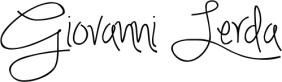
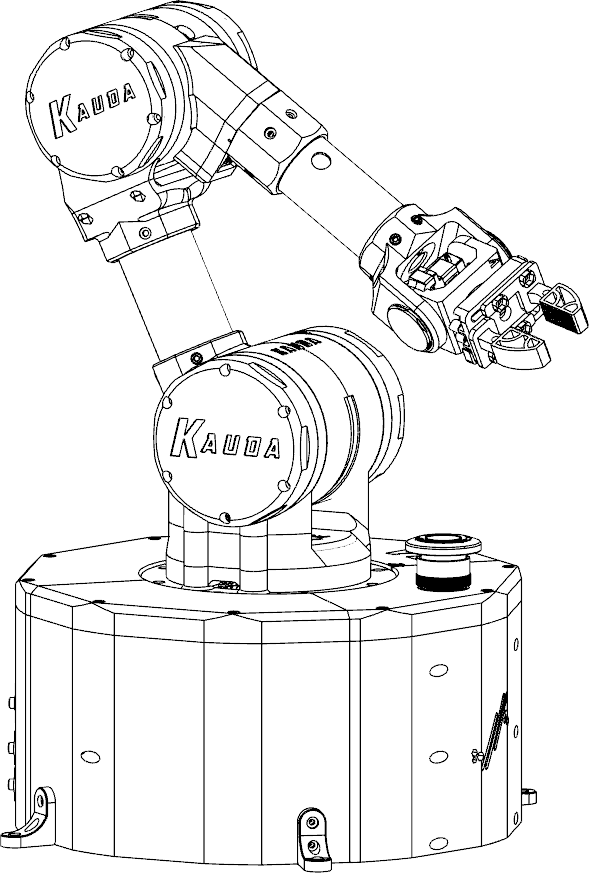
KAUDA PRO

Table of Contents

1. Title Page
2. Table of Contents
3. Introduction
   1. Overview of the Project
   2. Purpose of the Manual
   3. Brief Description of the Robotic Arm
4. Hardware Components
   1. 3D Printed Parts Assembly (based on Giovanni Lerda's design)
      1. List of 3D Printed Parts
      2. Assembly Instructions
   2. Electrical Components (based on Giovanni Lerda's design)
      1. List of Electrical Components
      2. Schematics and Wiring Diagrams
      3. Assembly Instructions
5. Software Documentation
   1. Installation Instructions
   2. Code Structure and Explanation
   3. Uploading and Running the Software
6. Operation Instructions
   1. Step-by-Step Guide to Operating the Robotic Arm
   2. Safety Precautions
   3. Maintenance Tips
7. Troubleshooting
   1. Common Issues and Solutions
   2. Frequently Asked Questions (FAQs)
8. Appendices
   1. Additional Resources
   2. Links to Further Reading

*KAUDA*

*ROBOTIC ARM*



*Summary*

[Summary 3](#_bookmark0)

1. [Introduction 4](#_bookmark1)
   1. [Purpose of the manual 4](#_bookmark2)
   2. [Structure of the manual 4](#_bookmark3)
2. [General information 5](#_bookmark4)
   1. [General working rules 5](#_bookmark5)
3. [Overview 6](#_bookmark6)
   1. [Axis & Nomenclature 6](#_bookmark7)
4. [Assembly 7](#_bookmark8)

[Material list 7](#_bookmark9)

[3D printed parts 7](#_bookmark10)

[External parts 8](#_bookmark11)

[Axis Z 10](#_bookmark12)

[Axis Y 20](#_bookmark13)

[Axis X 28](#_bookmark14)

[Axis V 37](#_bookmark15)

[Axis W 38](#_bookmark16)

[Gripper 41](#_bookmark17)

[Axis Assembly 43](#_bookmark18)

***1.***

***Introduction***

***KAUDA*** is a 5-axis anthropomorphic robot, suitable for applications with educational and recreational purposes only. It is **NOT** a suitable product for production.

### Purpose of the manual

This manual covers the main mechanical, electrical and software checks, the certifications, and the assembly of components supplied in bulk, to carry out the delivery of the new factory product (the sequence of operations is not binding).

It is very important to strictly follow what has been described. Interventions superficially carried out or even omitted can generate personal damages to the buyer, to the product, etc ... or produce, in the simplest of hypotheses, unpleasant disputes.

### Structure of the manual

|  |  |
| --- | --- |
| ***Chapter*** | ***Description*** |
| **1** | First introduction to the documentation |
| **2** | General product information |
| **3** | Robot overview |
| **4** | Assembly guide |
| **5** | Software installation |
| **6** | First start-up and parameterization |

1. *General information*

### General working rules

The following advice, recommendations and warnings guarantee rational interventions in the maximum operational safety, significantly reducing the probability of accidents, damage of any nature and downtime. Yes therefore, he advises to observe them scrupulously.

TIPS:

* + - Always use top quality equipment.
    - Keep the tools close from hand during the operations, preferably according to a predetermined sequence and in any case never on the vehicle or in hidden or inaccessible positions.
    - Keep the workplace tidy and clean.
    - To tighten screws and nuts, start with those of larger diameter or internal ones, proceeding at "cross" with subsequent "pulls".

1. *Overview*

The robot consists of 5 axes, which can be assembled separately and subsequently joined together.

### Axis & Nomenclature

|  |  |
| --- | --- |
| ***AXIS*** | ***GROUP*** |
| **Z** | Base |
| **Y** | Y Axis |
| **X** | X & V Axis |
| **V** | X & V Axis |
| **W** | W Axis |
| **G** | Gripper |

*V*

*X*

*W*

*G*

*Y*

*Z*

1. *Assembly*

The assembly of the robot is divided into 2 steps:

* + - *Assembly of the 5 axes*
    - *Complete assembly & Wiring*

### Material list

### 3D printed parts

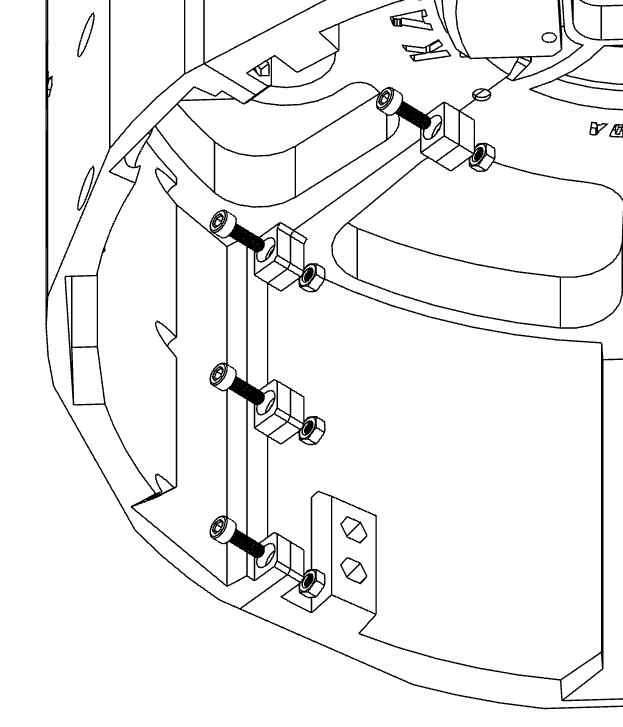
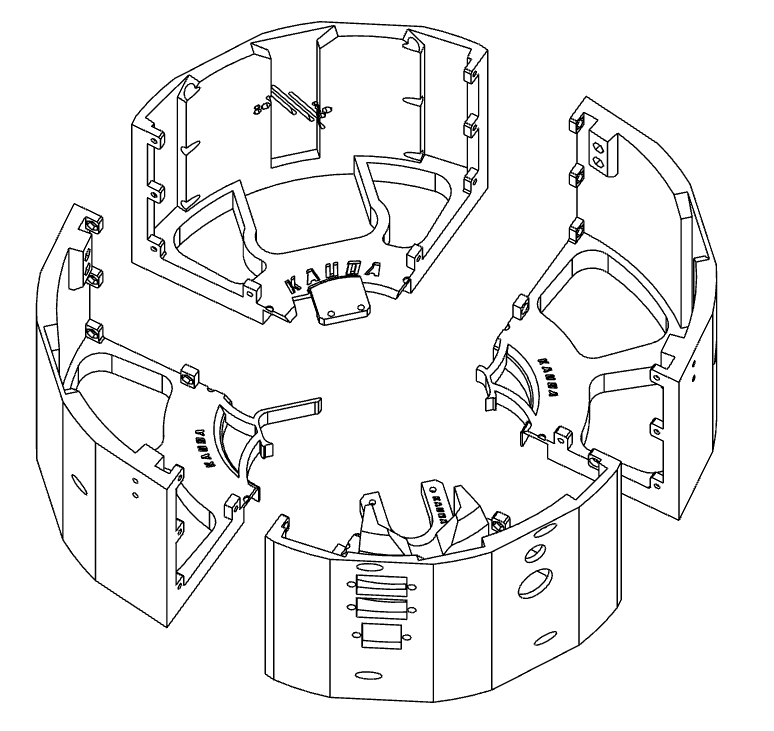
|  |  |  |  |
| --- | --- | --- | --- |
| ***N°*** | ***QTY*** | ***NAME*** | ***REF.P.*** |
| 1 | **1** | BS-A-KRA-0 | **10** |
| 2 | **1** | BS-B-KRA-1 | **10** |
| 3 | **1** | BS-C-KRA-2 | **10** |
| 4 | **1** | BS-D-KRA-3 | **10** |
| 5 | **1** | BS-RG-A-KRA-0 | **15** |
| 6 | **1** | BS-RG-B-KRA-0 | **15** |
| 7 | **1** | BS-RG-C-KRA-0 | **15** |
| 8 | **1** | BS-RG-D-KRA-0 | **15** |
| 9 | **1** | BS-V3-A-KRA-0 | **17** |
| 10 | **1** | BS-V3-B-KRA-0 | **17** |
| 11 | **1** | BS-V3-C-KRA-0 | **17** |
| 12 | **1** | BS-V3-D-KRA-0 | **17** |
| 13 | **1** | BS-TP-KRA-0 | **18** |
| 14 | **1** | CVR-A-KRA-0 | **14** |
| 15 | **1** | CVR-B-KRA-0 | **14** |
| 16 | **1** | CVR-NW-KRA-0 | **14** |
| 17 | **1** | RD-CVR-KRA-0 | **14** |
| 18 | **1** | RD-ENC-KRA-0 | **12** |
| 19 | **1** | RD-KRA-0 | **13** |
| 20 | **1** | RG-KRA-0 | **16** |
| 21 | **1** | SP-A-KRA-0 | **13** |
| 22 | **1** | SP-B-KRA-0 | **13** |
| 23 | **4** | FT-KRA-0 | **12** |
| 24 | **1** | BS-CN-KRA-0 | **/** |
| 25 | **1** | CBL-A-KRA-0 | **18** |
| 26 | **1** | CBL-B-KRA-0 | **18** |
| 27 | **1** | AX1-B-KRA-1 | **22** |
| 28 | **1** | AX1-L-KRA-1 | **22** |
| 29 | **1** | AX2-A-KRA-1 | **20** |
| 30 | **1** | AX2-B-KRA-1 | **21** |
| 31 | **1** | FPLT-R-KRA-1 | **24** |
| 32 | **1** | RG-FL-KRA-1 | **22** |
| 33 | **1** | SPT-KRA-1 | **21** |
| 34 | **1** | FPLT-L-KRA-1 | **27** |
| 35 | **3** | PLT-KRA-1 | **26** |
| 36 | **3** | PLT-SF-KRA-1 | **25** |
| 37 | **2** | RG-KRA-1 | **25** |

|  |  |  |  |
| --- | --- | --- | --- |
| 38 | **1** | RG-TW-KRA-1 | **26** |
| 39 | **1** | SUN-TW1-KRA-1 | **26** |
| 40 | **1** | SUN-TW2-KRA-1 | **27** |
| 41 | **3** | TND-KRA-1 | **26** |
| 42 | **1** | AX1-A-KRA-1 | **23** |
| 43 | **1** | AX2-A-KRA-2 | **31** |
| 44 | **1** | AX2-L-KRA-2 | **32** |
| 45 | **1** | AX2-R-KRA-2 | **31** |
| 46 | **1** | AX3-A-KRA-2 | **28** |
| 47 | **1** | AX3-B-KRA-2 | **30** |
| 48 | **1** | AX4-A-KRA-2 | **37** |
| 49 | **1** | CVR-KRA-2 | **36** |
| 50 | **1** | FPLT-R-KRA-2 | **36** |
| 51 | **1** | RG-FL-KRA-2 | **31** |
| 52 | **1** | SPT-KRA-2 | **31** |
| 53 | **1** | SRV-KRA-2 | **37** |
| 54 | **1** | FPLT-L-KRA-2 | **35** |
| 55 | **3** | PLT-KRA-2 | **34** |
| 56 | **3** | PLT-SF-KRA-2 | **33** |
| 57 | **2** | RG-KRA-2 | **32** |
| 58 | **1** | RG-TW-KRA-2 | **34** |
| 59 | **1** | SUN-TW1-KRA-2 | **33** |
| 60 | **1** | SUN-TW2-KRA-2 | **34** |
| 61 | **3** | TND-KRA-2 | **34** |
| 62 | **1** | AX4-B-KRA-2 | **38** |
| 63 | **1** | AX5-KRA-3 | **38** |
| 64 | **1** | CVR-KRA-3 | **40** |
| 65 | **1** | FPLT-L-KRA-3 | **40** |
| 66 | **1** | FPLT-R-KRA-3 | **40** |
| 67 | **1** | TND-3-KRA-3 | **39** |
| 68 | **1** | TND-4-KRA-3 | **/** |
| 69 | **1** | BS-TLS-0 | **41** |
| 70 | **1** | GR-TLS-0 | **41** |
| 71 | **2** | JW-TLS-0 | **41** |
| 72 | **2** | RK-TLS-0 | **41** |

### External parts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***N°*** | ***QTY*** | ***CODE*** | ***REF.P.*** | ***NOTE*** |
| 1 | **1** | 17HS19-2004S1 | **13** | Stepper Motor 2A- 0,59Nm |
| 2 | **1** | 17HS19-2004S1 | **13** | Stepper Motor 2A- 0,59Nm |
| 3 | **1** | 17HS19-2004S1 | **13** | Stepper Motor 2A- 0,59Nm |
| 4 | **1** | USBPNLBFBM1 | **14** | USB B Cable front panel |
| 5 | **1** | KY-040 | **12** | Rotary encoder |
| 6 | **1** | KY-040 | **12** | Rotary encoder |
| 7 | **1** | KY-040 | **12** | Rotary encoder |
| 8 | **1** | DHT-11 | **13** | Temperature sensor |
| 9 | **1** | DHT-11 | **13** | Temperature sensor |
| 10 | **1** | DHT-11 | **13** | Temperature sensor |
| 11 | **1** | USBPNLAFAM1 | **14** | USB A Cable front panel |
| 12 | **1** | CNCSHIELD V3 | **18** | Hardware used to control stepper motors |
| 13 | **1** | TMC2209 | **18** | Motor driver IC for two phase stepper motors |
| 14 | **1** | TMC2209 | **18** | Motor driver IC for two phase stepper motors |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 15 | **1** | TMC2209 | **18** | Motor driver IC for two phase stepper motors |
| 16 | **1** | Arduino Mega | **18** | Arduino Mega Board |
| 17 | **1** | Raspberry Pi | **18** | Raspberry PI 4 |
| 18 | **1** | MG996R | **29** | Servo motor |
| 19 | **1** | MG90S | **38** | Servo motor |
| 20 | **20** | 22AWG | **/** | 22AWG Cable |
| 21 | **2** | Braided Sheath | **/** | Braided cable sheath Ø19 mm |
| 22 | **1** | Emergency PB | **15** | Emergency push button Ø22 mm |
| 23 | **12** | 40621 | **/** | Electrical terminals. |
| 24 | **1** | LM2596S | **18** | DC/DC Converter |
| 25 | **1** | Jack Connector | **14** | Female Jack connector Ø11 mm |
| 26 | **1** | GX20 | **14** | GX20 Connector |
| 27 | **1** | Tube Ø40 x 100 | **43** | Tube Ø40 x 100 mm (Aluminium) |
| 28 | **1** | Tube Ø35 x 100 | **43** | Tube Ø35 x 100 mm (Aluminium) |
| 29 | **1** | / | **14** | Male/Female Ethernet Extension Cable |
| 30 | **12** | 686ZZ | **25** | Bearing 6x13x5 |
| 31 | **1** | 51116 | **11** | Bearing 80x105x19 |
| 32 | **1** | 608ZZ | **39** | Bearing 8x22x7 |
| 33 | **2** | 6804ZZ | **22** | Bearing 20x32x7 |
| 34 | **1** | 28BYJ-48 | **39** | Stepper Motor 5V (360°) |
| 35 | **1** | ULN2003 | **18** | 5V Driver (stepper) |
| 36 | **66** | M3 x 5,7 Inserts | **/** | M3x5,7 Insert |
| 37 | **7** | TCEI M3 x 6 | **/** | Screw TCEI M3 x 6 |
| 38 | **34** | TCEI M3 x 8 | **/** | Screw TCEI M3 x 8 |
| 39 | **47** | TCEI M3 x 10 | **/** | Screw TCEI M3 x 10 |
| 40 | **12** | TCEI M3 x 12 | **/** | Screw TCEI M3 x 12 |
| 41 | **42** | TCEI M3 x 16 | **/** | Screw TCEI M3 x 16 |
| 42 | **4** | TCEI M3 x 20 | **/** | Screw TCEI M3 x 20 |
| 43 | **2** | TCEI M3 x 30 | **/** | Screw TCEI M3 x 30 |
| 44 | **6** | TCEI M4 x 40 | **/** | Screw TCEI M4 x 40 |
| 45 | **6** | DIN 993 M6 x 20 | **/** | Screw DIN 993 M6 x 20 |
| 46 | **60** | M3 NUT | **/** | Nut M3 |
| 47 | **2** | ALLOY DIN916 M3 x 6 | **/** | Alloy DIN916 M3 x 6 |



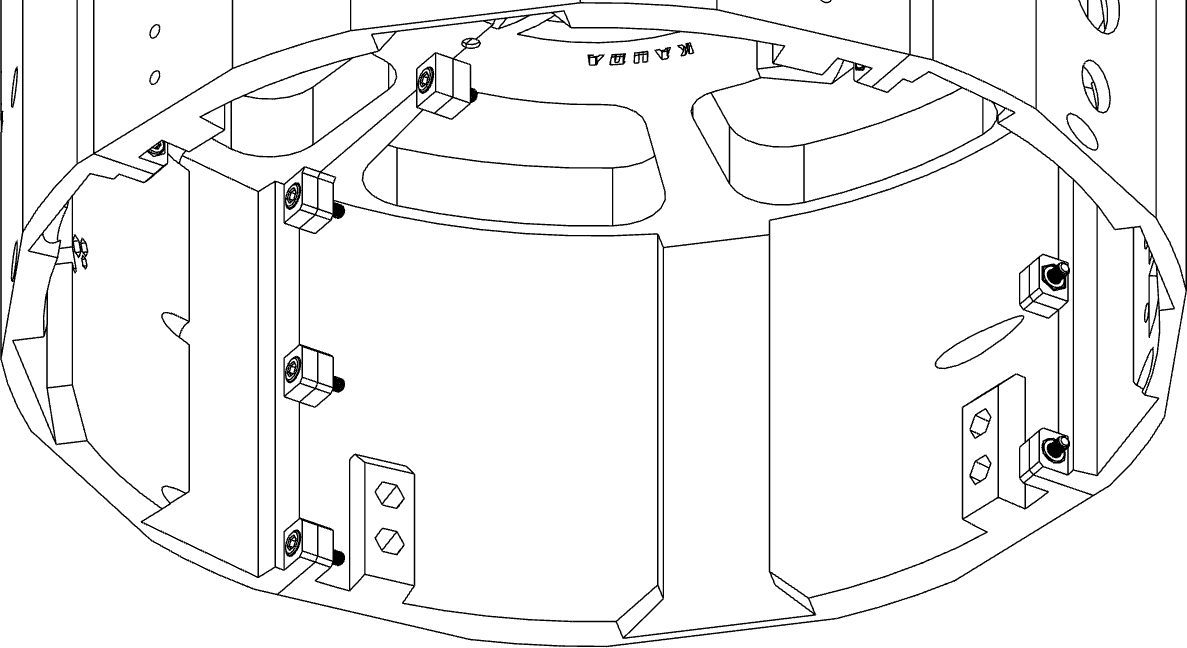
# Axis Z

1. Collect all the components needed to build the **Base**.
2. Assemble the 4 parts that make up the **Base**.

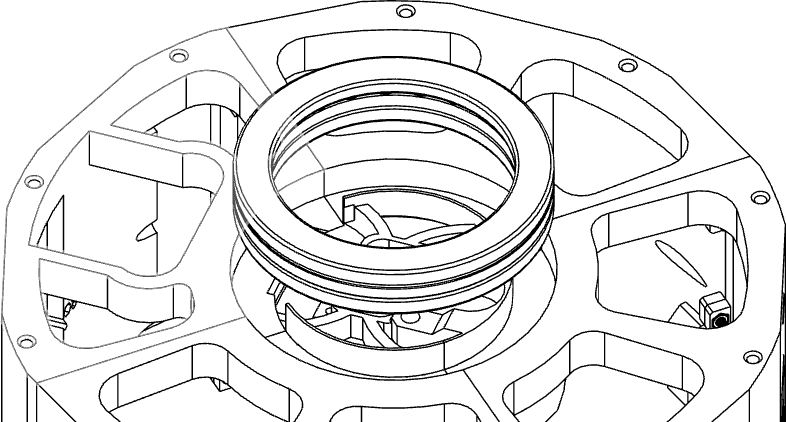
**X4**

#### BS-A-KRA-0, BS-B-KRA-0, BS-C-

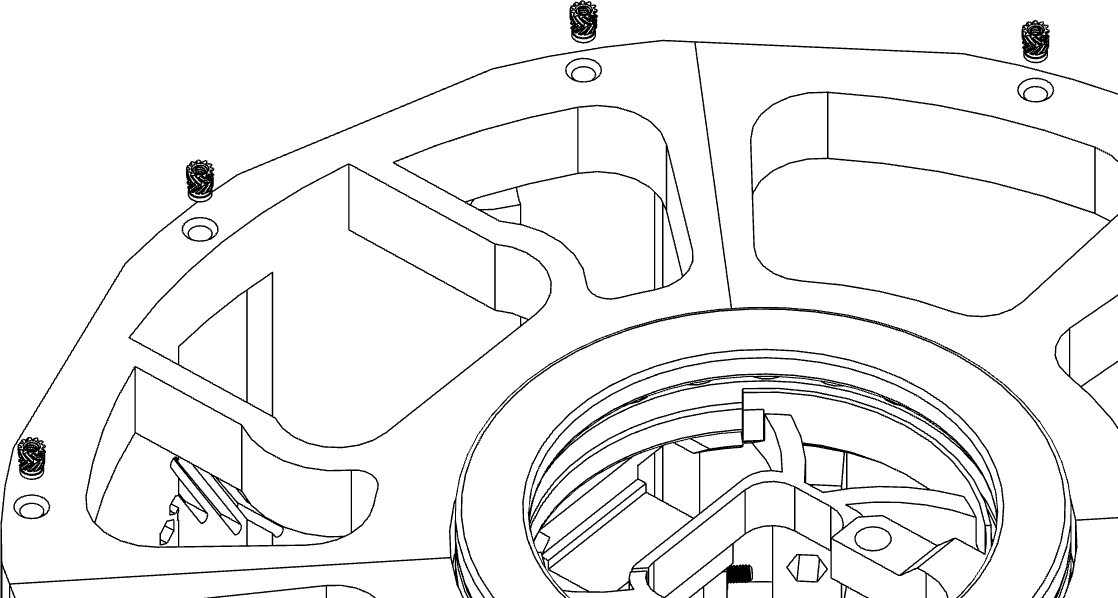
**KRA-0**, **BS-D-KRA-0**, must be joined using a total of **16 M3 x 10** socket head screws and **16 M3 nuts**.

To screw the screws, use the holes that allow the screwdriver to pass.

1. Position the bearing **BR 51116**.



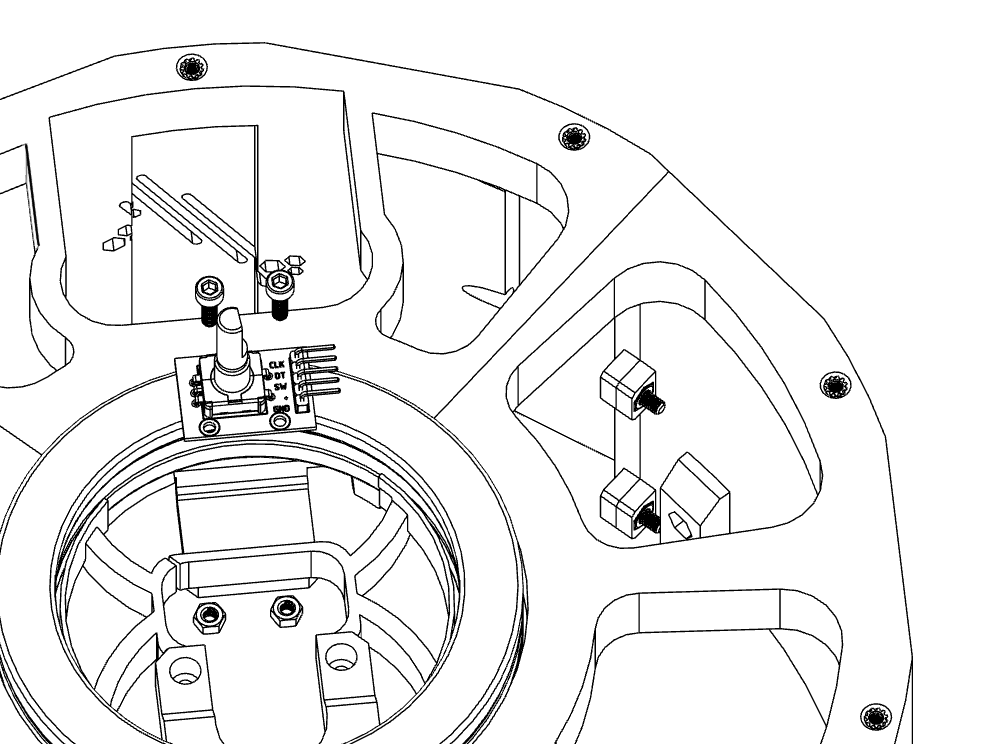
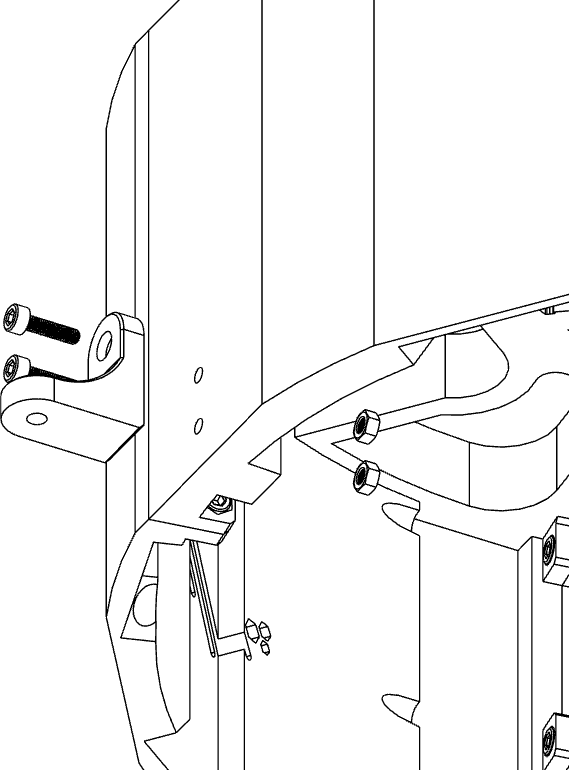
Insert the M3 threaded inserts by **4**



**X10**

heating them with the tip of a

soldering iron. A total of **10 inserts**.



Mount the feet of the **FT-KRA-0** robot base with **2 M3 x 12** socket head screws and **2 M3 nuts**.

For a total of 4 **FT-KRA-0**.

**X**

**4**

**5**

**6**

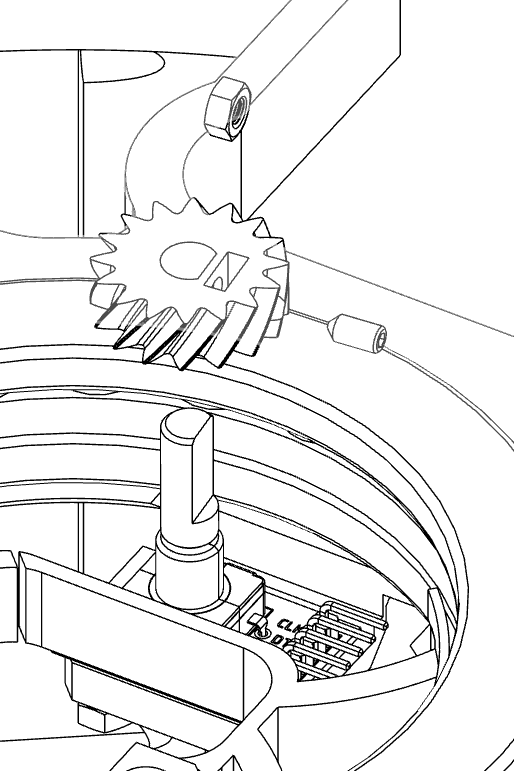
**X**

Mount the **KY-040** encoder

using **2 M3 x 10** socket head

screws and **2 M3 nuts**.

**2**

Mount **RD-ENC-KRA-0** on the encoder shaft, inserting an M3 nut in the appropriate slot and screwing a **DIN916 M3 x 6 screw**.



**7**

**X1**

Correct positioning can be done during the assembly phase of the **RG-KRA-0.**

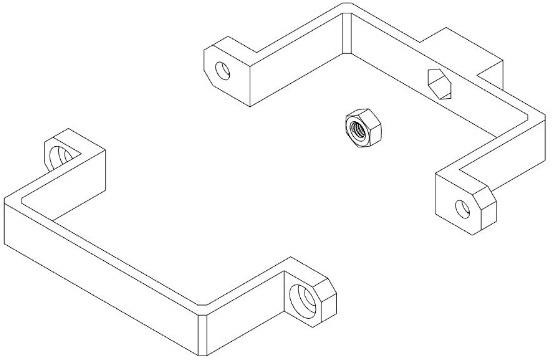
Mount **RD-KRA-0** on the **NEMA 17** motor shaft, inserting an **M3 nut** in the appropriate slot and screwing a **DIN916 M3 x 6 screw**.

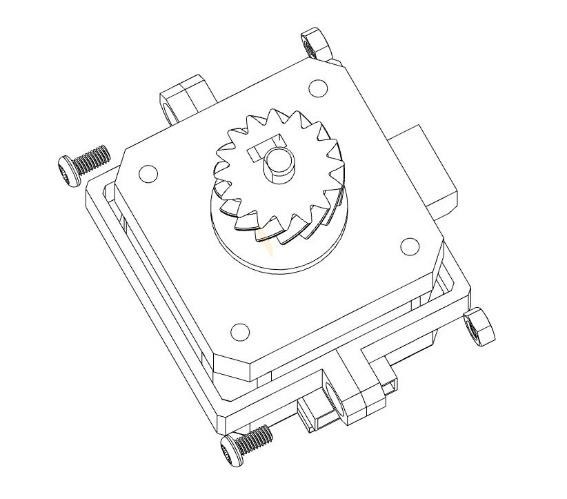


**8**

**X1**

Correct positioning can be done during the assembly phase of the **RG-KRA-0**

Insert an M3 nut ins**X**ide **SP-A-K1RA-0**. 



Mount the **DHT-11** sensor to the

holder using an **M3 x 8** socket head

**X**

**1**

screw.



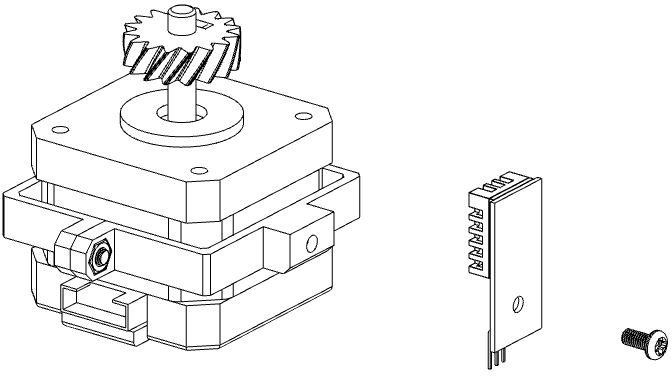
**9**

**X2**

Fix the **SP-A-KRA-0** and **SP-B-KRA-**0 supports using **two M3 x 10 socket** head screws and two **M3 nuts**.



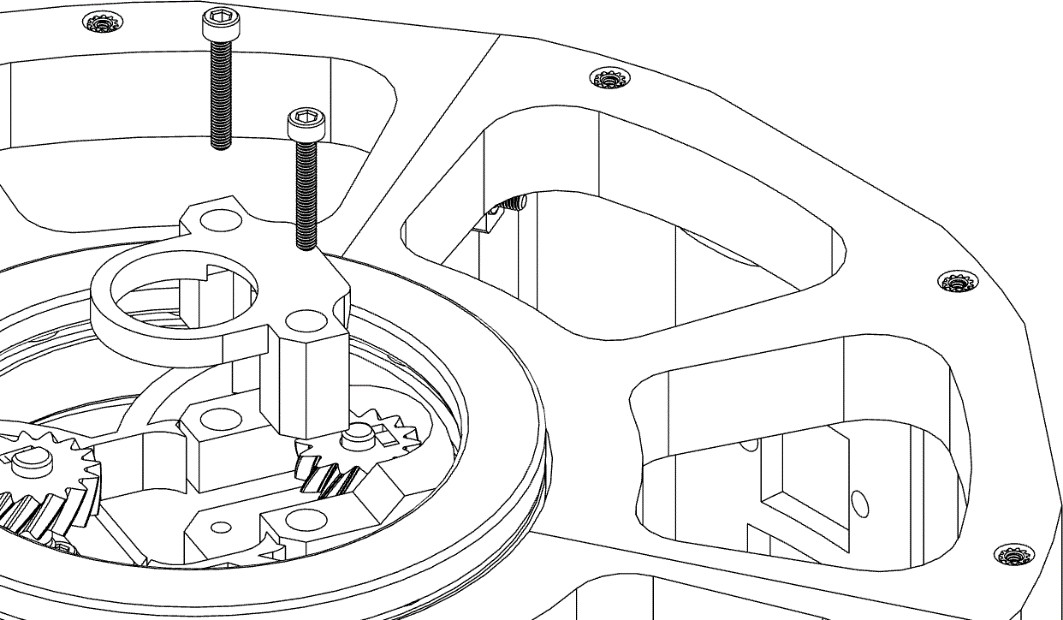
**10**





**11**

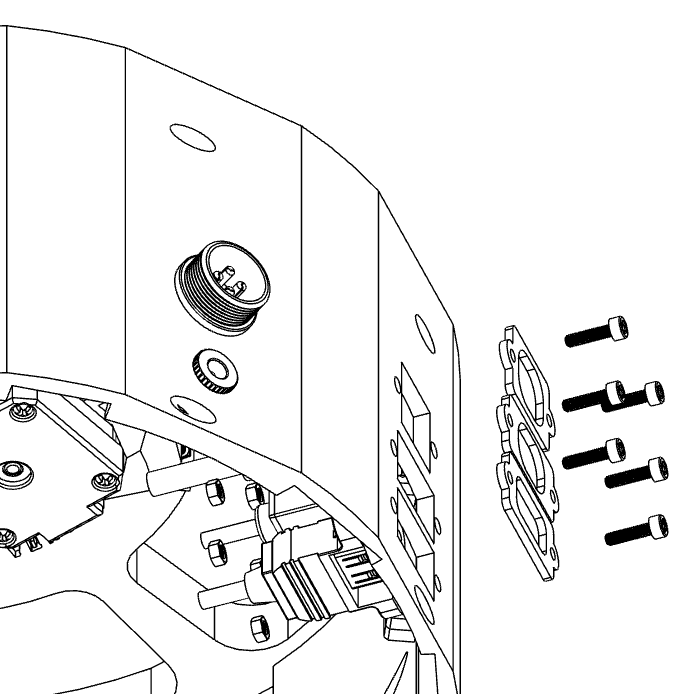
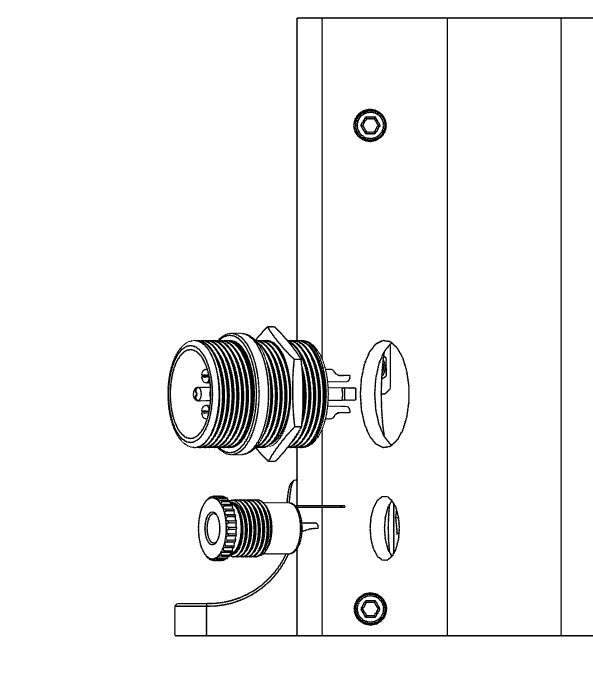
**12** The NEMA motor complete with **DHT-11** sensor must be fixed to the base, with the insertion of the **RD-CVR-KRA-0**, using **two M3 x 20** socket



**X2**

head screws.

Mount the power connector and the **13**



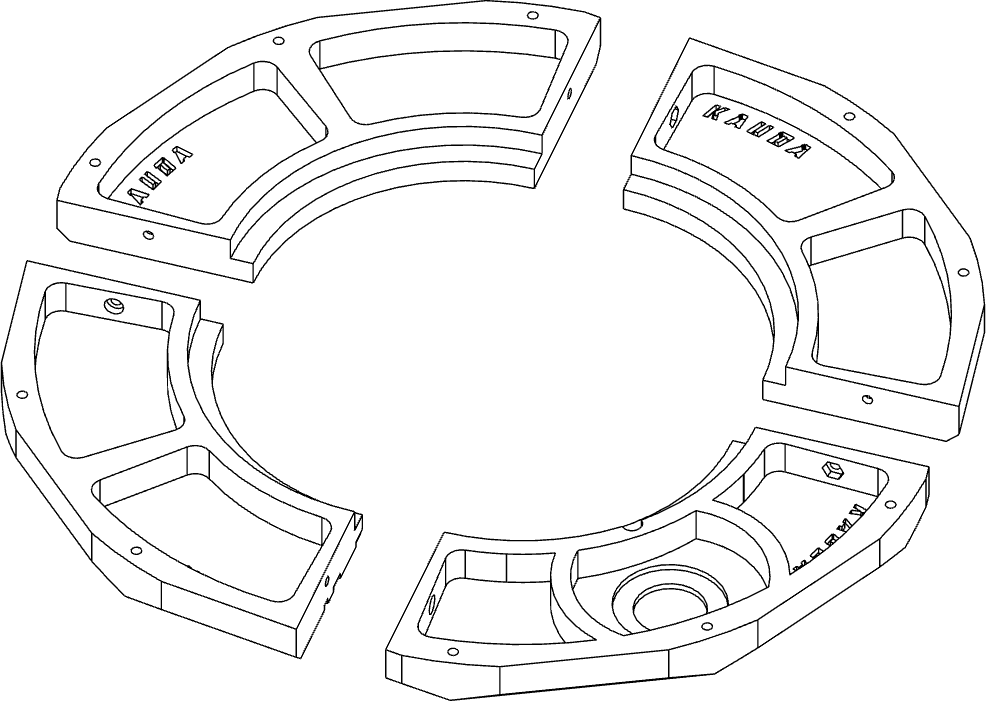
**GX-20** connector using the dedicated

nuts.

**14** Fix the 3 USB and RJ45 connectors / extensions using **6 M3 x 16** socket head screws and 6 M**3 nuts.**

**X6**

For more information on the order of assembly and subsequent installation, consult the wiring diagram.



Assemble the **4 parts** that make up

**X**

the base cover.

**4**

**15**

**BS-RG-A-KRA-0, BS-RG-B-KRA-0, BS-RG-C-**

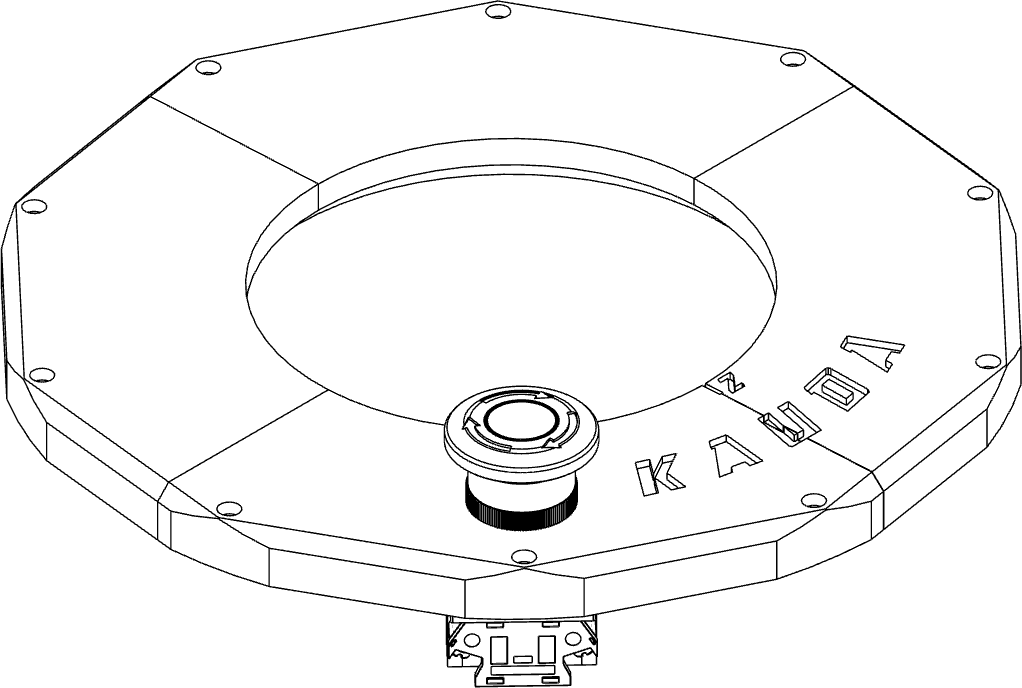
**KRA-0, BS-RG-D-KRA-0**, must be joined using a total of 4 socket head screws **M3 x 10** and **16 M3**

**nuts.**

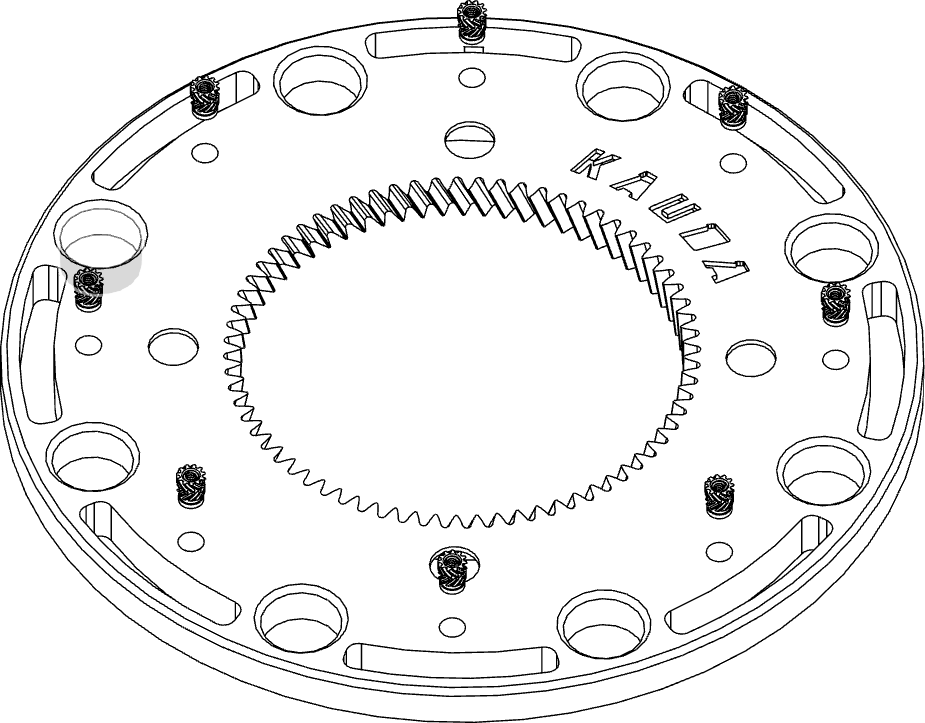
**X**

**4**

Fix the emergency button. **16**



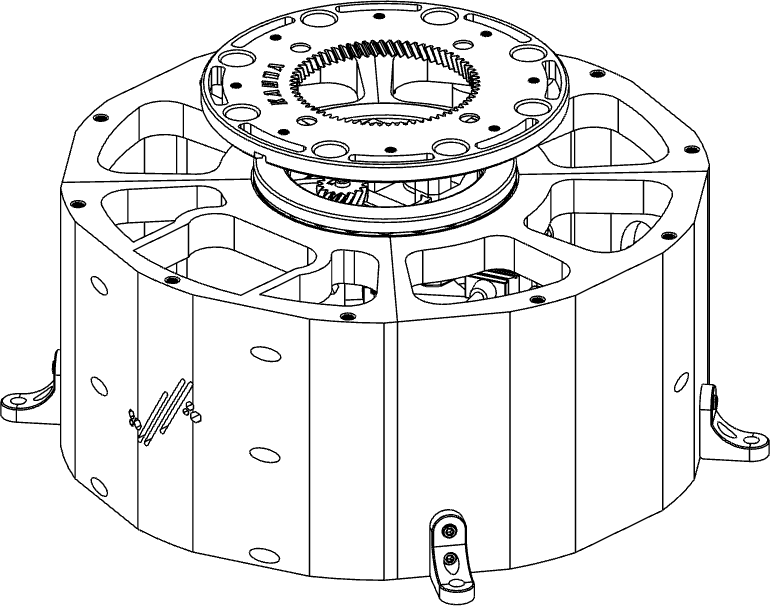
Insert the **M3 threaded inserts** by heating them with the tip of a soldering iron. A total of **8 inserts**.



**17**

**X8**

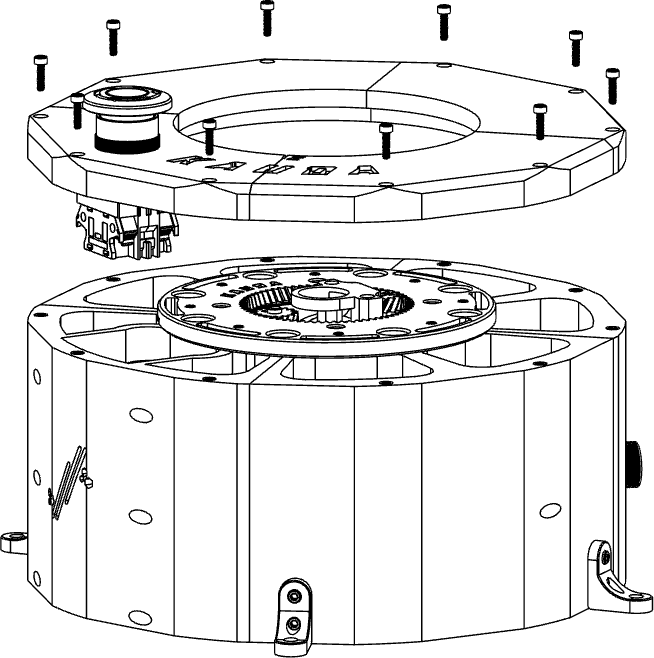
Place **RG-KRA-0** on bearing

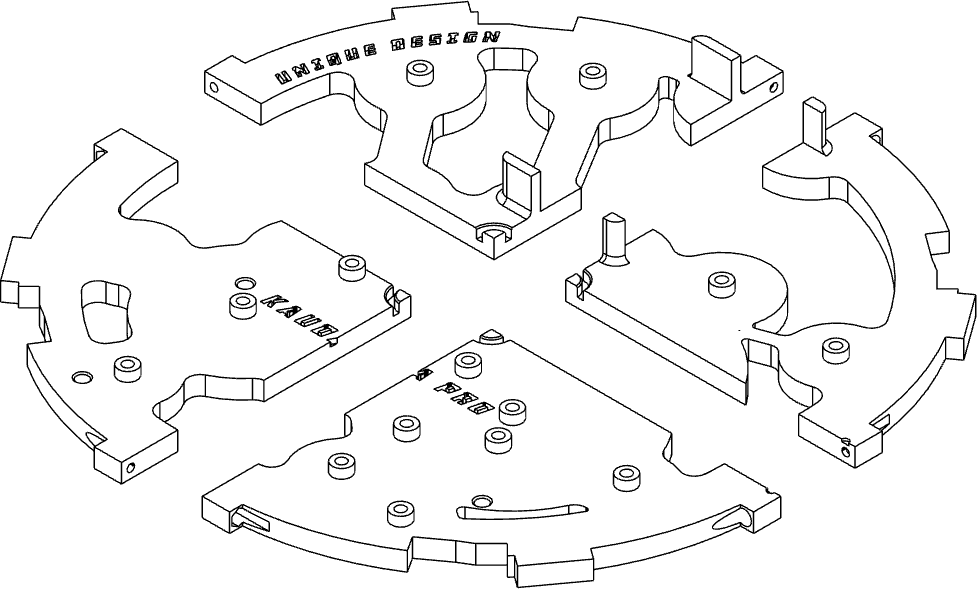


**18**

#### BR 51116.

**X10**



Assemble the 4 parts that make up the **BS-V3-KRA-0**



**19**



**20**

**X4**

support.

**BS-V3-A-KRA-0, BS-V3-B-**

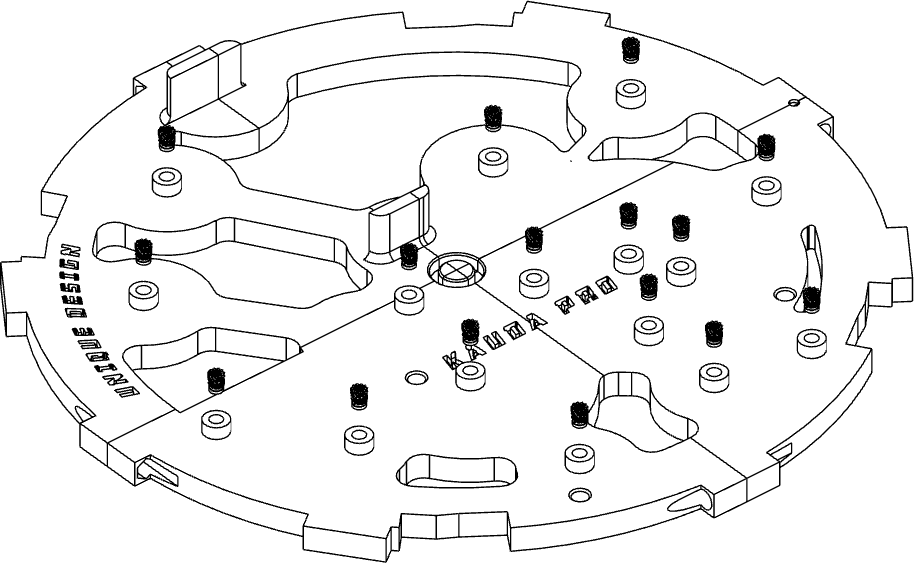
**KRA-0, BS-V3-B-KRA-0, BS-**

**V3-D-KRA-0**.

Mount the **BS-RG-KRA-0** assembly with the emergency button on the base, using **10 M3 x 16** socket head screws.



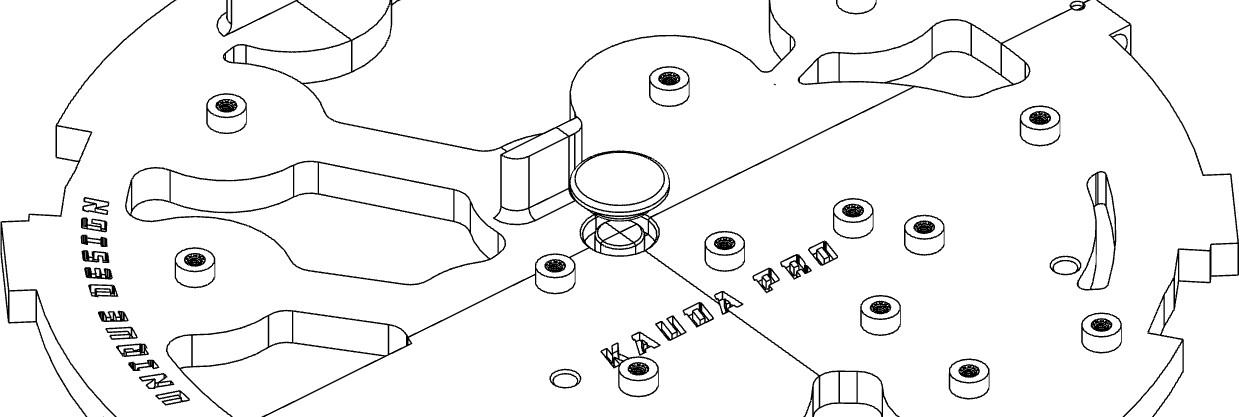
Fix the **M3 inserts** inside the appropriate cavities, for a total of **16 inserts.**



**19**

**X16**

Fix **BS-TP-KRA-0** in the center of the 4 supports in the appropriate cavity.



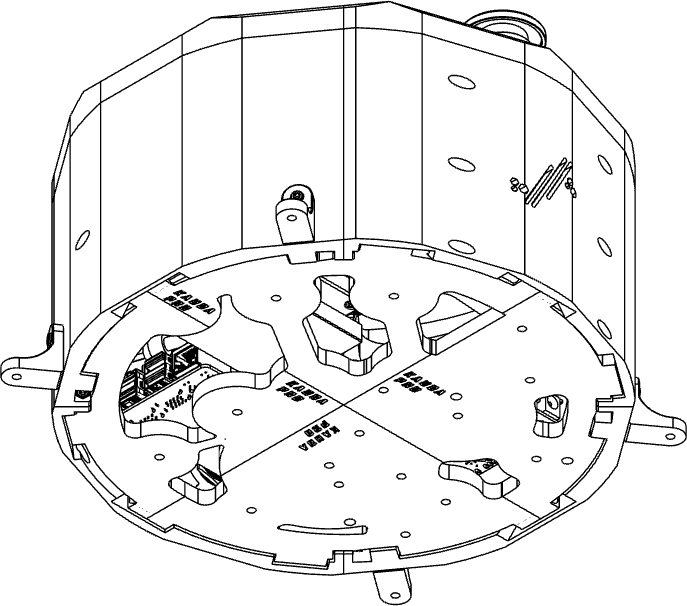
**20**

Fix all the electrical power supply and control using **M3 x 6** socket head screws, for a total of **13 screws**.



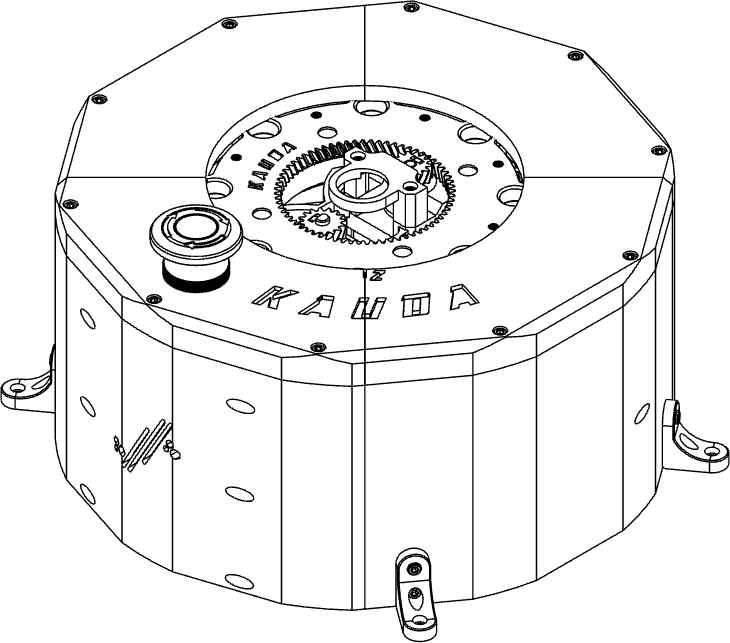
**21**

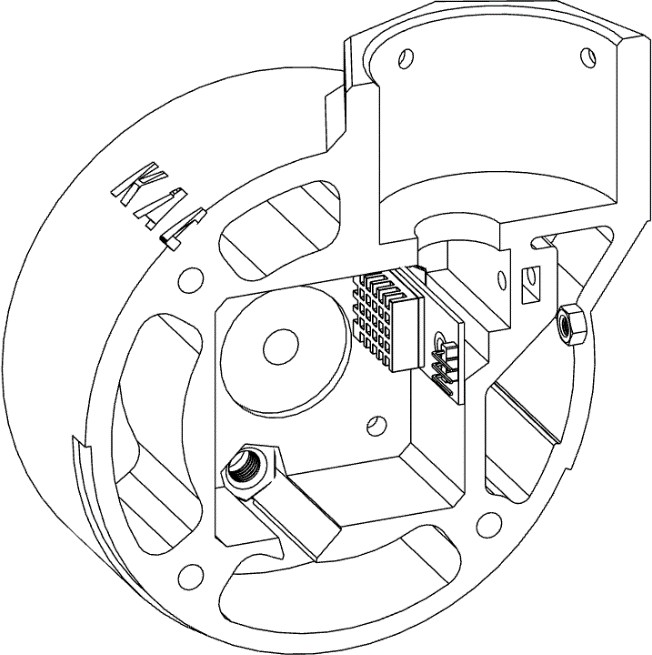
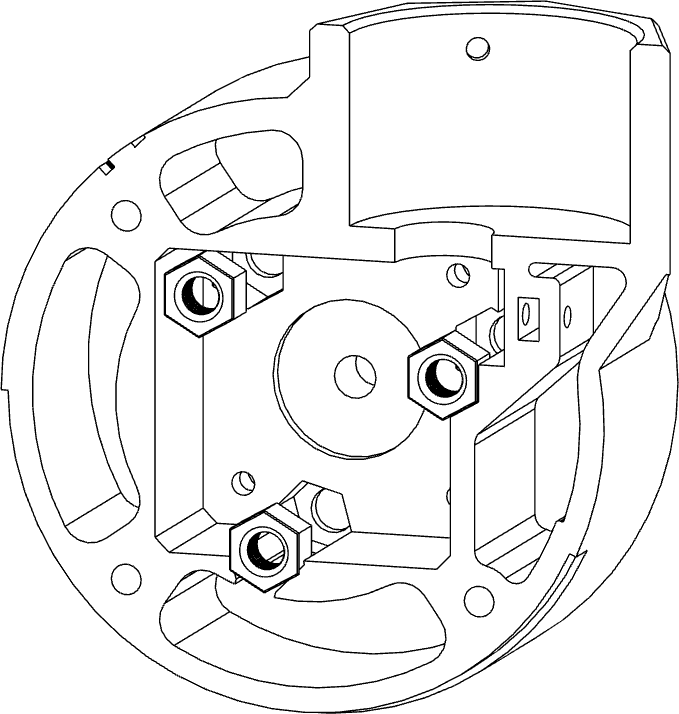
**X13**

Place the support base of the **BS-V3-KRA-0** components inside the **BS-KRA-0**.



**22**





*Axis Y*

Insert 3 M6 nuts inside the

appropriate slots of **AX2-KRA-1**.

**X**

**3**

**1**

**2** Install the **DHT-11** sensor on the **AX2-KRA-1** using a **M3 x 6** TCEI **screw** and **M3 nut**.

**3** Mount the Nema 17 motor on the

**X**

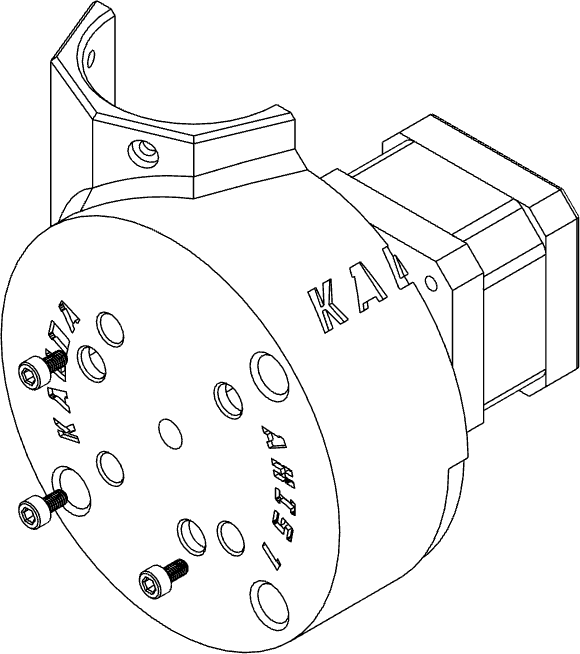
**AX2-A-KRA-1** using 3 socket head

screws **M3 x 8**.

**3**

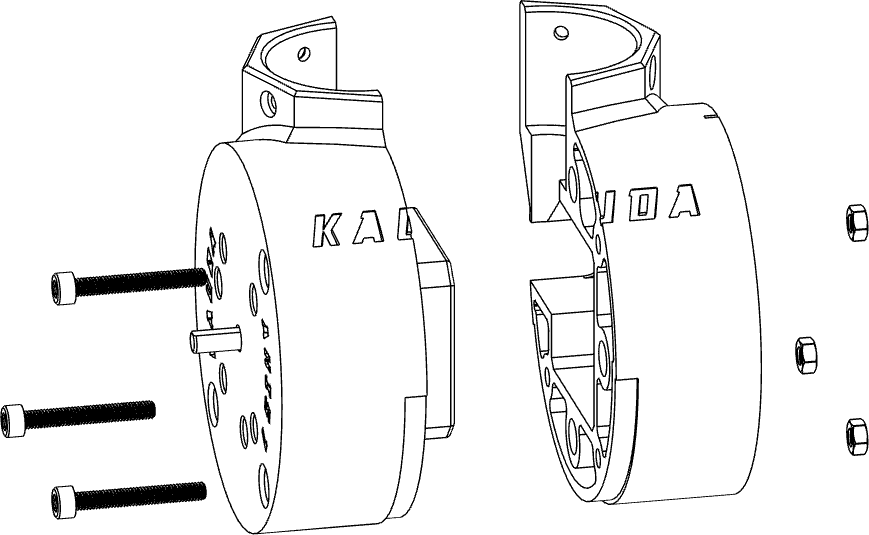
**NOTE !**

Pay attention to the motor mounting direction, the cable must be facing the rear hole.





**4** Mount **AX2-A-KRA-1** and **AX2-B-**



**X3**

**KRA-1** using **3 M4 x 40** screws and

#### 3 M4 nuts.

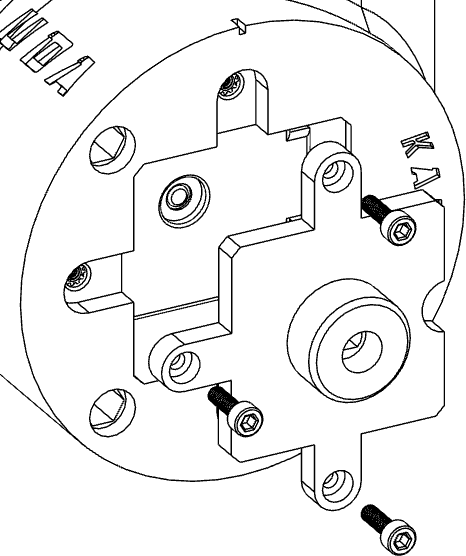
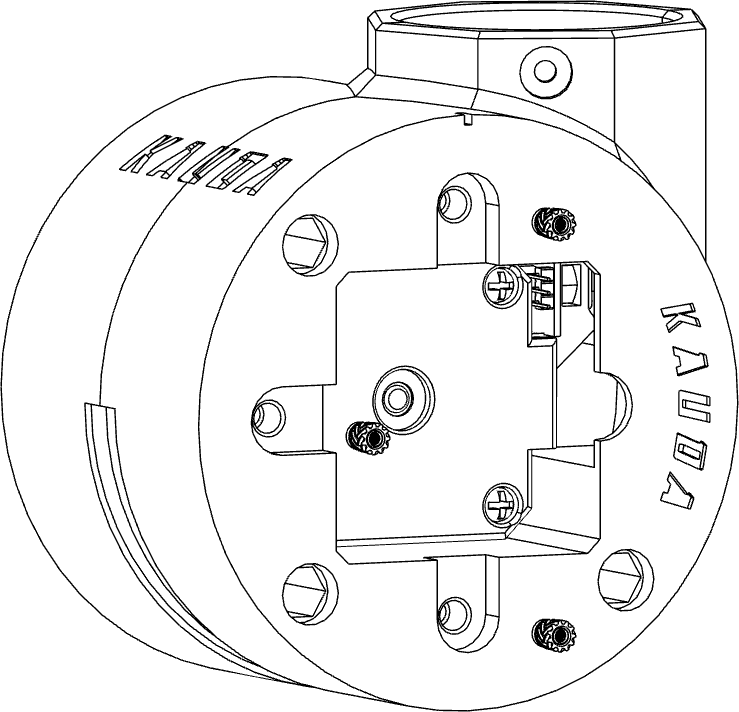
Insert **3 M3 inserts** inside the



**5**

**X3**

appropriate cavities.

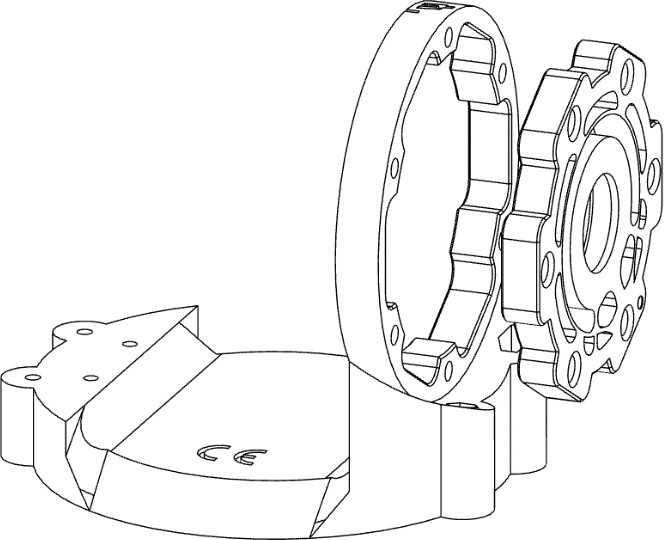
Install **SPT-KRA-1** on **AX2-B-KRA-1**



**6**

**X3**

using **3 M3 x 10** socket head screws.

Insert the **RG-FL-KRA-1** inside the appropriate space on **AX1-B-KRA-**

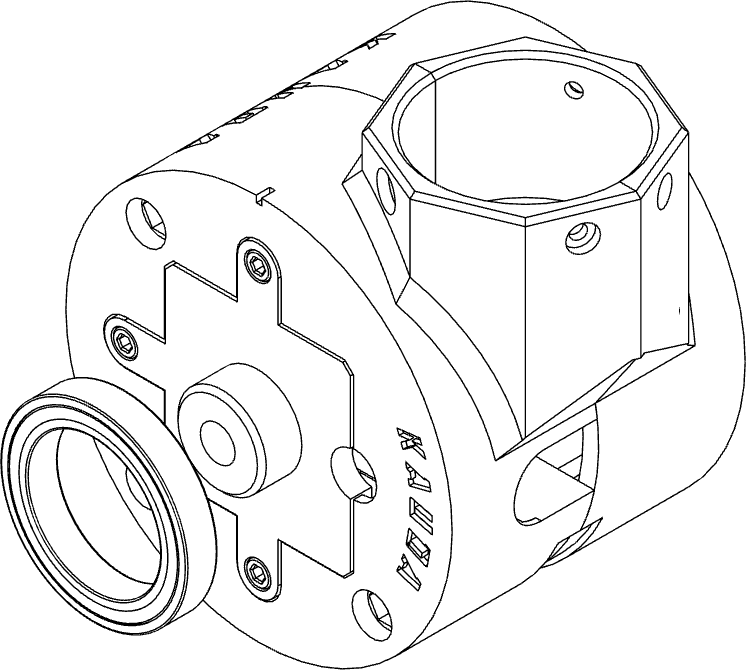


**7**

**1**.



**8**

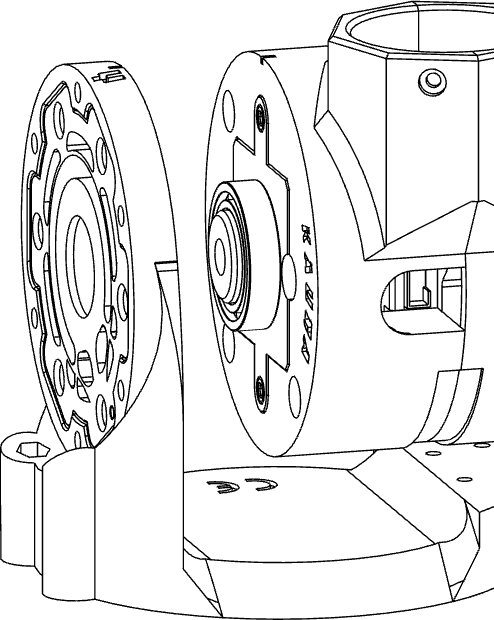


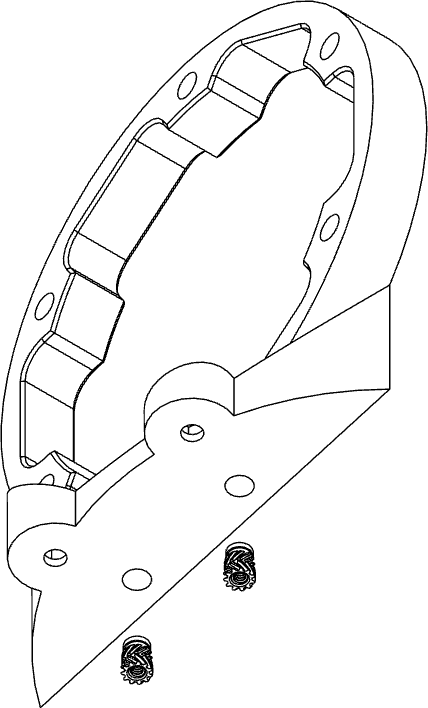
Join **AX1-B-KRA-1** to **AX2-KRA-1**,



**9**

fixing the bearing **6806ZZ** in the appropriate slot of **RG-FL-KRA-1**.

Fix bearing **6806ZZ** to **SPT-KRA-1.**

Insert **2 M3 inserts** inside the appropriate cavities of **AX1-L-KRA-1**.



**10**

**X2**

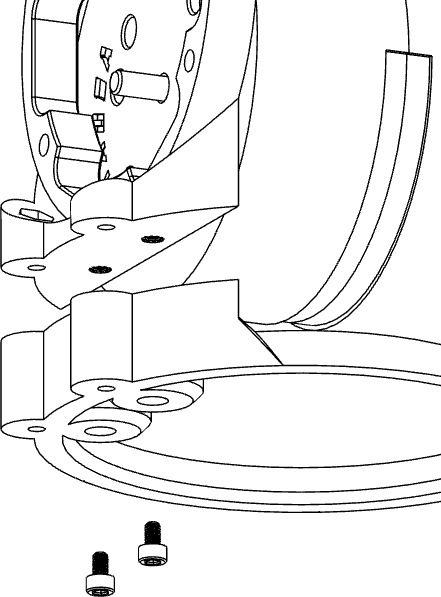
Fix **AX1-L-KRA-1** to **AX1-B-KRA-1**



**11**

**X2**

using **2 M3 x 6** socket head screws.



Fix **AX1-A-KRA-1** to **AX1-B-KRA-1**



**12**

**X4**

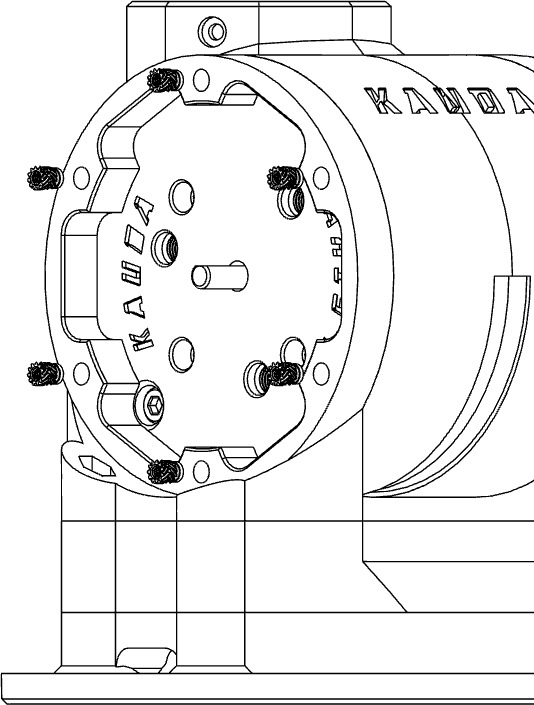
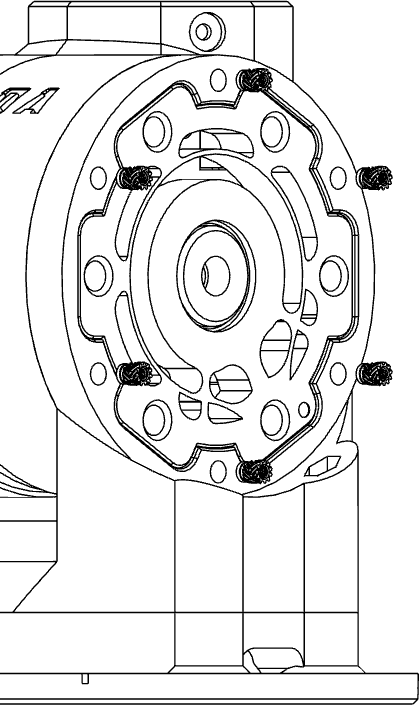
using **4 M3 x 30** screws and **4 M3 nuts**.

#### A picture containing linedrawing Description automatically generatedNOTE !

Pay attention to the mounting direction of the **AX1-A-KRA-1** part, the largest groove must be on the back side.

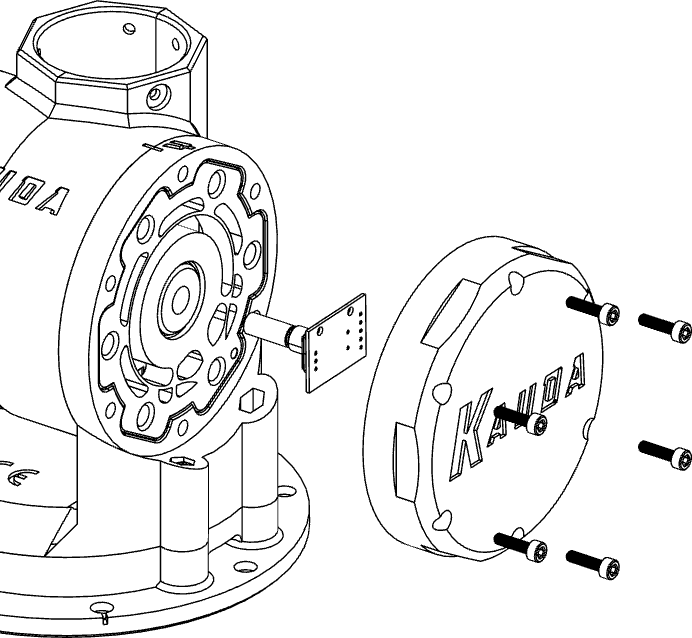


**13**

Insert a total of **12 M3 inserts** (6 per side) on the **AX1-B-KRA-1** and **AX1- L-KRA-1** using a soldering iron.

**X8**

**14** Place the encoder inside the special slot of **SPT-KRA-1**, and then fix the cover of the **FPLT-R-KRA-1** axis with a total of **6 M3 x 16** socket head screws.



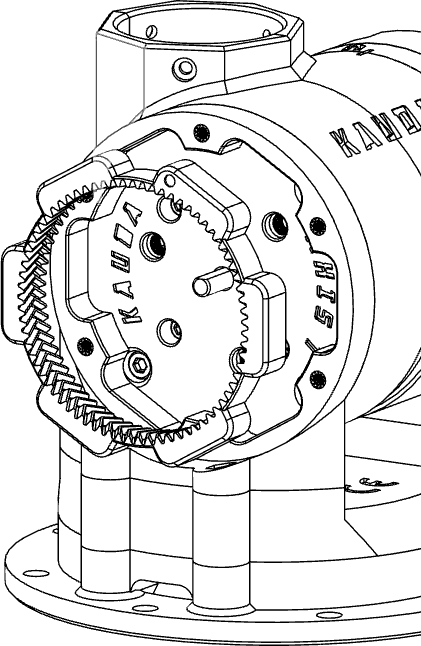
**X6**

Place **RG-KRA-1** inside **AX1-L-KRA- 1**, paying attention to the direction of the teeth, it must be as shown in the

figure.



**15**

Mount the 3 **PLT-SF-KRA-1** on **AX2-A-KRA-1**,



**16**

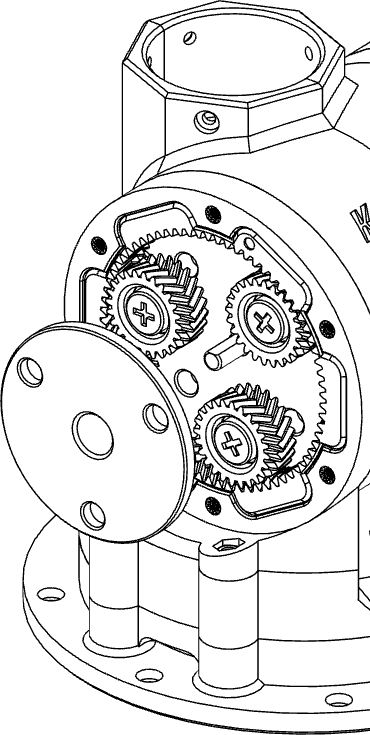
**X3**

first inserting 3 **686ZZ** bearings inside the gear wheel and then fasten everything with **3 M3 x 20** screws.

#### Diagram Description automatically generatedNOTE !

Do not force the screw, it is necessary to keep the waist slightly loose to allow the rotation of the toothed wheel.

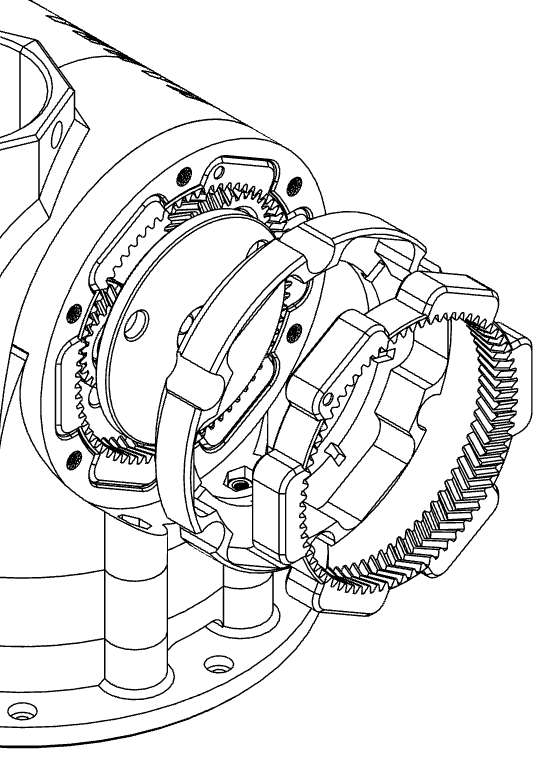
Insert **SUN-TW2-KRA-1** between the 3 **PLT-SF-KRA-1** sprockets.

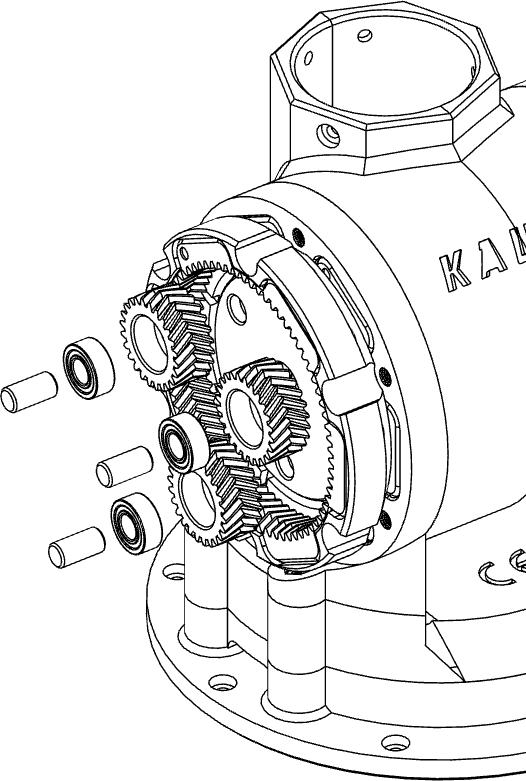


**17**

**NOTE !**

**18** Insert **RG-KRA-1** inside **RG-TW-KRA-1**, placing the whole on **AX1-L-KRA-1**.



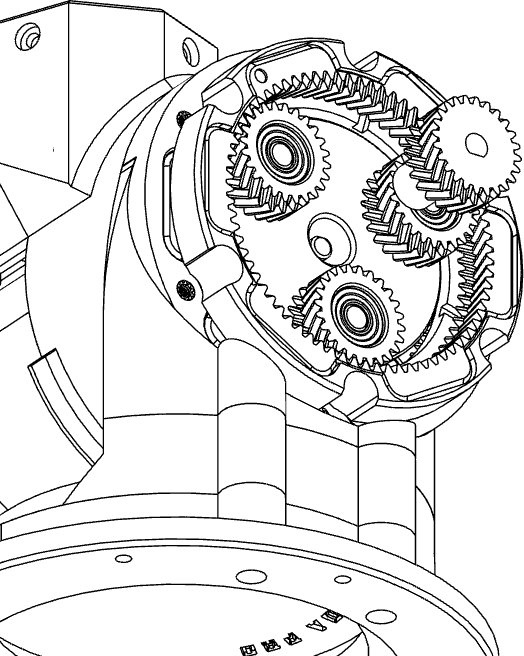
Insert 3 **608ZZ** bearings inside the **PLT- KRA-1** gear wheels and mount them on **SUN-TW2-KRA-1** using **TND-KRA-1** rods.

**X3**

The components must be assembled in the correct direction, pay attention to what is indicated in the images.



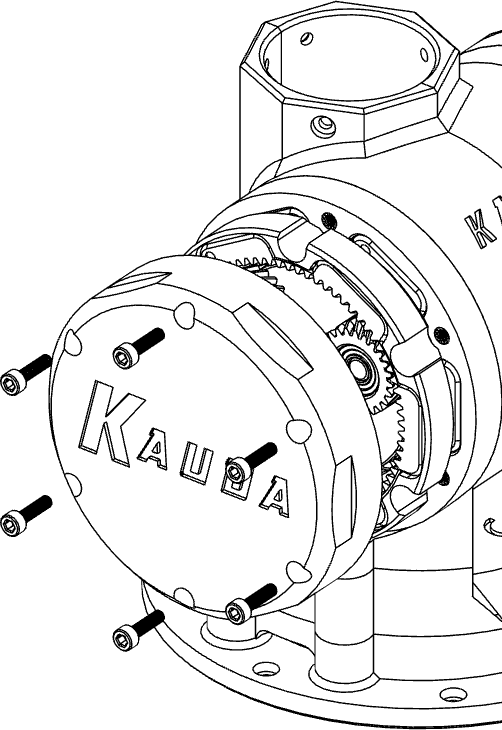
**19**

Insert **SUN-TW1-KRA-1** between the 3 **PLT- KRA-1** sprockets.



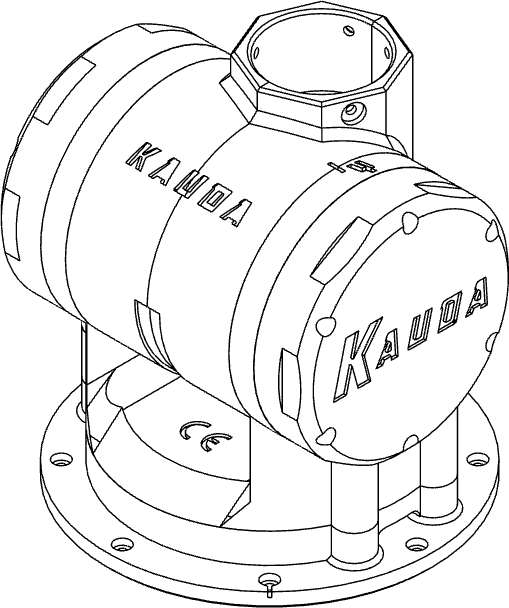
**20**

Fix the cover of the **21**



**X6**

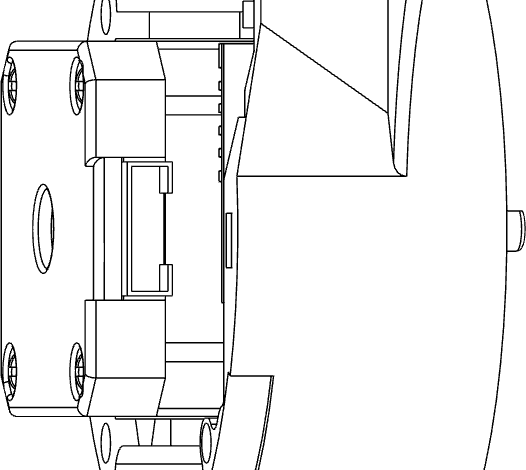
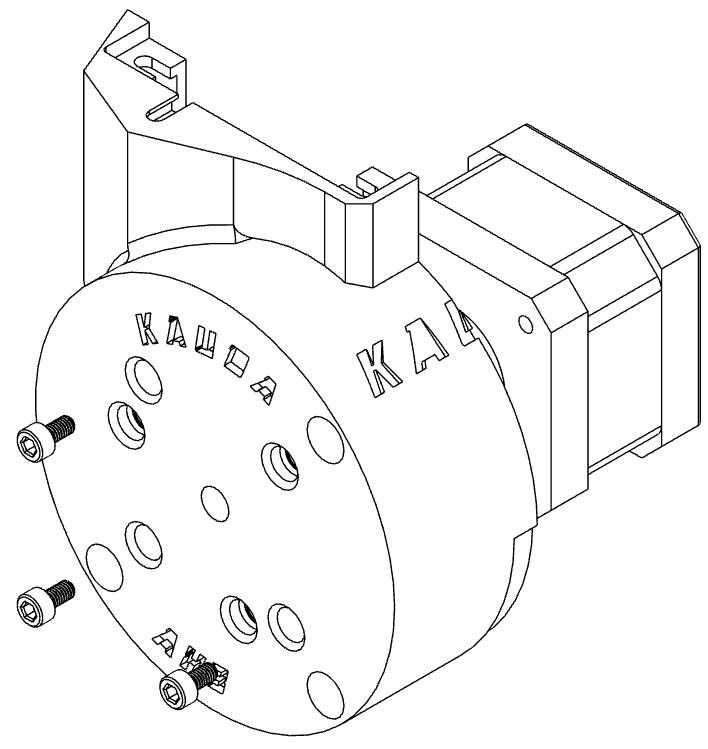
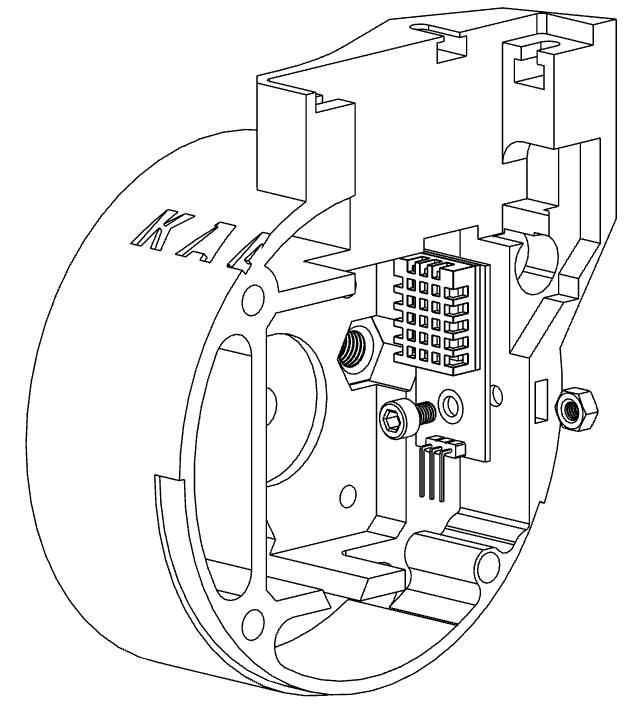
**FPLT-L-KRA-1** axis with a total of 6 socket head screws **M3 x 16**.



*Axis X*



Install the **DHT-11** sensor on the



**AX3-A-KRA-2** using a **M3 x 6** TCEI

screw and **M3 nut**.

Insert 3 **M6 nuts** inside the appropriate slots of **AX3-A-KRA-2**.



**2**

**3** Fix the **NEMA 17** motor to the



**1**

**X3**

**X3**

**AX3-A-KRA-2** using 3 **M3 x 8**

socket head screws.

**!**

#### NOTE

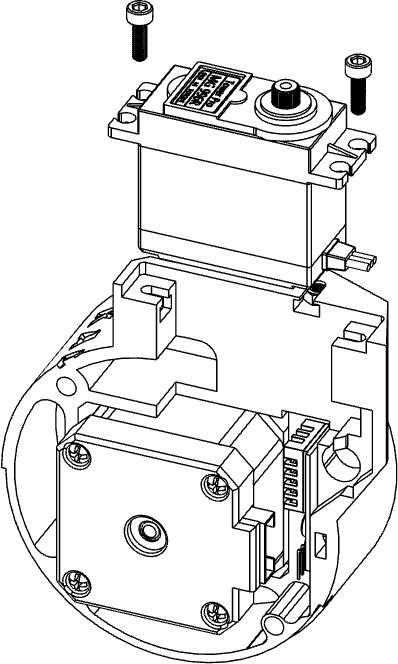
Pay attention to the motor mounting direction, the cable must be facing the rear hole.



Fix the **MG996R** servo motor inside the special cavity of the **AX3-A- KRA-2** axis using two **M3 x 10**



**4**



**X**

appropriate slots of **AX3-B-KRA-2**.

**3**



**5**

**X3**

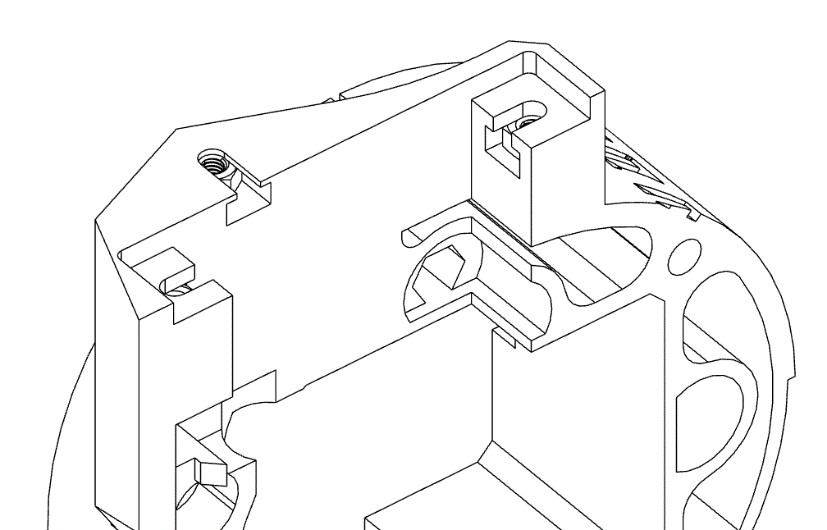
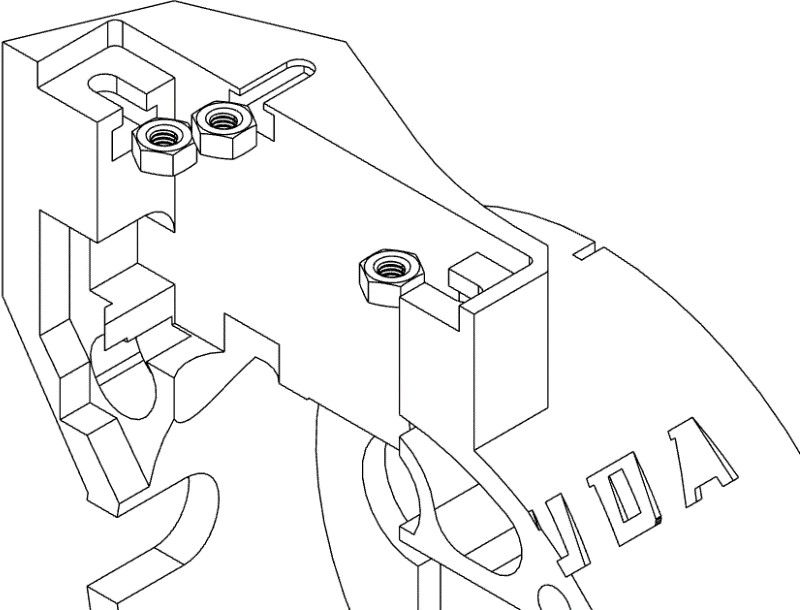
**X2**

socket head screws.

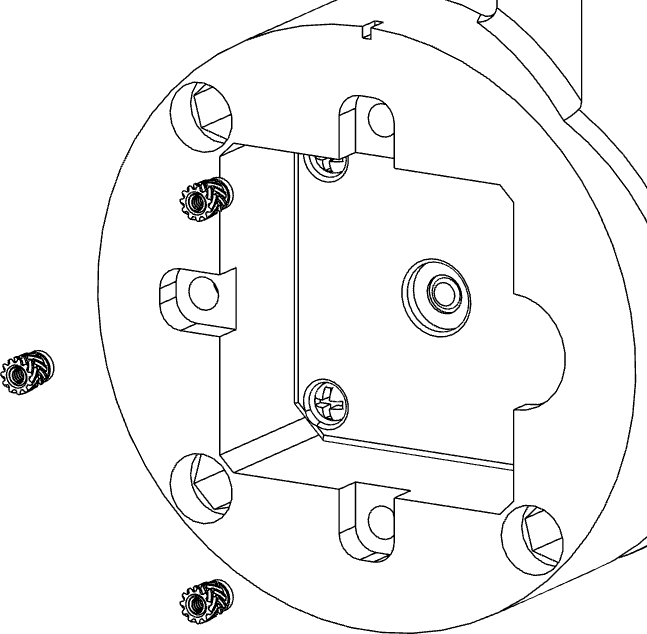
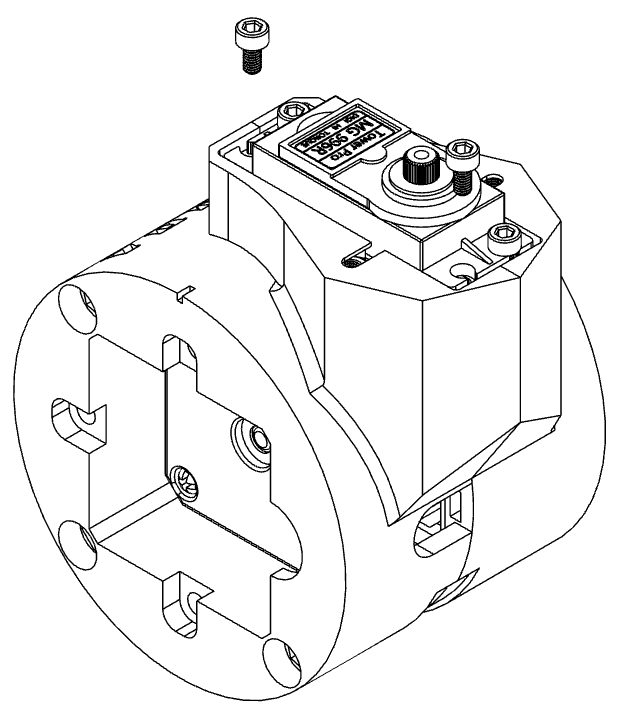
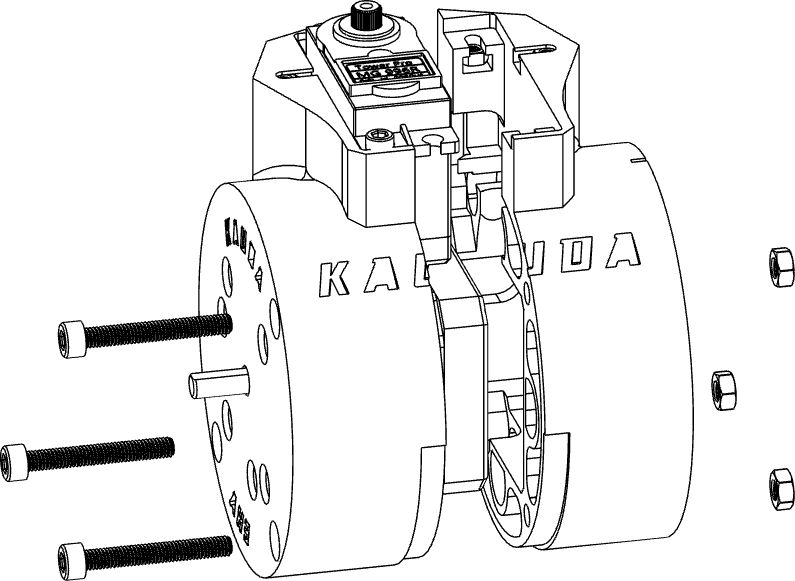
Insert 3 **M3 nuts** inside the appropriate slots of **AX3-A-KRA-2**.



**6**



Insert 3 **M3 nuts** inside the



**7** Mount **AX3-A-KRA-2** and **AX3-B-**

**X3**

**KRA-2** using **3 M4 x 40** screws and

#### 3 M4 nuts.

Fix the **MG996R** servo motor inside **8**

**X2**

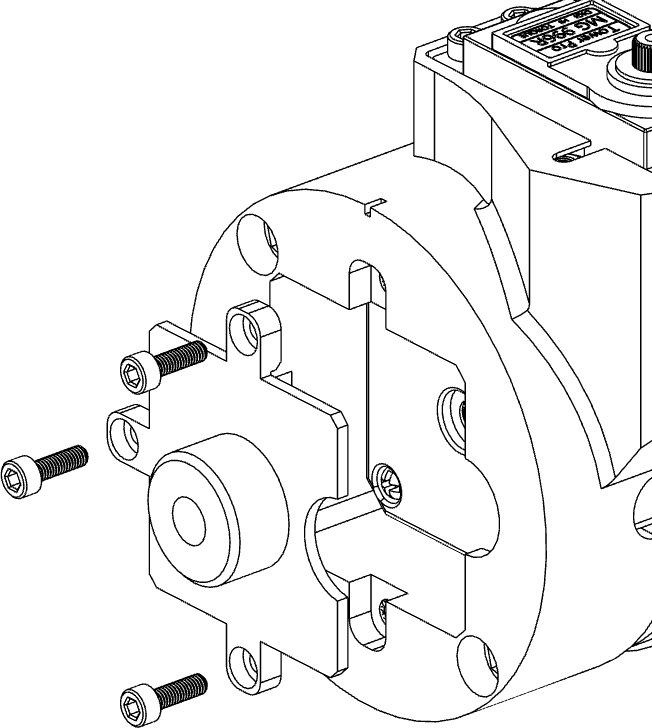
the special cavity of the **AX3-B- KRA-2** axis using two **M3 x 10**

socket head screws.

**9** Insert 3 **M3 inserts** inside the special cavities of the **AX3-B-KRA-2** by heating the inserts with a soldering iron.

**X3**

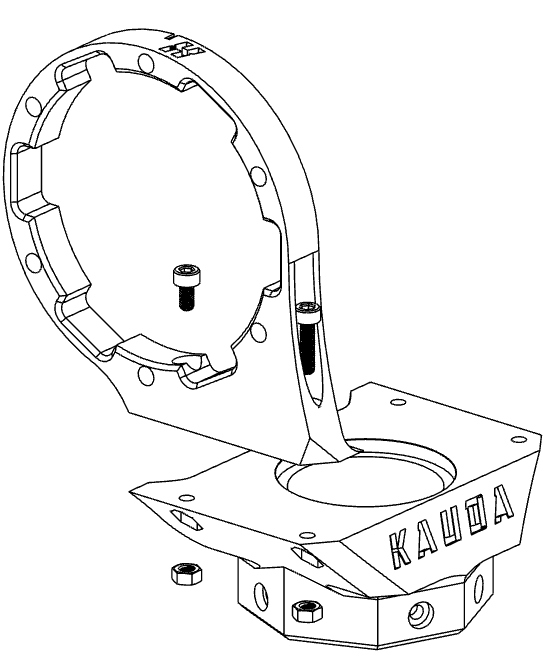
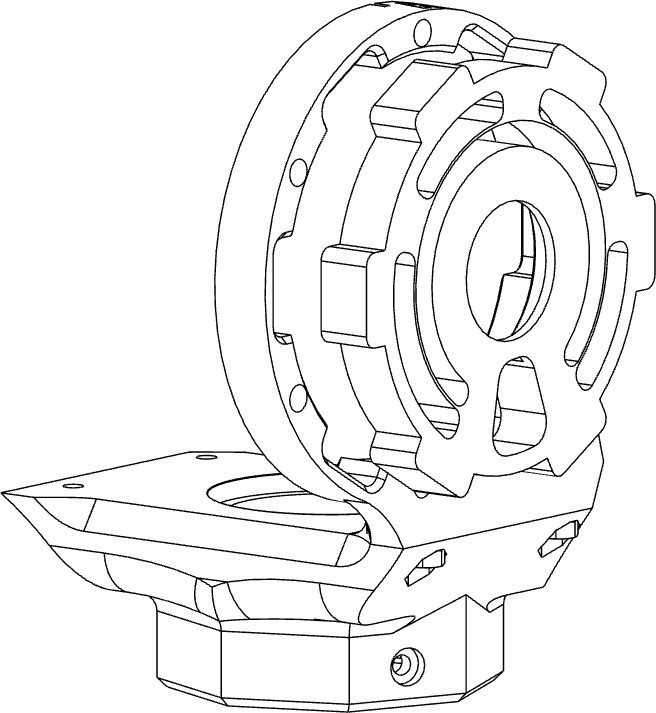
Install **SPT-KRA-2** on **AX2-B-KRA-2 10**



**X4**

using **3 M3 x 10** socket head screws.

**11** Fix **AX2-R-KRA-2** to **AX2-A-KRA-2**



**X2**

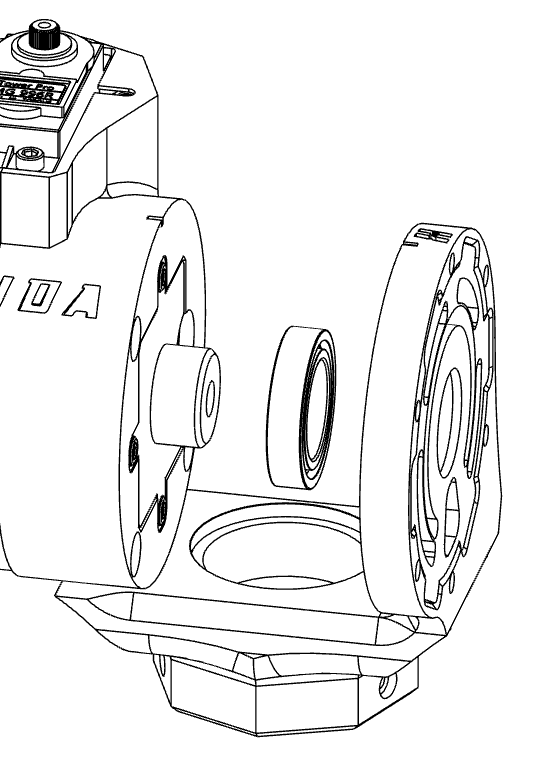
using one **M3 x 8** socket head screw and **M3 x 16** socket head screw, plus 2 **M3 nuts**.

Mount **RG-FL-KRA-2** on **AX2-R- 12**

**KRA-2**, in the direction shown in the

figure.

Mount **AX2-R-KRA-2** to **AX3-B-**



**13**

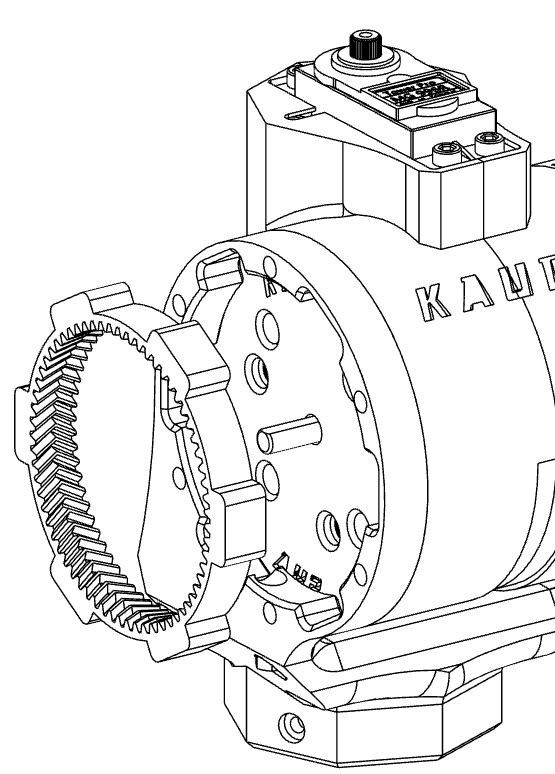
**KRA-2** by inserting the **6806ZZ** bearing in half, inside the appropriate slot.

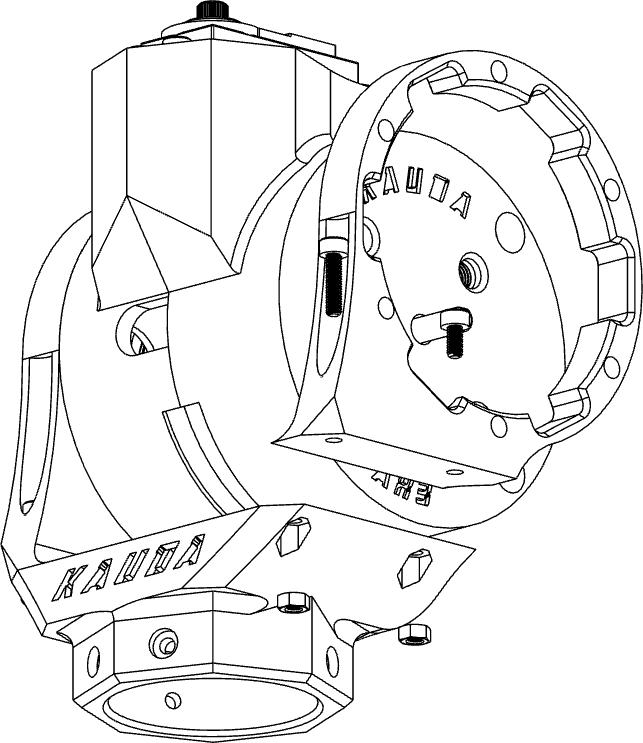
Fix **AX2-L-KRA-2** to **AX2-A-KRA-2**



**14**

**X2**

using one **M3 x 8** socket head screw and **M3 x 16** socket head screw, plus 2 **M3 nuts**.

Mount **RG-KRA-2** on **AX2-L-KA-2**,



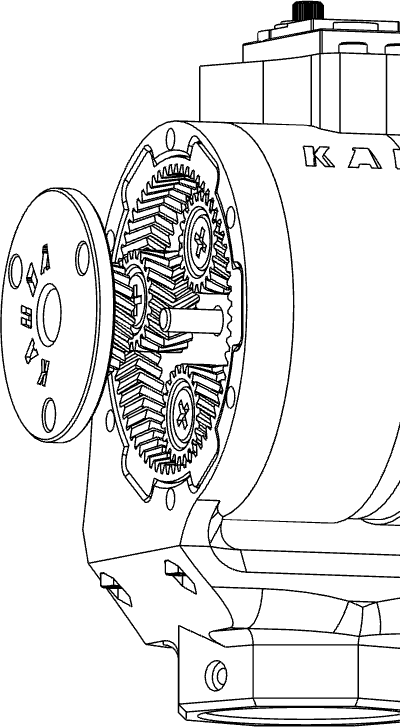
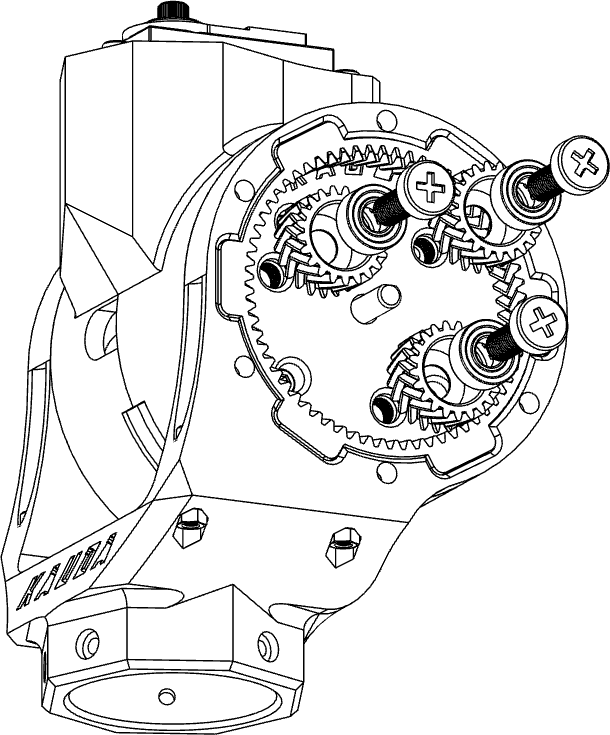
**15**

as shown in the figure.

#### NOTE !

Pay attention to the mounting direction of the

**RG-KRA-2** part. It must be as in the picture.



Mount the **3 PLT-SF-KRA-2** on **AX3- 16**

**X3**

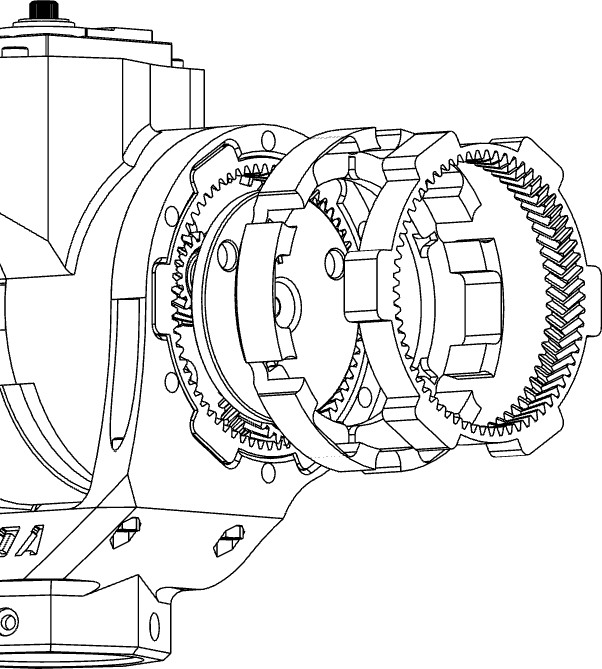
**A-KRA-2**, first inserting 3 **686ZZ** bearings inside the gear wheel and then fasten everything with 3 **M3 x**

**20** screws.

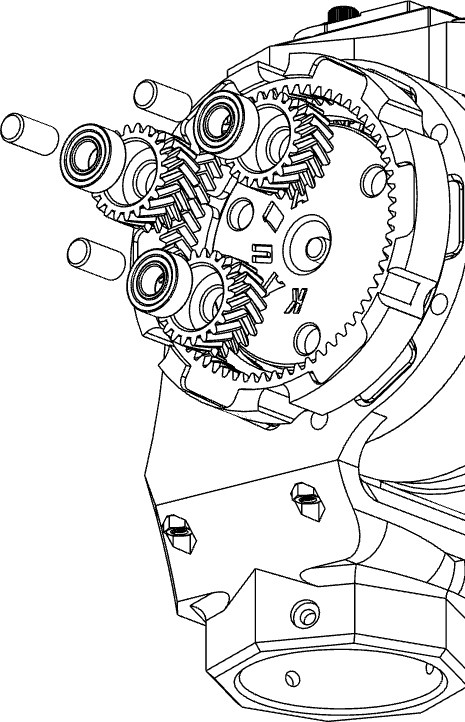
#### NOTE !

Do not force the screw, it is necessary to keep the waist slightly loose to allow the rotation of the toothed wheel.

1. Insert **SUN-TW2-KRA-2** between the 3 **PLT-SF-KRA-2** sprockets.
2. Insert **RG-KRA-2** inside **RG-TW-KRA-2**, placing the whole on **AX2-L-KRA-2**.



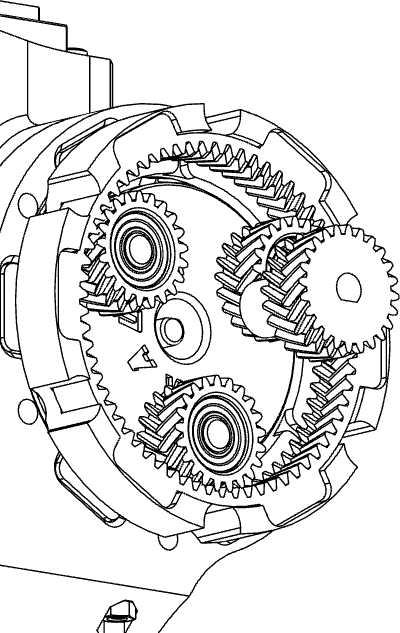
Insert 3 **608ZZ** bearings inside the **19**



**X3**

**PLT-KRA-2** gear wheels and mount them on **SUN-TW2-KRA-2** using

**TND-KRA-2** rods.

Insert **SUN-TW1-KRA-2** between the 3 **PLT- KRA-2** sprockets.

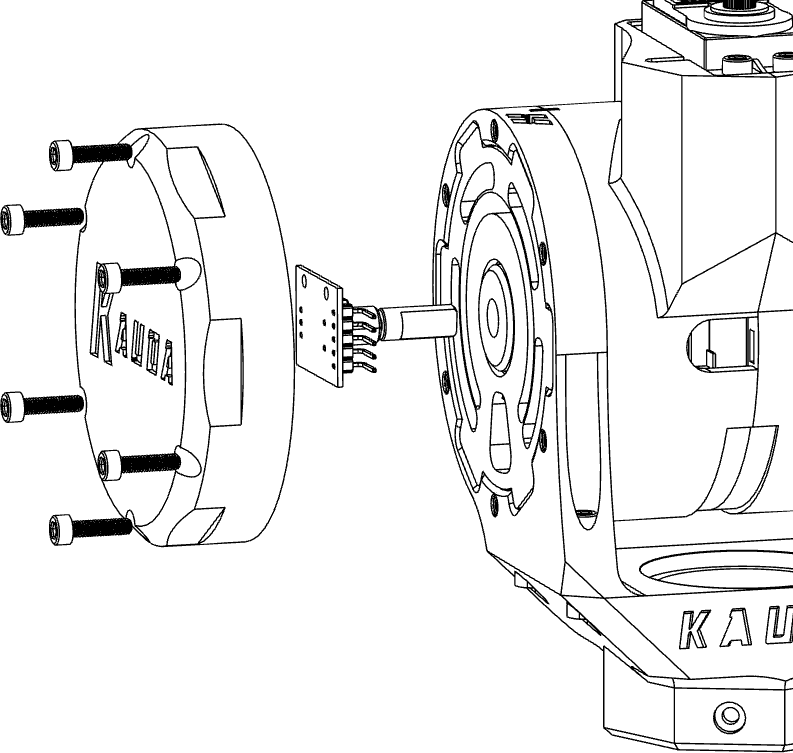


**20**

**21**

**X12**

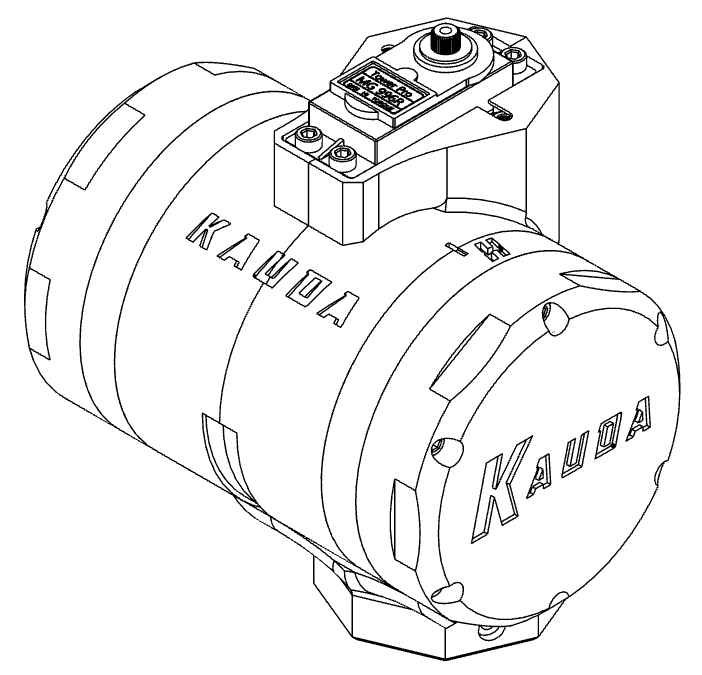
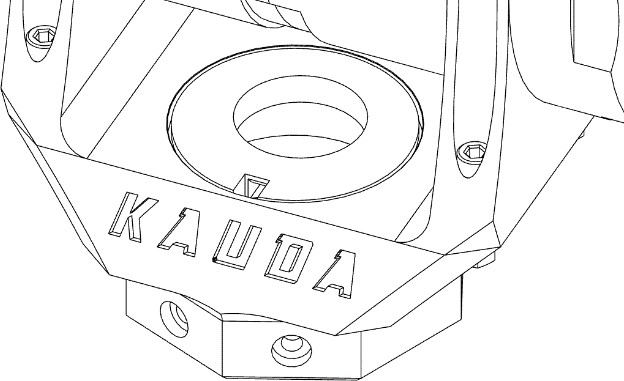
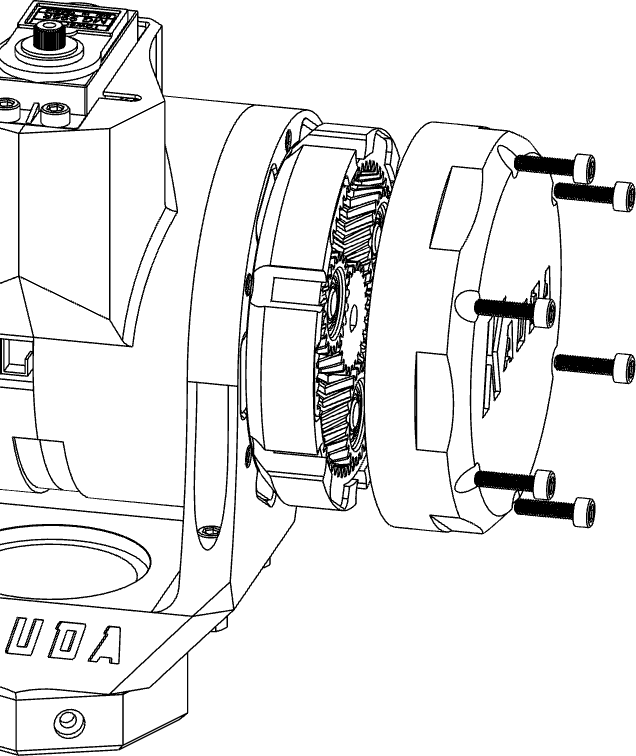
Insert a total of 12 **M3 inserts** (6 per side) on the **AX1-B-KRA-1** and **AX1-L-KRA-1** using a soldering iron.

Place the encoder inside the special slot of **SPT-KRA-2**, and then fix the cover of the **FPLT-R-KRA-2** axis with a total of 8 **M3 x 16** socket head screws.



**22**

**X6**



**23** Secure the cover of the **FPLT-L-KRA-2** axis with a total of 8 socket head screws **M3 x 16**.

**X6**

#### NOTE !

The cables must be run inside the **CVR- KRA-2** cavity.

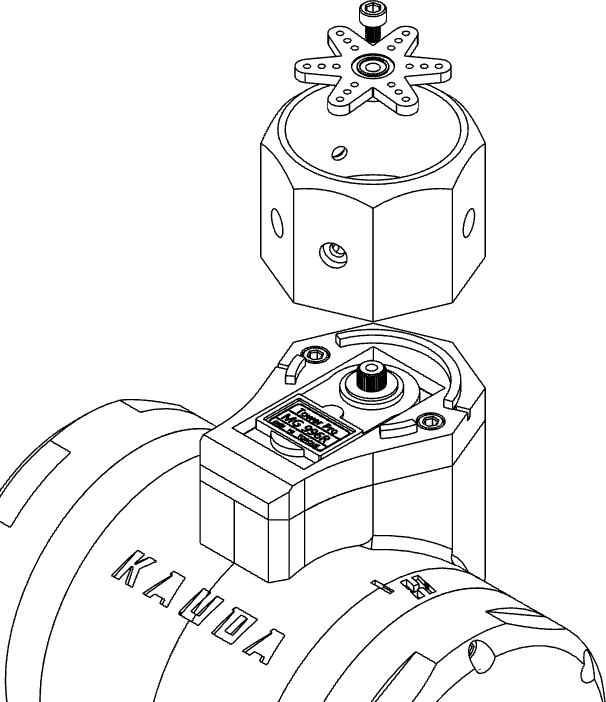


**24**

Mount **CVR-KRA-2** on **AX1-A- KRA-2**.

*Axis V*

**1** Secure **SRV-KRA-2** to the two parts that make up the X axis, **AX3-A / B-KRA-2**, with 2 **M3 x 10** socket head screws.



**X2**

Secure **AX4-A-KRA-2** to **SRV-KRA-2 2**

**X1**

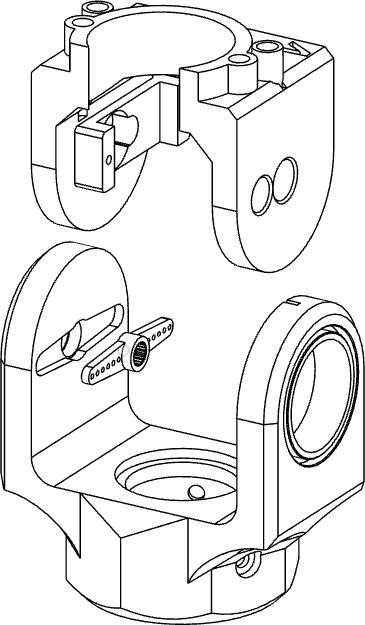
using the servo motor shaft and a **M3 x 6**

socket head screw.

#### NOTE !

It is recommended to bring the servo motor shaft to a central position, therefore at 90 ° to mount the **AX4-A-KRA-2** axis in a central position.

# Axis W

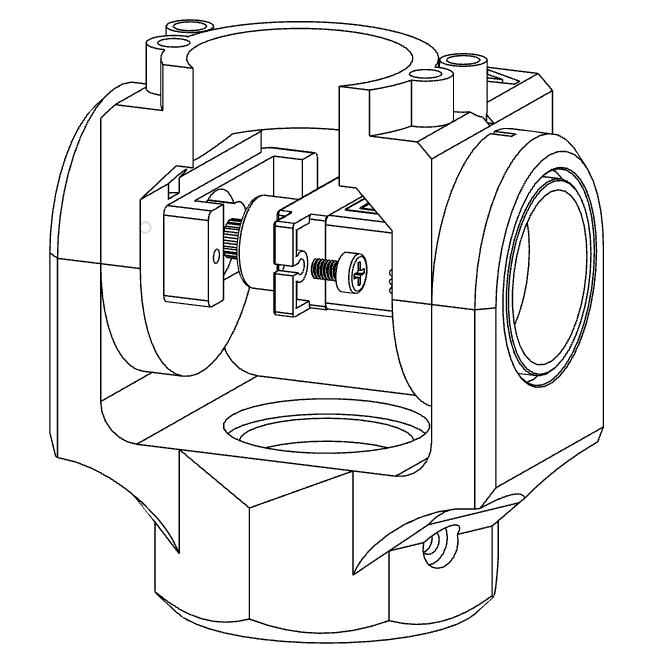
Fix the **MSG90** servo motor to the **AX5- KRA-3** axis using two 2.5 mm diameter

**X2**

threading screws.

Position the **AX5-KRA-3** axis on the **AX4-B- KRA-2**, taking care to also position the shaft of the **MSG90** servo motor which will subsequently be coupled.

Fix the servomotor in place using a **M2.5 x 4**



**2**



**1**



**3**

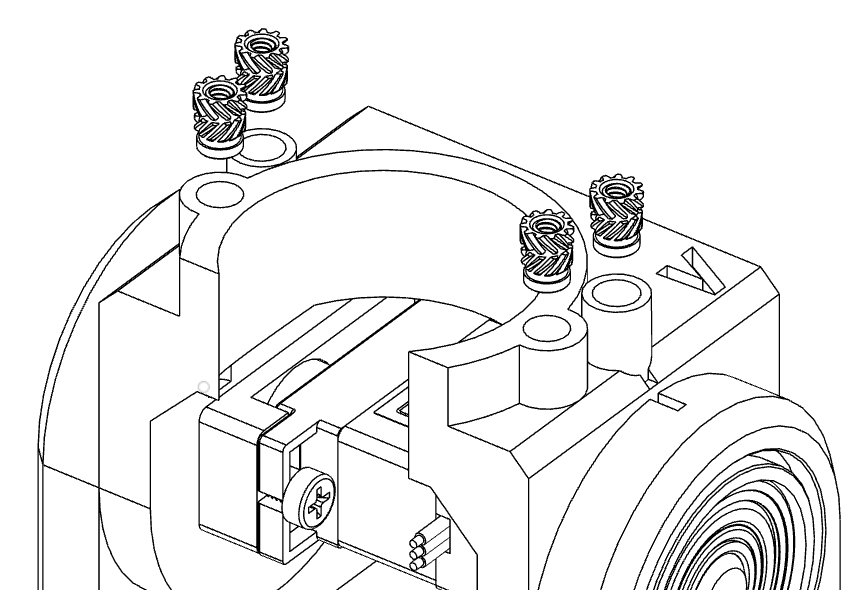
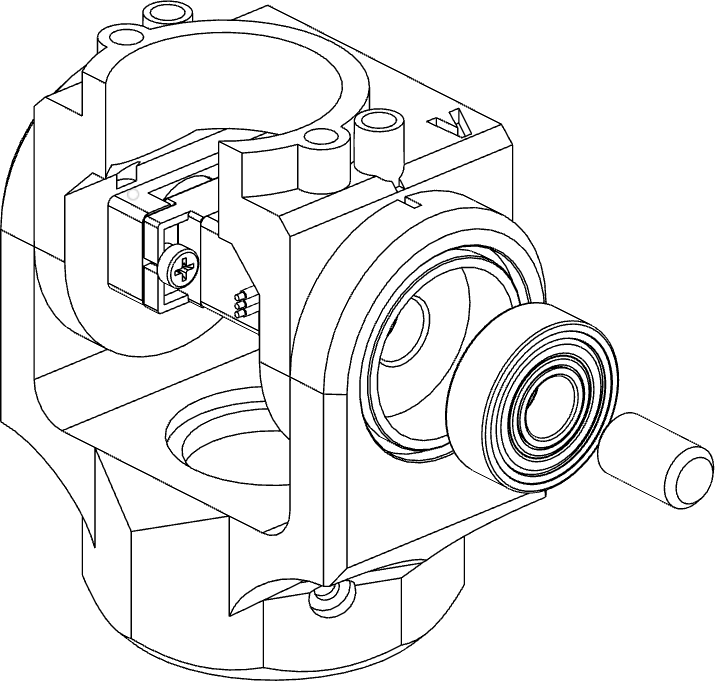
**X1**

socket head screw, as shown in the figure.

#### NOTE !

It is recommended to bring the servo motor shaft to a central position, therefore at 90 ° to mount the **AX5-KRA-3** axis in a central position.

Fix the **28BYJ-48** stepper motor to the



Assemble the **608ZZ** bearing and its

**TND-3-KRA-3** axis as shown in the

figure.

**4**

**5**

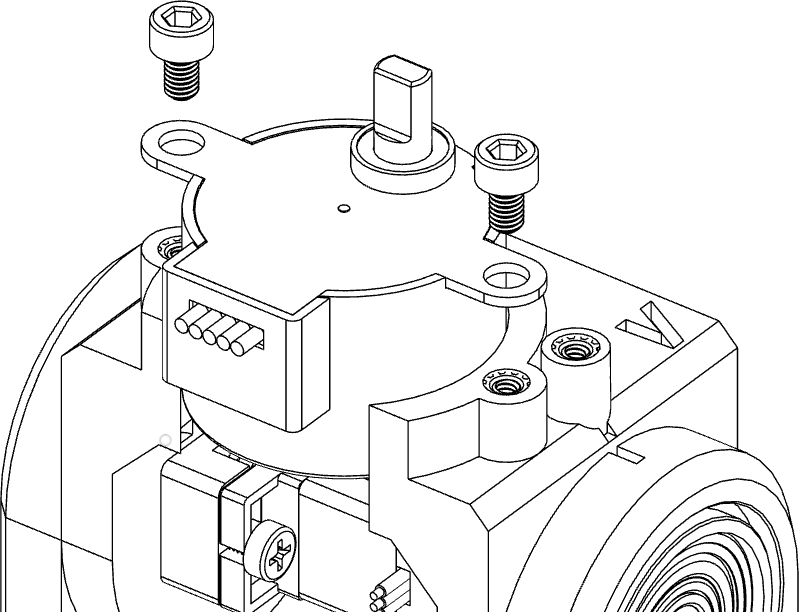
Insert **4 M3 inserts** inside the appropriate

**X**

cavities of the **AX5-KRA-3** axis using a

soldering iron.

**4**

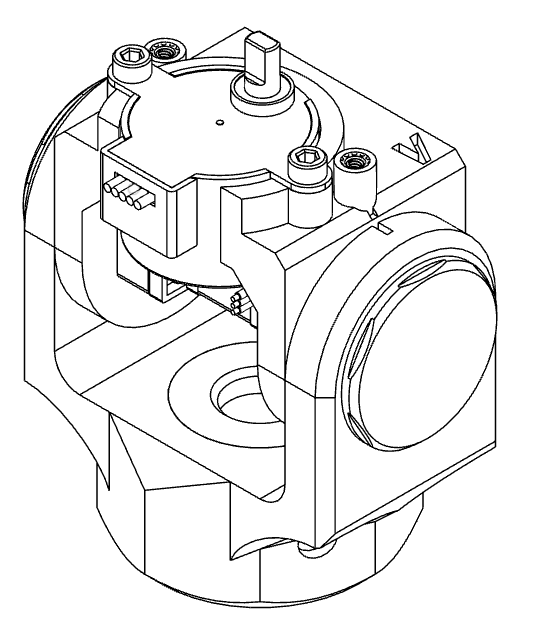
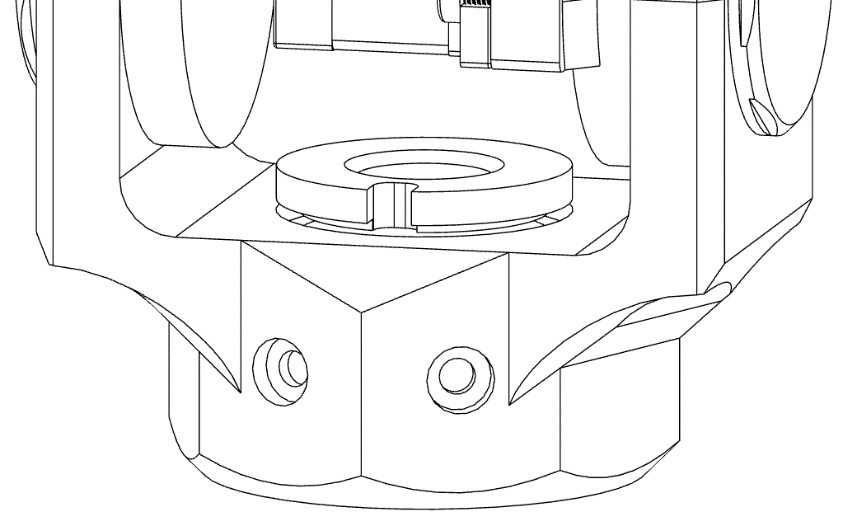
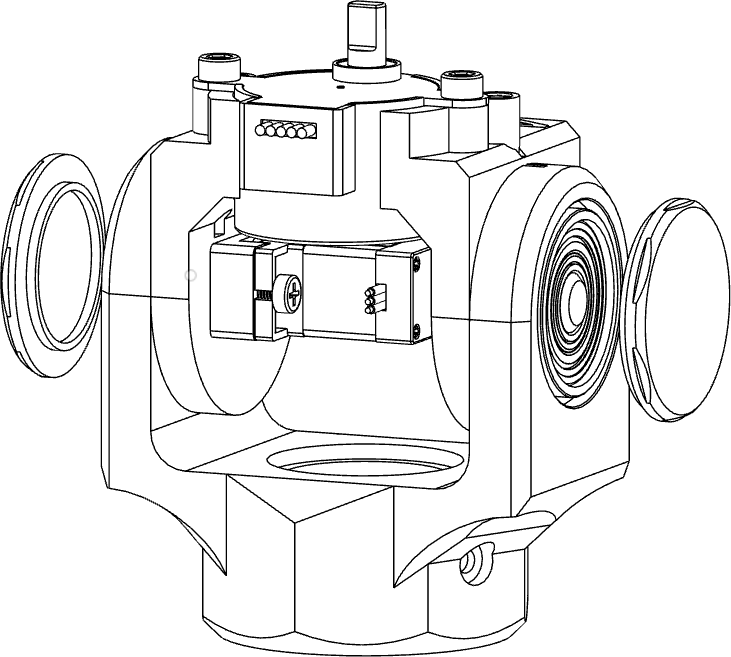


**6**

**X2**

**AX5-KRA-3** axis using two **M3 x 6** socket

head screws.



**7** Mount the protective covers **FPLT-L-KRA-3** and

**FPLT-R-KRA-3** on the **AX4-B-KRA-2** axis.

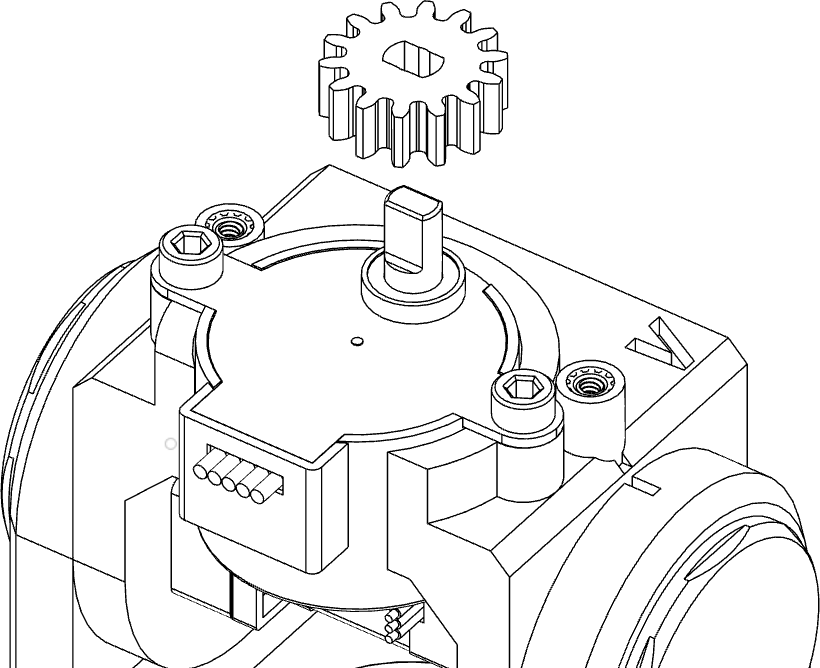
Mount **CVR-KRA-3** on **AX4-B-KRA-2. 8**

#### NOTE !

The cables must be run inside the CVR- KRA-3 cavity.

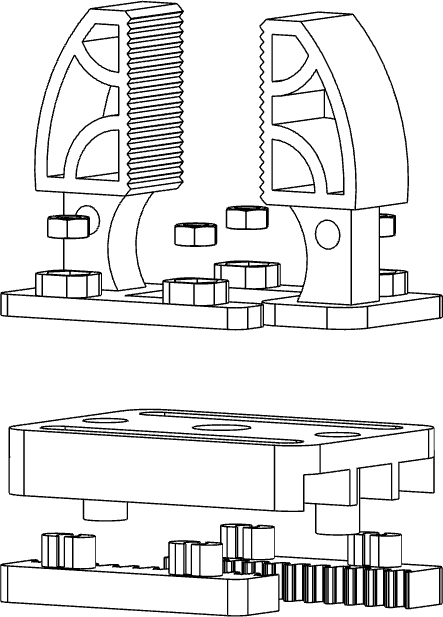
*Gripper*

Fix the gear wheel **GR-TLS-0** on the shaft of the



**1**

**28BYJ-48** motor.

Assemble the gripper by matching 2 **RK-TLS-0** with 2 **JW-TLS-0** parts, joining them using 4 **M3 nuts** and 4 **M3 x 10** socket head screws.



**2**

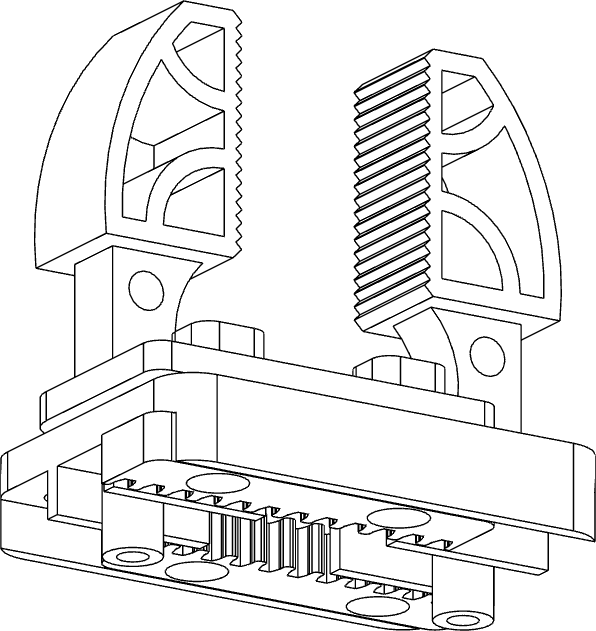
**X4**

Between the two parts previously mentioned

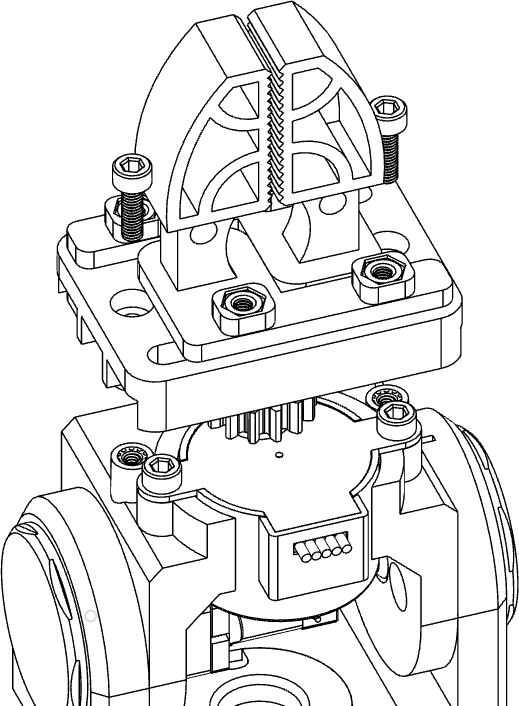
there is the **BS-TLS-0** base.

Check the correct direction of the components,

as shown in the figure.



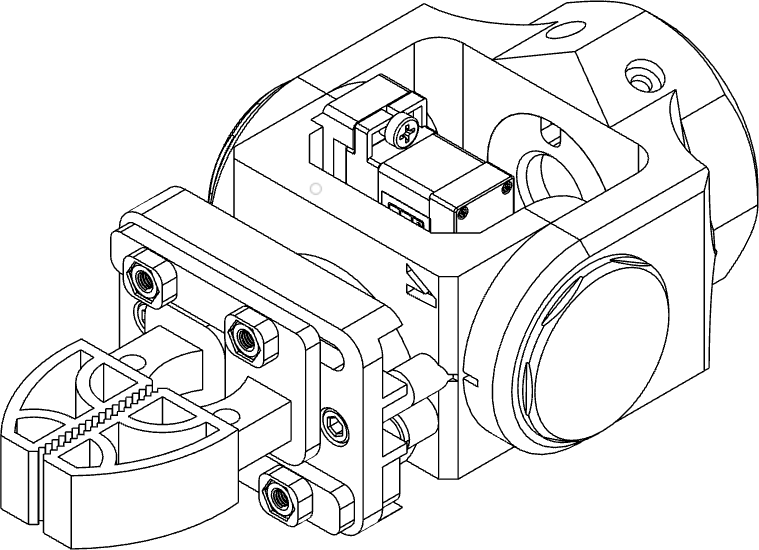
Fix the gripper previously mounted to axis 5 (**AX5-KRA-3**), using 2 **M3 x 10** socket head



**3**

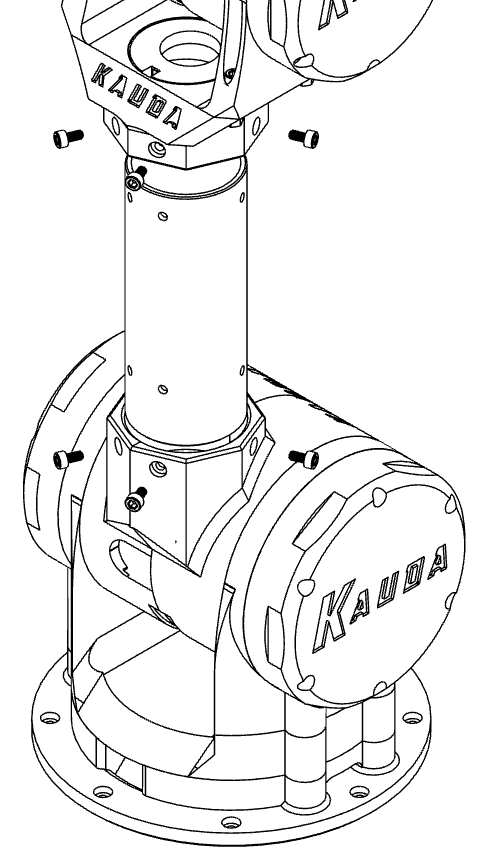
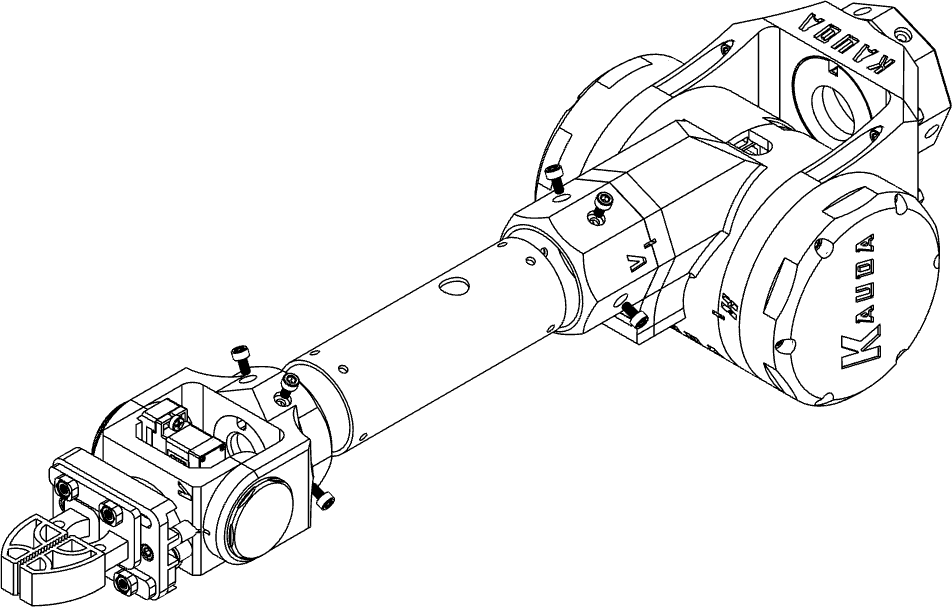
**X2**

screws.



*Axis Assembly*

Fix axis 5 to axis 4 with the tube ø35 mm and **1**



**X8**

100 mm long using 8 socket head screws **M3 x**

#### 8 mm.

**NOTE !**

At this stage, the cable must be passed inside the tube, and then it comes out of the

hole (F1). **F1**

## F2

**X8**

Fix axis 3 to axis 2 with the tube ø40 mm and **2**

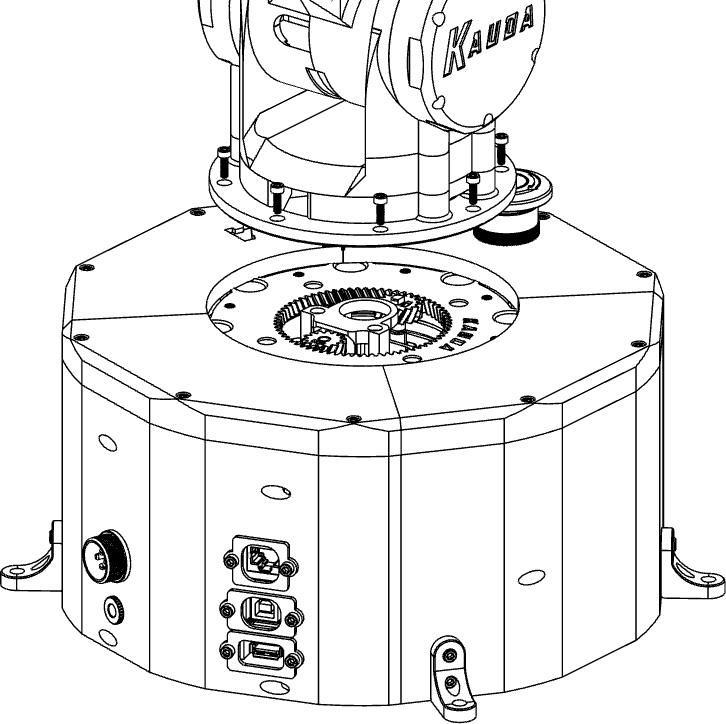
100 mm long using 8 **M3 x 8 mm** socket head

#### NOTE !

At this stage, the cable must be passed inside the tube (F2), and then it comes out of the hole (F3).

## F3

**3** Secure axis 1 to the base with a total of 8 **M3 x 8** socket head screws, as shown in the figure.

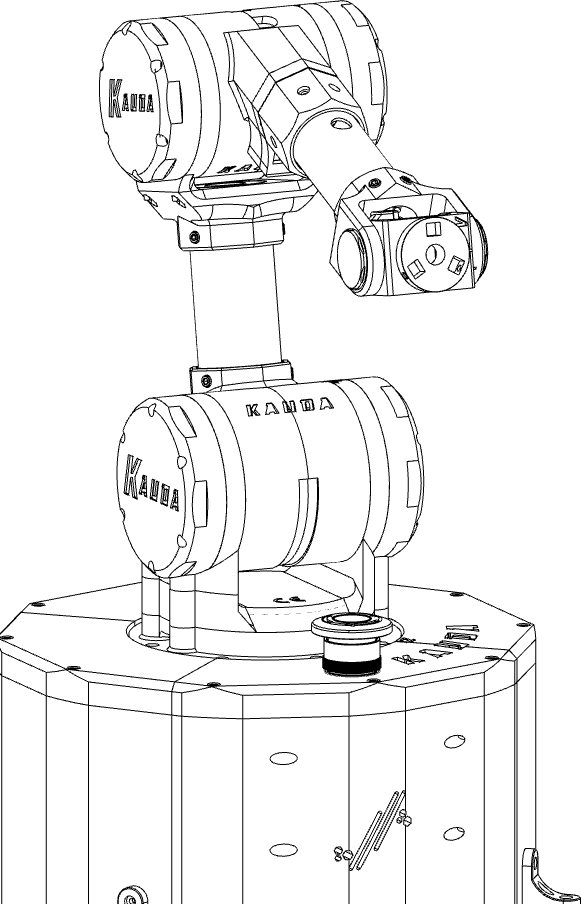


**X8**

#### NOTE !

The set of cables exiting from axis 1 (F4) must subsequently pass through the cavity of the base (F5) and through the hole (F6), otherwise, the rotation of the base could be faulty due to the cables that obstruct the movements.

Electrical components







|  |  |
| --- | --- |
| **KAUDA ROBOTIC ARM PRO V.** | |
| **PROJECT NAME** | 030-EL.DRW\_01 |
| **REV.** | 02 |
| **JOB** | 030-Kauda PRO |
| **P. SUPPLY** | 12V DC – 10A |

1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BILL OF MATERIALS** | | | | |
| **N°** | **Code** | **Description** | **Qty.** | **Note** |
| **1** | 17HS19-2004S1 | Stepper Motor 2A- 0,59Nm | 3 |  |
| **2** | MG996R | Servomotor 20Kg – Metal Gear | 1 |  |
| **3** | MG90S | Servomotor 6V – Metal Gear | 1 |  |
| **4** | Arduino Mega | Arduino Mega 2560 | 1 |  |
| **5** | Raspberry Pi3 | Raspbbery Pi 3 Board | 1 |  |
| **6** | KY-040 | Rotary Encoder | 3 |  |
| **7** | DHT-11 | Temperature/Humidity sensor | 3 |  |
| **8** | CNCSHIELD V3 | CNCSHIELD V3 Board | 1 |  |
| **9** | TMC2209 | Stepper Motor driver 2.8A | 3 |  |
| **10** | Emergency PB | Emergency PB Ø22 mm | 1 |  |
| **11** | USBPNLBFBM1 | USB 2.0B extension cable (panel mount) | 1 |  |
| **12** | USBPNLAFAM1 | USB 2.0A extension cable (panel mount) | 1 |  |
| **13** | n.a. | Ethernet extension cable (panel mount) | 1 | RJ45 Panel mount connector. |
| **14** | Breadboard | Breadboard Mini 400 Pin | 1 |  |
| **15** | LM2596S | DC/DC Converter | 1 |  |
| **16** | Jack connector | Front panel jack connector Ø11 mm – 24V 10A | 1 |  |
| **17** | GX20 | GX20 Connector Ø20 mm | 1 |  |
| **18** | Dupont connector | Dupont connectors to crimp | N |  |

2



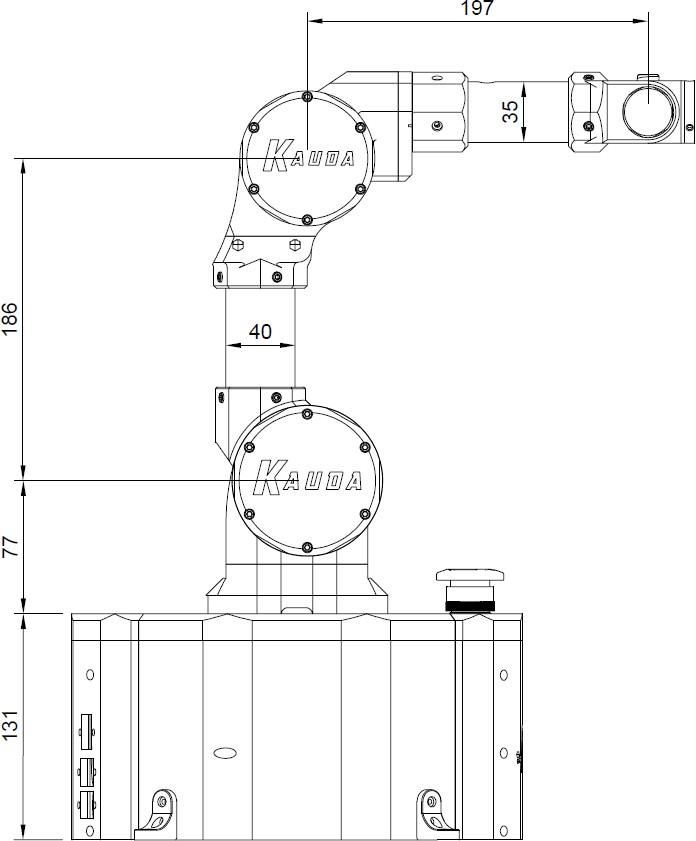
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BILL OF MATERIALS** | | | | |
| **N°** | **Code** | **Description** | **Qty.** | **Note** |
| **19** | Jack connector | Male Jack connector | 1 |  |
| **20** | 28BYJ-48 | Stepper Motor 5V | 1 |  |
| **21** | ULN2003 | Stepper Driver 5V | 1 |  |

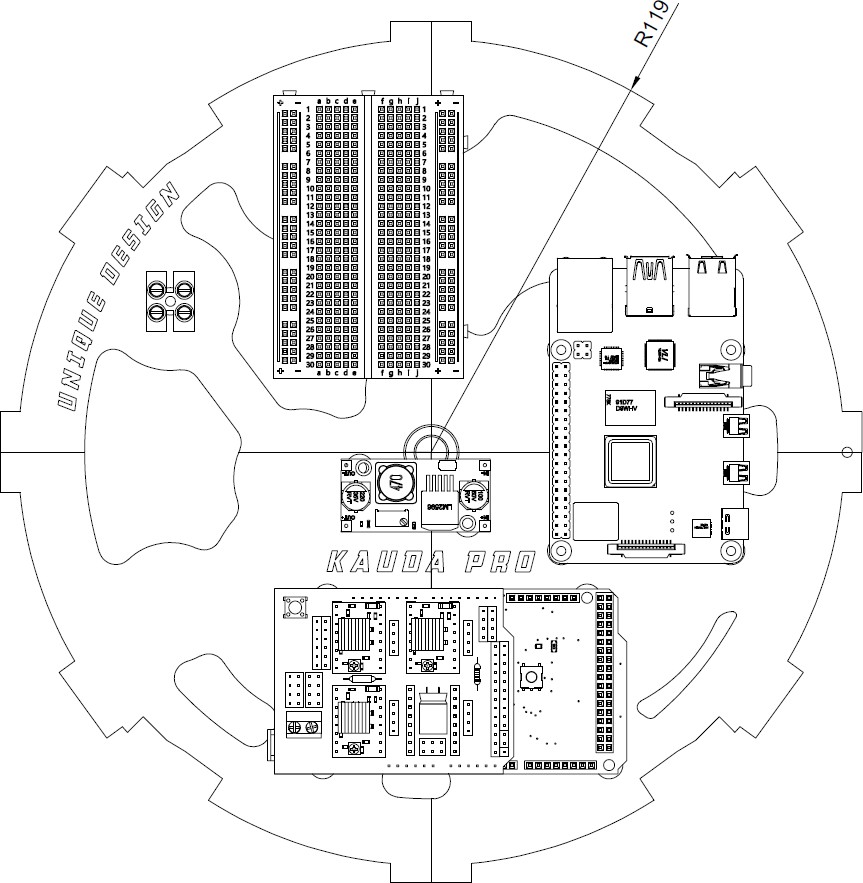


3

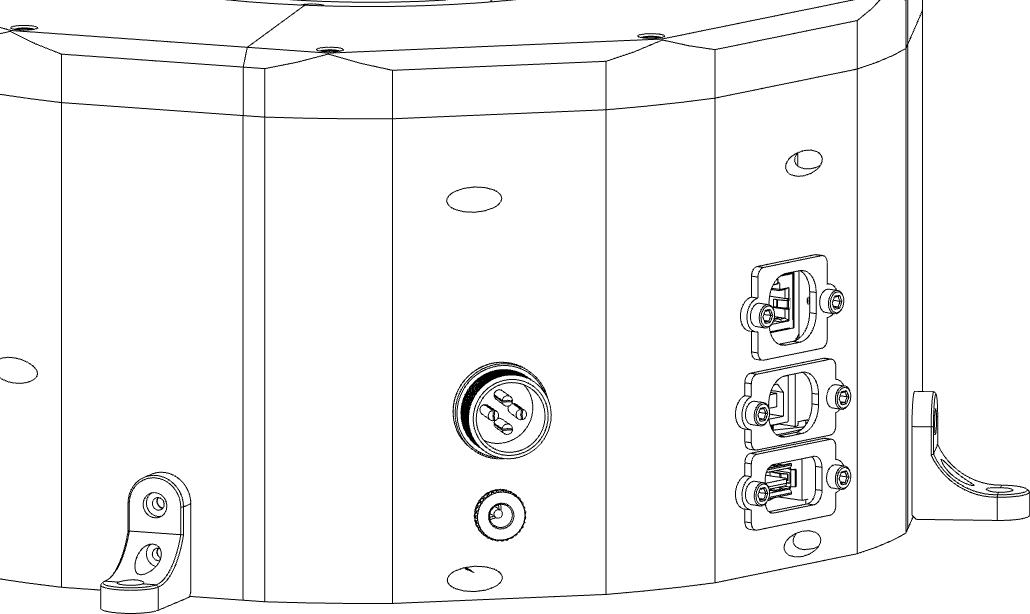


|  |
| --- |
| **LAYOUT** |
| V1.0 |

Measurements expressed in mm

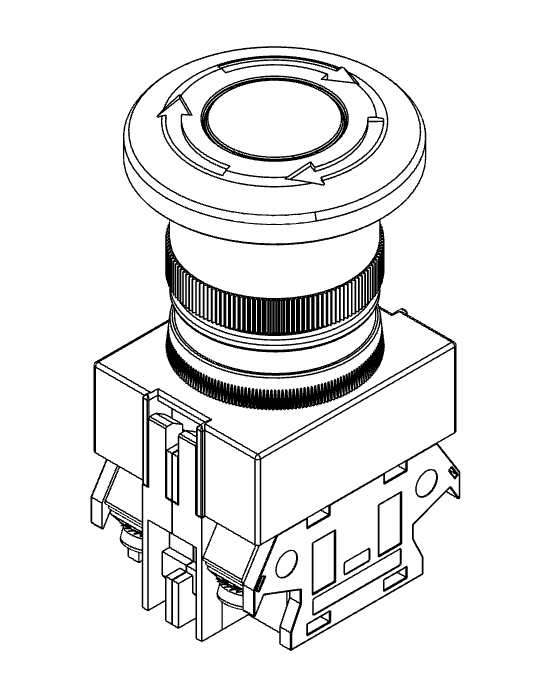
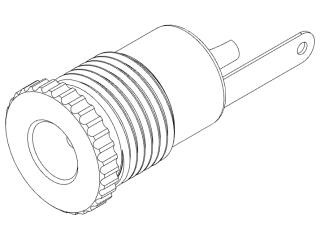
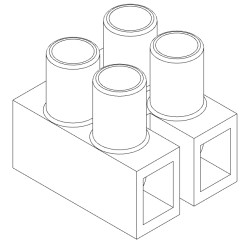
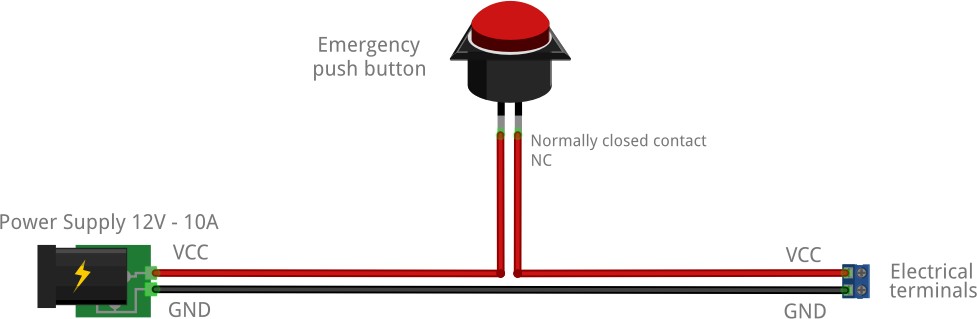


4



|  |
| --- |
| **POWER SUPPLY** |
|  |
| ***STEP 1***  The first step is to connect the female Jack connector (A) mounted on the wall of the robot base to the electrical terminals shown in the figure (C), by passing the cable with the 12V positive pole through the emergency push button (B).  The Jack connector will require two very simple but necessary soldering points. |

C



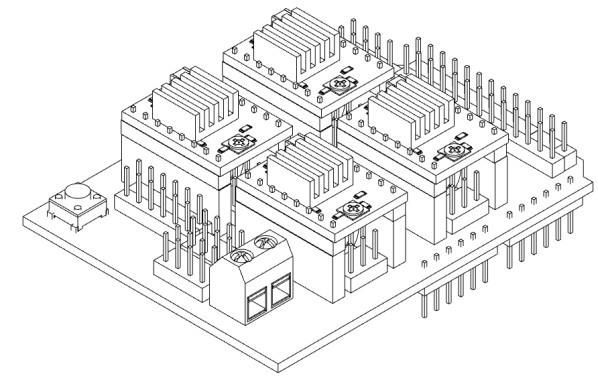
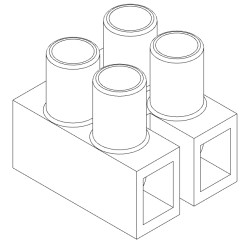
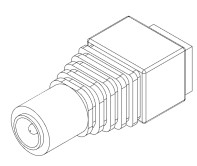
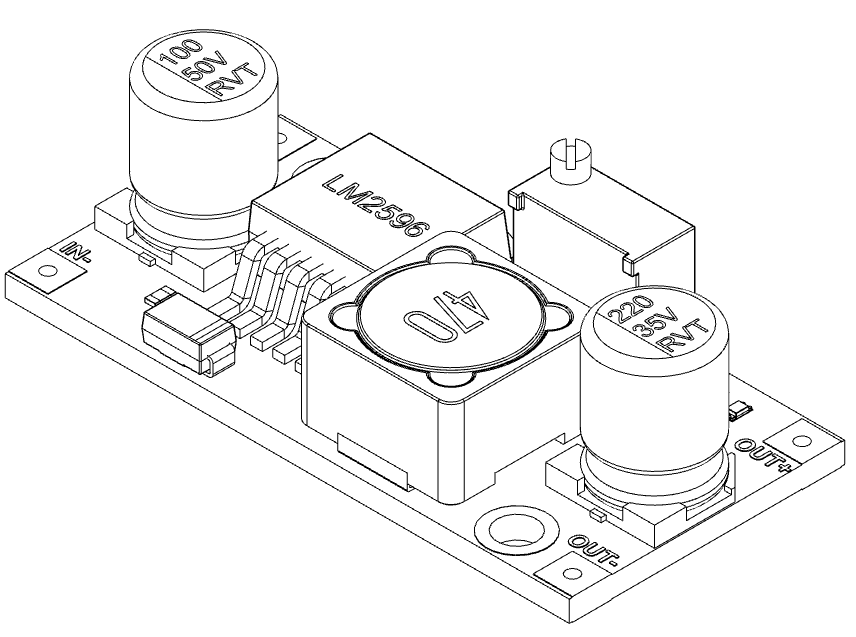
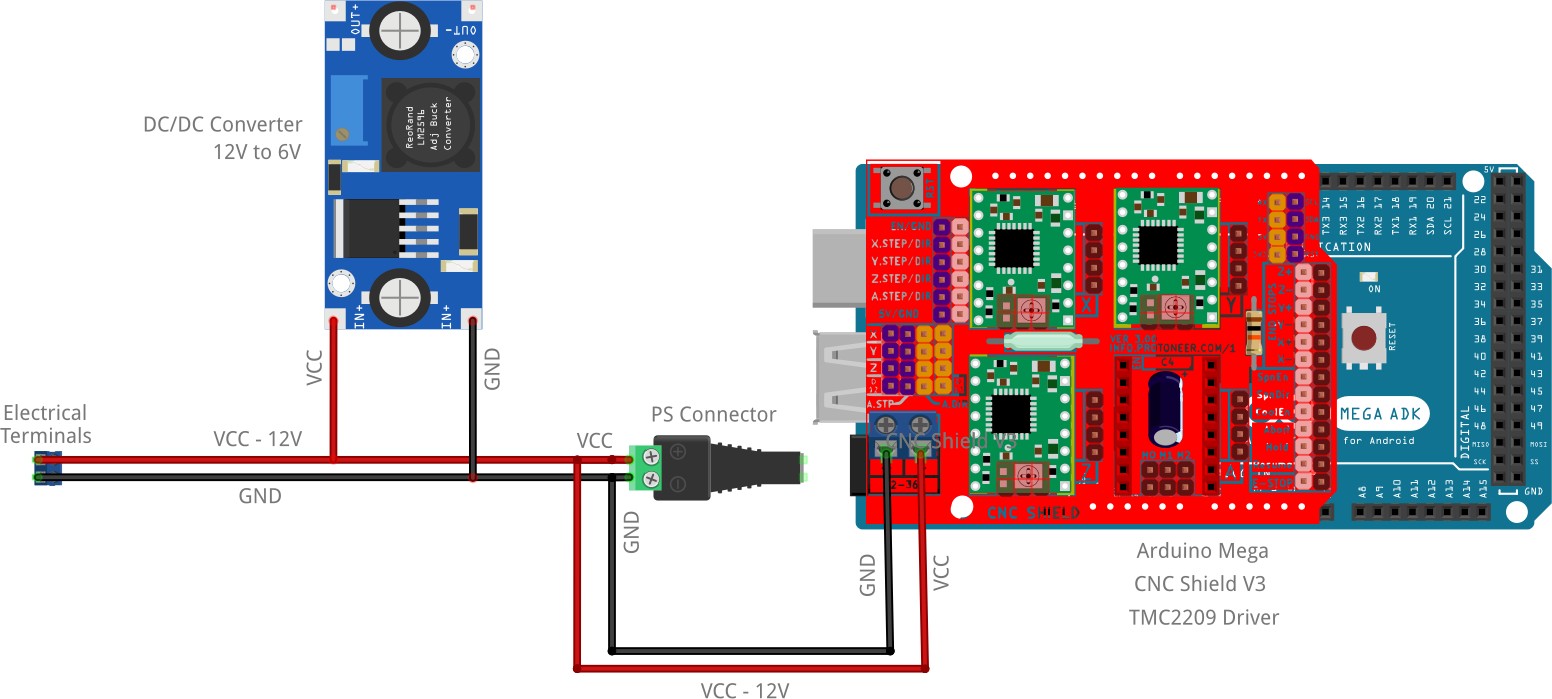
B

A

5



D



|  |
| --- |
| **POWER SUPPLY** |
|  |
| ***STEP 2***  Three different devices must be connected and jumpered from the electrical terminals (A), the male Jack connector (B), the DC / DC converter (C) (To the “IN+/IN-) and the CNC board (D).  Therefore, the cables will be split as shown in the figure.  ***Note:***  *The only of the three devices that requires two simple soldering points is the DC / DC converter (C), the other devices have dedicated terminals to which you can simply attach the end of the cable.* |

C

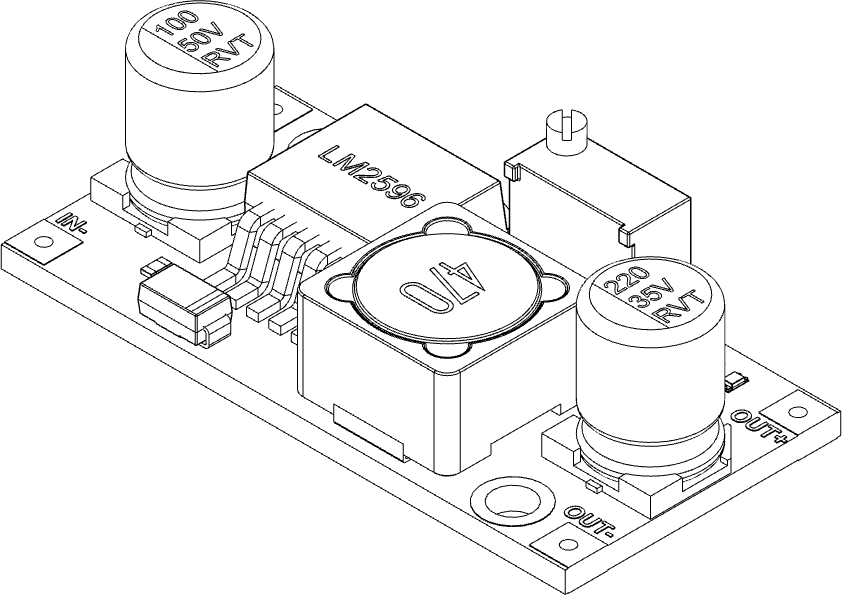
B

A

6

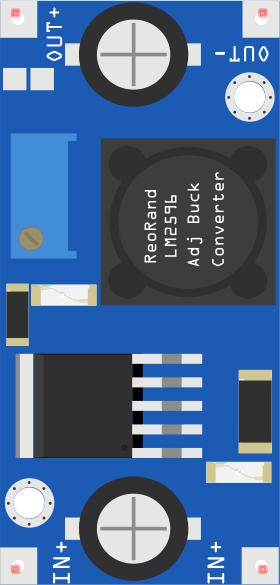


# b



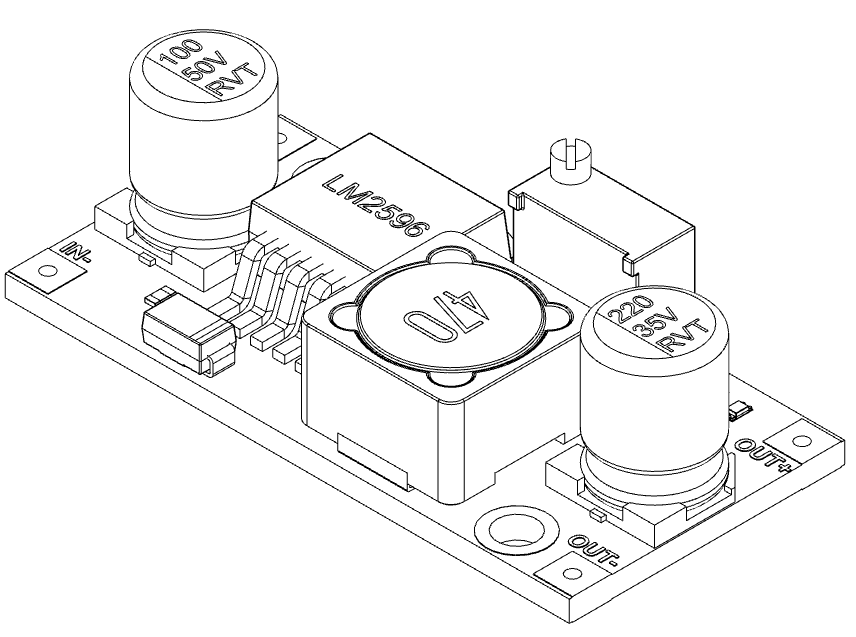
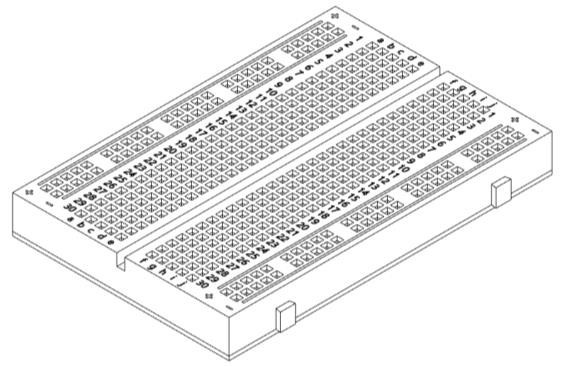
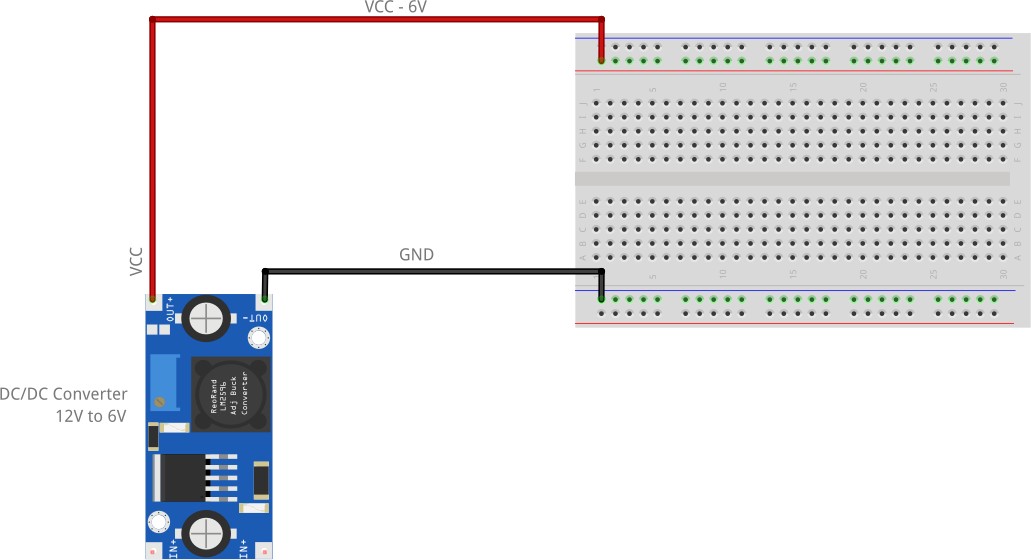
a

b



|  |
| --- |
| **POWER SUPPLY** |
|  |
| ***STEP 3***  Before connecting any device to the ends of the DC / DC Converter (A) it is necessary to adjust the voltage delivered in succession using a screwdriver and a multimeter.  First the circuit must be powered, therefore 12V will arrive on IN + / IN-  .  The terminals of the multimeter set on direct current at 20V must be placed to measure the output voltage, a value will be initially read that must be varied by acting on the small screw (B) indicated in the figure, rotating from one direction or the other.  The goal is to reach 6V output from  the module. |

7



B

a

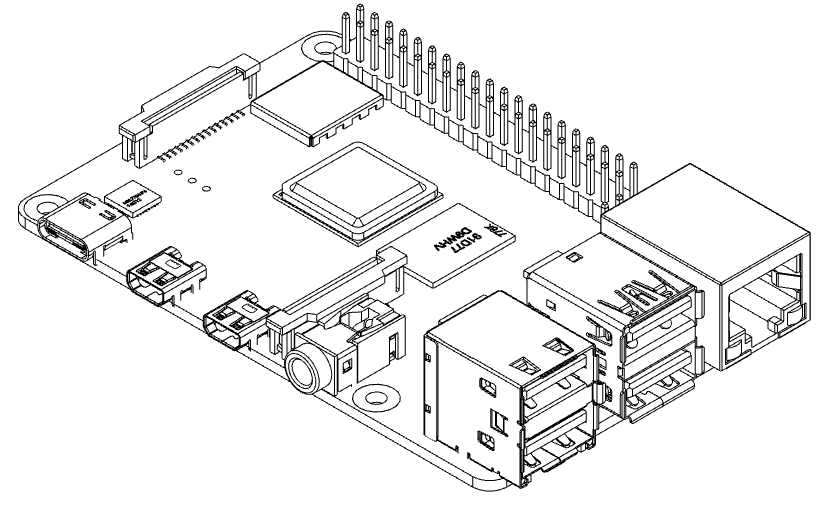
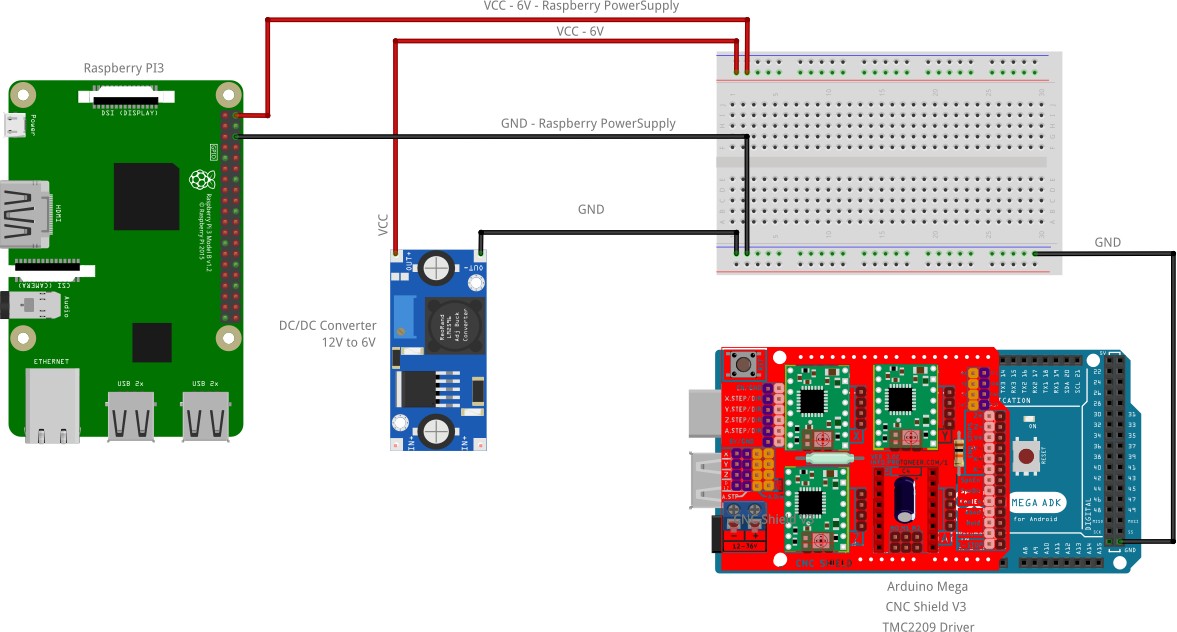
|  |
| --- |
| **POWER SUPPLY** |
|  |
| ***STEP 4***  From the output of the DC / DC Converter (A) the 6V previously adjusted will be brought to the Mini Breadboard (B) (standard half breadboard), and also in this case two soldering points on the converter will be necessary.  The breadboard is designed to ensure that by connecting the 6V to the terminals indicated in the figure, the voltage will automatically be distributed along the entire line of holes indicated either in red (+ 6V) or in blue (GND).  From this distribution point, the various devices that require a 6V power supply will be connected. |

8



|  |
| --- |
| **POWER SUPPLY** |
|  |
| ***STEP 5***  Subsequently, the RaspBerry Pi3(A) must be powered, bringing the 6V and the GND on PIN 2 (6V) and 6 (GND).  In addition to this, a GND must also be brought to the pin indicated with "GND" on the Arduino board. |

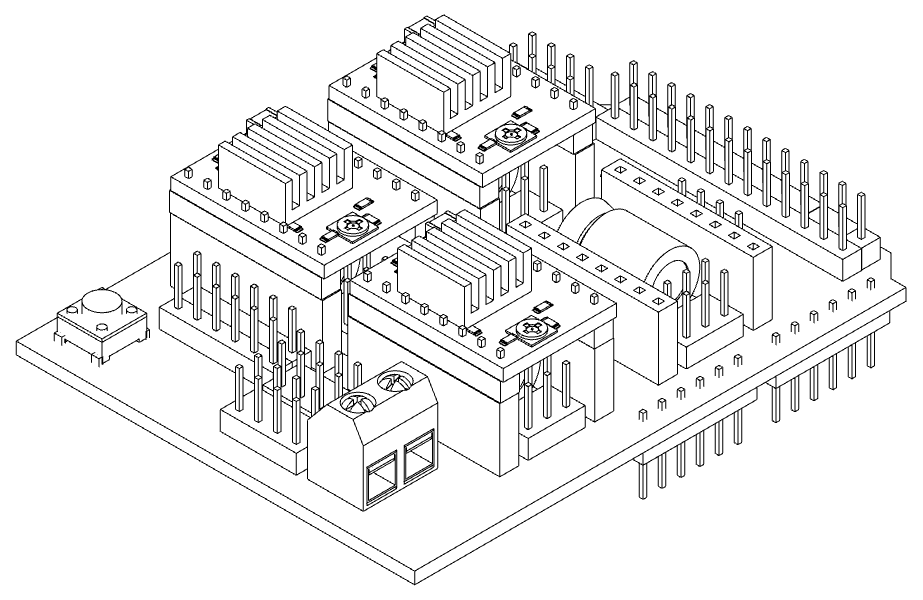
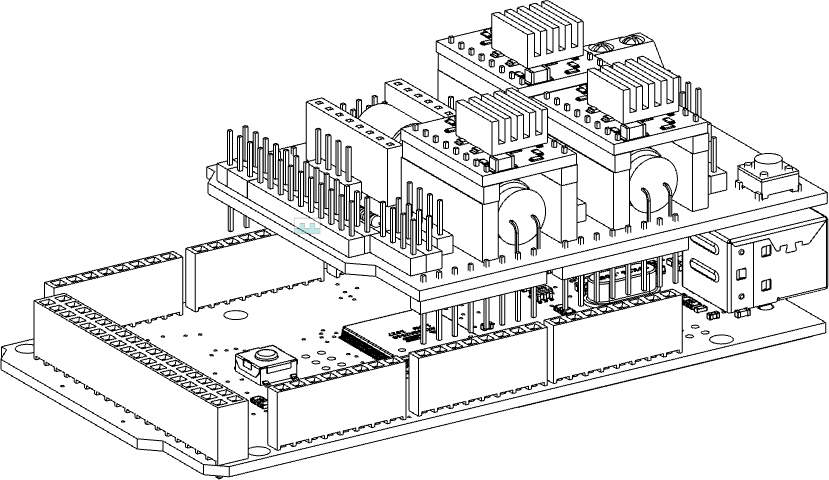
# a



9



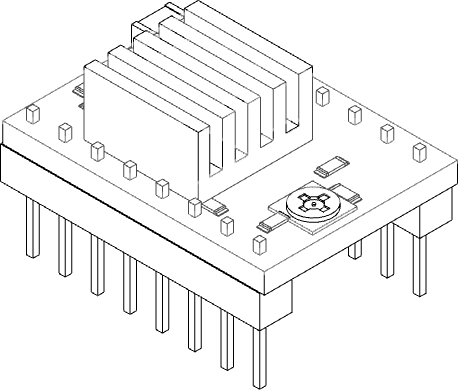
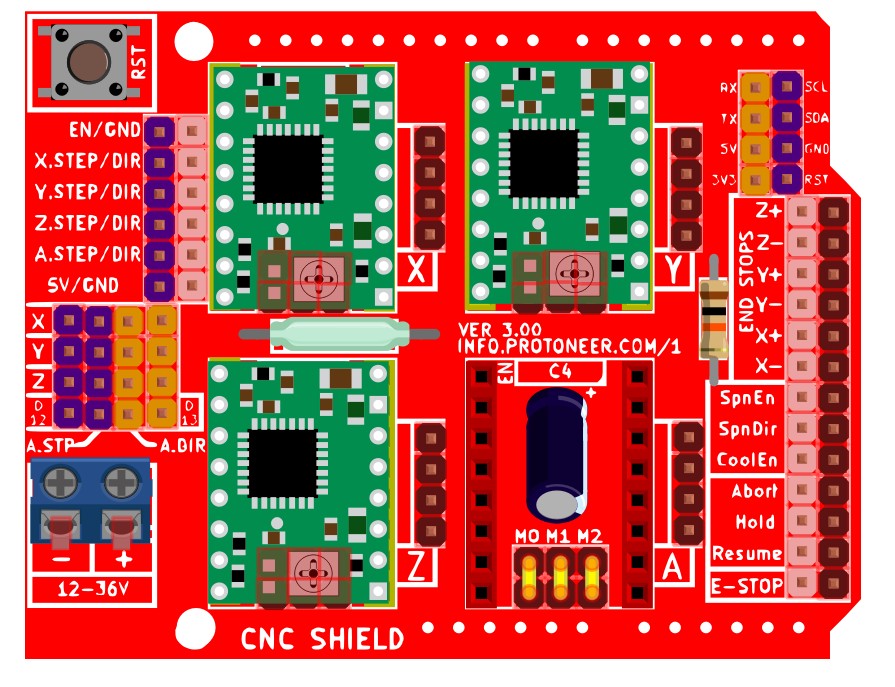
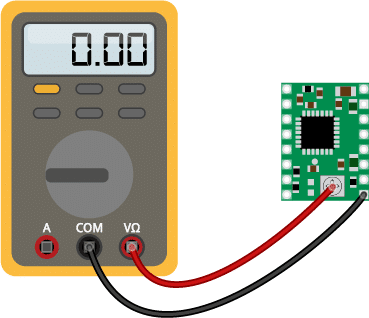
a



|  |
| --- |
| **CONFIGURATIONS** |
|  |
| ***STEP 6***  When assembling the CNCShield board on the Arduino board, pay attention to the direction and the pins must match as shown in figure A.  The drivers (B) must also be installed on the board, which will also have a mounting direction, as shown in the figure. |

B

10

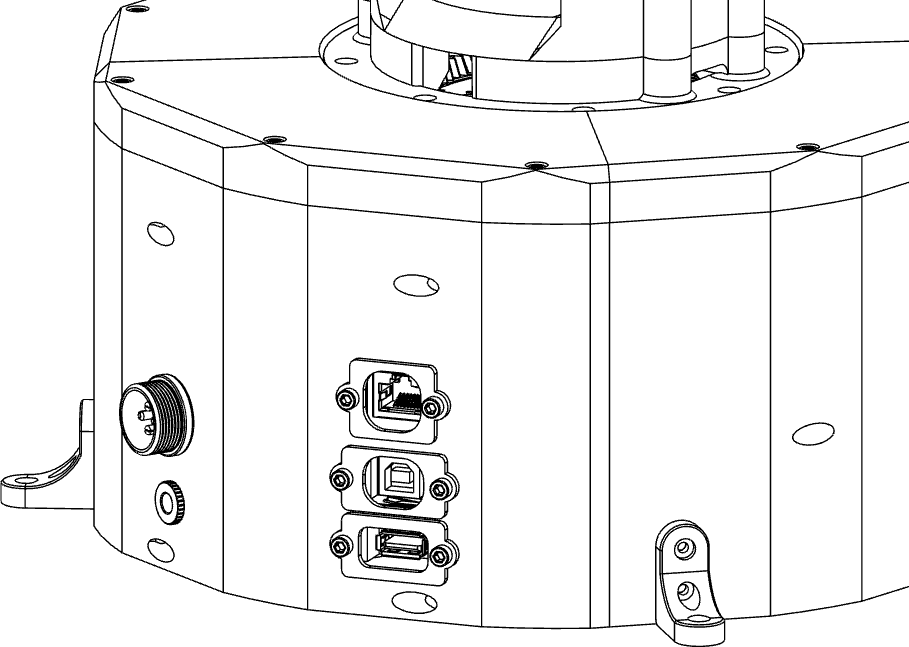


a

B

|  |
| --- |
| **CONFIGURATIONS** |
|  |
| ***STEP 7***  The A4988 allows you to set a target current anywhere between some mA up to a bit less than 2A, this is accomplished by adjusting what is called the Vref (Reference Voltage) when turning the pot on a clockwise direction the Vref voltage will increase and decrease when rotating it counterclockwise.  It is important to set this value to (indicated in volts, unit used to measure) 1,8V , thus turning the trimmer that allows the increase of the value on each driver.  To measure the adjusted value, just use a multimeter set on volts, and place it on the powered driver (mounted on the CNCShield board) and place the terminals of the multimeter as shown in the figure.  To conclude the configuration, the maximum microsteps that can be set must be set by jumpering the pins on the CNCShield board, as shown in figure B.  The jumpers must be placed on each position of the card drivers, therefore 3 positions for the three drivers. |

11



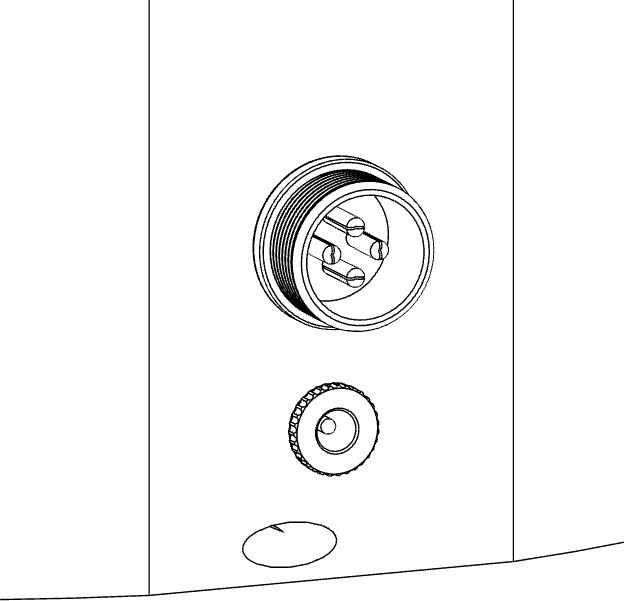
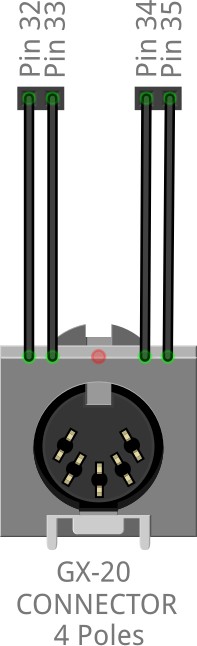
a

A

b C

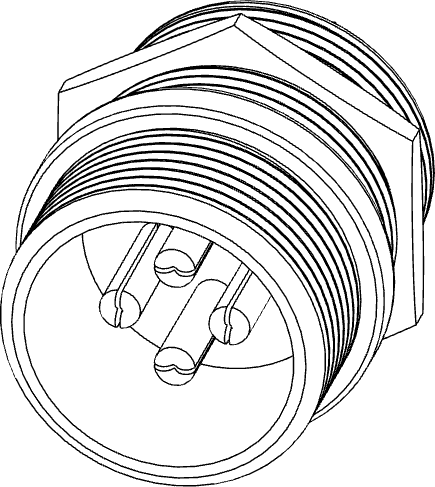
|  |
| --- |
| **COMMUNICATION** |
|  |
| ***STEP 8***  On the base, as indicated in the installation manual, there are 3 extensions, two USB type and one RJ45. The cables in question must be connected to the Arduino board and to the Raspberry.  The Ethernet cable (A) to the Raspberry, such as the  USB type A cable (C).  The USB type B cable (B) must be connected to the Arduino board..  ***Note:***  *The RJ45 connector / extension has no code, an item that can be mounted on the front of the panel with the following (fig. A) mounting dimensions (mm) must be selected.* |

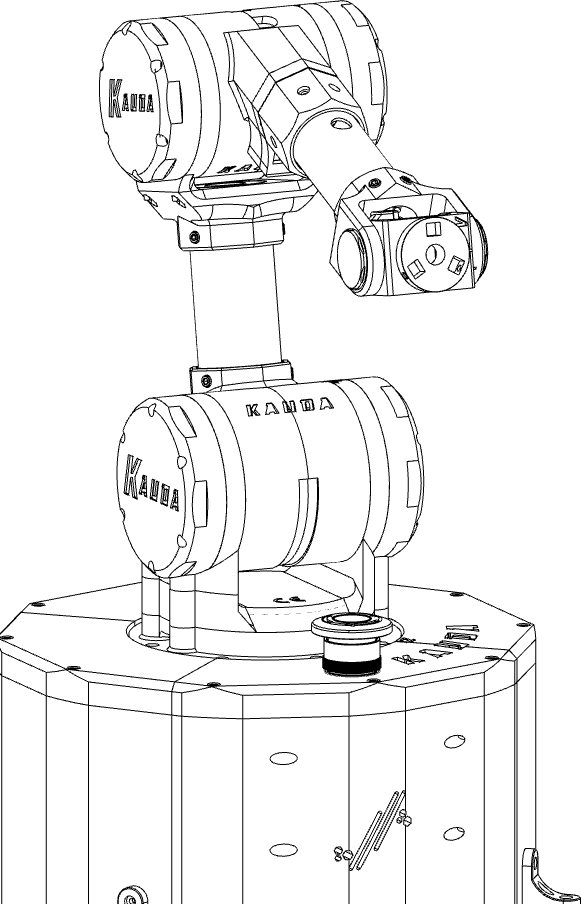
12



|  |
| --- |
| **COMMUNICATION** |
|  |
| ***STEP 9***  It is possible to connect the Arduino and / or Raspberry board to external devices through use of the GX20 connector.  In the following case, a 4-pole GX20 connector connected to the Arduino board is used. |

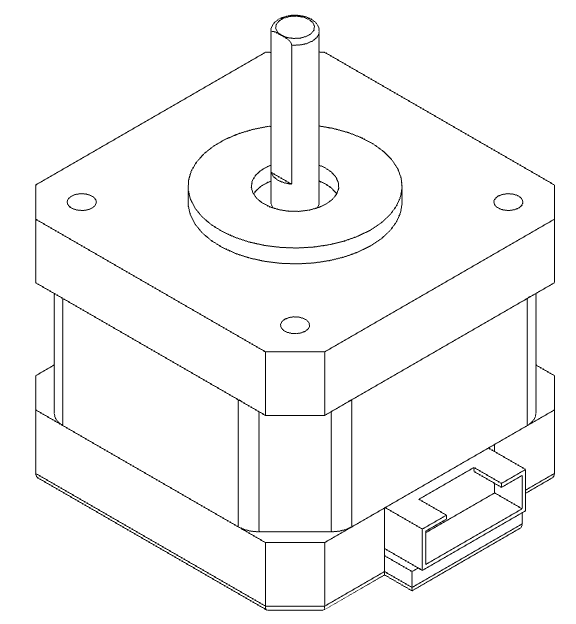
13





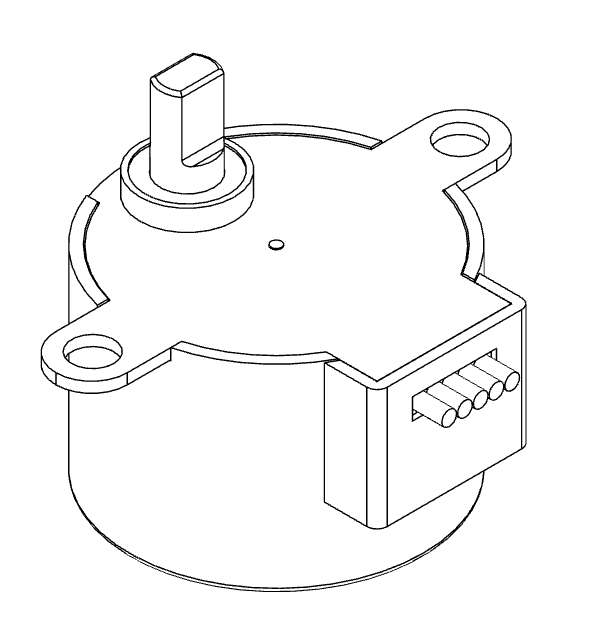
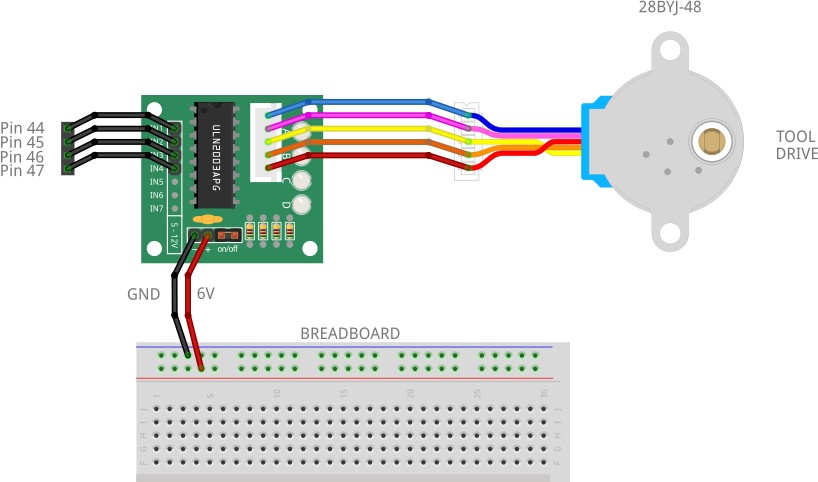
|  |
| --- |
| **OUT OF BASE** |
|  |
| ***NOTE***  Once you have reached this point, all the devices useful for the power supply have been connected, subsequently, for the connection of the remaining devices it will be necessary to take care to alternate the assembly of the mechanical parts with the wiring of the electrical parts.  It is therefore necessary to give priority to the mechanical installation by sliding the cables correctly to ensure that they all arrive correctly at the base. |

14



|  |
| --- |
| **STEPPER MOTOR** |
|  |
| ***STEP 10***  The three stepper motors must be connected, through their pre-wired cable, to the CNC Shield, on the three dedicated connectors.  ***Note:***  *As indicated on the previous page, care must be taken when passing the cables inside the mechanical parts of the robot.*  *The direction of assembly of the connector is of minor importance, it will be managed via SW.* |

15

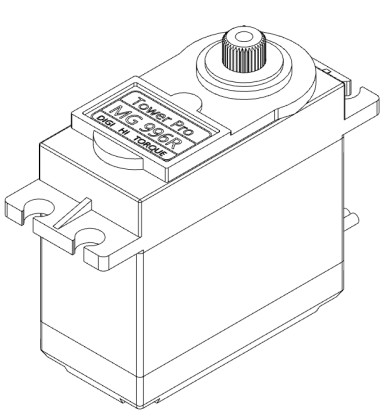
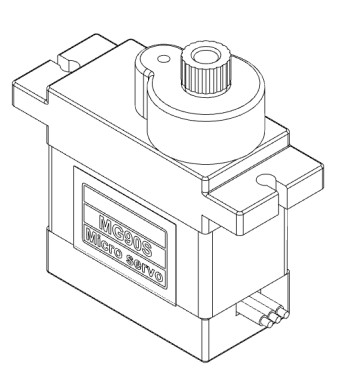
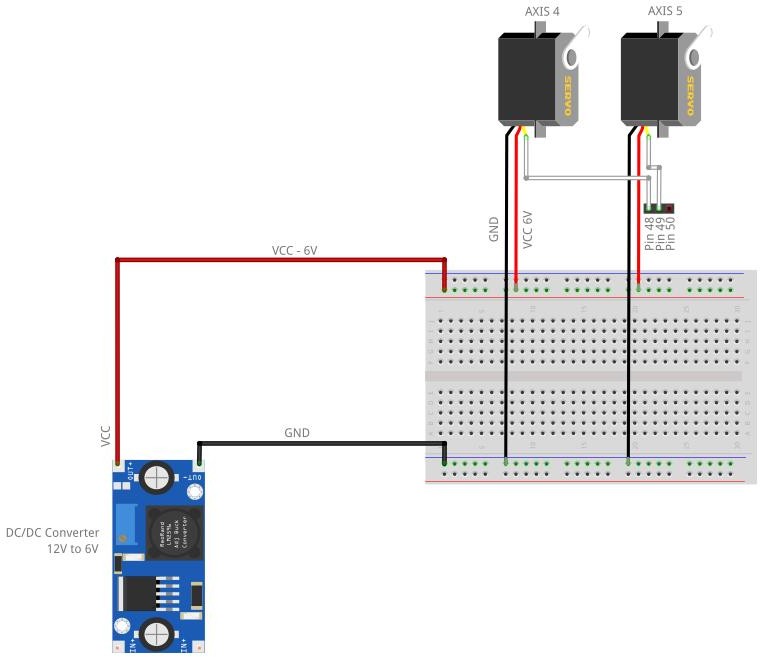


|  |
| --- |
| **STEPPER MOTOR** |
|  |
| ***STEP 11***  The stepper motor (28BYJ-48) dedicated to the tool operation (e.g. gripper) must be connected to the relative ULN2003 driver, as shown in the figure.  The driver is then connected to the Arduino board on pins 44, 45, 46,  47.  ***Note:***  *Motor power is supplied by the Breadboard as shown in the figure.* |

16

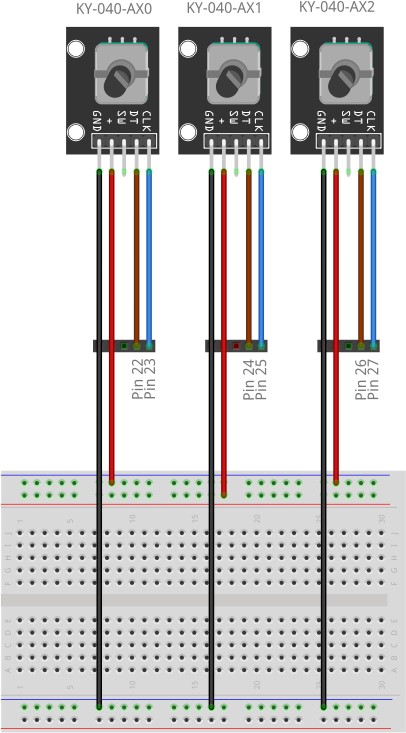
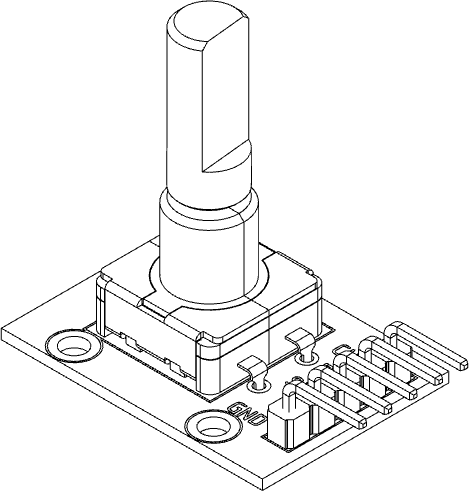


17



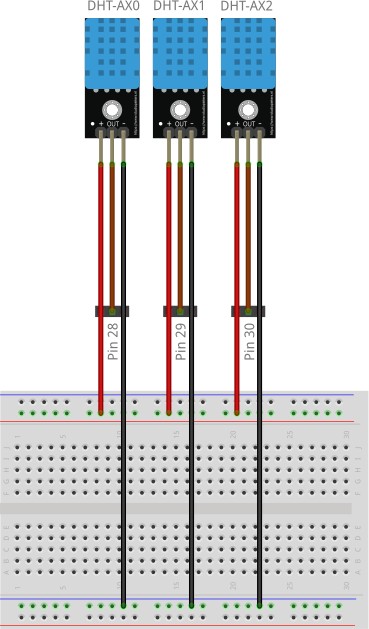
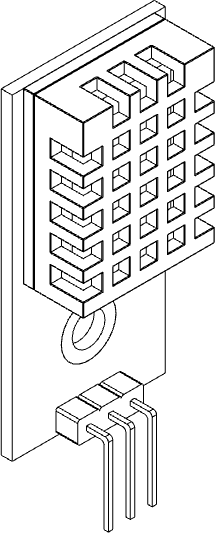
|  |
| --- |
| **SERVO MOTOR** |
|  |
| ***STEP 12***  The two servomotors must be connected to the previously preset power supply, that is to the bradboard placed at the base of the robot, while the signal cables must be connected on the Arduino board, to PIN 48 and 49.  ***Note:***  *As indicated on the previous page, care must be taken when passing the cables inside the mechanical parts of the robot.* |



  18

|  |
| --- |
| **ENCODER** |
|  |
| ***STEP 13***  The three encoders of the three axes driven by the stepper motors must be connected, as for the servomotors, to the breadboard from which they will take sufficient power for their operation and the signals (CLK and DT) will be connected to the PINs of the Arduino board as indicated in the figure .  ***Note:***  *As indicated on the previous page, care must be taken when passing the cables inside the mechanical parts of the robot.* |



|  |
| --- |
| **TEMPERATURE**  **SENSOR** |
|  |
| ***STEP 14***  The three temperature sensors of the three axes driven by the stepper motors must be connected, as for the encoders, to the breadboard from which they will take sufficient power for their operation, while the (OUT) temperature signals must be connected to the PIN of the Arduino board. as shown in the figure.  ***Note:***  *As indicated on the previous page, care must be taken when passing the cables inside the mechanical parts of the robot.* |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ARDUINO PIN MAPPING TABLE** | | | | |
|  | | | | |
|  | | | | |
| A picture containing text, electronics, circuit  Description automatically generated | **PIN NUMBER** | **PIN NAME** | **MAPPED PIN NAME** | **CONNECION** |
| **1** | PG5 ( OC0B ) | Digital pin 4 (PWM) |  |
| **2** | PE0 ( RXD0/PCINT8 ) | Digital pin 0 (RX0) |  |
| **3** | PE1 ( TXD0 ) | Digital pin 1 (TX0) |  |
| **4** | PE2 ( XCK0/AIN0 ) |  |  |
| **5** | PE3 ( OC3A/AIN1 ) | Digital pin 5 (PWM) |  |
| **6** | PE4 ( OC3B/INT4 ) | Digital pin 2 (PWM) |  |
| **7** | PE5 ( OC3C/INT5 ) | Digital pin 3 (PWM) |  |
| **8** | PE6 ( T3/INT6 ) |  |  |
| **9** | PE7 ( CLKO/ICP3/INT7 ) |  |  |
| **10** | VCC | VCC |  |
| **11** | GND | GND |  |
| **12** | PH0 ( RXD2 ) | Digital pin 17 (RX2) |  |
| **13** | PH1 ( TXD2 ) | Digital pin 16 (TX2) |  |
| **14** | PH2 ( XCK2 ) |  |  |
| **15** | PH3 ( OC4A ) | Digital pin 6 (PWM) |  |
| **16** | PH4 ( OC4B ) | Digital pin 7 (PWM) |  |
| **17** | PH5 ( OC4C ) | Digital pin 8 (PWM) |  |
| **18** | PH6 ( OC2B ) | Digital pin 9 (PWM) |  |
| **19** | PB0 ( SS/PCINT0 ) | Digital pin 53 (SS) |  |
| **20** | PB1 ( SCK/PCINT1 ) | Digital pin 52 (SCK) |  |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ARDUINO PIN TABLE** | | | | |
|  | | | | |
|  | | | | |
| A picture containing text, electronics, circuit  Description automatically generated | **PIN NUMBER** | **PIN NAME** | **MAPPED PIN NAME** | **CONNECION** |
| **21** | PB2 ( MOSI/PCINT2 ) | Digital pin 51 (MOSI) |  |
| **22** | PB3 ( MISO/PCINT3 ) | Digital pin 50 (MISO) |  |
| **23** | PB4 ( OC2A/PCINT4 ) | Digital pin 10 (PWM) |  |
| **24** | PB5 ( OC1A/PCINT5 ) | Digital pin 11 (PWM) |  |
| **25** | PB6 ( OC1B/PCINT6 ) | Digital pin 12 (PWM) |  |
| **26** | PB7 ( OC0A/OC1C/PCINT7 ) | Digital pin 13 (PWM) |  |
| **27** | PH7 ( T4 ) |  |  |
| **28** | PG3 ( TOSC2 ) |  |  |
| **29** | PG4 ( TOSC1 ) |  |  |
| **30** | RESET | RESET |  |
| **31** | VCC | VCC |  |
| **32** | GND | GND |  |
| **33** | XTAL2 | XTAL2 |  |
| **34** | XTAL1 | XTAL1 |  |
| **35** | PL0 ( ICP4 ) | Digital pin 49 | Servo Motor - AXIS 5 |
| **36** | PL1 ( ICP5 ) | Digital pin 48 | Servo Motor - AXIS 4 |
| **37** | PL2 ( T5 ) | Digital pin 47 | ULN2003 – IN4 (Driver stepper motor Tool) |
| **38** | PL3 ( OC5A ) | Digital pin 46 (PWM) | ULN2003 – IN3 (Driver stepper motor Tool) |
| **39** | PL4 ( OC5B ) | Digital pin 45 (PWM) | ULN2003 – IN2 (Driver stepper motor Tool) |
| **40** | PL5 ( OC5C ) | Digital pin 44 (PWM) | ULN2003 – IN1 (Driver stepper motor Tool) |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ARDUINO PIN TABLE** | | | | |
|  | | | | |
|  | | | | |
| A picture containing text, electronics, circuit  Description automatically generated | **PIN NUMBER** | **PIN NAME** | **MAPPED PIN NAME** | **CONNECION** |
| **41** | PL6 | Digital pin 43 |  |
| **42** | PL7 | Digital pin 42 |  |
| **43** | PD0 ( SCL/INT0 ) | Digital pin 21 (SCL) |  |
| **44** | PD1 ( SDA/INT1 ) | Digital pin 20 (SDA) |  |
| **45** | PD2 ( RXDI/INT2 ) | Digital pin 19 (RX1) |  |
| **46** | PD3 ( TXD1/INT3 ) | Digital pin 18 (TX1) |  |
| **47** | PD4 ( ICP1 ) |  |  |
| **48** | PD5 ( XCK1 ) |  |  |
| **49** | PD6 ( T1 ) |  |  |
| **50** | PD7 ( T0 ) | Digital pin 38 |  |
| **51** | PG0 ( WR ) | Digital pin 41 |  |
| **52** | PG1 ( RD ) | Digital pin 40 |  |
| **53** | PC0 ( A8 ) | Digital pin 37 |  |
| **54** | PC1 ( A9 ) | Digital pin 36 |  |
| **55** | PC2 ( A10 ) | Digital pin 35 | GX20 Connector – Pin 4 |
| **56** | PC3 ( A11 ) | Digital pin 34 | GX20 Connector – Pin 3 |
| **57** | PC4 ( A12 ) | Digital pin 33 | GX20 Connector – Pin 2 |
| **58** | PC5 ( A13 ) | Digital pin 32 | GX20 Connector – Pin 1 |
| **59** | PC6 ( A14 ) | Digital pin 31 |  |
| **60** | PC7 ( A15 ) | Digital pin 30 | DHT11 – AX2 |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ARDUINO PIN TABLE** | | | | |
|  | | | | |
|  | | | | |
| A picture containing text, electronics, circuit  Description automatically generated | **PIN NUMBER** | **PIN NAME** | **MAPPED PIN NAME** | **CONNECION** |
| **61** | VCC | VCC |  |
| **62** | GND | GND |  |
| **63** | PJ0 ( RXD3/PCINT9 ) | Digital pin 15 (RX3) |  |
| **64** | PJ1 ( TXD3/PCINT10 ) | Digital pin 14 (TX3) |  |
| **65** | PJ2 ( XCK3/PCINT11 ) |  |  |
| **66** | PJ3 ( PCINT12 ) |  |  |
| **67** | PJ4 ( PCINT13 ) |  |  |
| **68** | PJ5 ( PCINT14 ) |  |  |
| **69** | PJ6 ( PCINT 15 ) |  |  |
| **70** | PG2 ( ALE ) | Digital pin 39 |  |
| **71** | PA7 ( AD7 ) | Digital pin 29 | DHT11 – AX1 |
| **72** | PA6 ( AD6 ) | Digital pin 28 | DHT11 – AX0 |
| **73** | PA5 ( AD5 ) | Digital pin 27 | KY-040 – AX2 - CLK |
| **74** | PA4 ( AD4 ) | Digital pin 26 | KY-040 – AX2 – DT |
| **75** | PA3 ( AD3 ) | Digital pin 25 | KY-040 – AX1 – CLK |
| **76** | PA2 ( AD2 ) | Digital pin 24 | KY-040 – AX1 – DT |
| **77** | PA1 ( AD1 ) | Digital pin 23 | KY-040 – AX0 - CLK |
| **78** | PA0 ( AD0 ) | Digital pin 22 | KY-040 – AX0 - DT |
| **79** | PJ7 |  |  |
| **80** | VCC | VCC |  |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ARDUINO PIN TABLE** | | | | |
|  | | | | |
|  | | | | |
| A picture containing text, electronics, circuit  Description automatically generated | **PIN NUMBER** | **PIN NAME** | **MAPPED PIN NAME** | **CONNECION** |
| **81** | GND | GND |  |
| **82** | PK7 ( ADC15/PCINT23 ) | Analog pin 15 |  |
| **83** | PK6 ( ADC14/PCINT22 ) | Analog pin 14 |  |
| **84** | PK5 ( ADC13/PCINT21 ) | Analog pin 13 |  |
| **85** | PK4 ( ADC12/PCINT20 ) | Analog pin 12 |  |
| **86** | PK3 ( ADC11/PCINT19 ) | Analog pin 11 |  |
| **87** | PK2 ( ADC10/PCINT18 ) | Analog pin 10 |  |
| **88** | PK1 ( ADC9/PCINT17 ) | Analog pin 9 |  |
| **89** | PK0 ( ADC8/PCINT16 ) | Analog pin 8 |  |
| **90** | PF7 ( ADC7/TDI ) | Analog pin 7 |  |
| **91** | PF6 ( ADC6/TDO ) | Analog pin 6 |  |
| **92** | PF5 ( ADC5/TMS ) | Analog pin 5 |  |
| **93** | PF4 ( ADC4/TCK ) | Analog pin 4 |  |
| **94** | PF3 ( ADC3 ) | Analog pin 3 |  |
| **95** | PF2 ( ADC2 ) | Analog pin 2 |  |
| **96** | PF1 ( ADC1 ) | Analog pin 1 |  |
| **97** | PF0 ( ADC0 ) | Analog pin 0 |  |
| **98** | AREF | Analog Reference |  |
| **99** | GND | GND |  |
| **100** | AVCC | VCC |  |



|  |
| --- |
| **FINAL NOTES** |
|  |
| REV.02  DIY-TECH  by Giovanni Lerda [https://www.diy-technology.com](https://www.diy-technology.com/) |

Table of Contents

1. Title Page
2. Acknowledgments
3. Table of Contents
4. Introduction
   1. Overview of the Project
   2. Purpose of the Manual
   3. Brief Description of the Robotic Arm
5. Hardware Components
   1. 3D Printed Parts Assembly (based on Giovanni Lerda's instructions)
      1. List of 3D Printed Parts
      2. Assembly Instructions
   2. Electrical Components
      1. List of Electrical Components
      2. Schematics and Wiring Diagrams
      3. Assembly Instructions
6. Software Documentation
   1. Installation Instructions
   2. Code Structure and Explanation
   3. Uploading and Running the Software
7. Operation Instructions
   1. Step-by-Step Guide to Operating the Robotic Arm
   2. Safety Precautions
   3. Maintenance Tips
8. Troubleshooting
   1. Common Issues and Solutions
   2. Frequently Asked Questions (FAQs)
9. Appendices
   1. Additional Resources
   2. Links to Further Reading

**ARM ROBOT Project**

Software Requirements Specification

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <04/13/07> | <1.0> | SRS 1.0 | Group-1 |

# Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete **ARM robot system** by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements of the **ARM robot system** are provided in this document.

## Purpose

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to consumers. Also, we shall predict and sort out how we hope this product will be used in order to gain a better understanding of the project, outline concepts that may be developed later, and document ideas that are being considered, but may be discarded as the product develops.

In short, the purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality. Nonetheless, it helps any designer and developer to assist in software delivery lifecycle (SDLC) processes.

## Scope

Primarily, the scope pertains to the features for making Arm robot project UI on WinForms and on the Arduino.

## Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| Configuration | It means a product which is available / Selected from a catalogue can be customized. |
| DOF | Degrees of Freedom |
| UI | User Interface (the app the user uses) |
| critical axis | Axis H, Axis Y, Axis Z (Large steppers that could overheat) |

## Overview

The code has two important parts. Code on the user side and code on the hardware side. These structures ar connected through serail communication.

Overview

|  |  |  |
| --- | --- | --- |
| # Coding Standard | Hardware code | Arduino code |
| 1. **Classes/Structs:** | * PascalCase: class MotorController, class SensorData | |
| 1. **Functions/Methods:** | * camelCase (for Arduino/C++ parts): void readSensor() | * PascalCase (for Windows Forms, following .NET convention): void ProcessData() |
| 1. **Variables:** | * camelCase: int motorSpeed, bool isEnabled | |
| 1. **Private Class Members:** | * Prefix with m\_: m\_currentValue, m\_sensorReading | |
| 1. **Constants:** | * UPPERCASE: const int MAX\_SPEED = 255 | |
| 1. **File Names:** | * PascalCase: MotorController.h, SensorReader.cpp | |

## Functionality

#### The system shall all robotic movement based on user command

#### The system shall display the status of arm robot

#### The system shall have a user manual

#### The system shall all robotic movement based on user command

The system shall allow users to control each axis of the robotic arm

The system shall allow synchronized movement of all the actuators

#### The system shall display the status of arm robot

The system shall display to the user a connection status to the ARM robot

The system shall display temperature and humidity of critical axis

The system shall display errors either software or hardware related

The system shall notify the user if a movement is completed/unfinished/error

## Usability

### **Graphical User Interface**

The system shall have a single window interface.

The system shall provide use of icons and toolbars.

## Performance

The product shall be based on WinForms and must run of a windows application

The performance shall depend upon hardware components of the client/customer.

## Safety

### **Data Transfer**

The system shall coordinate with the arm robot regarding the interrupt buttons such as the emergency stop.

.

## Supportability

### **Configuration Management Tool**

The source code for both hardware and software is to be updated yearly

## 

## Design Constraints

### **Standard Development Tools**

The system shall be built using a standard WinForms tools and Arduino Ide capabilities.

### **Windows Based Product**

        Memory requirements at least 40mb

        The computer must run on x64 based

        The product must have serial port for communication to ARM robot

        Response time for loading the product should take no longer than five minutes.

        knowledge of basic computer skills is required to use the product

## Interfaces

USB 2.0 Cable Type A/B

### **User Interfaces**

The user interface for the software shall be compatible to any browser such as Internet Explorer, Mozilla or Netscape Navigator by which user can access to the system.

The user interface shall be implemented using any tool or software package like Java Applet, MS Front Page, EJB etc.

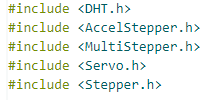
### **Hardware Interfaces**

USB 2.0 Cable Type A/B

POWER SUPPLY

**\*\*\*\*\*BEFORE PROCEEDING IT IS NOT ADVISABLE TO MODIFY THE SOURCE CODE FOR TROUBLESHOOTING\*\*\*\*\*\*\*\*\***

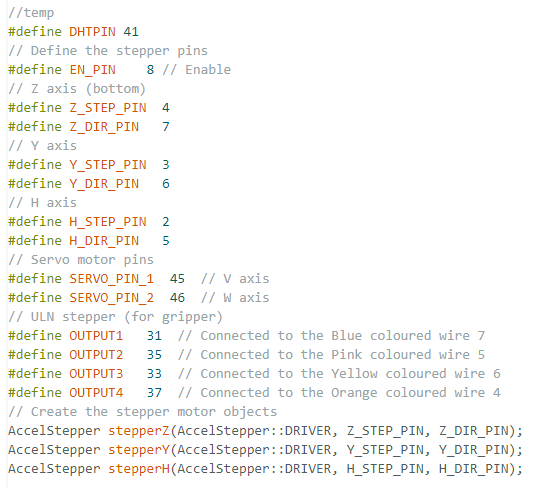
1. Hardware code:
   1. These are the libraires that the system shall use



Documentation for each will not be provided in this document.

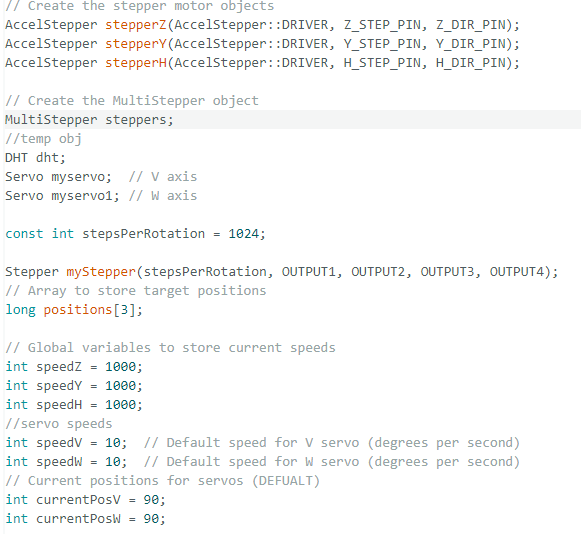
If you wish to read about it, all is available in the standard libraries’ documentation in the Arduino website

* 1. Data Declarations



The pins directly relate to the configuration of pins on the hardware. Ex Z\_STEP\_PIN 7 would mean the 7th pin on the Arduino mega is connected to the Z stepper pin.

More declarations:



Values seen here a default based on safety. But they can be changed on the windows application.

The use of arrays are minimized to try and keep all functions at O(1) time.

Time based commands aren’t used either for both hardware and software code as uncontrolled interrupts can destroy the serial connection.

C) setup

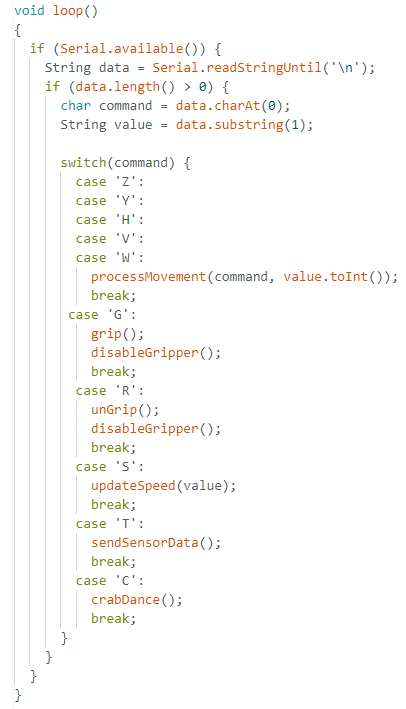


Initial setup (run once code)

9600 – is standard

15 – default speeds but can be adjusted on the WinForms.

D ) Main loop



Except for the gripper all actuators will fall to the process movement function

Gripper function is separated for fast response.

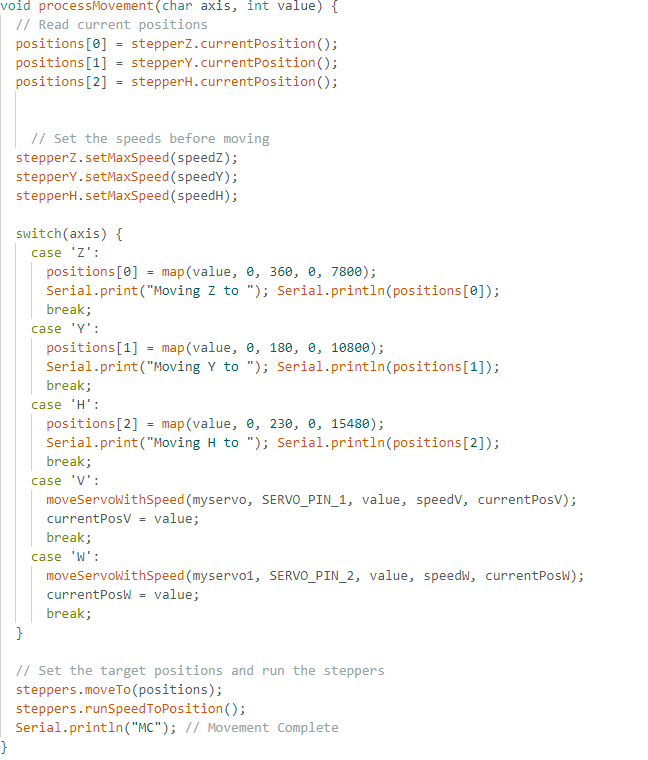
Crab dance is a hardcoded component checker. On the application once this is executing it works all the sensors and actuators to see it all is working well.

----------- PROCEDDING is a list of all functions------

E ) Update speed function



F ) Movement Function



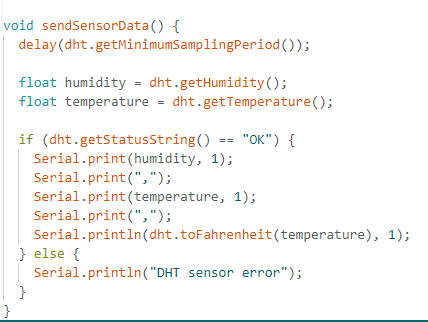
G ) Component check \*(to be revised once all the hardware is implemented)

As of this version 1.0 this document only checks the available hardware the robot has.



H ) Temp sensor

Standard for dht library



I ) Micro and servo control



The robot arm uses two types of motors: servos and stepper motors, each controlled by their respective libraries.

The servo motors handle smaller, non-critical movements in the arm. To prevent overheating, these servos are automatically disabled after each operation. While disabling reduces power consumption, it doesn't significantly affect holding torque. Even if these servos drift slightly from their position when disabled, it doesn't impact critical operations since they don't handle structural support.

The core structure of the arm relies on NEMA 17 stepper motors acting as actuators. These steppers maintain the arm's primary structural integrity and positioning. Unlike the servos, these stepper motors remain constantly enabled to maintain precise positioning and structural stability, even when not actively moving.

Thus, it is important to effectively relay a temperature status to the user on the application side/.

1. Software code:

Will be available of the next version of the document.

# Robot Arm System Manual

## Operating Manual and Troubleshooting Guide

### System Overview

The system consists of two main components:

1. Windows Forms Control Interface

2. Arduino-controlled Robot Arm

### Components Description

#### Z-Axis (Base Rotation)



- Function: 360-degree rotation of base plate

- Control: Stepper motor

- Limitations:

- Full rotation possible but not recommended for safety

- Monitor cable twist when rotating

#### Y-Axis (Lower Arm)



- Function: Primary arm movement (similar to cat arm)

- Control: Stepper motor

- Range: Extended movement possible beyond set limits

- Warning: Exercise caution due to:

- Potential collisions with obstacles

- Risk to personnel

- Possible mechanical stress

#### X-Axis (Upper Arm)



- Function: Secondary arm joint movement

- Control: Stepper motor

- Behavior: Similar to cat arm articulation

- Note: Capable of extended backward movement

#### V-Axis (Roll)



- Function: Rotational movement

- Control: Servo motor

- Movement: Self-rotation (roll motion)

#### Axis-2 (Wrist)



- Function: Wrist-like articulation

- Control: Servo motor

- Position: Located before gripper

- Movement: Similar to human wrist motion

#### G-Axis (Gripper)

- Function: Open/close gripper mechanism

- Control: Servo motor

- Operations: Binary movement (open/close)

### Operating Instructions

#### Starting the System

1. Power up the Arduino controller

2. Launch Windows Forms application

3. Verify connection status

4. Perform initial homing sequence if required

#### Control Panel Navigation

Each actuator has a dedicated control panel containing:

- Position control

- Speed adjustment

- Direction selection

- Execution button

- Speed setting control

#### Movement Control Procedure

1. Select desired axis panel

2. Set target position (degrees)

3. Adjust movement speed

4. Select direction

5. Click execute button

6. Monitor movement

7. Wait for completion

### Safety Guidelines

1. Always maintain clear workspace

2. Monitor cable management during rotation

3. Avoid unnecessary extreme positions

4. Keep emergency stop accessible

5. Observe movement limits

|  |  |  |
| --- | --- | --- |
| Common Issues and solution | Issue | Solution |
| Communication Problems | No Arduino connection | Restart application  Check port number |
| Delayed response | Verify speed settings  Monitor system |
| Movement Issues | Jerky movement | Check speed settings  Verify power supply  Inspect mechanical connections |
| No movement | Verify actuator power  Check control signals  Inspect wiring |
| Unusual noise | Check for obstacles  Verify lubrication  Inspect for wear |
| Position Errors | Inaccurate positioning | Calibrate system  Check encoder feedback  Verify step settings |
| Position drift | Check holding torque  Verify power settings  Inspect mechanical backlash |

### Maintenance Checklist

|  |  |  |
| --- | --- | --- |
|  | Component | CHECKs and Best Practices |
| Power supply |  | If you’re using a power supply. Make sure to always set output voltage to 5-6 volts. |
|  |  | Try to plugging in different ports or different devices to see if the port is damaged or if it’s the cable. |
|  |  | This servo is the weakest part of the arm  NEVER FORCE TO MOVE IT MANUALY.  Movement on this side shall only be done by the application  Its best practice that all movements be handled by the application. |
|  |  | Most trouble shooting should be limited to these parts.  The arm is not built for 360 degrees movement.  Most troubleshooting involves untangling the wires and making sure the pins are in place. Refer to the hardware installation for the pin connections. |
|  |  | This is a bad position of the arm robot.  best position at shut down is gripper on the floor.  During operation keep fragile objects aways from the arm.  Keep the arm clean as any build up on the surface may get stuck inside which is harder to clean and at worst heat components faster. |
|  |  | Before any operation always test the functionality of the stop button |

#### Daily Checks

- [ ] Power supply voltage

- [ ] Cable condition

- [ ] Movement smoothness

- [ ] Emergency stop function

- [ ] Control interface response

#### Weekly Checks

- [ ] Mechanical connections

- [ ] Lubrication points

- [ ] Calibration accuracy

- [ ] Backlash inspection

- [ ] Software updates

### Emergency Procedures

1. Immediate Stop

- Press emergency stop button

- Close application

- Power down system

2. Recovery Procedure

- Assess cause of emergency

- Clear any obstacles

- Reset emergency stop

- Perform homing sequence

- Resume operations

### Technical Support

For technical assistance:

1. Document the issue

2. Take photos/videos if applicable

3. Note any error messages

4. Record recent changes

5. Contact makerspace

### System Specifications

- Communication: USB/Serial

- Control Software: Windows Forms

- Firmware: Arduino

- Maximum Payload: worst case 0.5 grams (this is when kinematics isn’t set properly)

### Best Practices

1. Movement Planning

- Plan movements in advance

- Consider obstacle locations

- Verify clearance

- Use appropriate speeds

2. Operation

- Start with slower speeds

- Increase gradually

- Monitor all movements

- Keep workspace clear

3. Maintenance

- Regular calibration

- Keep logs

- Document changes

- Backup settings