UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

FINAL DEGREE THESIS

Bachelor's Degree in Industrial Electronics and Automatic Control

CHARACTERIZING THE SPECIFICATIONS OF A 6 DoF INDUSTRIAL ROBOTIC ARM

Technical Report

Author: Aleksandar Lazaroie
Supervisor: Sebastián Tornil Sin
Department ESAII
Company tutor: Ignacio Pedrosa Lojo
Semester: Spring 2023

Abstract

This project presents the study and characterization of the most important specifications of the robotic arm with six degrees of freedom. The paper presents a replicable test protocol valid for any type of industrial robot, but is focused on the CM robotics CM607-L robotic arm.

The main objective of the project is to create and present a test protocol and some possible equipment needed to carry out the tests so that in the future it can be used in the characterization of several robots, although some setups will be created and carried out to practice some of the tests to verify that the protocol is correct. Due to the short duration of the project, and the large number of tests necessary for the full characterization of the robot, this document will present the results of the speed tests of each axis, the total speed and the maximum operating range of each shaft

The document is divided into several sections. In the first section you can find the motivation and objectives of the project, the second section comments on the main characteristics of the robot used (CM607-L), the third section presents the different alternatives studied and the definitive test protocol, the fourth section shows the setups designed to carry out the practical tests and in the fifth section the results of the tests are collected. Finally, you can find the environmental study, the economic study and the conclusions.

Resum

En aquest projecte es presenta l'estudi i la caracterització de les especificacions més importants del braços robòtics de sis graus de llibertat. El document presenta un protocol replicable de proves vàlid per a qualsevol tipus de robot industrial, però està centrat en el braç robòtic CM607-L de CM robotics.

L'objectiu principal del projecte és crear i presentar un protocol de *tests* i alguns possibles equipaments necessaris per realitzar les proves per a què en un futur es pugui utilitzar en la caracterització de diversos robots, tot i això es crearan alguns *setups* i es realitzaran de forma pràctica algunes de les proves per comprovar que el protocol és correcte. Degut a la curta durada del projecte, i la gran quantitat de proves necessàries per a la total caracterització del robot, en aquest document es presentaran els resultats de les proves de velocitat de cada eix, la velocitat total i el màxim rang de funcionament de cada eix.

El document està dividit en diverses seccions. En el primer apartat es poden trobar la motivació i els objectius del projecte, el segon apartat comenta les principals característiques del robot utilitzat (CM607-L), el tercer apartat presenta les diferents alternatives estudiades i el protocol de test definitiu, el quart apartat mostra els setups dissenyats per a realitzar les proves pràctiques i en el cinquè apartat estan recollits els resultats de les proves. Finalment, es poden trobar l'estudi ambiental, l'econòmic i les conclusions.

Resumen

En este proyecto se presenta el estudio y caracterización de las especificaciones más importantes de los brazos robóticos de seis grados de libertad. El documento presenta un protocolo replicable de pruebas válido para cualquier tipo de robot industrial, pero está centrado en el brazo robótico CM607-L de CM robotics.

El objetivo principal del proyecto es crear y presentar un protocolo de *tests* y algunos posibles equipamientos necesarios para realizar las pruebas para que en un futuro se pueda utilizar en la caracterización de varios robots, sin embargo, se crearán algunos *setups* y se realizarán de forma práctica algunas de las pruebas para comprobar que el protocolo es correcto. Debido a la corta duración del proyecto, y la gran cantidad de pruebas necesarias para la total caracterización del robot, en este documento se presentarán los resultados de las pruebas de velocidad de cada eje, la velocidad total y el máximo rango de funcionamiento de cada eje.

El documento está dividido en varias secciones. En el primer apartado se puede encontrar la motivación y los objetivos del proyecto, el segundo apartado comenta las principales características del robot utilizado (CM607-L), el tercer apartado presenta las diferentes alternativas estudiadas y el protocolo de test definitivo, el cuarto apartado muestra los setups diseñados para realizar las pruebas prácticas y en el quinto apartado están recogidos los resultados de las pruebas. Por último, se pueden encontrar el estudio ambiental, el económico y las conclusiones.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my thesis adviser, Prof. Sebastian Tornil. It would have been much more difficult to develop this project without his supervision and advises. When I had a problem or had a question regarding my research or writing, the door was always open. Throughout this research, I have valued his expertise, regular guidance, insightful discussion, and valuable suggestion.

I am also very grateful to my family for the love and support provided throughout my entire life and during my undergraduate degree, during which they have provided all the necessaries so that I could be fully focused on completing it without having to worry about anything else. Also, I would like to express my thanks to my friends for providing moral support, and making my student life much more entertaining, and easier to carry out.

Last but not least, I want to thank Ignacio Pedrosa, my tutor at Ingenia. I am very grateful for all the support he gave me all these months I have been doing my internship. He has always tried to teach me to be a better professional.

List of abbreviations

Degrees of Freedom	DoF
Key Performance Indicator	KPI
International Organization for Standardization	ISO
Safe Torque Off	STO
Printed Circuit Board	PCB
Light Emitting Diode	LED
Linear Variable Differential Transformer	LVDT
Graphical User Interface	GUI
Digital Inputs and Outputs	DIO
Digital Inputs	DI
Digital Outputs	DO
Pulse Width Modulation	PWM
Universal Serial Bus	USB
Controller Area Network	CAN
High-Definition Serial Interface	HDSI
Microcontroller Unit	MCU
Tool Center Point	TCP

List of Figures

List of Equations

List of tables

List of Content

1.- Introduction

1.1.- Motivation

For some years, the robotic arms market has been growing at a very high speed. They are used in many sectors, such as, industrial where we can find them in pick and place tasks, assemblies, palletizing; medical, i.e., surgical robots and others. The global robotic arm market was valued at USD 26.24 billion in 2021 and is expected to reach USD 74.35 billion by 2029, registering a CAGR of 13.90% during the forecast period of 2022-2029. **[1]**

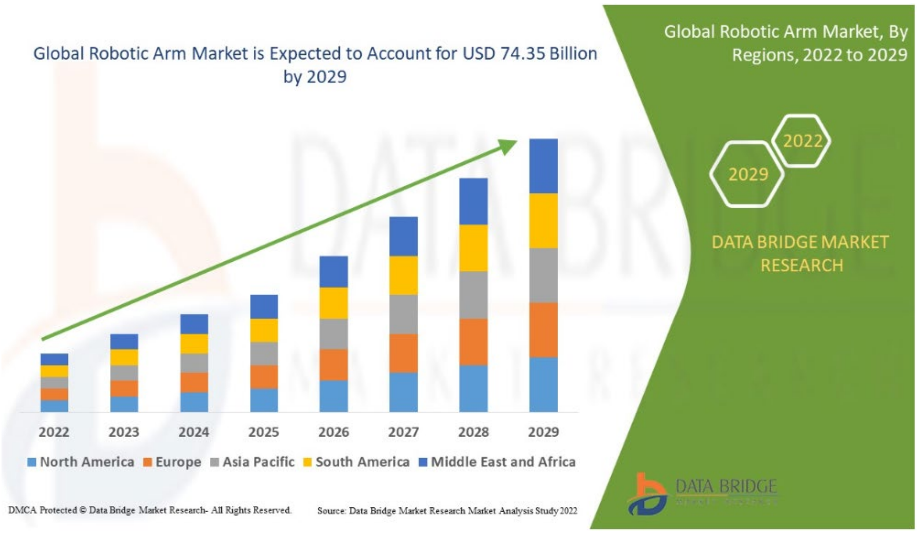


Figure 1.1: *Expected growth of the robotic arms market.* ***[1]***

It is expected a higher growth in the next years, but this could be bad for the quality of the robots, more and more companies try to get their own robot and idea into the market with more and more competitive prices and better performance, or at least, this is what they say.

Even though there are some norms that explain how the robots must be tested to ensure their quality, these norms are not evolving as fast as the robots are. For example, the ISO 9283, which explains how to perform some tests, was released in 1998. Moreover, these norms do not collect all the tests a robot should pass.

Hence, one of the motivations behind this project is to research how companies characterize the specifications of their robotic arms. Which are the tests they perform to ensure that the speed of the first joint is the one we can find in the datasheets?

On the other hand, this project will be performed during an internship at Ingenia Motion Control, where there is a 6 DoF industrial robotic arm that can be used to perform the defined tests and get some real results.

1.2.- Objectives

The main goal of this project is to create a replicable test protocol for industrial robotic arms of 6 DoF. To achieve this, it must be followed

- Search for information about industrial robotic arms
- Find the key performance indicators for this kind of robot
- Investigate how manufacturers test their robots, ISO standards?
- Create a test protocol and select the needed components
- Build the needed setups
- Test the characteristics
- Redesign the tests (if necessary)
- Compare the experimental values with the expected ones
- Determine robot's performance

Due to the short period of time available to develop this project, and the large number of tests needed to characterize a whole robotic arm, this project will be focused on the development of the protocol tests and only some of them will be actually tested.

The expected tests to be performed are the maximum speed of each joint test, the maximum speed of the whole robot test, and the maximum range of the robot.

The idea is to do the research of information as fast as possible because the development of the test protocol will take a lot of time and building the setups for the tests, plus the shipping time of the components and the actual performance of the tests will occupy the rest of the time available for the project.

2.- Presenting the robotic arm

The main objective of this section is to introduce the studied robotic arm because it is not from top manufacturers such as, Fanuc, Universal Robots, ABB, etc.

The CM607-L is a low-cost industrial robotic arm capable of performing a big number of jobs. Compared to its direct competitors, this robot has a very good cost-performance relation due to the range, the speed of the axis and the cost, the major weaknesses may be the maximum payload and the repeatability but even so, these parameters are quite acceptable for most of the applications. **[2]**

Figure 2.1: *CM607-L robotic arm.* ***[2]***

2.1.- Main features and specifications

The main features of this six DoF robotic arm are collected in the table below (*Table 2.1*).

Table 2.1: *Main features CM607-L*

Structure	Multijoint
Number of DoF	6

Placement method	Any angle
Maximum load capacity	6 Kg
Position repeatability	± 0.05 mm
Maximum range of motion	Joint 1: ± 170° Joint 2: +83°/-110° Joint 3: +180°/-67° Joint 4: ± 180° Joint 5: ± 120° Joint 6: ± 360°
Maximum speed of motion	Joint 1: 295°/s Joint 2: 245°/s Joint 3: 295°/s Joint 4: 365°/s Joint 5: 295°/s Joint 6: 370°/s

This robot is mostly used in repetitive or dangerous tasks. The typical applications are:

- Hardware Stamping
- Vehicle Parts Manufacturing and Installation
- Mobile Phones and Computer Manufacturing
- Machine Tools Manipulation

It is important to keep in mind the specifications of the robotic arm before using it.

Table 2.2: *Electrical and Power Specifications*

Minimum Power Supply	200 V AC, 50 Hz
Maximum Power Supply	230 V AC, 60 Hz
Recommended Power Supply	220 V AC, 50 Hz
Logic Power Supply	24 V DC
Maximum Power Consumption (robot)	1,8 kW
Maximum Power Consumption (cabinet)	2 kW

Table 2.3: *Communication Options*

Ethernet	For PC connection
Modbus	For other devices connection
RS-232	For sensor connections
RS-485	For sensor connections

Table 2.4: *Mechanical Specifications*

Robot Weight	35.92 Kg
Cabinet Weight	8 Kg
Robot size (zero position)	Height: 855 mm, width: 585 mm; Base size: 230 x 230 mm
Cabinet size	400 x 280 x 165 mm

Table 2.5: *Environmental Conditions.*

Robot IP	IP56

Cabinet IP	IP20
Cabinet Maximum Humidity	85% RH non-condensing
Robot Temperature	0 to 45 ºC
Cabinet Temperature	0 to 45 ºC

2.2.- Dimensions

The dimensions and the working range of the robot are presented in the next diagram.

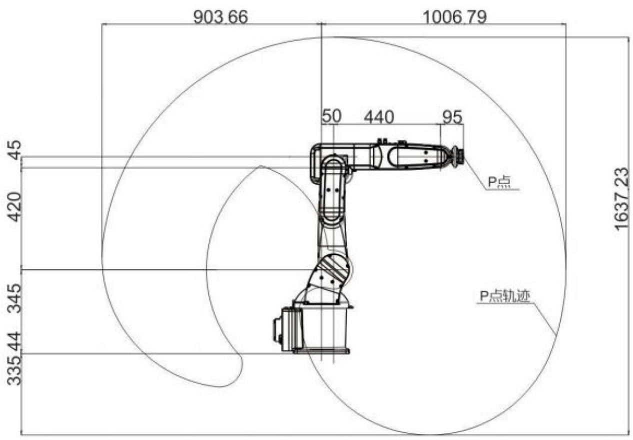


Figure 2.2: *Working range CM607-L.*

The robot has some marks to place it in its zero position, this is really important in order to achieve the best performance and the maximum range of each joint.

Joint 1:

Figure 2.3: *Zero position 1st joint*

Joint 2:

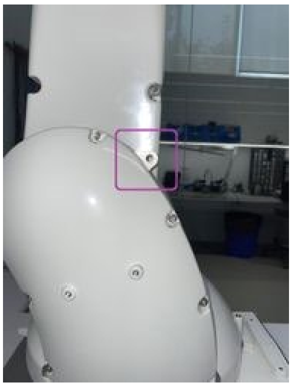


Figure 2.4: *Zero position 2nd joint.*

Joint 3:

Figure 2.5: *Zero position 3rd joint.*

Joint 4:

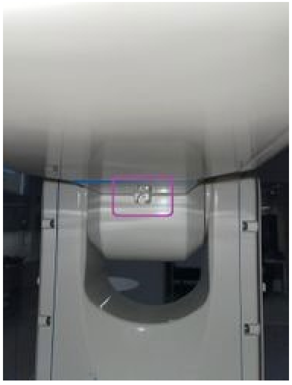


Figure 2.6: *Zero position 4th joint.*

Joint 5:

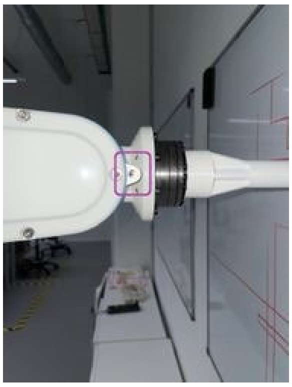


Figure 2.7: *Zero position 5th joint.*

Joint 6:

The sixth joint does not need to be placed in a specific zero position; it can be chosen any.

Once all the joints are in their respective zero position, the robot will look like the image below (*Figure 2.8*).

Figure 2.8: *Robot placed on the zero position.*

2.3.- Connectors and wiring

Figure 2.9: *General view of the cabinet.*

2.3.1.- Power supply

Figure 2.10: *Power supply connector.*

Table 2.6: *Power supply pinout.*

Pin	Signal	Function
1	L1 (black)	First phase
2	L2 (brown)	Second phase
3	L3 (white)	Third phase
4	PE (yellow-green)	Protective ground
Notes		
This connector has only one way to be connected, this is a security system, so the operator cannot do a wrong connection and damage the equipment. The cabinet must be connected at 220V and 50 Hz.		

2.3.2.- System digital inputs and outputs

Figure 2.11: *System digital inputs and outputs connector.*

Table 2.7: System DIO pinout.

Pin	Signal	Function
1	DIN33	Digital input1
2	DIN34	Digital input2
3	DIN35	Digital input3
4	DIN36	Digital input4
5	DIN37	Digital input5
6	DIN38	Digital input6
7	DIN39	Digital input7
8	DIN40	Digital input8
9	SDI_COM	Digital input common
10	SDI_COM	Digital input common
11	IO_24V	IO power output positive pole: 24V
12	IO_0V	IO power output negative pole: 0V
13	DOUT17+	Digital output 1
14	DOUT17-	
15	DOUT18+	Digital output 2

16	DOUT18-	
17	DOUT19+	Digital output 3
18	DOUT19-	
19	DOUT20+	Digital output 4
20	DOUT20-	
21	DOUT21+	Digital output 5
22	DOUT21-	
23	DOUT22+	Digital output 6
24	DOUT22-	
25	DOUT23+	Digital output 7
26	DOUT23-	
27	DOUT24+	Digital output 8
28	DOUT24-	
Notes		
This connector has 8 double-ended inputs and 8 double-ended outputs.		

2.3.3.- User digital inputs

Figure 2.12: *User digital inputs connector.*

Table 2.8: *User DI pinout.*

Pin	Signal	Function
1	DIN1	Digital input 1
2	DIN2	Digital input 2
3	DIN3	Digital input 3
4	DIN4	Digital input 4
5	DIN5	Digital input 5
6	DIN6	Digital input 6
7	DIN7	Digital input 7
8	DIN8	Digital input 8
9	DI_COM1	Digital input 1~8 common
10	DI_COM1	
11	IO_24V	IO power output positive pole: 24V
12	IO_0V	IO power output negative pole: 0V

13	DI_COM2	Digital input 9~16 common
14	DI_COM2	
15	DIN9	Digital input 9
16	DIN10	Digital input 10
17	DIN11	Digital input 11
18	DIN12	Digital input 12
19	DIN13	Digital input 13
20	DIN14	Digital input 14
21	DIN15	Digital input 15
22	DIN16	Digital input 16
23	DIN17	Digital input 17
24	DIN18	Digital input 18
25	DIN19	Digital input 19
26	DIN20	Digital input 20
27	DIN21	Digital input 21
28	DIN22	Digital input 22
29	DIN23	Digital input 23

30	DIN24	Digital input 24
31	DI_COM3	Digital input 17~24 common
32	DI_COM3	
33	IO_24V	IO power output positive pole: 24V
34	IO_0V	IO power output negative pole: 0V
35	DI_COM4	Digital input 25~32 common
36	DI_COM4	
37	DIN25	Digital input 25
38	DIN26	Digital input 26
39	DIN27	Digital input 27
40	DIN28	Digital input 28
41	DIN29	Digital input 29
42	DIN30	Digital input 30
43	DIN31	Digital input 31
44	DIN32	Digital input 32
Notes		
This connector has 32 channels available.		

2.3.4.- User digital outputs

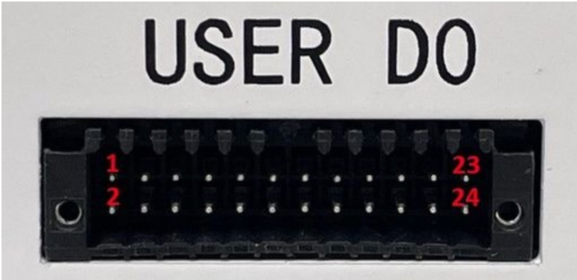


Figure 2.13: *User digital outputs connector.*

Table 2.9: *User DO pinout.*

Pin	Signal	Function
1	DOUT1+	Digital output 1
2	DOUT1-	
3	DOUT2+	Digital output 2
4	DOUT2-	
5	DOUT3+	Digital output 3
6	DOUT3-	
7	DOUT4+	Digital output 4
8	DOUT4-	
9	DOUT5+	Digital output 5
10	DOUT5-	
11	DOUT6+	Digital output 6

12	DOUT6-	
13	IO_24V	IO power output positive pole: 24V
14	IO_0V	IO power output negative pole: 0V
15	DOUT7+	Digital output 7(output 0v)
16	DOUT8+	Digital output 8
17	DOUT9+	Digital output 9
18	DOUT10+	Digital output 10
19	DOUT11+	Digital output 11
20	DOUT12+	Digital output 12
21	DOUT13+	Digital output 13
22	DOUT14+	Digital output 14
23	DOUT15+	Digital output 15
24	DOUT16+	Digital output 16
Notes		
This connector has 6 double-ended outputs and 10 single-ended outputs.		

2.3.5.- Analog inputs and outputs

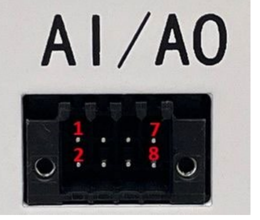


Figure 2.14: Analog inputs and outputs connector.

The manufacturer does not give any information about these inputs and outputs in the manual.

2.3.6.- Safe torque off

Figure 2.15: Safe torque off connector.

Table 2.10: STO pinout.

Pin	Signal	Function
1	IO_24V	IO power output positive pole
2	IO_24V	IO power output positive pole
3	STO_01	STO input 1
4	STO_02	STO input 2
5	STO_OUT	STO output
6	STO_0V	STO output common

Notes
STO_OUT cannot be used, and STO_0V has no utility.

2.3.7.- Communication connectors

LAN and USB:

Figure 2.16: *RJ45 and USB connectors.*

Table 2.11: *Communication connectors.*

Notes
LAN → Connection with Ethernet cable. USB → Connection with USB-A cable or pen drive.

RS232/RS485/CAN:

Figure 2.17: *RS232/RS485/CAN connector.*

Table 2.12: *RS232/RS485/CAN pinout.*

Pin	Signal	Function
1	RS485+	RS485 positive
2	RS232_RX	RS232 receiver

3	RS232_TX	RS232 sender
4	-	Null
5	GND	RS232 signal reference ground
6	RS485-	RS485 negative
7	CAN+	CAN transceiver+
8	CAN-	CAN transceiver-
9	PE	Shield ground
Notes		
This connector is common for the three protocols available.		

EXTENC:

Figure 2.18: *External encoder connector.*

Table 2.13: *External encoder pinout.*

Pin	Signal	Function
1	ENC_5V	Encoder power supply 5V
2	GND	Encoder signal ground

3	A+	Encoder A+
4	A-	Encoder A-
5	PE	Shield ground
6	B+	Encoder B+
7	B-	Encoder B-
8	Z+	Encoder Z+
9	Z-	Encoder Z-
Notes		
This connector is used to add auxiliary encoders to the robotic arm.		

PWM/HDSI:

Figure 2.19: *PWM/HDSI connector.*

Table 2.14: *PWM/HDSI pinout.*

Pin	Signal	Function
1	HSDI2+	High-speed input port 2+
2	HSDI1+	High-speed input port 1+

3	-	Null
4	PWM2_OUT	PWM output 2
5	PWM1_OUT	PWM output 1
6	HSDI2-	High-speed input port 2-
7	HSDI1-	High-speed input port 1-
8	-	Null
9	DGND	Signal ground
10	DGND	Signal ground
11	-	Null
12	PE	Shield ground
13	-	Null
14	-	Null
15	-	Null

2.3.8.- Robot connectors

General connector:

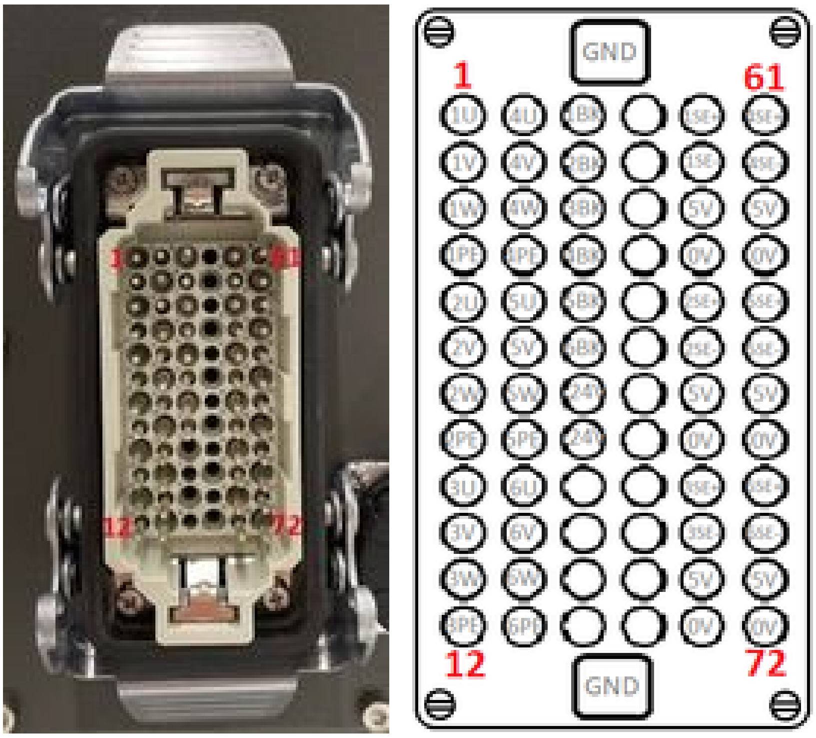


Figure 2.20: *General robot's connector.*

Table 2.15: *Notes about the general connector.*

Notes
This connector is situated on the back plane of the axis 1 of the robot is the one that goes to the cabinet. It has only one possible way to connect, in order to prevent any damage if connected otherwise. This connector is provided by the producer, so it should not be modified, the pinout is only informative.

16-pin aerial plug without built-in solenoid valve:

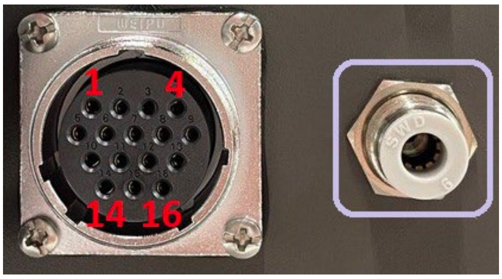


Figure 2.21: 16-pin aerial connector.

Table 2.16: 16-pin aerial connector pinout.

Pin	Function
1	Custom
2	Custom
3	Custom
4	Custom
5	Custom
6	Custom
7	Custom
8	Custom
9	Custom
10	Aviation plug shell
11	Reserve

12	Reserve
13	Reserve
14	Reserve
15	Reserve
16	Reserve
Notes	
This connector is situated on the back plane of the axis 1 of the robot, and it is usually connected to the connector on the axis 5. The lilac square marks the air intake, which can be used from 0 to 0.6 MPa.	

10 pin aerial plug:

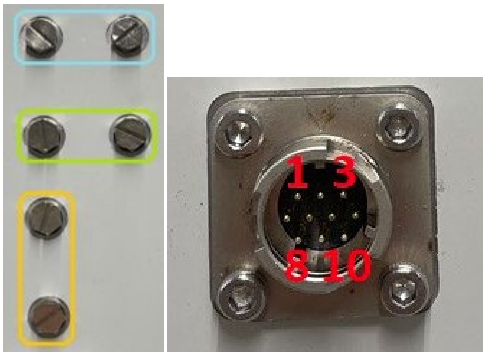


Figure 2.22: *Solenoid valves and 10 pin aerial connector.*

Table 2.17: *10-pin aerial connector pinout.*

Pin	Function
1	Custom
2	Custom

3	Custom
4	Custom
5	Custom
6	Custom
7	Custom
8	Custom
9	Null
10	Shell
Notes	
This connector is situated on the front plane of the axis 5 of the robot, and it is usually connected to the connector on the axis 1. The different squares are the places where the 3 possible solenoid valves would be. In this case, the robot has no solenoid valves.	

2.3.9.- Basic connection

It is very simple to connect the robotic arm, there are three important connections on the cabinet. The one on the left is connected to the controller, the big one in the middle of the cabinet is the connection with the robotic arm and finally, the one on the right side is the power supply.

Figure 2.23: *General connection of the cabinet.*

2.4.- Teaching pendant

In this section, it is explained the controller used to program the robot.

As the robotic arm manufacturer is a Chinese company focused on the Chinese robots market, most of the things, such as the controller buttons, software, indications, are written in Chinese.

Figure 2.24: *Robot's controller.*

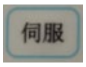
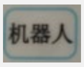
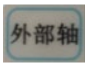
2.4.1.- Buttons on the right

Table 2.18: *Buttons on the right side of the controller.*

	Stop	Pause the program in run mode
	Start	Start program in run mode
	Minus	When teaching, the corresponding axis runs in the negative direction
	Plus	When teaching, the corresponding axis runs in the positive direction

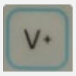
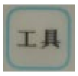
2.4.2.- Buttons on the left

Table 2.19: *Buttons on the left side of the controller.*

	Servo	Switch current servo state
	Robot	Switch the current robot. (Only available in multi-machine mode)
	External shaft	Switch between the current robot and external axis. (Only available when there is an external axis)
	Home	Home button
	Reduce	Recovery site button
	Clear mistakes	The error is cleared after the servo reports an error. (Only valid in teaching mode)）
		Reserved

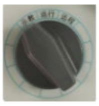
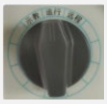
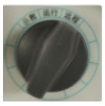
2.4.3.- Buttons on the bottom

Table 2.20: *Buttons on the bottom of the controller.*

	Forward/Backward	Switch between sequential execution or reverse execution when single-step running program in teaching mode.
	One step	Run the program step by step in the teaching mode.
	Less velocity	Reduce the teaching or running speed.
	More velocity	Increase the teaching or running speed.
	Tool	Switch tool hand (reserved).
	Coordinate	Switch whether to execute the program in a single step in the teaching mode in order or in reverse order.

2.4.4.- Switch on the top

Table 2.21: *Switch positions.*

	Teach	On the left, switch to teaching mode
	Run	In the middle, switch to run mode
	Remote	On the right, switch to remote mode

2.4.5.- Selection wheel on the right

Table 2.22: *Selection wheel on the right side.*

	Program interface selection to switch the previous line and the next line

2.5.- Signalling LEDs

2.5.1.- Controller's LEDs

Figure 2.25: *Teaching pendant's status LEDs.*

Table 2.23: *Controller's LEDs.*

LED	Name	Colour	Description
1	Power supply	Red	This LED indicates when the system is powered.
2	Operation Status	Green	This LED indicates when a program is running.
3	Servo State	Orange	This LED indicates when the servos are ON.
4	Error	Red	This LED indicates if there is an error or if the emergency button was pushed.
5	LED 4	Green	No information, is always off.
6	LED 5	Green	No information, is always on.
Notes			
When using only the power supply and LED5 should be ON, and when moving the robotic arm it should light the Servo state LED too.			

2.5.2.- Cabinet's LEDs

Figure 2.26: *Cabinet's status LEDs.*

Table 2.24: *Cabinet's LEDs.*

LED	Name	Colour	Description
1	PWR	Red	This LED indicates when the system is powered.
2	ALM	Red	This LED indicates if there is an error or if the emergency button was pushed.
3	SON	Green	This LED indicates the servo status.
4	RUN	Green	This LED indicates that the system is ON and running.
Notes			
When using only PWR and RUN should be ON, and when moving the robotic arm it should light the Servo state LED too.			

2.6.- Programming

There are multiple ways to program this robot, such as a Melfa BASIC V lookalike language, Python, ROS, C#. In this project, it will be used only the robot's programming language and Python.

2.6.1.- Motion Control Commands

The robots' software has so many different commands in order to perform all kinds of tasks. As can be seen in the image below (*Figure 2.27*), it can be coded palletizing tasks, visual based tasks, get inputs and give outputs to implement conveyor belts or to enter interruptions, and it has the option to code welding tasks. Even so, this project will focus on the motion control commands, some conditional commands and timers.

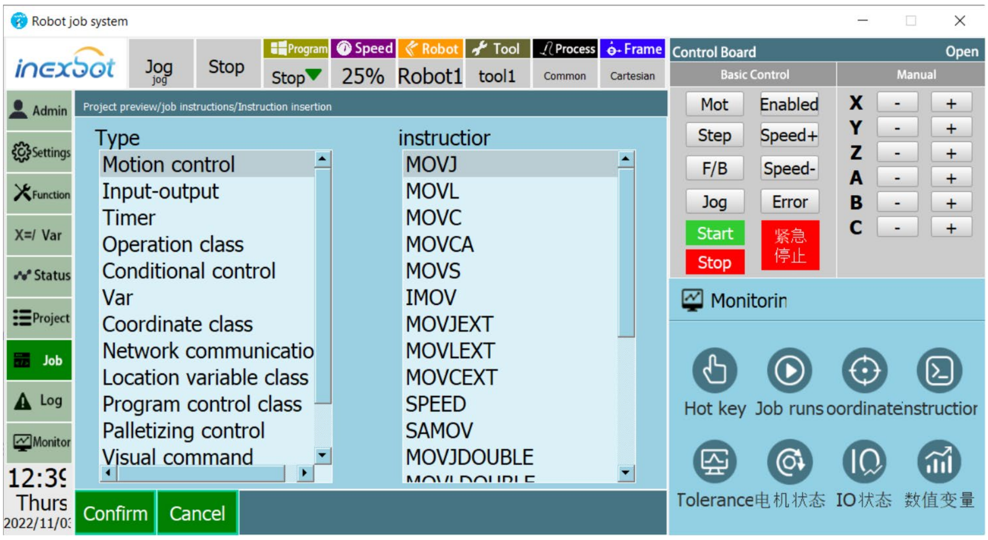


Figure 2.27: *Commands window.*

The most important motion control commands will be explained to have some background.

2.6.1.1.- MOVJ

Function: The main function of this instruction is to take the end effector of the robot from one position to another using joint interpolation to move to the target point. When this instruction is used, the robot tries to get to the position in the fastest way possible, it is not possible to predict the movement it will make.

Parameters description:

Figure 2.28: *MOVJ parameters.*

1. **VJ:** The speed of joint interpolation, the range is 1-100, and the unit is percentage. The actual movement speed is the maximum speed of the axis in the robot joint parameters multiplied by this percentage.
2. **PL:** Smooth transition level, the range is 0-5.
3. **ACC:** Acceleration ratio, the range is 0-100, it is displayed in percentage.
4. **DEC:** Deceleration ratio, the range is 0-100, it is displayed in percentage.
5. **TIME:** Time, the range is a non-negative integer, and the unit is milliseconds. Advance the time to execute the next instruction.

2.6.1.2.- MOVL

Function: The main function of this instruction is to take the end effector of the robot from one position to another using linear interpolation to move to the target point. When this instruction is used, the robot tries to get to the position without changing the end effector position at any time.

Parameters description:

Figure 2.29: MOVL parameters.

1. **V:** Movement speed, the range is 2-1500, the unit is mm/s.
2. **PL:** Smooth transition level, the range is 0-5.
3. **ACC:** Acceleration ratio, the range is 0-100, it is displayed in percentage.
4. **DEC:** Deceleration ratio, the range is 0-100, it is displayed in percentage.
5. **TIME:** Time, the range is a non-negative integer, and the unit is milliseconds. Advance the time to execute the next instruction.

2.6.1.3.- MOVC

Function: The robot moves in a circle through the 3 points taught by circular interpolation. The starting point of the first arc of a single arc and continuous arc can only be MOVJ or MOVL. It can be used as a single arc or double arc if it is used the endpoint of the first arc as the starting point of the second arc.

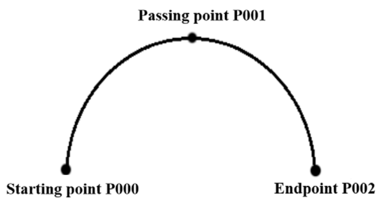


Figure 2.30: Arc movement scheme.

Parameters description:

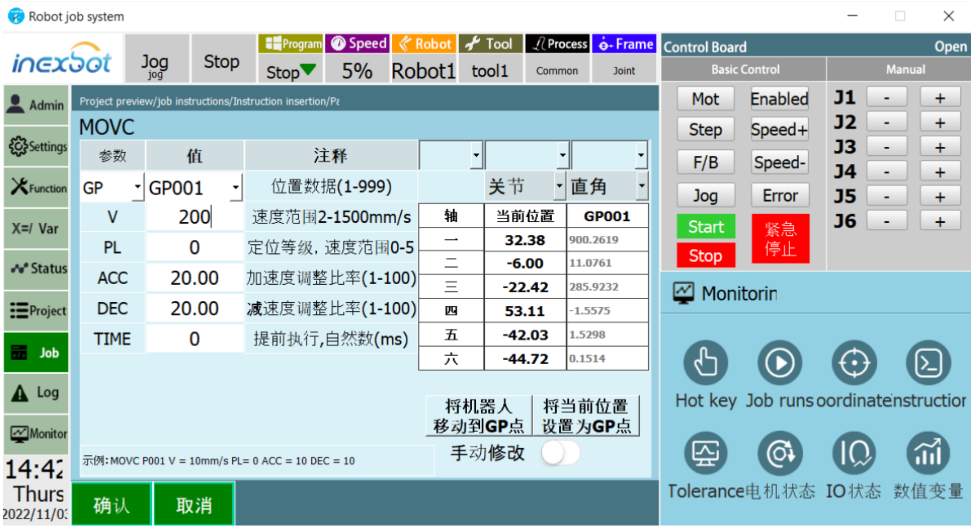


Figure 2.31: MOVC parameters.

1. **V:** Movement speed, the range is 2-1500, the unit is mm/s.
2. **PL:** Smooth transition level, the range is 0-5.
3. **ACC:** Acceleration ratio, the range is 0-100, it is displayed in percentage.
4. **DEC:** Deceleration ratio, the range is 0-100, it is displayed in percentage.
5. **TIME:** Time, the range is a non-negative integer, and the unit is milliseconds. Advance the time to execute the next instruction.

2.6.1.4.- MOVCA

Function: Through the starting point and two passing points of the teaching circle, the robot does a complete circle.

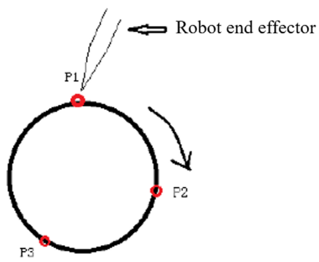


Figure 2.32: Circle movement scheme.

Parameters description:

Figure 2.33: MOVCA parameters.

1. **V:** Movement speed, the range is 2-1500, the unit is mm/s.
2. **PL:** Smooth transition level, the range is 0-5.
3. **ACC:** Acceleration ratio, the range is 0-100, it is displayed in percentage.
4. **DEC:** Deceleration ratio, the range is 0-100, it is displayed in percentage.
5. **TIME:** Time, the range is a non-negative integer, and the unit is milliseconds. Advance the time to execute the next instruction.
6. **SPIN:** Select the type of spin. Unchanged position, 6-axis no rotation, 6 axis rotation.

2.6.2.- Python

Furthermore, it will be used Python for some codes to improve the repeatability of the tests because it is easier to save positions and use them more than once. The robots' software has the option of saving the positions, but is much more difficult.

The Python code is a bunch of definitions for each command needed. It is used a library provided by the robots' manufacturer. This library is private, so it is impossible to know how each command is programmed.

All the defined commands look like the ones in the image below (*Figure 2.34*).

```python
def robot_movl(self, position, speed=10, coord=0, acc=80, dec=80):  # Teaching Mode Linear Motion
    """
            This function realise a linear movement to a position

            :param self: Represents the instance of the class
            :param position: Select the position to be reached
            :param speed: Select the speed of the movement in mm/s (from 1 to 1000)
            :param coord: Select the coordinate system used (0:joint, 1:cartesian, 2:tool, 3:user)
            :param acc: Select the value of the acceleration of the movement
            :param dec: Select the value of the deceleration of the movement
            :return: Nothing
            """
    try:
        self.nrc_lib.robot_movl(position, speed, coord, acc, dec)
    except Exception as e:
        print(e)

def robot_movj(self, position, speed=5, coord=0, acc=80, dec=80):  # Teach Mode Joint Movement
    """
            This function realise a joint movement to a position

            :param self: Represents the instance of the class
            :param position: Select the position to be reached
            :param speed: Select the speed of the movement in % (from 1 to 100)
            :param coord: Select the coordinate system used (0:joint, 1:cartesian, 2:tool, 3:user)
            :param acc: Select the value of the acceleration of the movement
            :param dec: Select the value of the deceleration of the movement
            :return: Nothing
            """
    try:
        self.nrc_lib.robot_movj(position, speed, coord, acc, dec)
    except Exception as e:
        print(e)
```

Figure 2.34: *Python defined commands.*

Once the commands are defined, it is possible to code any tasks needed.

```python
if __name__ == '__main__':
    nrc = robot_nrc_api()  # Define the library robot_nrc_api
    nrc.robot_connect('192.168.2.113', r'\nrc_host.dll')  # Connect the robot using the selected IP

    nrc.nrc_lib.set_servo_state(1)  # Put the servos in ready state
    nrc.robot_servo_on()  # Turn the servos on
    coordinate = int(input("Please select coordinate mode: "))  # Ask for the coordinate mode
    nrc.robot_get_current_position(coordinate)  # Display the current position in the selected coordinate mode
    spd = int(input("Enter the jogging speed: "))  # Ask for the jogging speed
    nrc.robot_set_jogspeed(spd)  # Select the entered jogging speed
    pos = (c_double * 6)(720, 0, 600, 3.12, 1.57, 0)  # Define the first position
    pos2 = (c_double * 6)(30, 37, -35, 0, 92, 5)  # Define the second position
    pos3 = (c_double * 6)(30, -70, -35, 0, 92, 5)  # Define the third position
    nrc.robot_movl(pos, 50, 1, 50, 50)  # Make a linear movement to the first position
    while 1:  # Infinite loop
        poss = nrc.robot_get_current_position(1)  # Define poss as the current position
        sleep(0.2)  # 200 ms delay
        if abs(poss[0] - pos[0]) < 0.05:  # Conditional command that checks if the robots' current position is the final position
            print(poss[0] - pos[0])  # Print the result of the subtraction
            sleep(1)  # 1 s delay
            nrc.robot_movj(pos2, 50, 0, 50, 50)  # Move the robot to the second position
            sleep(1)  # 1 s delay
        if abs(poss[0] - pos2[0]) < 0.05:  # Conditional command that checks if the robots' current position is the final position
            sleep(1)  # 1 s delay
            nrc.robot_movj(pos3, 50, 0, 50, 50)  # Move the robot to the third position
        if abs(poss[1] - pos3[1]) < 0.05:  # Conditional command that checks if the robots' current position is the final position
            nrc.robot_servo_off()  # Turn off the robot
```

Figure 2.35: *Python program example.*

The definitions are part of a demo provided by the manufacturer of the robotic arm; the problem is that some of the functions such as “movc” or “movca” do not work even if you call them the same way it is explained in the documentation. It was asked for help to the support of the company, but their answer was that their engineers have no experience in Python and that they cannot help us. We asked for the libraries used in the code in order to understand better how to use the functions, but they did not want to share the libraries with us.

For this reason, this section is quite short and the implementation of the Python code in the project was limited.

3.- Development of a test protocol

The aim of this project is to study the most important characteristics of industrial robotic arms for the customers, take those parameters and create a test protocol in order to be able to check if the experimental values match the theoretical ones given by the manufacturers. The idea is to create a generic test protocol that could be performed on any kind of robotic arms. To achieve this, it is mandatory to select some KPIs (Key Performance Indicators) for industrial robotic arms. **[3]**

3.1.- Key Performance Indicators

The selected KPI's are defined in the next list.

1. **Static accuracy and repeatability:**

Static accuracy is how close a stage can position to the actual (true) value. Repeatability is a measure of the robot's ability to sequentially position to the same target value.

2. **Maximum speed of each joint:**

This parameter describes which is the maximum speed a joint can perform.

3. **Maximum speed of the robot:**

This parameter describes which is the maximum speed the whole robot can perform.

4. **Maximum load:**

This parameter describes which is the maximum load applied to the end effector of the robotic arm without affecting its performance.

5. **Power consumption:**

This parameter describes which is the energy the robot consumes in different stages.

6. **Path accuracy and repeatability:**

Path accuracy characterizes the ability of a robot to move its mechanical interface along the command path in the same direction n times. Path repeatability expresses the closeness of the agreement between the attained paths for the same command path repeated n times.

7. Maximum range:

This parameter describes which is the maximum range each joint can reach.

3.2.- Study of the alternatives

3.2.1.- Static accuracy and repeatability

Definition of the test:

The accuracy and repeatability are very important characteristics in robotic arms because are usually used in repetitive applications and, depending on the task, being able to repeat the same position thousands of times with the best precision possible is needed. This test focuses on the capacity of the robotic arm to reach the same position every time and ensure that this position is the correct one. Then could be interesting to repeat the test with our drives and see if there is an improvement of this parameter.

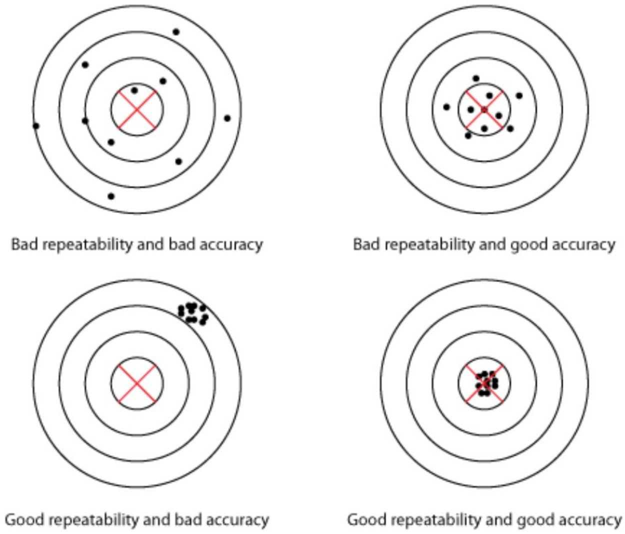


Figure 3.1: *Accuracy and repeatability diagrams.* ***[5]***

The ISO 13309 gives examples of testing the static and dynamic repeatability and accuracy in accordance to ISO 9283. **[17] [18]**

For static repeatability and accuracy, the norm shows a few more examples.

Trilateration methods

Trilateration (meaning "using three sides") is a method of determining the Cartesian coordinates (x, y, z) of a Point P in three-dimensional space with three distance values between the Point P and the three observation stations, and the base lengths between three fixed stations. The figure below (*Figure 3.2*) explains the principle of trilateration in two-dimensional representation.

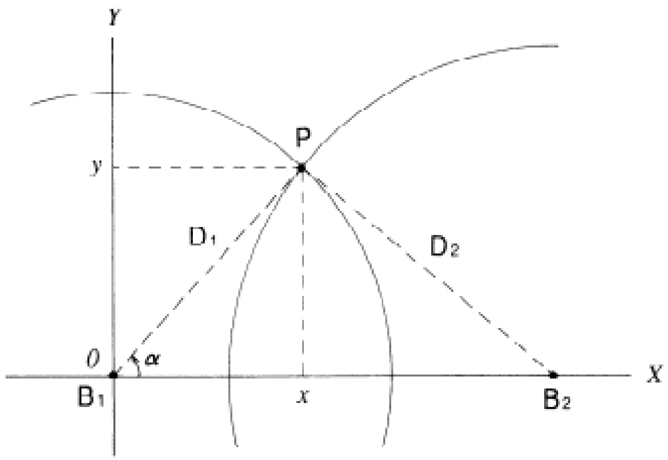


Figure 3.2: *Trilateration representation.* ***[18]***

Multi-laser tracking interferometry

This method is based on using three laser beams produced from three laser interferometers with two-axis servo-controlled tracking aimed at a common target located on the robot's wrist. The system setup is shown in the figure below (*Figure 3.3*). The robot pose characteristic in three-dimensional space ca n be determined based on distance data obtained from the three interferometers. The orientation can be measured if six interferometers are used in a setup in which the six beams are aimed at three independent targets on the robot.

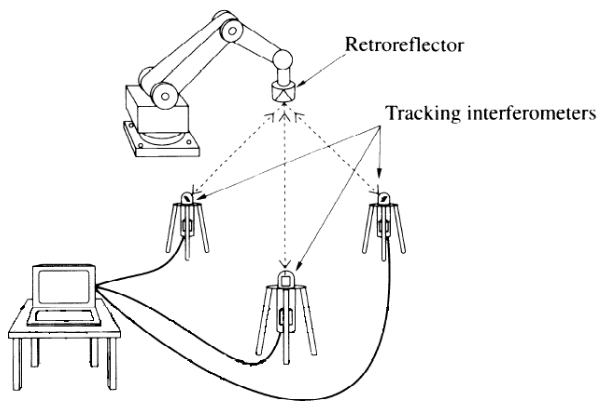


Figure 3.3: *Multi-laser tracking interferometry setup.* ***[18]***

Ultrasonic trilateration

The robot's position in three-dimensional space can be calculated with distance data from three stationary ultrasonic microphones, which receive ultrasonic pulse trains from a sound source mounted on the robot. The system setup is shown in the image below (*Figure 3.4*). The robot's orientation can be measured if the robot has three independent sound sources and each stationary microphone can detect pulse trains from all three sound sources.

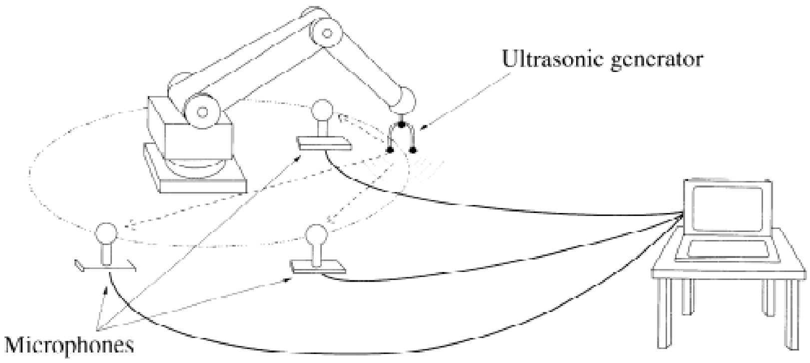


Figure 3.4: *Ultrasonic trilateration setup.* ***[18]***

Mechanical cable trilateration

This method is based on connecting three cables originated from three fixed cable-feeding devices to the robot's end effector, as shown in the figure below (*Figure 3.5*). By evaluating the length of each cable, such as using potentiometers or encoder on the cable feeding devices which maintain the cables under tension, the position of the robot's end effector can be determined.

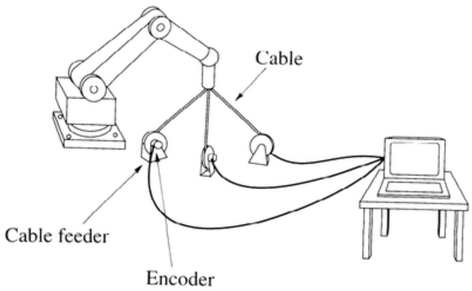


Figure 3.5: *Mechanical cable trilateration setup.* ***[18]***

Inertial Measuring Unit (IMU)

The IMU is placed in the end effector of the robotic arm. Knowing the initial position of the robotic arm, with the accelerations and angles read by the sensors, it can be calculated the final position of the end effector. The biggest problem of this method is that the compounding error in the calculus done would be enormous.

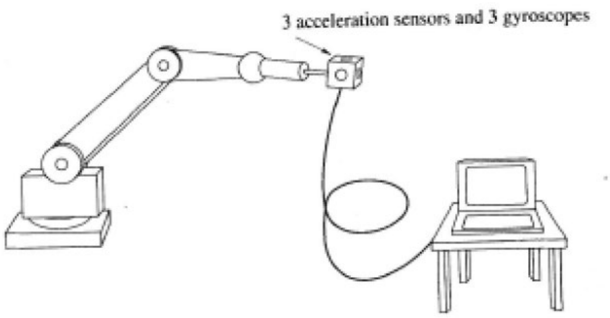


Figure 3.6: *Inertial Measuring Unit setup.* ***[18]***

Polar coordinate measuring methods

Polar coordinate measuring methods can be used to determine the Cartesian coordinate (x,y,z) of a point in space by measuring a distance (D), azimuth (α) and elevation (β) values as shown in the figure below (*Figure 3.7*).

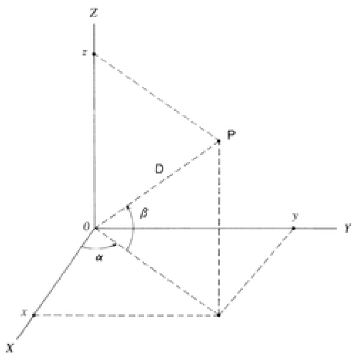


Figure 3.7: *Polar coordinate representation.* ***[18]***

Single laser tracking interferometry

The laser tracking interferometry method can be used to measure the robot's position or orientation. The robot's position can be calculated with the distance data from the laser interferometer and azimuth/elevation data which is obtained from a stationary tracking system aimed at a retroreflector mirror mounted on the robot's end point.

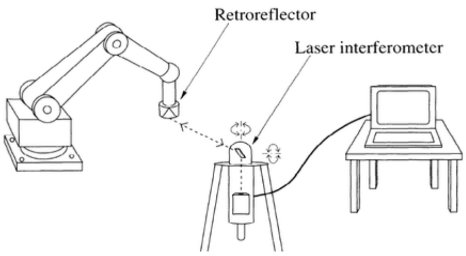


Figure 3.8: Single laser tracking interferometry setup. [18]

The robot's orientation (pitch and yaw) can also be measured using the same system if the retroreflector mirror system has the capability of keeping its optical axis pointed to the stationary tracking system, or if the stationary tracking system can analyse the diffracted image reflected by the retroreflector. This method can test 6 DoF robots.

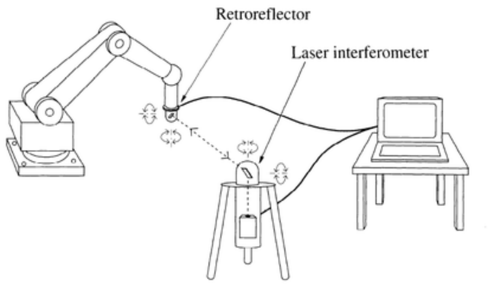


Figure 3.9: Setup with orientation. [18]

Proposed tests:

This test is the most difficult to perform and also the most expensive one. There are lots of variants, but it is needed to choose the one that matches our budget cap and the technical necessities.

Linear potentiometers: This is the cheapest option, very simple to use and to digitalize the data. The big con it has is that the potentiometers have errors due to tolerances in the resistance, tolerances in the distance, and this would give us a very poor result.

Figure 3.10: Example of a linear potentiometer.

LVDT sensors: This option uses maybe the most precise industrial linear displacement sensors, they are quite easy to digitalize, the big problem is that those sensors need conditioning electronics to provide the data we want. Furthermore, these sensors are very expensive and would hinder respect for the budget cap.

Figure 3.11: *Example of a LVDT sensor*

Digital dial indicators: This option is one of the best, it has a very high precision, are very easy to digitalize because can include Bluetooth and a software which displays the data in real time and can also create diagrams. Those sensors are very expensive, but we could afford them. **[9] [10]**

Figure 3.12: *Example of a dial indicator.*

Ballbar test: This test is one of the best we could perform, it is one of the most precise tests for repeatability and accuracy, it follows the ISO 9283:1998 norm and has a software that provides a lot of data of errors such as scale mismatching, squareness, backlash, servo mismatch… This is one of the most used tests for robots, CNCs and other machinery. The biggest problem is that this system is totally out of our budget cap. **[11]**

Figure 3.13: *Example of the ballbar test.* ***[11]***

Radian laser tracker test: The last test is, like the previous one, is one of the most precise tests for repeatability and accuracy and also follows the ISO 9283:1998 norm. Even so, it has the same problem, it is totally out of our budget cap. **[12]**

Figure 3.14: *Example of a radian laser tracker.*

3.2.2.- Maximum speed of each joint

Definition of the test:

The idea of this test is to determine what is the speed of each axis, so we can compare it to the specifications the robot manufacturer provides in the datasheets. Then could be interesting to repeat the test with our drives and see if there is an improvement of this parameter.

Proposed tests:

The most simple way to test this parameter is to make the robot do a task or to move from one position to another at maximum speed and calculate the time the robot spends in reaching two different points in the trajectory of the movement. To calculate the time we will need a sensor because using a chronometer would introduce a lot of error in the measurements which must be the most precise possible.

Industrial fork barrier: One option is to choose an industrial sensor which will have the best characteristics for the test and will provide the best performance. The cons are that these kinds of sensors are very expensive, and the repeatability test requires a big part of the budget cap. Furthermore, the less expensive industrial fork barriers have very little space between the laser emitter and receiver and this could be dangerous because if the robotic arm has an error in the trajectory could damage the sensor.

Figure 3.15: *Industrial fork barrier.*

Self-made fork barrier: A second option could be building the fork barrier using Arduino laser emitters and receivers and designing a 3D part to attach those sensors, just like an industrial fork barrier. This option is much cheaper than the first one, and the only con would be that maybe these sensors are slower and could add a delay to the measures.

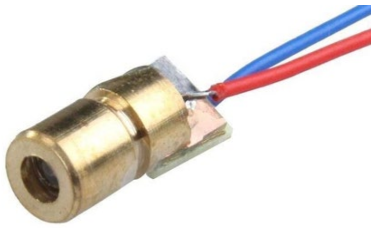


Figure 3.16: *Laser emitter.*

End limit switch: A different option could be buying end limit switches, which are cheaper than the industrial fork sensors and maybe faster than the DIY fork barrier. Even so, this option has a big problem, which is that those sensors are mechanical, and the robot would have to collide with them during the test and the sensors could be damaged.

Figure 3.17: *Limit switch.*

3.2.3.- Maximum speed of the robot

Definition of the test:

For this test, we previously should detect which kind of tasks are more interesting to companies. It depends on the characteristics of the robotic arm to be better at one task or another. In our case, the CM607-L is an industrial robotic arm with a great range, quite good speed but with a small payload and not as good repeatability as other manufacturers. Due to these specifications, the companies would not use this kind of robotic arm for collaborative applications or high payload applications such as palletizing. In the top 5 applications for robotic arms this last years we have pick and place tasks, our robot could perform really good in these kinds of tasks, so we could focus this test on different pick and place applications with different characteristics and measure how fast the robot can perform. Those tests must be easily repeated in the future, so we can precisely detect if there is any change when changing the drives.

Proposed tests:

Chronometer: We could use a simple chronometer to measure the start and the end of the pick and place program, but this will bring a lot of error in the measurements. We could also start and stop the chronometer using the robotic arm, but this would be very dangerous and could damage the chronometer. For this reason, even though this test is very cheap can bring a lot of problems.

Figure 3.18: *Chronometer.*

Self-made fork barrier: A second option could be to use the fork barrier from the speed of each joint test. This option is cheap too, but would be very difficult to place correctly the sensors at the beginning and end of the task without introducing delays.

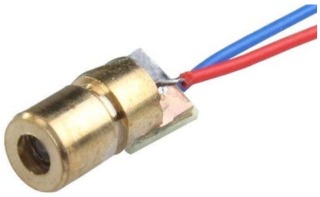


Figure 3.19: *Laser emitter.*

End limit switch: A different option could be using the end limit switches. This option has a big problem, which is that those sensors are mechanical, and the robot would have to collide with them during the test and the sensors could be damaged, besides this kind of sensor would introduce delays in the measurements too.

Figure 3.20: *Limit switch.*

Software: The software provides some instructions that can add a clock to the codes and save the data of the calculated time.

Figure 3.21: *Robot's GUI.*

3.2.4.- Maximum load

Definition of the test:

One of the most important characteristics of a robotic arm may be the maximum load it can have without losing performance, for this reason, it would be very interesting to prove that the data the manufacturer of the robot gives is correct. This test is focused on finding the maximum and rated load of the robot and comparing those results to the specifications that the manufacturer provide. Once identified those characteristics, the test will focus on developing diagrams such as range vs weight, speed vs weight, etc.

Proposed tests:

There are two clear variants to perform this test, one is more complicated to implement, and the other one is easier.

Force sensor: The first possible option is to use a force sensor that could be coupled to the robot with a custom end effector, so we could measure the pressure we are applying. The principal problem is that we would need conditioning electronics, so the end effector would be bigger than needed. Another problem is that usually these kinds of sensors are only for one type of force (compression or tension) so they are more limited than the dynamometer and if they can measure both types of forces, the price increases a lot, and we could create a part to permit the force sensor to measure both forces but would be more difficult. The pros are that it is not very expensive and would help us measure the dynamic force that is applied to the arm when doing a task with weights on the end effector. The cons are the complexity of adding this sensor to the robot and that this method would not allow to us reuse the same sensor on another robotic arm.

Figure 3.22: *Force sensor.* ***[13]***

Dynamometer: The second option is buying a digital dynamometer which is able to measure both tension and compression forces, does not need any conditioning electronics or MCUs and can save the maximum value measured during a test, and it could be used in future robotic arms. The cons are that is not able to measure the dynamic force.

Figure 3.23: *Digital dynamometer.*

Force/torque sensor: One possibility could be using a force sensor and attaching it to the end effector of the robotic arm and perform different linear movements and see how the force is changing in each axis, like this we could know what is the dynamic accuracy of the robot. The biggest cons are that we would have to use mathematics to convert the newtons provided by the sensor into distance, and this could bring errors. Another problem is that those sensors are very expensive, and the budget cap would not allow us to get one of them. We could ask ATI for a force/torque sensor for free. **[14]**

Figure 3.24: *Force/torque sensor.*

3.2.5.- Power consumption

Definition of the test:

We could measure the power consumption of the robot on standby, on a pick and place application with different speeds and weights and create diagrams with all this data, then when changing the drives and using ours, we could repeat the process to compare how much energy we could save by implementing this change, and then we could do research of how much money could save to companies by using our drives instead of the ones of the competence. This is a test focused on a more economical way and not on a technical one, but could help on sharing companies with the idea that they could become more "eco-friendly" and not only win time (if we can increase the speed of the robot) but win money too (if we can decrease the power consumption) by just using our drives.

Proposed tests:

Electricity meter: The first possibility is to use an electricity meter, a device that can measure the active energy of three phases at once and display the value. Those are very easy to use by only connecting the three phases and the neutral and can be very versatile because could be used for any AC device with three phases. The cons are that those are expensive devices and to connect the wires we should cut them, and this could be a problem.

Figure 3.25: *Electricity meter.*

Oscilloscope: Using an oscilloscope would be the cheapest option, just because we have a very good oscilloscope that could be used. The measures would be quite precise, and it could display the phases on the screen, not only the value of the energy. The cons are that the connection would be more difficult, and we would have to calculate the energy because the oscilloscope can read only voltage and current.

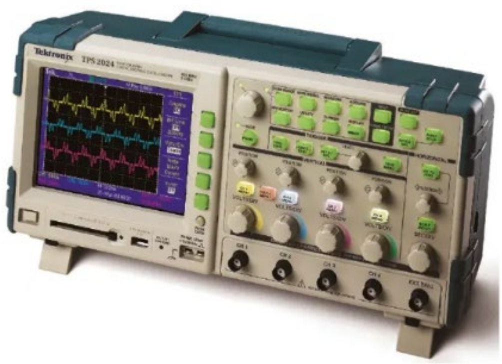


Figure 3.26: *Oscilloscope.*

Kill-a-Watt monitor: This last option is not that expensive and is the easiest one because it is not necessary to cut the wires, it is just like a socket that displays the voltage, current and energy consumption. The biggest cons are that those devices have less accuracy.

Figure 3.27: *Kill-a-Watt monitor.*

3.2.6.- Dynamic accuracy and repeatability

Definition of the test:

Path accuracy and repeatability describes a robot's ability to repeatedly move its end effector along the same commanded path in the same direction. It is defined as the maximum path deviation along the commanded path in terms of positioning and orientation.

For dynamic repeatability and accuracy, the norm shows two examples.

Mechanical gage

This method is based on comparing an attained path with a command path, which could be composed of linear or circular path segments. The path is constructed using a precision mechanical gage or other position reference structure. The image below (*Figure 3.28*) shows a setup for the method where the proximity sensors are fitted on a cube probe and the artefact is a straight edge representing the command path. Deviations occurring during the execution of the path are sensed by an appropriate number of sensors and used to determine characteristic parameters (accuracy and repeatability) of the attained path. Complete pose deviations (position and orientation) can also be determined when sufficient sensors are used. **[18]**

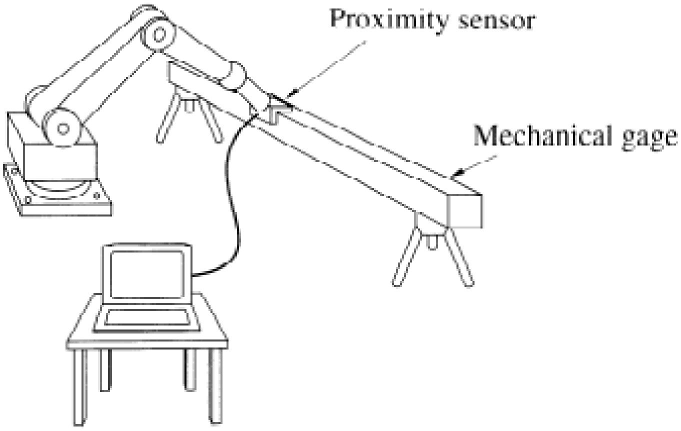


Figure 3.28: Dynamic repeatability & accuracy test setup. [18]

This test could also be performed with dial indicators.

Laser beam

Path accuracy/repeatability along a laser beam can be measured with a photosensitive transducer which has the capability of detecting the position error of the incident beam from the centre of the transducer. The system setup is shown in the figure below (*Figure 3.29*). The robot's pose along the beam can be calculated as a function of time if the laser source is replaced by a laser interferometer and the photosensitive transducer has light reflecting capability.

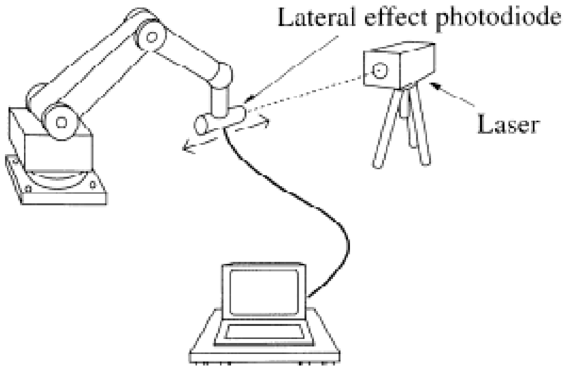


Figure 3.29: Laser beam setup. [18]

Proposed tests:

Force sensor: One possibility could be using a force sensor and attaching it to the end effector of the robotic arm and perform different linear movements and see how the force is changing in each axis, like this we could know what is the dynamic accuracy of the robot. The biggest cons are that we would have to use mathematics to convert the newtons provided by the sensor into distance, and this could bring errors. Another problem is that those sensors are very expensive, and the budget cap would not allow us to get one of them. We could ask ATI for a force/torque sensor for free. **[14]**

Figure 3.30: *Force/torque sensor.*

Digital dial indicator: This option has a very big benefit and is that we could reuse the same digital dial indicator from the repeatability test, so it would not affect our budget cap.

Figure 3.31: *Bluetooth dial indicator.*

Laser based test: a 5.5 mm-wide 'curtain' of 1000 laser beams would allow contact-free path measurements in two dimensions. The curtain could be created by a precision-aligned laser transmitter/receiver pair.

Figure 3.32: *Laser 'curtain'.*

3.2.7.- Maximum range

Definition of the test:

This parameter seems not important because the manufacturer shares it, and it is related to the mechanical links. Even so, we want to prove that it is true what they say. Furthermore, when changing the drives, we could remove the mechanical limits and increase the maximum range of the axis, but taking care of the internal wiring of the robot, because like that we could tense too much the wires and break down some of them and this will cause serious damage on the robot.

Proposed tests:

Controller's software: It could be used the software provided by the manufacturer which has an option to calculate the distance in degrees that each axis is moving, the only problem is that we do not know the real accuracy of this measure.

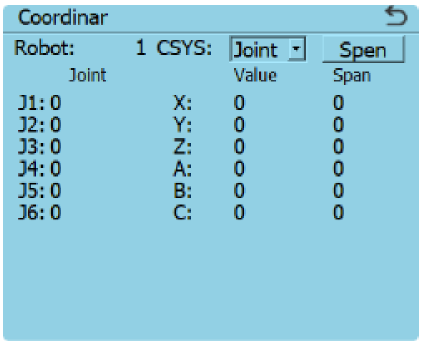


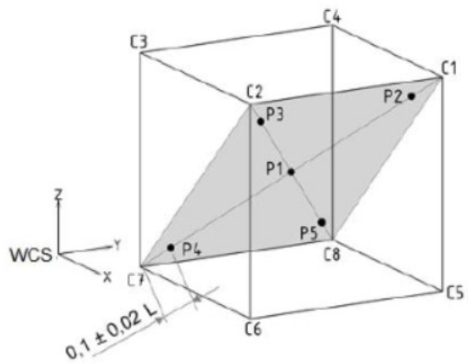
Figure 3.33: *Software's measuring tool.*

3.3.- Selected tests

3.3.1.- Static accuracy and repeatability

For this test, it is indispensable to use high precision tools or at least with better precision than the rated repeatability of the robotic arm, such as dial indicators.

The ISO 9283:1998 standard prescribes some positions to measure the accuracy and repeatability of robotic arms. The measurement consists of 30 measuring cycles, within which the TCP point moves to individual points, i.e., tested positions P5 to P1. Each of the points is taken by one-directional approaching of TCP point. Coordinates of each point are measured after reaching its pose, and then the accuracy and repeatability are calculated. **[17]**



(L is the diagonal length in mm)

Figure 3.34: *ISO cube for static repeatability & accuracy tests.* ***[17]***

ISO 9283:1998 norm defines the further conditions for the position performance testing. They are related to ambient temperature, loading of the end of the robot´s arm and speed of TCP´s movement.

By placing three dial indicators in the five positions that the ISO 9283:1998 defines, we could read the three-axis value at each repetition and calculate more precisely the performance.

The idea is to develop a setup which replicates the ISO cube in a 3D structure with the five positions. This should be mechanized with a very high accuracy in order to achieve the most precise results possible, an example of this ISO cube can be seen on the CoRoETS YouTube channel. **[4]**

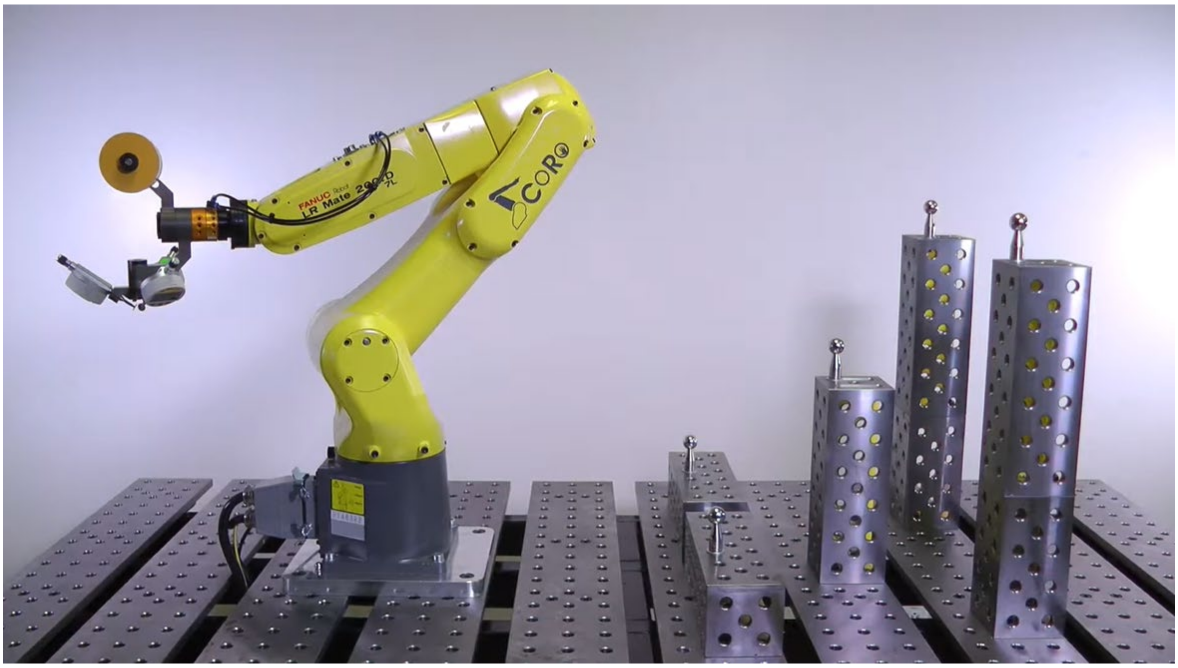


Figure 3.35: *Example of a mechanized ISO cube.* ***[4]***

The ideal test should be performed with three dial indicators as it can be seen in the image below (*Figure 3.36*) and reach each one of the positions once the dial indicators are correctly calibrated. Knowing the exact position of the spheres, we can calculate the accuracy and repeatability with this setup.

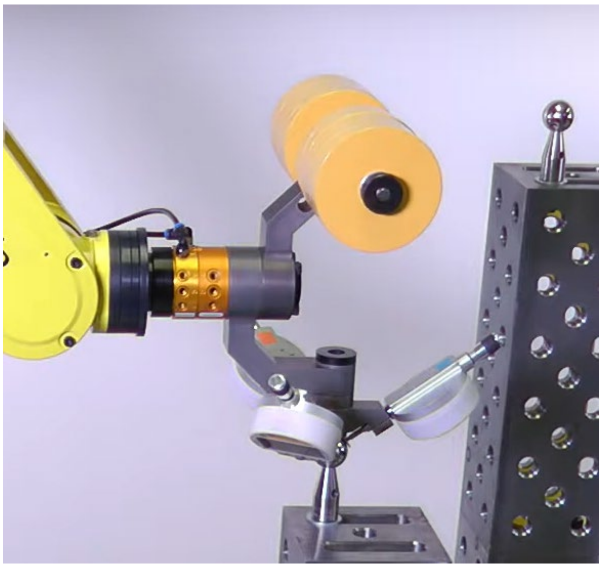


Figure 3.36: *Example of static repeatability & accuracy test.* ***[4]***

We could define an ISO cube as the one in the image above (*Figure 3.36*) and then create a real diagonal plane with the 5 positions marked on it and place it in a rigid structure, so there is no minimum dispersion on the positions. This would be the most interesting option and the chosen now but using only one dial indicator.

The test must be done at 100% of load, in our case this means applying 6 kg to the end effector. The 30 cycles mentioned in the ISO norm must follow one of the next options.

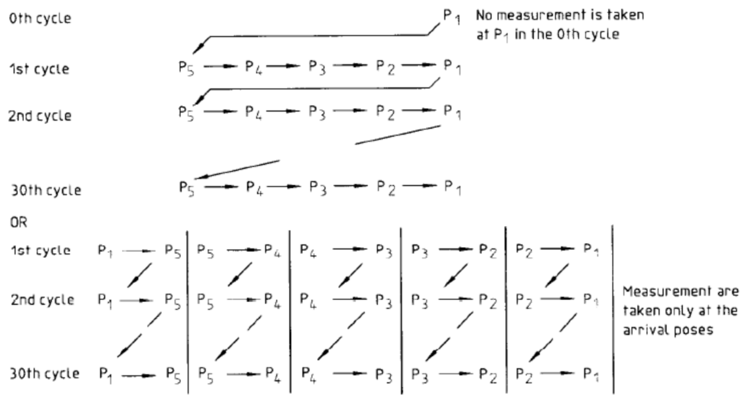


Figure 3.37: *Commanded path for the repeatability & accuracy test.*

Both options are correct, so the engineer in charge of the test could choose the one he thinks is easier to develop, or it could be performed in both options to achieve a better result.

Another option is to take that ISO cube and divide each axis into 6 positions, then we could measure only the unidirectional accuracy and repeatability and implement a code that works forward and backwards just to improve the measurements taken. This would help us because we would have to buy only one item and reduce a lot the budget cap. This cube is not the one shown in the ISO 9283:1998 norm, so we should not use this option as the main one, only as a support test. **[17]**

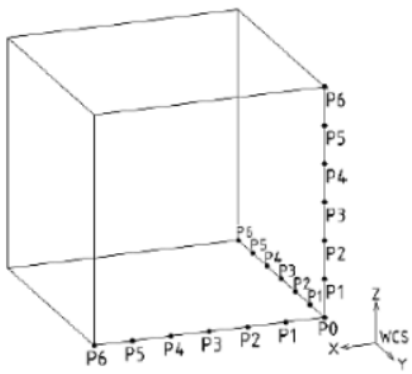


Figure 3.38: *Variation of the ISO cube.*

The **position accuracy (AP)** represents the deviation between the TCP point´s commanded position (N) and the mean value (barycentre; G) calculated from the cluster of TCP´s positions reached repeatedly.

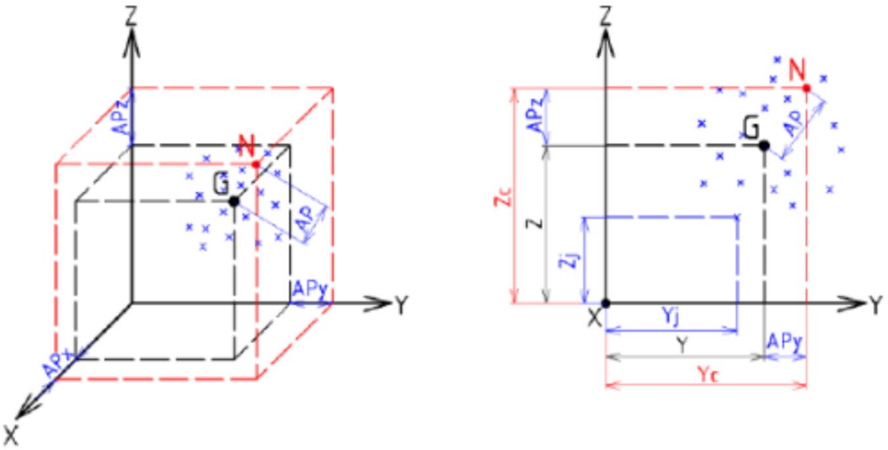


Figure 3.39: *Position accuracy definition.*

The accuracy (AP) is given by the next formulas.

$$AP = \sqrt{AP_x^2 + AP_y^2 + AP_z^2}$$

Equation 1

$$AP_x = \bar{x} - x_c$$

Equation 2

$$AP_y = \bar{y} - y_c$$

Equation 3

$$AP_z = \bar{z} - z_c$$

Equation 4

Where coordinates of the barycentre are given by.

$$\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_j$$

Equation 5

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

$$\bar{y} = \frac{1}{n}\sum_{j=1}^{n} y_j$$

Equation 6

$$\bar{z} = \frac{1}{n}\sum_{j=1}^{n} z_j$$

Equation 7

**$\bar{x}, \bar{y}, \bar{z}$ are the coordinates of the barycentre of the cluster of points, obtained after repeating the same pose n times.*

***x_c, y_c, and z_c are the coordinates of the command pose.*

**** x_j, y_j, and z_j are the coordinates of the j-th attained pose*

The **position repeatability (RP)** is defined as "the closeness of agreement between the attained positions after n repeat visits TCP point to the same command pose in the same direction". RP represents the radius of a sphere, which centre is a barycentre.

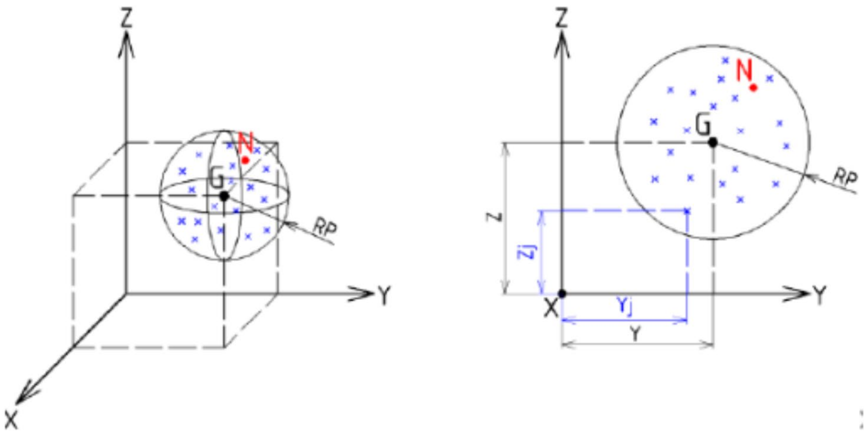


***Figure 3.40:** Position repeatability definition.*

For a given pose, the repeatability is expressed by.

- The value of RP_l, which is the radius of the sphere whose centre is the barycentre, and which is calculated as below.
- The spread of angles $\pm 3S_a$, $\pm 3S_b$, $\pm 3S_c$ about the mean values $\bar{x}, \bar{y}, and\ \bar{z}$ where S_a, S_b, and S_c are the standard deviations.

where,

$$RP_l = \bar{l} + 3S_l$$

Equation 8

with

$$\bar{l} = \frac{1}{n}\sum_{j=1}^{n} l_j$$

Equation 9

$$l_j = \sqrt{(x_j - \bar{x})^2 + (y_j - \bar{y})^2 + (z_j - \bar{z})^2}$$

Equation 10

with

$$S_l = \sqrt{\frac{\sum_{j=1}^{n}(l_j - \bar{l})^2}{n - 1}}$$

Equation 11

3.3.2.- Maximum speed of each joint

The chosen test is to design a fork light barrier, as can be seen in the diagram, and put the sensors in two different positions in the trajectory of the arm. These positions should be located far enough from the starting and ending point to avoid the effect of the acceleration and deceleration because it would change the result of the test. Then the data of the sensor would go to an MCU with an external global clock to measure the time it takes the robot to get from one sensor to another. With this time and knowing how many degrees the joint has moved, we could calculate the real speed of the tested axis. Then this process would be repeated for each axis. This test is easy to digitalize, with a simple esp32 or Arduino we could detect when the robotic arm cuts the laser and with an external clock. Then we could do a program that calculates the speed of the joint by entering the number of degrees the robot is moving and send all this data to an Excel or other file. The test should be repeated a minimum of 30 times (which is specified in the ISO 9283:1998) to get a valid result. **[25]**

This test consists of placing two fork laser barrier sensors in the trajectory of the movement of the robot and calculating the time it takes the robot to get from the first sensor to the second one.

It will be used a Teensy 4.1 board, a very powerful 600 MHz MCU and compatible with the Arduino IDE. This MCU has a very accurate quartz crystal which also has a temperature compensation, this is very important for getting better results on the speed tests. **[29]**

Figure 3.41: *Teensy 4.1.* ***[29]***

The clock used for this application must be very precise, so the calculated time can be considered correct. In this case, the clock used will be the one the Teensy 4.1 board has implemented. This clock has an average error of 21 microseconds every second at 25 degrees Celsius, this is the same as 1.81 seconds every day or maybe is more suitable for our application an error of 1.26 milliseconds every minute calculated. A 21-ppm error is totally accepted, so we could consider the measures are correct.

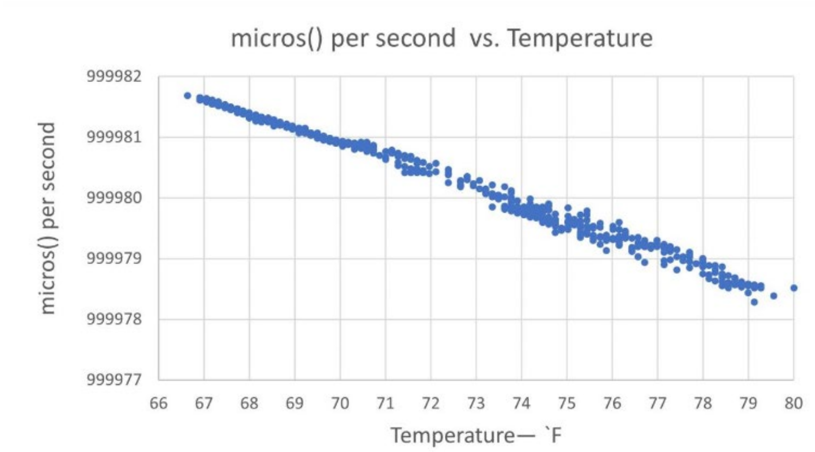


Figure 3.42: *Lose of performance of the time measurement against temperature.* ***[30]***

The laser barrier will be designed in 3D, and it will be composed of two elements, a laser emitter, and a laser receiver. The laser emitter has two wires, 5V and GND, and when supplied starts emitting a red light with 950 nm wavelength. The laser receiver is basically a sensor that provides 5V when detecting a laser light and 0V when not. The distance between the two components should be at least 5 cm to ensure that when the robot goes through the fork sensor this is not damaged.

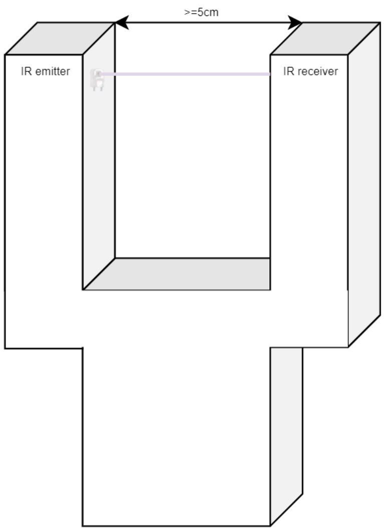


Figure 3.43: *Self-made fork barrier.*

For easier manipulation of the board, a shield will be developed with terminal block connectors to ensure the wires are properly connected.

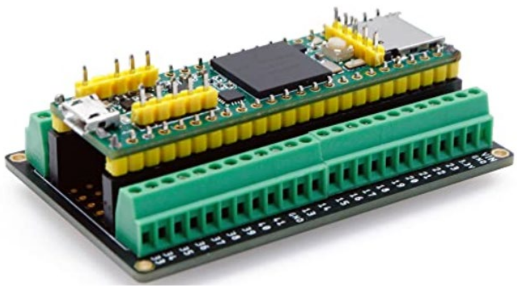


Figure 3.44: *Teensy 4.1 shield board.*

3.3.3.- Maximum speed of the robot

The ISO 9283:1998 norm shows a recommended path to develop some of the tests, such as the path accuracy and path repeatability. They present two variants, one with a plane of 800 mm and another one with 400 mm. For both, they give a table with the position of each point. The benefit of this path is that it would be common on all the robotic arms, and it would be easier to compare their velocities.

The path consists of a 28 points circuit in which the robot must do linear movements, arcs, circles and other to make the robot suffer in changing directions and accelerating and decelerating. **[17]**

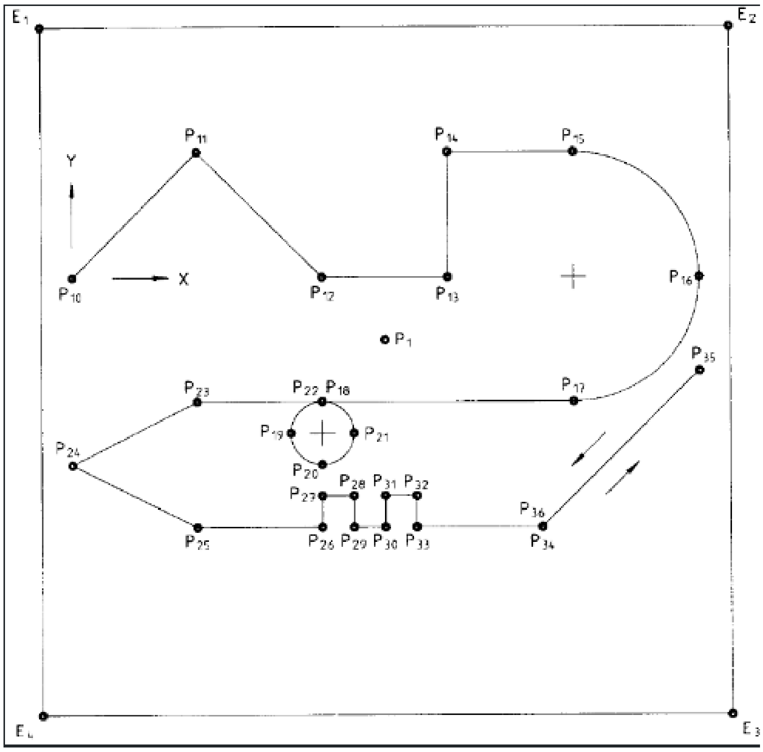


Figure 3.45: *ISO9283:1998 path for speed test.* ***[17]***

And each point for each plane is documented in the following table.

Point	Plane 400 x 400 mm		Plane 800 x 800 mm	
	X =	Y =	X =	Y =
1	180	- 50	360	- 100
10	0	0	0	0
11	70	70	140	140
12	140	0	280	0
13	210	0	420	0
14	210	70	420	140
15	280	70	560	140
16	360	- 10	720	- 20
17	280	- 90	560	- 180
18	140	- 90	280	- 180
19	120	- 110	240	- 220
20	140	- 130	280	- 260
21	160	- 110	320	- 220
22	140	- 90	280	- 180
23	70	- 90	140	- 180
24	0	- 130	0	- 260
25	70	- 170	140	- 340
26	140	-170	280	- 340
27	140	- 160	280	- 320
28	150	- 160	300	- 320
29	150	- 170	300	- 340
30	160	- 170	320	- 340
31	160	- 160	320	- 320
32	170	- 160	340	- 320
33	170	- 170	340	- 340
34	260	- 170	520	- 340
35	360	- 70	720	- 140
36	260	- 170	520	- 340

Figure 3.46: *Commanded positions for the speed test.*

Even though this test is shown in the ISO norm, another important test to measure the maximum speed of a robotic arm is the Adept cycle, which is given in almost all the datasheets of robotic arms.

The Adept cycle, which is defined as the time it takes a robotic arm to perform continuous path, straight-line motions of 25 mm up, 305 mm over, 25 mm down, and back along the same path.

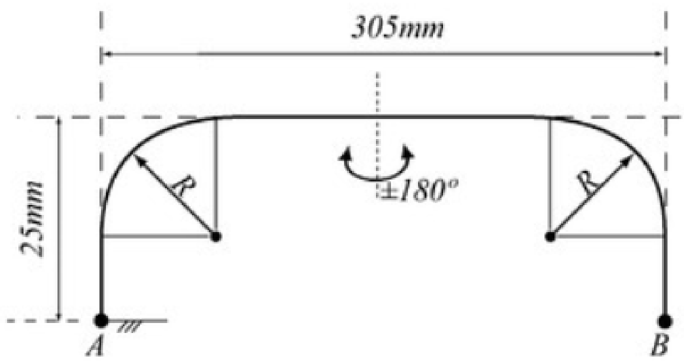


Figure 3.47: *Diagram of the Adept cycle path.*

3.3.4.- Maximum load

Due to some interests in the company, for this test will be used a force/torque sensor from ATI. This is because both companies belong to the same group of companies.

The selected sensor is the ATI Axia80, all the end effectors must be designed to allow the attachment of this sensor. **[14]**

Figure 3.48: *ATI Axia80 force/torque sensor.*

This sensor will be placed between the end effector and the robot's end flange, like this we will be able to measure the forces in the three axis and also the torque with a very high precision. This sensor could help in the measurement of some other tests such as, the static and dynamic repeatability and accuracy tests, the load and the torque tests.

3.3.5.- Maximum power consumption

The selected test is to use the oscilloscope we already own because it is one of the devices with more accuracy we could get and the fact that can display the waves and the values of the measures. The connection we could use is a 2V2I that uses only 4 channels of the oscilloscope. **[31] [32]**

Blondel's theorem states that for an N-wire system, with voltage measured relative to one of the wires, total power can be measured using N–1 wattmeters or in that case, two voltage, current measures.

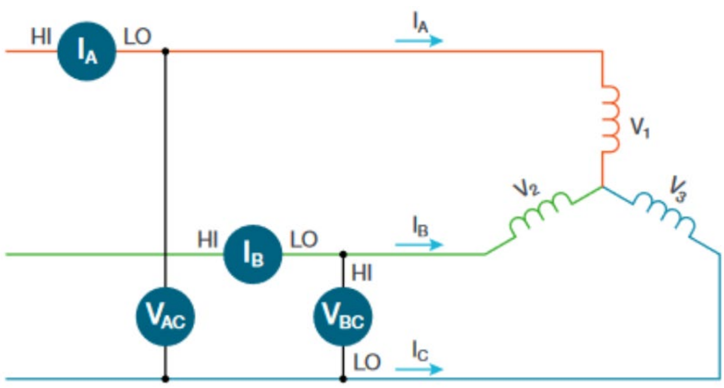


Figure 3.49: *Blondel's theorem diagram.*

The instantaneous power measured by each wattmeter is the product of the instantaneous voltage and current samples.

The first wattmeter is composed of $\mathbf{i_A}$ and $\mathbf{v_{AC}}$, where,

$$p1 = i_A(v_{AC}) = i_A(v_1 - v_3)$$

Equation 12

The second wattmeter is composed of $\mathbf{i_B}$ and $\mathbf{v_{BC}}$, where,

$$p2 = i_B(v_{BC}) = i_B(v_2 - v_3)$$

Equation 13

$$p1 + p2 = i_A(v_1 - v_3) + i_B(v_2 - v_3) = i_A v_1 - i_A v_3 + i_B v_2 - i_B v_3$$

$$p1 + p2 = i_A v_1 + i_B v_2 - (i_A + i_B) v_3$$

Equation 14

Per Kirchhoff's Current Law,

$$i_A + i_B + i_C = 0, \text{so } i_A + i_B = -i_C$$

Equation 15

Substituting for **($i_A + i_B$)** in Equation 14,

$$p1 + p2 = i_A v_1 + i_B v_2 + i_C v_3$$

Equation 16

Thus, the total power in the 3-wire system can be determined by using two voltage channels and two current channels to form two wattmeters. **[33] [34]**

3.3.6.- Path accuracy and repeatability

The test it could be performed by taking one or two dial indicators and using them as tracers by attaching them to the end effector of the robot and creating a track made of a very smooth material and making a linear movement with the robot and recording all the data of the dial indicator, so we would see any displacement, once repeated this test a number of times we would have the dynamic accuracy of the robot in a particular axis. This test should be repeated in all axes and against gravity too. If we use the Bluedial product, we could see all the data displayed in a diagram. **[17]**

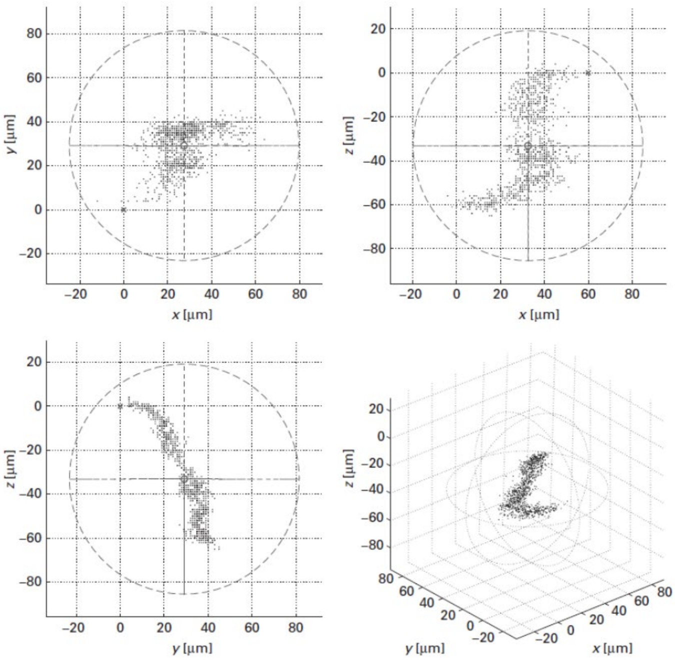


Figure 3.50:** Dynamic repeatability & accuracy test diagram.* ***[17]

There are algorithms presented in the ISO standard 9283 and will be the ones used in this test.

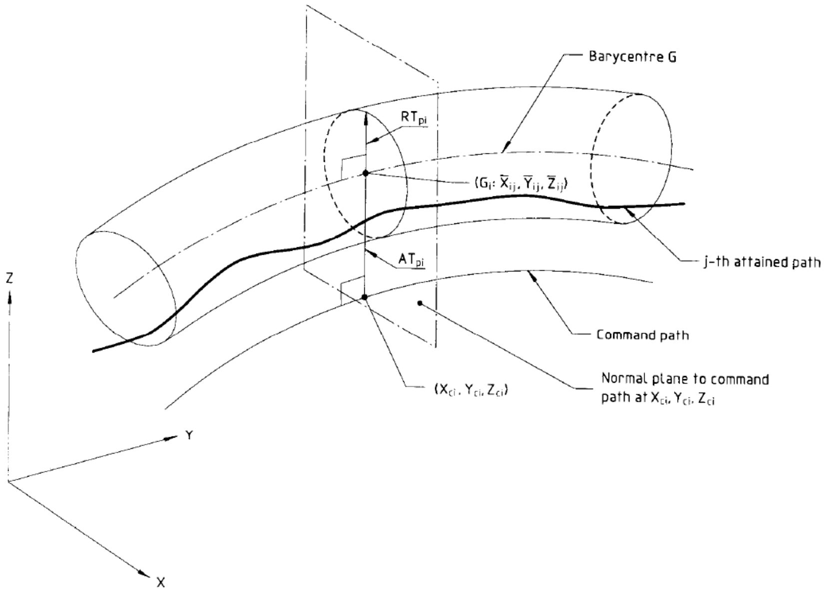


Figure 3.51:** Dynamic repeatability & accuracy test example.* ***[17]

Path accuracy characterizes the ability of a robot to move its mechanical interface along the command path in the same direction n times.

Path accuracy is determined by two factors.

- The difference between the positions of the command path and the barycentre line of the cluster of the positions of the attained paths
- The difference between command orientations and the average of the attained orientations

The path accuracy is the maximum path deviation along the path obtained in positioning and orientation.

$$AT_P = max\sqrt{(\overline{x_i} - x_{ci})^2 + (\overline{y_i} - y_{ci})^2 + (\overline{z_i} - z_{ci})^2} \qquad i = 1 \ldots m$$

***Equation** 17*

where,

$$\overline{x_i} = \frac{1}{n}\sum_{j=1}^{n} x_{ij}$$

Equation 18

$$\overline{y_i} = \frac{1}{n}\sum_{j=1}^{n} y_{ij}$$

Equation 19

$$\overline{z_i} = \frac{1}{n}\sum_{j=1}^{n} z_{ij}$$

Equation 20

Path repeatability expresses the closeness of the agreement between the attained paths for the same command path repeated n times.

For a given path followed *n* times in the same direction, path repeatability is expressed by,

- RT_p is the maximum RT_{pi} which is equal to the radius of a circle in the normal plane and with its centre on the barycentre line.
- The maximum of the spread of angles about the mean value at the different calculated points.

The path repeatability is calculated as follows.

$$RT_p = \max RT_{pi} = max\left[\overline{l_i} + 3S_{li}\right] \qquad i = 1 \ldots m$$

Equation 21

where,

$$\overline{l_i} = \frac{1}{n}\sum_{j=1}^{n} l_{ij}$$

Equation 22

$$S_{li} = \sqrt{\frac{\sum_{j=1}^{n}(l_{ij} - \overline{l_i})^2}{n-1}}$$

Equation 23

$$l_{ij} = \sqrt{(x_{ij} - \overline{x_i})^2 + (y_{ij} - \overline{y_i})^2 + (z_{ij} - \overline{z_i})^2}$$

Equation 24

finally,

$$RT_a = \max\ 3\sqrt{\frac{\sum_{j=1}^{n}(a_{ij} - \overline{a_i})^2}{n-1}} \qquad i = 1 \ldots m$$

Equation 25

$$RT_b = \max\ 3\sqrt{\frac{\sum_{j=1}^{n}(b_{ij} - \overline{b_i})^2}{n-1}} \qquad i = 1 \ldots m$$

Equation 26

$$RT_c = \max\ 3\sqrt{\frac{\sum_{j=1}^{n}(c_{ij} - \overline{c_i})^2}{n-1}} \qquad i = 1 \ldots m$$

Equation 27

Path repeatability shall be measured using the same test procedure as that used for the measurement of path accuracy.

3.3.7.- Maximum range of each joint

The chosen test is to use the software provided by the manufacturer which has an option to calculate the distance in degrees that each axis is moving, the only problem is that we do not know the real accuracy of this measure.

4.- Design of the tests and setups

4.1.- Maximum speed of each joint

The first step was to build the modules for the test, which are two. The first iteration of the test was built on a prototype board to check if everything worked well before designing PCBs.

The first one is an MCU, in our case, the Teensy 4.1 which has been soldered and added terminal blocks for each pin to make easier connections.

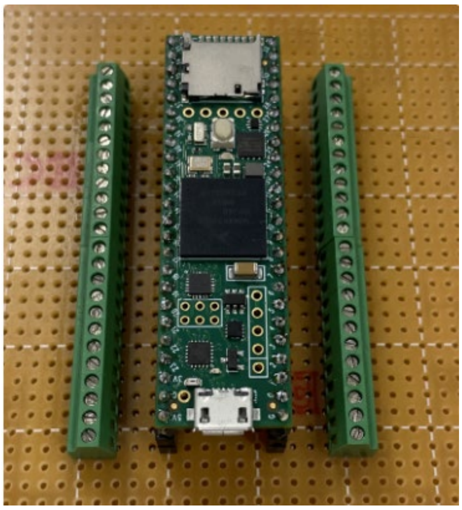


Figure 4.1: *Prototype of Teensy 4.1 with terminal blocks.*

The second module is the one that has the IR emitters and receivers. The emitter is the GL4800E0000F , a side view Infrared Emitting Diode. This diode is plastic moulded with resin lens. It has medium directivity angle and minimum radiant flux. It supports 950 nm typical peak emission wavelength. It is suitable for use in optoelectronic switching, office automation equipment, audiovisual equipment, home appliances, telecommunication equipment, measuring equipment, tooling machines and computers. The receiver is the TSOP34833 , a miniaturized receiver for infrared remote control systems. A PIN diode and a preamplifier are assembled on a lead frame, the epoxy package acts as an IR filter. This receiver needs a capacitor and a resistor to as protection against EOS (Electrical Overstress).

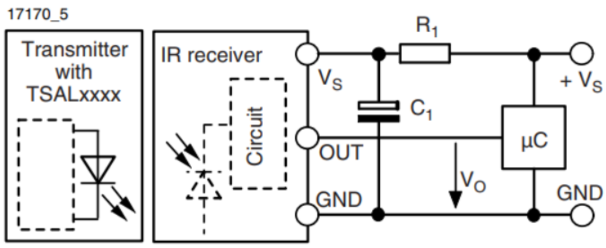


Figure 4.2: *IR receiver typical application.*

The whole module mounted is shown in the image below (*Figure 4.3*).

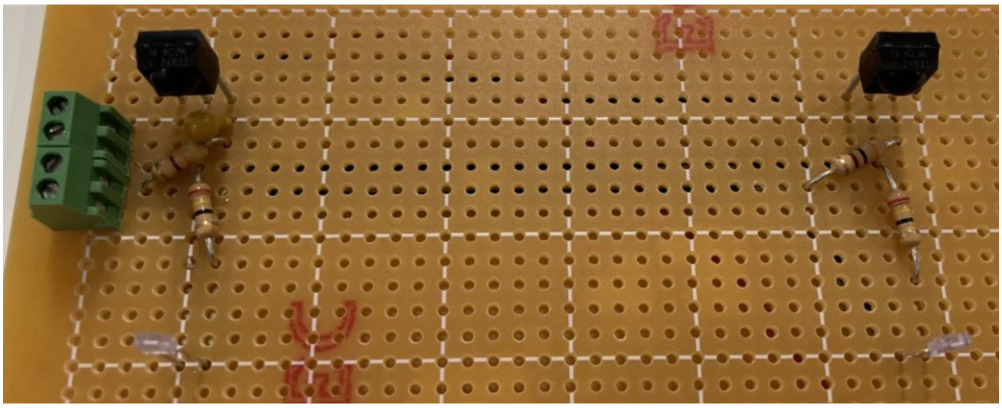


Figure 4.3: *Prototype of the IR sensor.*

The consumption of the IR emitter has been limited to 70 mA, with a 24 Ohm resistor, due to the maximum 250 mA the Teensy is able to supply. With this power supply, the distance between the sensor should be around the 3 cm to achieve a good performance.

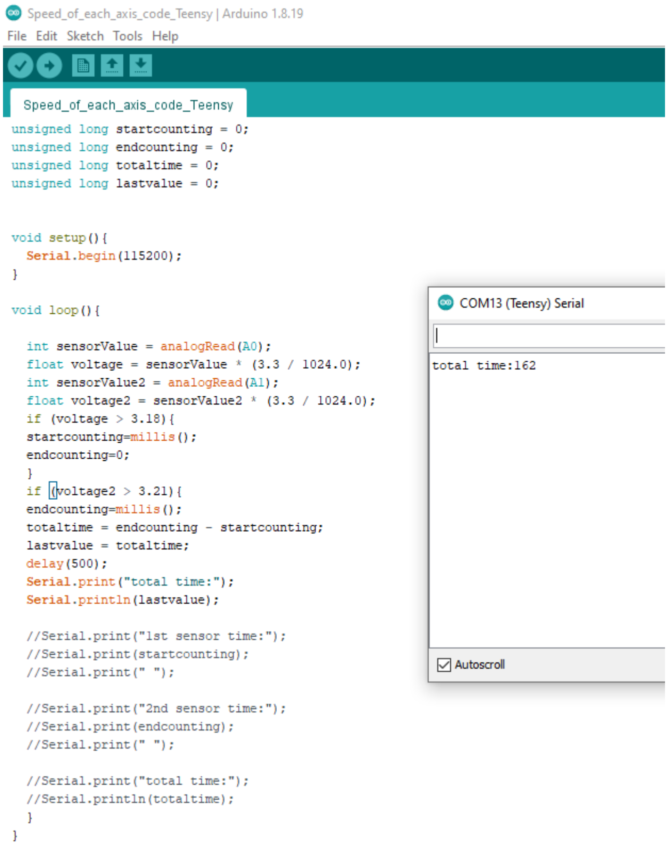


Figure 4.4: *Result of the first test with the prototype.*

The code worked perfectly, and it displayed the time in milliseconds between the activation of the first and second sensor.

The next step was to design a case for the sensors, similar to the industrial fork sensors.

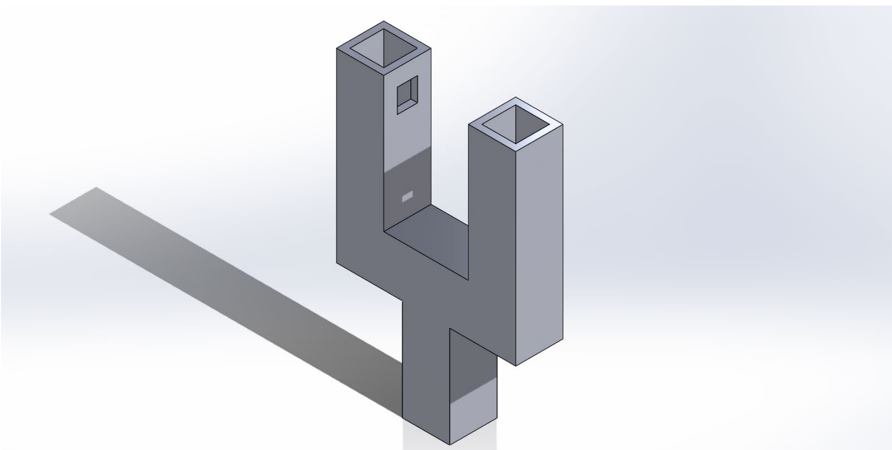


Figure 4.5: *Design of the fork barrier sensor.*

Once the 3D case was printed, the sensors were soldered, connected, and tested again.

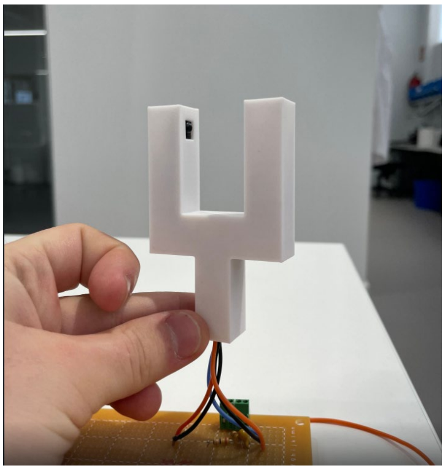


Figure 4.6: *Printed fork barrier sensor.*

As everything worked fine, it was mounted the setup and tested the sensors with the robot.

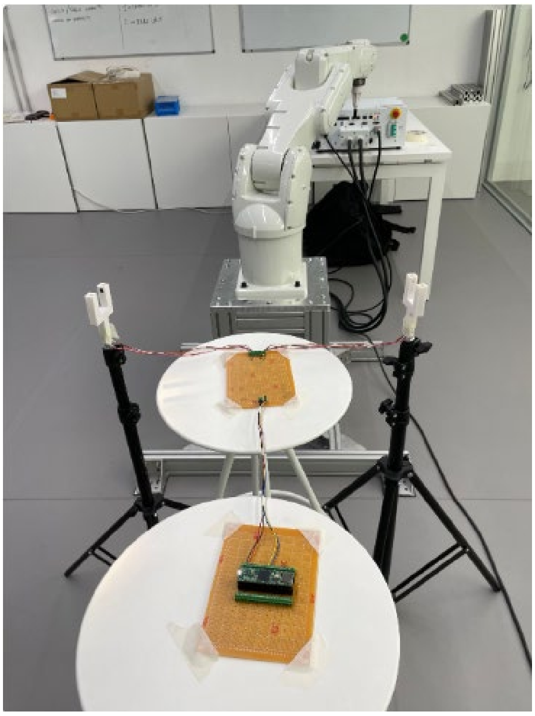


Figure 4.7: *First setup for the speed of each joint test.*

This time the performance was way worse, the first hypothesis was that maybe the lights were radiating IR light and interfering with the data. This was not right, and the solution was to increase the current on the emitter, seeking an improvement, this made the system work better but not perfectly.

Then, by testing the setup, it was discovered that when the robot was near the sensors, the motors created a lot of interferences, so it was decided to braid all the wires to make the interferences lower. This worked better, but sometimes these interferences affected the data.

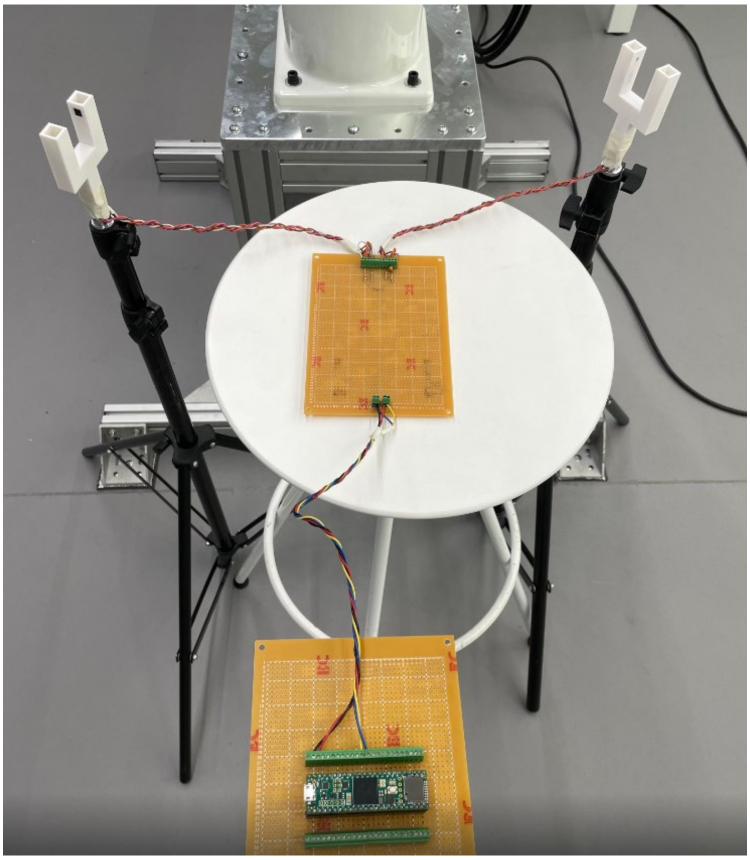


Figure 4.8: *Improved setup for the speed of each joint test.*

This problem is generated because when the robot is moving only one joint, all the others are ON and ready to move. For this reason, joints 5 and 6 create EMIs because are the ones passing through the sensor. Therefore, if the robot is running at 100% the EMIs are a lot bigger and affect the data more. At 25%, the data read by the sensors was quite precise and with no problems.

A professional fork barrier sensor maybe will be more precise but also more expensive, and it is not clear if it will be good enough to avoid the EMIs created by the two joints when going through the sensors. A possibility could be to use the STP software for the tests because it allows the user to connect the motors of each joint one by one, so the others will not generate EMIs. Even so, there are some cons, such as we would not be able to create a program and automatize the process.

As the previous test gave us so many problems with EMIs and the radian flux of the IR LED was not enough to make the phototransistor work well, it was decided to change the sensors and buy the slotted optical switch instead of building it.

4.1.1.- Improvement of the test

The idea was to buy a new sensor. The chosen one is the **OPB315WZ**, a slotted optical switch compound of an IR LED and a phototransistor. The sensing distance is 22.86 mm, what is perfect for the end effector that we already have.

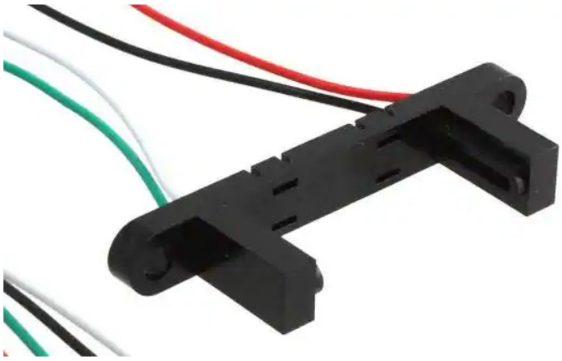


Figure 4.9: *New slotted optical sensor.*

The I_F indicated in the datasheet for the V_{CC} = 5V is 20 mA.

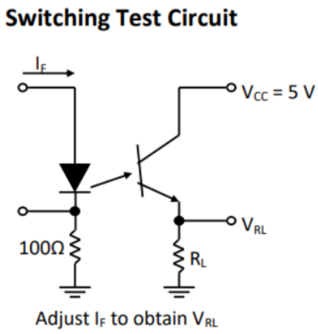


Figure 4.10: *Typical application of the slotted optical sensor.*

The phototransistor is the OP550, and we can see in the datasheet of the component that depending on the load resistance connected to it the rise and fall time changes. We should use a R_L lower than 2 kOhm to achieve a t_r and t_f lower than 50 us.

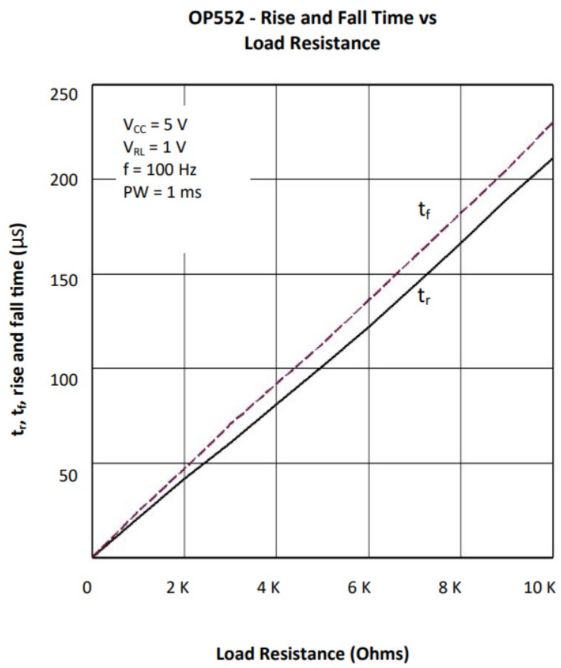


Figure 4.11: *Rise & Fall Time vs Load Resistance diagram.*

Then the idea is to take four sensors and place them on an aluminium profile with an adhesive measuring tape on it. The sensors must have mobility all along the aluminium profile to adjust them for the test of each joint because the distance will change. The sensors will have mobility along the angle to adjust them better for future robots. Like this we will be able to perform tests on different robots even if their range characteristics are different to the actual.

This setup will be placed on a tripod to have much more versatility when testing the different joints because some of them will need the sensors to be in a horizontal position and others in vertical.

The idea is to use a slotted aluminium profile that can be inserted through the main aluminium profile, and it has holes for screw nuts.

4.1.2.- Design of the sensor's setup

For both velocity tests, it will be needed to design and build a setup for the sensors. The idea is to place the sensors on a tripod to improve the mobility of them and be able to adjust the sensors for each one of the joints.

The first part is an aluminium profile of 1200 millimetres, in which will be placed an adhesive metric tape to measure the distance between the sensors accurately.

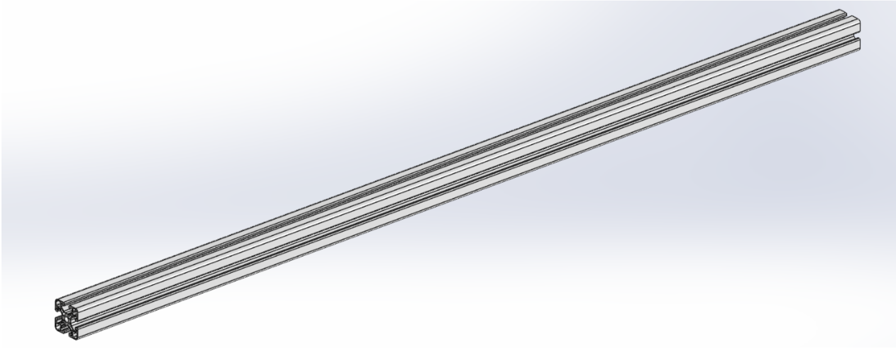


Figure 4.12: *1200 mm aluminium profile.*

This aluminium profile will be attached directly to the tripod with an aluminium groove profile of 280 mm. The holes are M8 with 25 mm between them.

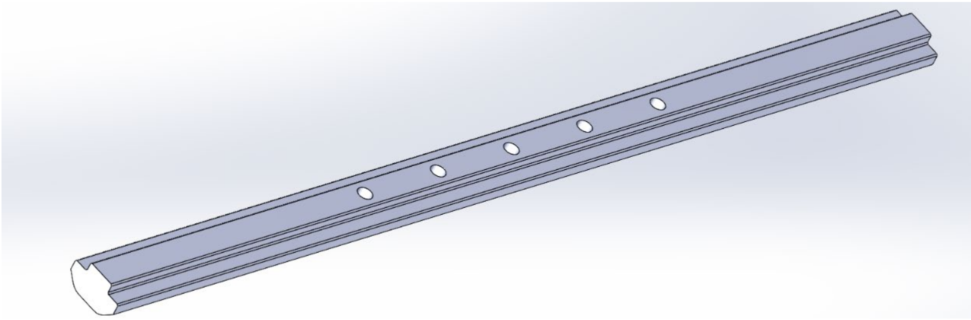


Figure 4.13: *Aluminium groove profile.*

The sensors will be placed on a bracket of 80x40x20 millimetres and attached to the main aluminium profile.

Figure 4.14: *80x40x20 mm bracket.*

The other sensors will be placed on a 250 millimeters aluminium profile for higher versatility.

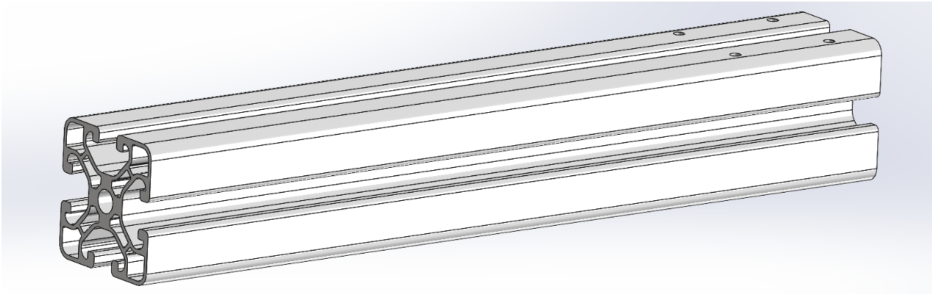


Figure 4.15: *250 mm aluminium profile.*

The smaller aluminium profile will be attached to the bigger one with a joining plate.

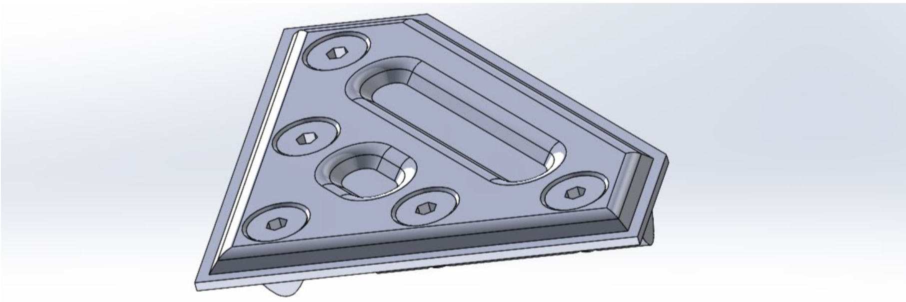


Figure 4.16: *Joining plate.*

The sensor is an optical-slotted sensor which will detect the end effector passing through to be able to measure the time that it takes to get from the first sensor to the last one.

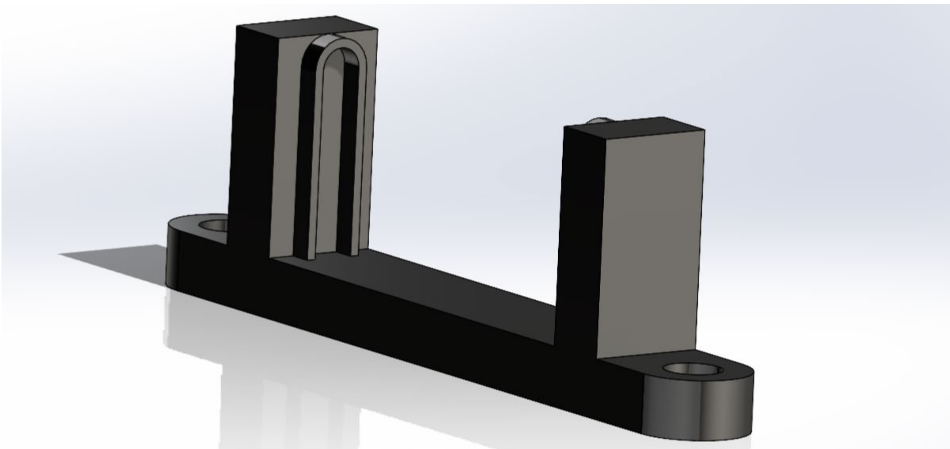


Figure 4.17: *Slotted optical sensor.*

The final assembly is shown in the image below (*Figure 4.18*).

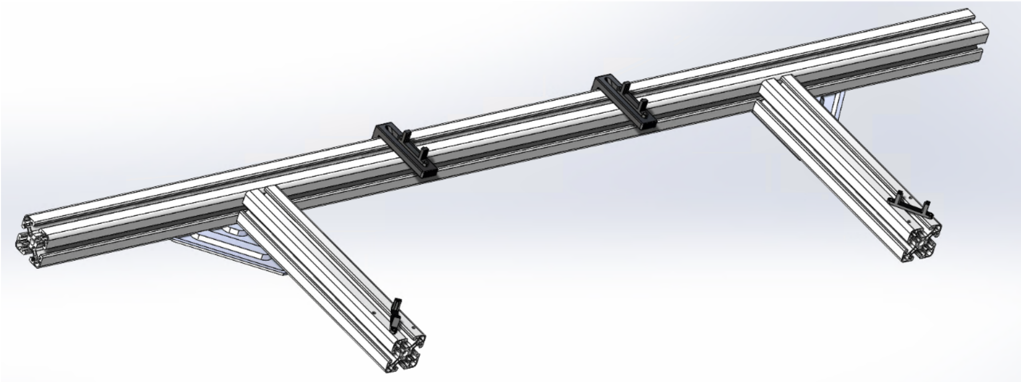


Figure 4.18: *Assembly of the setup for the speed tests.*

The final design was built, and the result is shown in the image below (*Figure 4.19*).

Figure 4.19: *Final mounted setup for the speed tests.*

4.2.- Maximum speed of the robot

For this test was used the same setup built for the previous one, like this the budget will be lowered, and the time spent reduced.

The code used to calculate the time is the same on both of the tests too.

Speed_of_each_axis_code_Teensy.ino

```
unsigned long startcounting = 0;
unsigned long endcounting = 0;
unsigned long totaltime = 0;
unsigned long lastvalue = 0;

  void setup(){
  Serial.begin(115200);
}

void loop(){

  int sensorValue = analogRead(A0);
  float voltage = sensorValue * (3.3 / 1024.0);
  int sensorValue2 = analogRead(A1);
  float voltage2 = sensorValue2 * (3.3 / 1024.0);
  if (voltage < 0.1){
  startcounting=millis();
  endcounting=0;
  }
  if (voltage2 < 0.1){
  endcounting=millis();
  totaltime = endcounting - startcounting;
  lastvalue = totaltime;
  delay(100);

  Serial.print("1st sensor time:");
  Serial.print(startcounting);
  Serial.print(" ");

  Serial.print("2nd sensor time:");
  Serial.print(endcounting);
  Serial.print(" ");

  Serial.print("total time:");
  Serial.println(totaltime);
  }
}
```

Figure 4.20: *Code for the time measurement.*

This is a very simple code, in which it is described that when the first sensor detects the end effector pass through, the timer starts counting until the second sensor detects the end effector too. Then the data is displayed in milliseconds. In the terminal it can be seen the time when the first sensor is activated, the time when the second sensor is activated and the time between the two events.

4.3.- Maximum range

The maximum range test needs no setup, the software of the robot allows measuring the range it moves in each of the axis.

4.4.- Design of the end effectors

For each of the tests we will need to put different weights on the end effector of the robot to be able to measure how this could affect the performance. We know that the maximum load the manufacturer says the robot can handle is 6 kg, but it would be interesting to use different loads between 0 and 6 kg to create diagrams of the loss of performance and even to overweight the robot and see how it responds.

The proposed end effector is composed of three different parts.

1st part:

The first part is the one that will be attached to the robotic arm or to the force torque sensor. It has six M5 holes in the same diameter as the force torque sensor, so if it's necessary to use it, we could attach both parts. It also has a cylinder of 30 mm in diameter, so we can put weights on it and on the end of the cylinder there is a thread, so we can attach another part and not allow the weights to fall.

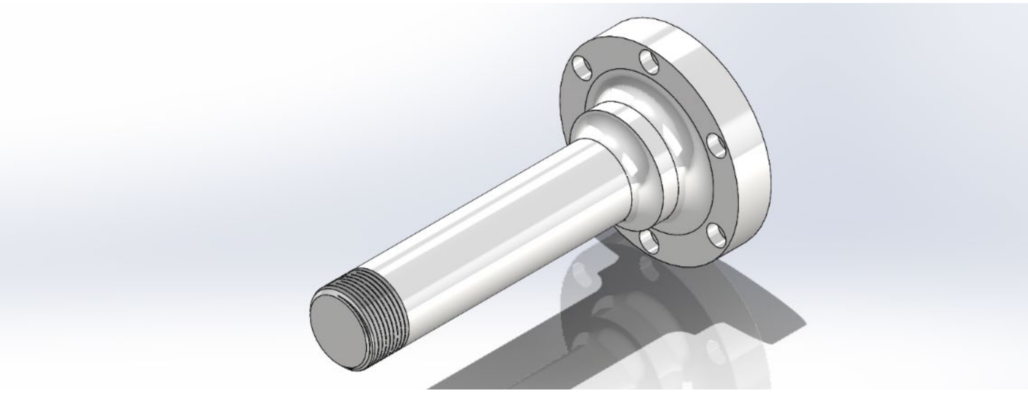


Figure 4.21: *Main end effector design.*

2nd part:

The second part is the "nut" of the end effector, which will not allow the weights to fall. It has the same diameter as the 1st part and the force torque sensor and six M5 holes.

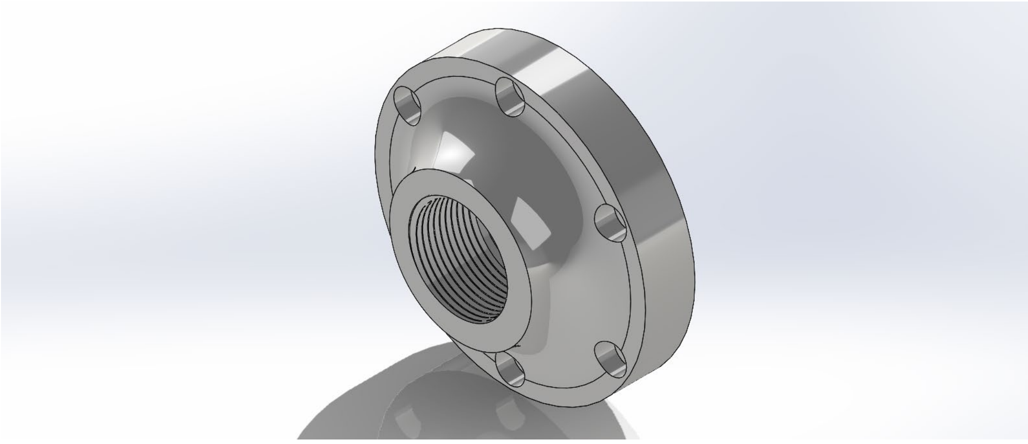


Figure 4.22: End effector 'nut' design.

3rd part:

The 3rd part is simply an adapter, so we can attach the end effector to the robotic arm, it has six M5 holes with the same diameter as the force torque sensor and the other parts of the end effector and six more M5 holes with the same diameter as the robotic arm.

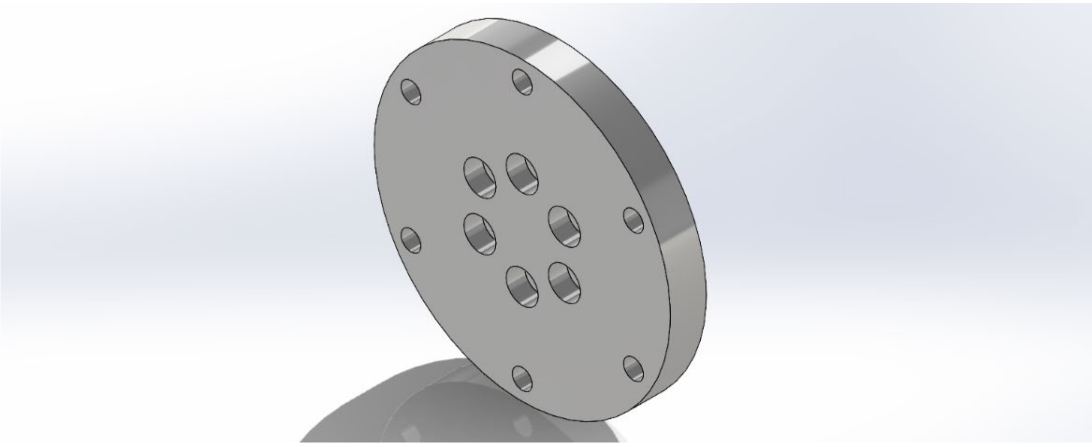


Figure 4.23: End effector adapter design.

Extra part:

The other part is the ATI force torque sensor which would be attached to the end effector and to the adaptor, so we can connect it to the robotic arm.

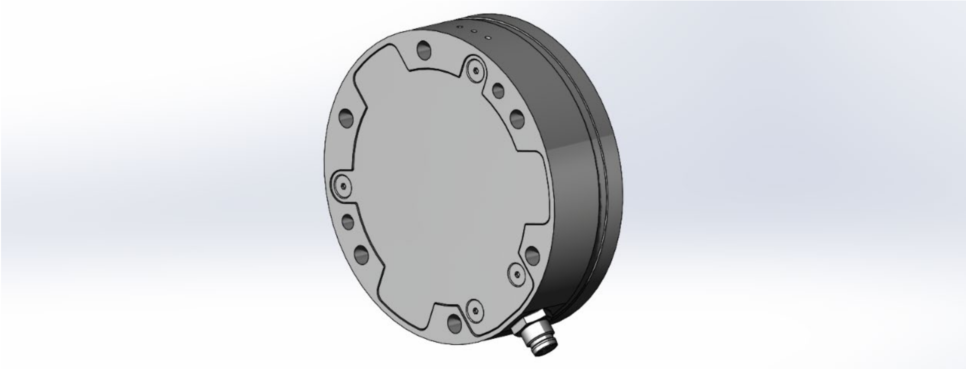


Figure 4.24: *ATI Axia80 design.*

Assembly:

The final assembly would look like the next image.

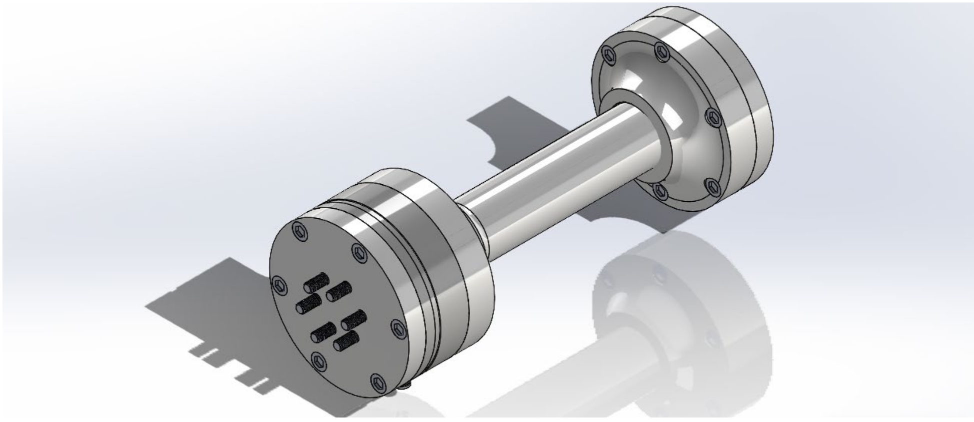


Figure 4.25: *End effector assembly.*

The idea is to put weights covered with rubber to prevent any kind of damage to the end effector.

Figure 4.26: *Rubber weights.*

The width of the different discs are the next ones. The cylinder of the end effector is 105 mm, so we can combine a big number of weights.

As the length of the cylinder is 105 mm, we could use:

- Six 0.5 kg discs
- Four 1.25 kg discs
- Four 2.5 kg discs
- Two 3 kg discs

The idea is to put some protection when not using all the weights to not allow the discs to move along the end effector.

Figure 4.27: *Protective closure for the weights.*

The final design would look like the next assembly with the four different weights we should buy.

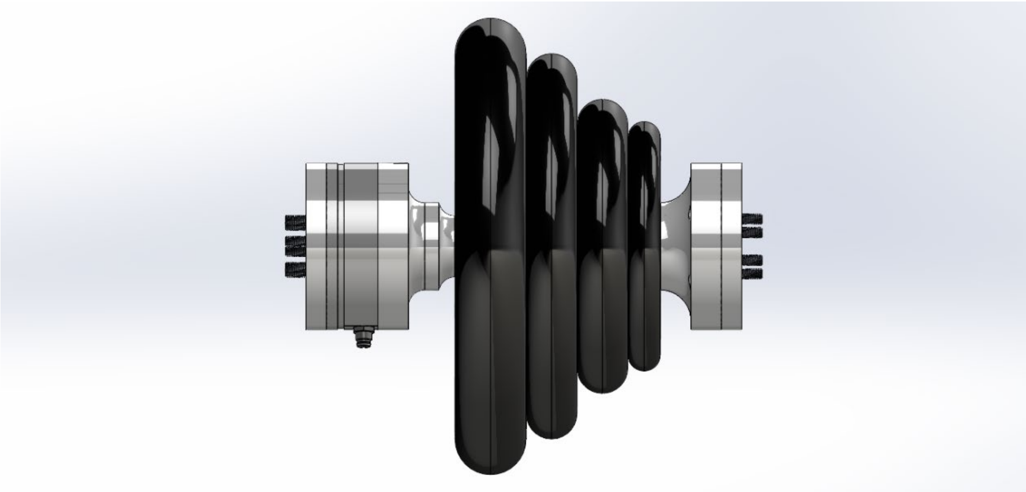


Figure 4.28: *End effector assembly with weights.*

We could even attach a personalized end effector to the weights.

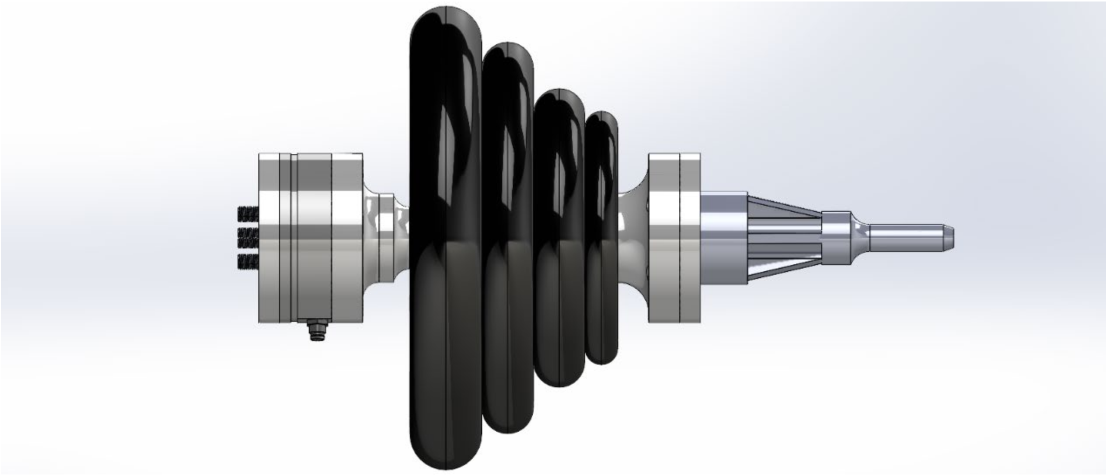


Figure 4.29: *End effector assembly with weights and personalized end effector.*

Finally, attached to the robotic arm will look like the image below (*Figure 4.30*).

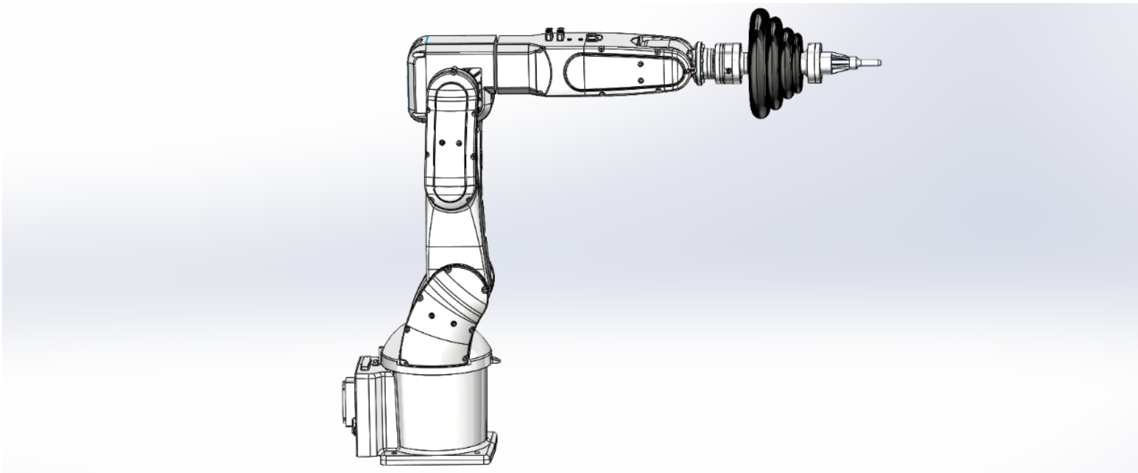


Figure 4.30: *End effector assembly on the robot.*

Two more end effectors were designed and printed in 3D to perform some tests faster.

1st end effector:

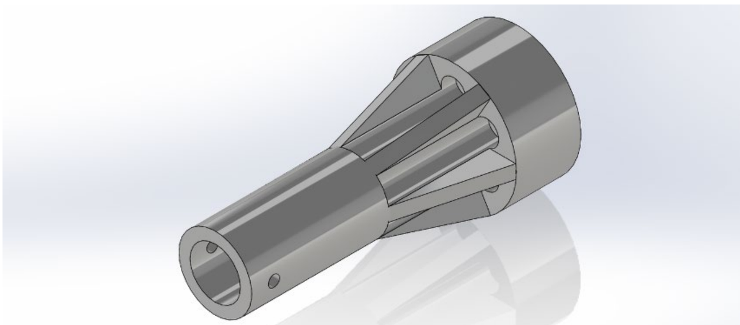


Figure 4.31: *End effector for markers.*

The idea for the first one is to attach a marker pen to the robot to write on a whiteboard and check the accuracy of the movements of the robot and other characteristics.

Real image:

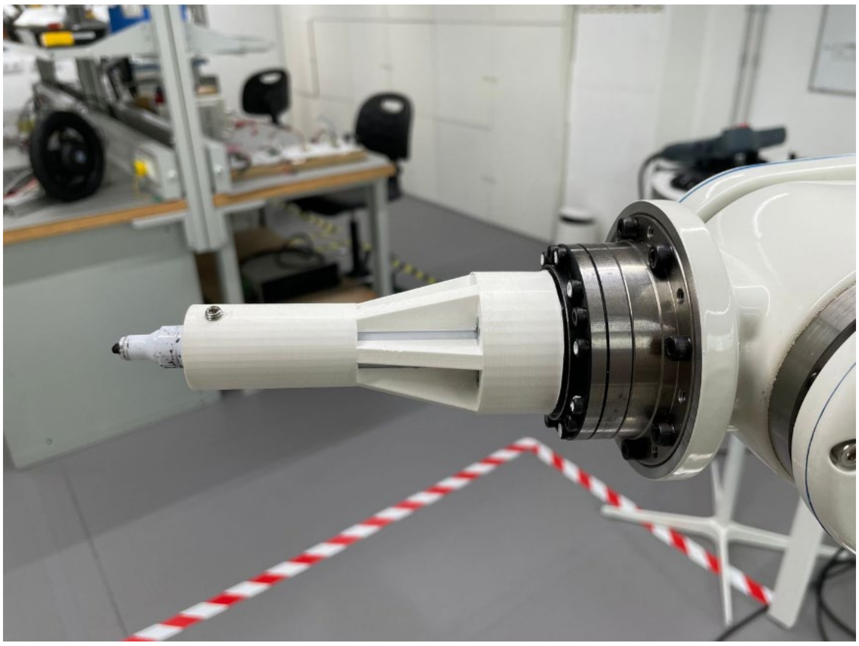


Figure 4.32: *Printed end effector for markers.*

This end effector fulfilled its function perfectly, and allowed to check how the movements were done.

2nd end effector:

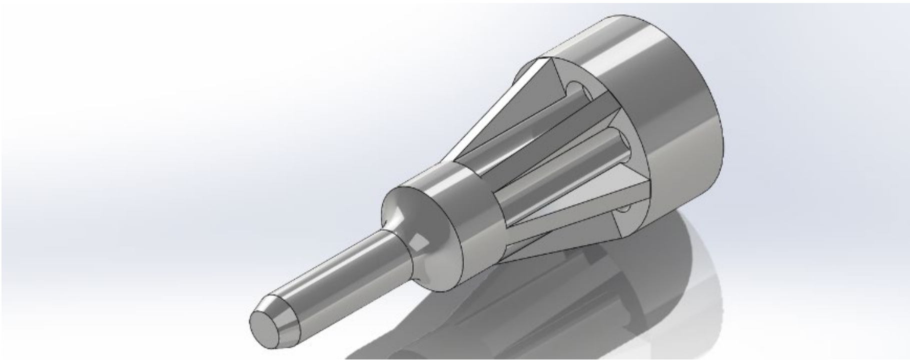


Figure 4.33: *End effector for speed tests.*

The idea of the second end effector was to perform all the speed tests without applied load. For this reason, the cylinder has much less diameter than the other end effector, and it is also pointier to perform other tests.

Real image:

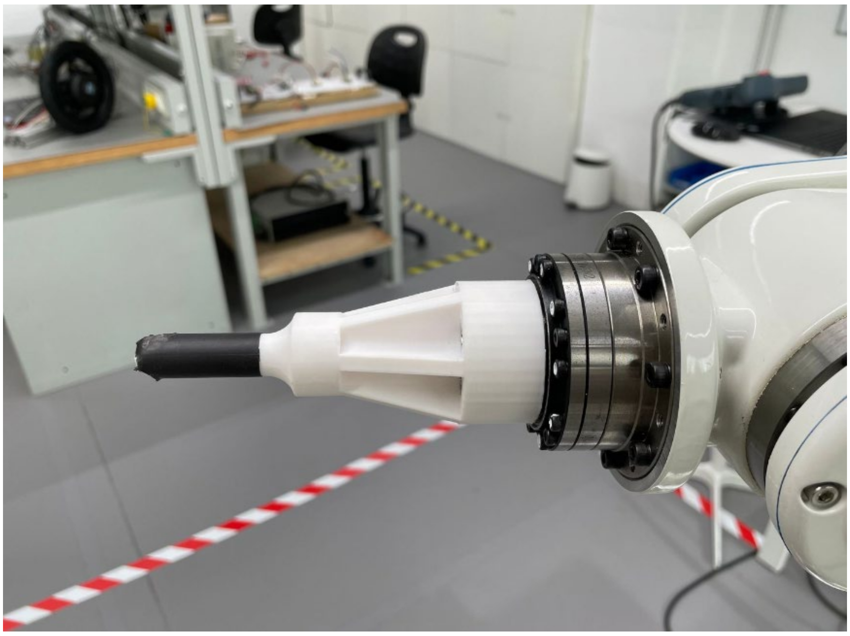


Figure 4.34: *Printed end effector for speed tests.*

This end effector fulfilled its function well too, even though, it had some problems because the first prototype of sensors used in the speed test were not able to detect the end effector passing through at high speed, for this reason it was used a black tape to improve the detection, this worked, and the end effector was useful for the speed tests.

Final end effector:

The first prototype of the final end effector was printed in 3D to check that all the dimensions were correct, before mechanizing it in aluminium.

The main dimensions such as the length of the end effector, the diameter of the parts attached to the robotic arm are correct. The holes for the screws are good, but maybe it should be a bit bigger because of the tolerances, some of the holes were a bit smaller, and we had to apply some pressure to make the screws and nuts fit in.

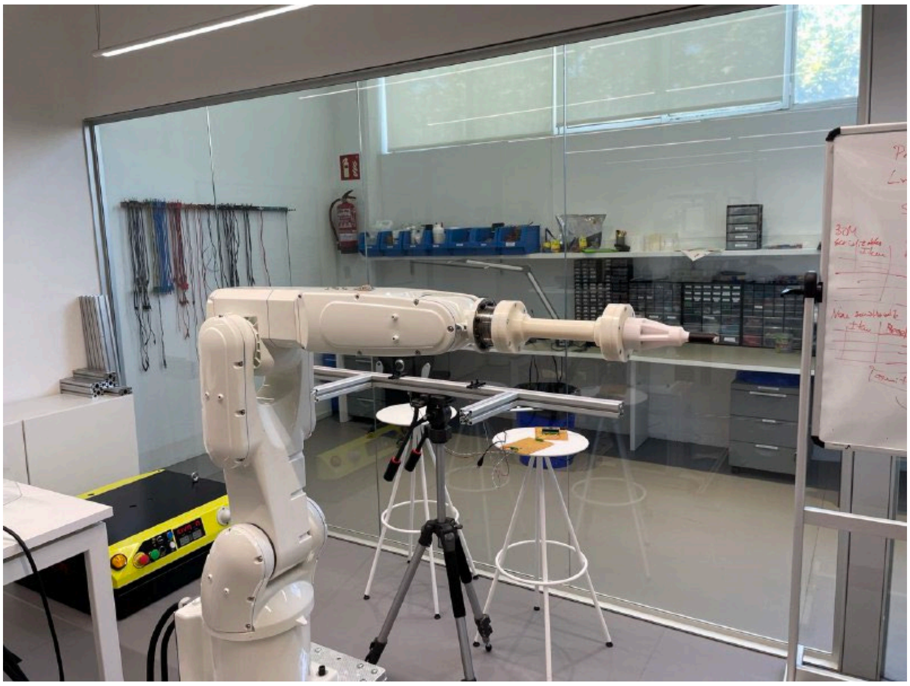


Figure 4.35: *Printed final end effector.*

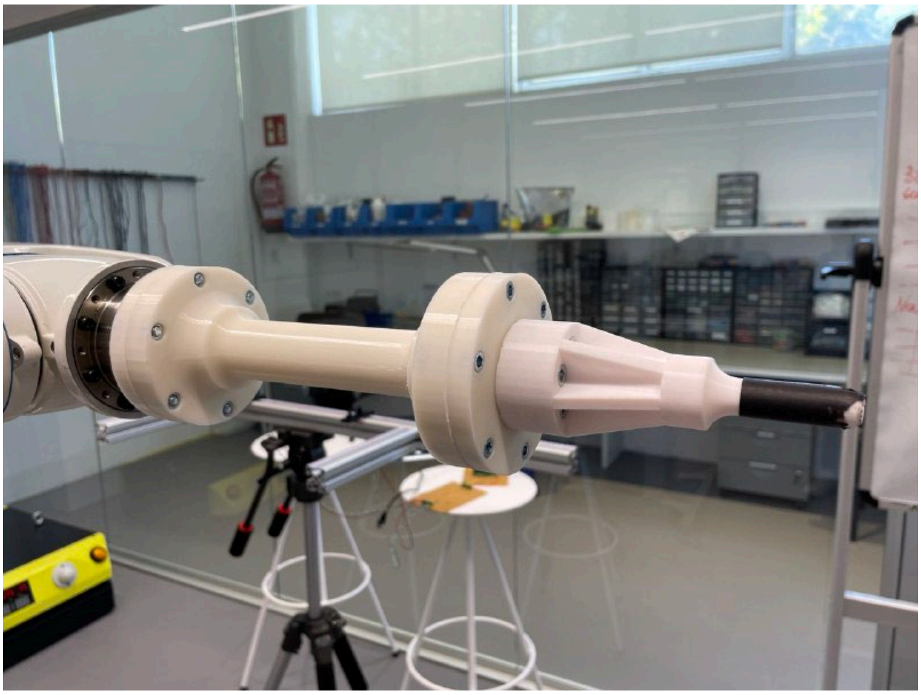


Figure 4.36: *Detailed view of the printed final end effector.*

The weight of the whole parts of the end effector printed in PLA is 452 grams. The structure feels quite solid, even if it is printed at 80% of infill. The biggest problem with the design is the thread because it does not fit on the end effector 'nut'. The problem could be an error in the design, which seems difficult because it was used a tool from the SolidWorks software which generates the same thread on both, an alternative could be that the 3D printer is not accurate enough and due to the tolerances, the parts do not fit. Either way, the design will be checked and improved.

4.5.- Design of the PCB

At the moment, the conditioning electronics of the sensors are soldered on a prototype board and the MCU is soldered on another prototype board, the two boards are connected using cables. This setup worked well for the moment, but is difficult to work with these two big boards. The proposed idea is to design a simple PCB where we could connect the Teensy to female pin headers, put the resistors and capacitors on the PCB and connect the sensors using terminal blocks to ensure the good connection of the cables.

The schematic is very simple, it consists of two terminal blocks (one for each pair of sensors) of 1x8 pins and 2.54 mm separation between each pin and two sets of female pin headers of 1x24 for the connection of the Teensy. It also has the 8 resistors needed and two operational amplifiers in voltage follower mode to act as an output buffer.

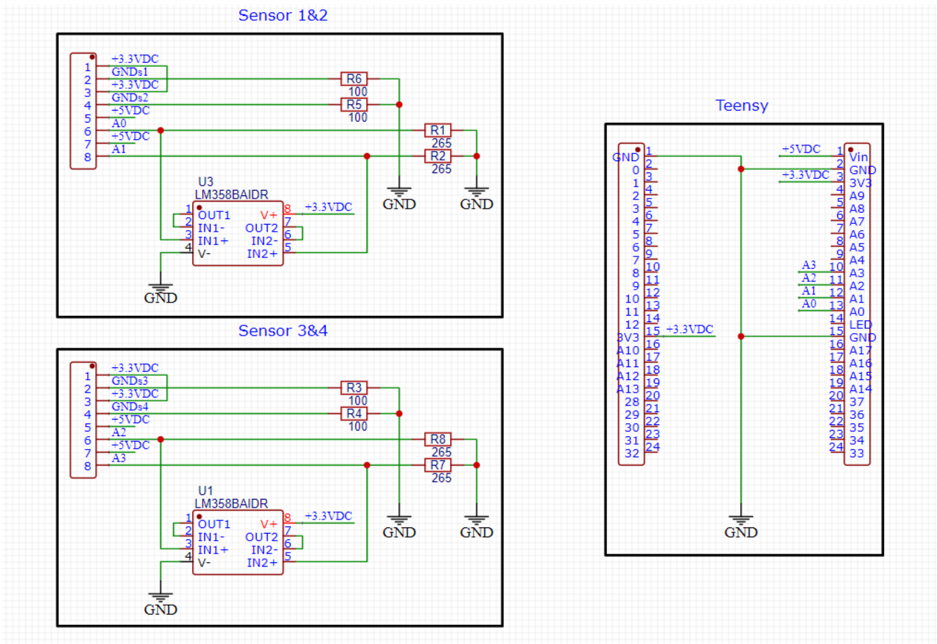


Figure 4.37: *Schematic of the PCB.*

The PCB is very simple too, the intention is to make it as small as possible to be able to place it on the tripod with the sensors, the dimensions of the PCB are 33 mm of length and 71.1 mm of height. It has four holes to make it easier to place it on the tripod.

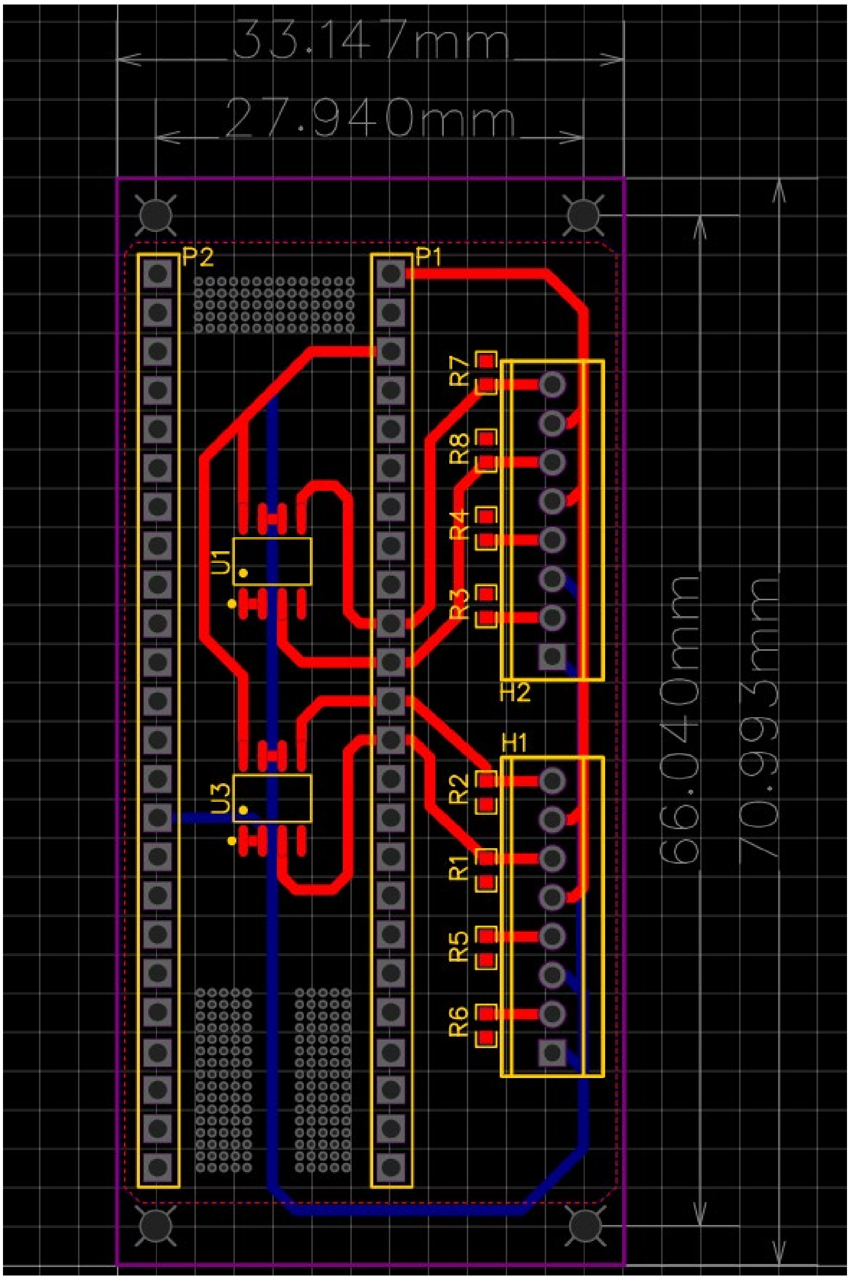


Figure 4.38: *Layout of the PCB.*

5.- Experimental results

5.1.- Conditions for the tests

The ISO 9283 states which are the recommended conditions to develop each of the tests, and this project follows those recommendations. **[17]**

Environmental conditions.

Temperature → The ambient temperature of the testing environment should be 20ºC. Other ambient temperatures shall be stated and explained in the test report. The testing temperature shall be maintained at 20ºC ± 2ºC. The robot and the measuring instruments should have been in the test environment long enough (preferably overnight) so that they are in a thermally stable condition before testing. They shall be protected from draughts and external thermal radiation (e.g., sunlight, heaters).

Instrumentation.

For path characteristics, overshoot and pose stabilization measurements, the dynamic characteristics of the data acquisition equipment (e.g., sampling rate) shall be high enough to ensure that an adequate representation of the characteristics being measured is obtained. The total uncertainty of measurement shall not exceed 25 % of the magnitude of the characteristic under test.

Load to the mechanical interface.

All tests shall be executed with a test load equal to 100 % of rated load conditions, i.e., mass, position of centre of gravity, moments of inertia, according to the manufacturer's specification. The rated load conditions shall be specified in the test report.

To characterize robots with load dependent performances, additional optional tests can be made with the mass of rated load reduced to 10 % as indicated in the table below (*Figure 5.1*) or some other value as specified by the manufacturer.

Characteristics to be tested	Load to be used	
	100 % of rated load (X = mandatory)	The mass of rated load reduced to 10 % (O = optional)
Pose accuracy and pose repeatability	X	O
Multi-directional pose accuracy variation	X	O
Distance accuracy and distance repeatability	X	—
Position stabilization time	X	O
Position overshoot	X	O
Drift of pose characteristics	X	—
Exchangeability	X	O
Path accuracy and path repeatability	X	O
Path accuracy on reorientation	X	O
Cornering deviations	X	—
Path velocity characteristics	X	O
Minimum posing time	X	O
Static compliance	—	See clause 10
Weaving deviations	X	O

Figure 5.1: *Loads to be used for each test.*

Test velocities.

All pose characteristics shall be tested at the maximum velocity achievable between the specified poses, i.e., with the velocity override set to 100 %, in each case. Additional tests could be carried out at 50 % and/or 10 % of this velocity.

Characteristics to be tested	Velocity	
	100 % of rated velocity (X = mandatory)	50 % or 10 % of rated velocity (O = optional)
Pose accuracy and pose repeatability	X	O
Multi-directional pose accuracy variation	X	O
Distance accuracy and repeatability	X	O
Position stabilization time	X	O
Position overshoot	X	O
Drift of pose characteristics	X	—
Exchangeability	X	O
Minimum posing time	See clause 9 and table 20	

Figure 5.2: *Velocities for each test.*

For path characteristics, the tests shall be conducted at 100 %, 50 %, and IO % of rated path velocity as specified by the manufacturer for each of the characteristics tested (see table 3). Rated path velocity shall be specified in the test report. The velocity specified for each test depends on the shape and size of the path. The robot shall be able to achieve this velocity over at least 50 % of the length of the test path. The related performance criteria shall be valid during this time.

Characteristics to be tested	Velocity		
	100 % of rated path velocity (X = mandatory)	50 % of rated path velocity (X = mandatory)	10 % of rated path velocity (X = mandatory)
Path accuracy and path repeatability	X	X	X
Path accuracy on reorientation	X	X	X
Cornering deviations	X	X	X
Path velocity characteristics	X	X	X
Weaving deviations	X	X	X

Figure 5.3: *Velocities for each test 2.*

Poses to be tested.

Five measurement points are located on the diagonals of the measuring plane and correspond to (P1 to P5) in the selected plane transformed by the axial (X_{MP}) and radial (Z_{MP}) measurement point offset. The points P1 to P5 are the positions for the wrist reference point of the robot.

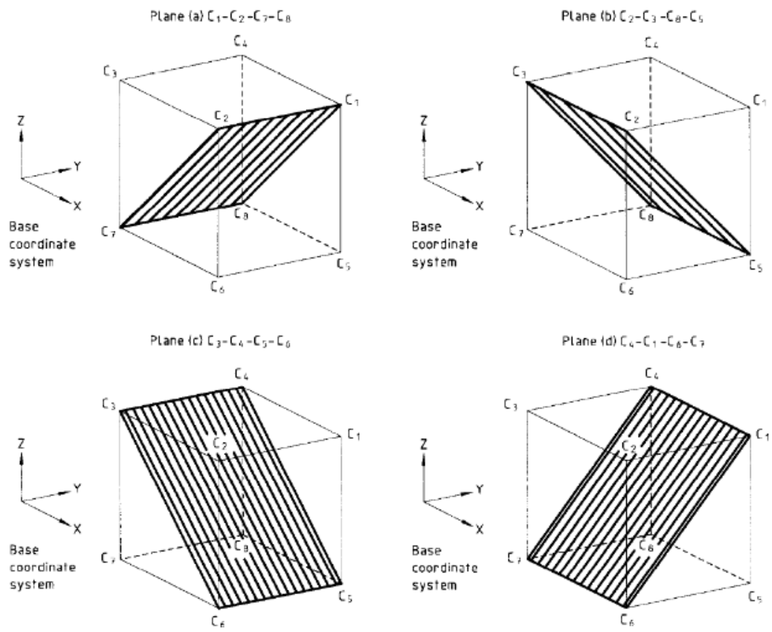


Figure 5.4: *Poses to be tested.*

P1 is the intersection of the diagonals and is the centre of the cube. The points P2 to P5 are located at a distance from the ends of the diagonals equal to (10 ± 2) % of the length of the diagonal. If this is not possible, then the nearest point chosen on the diagonal shall be reported.

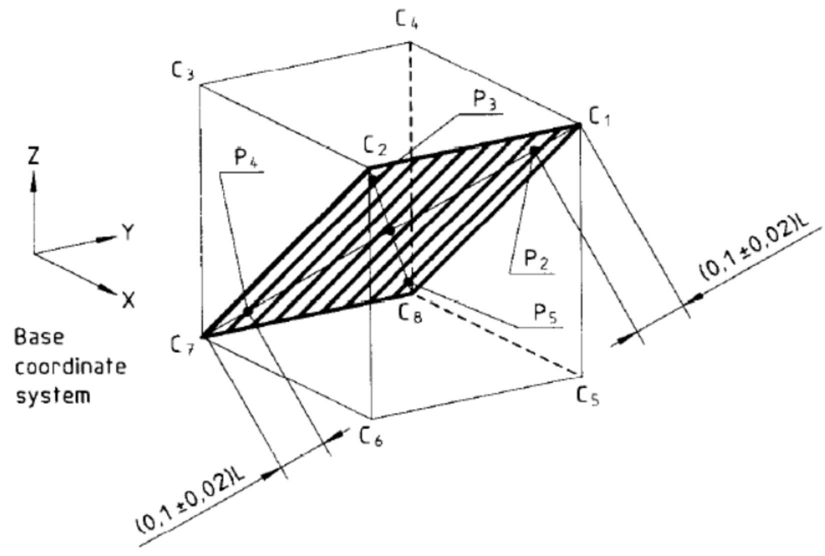


Figure 5.5: *ISO cube with diagonal plane.*

Number of cycles

Each test has its own number of cycles to make sure that the data collected is the most accurate possible.

Characteristic to be tested	Number of cycles
Pose accuracy and pose repeatability	30
Multi-directional pose accuracy variation	30
Distance accuracy and distance repeatability	30
Position stabilization time	3
Position overshoot	3
Drift of pose characteristics	Continuous cycling during 8 hours
Exchangeability	30
Path accuracy and path repeatability	10
Path accuracy on reorientation	10
Cornering deviations	3
Path velocity characteristics	10
Minimum posing time	3
Weaving deviations	3

Figure 5.6: *Number of cycles for each test.*

5.2.- Static accuracy and repeatability

Due to the lack of time, this parameter will not be studied in this project. Even so, in a future it will.

Table 5.1: *Static accuracy and repeatability.*

Load	Velocity	P1 (acc)	P1 (rep)	P2 (acc)	P2 (rep)	P3 (acc)	P3 (rep)	P4 (acc)	P4 (rep)	P5 (acc)	P5 (rep)
0%	10%	AP_p=	RP_l=	AP_p=	RP_l=	AP_p=	RP_l=	AP_p=	RP_l=	AP_p=	RP_l=
10%	50%	AP_a=	RP_a=	AP_a=	RP_a=	AP_a=	RP_a=	AP_a=	RP_a=	AP_a=	RP_a=
100%	100%	AP_b=	RP_b=	AP_b=	RP_b=	AP_b=	RP_b=	AP_b=	RP_b=	AP_b=	RP_b=
		AP_c=	RP_c=	AP_c=	RP_c=	AP_c=	RP_c=	AP_c=	RP_c=	AP_c=	RP_c=

5.3.- Maximum speed of each joint

Table 5.2: *Maximum speed of each joint.*

Load	Joint	Datasheet values	Covered angle	T1	T2	…	Tn	Average time	Experimental values
0%	1	295	D=	T_1=	T_2=	…	T_n=	T_{AV}=	V=
10%	2	245							
100%	3	295							
	4	365							
	5	295							
	6	370							

The error presented in the tables below (from *Table 5.3* to *Table 5.8*) is calculated using the next expression (*Equation 28*).

$$\frac{Experimental\ value - Theoretical\ value}{Theoretical\ value} * 100$$

Equation 28

5.3.1.- First test

Joint 1:

Table 5.3: *Joint 1 results.*

Load	Joint	Covered angle (º)	Average time (s)	Datasheet values	Experimental values (º/s)	Error (%)
0%	1	D_1=23.110	T_{AV}=0.1029	295	V_1=224.59	E_1=31.35
		D_2=17.499	T_{AV}=0.0757		V_2=231.16	E_2=27.62
		D_3=11.554	T_{AV}=0.0480		V_3=241.04	E_3=22.38
		D_4=5.799	T_{AV}=0.0206		V_4=281.50	E_4=4.79
		D_5=2.743	T_{AV}=0.0093		V_5=294.95	E_5=0.46

Joint 2:

Table 5.4: *Joint 2 results.*

Load	Joint	Covered angle (º)	Average time (s)	Datasheet values	Experimental values (º/s)	Error (%)
0%	2	D_1=12.141	T_{AV}=0.0630	245	V_1=192.71	E_1=27.13
		D_2=7.369	T_{AV}=0.0365		V_2=201.71	E_2=21.46
		D_3=5.716	T_{AV}=0.0266		V_3=214.89	E_3=14.01
		D_4=4	T_{AV}=0.0176		V_4=227.27	E_4=7.80
		D_5=2.531	T_{AV}=0.0120		V_5=210.92	E_5=16.16

Joint 3:

Table 5.5: *Joint 3 results.*

Load	Joint	Covered angle (º)	Average time (s)	Datasheet values	Experimental values (º/s)	Error (%)
0%	3	D_1=19.168	T_{AV}=0.083	295	V_1=230.94	E_1=27.74
		D_2=11.496	T_{AV}=0.048		V_2=239.5	E_2=23.17
		D_3=9.218	T_{AV}=0.036		V_3=256.05	E_3=15.21
		D_4=6.497	T_{AV}=0.024		V_4=270.71	E_4=8.97
		D_5=3.911	T_{AV}=0.013		V_5=293.33	E_5=0.57

Joint 4:

Table 5.6: *Joint 4 results.*

Load	Joint	Covered angle (º)	Average time (s)	Datasheet values	Experimental values (º/s)	Error (%)
0%	4	D_1=29.099	T_{AV}=0.104	365	V_1=279.79	E_1=30.45
		D_2=24.731	T_{AV}=0.083		V_2=297.96	E_2=22.50
		D_3=19.274	T_{AV}=0.060		V_3=321.23	E_3=13.62
		D_4=14.088	T_{AV}=0.041		V_4=341.11	E_4=7.00
		D_5=8.119	T_{AV}=0.022		V_5=364.08	E_5=0.25

Joint 5:

Table 5.7: Joint 5 results.

Load	Joint	Covered angle (º)	Average time (s)	Datasheet values	Experimental values (º/s)	Error (%)
0%	5	D_1=27.146	T_{AV}=0.113	295	V_1=240.23	E_1=22.80
		D_2=22.027	T_{AV}=0.088		V_2=250.31	E_2=17.86
		D_3=16.898	T_{AV}=0.062		V_3=272.55	E_3=8.24
		D_4=13.617	T_{AV}=0.049		V_4=277.89	E_4=6.15
		D_5=8.066	T_{AV}=0.0275		V_5=292.59	E_5=0.82

Joint 6:

Table 5.8: Joint 6 results.

Load	Joint	Covered angle (º)	Average time (s)	Datasheet values	Experimental values (º/s)	Error (%)
0%	6	D_1=61.22	T_{AV}=0.221	370	V_1=276.97	E_1=33.59
		D_2=49.414	T_{AV}=0.174		V_2=283.98	E_2=30.29
		D_3=38.997	T_{AV}=0.135		V_3=288.87	E_3=28.09
		D_4=27.942	T_{AV}=0.089		V_4=313.95	E_4=17.85
		D_5=16.174	T_{AV}=0.046		V_5=351.61	E_5=5.23

The first test was developed without warming up the robotic arm in order to observe if there will be any difference when warming it up.

The results are not that bad, some values are really accurate and have less than a 1% error, mostly when the distance between the sensors is lower. That could be because when the distance is higher, maybe the robot is still accelerating or already decelerating once it has activated the sensors. For this reason, for the next tests it would be interesting to observe the commanded and feedback velocities using the STP software and calculate is when the robot stops accelerating or starts decelerating.

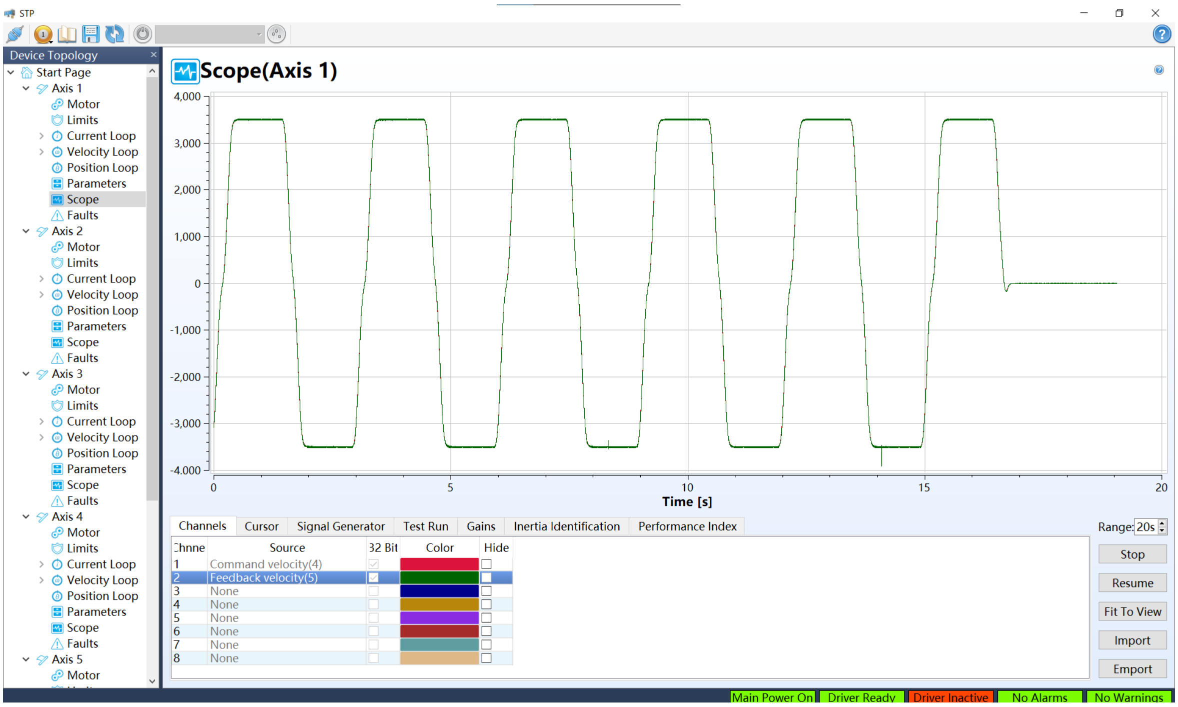


Figure 5.7: *Commanded and feedback velocity plot.*

It was plotted the commanded and feedback speed of each one of the joints to check how long it takes the robot to reach the constant velocity zone. It can be seen that the rise and fall time is quite quick, it takes very few degrees for the robot to reach the constant maximum velocity. This rejects the hypothesis, because the sensors are placed always in the inside the constant zone.

Finally, another error could in the perception time of the sensors, maybe the MCU is not fast enough to process the data when the sensors are too close and the time measurement loses accuracy.

5.4.- Maximum speed of the robot

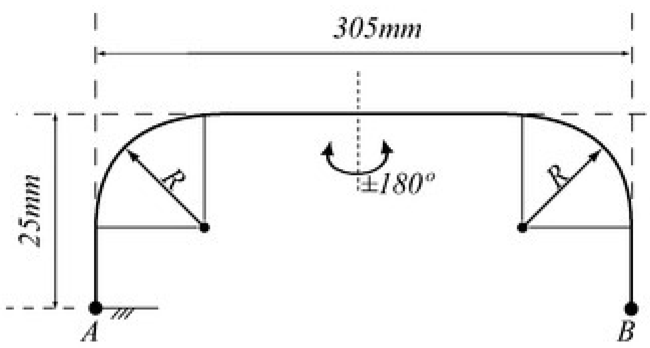
Table 5.9: *Maximum speed of the robot.*

Load	Selected shape and length of the path	T1	T2	...	Tn	Average time	Cycles per minute
0% 10% 100%		T_1=	T_2=	...	T_n=	T_{AV}=	cpm=

5.4.1.- First test

No warm-up:

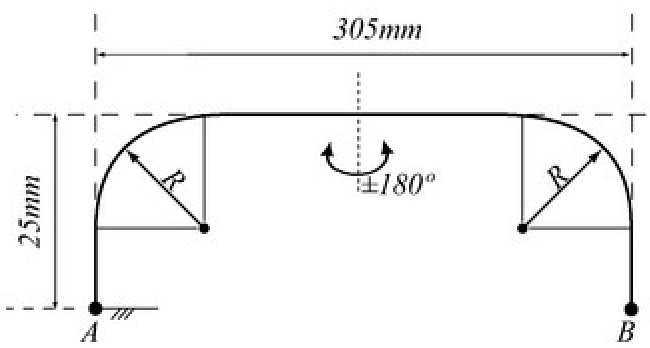
Table 5.10: *No warm-up results.*

Load	Selected shape and length of the path	Average time	Cycles per minute
0% 10% 100%		T_{AV}=0.558	cpm= 108

The performance of the robot is tested without load because the actual end effector cannot support load, the cycles per minute are 108, much lower than any other industrial robotic arm.

30 min warm-up:

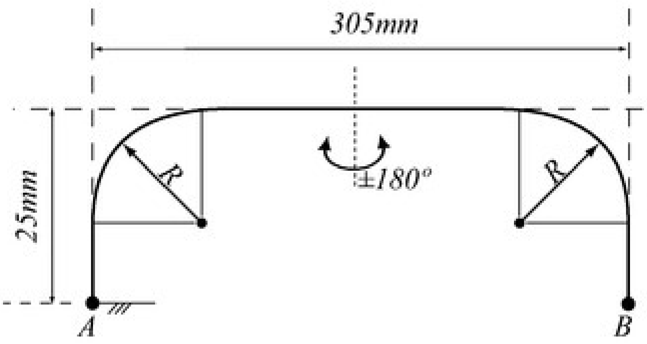
Table 5.11: *30 min warm-up results.*

Load	Selected shape and length of the path	Average time	Cycles per minute
0% 10% 100%		T_{AV}=0.563	cpm= 107

With a 30 min warm-up, the performance is nearly the same, only one cycle per minute slower. This could be due to an error in the time measured by the sensors.

1 hour warm-up:

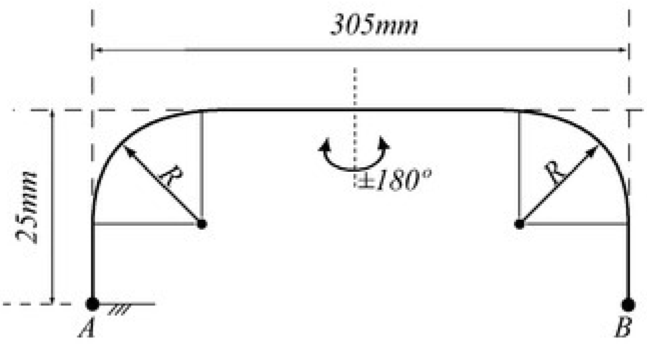
Table 5.12: *1 hour warm-up results.*

Load	Selected shape and length of the path	Average time	Cycles per minute
0% 10% 100%		T_{AV}=0.570	cpm= 105

This time the robot had a less good performance, after 1 hour warm-up the result was 105 cycles per minute, two less than with a 30 min warm-up and three less than without warming up the robot. This shows that the hottest are the motors, the performance reduces a little.

1 hour cool down:

Table 5.13: *1-hour cool down results.*

Load	Selected shape and length of the path	Average time	Cycles per minute
0% 10% 100%		T_{AV}=0.560	cpm= 107

After the one-hour cool down, the robot got the same results as in the first test, so this could mean that the fluctuation in the values after warming it up were correct.

5.5.- Maximum load

Due to the lack of time, this parameter will not be studied in this project. Even so, in a future it will.

Table 5.14: *Maximum load.*

Velocity	Applied load	Maximum reachable distance
10%	1kg	D_x=
50%	2kg	D_y=
100%	3kg	D_z=
	4kg	
	5kg	
	6kg	
	More (……)	

5.6.- Power consumption

Due to the lack of time, this parameter will not be studied in this project. Even so, in a future it will.

Standby:

Table 5.15: *Standby power consumption.*

V1 (V)	V2 (V)	I1 (A)	I2 (A)	P1 (W)	P2 (W)	P total (W)

Servo ON:

Table 5.16: *Servo ON power consumption.*

V1 (V)	V2 (V)	I1 (A)	I2 (A)	P1 (W)	P2 (W)	P total (W)

Movement:

Table 5.17: *Movement power consumption.*

Speed	Load	Joint	V1 (V)	V2 (V)	I1 (A)	I2 (A)	P1 (W)	P2 (W)	P total (W)
10% 100%	0% 10% 100%	1 2 3 4 5 6							

5.7.- Path accuracy and repeatability

Due to the lack of time, this parameter will not be studied in this project. Even so, in a future it will.

***Table 5.18:** Path accuracy and repeatability.*

Load	Velocity	Selected shape and length of the path	Accuracy	Repeatability
0%	10%		AT_p=	RT_p=
10%	50%		AT_a=	RT_a=
100%	100%		AT_b=	RT_b=
			AT_c=	RT_c=

5.8.- Maximum range

***Table 5.19:** Maximum range results.*

Joint	Datasheet values	Experimental values (before modifications)	Experimental values (after modifications)
1	±170°	R_{J1}= ±170°	R_{J1}= …
2	+83°/-110°	R_{J2}= +83°/-110°	R_{J2}= …
3	+180°/-67°	R_{J3}= +180°/-67°	R_{J3}= …
4	±180°	R_{J4}= ±180°	R_{J4}= …
5	±120°	R_{J5}= ±120°	R_{J5}= …
6	±360°	R_{J6}= ±360°	R_{J6}= …

The results show that the values are exactly the same, this is because the robot has a software-based limit for each joint and not only the mechanical ones. On every joint there is a little bit more range available, but for safety reasons the software does not allow them to get too close to the mechanical limits to prevent any collision.

The joints with a bigger improvement in the maximum range are the first one, it has a mechanical limit easy to remove, and we could make the robot have more than 360° of movement on this joint.

Figure 5.8: *Mechanical limit of the first joint.*

The second, third and fifth cannot be improved because these would collide with other parts of the robot.

Finally, the fourth and sixth joints are limited by software, so we could improve their range too.

6.- Environmental study

On this project, the environmental impact is quite small. There are two big factors to take care of, the first one is the energy consumed by the robot while testing its specifications, the second one is the components bought for the setups, sensors and more.

The robot's power consumption is rated at 1000 W with a maximum of 1800 W, as all the hours worked at Ingenia must be logged, it is easy to calculate the number of hours worked with the robot testing the specifications. In this case, the testing time was 205 hours out of the 710 total hours of the project. The energy can be easily calculated with the next formula.

$$E_t = P_t * hours$$

Equation 29

Counting the 205 hours and using the rated power consumption we get a total energy consumption of 205 kWh during the development of the project, the power consumption of other devices like the computer will not be calculated as the importance is much lower in the project.

The second factor is the awareness when buying electronic components. Ingenia is part of the Novanta corporation, which is integrating an environmental sustainability all across the Novanta companies. All the electronic devices and components acquired for the project are RoHS compliant, all the procedures were performed following the ISO 14001 standard. During the duration of the project, Novanta shared tests to raise awareness about sustainability too.

To conclude, it can be said that the environmental impact of this project was reduced to its minimum, the only improvement could have been the energy consumption of the robot, the problem is that this parameter cannot be reduced until the drives are changed.

7.- Conclusions

Throughout this project, a complete replicable test protocol has been developed for the characterization of the s a 6 DoF industrial robotic arm, and then it has been partially tested with a real robot to prove that the protocol is correct. Based on the results obtained on the tests, the protocol has been successfully designed and it meets all the primary objectives. The results clearly show some deficiencies in the CM607-L robotic arm compared to its competitors.

The approach used to obtain the final test protocol consists of an evolutionary process where it is studied how the specifications are tested in the industry, select the best fitting tests for the specific robotic arm, then select which components and devices are needed, design the setups and perform the tests. If the setups work properly, conclude the test, if not, repeat the process until the correct way to test is found.

In this project are presented seven different tests, which must be performed to fully characterize the specifications of the robot, even so, due to the short time available to develop the project, only three of them are actually tested in this thesis.

The first performed test was the maximum range of each of the joints. This test is the easiest to perform as it only needs the software of the robotic arm, it could be seen in the results that all the joints fulfil the specifications presented by the manufacturer. This is normal as the range is limited by software and cannot reach the mechanical limits. This parameter can be improved by changing the drives of the robot and improving the cabling, the first, fourth and sixth joints can increase the range of motion.

The second performed test was the maximum speed of the robot, which is the same as the maximum speed of the TCP. By researching how other manufacturers test this parameter, it can be seen that the most used path is the Adept cycle, even so there is an ISO path but no datasheet shows this value. For the Adept cycle was built a sensors setup and placed to achieve the demanded path. When testing the robot, it can be seen that its maximum speed is really poor, it achieves 107 cycles per minute without any load applied on the end effector, while the competitors average are 150 cycles per minute with a kilogram on the end effector. The performance of the CM607-L is nearly a 30% worse than the competitors.

The third and last test performed in this project was the maximum speed of each joint. The same setup as in the previous test was used in order to reduce the budget and the time. The idea was to create a personalized arc move for each of the joints and place the sensors on the path of the end effector and calculate the time it takes the robot to get from the first to the second sensor, and knowing the degrees between them calculate the velocity. To make sure the acceleration and deceleration did not affect the values, it was used the STP software to check the rise and fall time of the move. The results show some irregularities, even if the sensors were placed in the constant velocity zone, depending on the distance between them, the value of the speed varies, and it shows a higher speed as the sensors get closer, this problem is present in all the joints and in all the tests performed. This could be an error of the sensors, but it was not found even if the setup was checked. The results show that the first, third, fourth and fifth joints velocities are quite similar to the presented in the datasheet, it has less than 1% of error, the second joint is slower, it has a 16% error and the sixth joint is also slower, it has a 5% error. This test should be revised and maybe changed in order to delete the irregularities.

Despite successfully reaching all the main objectives, the secondary goal has not been completely accomplished. The reason behind this is that given the complexity of this project, and the short time available to develop it, priority has been given to the design of a complete and replicable test protocol rather than testing all the specifications with less accuracy. As this project was developed during an internship, the continuous revisions to check if the project followed the right path, the delays when buying components and other difficulties such as the budget, have also made it more complicated to develop all the test practically.

To conclude, this work is relevant because only with some tests performed it can be seen how this low cost/high performance Chinese robotic arm is not as good as its theoretical specifications could show, its velocity specifications do not always match the theoretical ones. This test protocol can be used to check that the specifications of other robots are correct and match the ones from the datasheets.

7.1.- Future work

This project is meant to be continued, at Ingenia, other students or engineers will follow this project in order to fully characterize the specifications of the CM607-L robotic arm. The importance of this test protocol to be replicable is because, in a future, other robotic arms will be bought and tested, this project will help to get the most accurate results possible, so the comparison between robots would be in the same conditions.

The next engineer is meant to check all the protocol and test it to confirm that is correct and, if not, change it and improve it. This project is part of a bigger project at Ingenia, where the idea is to replace all the drives of the robotic arm, place the ones designed at Ingenia and recharacterize all the specifications of the robot to check if they are able to improve these specifications and demonstrate added value of their drives.

8.- Economic analysis

This section contains a cost estimation for completing the entire project. It includes the number of hours required to design, implement, test and verify the results of the test protocol, the money spent on the setups and devices needed for the tests performed and the future components that will be needed to finalise the project, plus the price of the robot and the drives.

Since this project is implemented by an engineering student, the price established is 8€/hour, which is the standard price stipulated in the educational cooperation agreement between the UPC and the company where the project was developed. By taking into consideration the number of hours destined for this project, the cost of the hours spent on this project is shown in the next table.

Table 6.1: *Total worked hours.*

	Total worked hours (h)	Price per worked hour (€/h)	Total price (€)
Engineering student	710	8	5680

Next, the cost of all the material needed to develop the entire project is estimated, since some components could be changed in a future.

Table 6.2: *Components price.*

	Quantity	Price (€)	Total price (€)
Digital dial indicator	3	423.16	1269.48
TRICAL end effector	1	2300	2300
ATI Axia80 sensor	1	4990	4990
Speed test setup	1	182.4	182.4
ISO standards	1	197.67	197.67
Load test end effector	1	734.59	734.59
Total			9674.14 €

Finally, the price of the robot and the Ingenia drives that will be implemented in a future are collected in the next table.

Table 6.3: *Robot and drives price.*

	Quantity	**Price (€)**	**Total price (€)**
CM607-L	1	5555	5555
Ingenia drives	6	598	3588
Total			9143 €

The total price of the project is collected in the table below (*Table 6.4*).

Table 6.4: *Project's total price.*

Bloc	**Price (€)**
Worked hours	5680
Materials for the tests	9674.14
Robot + drives	9143
Total	24497.14 €

Once all the costs are added together, the total price of the project is expected to be around, 24500 €.

References

[1] *Robotic Arm Market demand, prize, scenario, & forecast analysis by 2029 (no date) Databridgemarketresearch.com. Available at: https://www.databridgemarketresearch.com/reports/global-robotic-arm-market (Last access: 5 June 2023).*

[2] *Products_Robot arm,6 axis robot,small robot arm,welding robot,mini robot arm,Suzhou Changmu Drive CO.,LTD (no date) Cmrobotic.com. Available at: https://www.cmrobotic.com/products_show.php?id=1&lm=1 (Last access: 5 June 2023).*

[3] *Mazzari, V. (2019) List of criteria to look at before buying a robot arm, Génération Robots - Blog. Available at: https://www.generationrobots.com/blog/en/list-of-criteria-to-look-at-before-buying-a-robot-arm/ (Last access: 5 June 2023).*

[4] *Laboratoire de commande et de robotique (no date) Etsmtl.ca. Available at: http://en.etsmtl.ca/unites-de-recherche/coro/accueil (Last access: 5 June 2023).*

[5] *Pollák, M. et al. (2020) 'Measurement of unidirectional pose accuracy and repeatability of the collaborative robot UR5', Advances in Mechanical Engineering, 12(12), p. 168781402097289. doi: 10.1177/1687814020972893.*

[6] *Morozov, M. et al. (2016) 'Assessing the accuracy of industrial robots through metrology for the enhancement of automated non-destructive testing', in 2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI). IEEE.*

[7] *Kuric, I. et al. (2021) 'Industrial robot positioning performance measured on inclined and parallel planes by double Ballbar', Applied sciences (Basel, Switzerland), 11(4), p. 1777. doi: 10.3390/app11041777.*

[8] *(No date) Researchgate.net. Available at: https://www.researchgate.net/publication/309713169_Testing_the_Performance_Characteristics_of_Manipulating_Industrial_Robots (Last access: 5 June 2023).*

[9] *Westfall, M. (no date) Motionics LLC – smart tools for monitoring and diagnostics of machinery, Motionics.com. Available at: https://motionics.com (Last access: 5 June 2023).*

[10] *Robotics, S. (2018) Measuring repeatability of robotic arm with dial indicator. YouTsube. Available at: https://www.youtube.com/watch?v=_Lgy9wlFQRw (Last access: 5 June 2023).*

[11] *Plc, R. (no date) Ballbar QC20, Renishaw. Available at: https://www.renishaw.es/es/sistema-de-ballbar-qc20--11075 (Last access: 5 June 2023).*

[12] *Metrology, A. P. I. (2019) API | robot accuracy enhancement | Cole tech. YouTube. Available at: https://www.youtube.com/watch?v=F86fKA782rE (Last access: 5 June 2023).*

[13] *FX1901-0001-0100-L (no date) Rs-online.com. Available at: https://es.rs-online.com/web/p/galgas-extensiometricas/8937398 (Last access: 5 June 2023).*

[14] *ATI Industrial Automation (no date) ATI Industrial Automation: F/T Models, Ati-ia.com. Available at: https://www.ati-ia.com/products/ft/ft_ModelListing.aspx (Last access: 5 June 2023).*

[15] *RGBFreak (2016) DIY | easy arduino laser tripwire security system! YouTube. Available at: https://www.youtube.com/watch?v=X18CfOKtyTQ (Last access: 5 June 2023).*

Accuracy and repeatability:

[16] *(No date) Researchgate.net. Available at: https://www.researchgate.net/publication/309713169_Testing_the_Performance_Characteristics_of_Manipulating_Industrial_Robots (Last access: 5 June 2023).*

[17] *UNE-EN ISO 9283:2003 (no date) Une.org. Available at: https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0029878 (Last access: 5 June 2023).*

[18] *ISO/TR 13309:1995 (no date) Une.org. Available at: https://www.une.org/encuentra-tu-norma/busca-tu-norma/iso?c=021679 (Last access: 5 June 2023).*

[19] *(No date b) Researchgate.net. Available at: https://www.researchgate.net/publication/238308032_Assessment_of_the_positioning_performance_of_an_industrial_robot (Last access: 5 June 2023).*

[20] *Vocetka, M. et al. (2021) 'Influence of drift on robot repeatability and its compensation', Applied sciences (Basel, Switzerland), 11(22), p. 10813. doi: 10.3390/app112210813.*

[21] *Raviola, A. et al. (2021) 'Effects of temperature and mounting configuration on the dynamic parameters identification of industrial robots', Robotics, 10(3), p. 83. doi: 10.3390/robotics10030083.*

[22] *Edmonds, S. and Lindberg, R. (no date) Energy optimisation of an industrial robot using iterative dynamic programming Master's Thesis in Systems, Control and Mechatronics, Chalmers.se. Available at: https://publications.lib.chalmers.se/records/fulltext/202879/202879.pdf (Last access: 5 June 2023).*

[23] *(No date) Researchgate.net. Available at: https://www.researchgate.net/publication/282774590_A_comparative_evaluation_of_three_industrial_robots_using_three_reference_measuring_techniques (Last access: 5 June 2023).*

[24] *(No date b) Researchgate.net. Available at: https://www.researchgate.net/publication/263180787_Characterization_and_experimental_evaluation_of_gear_transmission_errors_in_an_industrial_robot (Last access: 5 June 2023).*

[25] *(No date c) Researchgate.net. Available at: https://www.researchgate.net/publication/233920374_Modeling_and_assessment_of_the_backlash_error_of_an_industrial_robot (Last access: 5 June 2023).*

Speed of each axis:

[26] *Arduino millis() vs micros() (2019) The Robotics Back-End. Robotics Back-End. Available at: https://roboticsbackend.com/arduino-millis-vs-micros/ (Last access: 5 June 2023).*

[27] *Alfred, C. W. (2021) How to count pulse using arduino pulse counter. YouTube. Available at: https://www.youtube.com/watch?v=jZzpKifPUiE (Last access: 5 June 2023).*

[28] *ds3231 SQW enable 1hz practicing (2018) Arduino Forum. Available at: https://forum.arduino.cc/t/ds3231-sqw-enable-1hz-practicing/547702/5 (Last access: 5 June 2023).*

[29] *General Description (no date) DS3231 Extremely Accurate IC-Integrated RTC/TCXO/Crystal, Maximintegrated.com. Available at: https://datasheets.maximintegrated.com/en/ds/DS3231-DS3231S.pdf (Last access: 5 June 2023).*

[30] *Teensy® 4.1 (no date) Pjrc.com. Available at: https://www.pjrc.com/store/teensy41.html (Last access: 5 June 2023).*

[31] *PJRC (teensy) forum (no date) Pjrc.com. Available at: https://forum.pjrc.com/search.php?searchid=7746423&pp=&page=2 (Last access: 5 June 2023).*

Power consumption:

[32] *Making measurements on 3-phase motor drives with an oscilloscope (no date) Tek.com. Available at: https://www.tek.com/en/documents/primer/making-measurements-on-3-phase-motor-drives-with-an-oscilloscope (Last access: 5 June 2023).*

[33] *Electric motor power measurement and analysis (no date) Yokogawa.com. Available at: https://www.yokogawa.com/eu/library/resources/media-publications/electric-motor-power-measurement-and-analysis/ (Last access: 5 June 2023).*

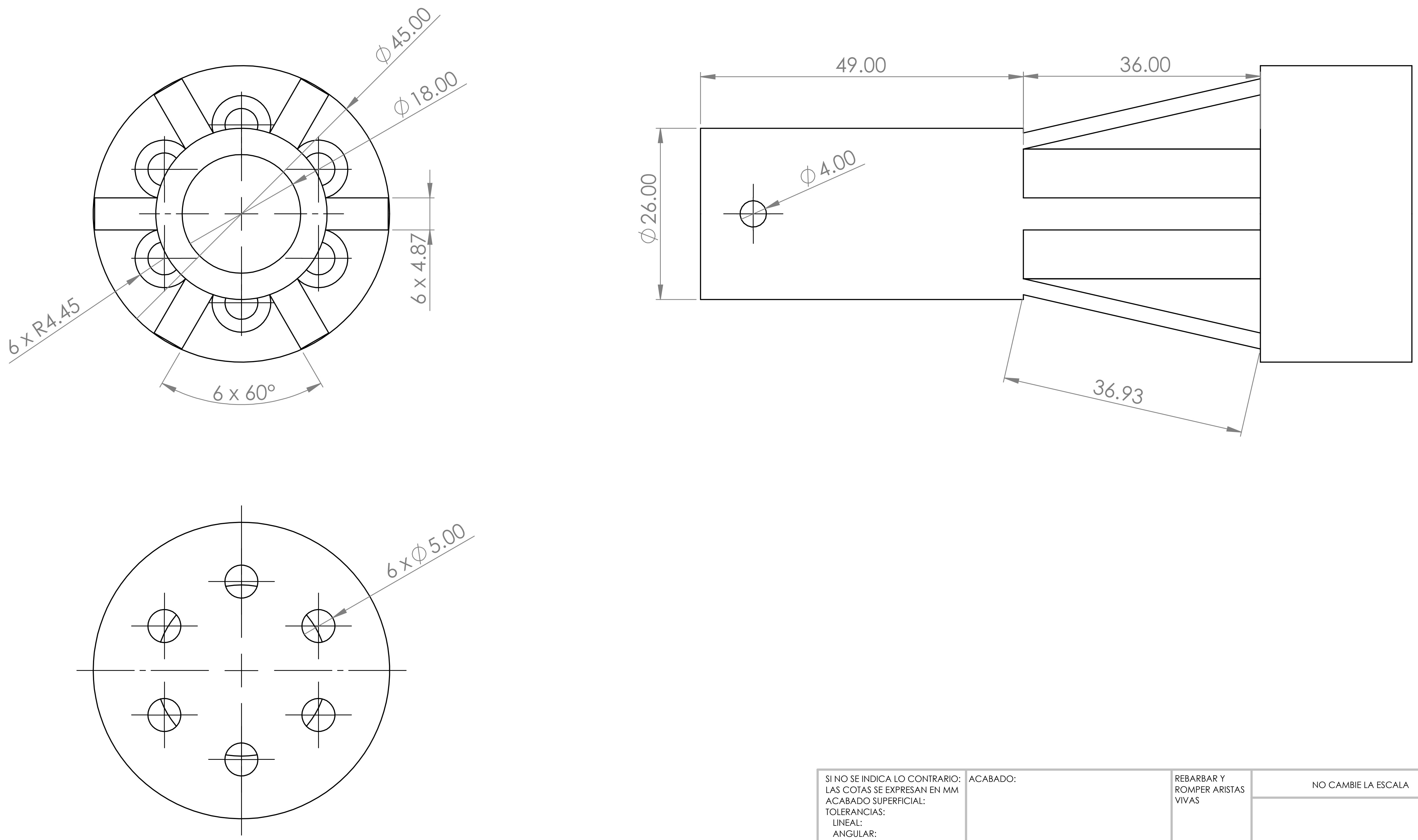
[34] *How to measure electrical power (no date) Yokogawa.com. Available at: https://www.yokogawa.com/eu/library/resources/media-publications/how-to-measure-electrical-power/ (Last access: 5 June 2023).*

[35] *Power Systems for Machinery and Est, E. (no date) Three phase current - calculation - power systems, Powergenerators-ksa.com. Available at: https://powergenerators-ksa.com/en/three-phase-current-calculation/ (Last access: 5 June 2023).*

Annex A

In the annexes can be found the drawings of all the end effectors designed for the tests.

A1. End effectors drawings



SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:	ACABADO:	REBARBAR Y ROMPER ARISTAS VIVAS	NO CAMBIE LA ESCALA	REVISIÓN
	NOMBRE	APELLIDO	FECHA	
DIBUJ.	Aleksandar	Lazaroie	06/06/2023	

TÍTULO:

Final thesis

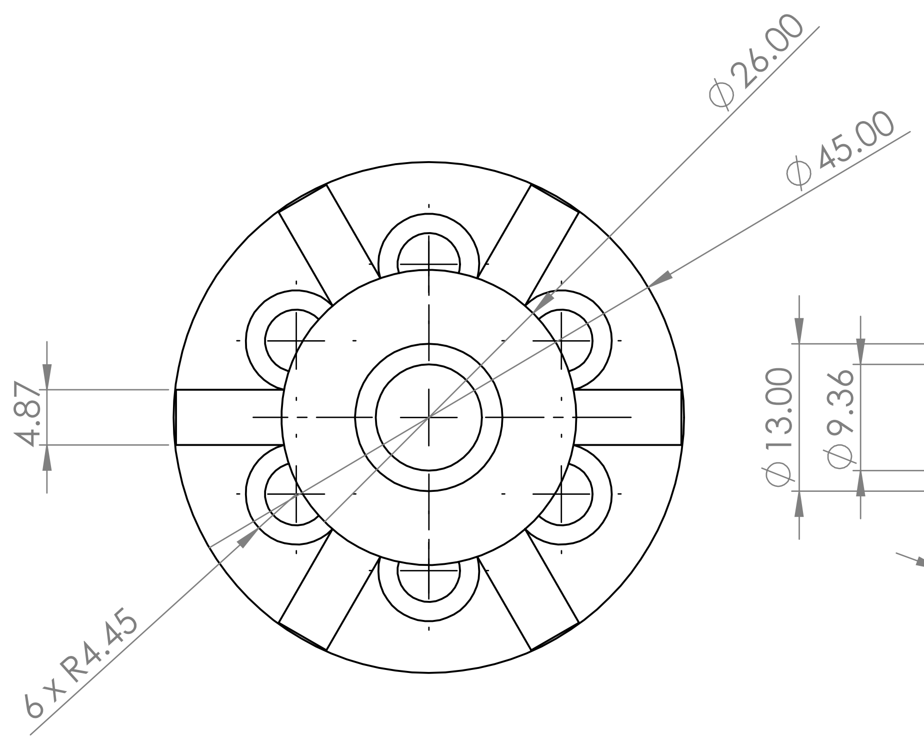
UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

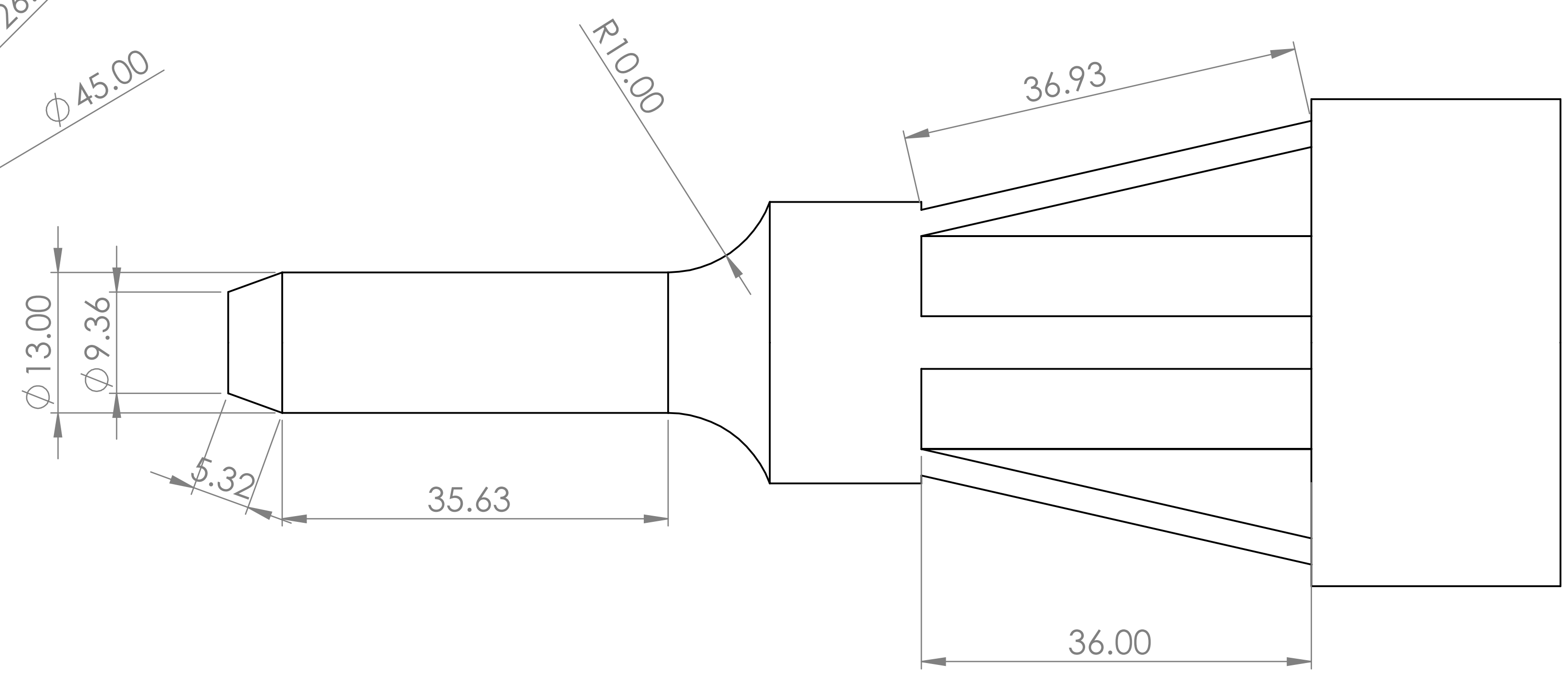
N.º DE DIBUJO

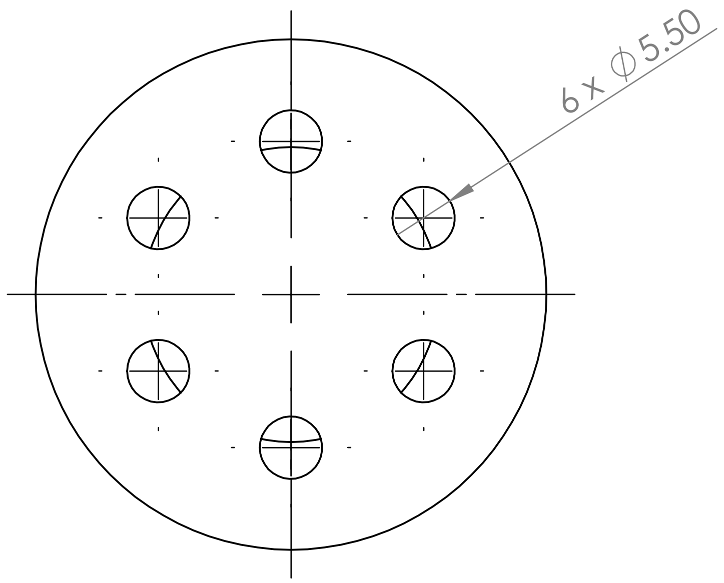
Marker end effector

A3

ESCALA:1.5:1 HOJA 1 DE 1







SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
ACABADO SUPERFICIAL:
TOLERANCIAS:
LINEAL:
ANGULAR:

ACABADO:

REBARBAR Y ROMPER ARISTAS VIVAS

NO CAMBIE LA ESCALA

REVISIÓN

	NOMBRE	APELLIDO	FECHA
DIBUJ.	Aleksandar	Lazaroie	06/06/2023

TÍTULO:

Final thesis

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

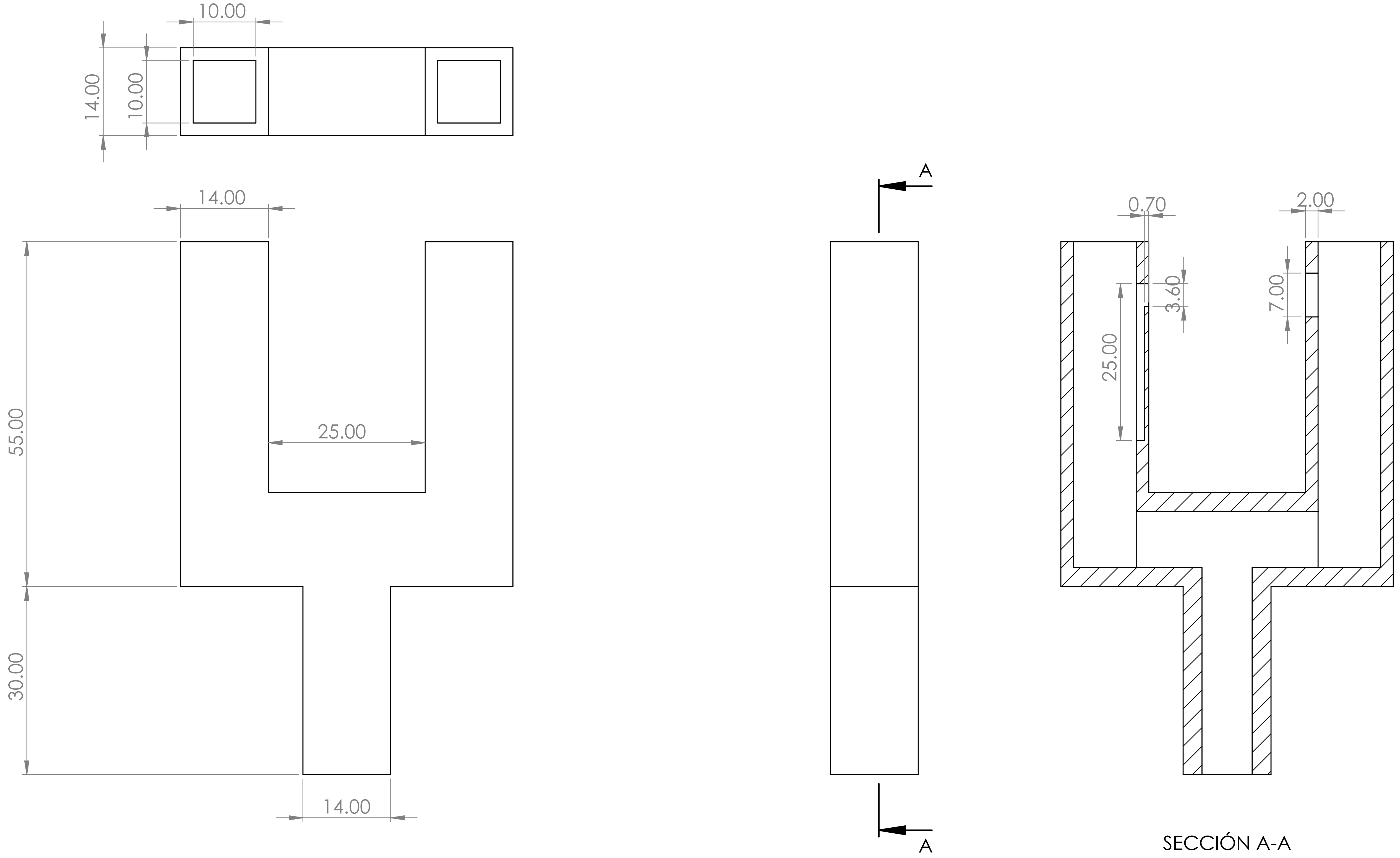
N.º DE DIBUJO

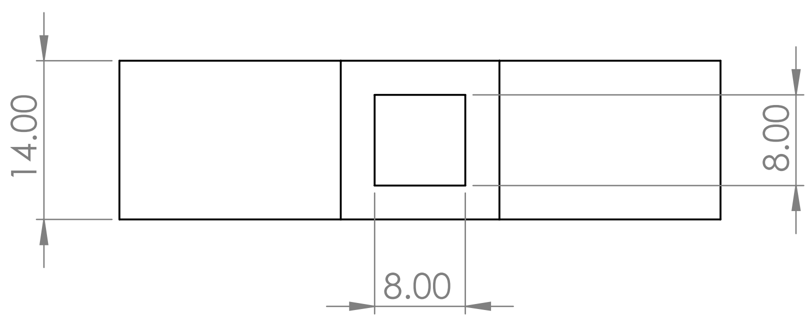
Speed test end effector

A3

ESCALA:1.5:1

HOJA 1 DE 1





SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
ACABADO SUPERFICIAL:
TOLERANCIAS:
LINEAL:
ANGULAR:

ACABADO:

REBARBAR Y ROMPER ARISTAS VIVAS

NO CAMBIE LA ESCALA

REVISIÓN

	NOMBRE	APELLIDO	FECHA
DIBUJ.	Aleksandar	Lazaroie	06/06/2023

TÍTULO:

Final thesis

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

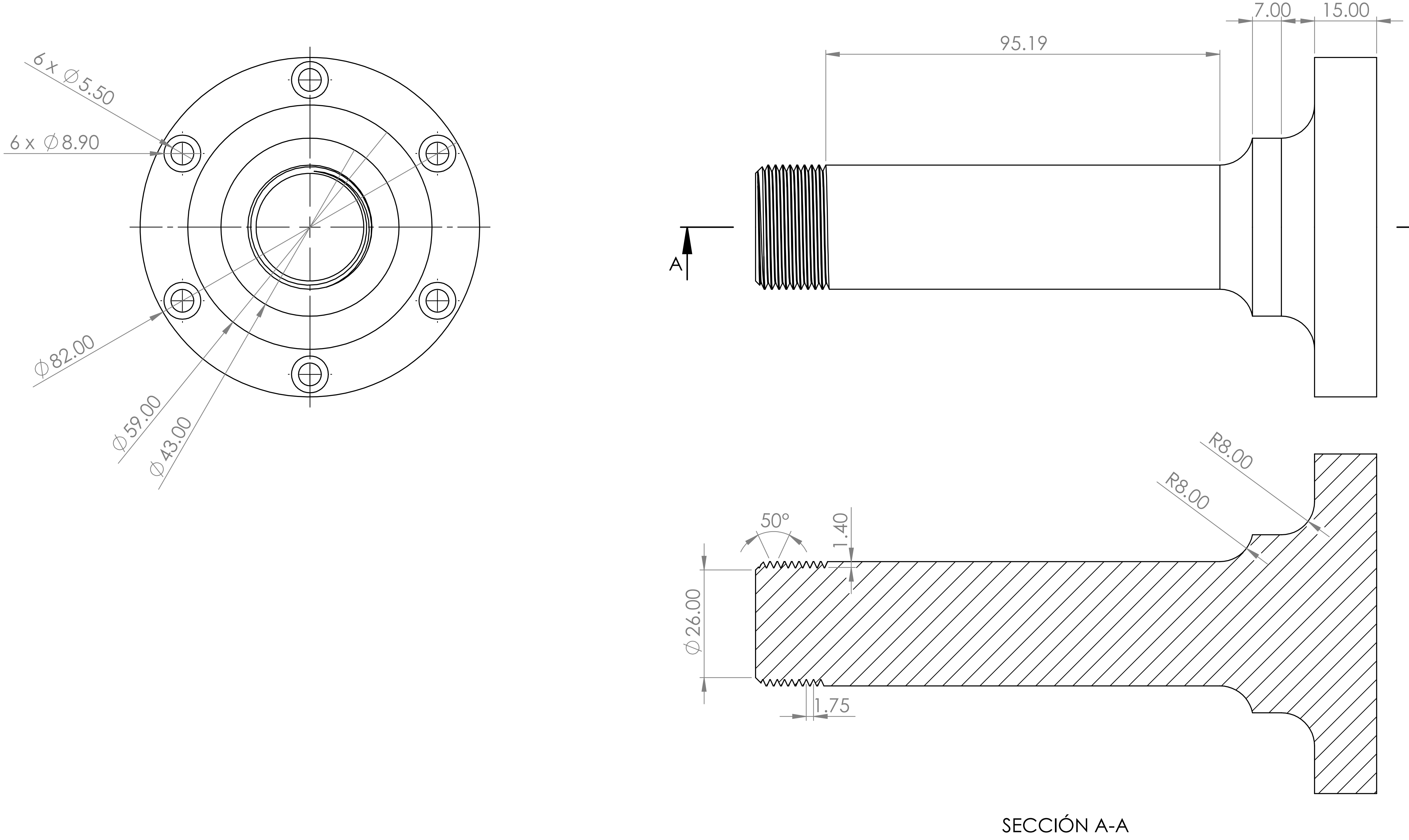
N.º DE DIBUJO

Laser fork barrier

A3

ESCALA:1.5:1

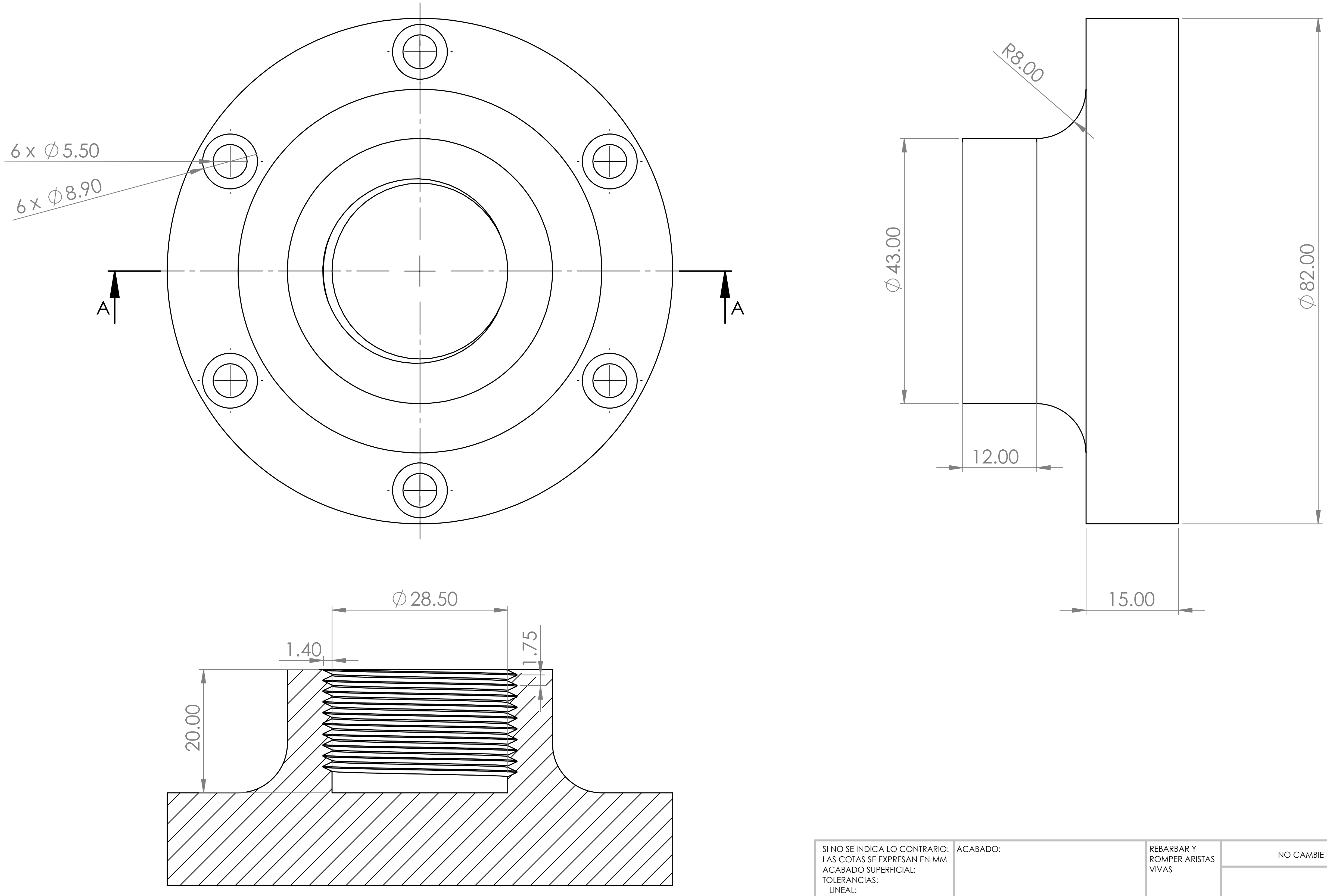
HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:	ACABADO:			REBARBAR Y ROMPER ARISTAS VIVAS	NO CAMBIE LA ESCALA	REVISIÓN
	NOMBRE	APELLIDO	FECHA		TÍTULO: Final thesis	
DIBUJ.	Aleksandar	Lazaroie	06/06/2023			

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

N.º DE DIBUJO
Final end effector
A3
ESCALA:1:1
HOJA 1 DE 1



SECCIÓN A-A

SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
ACABADO SUPERFICIAL:
TOLERANCIAS:
LINEAL:
ANGULAR:

ACABADO:

REBARBAR Y ROMPER ARISTAS VIVAS

NO CAMBIE LA ESCALA

REVISIÓN

	NOMBRE	APELLIDO	FECHA
DIBUJ.	Aleksandar	Lazaroie	06/06/2023

TÍTULO:

Final thesis

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

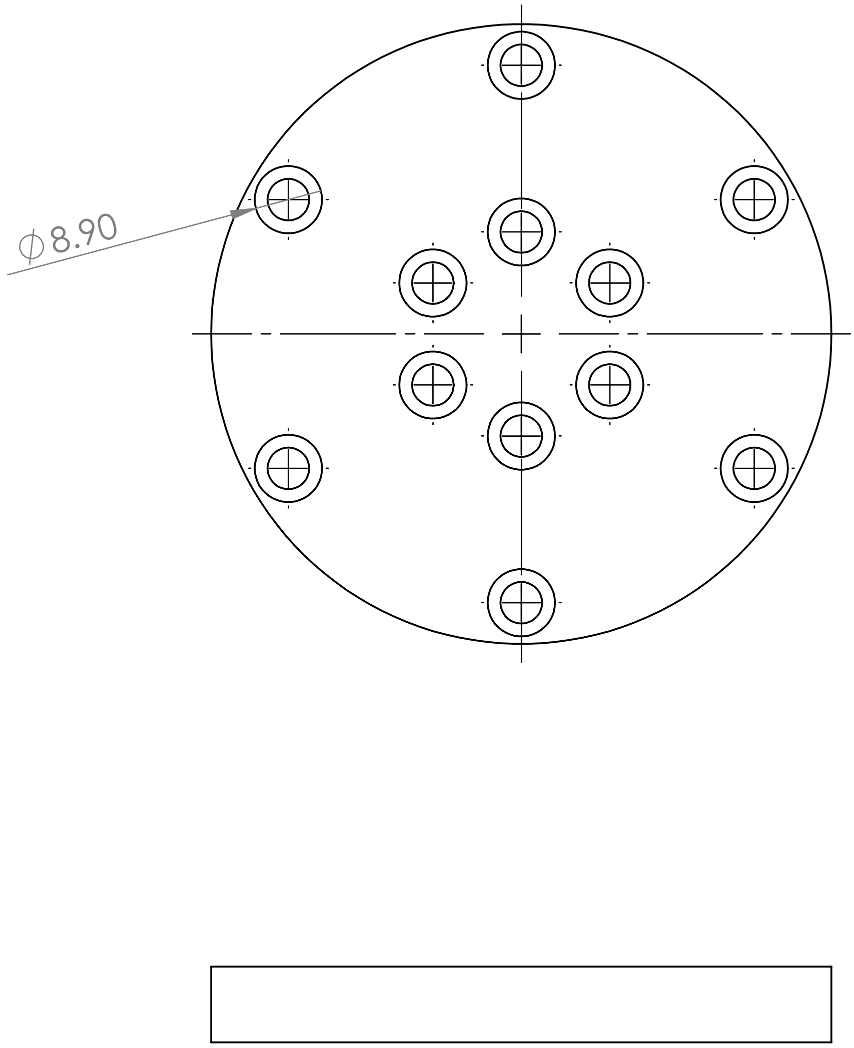
N.º DE DIBUJO

Final end effector nut

A3

ESCALA:1.5:1

HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
ACABADO SUPERFICIAL:
TOLERANCIAS:
LINEAL:
ANGULAR:

ACABADO:

REBARBAR Y ROMPER ARISTAS VIVAS

NO CAMBIE LA ESCALA

REVISIÓN

	NOMBRE	APELLIDO	FECHA
DIBUJ.	Aleksandar	Lazaroie	06/06/2023

TÍTULO:

Final thesis

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

N.º DE DIBUJO
End effector adapter

A4

ESCALA:1:1

HOJA 1 DE 1