm, it was needed to define at least two angles of rotation. Wrist rotation angle can be determined, according to the payload position with respect to the base coordinates, which the object recognition device should define. As a second angle, it was decided to determine rotation angle of shoulder relative to the body. This angle was determined with the use of MatLab neural network toolbox.

6.4 Neural network-based solution for inverse kinematics task

6.4.1 Neural networks description

Nowadays artificial neural network are used in robotics for various tasks solution. Artificial neural networks are called this way, because the structure of NN resembles the structure of organic brain. Neural networks uses hidden layers, which are consists of nodes – artificial neurons, in order to define patterns between input and output data. NN can be classified several ways. By learning type neural networks can be supervised, when training based on known inputs and corresponding outputs, provided by user – teacher, unsupervised, when training is conducted only based on known inputs, without knowing outputs, and reinforcement, when environment, in which neural network controlled process is proceeds, takes the role of the teacher and defines if neural network answer is correct or not. By the connection type between neurons NN can be feed-forward, when connection of neurons goes strictly from inputs to outputs, recurrent, when data, computed on layer, sent back to the inputs, in order to stabilize output data [35].

In robotics application, neural networks used for control task, image processing, pattern recognition, e.t.c.

The easiest neural networks type to built, tune and teach are feed-forward NN with supervised learning, which were used in this work in order to prove, that neural network-based inverse kinematics solution is possible.

The most time-consuming part of neural network creations is training, but when NN is trained, the answer response is rather high, which could help conducting control task on low-performance systems.

The decision to use neural networks in the inverse kinematics task were inspired by the scientific article, which describes the inverse kinematics solution for humanoid robotic arm [36].

6.4.2 Neural network training data sets composition and parameters selection

During neural network parameters selection, two different training algorithm were tested: Levenberg-Marquardt (trainlm) and Variable Learning Rate Backpropagation (traingdx). For these two algorithms two methods of training data acquisition were used. The first method implied using randomly generated angles for solved forward kinematics, as shown on Figure 6.4,

```
for i=1:5000
alp1=90*rand;
alp2=-180+220*rand;
alp3=90+180*rand;
alp4=90*rand;

t1=[cosd(alp1) -sind(alp1) 0 a0;
    sind(alp1) cosd(alp1) 0 0;
    0 0 1 0;
    0 0 0 1];

t2=[1 0 0 a1;
    0 cosd(alp2) -sind(alp2) d1;
    0 sind(alp2) cosd(alp2) 0;
    0 0 0 1];

t3=[cosd(alp3) 0 sind(alp3) a2;
    0 1 0 -d2;
    -sind(alp3) 0 cosd(alp3) 0;
    0 0 0 1];

t4=[1 0 0 0;
    0 cosd(alp4) -sind(alp4) -d3;
    0 sind(alp4) cosd(alp4) 0;
    0 0 0 1];

t5=[cosd(alp5) 0 sind(alp5) 0;
    0 1 0 -d4;
    -sind(alp5) 0 cosd(alp5) 0;
    0 0 0 1];

t6=[1 0 0 0;
      0 1 0 -d5;
      0 0 1 0;
      0 0 0 1];

T=t1*t2*t3*t4*t5*t6;

 ang(i,:)=[alp1; alp2 ;alp3; alp4];
 crd(i,:)=[T(1,4); T(2,4); T(3,4)];

end
```

Figure 6.4 First method of training data requisition

The second method was based on consecutive forward kinematics solution, for possible joint angles, as presented on Figure 6.5.

```
for alp1=0:b:90
    for alp2=-180:b:40
        for alp3=90:b:270
            for alp4=0:b:90

t1=[cosd(alp1) -sind(alp1) 0 a0;
    sind(alp1) cosd(alp1) 0 d0;
    0 0 1 k0;
    0 0 0 1];

t2=[1 0 0 a1;
    0 cosd(alp2) -sind(alp2) d1;
    0 sind(alp2) cosd(alp2) 0;
    0 0 0 1];

t3=[cosd(alp3) 0 sind(alp3) a2;
    0 1 0 -d2;
    -sind(alp3) 0 cosd(alp3) 0;
    0 0 0 1];
```

Figure 6.5 Second method of training data requisition

For both methods, corresponding coordinates of end effector has been found. Angles and coordinates then were collected intro ang and crd arrays. It is possible to change number of training sets, by changing the amount of cycle iterations for the first method, or by modifying step size for the second.

6.4.3 Neural network training process description

On two arrays of training data, described in 6.4.2, feed-forward neural networks were created and trained with trainlm and traingdx training algorithms, with the neural network structure as shown on the Figure 6.6.

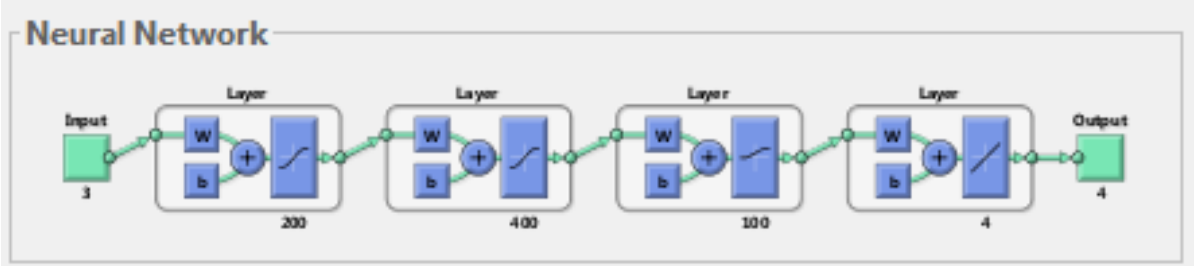


Figure 6.6 Used neural network structure

For training, TTU mechatronics department's laboratory computer was used. PC technical parameters presented below:

Processor: Inter Core i5 - 4590 3,3 GHz

RAM: 8 GB

System type: 64-bit operation system

On this PC training time for trainlm algorithm was 3 days, and learning process was stopped, because after that time training performance were improving slowly, which means that neural network stopped learning. For traingdx algorithm, training took around 5 hours.

The best mean square error performance result with trainlm algorithm was 23.9, with traingdx – around 105. Tangential sigmoid, and logarifmic sigmoid training functions were used, during training process.

6.4.4 Trained neural network testing

After learning process was finished, for manually selected desired coordinates, angles were found, using neural networks, then, in order to check accuracy of determination, found angles were implemented in forward kinematics solution, and it was found, that desired point coordinates and coordinates, based on NN-found angles, does not match. As described in 6.3 , it was decided to determine one angle of rotation, using neural networks, and others, by solving matrices of translation with set coordinates. After test runs of neural network angle determination, it was found that neural network finds some angles better than the others. By using MatLab code, presented in Appendix 1, the most accurate definable angle was found, it was the first body-shoulder rotation joint angle. Then, using that angle, the rest parameters for inverse kinematics solution were found, using MatLab vpasolver, as shown on Figure 6.7.

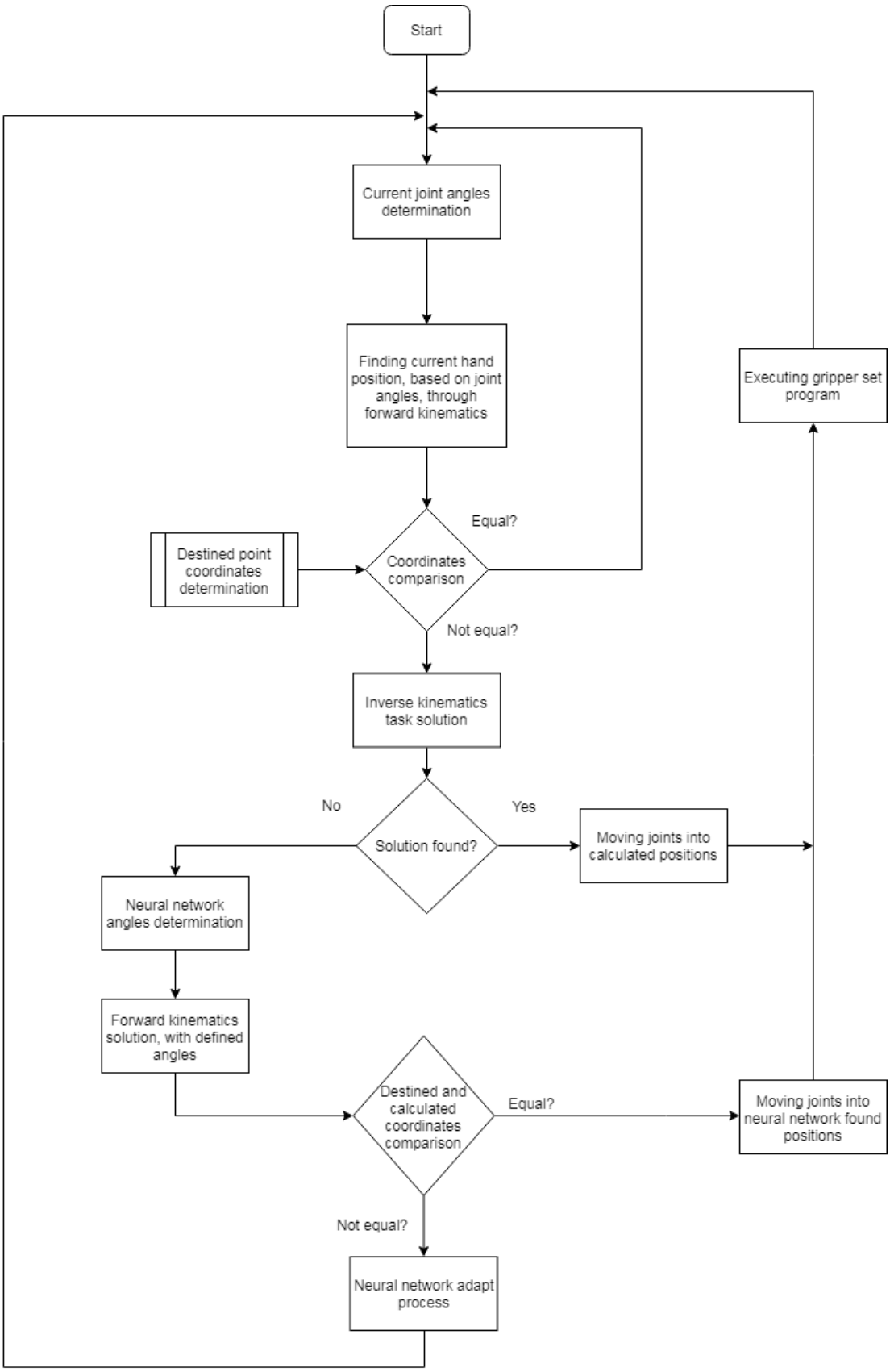
```
syms t2(alp2) t3(alp3) t4(alp4)  x y z

x = T(1,4) == xN;
y = T(2,4) == yN;
z = T(3,4) == zN;

S=vpasolve([x, y, z], [alp2, alp3, alp4],[-185*pi/180 45*pi/180;85*pi/180 275*pi/180;0 95*pi/180]);

% S1=S.alp1
S2=S.alp2;
S3=S.alp3;
S4=S.alp4;
% S5=S.alp5
alp1=alp1*180/pi
alp2 = double(S2*180/pi)
alp3 = double(S3*180/pi)
alp4 = double(S4*180/pi)
```

Figure 6.7 - Inverse kinematics solution

After solving inverse kinematics task, robotic arm movement simulation was created, using SimScape model, described in 1.4.1 and angle parameters, determined as inverse kinematics solution.

After set of simulation test runs, was found that solver could not find solution for some points. The problem was that MatLab solver could not define angles, for non-integer fixed angle parameters, in order to solve this problem, neural network defined angle value were rounded to closest integer.

6.5 ARM CONTROL FLOW CHART



7. FURTHER WORK

As a further improvement for designed arm following goals were stated:

1) Neural network-based inverse kinematics solution proved to be possible end-effector positioning control algorithm, but to increase the accuracy of angle determination, further tuning of neural network training process should be conducted. It is also possible to use different learning and structure types of neural networks.

2) Further work on reducing arms weight could be done, by choosing different actuation methods for elbow joint rotation, or by choosing gearboxes with different parameters. In this work, chosen gearsets were regulated by the gear ratio demands, so in future, with the improvements on the field of robotic actuators, it is possible to reduce gears parameters, or remove them completely.

3) Hand actuation could be changed as well, in order to increase the amount of weight, that arm could carry.

8. SUMMARY

During this work literature and theory review was made. Based on review, designing task was composed. According to the designing task five degree of freedom humanoid robotic arm mechanical design solution was proposed. The newly designed arm consists of stand, shoulder-body, shoulder-arm, elbow, forearm, wrist and hand parts, which were modeled and calculated in SolidWorks CAD software. Electrical components, that newly designed arm consists of, are chosen and their selection process was described. The hull weight of the designed arm was reduced. The actuation method of elbow joint was modified with the use of bevel gear set. According to calculations, lifting weight was increased, in comparison to previous robotic arm version. Goals for further improvement were stated. Neural network-based solution for inverse kinematics task for arm position control was proposed. On two differently acquired training sets of data, that was consists of angles and corresponding coordinates, two supervised feed-forward neural network was created and trained, using Levenberg-Marquardt (trainlm) and Variable Learning Rate Backpropagation (traingdx) training algorithms, and their performance was evaluated, using mean square error. Then, a series of test runs were conducted, from where It was found that using only neural network for inverse kinematics solution is not enough, since angles, that defined by neural network are not leading end-effector into destined point. In order to solve this problem, different method of required angle calculation was proposed: translation matrix, that was defined by forward kinematics solution, was solved, using MatLab vpasolver, which was required setting of one of four calculated angles, in order to solve system, that consists of three equations for x, y and z coordinates. The simulation model was created, using MatLab add-on for translation SolidWorks model into SimScape model. Data, retrieved from the inverse kinematics solution, was later used in creation of designed arm movement simulation.

Neural network-based control proved to be viable solution for robotic manipulator control task, by changing its structure, training types and algorithms, accuracy, compared to the mathematical model-based control could be achieved. The use of neural networks ease the positioning control task, since only forward kinematics solution is needed. With the rapid growth of technologies, it would be possible to reduce training time of neural networks, which would results in possibility of trying different NN configuration, and eventually in increasing control accuracy.

REFERENCES

1. SoftbankRobotics, "Pepper humanoid robot," SoftbankRobotics, 03 08 2017. [Online]. Available: https://www.softbankrobotics.com/emea/en/robots/pepper. [Accessed 20 04 2018].

2. SoftbankRobotics, "Pepper robot technical specifications," SoftbankRobotics, 03 08 2017. [Online]. Available: http://doc.aldebaran.com/2-4/family/pepper_technical/index_pep.html. [Accessed 20 04 2018].

3. G. Langevin, "InMoov 3D-printed open source life-size humanoid robot," Gael Langevin, 01 01 2012. [Online]. Available: http://inmoov.fr/. [Accessed 20 04 2018].

4. A. M. R. P. B. L. S. I. &. P. S. Zanchettin, "Kinematic analysis and synthesis of the human arm motion during a manipulation task.," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on (pp. 2692-2697). IEEE.*, 2011.

5. L. J. Klopčar N., "Kinematic model for determination of human arm reachable workspace," *Meccanica,* vol. 40, no. 2, pp. 203-219, 2005.

6. 3D Today, "State of art FDM-technology materials description," 3D today, 2018. [Online]. Available: http://3dtoday.ru/wiki/FDM_materials/. [Accessed 15 05 2018].

7. E. T. S. Ebel, "Fabrication of FDM 3D objects with ABS and PLA and determination of their mechanical properties.," RTejournal, 2014.

8. S. E. Z. G. S. V. Politavkin A.M., Зубчатые передачи в приборах. Учебное пособие, ЛИТМО, 1985.

9. KHK-gears, "Geometrical gear calculation methods," KHK-gears, 2013-2017. [Online]. Available: https://khkgears.net/new/gear_knowledge/gear_technical_reference/calculation_gear_dimensions.html. [Accessed 14 04 2018].

10. L. a. C. M. G. Birglen, "Kinetostatic analysis of underactuated fingers.," *IEEE Transactions on Robotics and Automation,* vol. 20, no. 2, pp. 211-221, 2004.

11. HobbyKing, "HK15298B High Voltage Coreless Digital MG/BB Servo specifications," HobbyKing, 2018. [Online]. Available: https://hobbyking.com/en_us/hobbykingtm-coreless-digital-hv-mg-bb-servo-20kg-0-16sec-66g.html. [Accessed 15 04 2018].

12. Electronicoscaldas, "MG995R TowerPro specifications," Tower Pro, 2018. [Online]. Available: http://www.electronicoscaldas.com/datasheet/MG996R_Tower-Pro.pdf. [Accessed 15 04 2018].

13. Firelap, "Firelap LS-RS150M servo specifications," Firelap, 2018. [Online]. Available: https://www.alibaba.com/product-detail/180-degree-robot-servo-60g-13kg_60573910836.html?spm=a2700.9099375.35.8.11c84be7rcwPjD . [Accessed 15 04 2018].

14. JX Servo, "JX Servo PDI-5521MG specifications," JX Servo, 2018. [Online]. Available: https://www.banggood.com/JX-Servo-PDI-5521MG-20KG-Metal-Gear-Digital-Standard-Servo-180-Degree-p-1064604.html?rmmds=buy&cur_warehouse=CN. [Accessed 15 04 2018].

15. FeeTech, "FeeTech FR51317M servo motor specifications," FeeTech, 2018. [Online]. Available: https://feetechrc.en.alibaba.com/product/60234260429-801405821/FEETECH_FR5317M_17kg_Metal_gear_high_Torque_Robot_servo.html . [Accessed 15 04 2018].

16. DS, "DS3115mg servo motor specifications," DS, 2018. [Online]. Available: https://ru.aliexpress.com/item/Freeship-servo-360-degree-Continuous-Rotation-Servo-DS3115-Metal-gear-arduino-servo-Digital-servo-15kg-cm/725132391.html. [Accessed 14 04 2018].

17. AXADD, "AXADD S1500MD servo drive specifications," AXADD, 2018. [Online]. Available: http://www.axadd.com/products/digital-servos/robot-servos/553.html. [Accessed 15 04 2018].

18. HiTech, "HiTech 805BB servo drive datasheet," HiTech, 2018. [Online]. Available: https://cdn.sparkfun.com/datasheets/Robotics/hs805.pdf . [Accessed 15 04 2018].

19. EM motion, "EM Motion NEMA 34 step motor specifications," EM motion, 2018. [Online]. Available: https://www.amazon.in/EM-Motion-Modules-35KG-Cm-Stepper/dp/B01CJOAHFM . [Accessed 15 04 2018].

20. G-Penny machine, "NEMA 34 planetary gearhead specifications," G-Penny machine, 2018. [Online]. Available: https://ru.aliexpress.com/item/Gearbox-1-4-Nema34-stepper-motor-planetary-reduction-Ratio-4-1-planet-gearbox-86-motor-speed/32848719871.html . [Accessed 15 04 2018].

21. EM motion, "NEMA 34 driver specification," EM motion, 2018. [Online]. Available: goo.gl/EYm6a3. [Accessed 15 04 2018].

22. ElectronicsTutorials, "Position sensors classification," ElectronicsTutorials, 2014. [Online]. Available: https://www.electronics-tutorials.ws/io/io_2.html. [Accessed 20 04 2018].

23. ElectronicsTutorials, "Basic potentiometer work principle description," ElectronicsTutorials, 2014. [Online]. Available: https://www.electronics-tutorials.ws/resistor/potentiometer.html . [Accessed 20 04 2018].

24. Dynapar, "Magnetic encoder description," Dynapar, 2018. [Online]. Available: https://www.dynapar.com/technology/encoder_basics/magnetic_encoder/ . [Accessed 20 04 2018].

25. FutureElectronics, "Angular position sensors description," FutureElectronics, 2018. [Online]. Available: http://www.futureelectronics.com/en/sensors/angular-position.aspx . [Accessed 20 04 2018].

26. Exp-tech, "SEEED-studio rotary angle sensor," SEEED studio, 2016. [Online]. Available: https://www.exp-tech.de/en/platforms/platforms/seeed-studio-grove/module/4361/seeed-studio-grove-rotary-angle-sensor. [Accessed 21 04 2018].

27. Exp-tech, "SPARKFUN rotary potentiometer," SPARKFUN, 2016. [Online]. Available: https://www.exp-tech.de/en/accessories/passive-components/7312/rotary-potentiometer-10k-ohm-linear . [Accessed 21 04 2018].

28. Megatron, "Conductive Plastic Potentiometer MPA20/21," Megatron, 2018. [Online]. Available: https://www.megatron.de/en/products/conductive-plastic/conductive-plastic-potentiometer-mpa2021.html . [Accessed 22 04 2018].

29. Novotechnik, "GL hollow shaft potentiometer data sheet," Novotechnik, 2018. [Online]. Available: http://www.novotechnik.com/pdfs/GL.pdf. [Accessed 22 04 2018].

30. ICStation, "Thin film pressure sensors specification," ICStation, 2018. [Online]. Available: http://www.icstation.com/100kg-thin-film-pressure-sensor-force-sensor-diameter-16mm-arduino-p-11690.html. [Accessed 25 04 2018].

31. Arduino, "Arduino mega microcontroller board specifications," Arduino, 2018. [Online]. Available: https://store.arduino.cc/usa/arduino-mega-2560-rev3 . [Accessed 03 05 2018].

32. Arduino, "Arduino nano specifications," Arduino, 2018. [Online]. Available: https://store.arduino.cc/usa/arduino-nano . [Accessed 03 05 2018].

33. Banggood, "24 VDC power supply description," 2018. [Online]. Available: https://www.banggood.com/AC-110-240V-Input-To-DC-24V-17A-360W-Switching-Power-Supply-Driver-Board-p-1272112.html?rmmds=buy&stayold=1&cur_warehouse=CN. [Accessed 09 05 2018].

34. Fulree, "24VDC to 7.5 VDC convertor," Fulree, 2018. [Online]. Available: https://www.alibaba.com/product-detail/Fulree-Small-Size-12V-24V-to_60669705529.html?spm=a2700.7724857.main07.41.42af206behuwtf . [Accessed 06 05 2018].

35. Y. B, Artificial neural networks., PHI Learning Pvt. Ltd., 2009.

36. A.-V. Duka, "Neural network based inverse kinematics solution for trajectory tracking of a robotic arm.," *Procedia Technology ,* no. 12, pp. 20-27, 2014.

APPENDICES

Appendix 1 – Wrist actuation servo drive options

Towerpro MG996R [12]

Firelap LS-RS150M [13]

JX Servo PDI-5521MG [14]

Feetech FR5317M [15]

DS3115 [16]

AXADD S1500MD [17]

Seeed Studio Grove - Rotary Angle Sensor [26]

SPARKFUN Rotary Potentiometer - 10k Ohm [27]

Conductive Plastic Potentiometer MP20/21 [28]

```
a0=0.6175;
d0=1.170;
k0=0.5;
a1 = 0.115;
d1 = 0.015;
a2 = 0.05697;
d2 = 0.05519;
d3 = 0.1878;
d4 = 0.160;
d5 = 0.23417;

comV1=0;
comV2=0;
comV3=0;
comV4=0;

for i=1:5000
compK=99999;
comp1=100000;
comp2=100000;
comp3=100000;
comp4=100000;

alp1=90*rand;
alp2=-180+220*rand;
alp3=90+180*rand;
alp4=90*rand;

t1=[cosd(alp1) -sind(alp1) 0 a0;
    sind(alp1) cosd(alp1) 0 d0;
    0 0 1 k0;
    0 0 0 1];
t2=[1 0 0 a1;
    0 cosd(alp2) -sind(alp2) d1;
    0 sind(alp2) cosd(alp2) 0;
    0 0 0 1];
t3=[cosd(alp3) 0 sind(alp3) a2;
    0 1 0 -d2;
    -sind(alp3) 0 cosd(alp3) 0;
    0 0 0 1];
t4=[1 0 0 0;
    0 cosd(alp4) -sind(alp4) -d3;
    0 sind(alp4) cosd(alp4) 0;
    0 0 0 1];
t5=[cosd(alp5) 0 sind(alp5) 0;
    0 1 0 -d4;
    -sind(alp5) 0 cosd(alp5) 0;
    0 0 0 1];
t6=[1 0 0 0;
      0 1 0 -d5;
      0 0 1 0;
      0 0 0 1];

T=t1*t2*t3*t4*t5*t6;

s=sim(nn,[T(1,4); T(2,4); T(3,4)]);
```

```
comp1=abs(s(1,1)-alp1);
comp2=abs(s(2,1)-alp2);
comp3=abs(s(3,1)-alp3);
comp4=abs(s(4,1)-alp4);

if comp1<compK
    compK=comp1;
end
if comp2<compK
    compK=comp2;
end
if comp3<compK
    compK=comp3;
end
if comp4<compK
    compK=comp4;
end
if comp1==compK
    comV1=comV1+1;
end
if comp2==compK
    comV2=comV2+1;
end
if comp3==compK
    comV3=comV3+1;
end
if comp4==compK
    comV4=comV4+1;
end
end
compAA=[comV1;comV2;comV3;comV4];
```

```
a0=0.6175;
d0=1.170;
k0=0.5;
a1 = 0.115;
d1 = 0.015;
a2 = 0.05697;
d2 = 0.05519;
d3 = 0.1878;
d4 = 0.160;
d5 = 0.23417;
% alp1 = 30;
% alp2 = -40;
% alp3 = 180-40;
% alp4 = 60;
alp5 = 0;
b=10;
ang=zeros(1,4);
crd=zeros(1,3);
for i=1:8000
alp1=90*rand;
alp2=-180+220*rand;
alp3=90+180*rand;
alp4=90*rand;

t1=[cosd(alp1) -sind(alp1) 0 a0;
    sind(alp1) cosd(alp1) 0 d0;
    0 0 1 k0;
    0 0 0 1];
t2=[1 0 0 a1;
    0 cosd(alp2) -sind(alp2) d1;
    0 sind(alp2) cosd(alp2) 0;
    0 0 0 1];
t3=[cosd(alp3) 0 sind(alp3) a2;
    0 1 0 -d2;
    -sind(alp3) 0 cosd(alp3) 0;
    0 0 0 1];
t4=[1 0 0 0;
    0 cosd(alp4) -sind(alp4) -d3;
    0 sind(alp4) cosd(alp4) 0;
    0 0 0 1];
t5=[cosd(alp5) 0 sind(alp5) 0;
    0 1 0 -d4;
    -sind(alp5) 0 cosd(alp5) 0;
    0 0 0 1];
t6=[1 0 0 0;
      0 1 0 -d5;
      0 0 1 0;
      0 0 0 1];

T=t1*t2*t3*t4*t5*t6;

 ang(i,:)=[alp1; alp2 ;alp3; alp4];
 crd(i,:)=[T(1,4); T(2,4); T(3,4)];

end
```

```
a0=0.6175;
d0=1.170;
k0=0.5;
a1 = 0.115;
d1 = 0.015;
a2 = 0.05697;
d2 = 0.05519;
d3 = 0.1878;
d4 = 0.160;
d5 = 0.23417;
% alp1 = 30;
% alp2 = -40;
% alp3 = 180-40;
% alp4 = 60;
alp5 = 0;
b=10;
i=0;
ang=zeros(1,4);
crd=zeros(1,3);
for alp1=0:b:90
    for alp2=-180:b:40
        for alp3=90:b:270
            for alp4=0:b:90

t1=[cosd(alp1) -sind(alp1) 0 a0;
    sind(alp1) cosd(alp1) 0 d0;
    0 0 1 k0;
    0 0 0 1];
t2=[1 0 0 a1;
    0 cosd(alp2) -sind(alp2) d1;
    0 sind(alp2) cosd(alp2) 0;
    0 0 0 1];
t3=[cosd(alp3) 0 sind(alp3) a2;
    0 1 0 -d2;
    -sind(alp3) 0 cosd(alp3) 0;
    0 0 0 1];
t4=[1 0 0 0;
    0 cosd(alp4) -sind(alp4) -d3;
    0 sind(alp4) cosd(alp4) 0;
    0 0 0 1];
t5=[cosd(alp5) 0 sind(alp5) 0;
    0 1 0 -d4;
    -sind(alp5) 0 cosd(alp5) 0;
    0 0 0 1];
t6=[1 0 0 0;
      0 1 0 -d5;
      0 0 1 0;
      0 0 0 1];

 T=t1*t2*t3*t4*t5*t6;
 i=i+1;
 ang(i,:)=[alp1; alp2 ;alp3; alp4];
 crd(i,:)=[T(1,4); T(2,4); T(3,4)];

            end
        end
    end
end
```

```
syms  t2(alp2) t3(alp3) t4(alp4) x y z

xN=T(1,4);
yN=T(2,4);
zN=T(3,4);

a0=0.6175;
d0=1.170;
k0=0.5;

a1 = 0.115;
d1 = 0.015;
a2 = 0.05697;
d2 = 0.05519;
d3 = 0.1878;
d4 = 0.160;
d5 = 0.23417;

s=sim(nn,[xN;yN;zN]);
alp1=floor(s(1,1))*pi/180;
alp5 = 0;


t1=[cos(alp1) -sin(alp1) 0 a0;
    sin(alp1) cos(alp1) 0 d0;
    0 0 1 k0;
    0 0 0 1];

t2=[1 0 0 a1;
    0 cos(alp2) -sin(alp2) d1;
    0 sin(alp2) cos(alp2) 0;
    0 0 0 1];

t3=[cos(alp3) 0 sin(alp3) a2;
    0 1 0 -d2;
    -sin(alp3) 0 cos(alp3) 0;
    0 0 0 1];

t4=[1 0 0 0;
    0 cos(alp4) -sin(alp4) -d3;
    0 sin(alp4) cos(alp4) 0;
    0 0 0 1];

t5=[cos(alp5) 0 sin(alp5) 0;
    0 1 0 -d4;
    -sin(alp5) 0 cos(alp5) 0;
    0 0 0 1];

t6=[1 0 0 0;
      0 1 0 -d5;
      0 0 1 0;
      0 0 0 1];

T=t1*t2*t3*t4*t5*t6;
```

Appendix 5 – MatLab inverse kinematics solver with shoulder angle defined by NN

```
syms  t2(alp2) t3(alp3) t4(alp4)  x y z

x = T(1,4) == xN;
y = T(2,4) == yN;
z = T(3,4) == zN;

S=vpasolve([x, y, z], [alp2, alp3, alp4],[-190*pi/180 50*pi/180;80*pi/180
280*pi/180;-10*pi/180 100*pi/180]);

S2=S.alp2;
S3=S.alp3;
S4=S.alp4;

alps1=alp1*180/pi
alps2 = double(S2*180/pi)
alps3 = double(S3*180/pi)
alps4 = double(S4*180/pi)
```