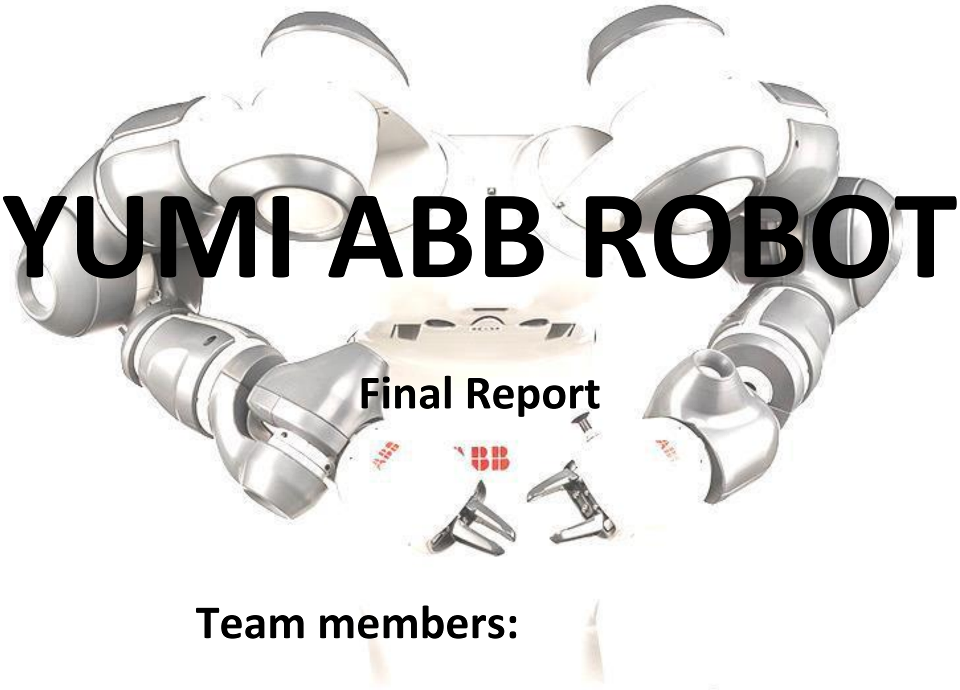
European Project Semester



YUMI ABB ROBOT

Final Report

Team members:

Johannes Ochsenknecht

Jessica Smith

Nick Bauwens

Xavier Carrera

Xavier de Miguel

Supervisor:

Mika Billings

Spring 2018

Abstract

Collaborative robots like ABB's YuMi, are a new tendency that will grow throughout industry in the coming years. The robots are a beneficial tool for helping and interacting with humans in the work place.

The purpose of this project is to design and develop a fully automated process for placing the information stickers and connectors in the relay boxes using YuMi robot with a maximum time of three minutes.

The time that European Project Semester offers is not long enough to develop a full industrial solution for such a big company as ABB, this project consists of basic ideas and programming to develop an excellent solution to be finished by a third company.

The steps of the project are mentioned below:

1. To program the YuMi robot to carry out the tasks as previously described.
2. To work with Tomi Latva in creating a layout that will be implemented to the ABB factory in the future.
3. To mount all the jigs and fixtures that the team have designed to show how the process will work when the layout is complete

The tasks that the third party will need to complete:

1. Programming of the big robot
2. Implementing the proposed layout
3. Re-building the jigs and grippers with a more sustainable material, resulting in a longer life span.

With the knowledge of the ideas above, not only programming the YuMi will be needed. Helpful accessories will need to be researched and optimized to achieve the best solution, as such as, a labelling machine to print the stickers, grippers, jigs and fixtures to improve the precision and another ABB robot, IRB 1600, to help YuMi. Despite the IRB 1600 is not available, we need to simulate his interventions with a mechanical support.

Acknowledgements

We are grateful for the Novia University of Applied Sciences for giving us the opportunity of working on this amazing project and to discover the Finnish country and culture.

We would like to send our gratitude to the following individuals for making this semester possible and one to remember:

- Roger Nylund for his informative classes, wisdom and sauna.
- Mika Billings for supervising our project, always making time to help when asked and for having such a great sense of humour.
- Tommi Latvala for sharing key parts of his project and always supporting us.
- Rayko Toshev and his EPS team for helping and assisting us with the 3D printing.
- Mikael Ehrs for giving us an excellent midterm report feedback.
- Mikko Viik and Jarmo Penttilä for being brief and direct with the requirements.
- Emilia Vikfors for the Swedish classes.
- Hanna Latvala for giving us the proper written English knowledge and report feedback.
- Camilla Mollis for giving us a great welcome and doing the paperwork essential for our studies.

Also, we would like to thank the other amazing EPS classmates, exchange students and tutors for the unforgettable, great time we spent together in Vaasa.

Without you this project would not been possible!

Contents

Figures list:

Table list:

1. Introduction

1.1 European Project Semester

In today's society, there is a much higher demand for a varied skillset of each individual when looking for employment. The large companies want to know how you can cope and develop in different surroundings and cultures to ensure you understand international clientele and working methods. The European Project Semester is an opportunity to show employers that you have this skillset and further confirms confidence and independence.

The European Project Semester is offered to all students in the nineteen partner universities throughout twelve countries in Europe, usually in their third or final year. It is the perfect way to grow as a person and discover new methods of working while developing a further competence in languages, team working and leadership. (Europeanprojectsemester.eu, 2018)

The largest participants in EPS are engineering students who split into multicultural teams and complete an intense project, working together alongside a company or supervisor. Usually teams are of four or five members, where the allocation of roles and tasks need to be carefully evaluated to ensure an optimum working environment. While completing this project, the students must also complete a series of classes which are related to teambuilding and project management. These classes are adjacent to the project and will improve the teams working environment with the knowledge that they gain. (Europeanprojectsemester.eu, 2018)

Figure 1 : Spring EPS Group

1.2 The Team

1.2.1 Belbin Team Roles

The Belbin Team Role allocator is a tool used widely by employers to indicate the personality traits of a specific individual. Consisting of a series of questions, each team member has allocated ten points to each answer to how they would most likely react to the scenario. At the end of the questionnaire a summary is given and derives what the strongest personality trait is that the individual holds, both thinking and action traits. There are nine different traits that can be awarded; shaper, implementer, finisher, plant, monitor, specialist, resource investigator, team worker and coordinator. It is best that within a team the role set is varied and therefore can produce an optimum working environment as all roles have different strengths and weaknesses that when combined create a healthy team.

(Belbin.com, 2018)

More in-depth explanations about the Belbin team roles and the nine traits, as well as the team members individual results, can be found in the Appendix. Although, in summary, our team consists of three implementors, two coordinators, two plants, a resource investigator, a completer and a shaper. Resulting in the only traits the team are lacking is a team worker and a specialist. This is a good diversity of roles within the team, Mika Billing acts as our specialist and is available for guidance and teaching of the Robotstudio software to be used. Therefore, the only role completely missing from the team is that of the team-worker. As this role consists of being a great listener and having the ability to avert friction – these may be issues that the team will need to adapt to and learn how to respect for the project to work as planned.

From left; Johannes Ochsneckt, Xavier Carrera, Xavier de Miguel, Mika Billing, Nick Bauwens, Jessica Smith, Tommi Latva

Figure 2 - ABB Team

1.2.2 EPS Team

Our international EPS team consists of the following members:


Figure 3 - Xavier de Miguel

Xavier de Miguel, Spain

Home University: Polytechnic University of Catalonia

Field of study: Electronical and Automation Engineering

Action: Implementer

Thinks: Coordinator

Role within the team: Team Leader

Johannes Ochsenknecht, Germany

Home University: Osnabrueck University of Applied Sciences

Field of study: Mechanical Engineering, specialising in Research and Development

Action: Implementor

Thinks: Shaper

Role within the team: Vice Team Leader


Figure 4 - Johannes Ochsenknecht


Figure 5 - Jessica Smith

Jessica Smith, Scotland

Home University: Glasgow Caledonian University

Field of study: Mechanical Engineering with Computer Aided Design

Action: Resource Investigator

Thinks: Plant

Role within the team: Secretary

Nick Bauwens, Belgium


Figure 6 - Nick Bauwens

Home University: Artesis Plantijn University College

Field of study: Electro-mechanics, specialising in Automation

Action: Completer/Finisher

Thinks: Plant

Role within the team: Scrum Leader


Figure 7 - Xavier Carrera

Xavier Carrera, Spain

Home University: Universitat de Lleida

Field of study: Master’s degree in industrial engineering

Action: Implementor

Thinks: Co-ordinator

Role within the team: Method and Materials Manager

2 ABB

2.1 ABB Company

ABB are a well-known, international company who has bases in over 100 countries. The company's headquarters are in Switzerland however, they consider themselves as Swedish-Swiss. It is one of the largest companies in the world, specializing in robotics, power and many other engineering departments. (New.abb.com, 2018)

The company is split into four different branches with specialists in each sector; Electrification, Robotics and Motion, Industrial Automation and Power Grids. Each of these departments produce and offer products to customers in other engineering firms and industries. (New.abb.com, 2018)

The electrification distribution offers products to clients in the building industries and utility operations. They specialise in manufacturing all electrical equipment such as wiring equipment, switches, power sensors etc. (New.abb.com, 2018)

In the robotics and motions main client base is the industrial protection companies. They offer products such as renewable energy convertors, robotics, motors and many other components related to power generation. (New.abb.com, 2018)

Industrial Automation specialise in producing measurement equipment and create components specifically to the company's request. The main customer to this base are industrial companies. (New.abb.com, 2018)

Finally, the power grids division distributes products relating to automation and electricity. Essentially, they supply components to companies with electrical equipment. (New.abb.com, 2018)

Being such a large company, ABB distribute to all different engineering companies all over the world. These include; the government, building industries, power suppliers and many more. ABB produce products that suit the needs of all their customers. These products range from switches, safety relays, spare parts, automation programs and many more products. Each sector in the ABB Company aims to produce products to an optimum standard to ensure the return of their customers. (New.abb.com, 2018)

2.2 YuMi Robot

The YuMi robot was designed and manufactured by ABB themselves. They had a team of engineers who produced the dual-arm collaborative robot which can be programmed to do simple tasks without being in a caged environment. (New.abb.com, 2018)

The aim of having the YuMi is to work alongside humans in a production line doing basic tasks, freeing the engineers to complete the more intricate duties at hand. The sensors embedded in the hands of the robot ensures that it is no danger to humans as it halts to a stop whenever it is close to another being or object. (New.abb.com, 2018)

Figure 8 - YuMi Robot. Source: ABB

The hands of the YuMi robot can consist of camera, grippers and a suction cup. Below are the designs of these that ABB have already configured:

Figure 9 - YuMi hands. Source:ABB

The grippers on these hands are the same in all, however, new designs can be easily drawn in Autodesk Inventor to be 3D printed. The only main specification that must be the same is the holes in which the screws go inside to hold the gripper in place. (New.abb.com, 2018)

YuMi SPECIFICATIONS

YuMi data sheet	
Brand	ABB
Model	YuMi
Specification	
Payload	500 g
Reach	559 mm
Weight	38 Kg
Base dimensions	399 x 496 mm
Power supply	100 - 240 V
Features	
Integrated signal and power supply	24 V Ethernet or 4 signals
Integrated air supply	1 per Arm on tool Flange (4 bar)
Integrated ethernet	One 100/10 Base-TX ethernet port/per arm
Repeatability	0,02 mm
Number of axis	7 per arm
Maximum axis speed	180°/s
Robot mounting	Table
Degree of protection	IP30
Functional safety	PL b Cat B
Input/output interfaces	
Ethernet IP, Profibus, USB ports, DeviceNet™, communication port, emergency stop and air-to-hands	
HMI devices including ABB's teach pendant, industrial displays, commercially available tablets and smartphones	

Table 1 - YuMi Specifications

2.3 YuMi Accessories Install

2.3.1 Left Side of YuMi Connections

The connections on the left side panel of YuMi are displayed in the following illustration:

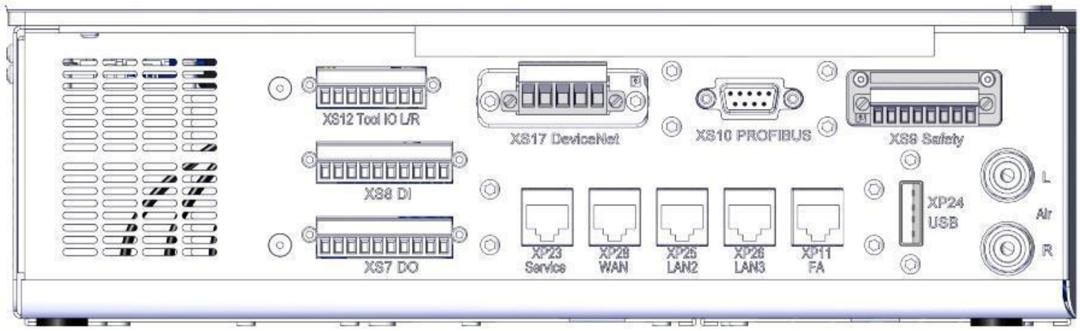


Figure 10 - YuMi Left Side. Source : Product Specificatiton IRB 14000

Where:

- XS12: Tool I/O, left and right arm. 4x4 digital I/O signals to the tool flanges, to be cross connected with XS8 and/or XS9
- XS17: DeviceNet Master/Slave
- XS10: Fieldbus adapter. PROFIBUS Anybus device (fieldbus adapter option)
- XS9: Safety signals
- XS8: Digital inputs. 8 digital input signals (approx. 5 mA) to the internal I/O board. Pin number 9 (24 V = max current 3A)
- XS7: Digital outputs. 8 digital output signals (150 mA/channel) from the internal I/O board. Pin number 9 (24 V = max current 3A)
- XP23: Service
- XP28: WAN (connection to factory WAN)
- XP25: LAN2 (connection of Ethernet based options)
- XP26: LAN3 (connection of Ethernet based options)
- XP11: FA = Fieldbus adapter. PROFINET or EtherNet/IP (fieldbus adapter option)
- XP24: USB port to main computer
- Air L: Air supply, left arm. O.D. 4 mm air hose, 0.6 MPa air pressure
- Air R: Air supply, right arm. O.D. 4 mm air hose, 0.6 MPa air pressure

2.3.2 Right Side of YuMi Connections

The following figure explains the connections on the right-side panel of the controller:

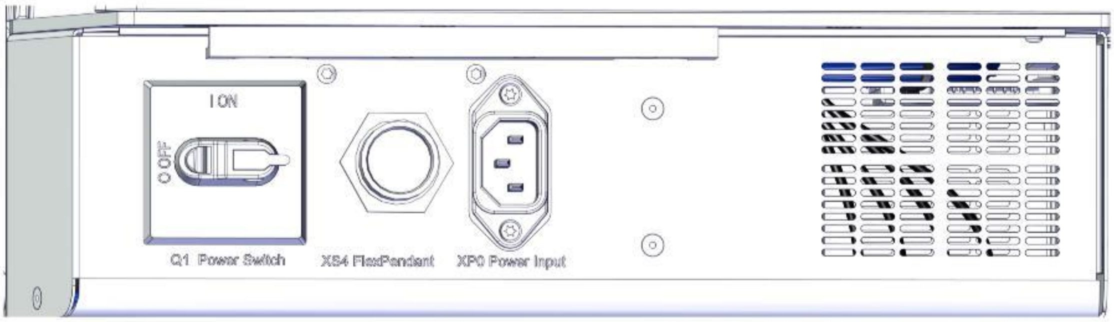


Figure 11 - YuMi Right Side. Source : Product Specification IRB 14000

Where:

- Q1: Power switch. On/Off
- XS4: FlexPendant
- XP0: Power input. Main AC power connector, IEC 60320-1C14, 100-240 VAC, 50-60 Hz

2.3.3 Connections Used

The connections that will be used for the robot operation are listed below:

Left side:
- XS8: Digital inputs
- XP23: Service
- XP28: WAN
- XP25: LAN2
- Air L: Air supply, left arm

Right side:
- Q1: Power switch
- XS4: FlexPendant
- XP0: Power input

Figure 12 - Connections Used Left and Right Side

2.3.4 Buttons and Wiring Install with Connections Installation

For the use of the robot, two buttons will be installed in the digital input (XS8). YuMi has eight digital input signals in the electronic board. According to the YuMi manual, the pin number nine supplies 24 V and a maximum current of 3A.

These switches will be connected to pins one, eight and nine. The digital inputs will be known in the program code (RobotStudio) as a I_S0 for the switch connected to pin one and I_S7 for the switch connected to pin eight.

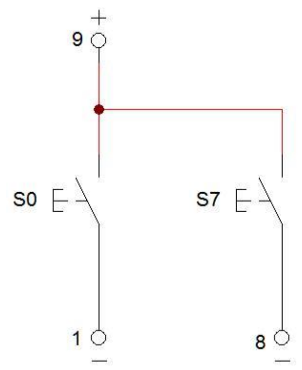


Figure 13 - Electric Scheme for the Digital Outputs

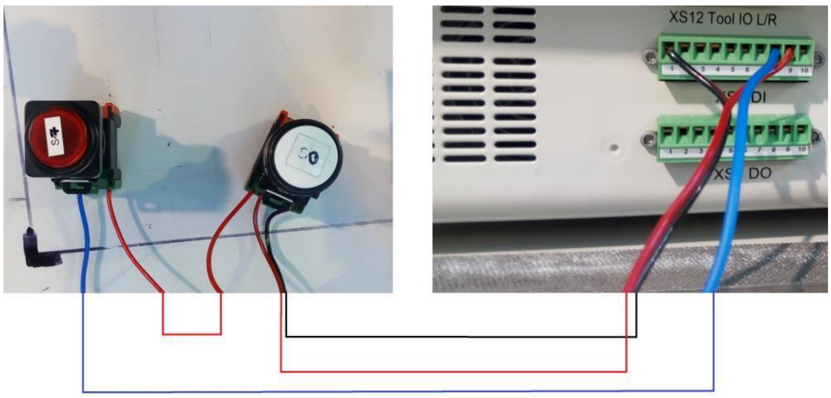


Figure 14 - Installation of the Switches to YuMi for the Digital Inputs

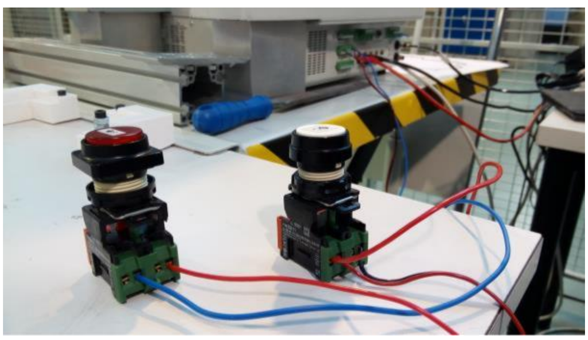


Figure 15 - Manual Buttons Used with YuMi

2.3.5 Pneumatic Install and Connection Explanation for Vacuum Suction Cup

The hands of YuMi consists of a vacuum suction cup, pressure sensor and a blow-off actuator. The vacuum system will be used to pick and the labels in the correct position.

Suction capacity can differ according to the following causes:

- Suction cup shape
- The item surface that will be picked
- Air pressure entrance to YuMi
- YuMi arm movement
- Item picking point and its centre of gravity

For the suction operation, a four millimetre air hose is installed from the air supply to the air input of the YuMi robot that is located on the left side. The labels will be placed with the left robot arm, so the hose will be plugged into the air left input.

Figure 16 - Air Hose Installation

The specification sticker on the right side of YuMi indicates that the maximum pressure is 5 bar as it can be seen on the following figure:

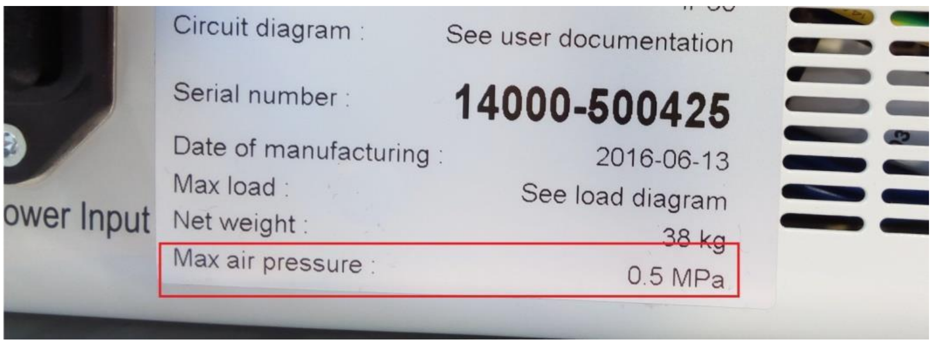


Figure 17 - YuMi Maximum Air Pressure

The air supply located near the robot will be opened and the pressure will be adjusted to 5 bar with the pressure regulator to match the requirements.

Figure 18 - Air Supply Regulator

A test is carried out to ensure that the suction system can pick the label, place it and finally do the blow off for placing the sticker. The results of the test were satisfactory and fulfilled the expectations.

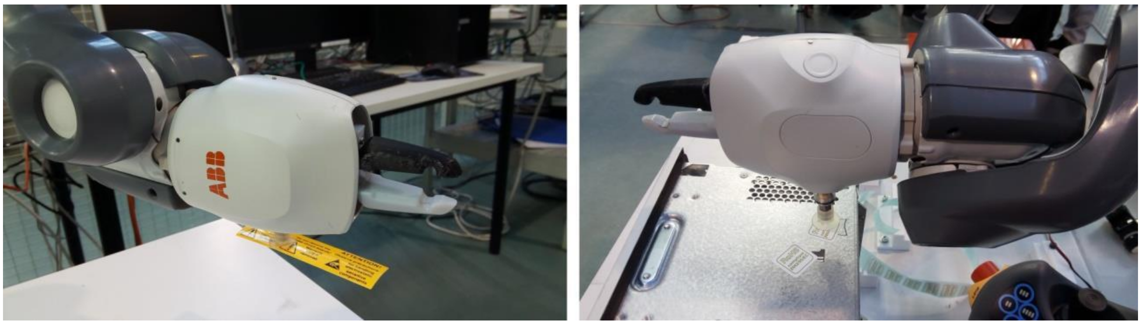


Figure 19 - Label Suction Test

3 Project Background

As a team of five international students, we have been given the opportunity to work alongside the world-wide company ABB. Our aim is to replace a human worker with the YuMi robot to enhance production and to make workings more efficient. All work will be documented, and the team will meet regularly with ABB and supervisors to ensure all work is being completed to a high enough standard with the company feeling pleased with the progress that is being made. There will also be two presentations that will be completed to make everyone aware of how the ideas are being implemented.

The task is very direct and exact, our main problem to solve is how to place a connector, as well as three stickers onto a relay box using YuMi robot. Before this can be completed, many brainstorming ideas must be had and the ones that the team feel most confident in will be brought to ABB and confirmed whether this is to the required standard and possible within budgets. The project is quite open, ABB confirmed that they will be pleased to try most of our ideas with hope that the desired outcome will be achieved and can be used within the company for future use.

As the YuMi robot already exists and is used today, we have been tasked to program the robot to place the connectors and the stickers onto the box for the safety relay. Accompanied by the programming, things such as; changing the full layout of the space allocated in the factory using RobotStudio simulation software, designing two sets of grippers for the YuMi and a jig for the relay box using Autodesk Inventor and 3D printing need to also be completed. Therefore, each team member will be responsible for a different task of which they feel they can undertake and complete to the highest standard.

3.1 Mission

Our mission is to substitute a human worker with the YuMi robot by placing both the labels and a connector plate into the casing for a safety relay.

3.2 Vision

We will do this by conducting intensive research about collaborative robots and the Robotstudio software before programming YuMi to achieve the required tasks.

3.3 Components to be worked with

3.3.1 Cases

There are two separate casings that the team will be working with, one smaller case and one larger one. The connector and stickers will be placed in the same location on each of these cases, therefore the main issue that the casings propose is the jig that should be designed to compensate for both sizes. Each case is used to manufacture different models of safety relay boxes.

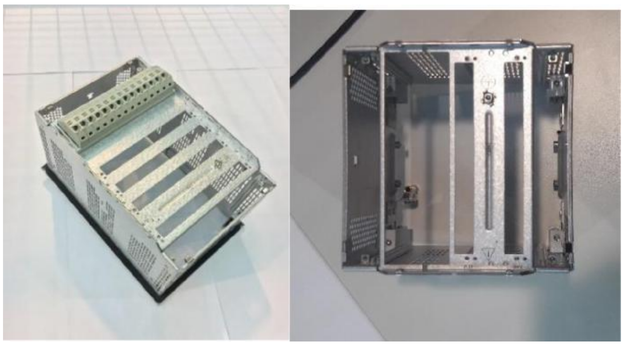


Figure 20 - Large Casing and Small Casing

The boxes used in the project that the YuMi robot will work with are part of the ABB's family product called Relion. Relion is an ABB product family for the protection, control, measurement and supervision of power systems.

The Relion family is composed by seven different models which are:

- 605 Series
- 611 Series
- 615 Series
- 620 Series
- 630 Series
- 650 Series
- 670 Series

Figure 21 - Relion Family Product. Source: ABB

Relion product range is used for:

- Feeder protection
- Transformer protection
- Motor protection
- Generator protection
- Voltage protection
- Frequency protection
- Capacitor bank protection
- Busbar protection
- Arc fault protection (arc short-circuit)

A plug-in unit that has all the necessary components for the correct functionality of the product is placed in the box where YuMi is to put the stickers and the connector.

In the picture below is shown how the plug-in unit is installed to the box.

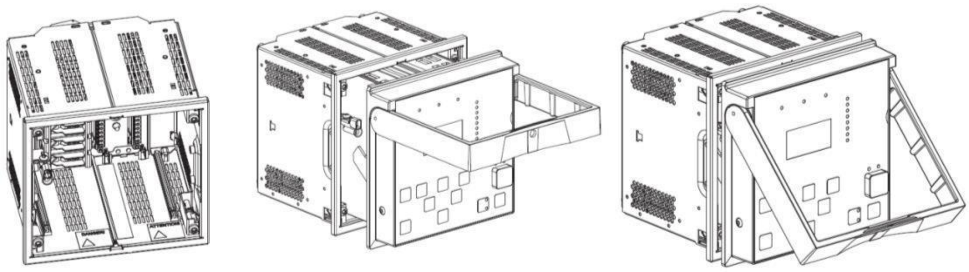


Figure 22 - Installation of a Relion Unit. Source: ABB

The models that ABB requires to optimize in their assembly line are 611, 615 and 620.

Figure 23 - Relion Models 611, 615 and 620. Source: ABB

The 611 and 615 series use the same box for both models. Therefore, the box has the same dimensions for two different products. However, the 620 series uses a different casing that is larger. The dimensions are in the appendix section.

3.3.2 Connectors

ABB requires three different connectors to be placed into the casings, the combinations of these to each of the connectors to the casings will be varied to match the customers' orders. This will not prove to be an issue because all the connectors are the same size, only the number of pins in each will vary. However, the programming of the YuMi will need to compensate for picking the three different connectors from three various locations.

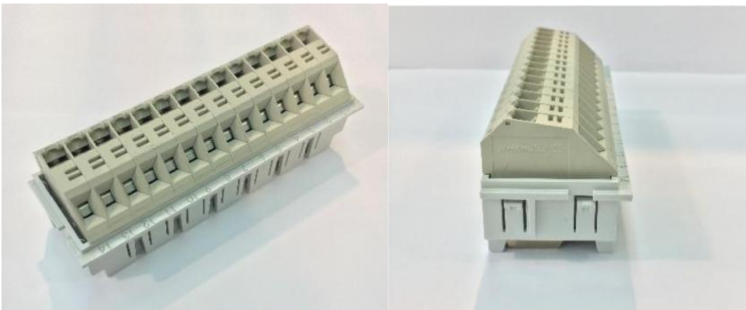


Figure 24 - Connector

3.3.3 Stickers

There are six different stickers that are to get placed on the casings, three in English and three in Chinese. These will dependant on the order or country of the customer. Our task is to find a way in which YuMi can place these on the inner surface of the casing.

Figure 25 - Labels

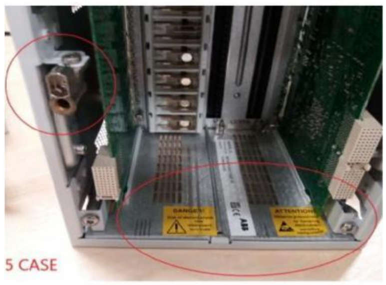


Figure 26 - Mounted Labels

4 Research

4.1 Collaborative Robot research

What are Collaborative Robots?

A collaborative robot is a machine capable of working alongside humans in a perfectly safe and comfortable environment. Due to their advanced technology, the cobots do not need to be kept enclosed in cages like previously more industrial robots. This is a result of the sensors deposited into the robots, they are aware of their surroundings and thus can stop movement if they touch something out of their programming capacities. They are usually lightweight and the danger the mass of the robot would be to a human is very miniscule, even if it were to go out of control. The only threat it poses to human life would be subject to the work it is carrying out i.e. with dangerous or flammable equipment. Therefore, all simple and boring jobs that humans do not like to complete, the collaborative robot can be programmed to do which makes the industrial working life better for employees as well as speeding up the process.

(care? and Digest, 2018)

Applications of Collaborative Robots

There are many ways in which a collaborative robot can work. Depending on the company and the needs of the customer, the robot can be designed and programmed to complete almost any production task required. Some of these applications include;

- Machine Maintenance
- Packaging
- Material Handling

(Bélanger-Barrette, 2018)

- CNC Machining
- Loading and Unloading
- Metal Fabrication
- Moulding Operations
- PCB Handling and ICT
- Test and Inspection

(Rethink Robotics, 2018)

These are to name but a few applications for collaborative robots. Specifically, to the YuMi robot we will be compiling a program that will instruct the YuMi to place stickers on a component and to insert the connector into the safety relay. It is in our best interest to find the most effective way to do this and to derive the coded program that will allow this to occur. As previously explained, these are simple applications for the cobot to endure, however, it will make the production process much quicker and allow for humans to complete the more intricate tasks at hand.

4.2 Collaborative Robot Competitors

The most important collaborative robot companies with his respective models are shown in the next table:

Company	Model	Payload [Kg]	Weight[Kg]	Reach[mm]	Degrees of freedom
ABB	YuMi	0,5 per arm	38	500	7 per arm
AUBO Robotics	I5	5	24	880	7
Comau S.p.A	AURA	110	685	2210	6
Denso Wave Incorporated	COBOTTA	0,5	3,8	310	6
F&P Robotics AG	PROB 2R	3	20	775	6
FANUC Corporation	CR 35iA	35	990	1813	6
	CR 4IA	4	48	550	6
	CR 7IA	7	53	717	6
	CR 7IA/L	7	55	911	6
Franka Emika GmbH	EMIKA	3	18,5	800	7
KAWADA Robotics	NEXTAGE	1,5 per arm	29	577	15
Kawasaki	Duaro1	2 per arm	200	760	15
KUKA AG	LBR IIWA 7 R800	7	22	800	7
	LBR IIWA 14 R820	14	30	820	7
Life Robotics	CORO	2	26	865	6
MABI AG	SPEEDY 6	6	28	800	6

	SPEEDY 12	12	35	1250	6
MRK-System GmbH	KR 5 SI	5	150	1423	6
Precise Automation	PAVP6	2,5	28	432	6
	PF400	1	20	576	4
	PP100	2	20	685 or 1270	4
Productive Robotics	OB7	5	24	1000	7
Rethink Robotics	SAWYER	4	19	1260	7
	BAXTER	4	19	1260	7
Robert Bosch GmbH	APAS Assistant	4	230	911	6
Stäubli	TX2-60	3,5	51,4	670	6
	TX2-60L	2	52,5	920	6
	TX2-90	7	114	1000	6
	TX2-90L	6	117	1200	6
	TX2-90XL	5	119	1450	6
Techman Robot	TM5-700	6	22	700	6
	TM5-900	4	22,2	920	6
Universal Robots A/S	UR3	3	11	500	6
	UR5	5	18,4	850	6
	UR10	10	28,9	1300	6
Yaskawa Motoman Robots	HC10	10	47	1200	6

Table 2 - Comparison Chart between collaborative robots. Source: Collaborative Robots Buyer's Guide, 6th Ed (Cobots et al., 2018)

(Collaborative Robots Buyer's Guide. 6th Edition., 2018)

4.3 Collaborative Robot Manufacturers

The most important collaborative robots companies with his respective models are shown in the next table:

Company	Company Logo	Country	Model	Image
ABB		**Switzerland**	**YuMi**	
AUBO Robotics		**USA**	**I5**	
Comau S.p.A		**Italy**	**AURA**	
Denso Wave Incorporated		**Japan**	**COBOTTA**	
F&P Robotics AG		**Switzerland**	**PROB 2R**	
FANUC Corporation		**Japan**	**CR Series**	

Company	Company Logo	Country	Model	Image
Franka Emika GmbH		Germany	EMIKA	
KAWADA Robotics		Japan	NEXTAGE	
Kawasaki		**Japan**	**Duaro1**	
KUKA AG		Germany	**LBR IIWA Series**	
Life Robotics		**Japan**	**CORO**	
MABI AG		Switzerland	**SPEEDY Series**	

Company	Company Logo	Country	Model	Image
MRK-Systeme GmbH		**Germany**	**KR 5 SI**	
			PAVP6	
Precise Automation		**USA**	**PF400**	
			PP100	
Productive Robotics		**USA**	**OB7**	
Rethink Robotics		**USA**	**SAWYER**	

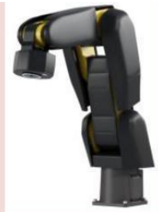
Company	Company Logo	Country	Model	Image
Rethink Robotics	rethink robotics	USA	BAXTER	
Robert Bosch GmbH		Germany	APAS Assistant	
Stäubli	STÄUBLI	Germany	TX2 Series	
Techman Robot		Taiwan	TM5 Series	
Universal Robots A/S	UNIVERSAL ROBOTS	Denmark	UR Series	
Yaskawa Motoman Robots	YASKAWA MOTOMAN ROBOTICS	Japan	HC10	

Table 3 - Collaborative Robot Competitors. Sources: Collaborative Robots Buyer's Guide. 6th Ed., www.mordorintelligence,com adn www.marketsandmarkets.com

4.4 Labelling Machines and Printing Methods

In this section, three different kinds of printing methods are discussed that are within the ABB requirements:

The requirements that ABB stablished are the following ones:

- The labels are warning labels, due this, they must be bright colored (yellow if it is possible).
- They should not fade over time. The reason of this condition is the stickers possibly being exposed to high contact and the labels needing to last for the lifespan of the relay.

Some secondary requirements that the company has not told us directly, but our team thinks that they are quite important are below:

- If there are some replacements of cartridges, ribbons, paper roll or other maintenance work that cannot be done automatically, it should take as little time as possible. Otherwise, the company would have to use some human resources to solve these kinds of situations.
- The printer must be programmable. The reason for this is because the printer,

  RobotStudio and the company's order manager software (SAP) have to be linked. That means that the printer must know which type of label distribution it has to print depending on the order, Chinese or English version. This can be done by different data transfers as Ethernet, Wi-Fi or Bluetooth. Ethernet has the most reliable connection due to it is a fixed cable and not a wireless technology.
- The labels must be printed without the back paper. It is easier for YuMi to work with them in that way.

Therefore, some research has been done to find the printer and printing method that fits better to the requirements.

<u>PRINTING METHODS</u>

All the following methods can combine with a programmable printer.

Normal ink printer

Ink printing is the most-common method of printing in the industry world. The stickers can be printed in all the desired sizes and colors. It needs replacements for the ink cartridges or toners. Labels usually will fade with high contact, however, it is an affordable technology.

Figure 27 - Ink Printer Cartridges

Direct thermal printing

Direct thermal printers utilize a chemically treated material that blackens when the thermal print-head applies heat to its surface. It does not require replacements of ink toners or ribbons and it is quite affordable. The problem of this technology for our project, is that it may fade over time and it can only print in black and white, unless the labels are preprinted with special thermos chromatic ink.

Thermal transfer printing

Thermal transfer printing consists of a thermal print-head that applies heat to a carbon, wax or resin-based ribbon, which is melted to the label's surface, resulting in the color being absorbed by the label. The problem of using this method in our project is that the ribbons must be replaced frequently, and it has some label size limitations. The advantages of it is that we can print in color, we need this property to print the advertisement labels that must be bright in colour. Other pros are that the labels will not fade over time, this method assures resistance against high contact, chemicals and extreme temperatures, and it provides high quality printings.

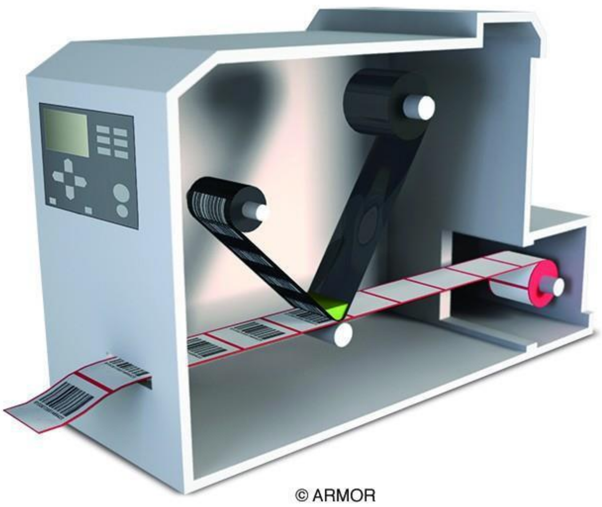


Figure 28 - Direct Thermal Printer

Which kind of printing method shall we use?

The following table displays the capabilities of each printing method and can indicate the best option to begin working with:

	Ink	Direct thermal	Thermal transfer
Maintenance replacements	Yes	No	Yes
Colour	Yes	No	Yes
Fades over time?	Yes	Yes	No
Affordable?	Yes	Yes	No
Size limitations?	No	Yes	Yes
Printing quality	Normal	Normal	High quality

Table 4 - Printing Method Comparison

We must focus on the two most important requirements that the company has given, after evaluation, the most efficient printer would be a thermal transfer option. Despite it having some maintenance problems, such as changing the ribbons, it is an expensive technology and the size of the labels is limited. These problems can be easily solved; the replacements will not be more than one every 24 hours, probably once each week, and for the size limitations, the stickers of this project fit inside the label size range of the actual most common printers of the market.

As all the printers have pros and cons, the final decision will be made by the company. To help them with the decision, some sector leading companies are suggested below:

Figure 29 - Printer Companies

In addition, we would like to recommend another reliable industrial label printer company, GODEX.

They have several models for programmable industrial label printers with excellent technical specifications. In the appendix it is shown the series ZX1x00i, that fits excellent with the project requirements.

Figure 30 - GODEX Labelling Machine

5 Layout Simulations

5.1 Specifications of the Space

The exact size of space that the team must work with is unclear because ABB are very lenient. However, the floor plans indicate that we have roughly 7.2m x 4.4m, equating to 31.68m^2. This is quite a small space to work in as there are a lot of components, robots and conveyors to consider. The concept sketches of this are shown below and pros and cons discussed.

5.2 Concept 1

The plan view for the first design is below:

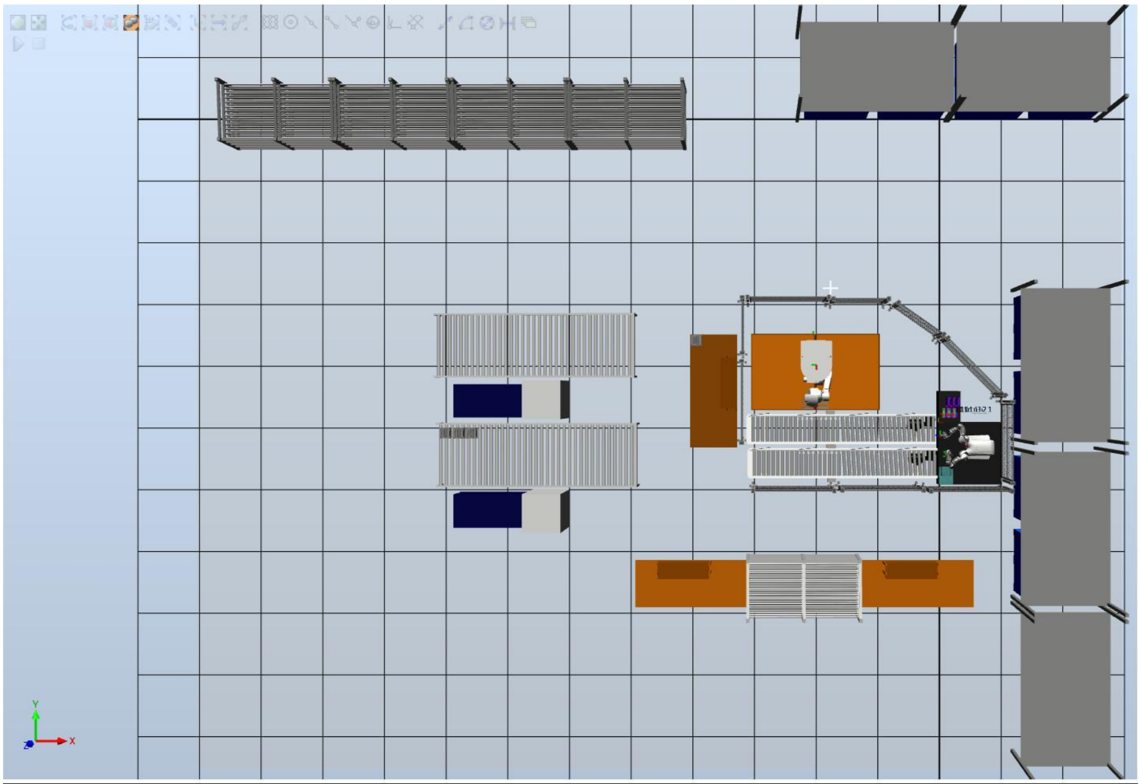


Figure 31 - Plan View of Layout Concept 1

Next picture shows the different elements that form the layout distribution:

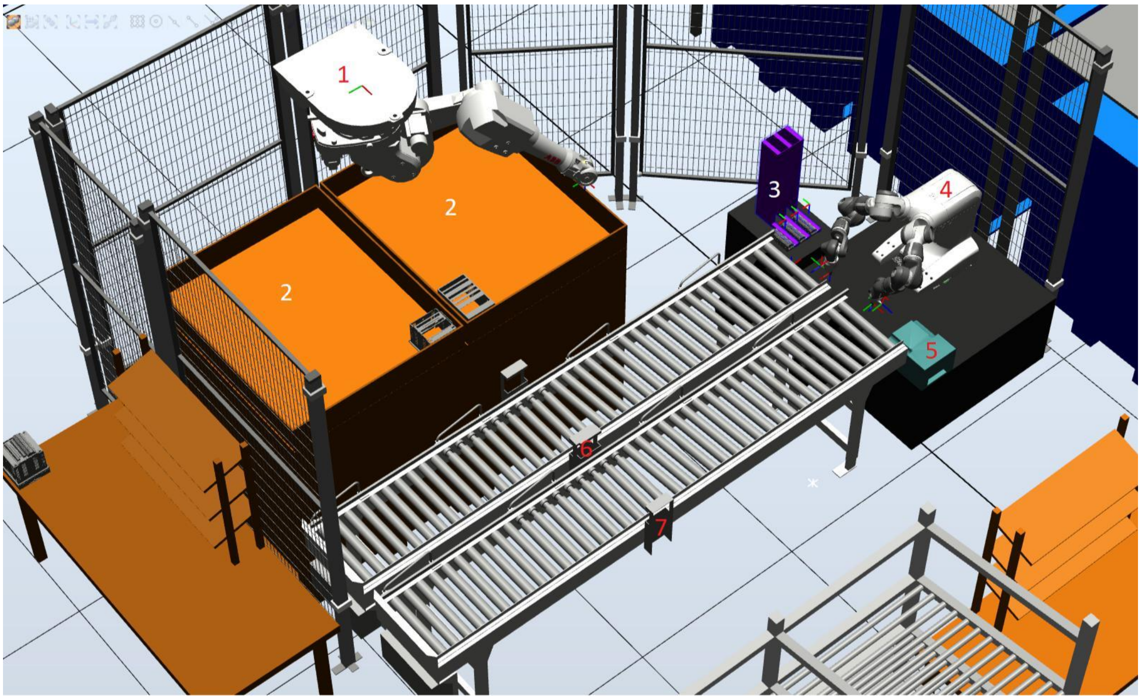


Figure 32 - Isometric View of Layout Concept 1

The formation of this distribution consists on the roof robot from ABB (1), two stands for the relay boxes (2), the support for the connectors (3), YuMi (4), the labelling machine (5) and finally the input conveyor belt (6) and the output conveyor belt (7).

The area is protected by security fences due to the big robot is not a collaborative robot, therefore it cannot interact with humans.

The steps of this layout are described below:

1- Roof robot moves boxes from the storage to the input conveyor belt. Boxes will be approached to YuMi by the conveyor belt.
2- YuMi will work with boxes. Connectors and stickers will be placed.
3- Finally, YuMi moves the box to the output conveyor belt. The conveyor belt will then bring the case to the next working station.

5.3 Concept 2

The picture below shows the plan view of the layout:

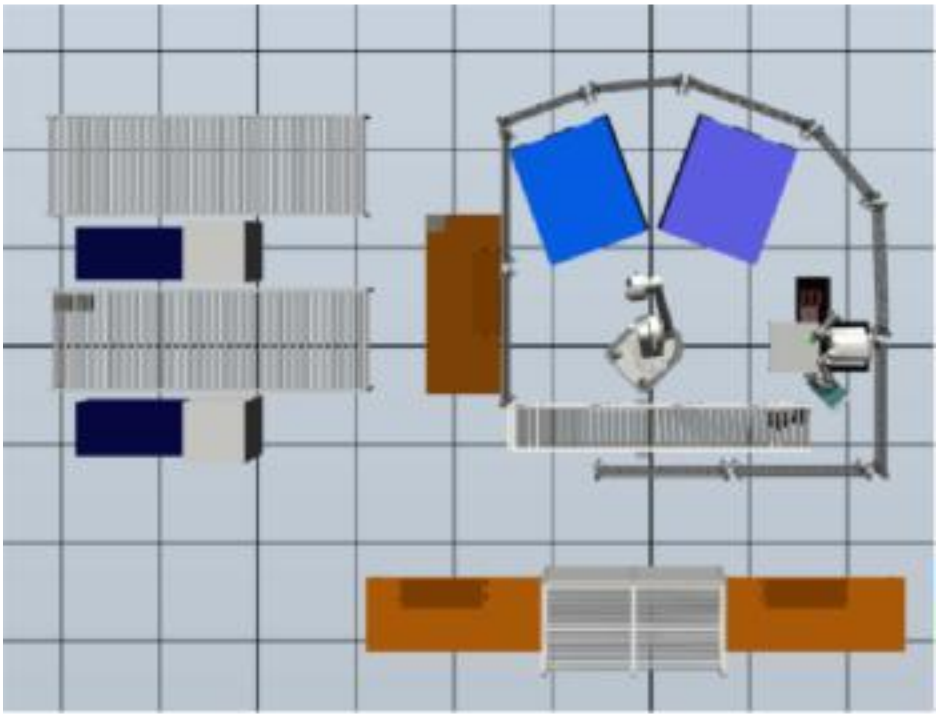


Figure 33 - Plan View of Layout Concept 2

In the whole automated process, another robot will be used. The model is IRB 1600, it is also from ABB. The bigger robot has simplified the project a lot, mostly in terms of time and precision.

IRB 1600 is a general purpose industrial robot with a payload from 6 to 10 kg depending on the model.

Figure 34 - ABB IRB 1600 Industrial Robot. Source: ABB

The key specifications can be seen on the table below:

Robot versions	Handling capacity (kg)	Reach (m)
IRB 1600 - 6/ 1.2	6	1,2
IRB 1600 - 6 / 1.45	6	1,45
IRB 1600 - 10 / 1.2	10	1,2
IRB 1600 - 10/ 1.45	10	1,45

Table 5 - Technical Data for the IRB 1600 Industrial Robot. Source: ABB

The selected robot that will be used in the layout and the project is the IRB 1600 - 6 / 1.45 with a payload of 6 kg and a range of 1,45 m.

In depth details of the IRB 1600 robot can be found on the appendix section.

Below is the order for the different steps of the system, the screenshots have been taken from the Robotstudio simulation software and the video of the process is available in the final presentation file:

Figure 35 - Layout Concept ft. IRB 1600

This is the initial position of the loop, the previous picture shows the different elements that compound the working station as YuMi (1), a working table for YuMi (2), a support for the connectors (next to YuMi at his right, red in colour) (3), the labelling machine (next to YuMi at his left, blue in colour) (4), the big robot IRB 1600 (5), two case stores that can be easily replaced by ABB workers (6) and finally the conveyor belt to bring the final product to the next working station (7). All of them are placed in strategical points to optimize the process.

The area is protected by security fences due to the big robot not being a collaborative robot, therefore it cannot interact with humans.

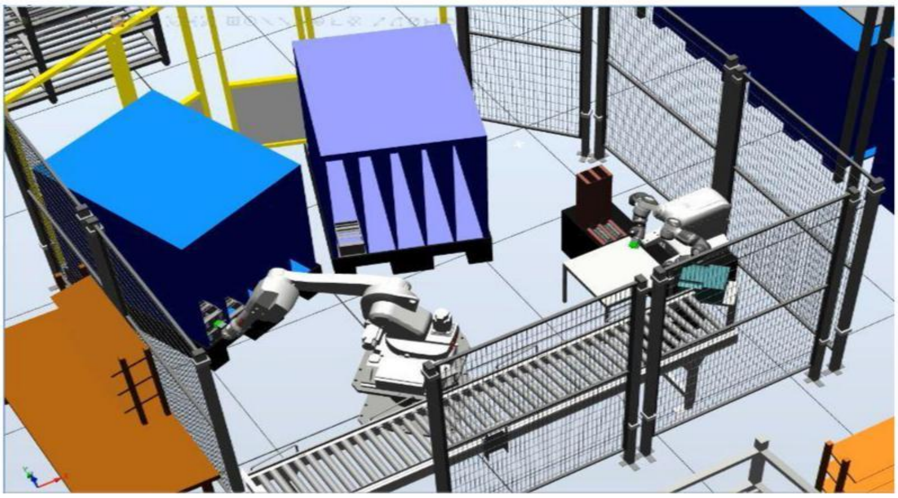


Figure 36 - IRB 1600 Taking Casings

To start with the process loop, cases are taken by the big robot, IRB 1600, to approach them to YuMi.

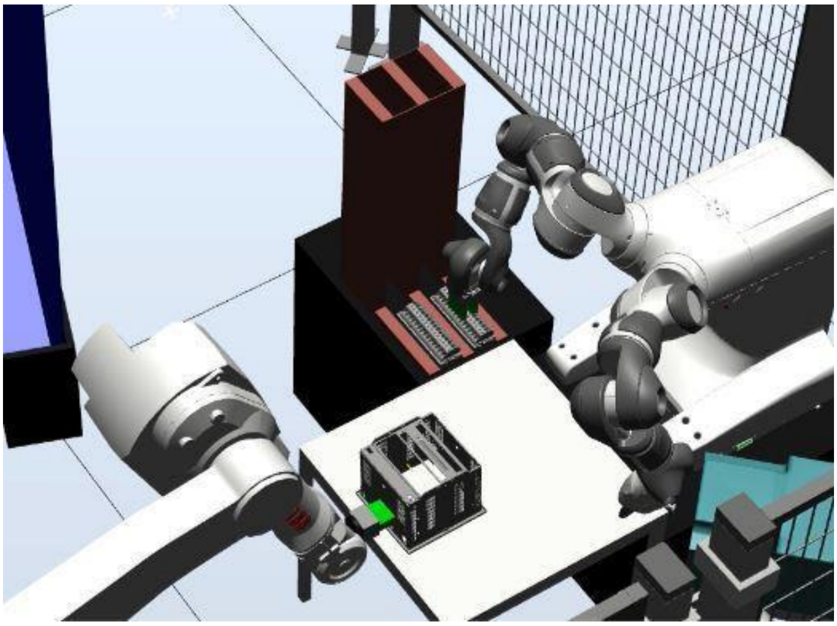


Figure 37 - IRB 1600 Placing Casing to YuMi

Once the cases have been brought beside YuMi, he prepares himself to pick the first outside sticker from the labelling machine with the left hand while the right hand picks the connector to place it.

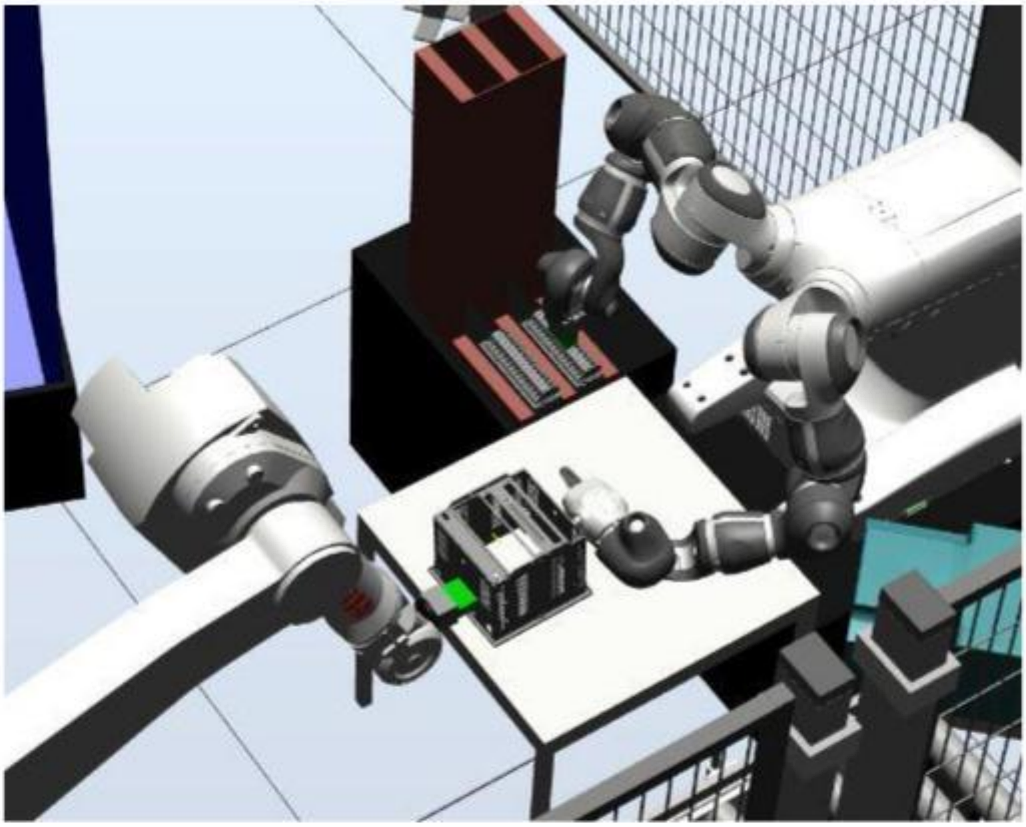


Figure 38 - YuMi Working with Casing

YuMi sticks the first outside sticker.

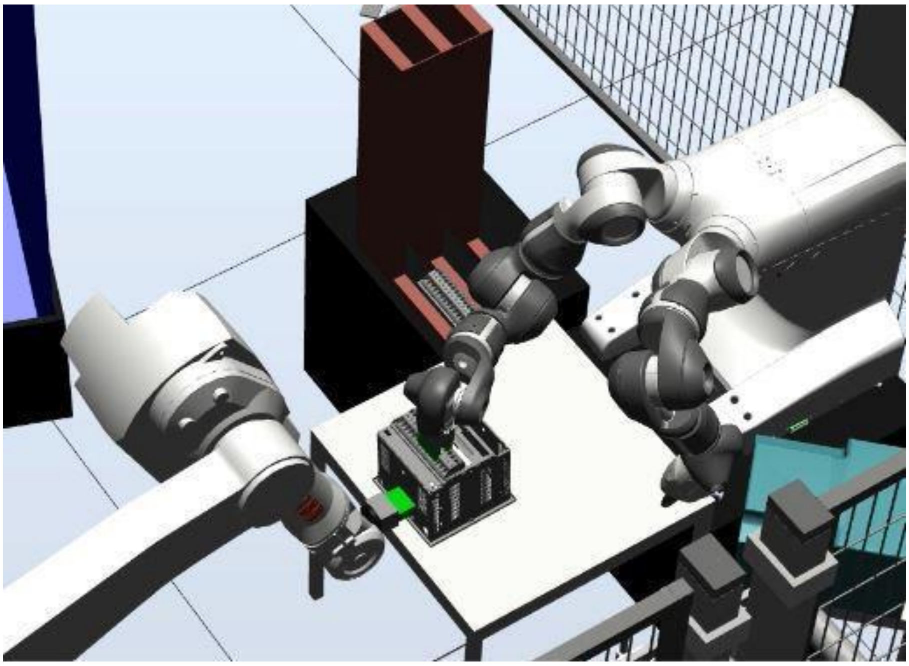


Figure 39 - YuMi Placing Sticker on Casing

Afterwards, YuMi places the connector. At the same time, he picks the second outside sticker from the labelling machine to stick it as it is shown in the picture below:

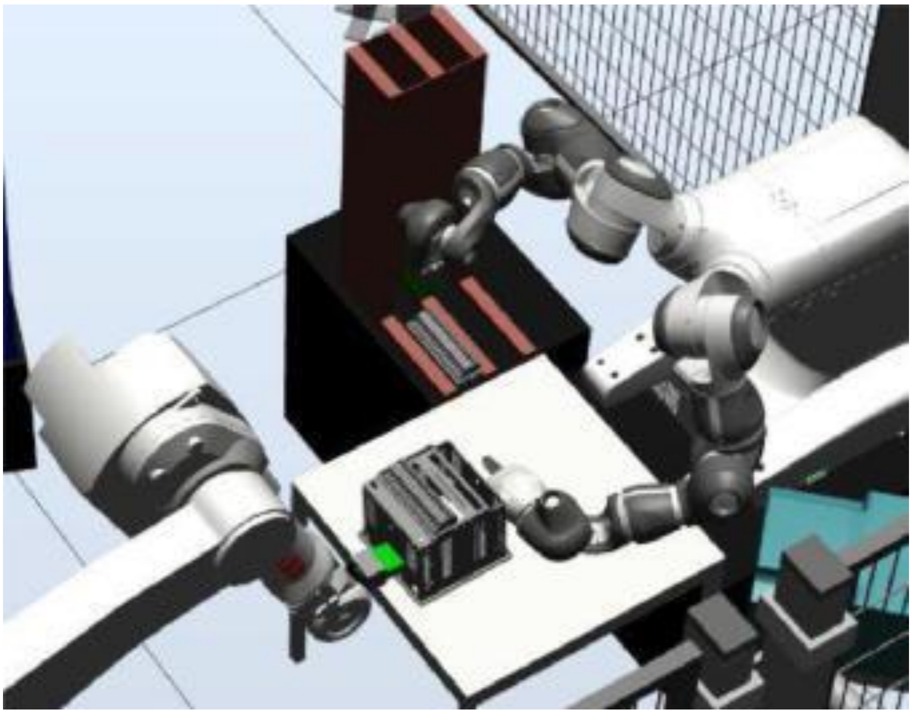


Figure 40 - YuMi Placing Outer Sticker

Between the last step and the next one YuMi picks the last sticker.

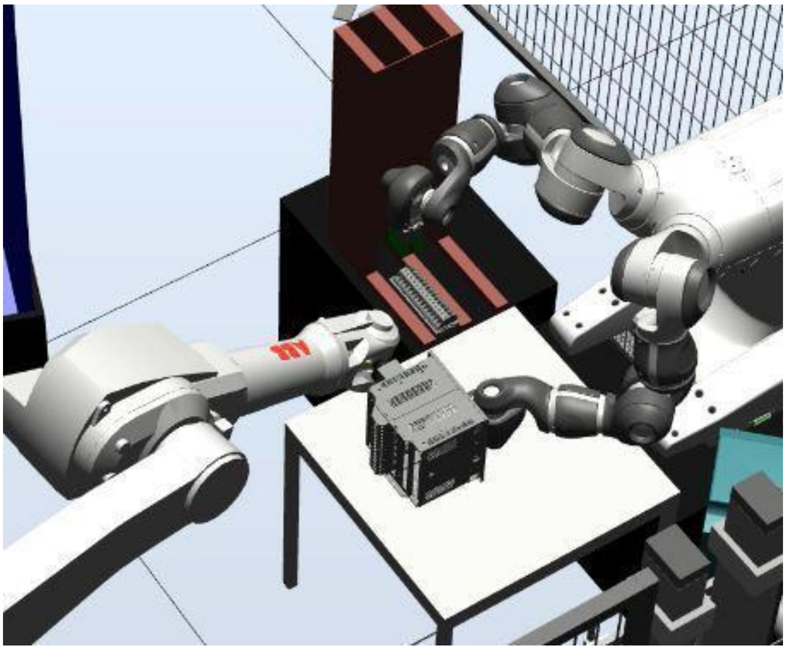


Figure 41 - YuMi Placing Inner Sticker

This last sticker is to be placed inside the box. With this last step the process is almost finished, the last thing to do is to place the case in to the box fixture by IRB 1600 and bring it to the conveyor belt using YuMi, while the IRB 1600 returns to the initial position to be ready to start the loop again.

Finally, to conclude with this mounting stage, the conveyor belt will bring the relay boxes to the next working station to continue with other mounting processes and security tests.

5.4 Confirmed Design of Layout

With careful evaluation and discussion within the team, it has been decided to implement layout concept 2 for use in the production line. This is due to the big robot being capable to hold the casing in place while the YuMi places the sticker onto the inner and outer walls. Using this method, it eliminates the need to turn or flip the box using YuMi and therefore results in a more smooth, simple and efficient process.

6 Manual Layout Process

As we do not possess some of the elements for this project such as; IRB 1600, the labelling machine, the store for connectors and the conveyor belts, all of them were simulated by different methods.

Below, the steps of the process that were simulated:

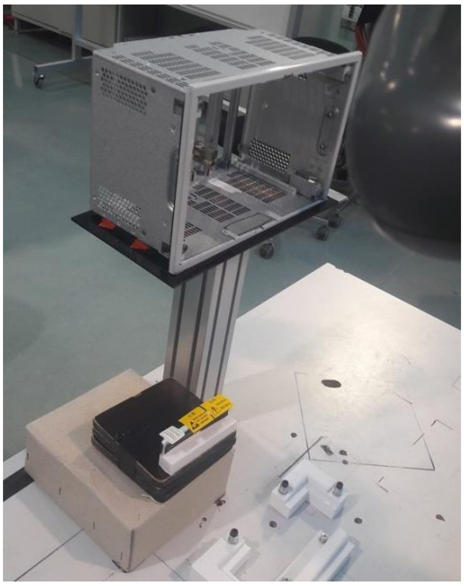


Figure 42 - Casing, Stand and Labels

The real process starts with the IRB 1600 approaching the relay boxes to YuMi for the robot to operate with them. This was simulated with a support for the box to be designed.

To simulate the labeling machine a support for the labels was constructed, labels are stuck in the support without the back paper.

The box is put in the support manually. The same happens with the stickers.

For the connectors a support is simulated too. Also, the connectors are placed by hand:

Figure 43 - Connector Place

Steps for the process:

To start, YuMi picks the first inside sticker and sticks it in the relay box. This step is repeated for the second sticker:

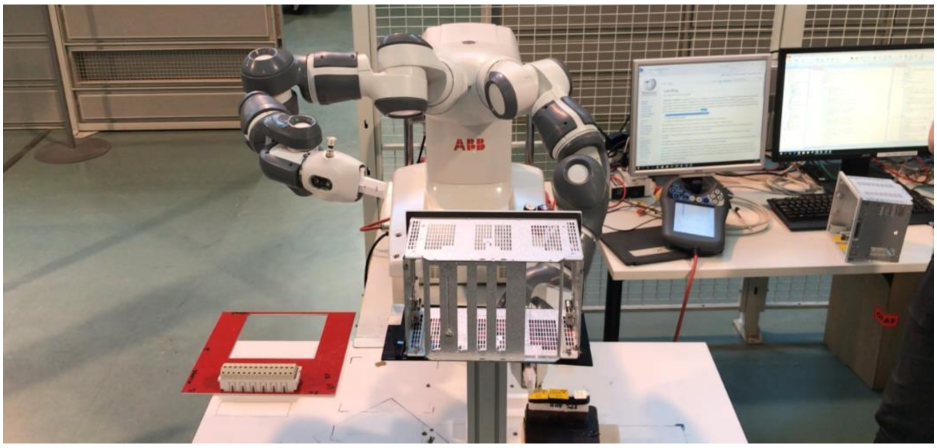


Figure 44 - YuMI Picking Sticker

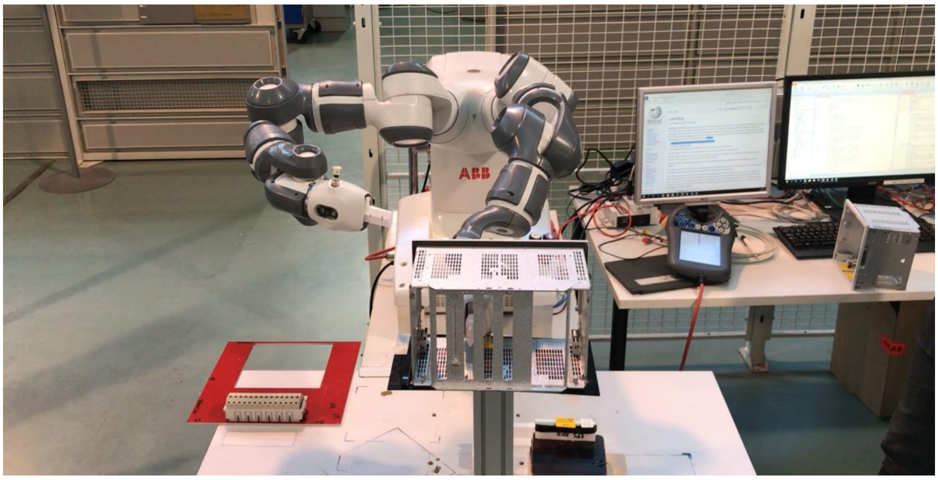


Figure 45 - YuMi Placing Sticker

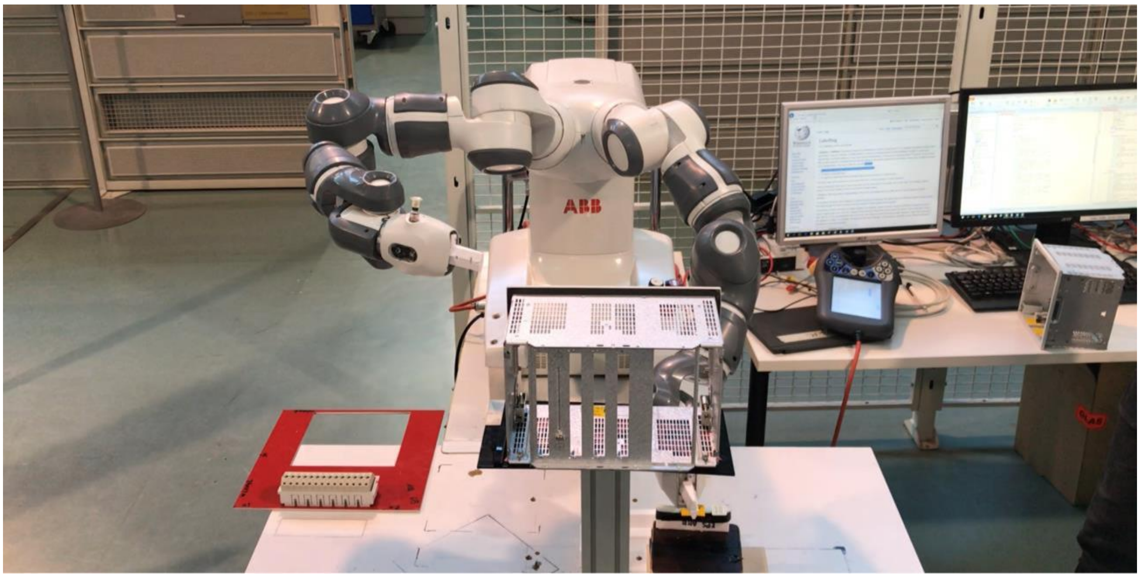


Figure 46 - YuMi Retrieving Second Sticker

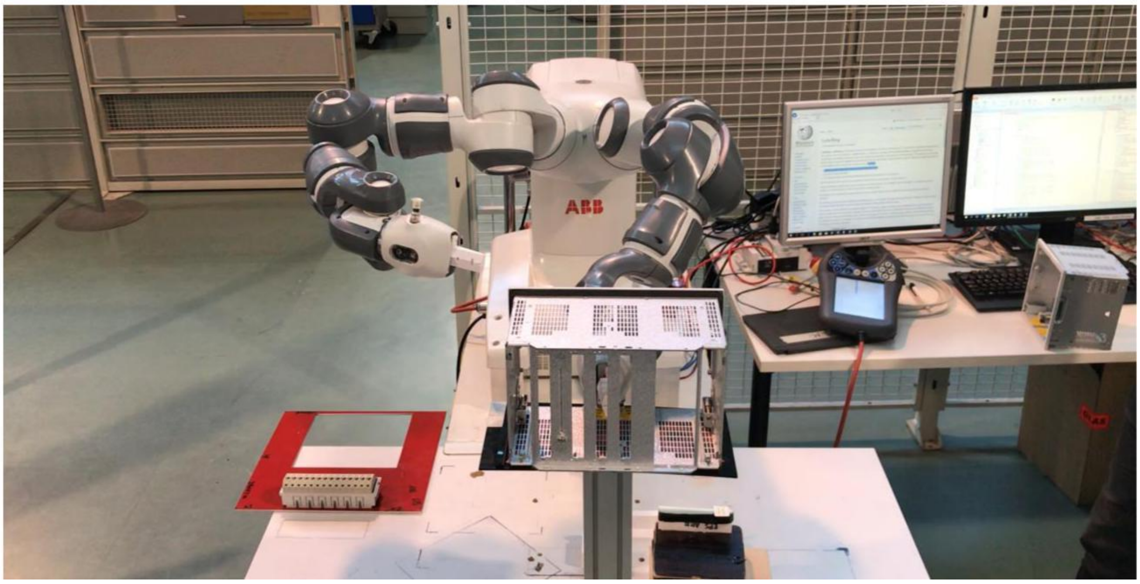


Figure 47 - YuMi Placing Second Sticker

The same happens with the outside sticker, YuMi picks it and places it in the correspondent position:

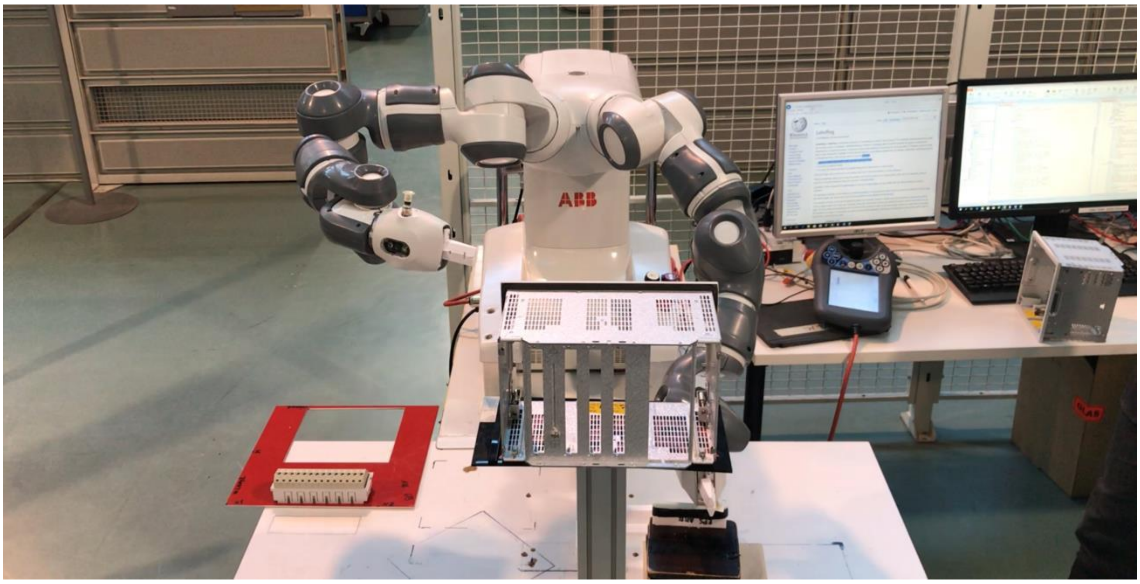


Figure 48 - YuMi Picking Outside Sticker

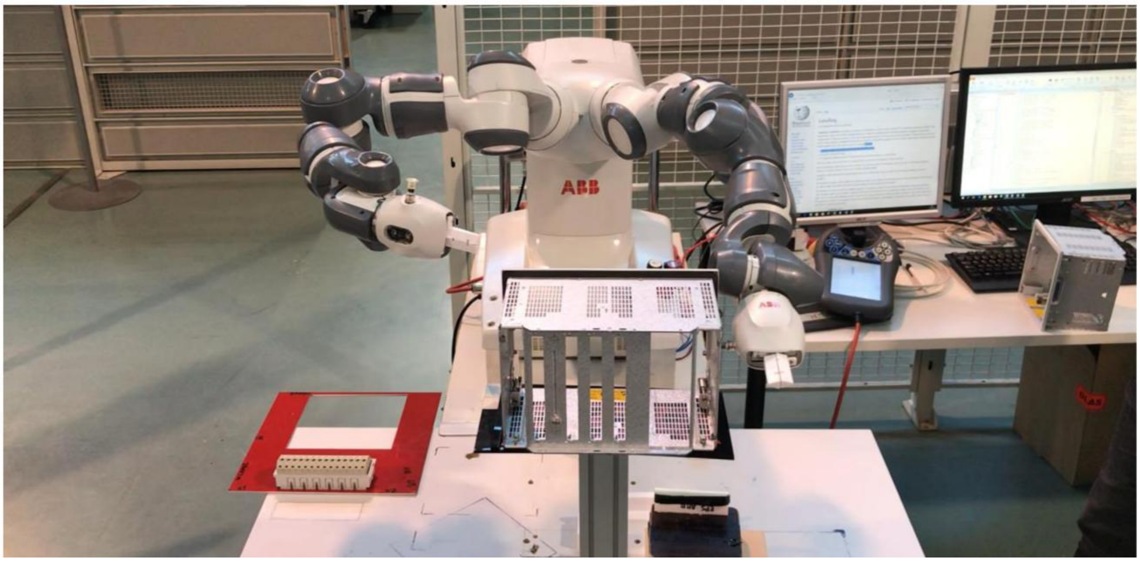


Figure 49 - YuMi Placing Outside Sticker

Once the stickers are stuck, YuMi begins with the connector placement. For that, it is needed to place the box manually from the support that simulates the IRB 1600 to the box jig:

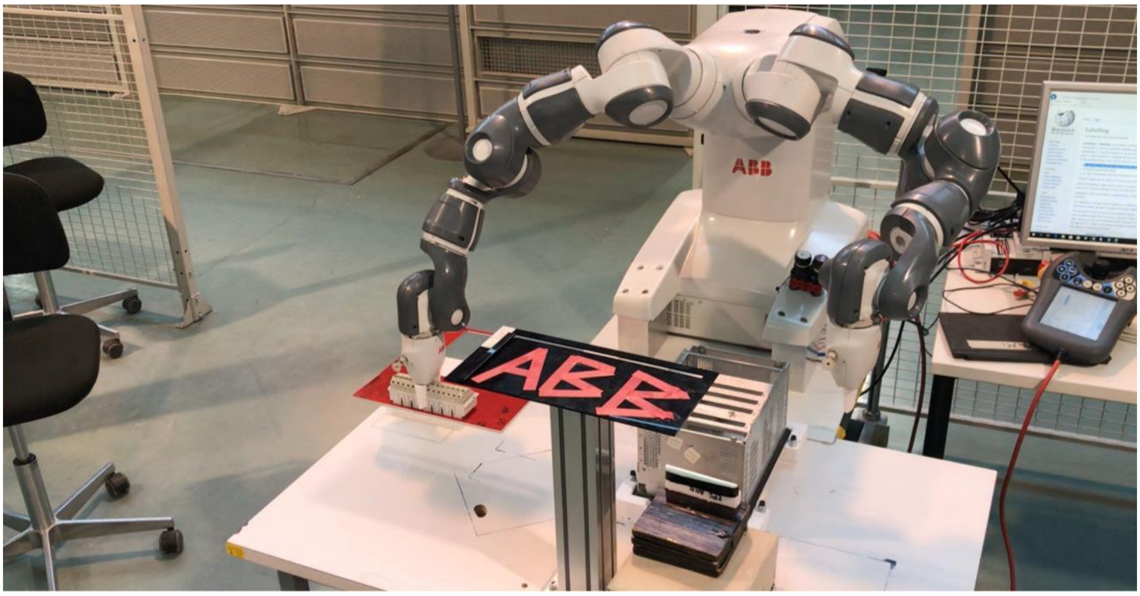


Figure 50 - YuMi retrieves Connector

Figure 51- YuMi Places Connector

Finally, YuMi pushes the connector into the relay box:

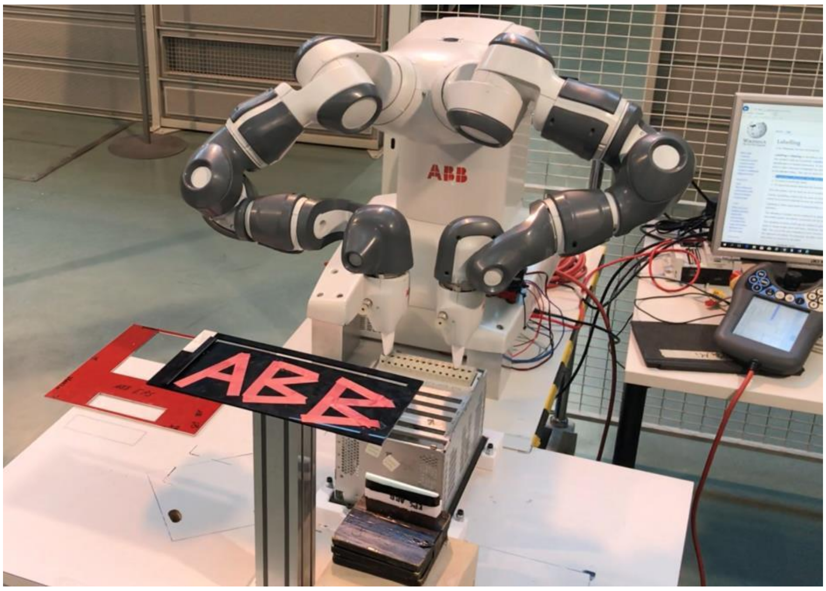


Figure 52 - YuMi Secures Connector into Relay

To conclude the process, the box is moved onto the conveyor belt. As we do not have the conveyor belt readily available, the box is left on the table to simulate this action:

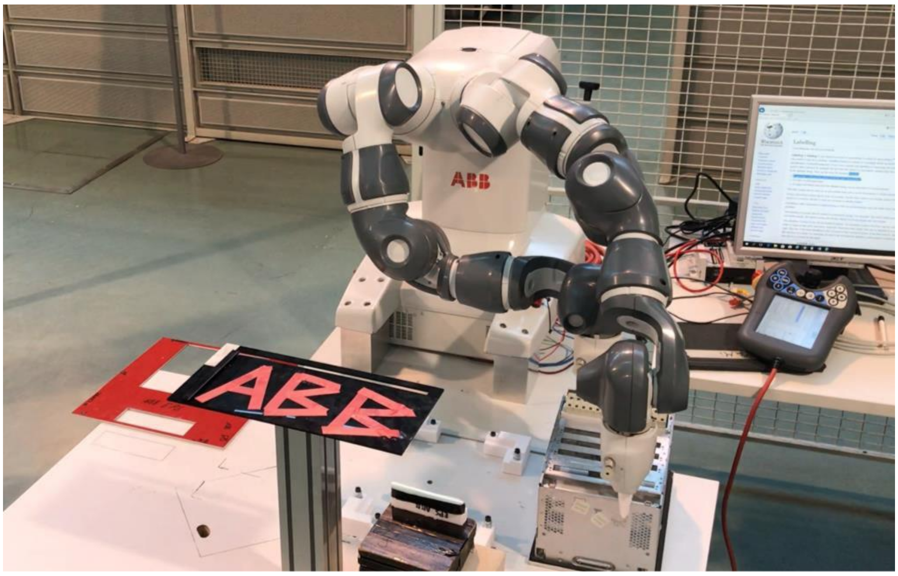


Figure 53 - YuMi Moving the Casing to the Conveyor Belt

7 Design Software

7.1 Autodesk Inventor

The Autocad software is highly significant in the design process of this project. It ensures dimensions are correct and is easily projected from a 3D shape into 2D drawings. Because of using this software, all members of the team were able to visually see what they were going to be manufacturing as well as the completed product, thus having a clear aim to finish with. Inventor will be used for the design of the jig as well as the grippers for YuMi. This software was chosen by the team as the parts can be easily transported into the 3D printing machine which is how the manufacture will take place. Furthermore, the team members have had experience using this software which prevents confusion in changing program and results in a quicker production. The software is also very beneficial when sharing with the others in the team members – if one member is absent then the other members can carry on doing the work from the same design. The designs are always to the specifications and dimensions can be easily seen on inventor. The program will be used throughout the full project by all team members. (Autodesk.eu, 2018)

7.2 3D Printing

3D is becoming more and more common in the production world today. It is due to its ease of manufacturing objects and how simply 3D modelling a component on inventor or another design software can be manufactured using this method. The system can then then be left for the time it takes to print, and the finished product can be retrieved once the time is complete. The time taken depends solely on the volume of the object, it can take anything from an hour to days. (Williams, 2018)

In this project, 3D printing will be used for the manufacture of the grippers and the jig. The printers available to the team are of a small size and thus the team should be cautious of the dimensions of the products being manufactured – especially the jig which can be of a large volume.

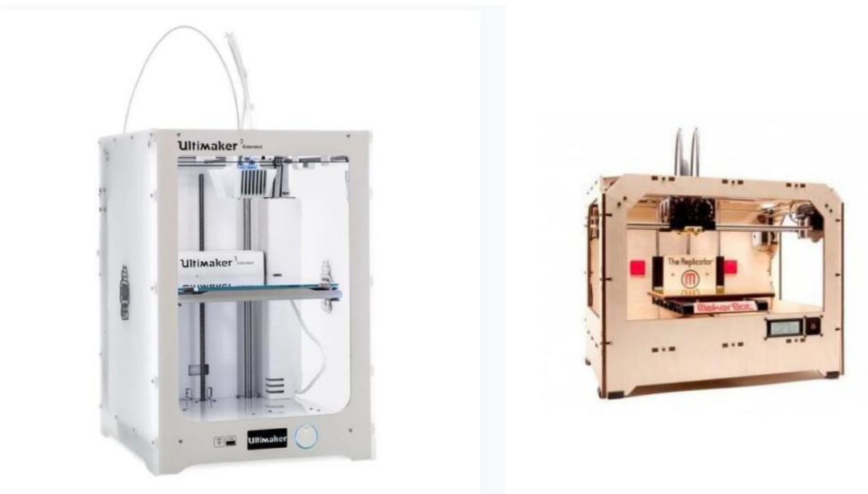


Figure 54 - 3D Printing Machines

Plastics are the most commonly used material for printing although metals, ceramics and resin can also be used. The readily available material that can be used in the lab is PVC and ABS plastics, therefore, to save costs and time efficiency, this is the material that will be used when printing the grippers and the jig. There are many positive properties of ABS plastic filament, that also results in this material being the perfect one to use for these components, such as; hardness, toughness and strength. (Tinkercad Blog, 2018)

7.3 Robotstudio

Robotstudio is a computer visualisation program, which was designed and developed by the company ABB. With this program the operator can control the YuMi and other industrial robots. In Robotstudio the high-level programming language Rapid is used.

It allows for easy programming of robots and provides simulations so that the customer or clientele can visualise how the robots or production line will work before purchase or renovation. The user draws the layout in the software and uses a coding method that generates the movement of the robot being used. This is how the team will program the YuMi robot. (New.abb.com, 2018)

The team members have not had any experience with this software so many tutorials will be watched, and research undertaken to ensure that the final output is to a high enough standard and that YuMi will do what is required of the team. Robotstudio enhances speed of production, as well as making the coding easy to change if required. Additionally, it can be easily transported over the internet and so displaying of the material can be completed with international clients and other ABB bases. (New.abb.com, 2018)

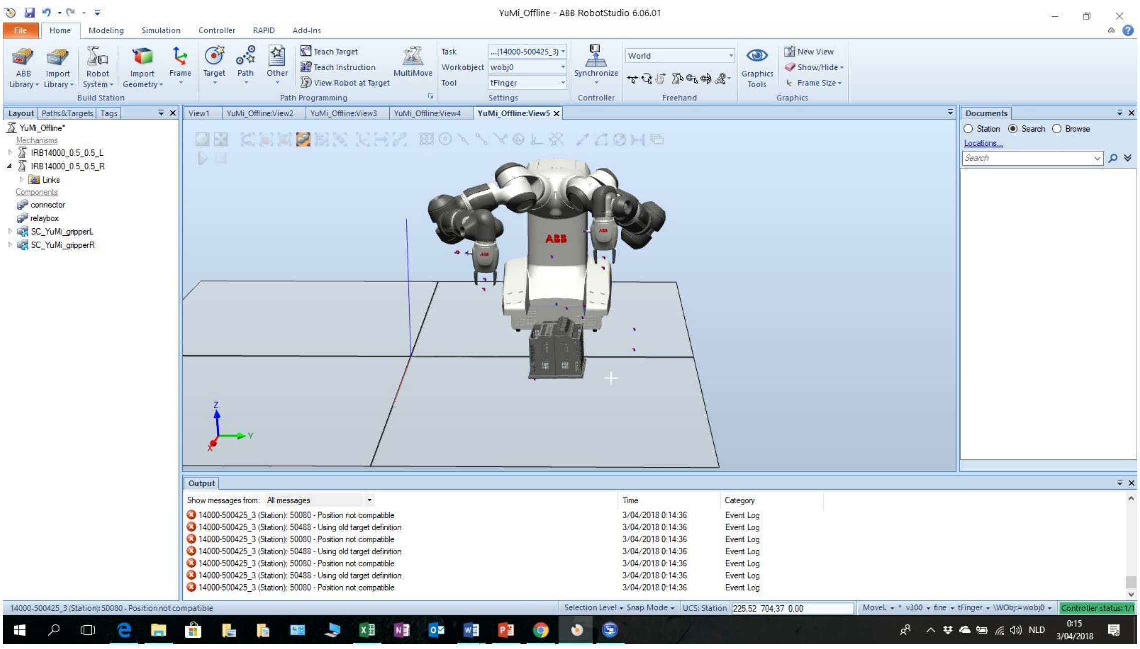


Figure 55 - YuMi Drawing in Robotstudio

7.3.1 Flex Pendant Explanation

WHAT IS A FLEX PENDANT?

The ABB's FlexPendant is a tool used for manipulating the YuMi robot. The main uses of FlexPendant are: executing programs, modifying programs, controlling the arms via joystick, etc.

It has been conceived for working in industrial environments. On top, the touch screen is resistant to water, oil and welding splashes.

Figure 56 - FlexPendant. Source: ABB

The FlexPendant is connected to YuMi by a 10 m cable and a connector plugged on the right side of the robot on the XS4 port.

Figure 57 - FlexPendant Connection. Source: Own Elaboration

DESCRIPTION OF FLEX PENDANT PARTS

The parts and its functions are listed below:

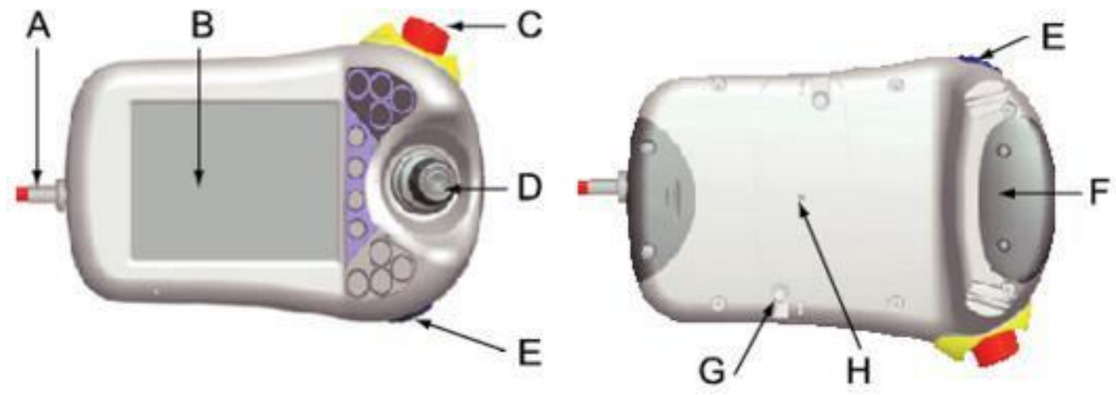


Figure 58 - FlexPendant Parts. Source: ABB

Part	Name and function
A	**Connector**
B	**Touch screen: For operating the FlexPendant, tap the screen with the stylus pen**
C	**Emergency stop button**
D	**Joystick: Used for moving the robot arms (jogging)**
E	**USB port: A USB memory can be connected to read or save programs**
F	**Enabling device: The FlexPendant is held with the hand in this part**
G	**Stylus pen: For liberating the pen, pull it from the holder**
H	**Reset button: This button only resets the FlexPendant (YuMi robot will not be reset)**

Table 6 - FlexPendant Parts. Source: Own Elaboration

DESCRIPTION OF FLEX PENDANT BUTTONS

The functions of the main FlexPendant buttons are described under the figure. There are 4 buttons of the FlexPendant that can be personalized.

Figure 59 - FlexPendant Buttons. Source: ABB

Button	Function
A	**Programmable key 1. User can define the function**
B	**Programmable key 2. User can define the function**
C	**Programmable key 3. User can define the function**
D	**Programmable key 4. User can define the function**
E	**Select mechanical unit**
F	**Toggle motion mode, reorient or linear**
G	**Toggle motion mode, axis 1-3 or axis 4-6**
H	**Toggle increments**
J	**Step BACKWARD button. Executes one instruction backward as button is pressed**
K	**START button. Starts program execution**
L	**Step FORWARD button. Executes one instruction forward as button is pressed**
M	**STOP button. Stops program execution**

Table 7 - FlexPendant Buttons. Source: ABB

7.3.2 Coding

DECLARATION OF THE VARIABLES

In the first part of the program we declared all variables and determined which data type it should have.

```
OB_L/MainModule*
1   MODULE MainModule
2       PERS tasks taskslist1{2}:=[["T_ROB_L"],["T_ROB_R"]];
3       VAR syncident sync1;
4       VAR syncident sync2;
5       VAR syncident sync3;
6       VAR syncident sync4;
7       CONST robtarget pHome:=[[534.22,122.96,379.18],[0.526311,0.538683,0.564936,-0.337144],[-1,0,1,11],[-170.038,9E+09,9E+09,9E+09,9E+09,9E+09]];
8       CONST robtarget pApproachPlace:=[[534.42,125.99,64.67],[0.00704159,-0.999774,0.0158698,-0.0122567],[-2,1,1,11],[170.199,9E+09,9E+09,9E+09,9E+09,9E+09]];
9       CONST robtarget pHomeConnector:=[[480.72,299.50,97.18],[0.000941248,0.99941,-0.0058208,0.0338478],[-1,1,1,11],[166.841,9E+09,9E+09,9E+09,9E+09,9E+09]];
10      PERS loaddata Maxload:=[0.20,[0,0,1],[1,0,0,0],0,0,0];
11      PERS loaddata Minload:=[0.02,[0,0,1],[1,0,0,0],0,0,0];
12      PERS loaddata Connectorload:=[0.3,[0,0,1],[1,0,0,0],0,0,0];
13      CONST robtarget pHomeLabeling:=[[410.01,211.92,379.20],[0.626518,0.28075,0.685098,-0.243504],[-1,-1,2,1010],[172.821,9E+09,9E+09,9E+09,9E+09,9E+09]];
14      CONST robtarget pApproachPickLabel1:=[[456.40,145.43,168.32],[0.682835,0.069835,0.725676,-0.0474824],[-1,-2,2,11],[-169.715,9E+09,9E+09,9E+09,9E+09,9E+09]];
15      CONST robtarget pApproachPickLabel2:=[[456.40,175.43,168.32],[0.682835,0.069835,0.725676,-0.0474824],[-1,-2,2,11],[-169.715,9E+09,9E+09,9E+09,9E+09,9E+09]];
16      CONST robtarget pPickLabel1:=[[575.01,91.75,163.20],[0.714789,0.0294322,0.698525,-0.0165228],[-2,-2,2,11],[-175.139,9E+09,9E+09,9E+09,9E+09,9E+09]];
17      CONST robtarget pPlaceLabel2:=[[769.60,10.93,327.12],[0.697766,0.0264131,0.715828,-0.0039206],[-2,-2,2,11],[-133.863,9E+09,9E+09,9E+09,9E+09,9E+09]];
18      CONST robtarget pStickLabel2:=[[776.11,17.92,292.81],[0.680013,0.0217463,0.732851,0.00625558],[-2,-2,2,11],[-133.808,9E+09,9E+09,9E+09,9E+09,9E+09]];
19      CONST robtarget pStickLabel1:=[[769.09,-18.31,295.44],[0.68474,0.0342238,0.72672,-0.0428727],[0,-2,2,0],[-120.299,9E+09,9E+09,9E+09,9E+09,9E+09]];
20      CONST robtarget pApproachPlaceLabel2:=[[594.72,10.19,327.77],[0.698905,0.0254903,0.714754,-0.00297448],[-1,-2,2,11],[-133.92,9E+09,9E+09,9E+09,9E+09,9E+09]];
21      CONST robtarget pPickLabel2a:=[[709.21,120.05,159.45],[0.750701,0.00414234,0.660609,0.00516011],[-2,-2,2,11],[178.067,9E+09,9E+09,9E+09,9E+09,9E+09]];
22      CONST robtarget pPickLabel2b:=[[584.91,118.92,119.52],[0.696426,-0.0200795,0.717161,0.0163684],[-2,-2,2,11],[177.79,9E+09,9E+09,9E+09,9E+09,9E+09]];
23      CONST robtarget pApproachPickLabel3:=[[429.27,150.57,140.47],[0.708839,0.0355911,0.704454,-0.00493695],[-1,-2,2,11],[-170.886,9E+09,9E+09,9E+09,9E+09,9E+09]];
24      CONST robtarget pPickLabel3a:=[[591.60,138.37,139.24],[0.697272,0.00314585,0.713685,0.0667421],[-2,-2,2,11],[-179.907,9E+09,9E+09,9E+09,9E+09,9E+09]];
25      CONST robtarget pPickLabel3b:=[[587.17,153.82,117.15],[0.703603,0.0217157,0.710129,0.0137136],[-2,-2,2,11],[-179.358,9E+09,9E+09,9E+09,9E+09,9E+09]];
26      CONST robtarget pTurnLabel3:=[[449.47,188.18,163.22],[0.424538,-0.520465,0.489666,-0.555978],[-2,-2,1,11],[-126.417,9E+09,9E+09,9E+09,9E+09,9E+09]];
27      CONST robtarget pApproachPlaceLabel3:=[[669.10,263.22,340.00],[0.492811,-0.515145,0.474018,-0.516789],[-2,-2,0,11],[-136.817,9E+09,9E+09,9E+09,9E+09,9E+09]];
28      CONST robtarget pPlaceLabel3:=[[668.76,214.10,348.46],[0.446055,-0.524034,0.569145,-0.449997],[-2,-2,1,11],[-139.082,9E+09,9E+09,9E+09,9E+09,9E+09]];
29      CONST robtarget pPickout:=[[535.60,134.65,91.06],[0.0237871,-0.999367,-0.0174926,-0.019847],[-1,1,1,11],[-155.565,9E+09,9E+09,9E+09,9E+09,9E+09]];
30      CONST robtarget pPushConnector:=[[342.70,137.70,126.11],[0.0198139,0.998769,0.0410941,0.0194694],[-1,1,1,11],[172.972,9E+09,9E+09,9E+09,9E+09,9E+09]];
31      CONST robtarget pPlaceout:=[[535.60,389.65,91.06],[0.0237871,-0.999367,-0.0174926,-0.019847],[-1,1,1,11],[-155.565,9E+09,9E+09,9E+09,9E+09,9E+09]];
32      CONST robtarget pApproachPlaceLabel1:=[[594.15,-14.76,322.87],[0.679389,0.0402471,0.730951,-0.0502113],[-2,-2,2,0],[-121.885,9E+09,9E+09,9E+09,9E+09,9E+09]];
33      CONST robtarget pPlaceLabel1:=[[765.97,-18.89,321.99],[0.678664,0.0417587,0.731557,-0.0499518],[0,-2,2,0],[-121.195,9E+09,9E+09,9E+09,9E+09,9E+09]];
34      CONST robtarget pPickBox:=[[553.44,-142.24,65.70],[0.00236511,0.999716,0.00226172,0.0236182],[-2,1,1,11],[170.17,9E+09,9E+09,9E+09,9E+09,9E+09]];
35      CONST robtarget pPlaceBox:=[[540.82,121.99,20.17],[0.0464419,0.996707,0.00264226,0.0664245],[-2,1,1,11],[174.178,9E+09,9E+09,9E+09,9E+09,9E+09]];
36
37
```

Figure 60 - Coding Variable Declaration

In the 3rd, 4th, 5th and 6th line of the program we have a normal variable per line. Each variable contains a data value; it will keep this value even when the program is stopped or started however, when the program pointer is moved the main variable data value is lost.

In the 2nd, 10th, 11th and 12th line of the program we have the persistent variable which is the same as an ordinary variable but with one difference: a persistent variable remembers the last value it was assigned, even if the program was stopped or started from the beginning again.

Most of the other program lines are constant (CONST) which means they contain values which are always assigned in the declaration. This value can never be changed. The constant can be used in the program the same way the variable is except that it is not allowed to assign a new value. The constants are basically all of our different positions of the robot in the program.

MAIN MODULE

The procedure of the main module is the most important thing of the program. The main module contains the whole program.

```
39
40      PROC main()
41          MoveJ pHome,v1000, fine, GripperL;
42          Hand_Initialize\maxSpd:=20\holdForce:=30\Calibrate;
43          Hand_GripInward;
44          Labeling;
45          Move_Relaybox;
46          Connector;
47          Move_Relayboxout;
48        ENDPROC
```

Figure 61 - Procedure of Main Module

Program line 41 is the start of the program. The robot arm will always first go to his home position (pHome). It will move with a joint movement. This means that the movement of the robot does not have to move in a straight line.

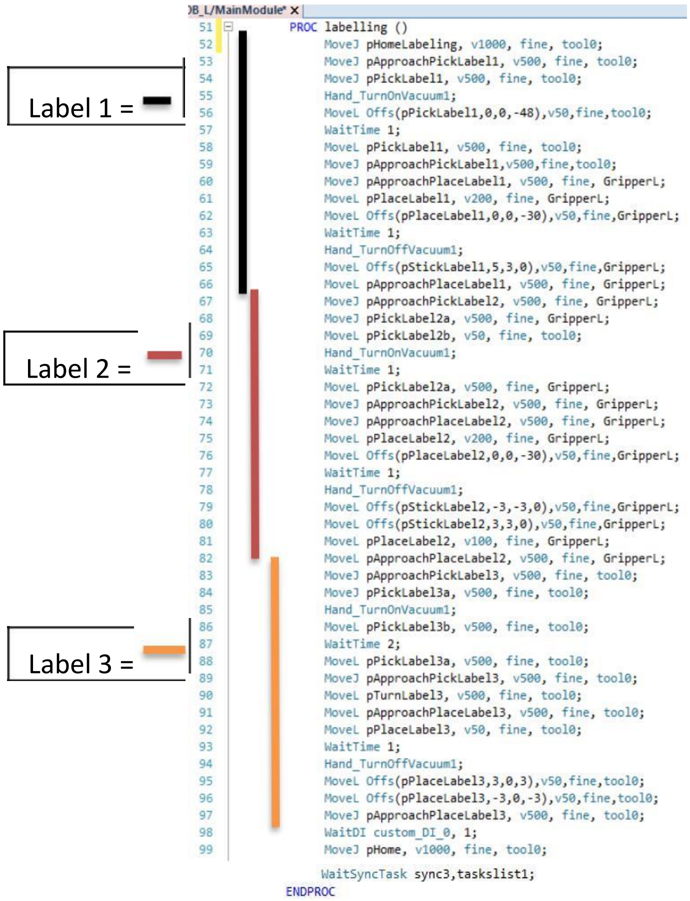
In program line 42 we initialized the maximum speed and the hold force of the hand.

Afterwards, we initialized the hand the robot closed his grippers.

In program line 44 we moved to the procedure of the labelling. When this procedure was done we moved to the next procedure "Move_Relaybox" and so on until we finished every procedure. Once we finished, the program started all over again.

LABELLING

The left arm of the YuMi robot did the labelling procedure.




```
)B_L/MainModule*
51      PROC labelling ()
52          MoveJ pHomeLabeling, v1000, fine, tool0;
53          MoveJ pApproachPickLabel1, v500, fine, tool0;
54          MoveJ pPickLabel1, v500, fine, tool0;
55          Hand_TurnOnVacuum1;
56          MoveL Offs(pPickLabel1,0,0,-48),v50,fine,tool0;
57          WaitTime 1;
58          MoveL pPickLabel1, v500, fine, tool0;
59          MoveJ pApproachPickLabel1,v500,fine,tool0;
60          MoveJ pApproachPlaceLabel1, v500, fine, GripperL;
61          MoveL pPlaceLabel1, v200, fine, GripperL;
62          MoveL Offs(pPlaceLabel1,0,0,-30),v50,fine,GripperL;
63          WaitTime 1;
64          Hand_TurnOffVacuum1;
65          MoveL Offs(pStickLabel1,5,3,0),v50,fine,GripperL;
66          MoveL pApproachPlaceLabel1, v500, fine, GripperL;
67          MoveJ pApproachPickLabel2, v500, fine, GripperL;
68          MoveJ pPickLabel2a, v500, fine, GripperL;
69          MoveL pPickLabel2b, v50, fine, tool0;
70          Hand_TurnOnVacuum1;
71          WaitTime 1;
72          MoveL pPickLabel2a, v500, fine, GripperL;
73          MoveJ pApproachPickLabel2, v500, fine, GripperL;
74          MoveJ pApproachPlaceLabel2, v500, fine, GripperL;
75          MoveL pPlaceLabel2, v200, fine, GripperL;
76          MoveL Offs(pPlaceLabel2,0,0,-30),v50,fine,GripperL;
77          WaitTime 1;
78          Hand_TurnOffVacuum1;
79          MoveL Offs(pStickLabel2,-3,-3,0),v50,fine,GripperL;
80          MoveL Offs(pStickLabel2,3,3,0),v50,fine,GripperL;
81          MoveL pPlaceLabel2, v100, fine, GripperL;
82          MoveL pApproachPlaceLabel2, v500, fine, GripperL;
83          MoveJ pApproachPickLabel3, v500, fine, tool0;
84          MoveJ pPickLabel3a, v500, fine, tool0;
85          Hand_TurnOnVacuum1;
86          MoveL pPickLabel3b, v500, fine, tool0;
87          WaitTime 2;
88          MoveL pPickLabel3a, v500, fine, tool0;
89          MoveJ pApproachPickLabel3, v500, fine, tool0;
90          MoveL pTurnLabel3, v500, fine, tool0;
91          MoveL pApproachPlaceLabel3, v500, fine, tool0;
92          MoveL pPlaceLabel3, v50, fine, tool0;
93          WaitTime 1;
94          Hand_TurnOffVacuum1;
95          MoveL Offs(pPlaceLabel3,3,0,3),v50,fine,tool0;
96          MoveL Offs(pPlaceLabel3,-3,0,-3),v50,fine,tool0;
97          MoveJ pApproachPlaceLabel3, v500, fine, tool0;
98          WaitDI custom_DI_0, 1;
99          MoveJ pHome, v1000, fine, tool0;

        WaitSyncTask sync3,taskslist1;
    ENDPROC
```

Figure 62 - Code for Labelling Procedure

The robot arm will first go to the home position of the labeling procedure.

In program lines 53 and 54 the arm will move to the position where he can pick up the label.

In program line 55 we turn the vacuum on so he can suck the 1st label.

In program line 56 the arm will move 48mm from the pPickLabel1 position in the Z direction (downwards).

Afterwards, the robot arm will wait for 1 second to make sure that the label sticks on the suction cup. In program lines 58, 59 and 60 the robot arm is moving to different positions.

In program line 61 the arm is moving to the position where he can place the label onto the safety relay box.

In program line 62 the arm will move 30mm from the pPlaceLabel1 position in the Z direction. After the program waits for 1 second the vacuum will turn off.

In program line 65 we want to make sure that the label sticks to the box by moving the suction cup 5mm in the X direction and 3 in the Y direction from position pStickLabel1.

The coding for the second and the third label is basically the same as the first but with different positions.

In program line 98 the program stops until we press button 1. Before we press the button we put the safety relay box on the table with our hands because we do not have the materials nor a robot to put it there. After we put the box on the table we press button 1 to continue the program.

When the program pointer is at program line 32 (right arm) and program line 100 (left arm) the program will continue.

```
31  PROC Labelling()
32      WaitSyncTask sync3,taskslist1;
```

```
100         WaitSyncTask sync3,taskslist1;
101     ENDPROC
```

Figure 63 – Start and End of Labelling Codes

<u>MOVE RELAY BOX INTO THE JIG</u>

In this procedure we will move, with both arms, the safety relay box from the table to the jig. In the program below we see the procedure for the left arm. The program for the right arm is the same except for the fact that it has different positions.

```
103     PROC Move_Relayboxin ()
104         Hand_MoveTo 9;
105         MoveJ pPickBox, v1000, fine, tFinger;
106         SyncMoveOn sync1,taskslist1;
107         MoveL Offs(pPickBox,0,0,-73.5)\ID:=15,v50,fine,tFinger\WObj:=wobj0;
108         WaitTime 1;
109         Hand_GripInward;
110         GripLoad Maxload;
111         MoveL Offs(pPickBox,0,0,0)\ID:=20,v50,fine,tFinger\WObj:=wobj0;
112         MotionSup \Off;
113         MoveL pApproachPlace\ID:=25, v200, fine, tFinger;
114         MoveL pPlaceBox\ID:=30, v200, fine, tFinger;
115         WaitTime 1;
116         Hand_MoveTo 8;
117         MotionSup \On;
118         MoveL pApproachPlace\ID:=35, v50, fine, tFinger\WObj:=wobj0;
119         SyncMoveOff sync1;
120         GripLoad Minload;
```

Figure 64 - Code for Moving the Safety Relay

In program line 104 the gripper opens 9mm.

In program line 105 the robot arm is going to the position where it is going to pick up the box

In program line 106 we start the sync movement with the other arm so both arms will pick up the box at the same time.

Afterwards both arms will move 73.5 mm from the pPickBox position in the Z direction.

The program will wait for one second now. After we wait for one second both grippers will move inwards and grab the safety relay box.

In program line 110 we declare the grip load. After we declare the grip load both arms will move the box to the jig.

In program lines 112 and 117 we are deactivating and activating the motion supervision function for the robot movements.

In program lines 113 and 114 both arms will place the box into the jig.

After we placed the box we go back to position pApproachPlace and we put the sync movement off and set the grip load to a minimum load.

CONNECTOR PLACEMENT

In Figure 65 we see the procedure for the left arm of the connector.

```
PROC Connector()
        MoveL pHomeConnector, v500, fine, tFinger;
        Hand_GripInward;
        WaitSyncTask sync4,taskslist1;
        MotionSup \Off;
        MoveL pPushConnector, v500, fine, tFinger;
        SyncMoveOn sync4,taskslist1;
        MoveL Offs(pPushConnector,0,0,-60)\ID:=40,v50,fine,tFinger\WObj:=wobj0;
        MoveL pPushConnector\ID:=45, v500, fine, tFinger;
        SyncMoveOff sync4;
        MotionSup \On;
ENDPROC
```

Figure 65 - Code for Connector Placement

The left arm will go to a temporary home position where it will wait to continue the program once the program pointer of the right arm reaches the waitsynctask sync 4. Before the program can do this, the grippers will move inward.

When the program pointer of the right arm reaches the waitsynctask sync 4 we deactivate the motion supervision to prevent an alarm. This is because the arm will make contact with the connector when it is going to push it in.

After we deactivate the motion supervision the left arm will move to position above the connector.

When the left arm reaches this position we will turn the synchronised movement on and push the connector with both arms at the same time.

After we push the connector inside the box both arms will go back to the position above the connector and we turn off the synchronised movement and turn back on the motion supervision.

In Figure 66 we see the procedure for the right arm.

```
55  PROC Connector()
56          MoveL pPickConnector1,v1000, fine, tFinger;
57          MoveL pPickConnector2,v1000, fine, tFinger;
58          Hand_Gripoutward;
59          MoveL Offs(pPickConnector2,0,0,-206),v50,fine,tFinger\WObj:=wobj0;
60          Hand_GripInward;
61          GripLoad Maxload;
62          MoveL pPickConnector2,v200, fine, tFinger;
63          MoveL pPlaceConnector1,v200, fine, tFinger;
64          MoveL pPlaceConnector2, v50, fine, tFinger;
65          MoveL Offs(pPlaceConnector2,0,0,-10),v50,fine,tFinger\WObj:=wobj0;
66          WaitTime 1;
67          Hand_Gripoutward;
68          MoveL pPlaceConnector1 ,v50, fine, tFinger;
69          Hand_GripInward;
70          WaitSyncTask sync4,taskslist1;
71          MotionSup \Off;
72          MoveL pPushConnector, v500, fine, tFinger;
73          SyncMoveOn sync4,taskslist1;
74          MoveL Offs(pPushConnector,0,0,-60)\ID:=40,v50,fine,tFinger\WObj:=wobj0;
75          MoveL pPushConnector\ID:=45, v500, fine, tFinger;
76          SyncMoveOff sync4;
77          MotionSup \On;
78      ENDPROC
```

Figure 66 - Coding for the Right Arm

In program lines 56 and 57 the right arm will go to the position where he picks up the connector.

After we reach this position the grippers will open.

In program line 59 the right arm will go 206mm in the Z direction. When the arm reaches this position it will close the grippers to grab the connector and set the grip load for the connector.

After we set the grip load of the connector the right arm will move to the position where he will place the connector inside the safety relay box.

When YuMi places the connector inside the box it will wait for one second and open the gripper of the right arm.

After the connector is placed the right arm will go back to the position above the connector; the grippers will close and then it will send a signal to the procedure of the left arm so the program can continue.

Before the program puts on the synchronised movement it will turn off the motion supervision.

After we put on the synchronised movement the program of the left and the right arm are the same.

At the end of the procedure we turn on the motion supervision and turn off the synchronised movement.

MOVE SAFETY RELAY OUT OF JIG

In Figure 67 we see the procedure of the left arm for moving the safety relay box out of the jig onto the conveyor belt.

```
136     PROC Move_Relayboxout ()
137         MotionSup \Off;
138         WaitSyncTask sync2,taskslist1;
139         Hand_MoveTo 7;
140         MoveL pPickout, v500, fine, tFinger;
141         WaitTime 1;
142         SyncMoveOn sync3,taskslist1;
143         MoveL Offs(pPickout,0,0,-78)\ID:=50,v50,fine,tFinger\WObj:=wobj0;
144         Hand_GripInward;
145         GripLoad Maxload;
146         MoveL pPickout\ID:=55, v200, fine, tFinger;
147         MoveL Offs(pPickout,0,255,0)\ID:=60,v50,fine,tFinger\WObj:=wobj0;
148         MoveL Offs(pPlaceout,0,0,-78)\ID:=65,v50,fine,tFinger\WObj:=wobj0;
149         Hand_MoveTo 7;
150         MoveL pPlaceout\ID:=70,v50,fine,tFinger;
151
152        SyncMoveOff sync2;
153        MotionSup \On;
154     ENDPROC
155 ENDMODULE
```

Figure 67 - Coding for Moving Relay Box to Conveyor Belt

In the start of the procedure we turn off the motion supervision to prevent the collision detection. Afterwards, the program will wait until the program pointer of the right arm reaches the same waiting task.

After the program pointer reaches the waiting task the gripper will open 7 mm.

In program line 140 the left arm will go to the position above the box where it will pick it up. Then the program will wait for one second and turn the synchronised movement on.

Afterwards both arms will move 78mm into the Z direction from the position pPickout.

In program line 144 the grippers will move inwards to grab the box.

After we grab the box we set the grip load and pick up the box and place it on the fictional conveyor belt with both arms.

In program lines 152 and 153 we turn off the synchronised movement and then turn on the motion supervision.

After this the program is finished and will start again.

The coding for the smaller box is almost the same as the bigger box but with different positions.

8 Solution Ideas

8.1 Labelling

The main problem of labelling is the inside sticker, due to YuMi needing to flip the box to access inside it. To solve this issue, the bigger robot has been introduced, IRB 1600, which will flip the box according to YuMi's needs.

8.1.1 Sticker

This is the most difficult of the three options for the label implementation, but on the other hand, it is the most affordable. The YuMi will need to have more advanced coding for this to work and the removal of the sticker from the backing may prove to be problematic. The solution of this requires the YuMi to not get confused with the sticker and prevent the stickers from becoming attached to the robot instead of the component. The removal of the sticker from the backing would need to be swift to allow for production to run smoothly, which may be hard to develop. As a result, extensive testing will need to be undertaken to understand whether this is a task capable for the YuMi robot.

The ideas for this will require a suction cup on the hands of the YuMi, which will take the sticker from the roll and place onto the component. Thus, the YuMi never touches the sticky part of the sticker and will prevent the problems previously stated. Again, testing is compulsory to develop knowledge on whether this is an appropriate method for production. Further, permission from ABB is to be granted and equipment will need to be investigated to see if this is possible.

Figure 68 - Stickers for the Casing

8.1.2 Laser Printing

It was discussed that there could also be a laser printing solution. Again, equipment needs to be addressed to confirm if this is possible. It would be easier to conduct this; however, this could cause corrosion on the metal in later life. Or it could diminish the material properties and lead to failures when put into the machines to work. This will need to be thoroughly researched and derived if it is a safe solution to undertake.

"Pros • Best mark quality • Non-contact process with good standoff distance • Highly flexible mark; text, graphics, data codes • High speed • Permanent marks • Mark can be dynamically sized • Wide range of markable materials • Easy integration • No mark consumables • No post processing

Cons • High capital cost of marking equipment • Safety shielding required"

(Amadamiyachi.com, 2018)

8.1.3 Ink Printing

This is another method that is to be investigated. The final mark may not last a long period of time and therefore life of the component itself will need to be discovered to ensure that the mark will not fade in this time. It is cheap and fast to do and therefore the project team need to discuss with ABB whether they want a fast and cheap mark or an expensive and good quality of the mark. Once this has been derived then decisions can begin to get made so that ABB and the customer are both satisfied with the outcome. *(Amadamiyachi.com, 2018)*

Pros and cons of inkjet marking	
Pros	
•	Capable of marking fast moving parts
•	Can create color marks, UV only readable marks
•	Cheap capital equipment costs
Cons	
•	Quality of marks is low, graphics and 2D data codes are difficult to mark
•	High running costs (ink cost)
•	Solvent bases, ink consumables and cleaning fluids create environmental and health issues for both usage and disposal – specifically solvents such as methyl ethyl ketones (MEK's)
•	Part must be moving to mark
•	Printing head needs to be very close to part
•	Marks are not permanent (fade under UV, or rub off)
•	Markable materials limited by ink adhesion
•	Print heads require regular cleaning
•	Equipment is generally not reliable

Table 8 - Pros and Cons of Inkject Marking. Source: Amadamiyachi.com

8.1.4 Labelling Method Confirmation

As a team, the pros and cons of each labelling method has been discussed and evaluated. With reference to Table 9, the properties of each has been noted to easily see which method will prove the most efficient:

	Fades over time	Colour printing	Metal corrosion	Manteinance replacements	Safety cage	Affordable
Sticker	NO	YES	NO	YES	NO	YES
Lazer printing	NO	NO	YES	NO	YES	NO
Ink printing	YES	NO	NO	NO	NO	NO

Table 9 - Labelling Method Comparison

The green coded boxes are positive properties and the red are the negative ones.Tthe clear majority here shows that the sticker will be the most optimal and therefore the team have decided to proceed in using this method and will begin coding the YuMi to work in this manner.

8.2 Connector Placement

The simplest of the tasks at hand, the solutions were discussed that a conveyor belt could be used and the YuMi robot would take the connector from here and place it into the allocated slots. This will be programmed through the Robotstudio software.

A second idea is a shelf holding many of the components in place and the robot puts a connector in each of the slots before the shelf moves along and the process repeats. Once availability of material has been confirmed we can make decisions on this matter.

Another idea that the team thought of is a storage box that distributes to the YuMi. This would start with the connectors being split into the three categories, held in vertically in single file and the connector at the bottom would be pushed, using a pneumatic cylinder, onto the conveyor belt for the YuMi to collect and place into the casing.

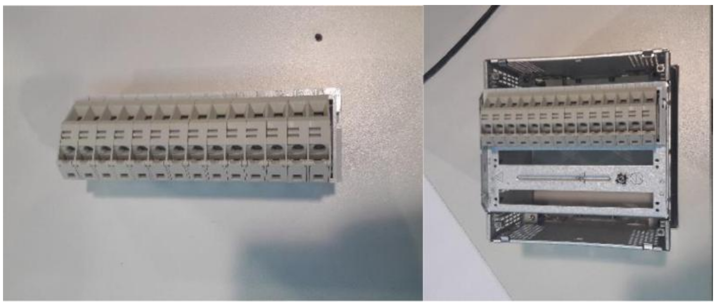


Figure 69 - Placement of Connector in Casing

8.3 Jig

It must be discussed how the component will be held in place while the cobot is working on the specific tasks at hand. This will be elaborated on once the methods have been concluded. It is essential for the component to be held in place so that there is no movement or tolerances with the work that the robot is undertaking. In addition, there is a part of the relay boxes that stands out and prevents the box to stay flat in a plain surface.

Figure 70 - Casings

For these reasons, it is required to design a jig to prevent these kinds of situations. These will be designed in inventor, if not already available, and produced in the way ABB deems fit.

9 Design of Components

9.1 Grippers

Following concept designs of the grippers were designed on Autodesk Inventor and further 3D printed.

9.1.1 Concept 1

These grippers are wide in shape and have a protruding part at the bottom that will be used for going under the lip of the connector and moving to its allocated location. Only one hand is needed for transporting the connector and thus only two were printed. Although this is an uncomplicated design, these have been tested and work well with the YuMi. However, they are unable to pick and transport the relay box in a steady manner.

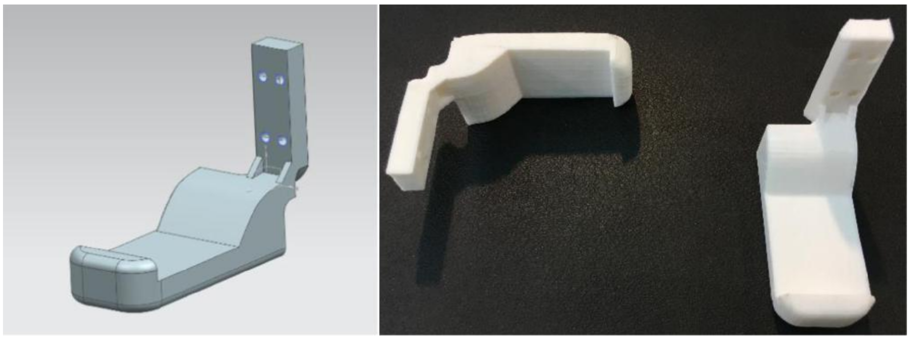


Figure 71 - Grippers for the Connector - Concept 1

9.1.2 Concept 2

The second concept sketch consists of a gripper able to pick the connectors and the box, with two slots. The upper slot in the picture below is for taking the connector, with a low tolerance. The other one is for taking the relay boxes. The programming needed for using this gripper requires exact precision and can therefore become more complicated.

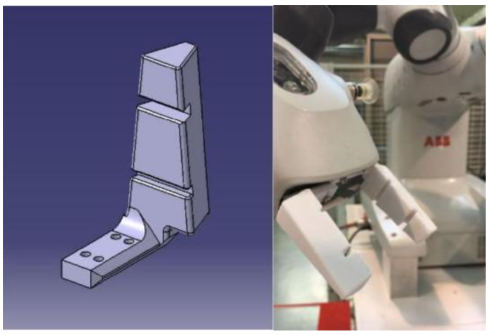


Figure 72 - Grippers for the Connector - Concept 2

9.1.3 Concept 3

These grippers were manufactured by ABB and are readily available to the team. They have proven to be efficient on a flat surface, however, with the awkward indents and lips on the components to be worked with, the flat surfaces of the grippers do not catch and thus they are unreliable and inefficient for the needs of the team.

Figure 73 - Grippers for the Connector - Concept 3

9.1.4 Testing

Testing the designs of the grippers is paramount in deciding which will work best for the task at hand. All three of the grippers were tested and the results indicated that there were advantages and disadvantages of each. Concept design 1 was too open in its grip, the casing moved in a horizontal direction when being lifted by these grippers. As a result, imprecision occurred, and the casing was never placed exactly in the same location every time.

The second concept design was excellent in transporting the connector and the casing, although more complicated to program, once this was completed there were very minute errors that occurred and very infrequently. The main issue is breakage, this is due to the 3D printing materials not being the greatest for a long-life span. When being implemented into production, it is suggested that more sustainable grippers should be manufactured in metal or wood – ensuring that the production line runs at an optimum.

The third concept design is the grippers that the team were presented with at the beginning of the project. They worked, although, not as effectively as desired. Furthermore, due to the plastic material, these grippers began to bend with time and could not hold the connector and casing without being reprinted and replaced. The short lifespan would increase costs for the ABB Company if they were to be put into production and therefore ineffective in the long term.

9.1.5 Final Gripper Design

The team has discussed the pros and cons of each of the designed grippers, and after the testing of each, concept design 2 works best with the connector placement and casing transportation. It is more difficult to program, but on the other hand, if programmed efficiently then this may prove to be the better option for taking the connector. Lifting the connector, placing it in the desired location and pushing it into the slot will be more precise. Transporting the case will be easier and more accurate due to the slots of the gripper fitting perfect into the box shape avoiding undesired movements during the transport.

9.2 Suction Cup

9.2.1 Concept 1

To collect the stickers from the labelling machine and place onto the casing has proven difficult. This design of the grippers consists of hook shape that will be collected by the right hand of the YuMi where there is a gap in the table. The suction cup will collect the label from the machine through the created hole in the table, turn 180◦ and release the vacuum onto the inner wall of the casing. It is unknown whether this will work and therefore testing is essential for this idea. Which is impossible for the team to design and produce in the time frame that is allocated, however, it may be an idea that ABB would be interested in for future production.

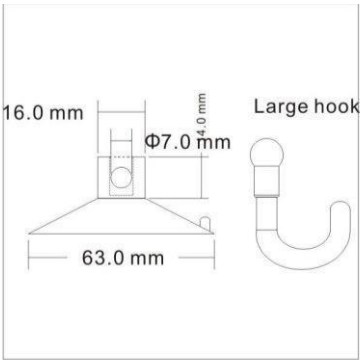


Figure 74 - Suction Cup Concept 1

9.2.2 Concept 2

For the second concept design, the team have adopted a suction cup that has been previously manufactured by ABB. The downside to this gripper is that it is very short and therefore it will be more difficult to get within the box. However, the hand of the YuMi does fit the full way inside both casings and so it will be possible when programmed correctly.

Figure 75 - Suction Cup Concept 2

9.2.3 Testing

A test was conducted on the existing suction cup to ensure that it could lift the stickers from the labelling machine (the team have manufactured an imaginary version of this) and onto the casing. The only issue that was posed was the labelling not being possible to be taken with the backing of the sticker still attached. Therefore, the labelling machine required must release the stickers without the backing plate for the YuMi to take. Other than this, the suction cups could take the sticker from the labelling machine and release it onto the walls of the casing.

9.2.4 Final Suction Design

The testing that has been undertaken indicates that the second concept design is the most efficient to work with, it is effective and is readily available to the team. Work will progress with this design and this will be programmed through robotstudio and the YuMi robot.

9.3 Jig

9.3.1 Concept 1

The first jig design is shown in Figure 76. This design will firmly hold both casings in place while the YuMi places the connector into the slot. There is a hole in the jig for where part of the casing will go through as this protrudes the box shape of the relay casing, resulting in it being unable to stand straight on a flat surface. This design has been discarded due to it requiring a lot of material to print it.

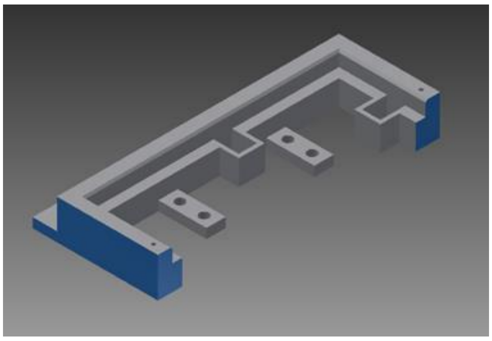


Figure 76 - Jig Concept 1

9.3.2 Concept 2

Concept design 2 of the jig consists of a permanent hole in the table where the box will be continually placed. On the outer side of the casing there will be four printed corners to prevent horizontal movement. This is a very cost-effective method and works perfectly for the casings. Using the table, there will be two jigs for the two varied sizes of relay casing, the only difference being the space between the corners. It is a very simple yet competent design that is easy to repair if damaged and the overall height of the casing is reduced compared to the concept design 1. Also, it saves space on the table and is aesthetically more pleasing.

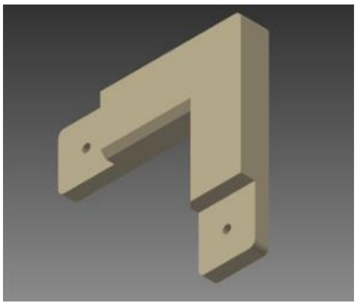


Figure 77 - Jig Concept 2

9.3.3 Concept 3

The final concept design of the jig consists of four corners and a wall, as shown in figure 78. The next step is to compensate for the part of the casing that extrudes the box and the small wall holds the smaller casing in place. This is the minimum size of jig that we could produce and thus saves a lot of space on the work table for components such as the connectors and label machine. It is 3D printed in ABS plastic and therefore will last a long time. It is also held onto the table using screws and bolts which optimally secure the jig in place, preventing movement and ensuring that YuMi will always be moving the casings to the same location.

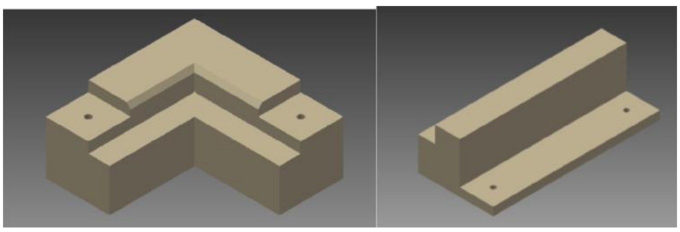


Figure 78 - Jig Concept 3

9.3.4 Final Jig Design

The jig that is to be taken forward is concept design 3. This is due to it being able to hold both jigs easily and securely without wasting material or time in the printing process. It is also easily mounted to the table and simple in design and structure, making it easy for the YuMi to place the casing into and out of. In addition, no gaps in the table are needed due to the height compensation.

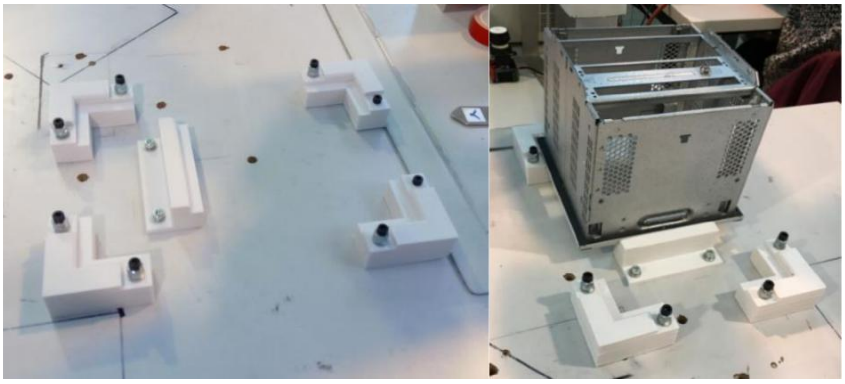


Figure 79 - Final Jig Design

9.4 Support

9.4.1 Reasoning

Due to flipping the boxes being a supposed problem, another robot will present the boxes to the YuMi workstation and will hold the relay box while the stickers are being placed. To mimic this action, a support has been manufactured to hold the casings in the same position each time where it is imagined that the other robot is holding this. After that, YuMi with his suction cup will pick the stickers from the labelling machine and it will place the stickers and connectors in the correspondent positions of the relay box.

For completing the simulation of this operation in the laboratory of Technobothnia, a stand has been designed and crafted for the two types of boxes. The height that the relay boxes will be suspended in the air by the other robot will be of 326 mm. The reason that this height has been chosen is because it is in the same level of the arms of YuMi and it is an easier movement to reach the box with the vacuum suction cup to place the sticker at the same time. Also, the stand will be positioned in the middle of the working table facing in front of the YuMi robot (distance 0 in the plan view image) for the same movement reach reasons.

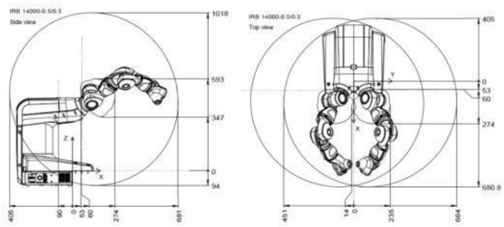


Figure 80 - YuMi Reach Measurements

9.4.2 Materials Available and Manufacturing Process

Firstly, a visit to the workshop was had to determine some ideas of how to manufacture the support and to search for reusable material that has been discarded from other projects. Secondly, how the stand will be constructed, and which parts are needed were understood.

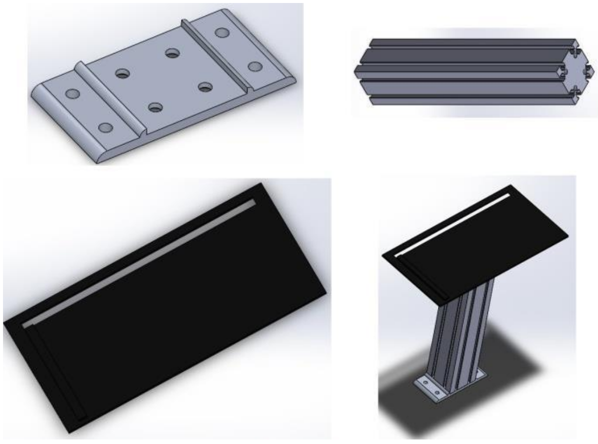


Figure 81 - CAD Parts and Assenbky for the Casing Stand. Source: Own Elaboration

The parts of the stand assembly and his dimensions will be:

- Aluminium base profile: 120x60x6 mm

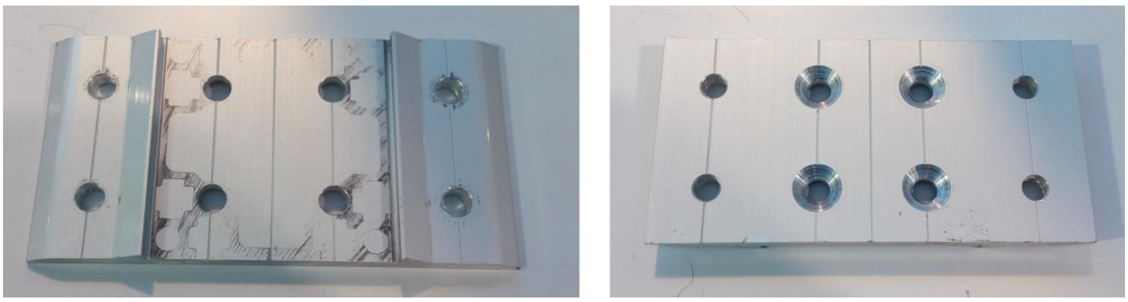


Figure 82 - Aluminium Base Profile. Source: Own Elaboration.

- Aluminium T-slot profile: 60x60x320 mm

Figure 83 - Aluminium T-Slot Profile. Source: www.diy-india.com

- PMMA plastic (acrylic) plate black: 300x150x3 mm

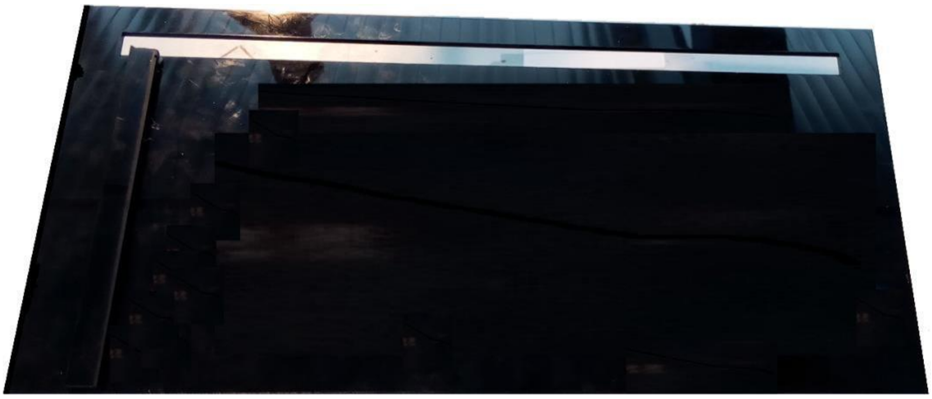


Figure 84 - PMMA Plastic Plate

Once the design is completed, different parts were cut with the required measurements. Afterwards, the assembly of the different parts could begin in the following order:

1. The aluminium T-slot profile is screwed to the aluminium base profile with 4 M6x30 countersunk screws.
2. The table where YuMi is installed, is drilled with 4 holes of 7 millimetres diameter with the base profile distances and respecting the middle of the table.
3. Following this, the base profile (with the aluminium T-slot profile joined) is screwed to the table with M6x55 screws.
4. Finally, the black acrylic plate is glued to the assembly using thermoplastic glue.

The result of the stand where the two types of boxes fits perfectly can be seen on the images:

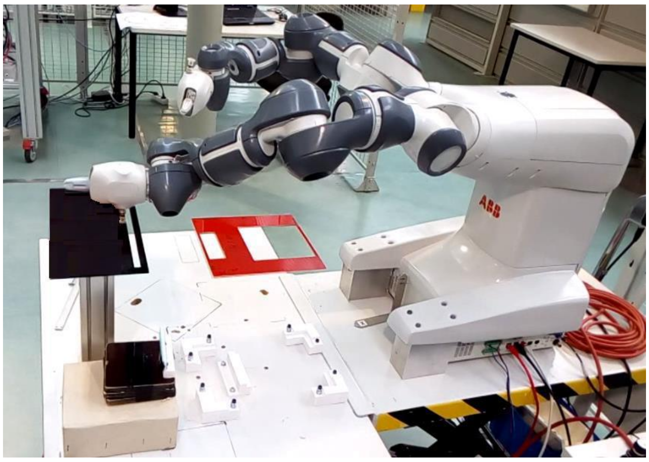


Figure 85 - Stand Installed on the Working Table. Source: Own Elaboration

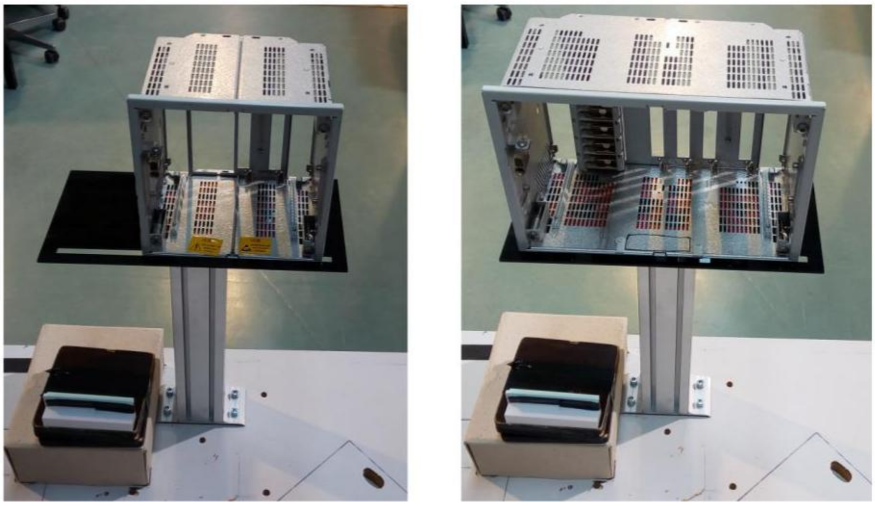


Figure 86 - Stand Supporting the Small and Large Box. Source: Own Elaboration

9.4.3 Testing

Continuing from the jig being designed and manufactured, it was to be tested to understand if it was efficient in holding both cases at the correct locations every time. This required some reprogramming of the YuMi and with use over time it is still proving to be as efficient as it did on the day of manufacture. Therefore, as a temporary solution until the larger robot is implemented manufacture, it can be trusted to work in place.

9.5 Jig for Connectors

Connectors must be picked always from the same location, due that a jig to store them was thought. The function of the jig is not only to store them but to serve the connectors to YuMi on a conveyor belt.

To comply with ABB requirements of the project having to be fully automated, it could work storing them vertically and with some mechanical structure, or even with a pneumatic cylinder, when the lowest connector is taken, the next connector should be moved to the outside of the jig, ready to be taken by YuMi.

The height of the jig may be based on the average of orders per day. As higher it is, more connectors can be stored. The only problem we have come across is that, at some point, ABB workers would have to refill it.

Below, some design suggestion that was made for the RobotStudio simulation:

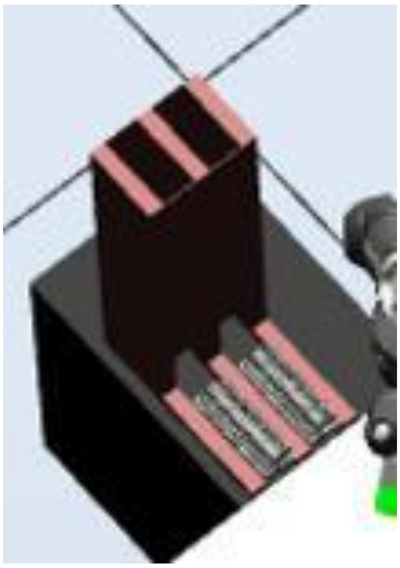


Figure 87 - Jig for Connectors

10 Evaluation

10.1 Final Results

Focussing on our mission below, the final results are explained in this section:

'' Our mission is to substitute a human worker with the YuMi robot by placing both the labels and a connector plate into the casing for a safety relay''.

With reference to the project, we have achieved the task set to us. The connector is placed using YuMi, as are the labels, in their desired locations and in under half of the allowable time. The grippers have been designed and the Robotstudio software has been used to complete all these tasks, as ABB required.

Despite this, a fully automated process has not been fully realised yet. There are some issues that require further work such as, implementing the full layout, programming the IRB 1600 robot and the labelling machine and SAP (ABB order manager). With confirmation from ABB, it has been understood that this will not arise issues due to a third-party company completing these tasks.

Refilling of the storage spaces will also need to be replaced by a human worker when they are empty such as, the connector and case storage. This results in ABB Having to spend the minimum on human resources.

10.2 Conclusion

To conclude, the team has had a fantastic time throughout the duration of the project. So many new skills have been learned such as; project and time management, Robotstudio, material resources and so many more. All members of the team contributed equally to the result of the project and are extremely proud to have arrived at the end, completing all tasks that were given.

A whole different type of education has been studied and the new learnings and skills that the team now possess will aid us in finding our future careers and further studies. Furthermore, working with many different cultures prepares us for the future and the team can now say that they are comfortable working in multi-cultural environments and can do so to a high standard. We are very thankful that the European Project Semester, ABB and all our lecturers and supervisors have given us this opportunity to grow as individuals and to help us learn so many new things.

Reference Page

Europeanprojectsemester.eu. (2018). *Introduction - European Project Semester*. [online] Available at: http://europeanprojectsemester.eu/info/Introduction [Accessed 26 Feb. 2018].

Belbin.com. (2018). *Belbin | Great teams start with Belbin*. [online] Available at: http://www.belbin.com/about/belbin-team-roles/ [Accessed 9 Feb. 2018].

New.abb.com. (2018). *IRB 14000 YuMi - Industrial Robots From ABB Robotics*. [online] Available at: http://new.abb.com/products/robotics/industrial-robots/yumi [Accessed 4 Mar. 2018].

New.abb.com. (2018). *About*. [online] Available at: http://new.abb.com/about [Accessed 7 Mar. 2018].

New.abb.com. (2018). *ABB products and services A - Z index*. [online] Available at: http://new.abb.com/offerings [Accessed 8 Mar. 2018].

New.abb.com. (2018). *YuMi*. [online] Available at: http://new.abb.com/future/yumi [Accessed 10 Mar. 2018].

New.abb.com. (2018). *YuMi - Teollisuusrobotit | ABB*. [online] Available at: http://new.abb.com/products/robotics/fi/teollisuusrobotit/yumi [Accessed 10 Mar. 2018].

care?, W. and Digest, P. (2018). *What are collaborative robots and why should you care? | Packaging Digest. [online] Packagingdigest.com*. Available at: http://www.packagingdigest.com/robotics/what-are-collaborative-robots-and-why-should-you-care1505 [Accessed 19 Feb. 2018].

Bélanger-Barrette, M. (2018). *3 Best Collaborative Robot Applications. [online] Blog.robotiq.com.* Available at: https://blog.robotiq.com/3-best-collaborative-robots-applications [Accessed 19 Feb. 2018].

Rethink Robotics. (2018). *Collaborative Robotics Applications and Industries | Rethink Robotics. [online]* Available at: http://www.rethinkrobotics.com/applications/ [Accessed 19 Feb. 2018].

Cobots, M., OB7, P., TX2, S., 5, C., TM5, T., MICO, K., Eva, A., HC10, Y., 2, F., Baxter, R., OUR, A., IIWA, K., 10, M., CR-35iA, F., KATIA, C., Nextage, K., Sawyer, R., Cobotta, D., Universal Robots: UR3, U., Yumi, A., HCR-5, H., OB7, P., TX2, S., 5, C. and TM5, T. (2018). *CobotsGuide | The Cobots*. [online] Cobotsguide.com. Available at: http://cobotsguide.com/cobots/ [Accessed 19 Feb. 2018].

Cobots, M., OB7, P., TX2, S., 5, C., TM5, T., MICO, K., Eva, A., HC10, Y., 2, F., Baxter, R., OUR, A., IIWA, K., 10, M., CR-35iA, F., KATIA, C., Nextage, K., Sawyer, R., Cobotta, D., Universal Robots: UR3, U., Yumi, A., HCR-5, H., OB7, P., TX2, S., 5, C. and TM5, T. (2018). *CobotsGuide | The Cobots*. [online] Cobotsguide.com. Available at: http://cobotsguide.com/cobots/#charts [Accessed 4 Mar. 2018].

Collaborative Robots Buyer's Guide. 6th Edition. (2018). [ebook] Available at: http://neffautomation.com/wp-content/uploads/COBOT-EBOOK-FINAL6.pdf [Accessed 4 Mar. 2018].

Amadamiyachi.com. (2018). *[online]* Available at: http://www.amadamiyachi.com/servlet/servlet.FileDownload?retURL=%2Fapex%2Feducationalresources_articles&file=0153400000255FW [Accessed 19 Feb. 2018].

Amadamiyachi.com. (2018). *[online]* Available at: http://www.amadamiyachi.com/servlet/servlet.FileDownload?retURL=%2Fapex%2Feducationalresources_articles&file=0153400000255FW [Accessed 19 Feb. 2018].

Autodesk.eu. (2018). *Inventor | Mechanical Design & 3D CAD Software | Autodesk*. [online] Available at: https://www.autodesk.eu/products/inventor/overview [Accessed 14 Mar. 2018].

Williams, H. (2018). *5 Impressive Ways 3D Printing is Being Used Around the World*. [online] Techworld. Available at: https://www.techworld.com/picture-gallery/tech-innovation/5-top-uses-of-3d-printing-3666509/ [Accessed 14 Mar. 2018].

Tinkercad Blog. (2018). *3D Printing Materials Guide*. [online] Available at: https://blog.tinkercad.com/materialsguide/ [Accessed 14 Mar. 2018].

New.abb.com. (2018). *RobotStudio - ABB Robotics*. [online] Available at: http://new.abb.com/products/robotics/robotstudio [Accessed 17 Mar. 2018].

En.wikipedia.org. (2018). *Earned value management*. [online] Available at: https://en.wikipedia.org/wiki/Earned_value_management [Accessed 1 Apr. 2018].

Wbdg.org. (2018). *Earned Value Analysis | WBDG Whole Building Design Guide*. [online] Available at: https://www.wbdg.org/resources/earned-value-analysis [Accessed 1 Apr. 2018].

Product specification IRB 1600/1660. Document ID: 3HAC023604-001. Revision: AF. ABB Group

IRB 1600 presentation: The highest performance general purpose 10 kg robot. ABB Group. April 24, 2013

IRB 1600 Ddata sheet. Doc No: PR10282EN_R8. Published: 2018-02-15. Revision: D. ABB Group

Collaborative Robots Buyer's Guide. 6th Edition. http://neffautomation.com/wp-content/uploads/COBOT-EBOOK-FINAL6.pdf [Visited 19/02/2018]

https://www.mordorintelligence.com/industry-reports/industrial-robotics-market?gclid=EAIaIQobChMIlOHQxP6x2QIVT7obCh0fcQ5kEAAYASAAEgKDVfD_BwE [Visited 19/02/2018]

https://www.marketsandmarkets.com/Market-Reports/collaborative-robot-market-194541294.html [Visited 19/02/2018]

http://developercenter.robotstudio.com/BlobProxy/manuals/IRC5FlexPendantOpManual/doc27.html CONSULT: 27/04/2018

[https://www.onlinelabels.com/Articles/printing_direct_thermal_transfer.htm]

[https://www.computype.com/blog/learn-the-difference-between-on-demand-label-printing-technologies]

[https://www.satoamerica.com/resource-library/learning-center/dt-vs-tt.aspx]

[https://www.onlinelabels.com/]

[https://www.youtube.com/watch?v=ouOZ9AbDyxs]

[https://www.alibaba.com/showroom/programmable-printers_3.html?spm=a2700.7724857.6.3.3a6dff1cXuNUoU]

[http://www.precisionroller.com/category/label-printers.htm

[http://www.oilchangestickers.com/cat-42-2-13/printers-and-supplies.htm]

[https://yourbusiness.azcentral.com/thermal-ribbon-last-longer-toner-28732.html]

[https://www.peak-ryzex.com/articles/the-complete-list-of-industrial-label-printer-manufacturers]

Project Risk Management. Basics in Project Management, Roger Nylund slides. Spring semester 2018.

Fundamentals of project management, 4th edition. Joseph Heagney. Consult date: 04-05-2018

[SOURCE BOOK**:** Product specification IRB 14000.

Trace back information: Workspace R17-2 version a19

Published 2017-10-17 at 05:04:41

Skribenta version 5.1.011

DATE CONSULT: 11/04/2018]

Appendix

1. Project Time Management

1.1 Work Breakdown Structure

The main purpose of Work Breakdown structure (also known as WBS) is to recognize all the tasks that need to be done for completing the project. The whole project is to be broken into small tasks until they are easier to manage.

We must take into account that the tasks that have been made in a sequential way, being careful that some tasks will require to be finished before other tasks can start.

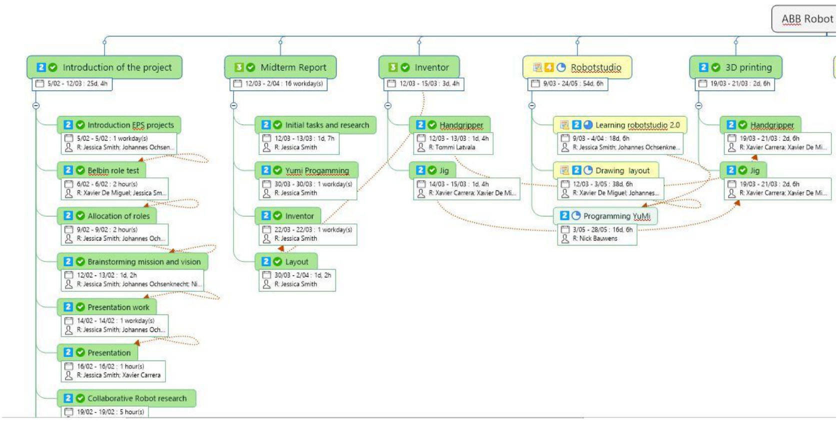


Figure 88 - WBS 1

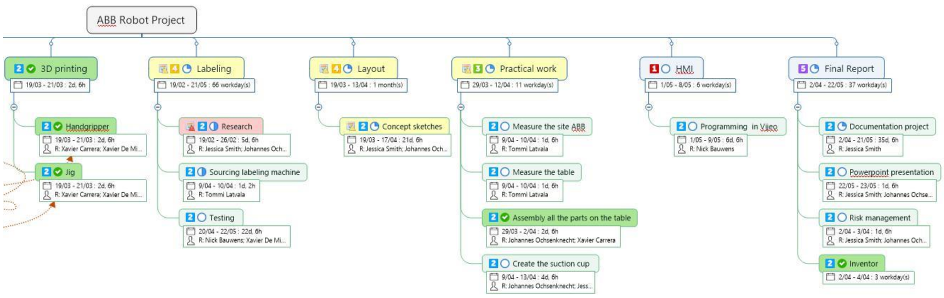


Figure 89 - WBS 2

1.2 Schedule

Once we had completed the Work Breakdown Structure, we started to work on the schedule. The schedule has been derived with the software Microsoft Project.

Microsoft Project is a powerful and dynamic scheduling tool for project management. The project activities and tasks are constructed in Gantt charts (the timeline is visualized also here) and the resources can be assigned to each task. Furthermore, it indicates the hours spent on every task for each resource and the total hours that individual members spent.

In the following images the Gantt chart of our project is show:

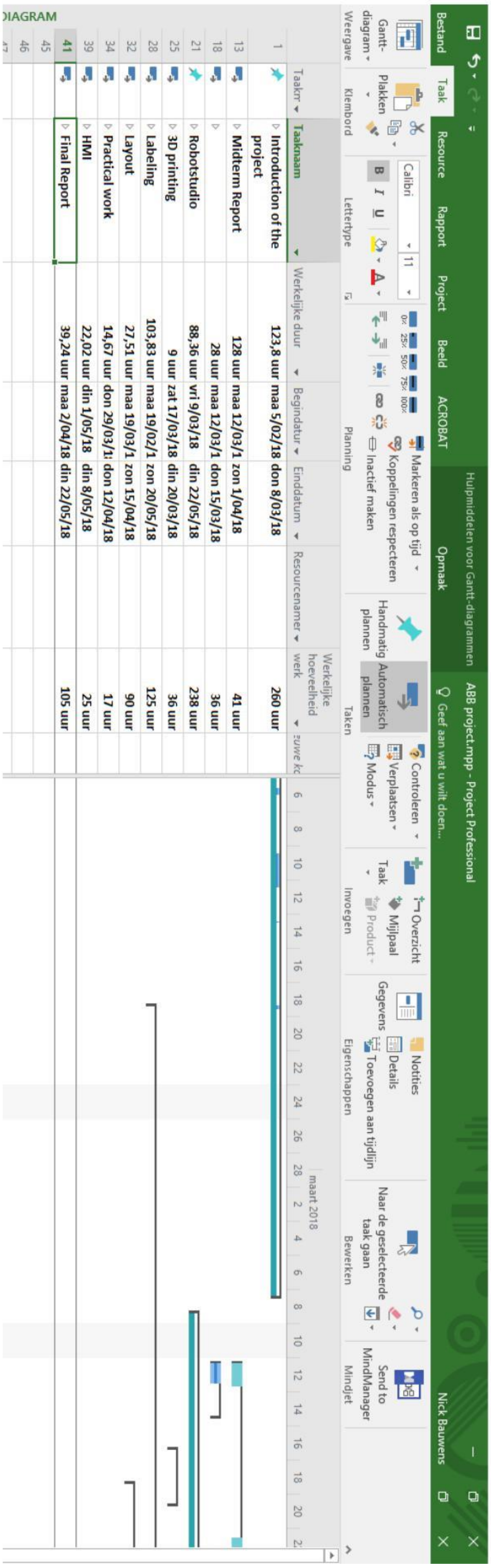


Figure 90 - Gantt Chart 1

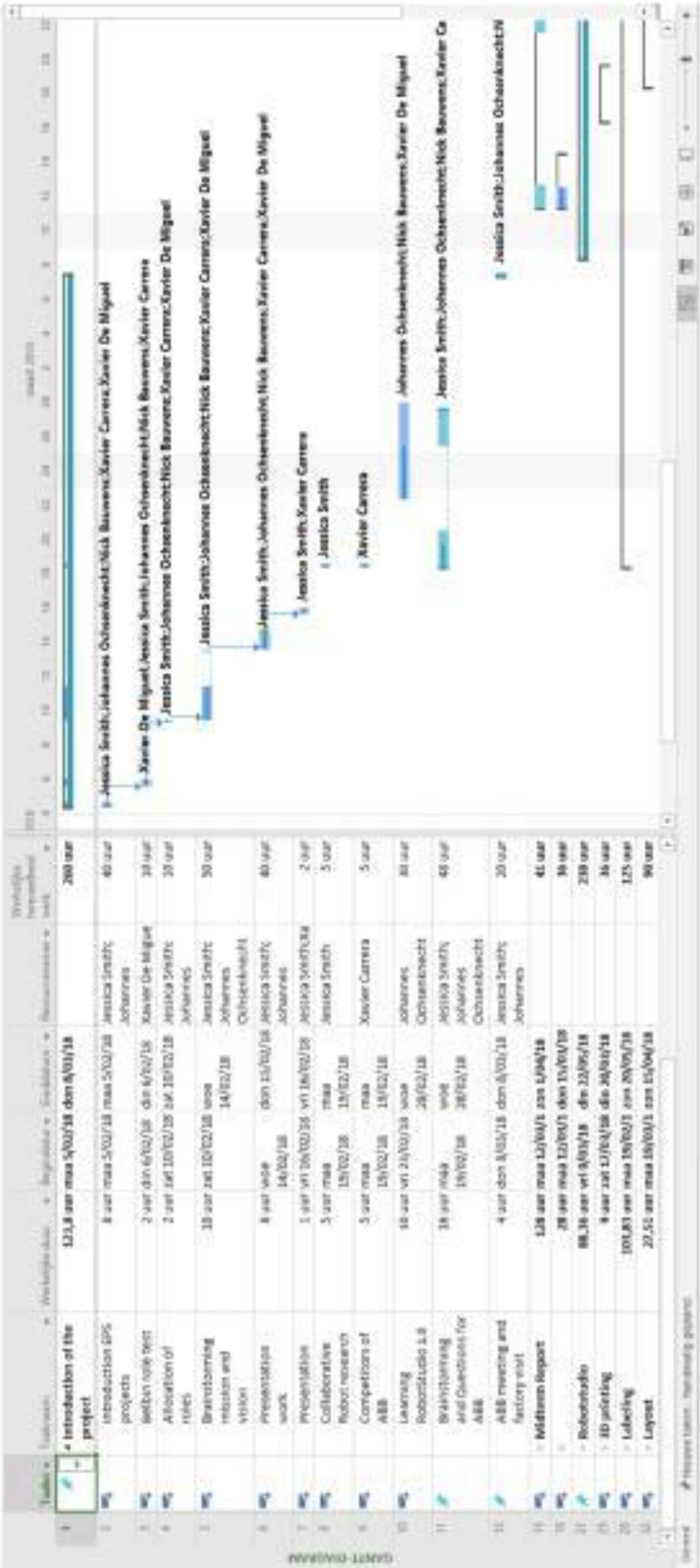


Figure 91 - Gantt Chart 2

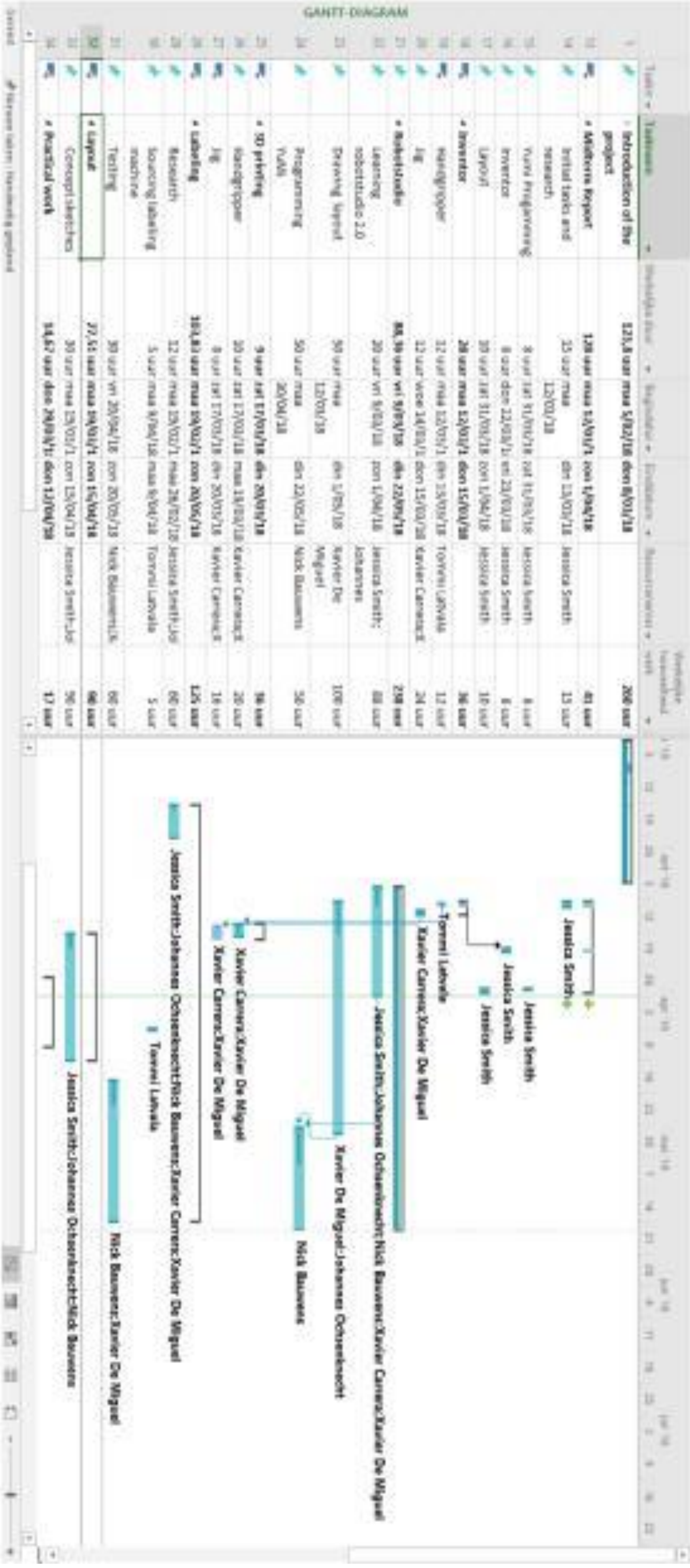


Figure 92 - Gantt Chart 3

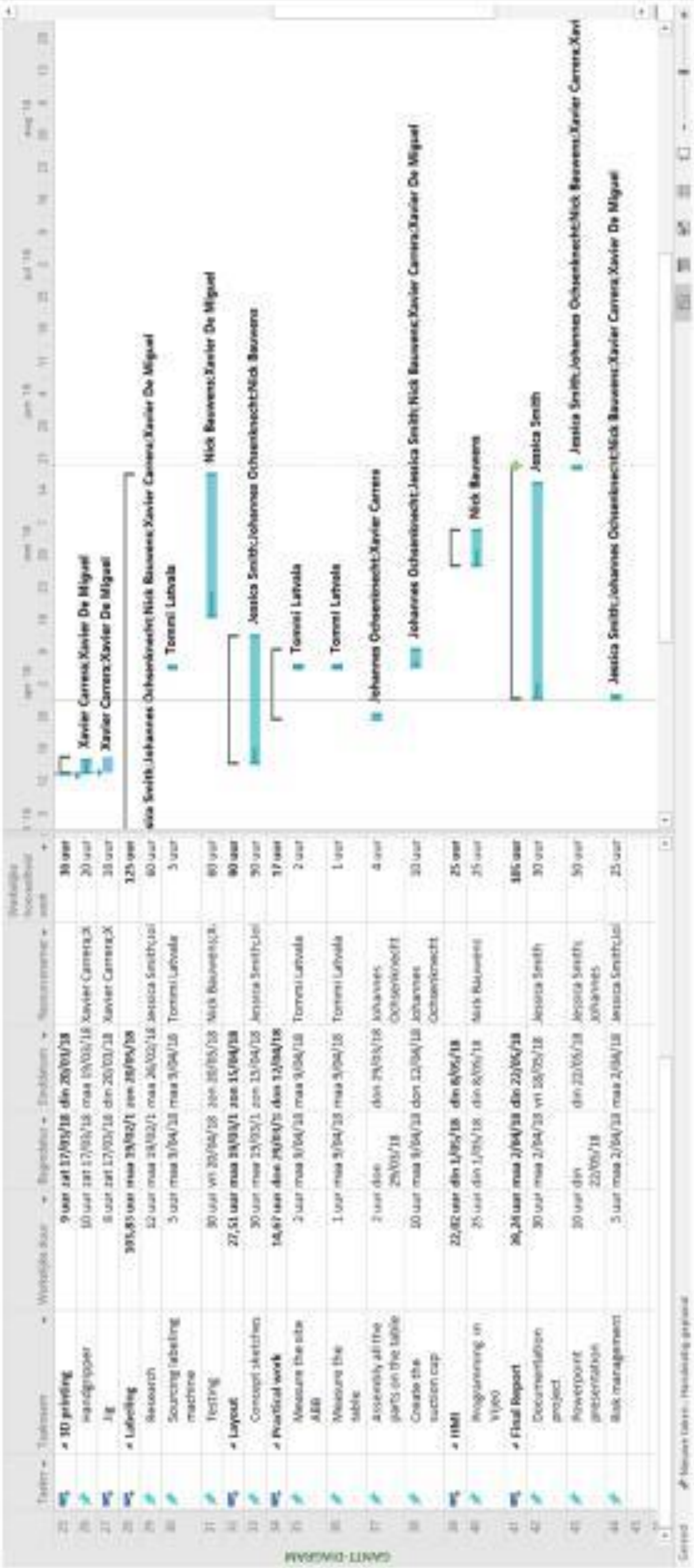


Figure 93 - Gantt Chart 4

1.3 Milestones

Milestones are significant stages that must be achieved to make a project a success. In addition, milestones permit doing project reviews at these key points.

The milestones of the present project are:

Stage	Date
Project start	**06/02/2018**
ABB plant visit	**08/03/2018**
Mid-term report and presentation	**03/04/2018**
Design a simulation of the layout	**25/04/2018**
Design and manufacture of the grippers	**20/04/2018**
Design and manufacture the jig and fixtures	**15/04/2018**
Connector placement	**08/05/2018**
Labelling method	**11/05/2018**
Final Report	**18/05/2018**
Final Presentation	**22/05/2018**

Table 10 - Milestones

2. Project Human Resource Management

2.1 Team Rules

As a team, ABB have derived a set of team rules that will be followed by all members always. This ensures that the team will work at an optimum and by using these statements as a guidance, possible conflicts will be resolved soundly and subsequently, the project will be completed to the highest standard:

1. Only English to be spoken, other languages to be translated into English when a word is not understood or known to a group member.
2. Attendance at arranged meetings is compulsory, if a team mate cannot attend then notification is necessary.
3. Respect each other.
4. Respect deadlines.
5. Listen to new ideas and encourage positivity.
6. Agree at meetings the best time for all participants to regroup during the week.
7. The roles of each teammate are to be respected.
8. All decisions are to be agreed on by all members.
9. Shared workload.
10. If task is unclear to a member, they must ask others to prevent confusion and enhance performance.
11. Lean about each other and the different cultures to gain a better team structure.
12. Focus on the task.
13. Conflicts should be handled professionally and effectively.
14. On completion of practical work – all tools should be tidied and cleaned. All team members must stay until this is finished.
15. Compliments and thanks for a job well done should be encouraged. Do not compare the best in you to the worst in them.

2.2 Belbin Test

2.2.1 Role Explanation

Resource Investigator

The resource investigator is very enthusiastic about their ideas and keen to get started on the project. They are usually extroverted and use their imagination to find solutions to problems. However, if the task is not moving at a fast-enough pace then they may lose interest.

Team Worker

Usually, the team worker acts as the glue for the other members, they are easy-going and like to ensure everyone's ideas are heard. However, people who identify as team workers often avoid conflict and will not involve themselves in any confrontation, resulting in them not being able to make hard decisions.

Co-Ordinator

The coordinator is a mature and confident individual who can identify the strengths of others which can make them able to give tasks to other members of the team. As a result of this, other team members may find the coordinator manipulative or giving the other members of the team too much work in comparison to their own tasks.

Plant

The main characteristics of the plant is that they can use their imagination to think of unusual ways of solving problems and using their creative mind in most tasks given. However, with this energy into finding solutions, they may not focus on enlightening the rest of the members to their ideas and not pay attention to detail.

Monitor/Evaluator

The monitor is a very logical thinker, they can make quick and effective judgements resulting in a fast decision maker. They always have their feet on the ground and take into consideration all different ideas. The weaknesses of the monitor is that they may not be very enthusiastic and can be over analytical of other members work.

Specialist

A specialist is an individual with a very particular skillset, they have exceptional knowledge on one key area and can therefore start working off their own initiative almost immediately when the project is presented to them. With this specific skillset, however, they tend to only focus on this aspect not involving themselves in any of the other tasks.

Shaper

The shaper is very good at motivating the rest of the team to get the job done quickly and efficiently, they work very well when under pressure and can focus well on the task at hand. Although, with their drive and determination, they may become bad tempered with the other members of the group.

Implementor

Implementors work very hard to get tasks done through organization and planning. They are productive and reliable to work hard to finish the jobs. However, they are prone to thinking that their idea is better than others' ideas and it will take a long time for them to come around to accepting someone else's idea is more efficient than theirs.

Completer/Finisher

The completer is ultimately a perfectionist, finding errors in others work when they think the task is complete – tying up any loose ends and making the report/project as best as it can be. As they are so cautious of being perfect, they may become anxious and worried about the outcome and also can become obsessed with the idea of perfectionism, irritating others within the group.

2.2.2 Team Test Results

Belbin test results Xavier de Miguel

	Coordinator	Shaper	Plant	Monitor	Implementer	Resource investigator	Team worker	Finisher
1	0	0	2	1	3	1	3	0
2	0	2	0	1	5	1	0	1
3	1	1	0	1	2	0	3	2
4	0	0	1	1	4	1	3	0
5	0	0	0	0	10	0	0	0
6	0	0	0	10	0	0	0	0
7	0	10	0	0	0	0	0	0
Tot	1	13	3	14	24	3	9	3

think act

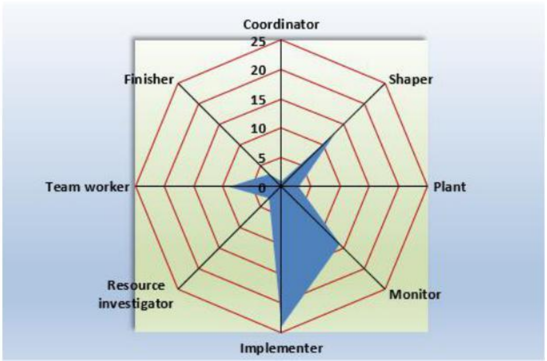


Figure 94 - Xavi de Miguel Belbin results

Belbin test results Johannes

	Coordinator	Shaper	Plant	Monitor	Implementer	Resource investigator	Team worker	Finisher
1	1	1	1	1	2	1	2	1
2	2	1	2	0	2	1	1	1
3	2	1	1	1	0	1	2	2
4	1	2	2	0	2	1	1	1
5	1	3	1	1	2	1	1	0
6	1	1	2	1	1	1	1	1
7	2	1	1	2	1	1	1	1
Tot	10	10	10	6	10	7	9	7

think act

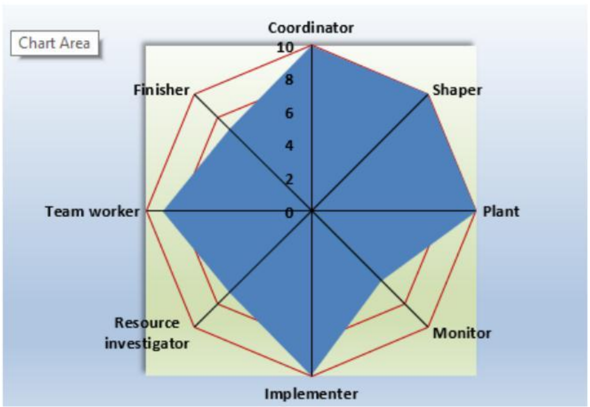
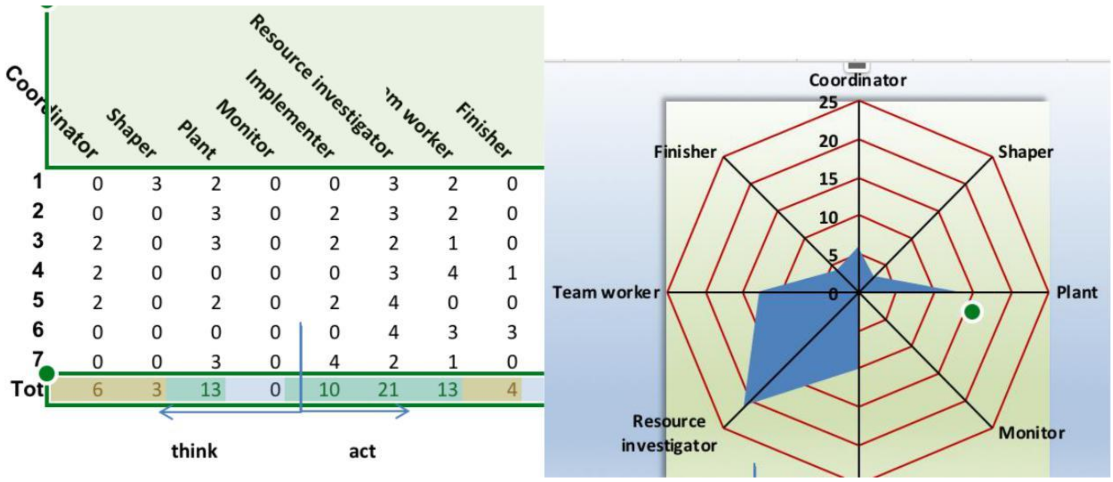


Figure 95 - Johannes Belbin results

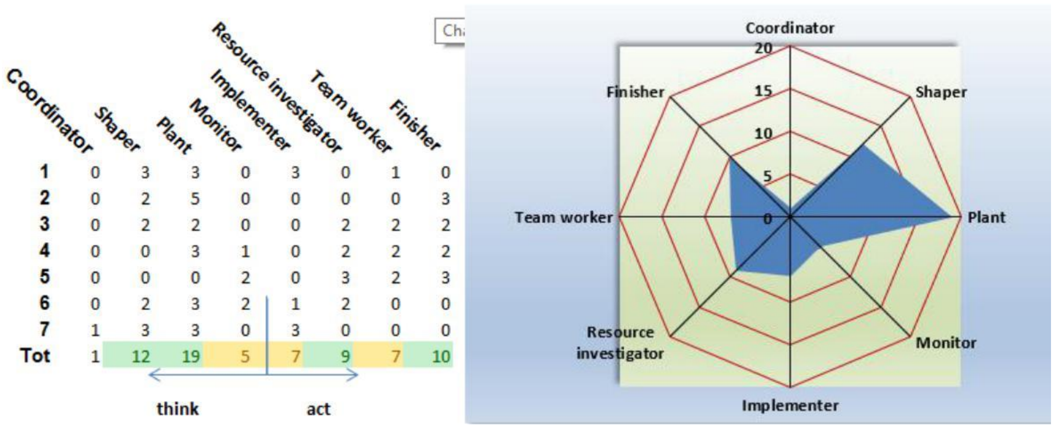
Belbin test results Jessica



	Coordinator	Shaper	Plant	Monitor	Implementer	Resource investigator	?m worker	Finisher
1	0	3	2	0	0	3	2	0
2	0	0	3	0	2	3	2	0
3	2	0	3	0	2	2	1	0
4	2	0	0	0	0	3	4	1
5	2	0	2	0	2	4	0	0
6	0	0	0	0	0	4	3	3
7	0	0	3	0	4	2	1	0
Tot	6	3	13	0	10	21	13	4

Figure 96 - Jessica Belbin results

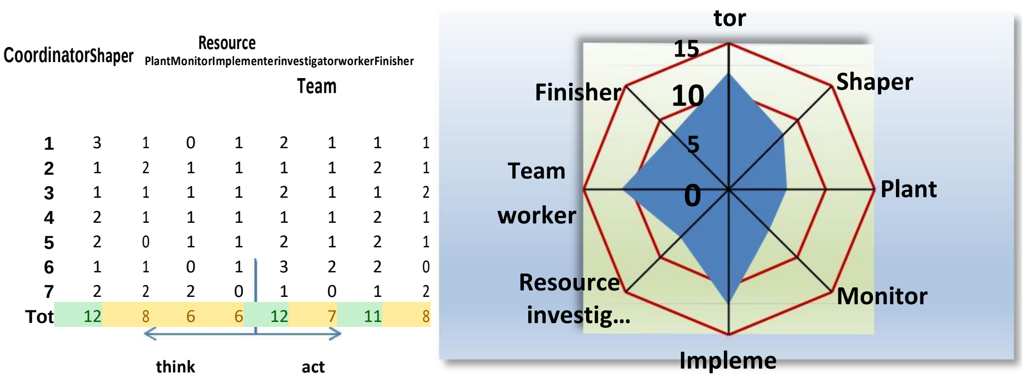
Belbin test results Nick



	Coordinator	Shaper	Plant	Monitor	Implementer	Resource investigator	Team worker	Finisher
1	0	3	3	0	3	0	1	0
2	0	2	5	0	0	0	0	3
3	0	2	2	0	0	2	2	2
4	0	0	3	1	0	2	2	2
5	0	0	0	2	0	3	2	3
6	0	2	3	2	1	2	0	0
7	1	3	3	0	3	0	0	0
Tot	1	12	19	5	7	9	7	10

Figure 97 - Nick Belbin results

Belbin test results Xavier Carrera



	Coordinator	Shaper	Plant	Monitor	Implementer	Resource investigator	Team worker	Finisher
1	3	1	0	1	2	1	1	1
2	1	2	1	1	1	1	2	1
3	1	1	1	1	2	1	1	2
4	2	1	1	1	1	1	2	1
5	2	0	1	1	2	1	2	1
6	1	1	0	1	3	2	2	0
7	2	2	2	0	1	0	1	2
Tot	12	8	6	6	12	7	11	8

Figure 98 - Xavier Carrera Belbin results

2.3 RACI Matrix

RACI matrix is a project management tool that identifies roles and responsibilities of a project team. With this, every component of the team knows his responsibility and evade doing a task twice, confusions of who does each task, etc. In definitive, clarifies to every participant what his role and task is.

RACI is the acronym of Responsible, Accountable, Consulted and Informed. The meaning of each word is:

- Responsible: Member that does the job to accomplish the task.
- Accountable: This person has decision authority and is responsible of the correct and finalization of the task.
- Consultant: Persons that can have more information about the tasks and his opinion must be considered before deciding. They can be experts in the matter.
- Informed: Those who are informed about the tasks progress and decisions.

RACI Materials			Project Team				
Role	**Project Area**	**Activity**	**Nick**	**Johannes**	**Jessica**	**Xavi C**	**Xavi de M**
	Project Management	Research	I	C	R	A	I
R		Schedule	R	I	A	I	C
		Risk Management	A	C	I	R	I
		Cost Management	R	C	A	I	I
A		Mid Term Report	A	A	R	A	A
		Final Report	A	A	R	A	A
		Presentations	A	A	R	A	A
	Design and Programming	Robotstudio	R	I	A	I	A
C		Layout	A	C	A	C	R
		Inventor	C	R	C	A	I
		Labelling	A	A	A	A	R
I		Connector	A	A	A	A	R
		3D Printing	I	A	I	R	A

Role Key	**R:** Responsible	**A:** Accountable	**C:** Consultant	**I:** Inform

Table 11 - RACI Matrix

3. Project Cost Management

3.1 Materials and Manufacturing Costs

Table X gives a clear indication of what the team have been spending their budget on and all the volume of materials that have been used throughout the project.

Item No.	Concept	Quantity	Units	€/unit	Price
1	Yumi Robot	1	pc	50,000.00	50,000.00
2	ABS plastic wall test	15	g	0.02	0.24
3	ABS plastic wall	20	g	0.02	0.32
4	ABS plastic corner 1	25	g	0.02	0.40
5	ABS plastic corner 2	25	g	0.02	0.40
6	ABS plastic corner 3	25	g	0.02	0.40
7	ABS plastic corner 4	25	g	0.02	0.40
8	ABS plastic gripper Mk 1, 2 units (1 unit = 5,9g)	12	g	0.02	0.19
9	ABS plastic gripper Mk 2, 2 units (1 unit = 8g)	12	g	0.02	0.19
10	Screw M5x60	10	pc	0.30	3.00
11	Washer M5	10	pc	0.11	1.10
12	Nut M5	10	pc	0.60	6.00
13	Screw M6x55	4	pc	0.36	1.44
14	Washer M6	4	pc	0.13	0.52
15	Nut M6	4	pc	0.50	2.00
16	Countersunk screw M6x30	4	pc	0.33	1.32
17	Aluminum T-slot profile 60x60	0.32	m	14.00	4.48
18	Aluminum base plate 120x60	1	pc	4.00	4.00
19	PMMA plastic (acrylic) plate black 300x150x3	1	pc	8.00	8.00
20	ABS plastic gripper Mk 3, 8 units (1 unit = 5,8g)	46.4	g	0.02	0.74
21	PLA plastic gripper Mk 3, 4 units (1 unit = 10,3g)	41.2	g	0.02	0.88
22	Labeling Machine	1	pc	1,500.00	1,500.00
				TOTAL	**51,536.01€**

Table 12 - Manufacturing Costs

3.2 Earned Value Analysis

Earned Value Analysis (EVA) is graphic to display a project's progress at any given point in time. The graphic can show us the past and the current performance of the progress and tries to predict the future of the project by using statistical techniques. In order to analyse the progress, the schedule, the costs of the project and the number of hours that the team has been working in the project are needed.

- Budget at Completion (BAC) is the total budget cost for the project
- Actual Costs (AC or ACWP) is the cost in relation to the actual hours worked by the team
- Planned Value (PV or BCWS) is the predicted work that should be done from the time of starting the project to the deadline date
- Earned Value (EV or BCWP) is all the work that has been logged, documented and completed by the team up until a certain date

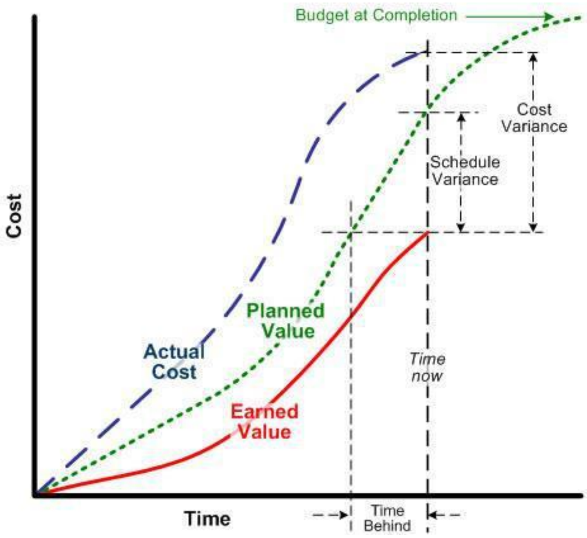


Figure 99 - Earned Value Analysis Graph Example

The ABB team have derived the tasks that will be completed throughout the duration of the project and given each an hourly rate accordingly. These tasks were then given a time frame and the costing was calculated, for the first eight weeks these are shown in figure X, the remaining eight weeks are shown in figure 100.

Planned Value

TASK	€/h	Week 1		Week 2		Week 3		Week 4		Week 5		Week 6		Week 7		Week 8		%Progress
Introduction of the project	15	70h	1 050 €	92h	1 380 €	45h	675 €	33h	495 €	20h	300 €		0 €		0 €		0 €	100%
Midterm Report	30		0 €		0 €		0 €		0 €		0 €	15h	450 €	8h	240 €	18h	540 €	100%
Inventor	25		0 €		0 €		0 €		0 €		0 €	36h	900 €		0 €		0 €	100%
Robotstudio	40		0 €		0 €		0 €		0 €	41,83h	1 673 €	41,83h	1 673 €	41,83h	1 673 €	41,83h	1 673 €	45%
3D printing	15		0 €		0 €		0 €		0 €		0 €	36h	540 €		0 €		0 €	100%
Labeling	30		0 €		0 €	60h	1 800 €		0 €		0 €		0 €		0 €		0 €	50%
Layout	25		0 €		0 €		0 €		0 €		0 €		0 €	22,5h	563 €	22,5h	563 €	60%
Practical work	30		0 €		0 €		0 €		0 €		0 €		0 €		0 €	4h	120 €	25%
HMI	25		0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	0%
Final Report	40		0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	0%
Total		**70h**	**1 050 €**	**92h**	**1 380 €**	**105h**	**2 475 €**	**33h**	**495 €**	**61,83h**	**1 973 €**	**128,83h**	**3 563 €**	**72,33h**	**2 476 €**	**86,33h**	**2 896 €**	**58%**

Table 13 Planned Value First 8 Weeks

Planned Value																
Week 9		Week 10		Week 11		Week 12		Week 13		Week 14		Week 15		Week 16		%Progress
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
39,78h	1 591 €	39,78h	1 591 €	87h	3 480 €	87h	3 480 €	39,78h	1 591 €	39,78h	1 591 €	39,79h	1 591 €		0 €	45%
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
	0 €		0 €	60h	1 800 €		0 €		0 €		0 €		0 €		0 €	50%
22,5h	563 €	22,5h	563 €		0 €		0 €		0 €		0 €		0 €		0 €	60%
	0 €	67,5h	2 025 €		0 €		0 €		0 €		0 €		0 €		0 €	25%
	0 €		0 €		0 €		0 €	54,5h	1 363 €		0 €		0 €		0 €	0%
	0 €		0 €		0 €	14,5h	580 €		0 €	43,29h	1 732 €	43,29h	1 732 €	14,5h	580 €	0%
62,28h	**2 154 €**	**129,78h**	**4 179 €**	**147h**	**5 280 €**	**101,5h**	**4 060 €**	**94,28h**	**2 954 €**	**83,07h**	**3 323 €**	**83,07h**	**3 323 €**	**14,5h**	**580 €**	**58%**

Table 14 - Planned Value Final 8 Weeks

Using these tables, the planned value could be plotted against the costings that the company have agreed as budgeted hours (table x). On the same graph; the actual costings is displayed in the same format, however; these are subject to the hours the team has actually worked within the timeframe (so far, an eight week period).

Actual Cost																		
TASK	€/h	Week 1		Week 2		Week 3		Week 4		Week 5		Week 6		Week 7		Week 8		%Progress
Introduction of the project	15	70h	1 050 €	92h	285 €	40h	600 €	33h	495 €	20h	300 €		0 €		0 €		0 €	100%
Midterm Report	30		0 €		0 €	4h	120 €		0 €		0 €	8h	240 €	12h	360 €	25h	750 €	100%
Inventor	25		0 €		0 €		0 €		0 €		0 €	24h	600 €		0 €		0 €	100%
Robotstudio	40		0 €		0 €		0 €		0 €	35h	1 200 €	35h	1 400 €	40h	1 600 €	55h	2 200 €	45%
3D printing	15		0 €		0 €		0 €		0 €		0 €	35h	525 €		0 €		0 €	100%
Labeling	30		0 €	20h	600 €	55h	1 650 €		0 €		0 €		0 €		0 €	8h	240 €	50%
Layout	25		0 €		0 €		0 €	10h	250 €		0 €		0 €	20h	500 €	30h	750 €	60%
Practical work	30		0 €		0 €		0 €		0 €		0 €		0 €		0 €	4h	120 €	25%
HMI	25		0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	0%
Final Report	40		0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	0%
Total		**70h**	**1 050 €**	**112h**	**885 €**	**99h**	**2 370 €**	**43h**	**745 €**	**55h**	**1 500 €**	**102h**	**2 765 €**	**72h**	**2 460 €**	**122h**	**4 060 €**	**58%**

Table 15 - Actual Value First 8 Weeks

Actual cost																
Week 9		Week 10		Week 11		Week 12		Week 13		Week 14		Week 15		Week 16		%Progress
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
35h	1 400 €	40h	1 600 €	50h	2 000 €	35h	1 400 €	30h	1 200 €	30h	1 200 €	40h	1 600 €		0 €	100%
	0 €		0 €		0 €		0 €		0 €		0 €		0 €		0 €	100%
20h	600 €	10h	300 €	10h	300 €	5h	150 €	5h	150 €		0 €		0 €		0 €	100%
10h	250 €	10h	250 €	5h	125 €	5h	125 €	10h	250 €		0 €		0 €		0 €	100%
10h	300 €	20h	600 €	10h	300 €		0 €		0 €		0 €		0 €		0 €	100%
	0 €		0 €		0 €	20h	800 €	20h	800 €	20h	800 €	20h	800 €		0 €	100%
75h	**2 550 €**	**80h**	**2 750 €**	**75h**	**2 725 €**	**65h**	**2 475 €**	**65h**	**2 400 €**	**50h**	**2 000 €**	**60h**	**2 400 €**	**0h**	**0 €**	**100%**

Table 16 - Actual Value Last 8 Weeks

Additionally, the earned value was calculated using the formula:

$$\frac{h \qquad \times}{}$$

This formula creates the BCWP line on the Earned Value Analysis graph. It shows that the team is working to budget while also ahead of schedule which is a positive aspect for both the company and the team.

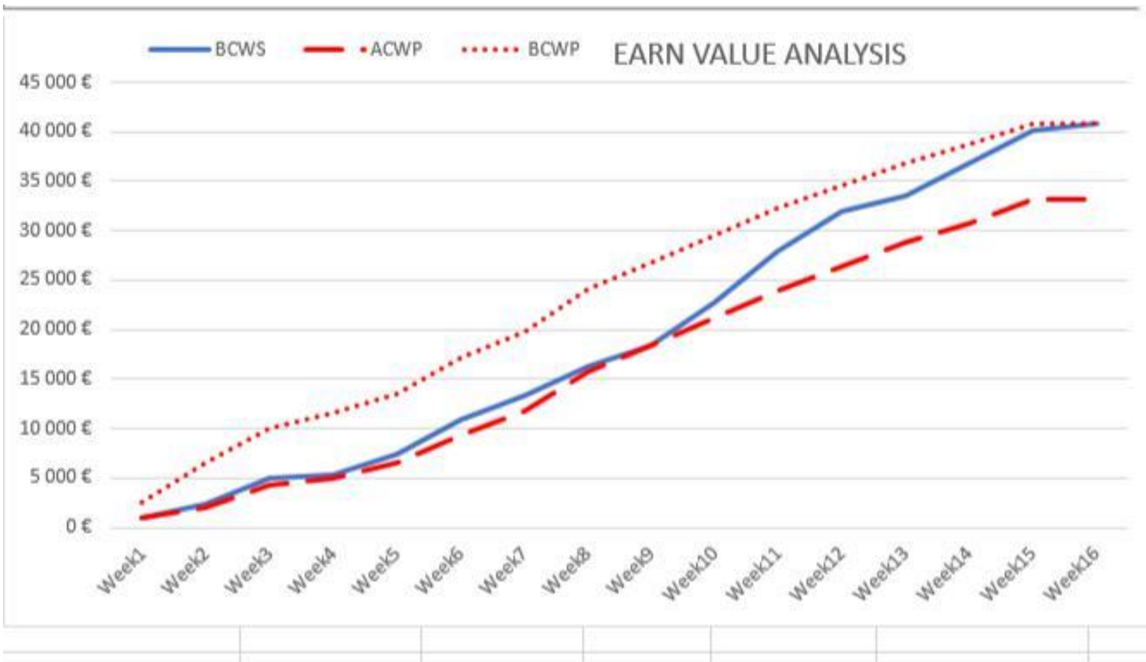


Figure 100 - Team ABB Earned Value Analysis Graph

http://www.chambers.com.au/glossary/earned_value_management.php

Budget at Completion (BAC). It represents the total budget cost for the project. Actual Costs (AC or ACWP). It is described as the expenses incurred in the project till the time of measurement Planned Value (PV or BCWS). It represents the planned work that should be completed until the time of measurement. Earned Value (EV or BCWP). The actual work that has been completed until the time of measurement.

4. Project Risk Management

When a project is being done, many issues can go wrong (risks) and there is the possibility that the project can fail.

With risk management, all the risks that can become real are defined, estimate how the severity will be for the project and finally, preventive/corrective measures for lower the negative impact. As it is mentioned in the book fundamentals of project management (4th ed.) written by Joseph Heagney, "risk management is the process of identifying, analyzing and responding to project risk".

4.1 Identification of Possible risks

The possible risks that the project will deal are defined via brainstorming with the team members:

- Bad planning
- Project requirements not clear
- Not equilibrated distribution of the work, one member can have overload of work
- Member not avaliable-travelling
- Member not avaliable-sick
- YuMi robot breaks
- 3D printer occupied
- 3D printer breakdown
- Bad 3D printed parts
- Loss of report documents, files and information
- Loss of robot program code
- Deficient attention of the supervisors
- Laboratory/workshop tools occupied
- Loss of inspiration
- Loss of members enthusiasms
- Bad coordination
- Disputes or arguments between team members
- Not enough experience in robots
- High learning time required
- Bad communication
- Inaccurate schedule time approximation

4.2 Probability and Impact of Risks

The probability of risk occurrence and the negative impact to the project is determined with a risk matrix. A risk matrix is an instrument for managing project risks.

- The probability of a risk being real is estimated in: Low, Medium and High
- The negative impact is estimated also in: Low, Medium and High

With that in mind, a table with the risk, probability and impact information is build.

Risk	Probability	Impact
Bad planning	**Medium**	**Medium**
Project requirements not clear	**Low**	**Low**
Not equilibrated distribution of the work, one member can have overload of work	**Low**	**Low**
Member not available-travelling	**Medium**	**Medium**
Member not available-sick	**Low**	**Medium**
YuMi robot breaks	**Low**	**High**
3D printer occupied	**High**	**Low**
3D printer breakdown	**Low**	**Low**
Bad 3D printed parts	**High**	**Low**
Loss of report documents, files and information	**Medium**	**High**
Loss of robot program code	**Medium**	**High**
Deficient attention of the supervisors	**Low**	**High**
Laboratory/workshop tools occupied	**Low**	**Low**
Loss of inspiration	**Low**	**Medium**
Loss of team members enthusiasms	**Low**	**Medium**
Bad coordination	**Medium**	**Low**
Disputes or arguments between team members	**Low**	**Low**
Not enough experience in robots	**High**	**Medium**
High learning time required	**High**	**Medium**
Bad communication	**Low**	**Low**
Inaccurate schedule time approximation	**Medium**	**Medium**

Table 15 - Risk, Probability and Impact Table. Source: Own Elaboration

Once the chart is completed, the risk matrix is built.

The risk matrix of ABB's YuMi robot project is shown in the following figure:

Impact \ Probability	Low	Medium	High
High	YuMi robot breaks Deficient attention of the supervisors	Loss of report documents, files and information Loss of robot program code	
Medium	Member not available-sick Loss of inspiration Loss of team members enthusiasms	Bad planning Member not available-travelling Inaccurate schedule time approximation	Not enough experience in robots High learning time
Low	Project requirements not clear Not equilibrated distribution of the work 3D printer breakdown Laboratory/workshop tools occupied Disputes or arguments between team members Bad communication	Bad coordination	3D printer occupied Bad 3D printed parts

Table 16 - Risk Management of YuMi Project. Source: Own Elaboration

4.3 Response to the Risks

For the risks that are more severe, a series of actions are defined to prevent or mitigate the negative impact for the project.

Risk:	**YuMi robot breaks.**
Impact level:	**High.**
Measures:	**Replace the part or parts as fast it can be possible. Meanwhile the robot is stopped, invest the time and resources in other project tasks.**

Risk:	**Loss of report documents, files and information.**
Impact:	**High.**
Measures:	**Do daily backups on USB memories and cloud system like Drop box.**

Risk:	**Loss of robot program code.**
Impact:	**High.**
Measures:	**Do daily backups on USB memories and cloud system like Drop box.**

Risk:	**Deficient attention of the supervisors.**
Impact:	**High.**
Measures:	**Be persistent on the demands and plan meetings with them regularly.**

Risk:	**Bad planning.**
Impact:	**Medium.**
Measures:	**More communication between team members and review the schedule constantly.**

Risk:	**Member not available-travelling.**
Impact:	**Medium.**

Measures:	**The member will have to finish his tasks before leaving or work extra hours to accomplish his duty.**

Risk:	**Member not available-sick.**
Impact:	**Medium.**
Measures:	**Another member of the group will have to do his task and if it is necessary, postpone the assignment.**

Table 17 - Risks and Measures Table. Source: Own Elaboration

Risk:	**Loss of inspiration.**
Impact:	**Medium.**
Measures:	**Disconnect of the project for a short period. Taking a break, doing sport or seeing nature can help.**

Risk:	**Loss of team members enthusiasms.**
Impact:	**Medium.**
Measures:	**Motivation between towards the team members that has lost the interest.**

Risk:	**Not enough experience in robots.**
Impact:	**Medium.**
Measures:	**Learning efforts and practise will be done in Technobothnia laboratory and EPS room. Ask for support to the supervisor or between team members.**

Risk:	**High learning time required.**
Impact:	**Medium.**
Measures:	**Do extra hours. Delay the schedule. Ask for support to the supervisor or between team members.**

Risk:	**Inaccurate schedule time approximation.**
Impact:	**Medium.**
Measures:	**More communication between the group components. Review the tasks and update the schedule.**

Table 18 - Risks and Measure Table (Continuation). Source: Own Elaboration

5. Data Sheets

5.1 Printer Data Sheet

ZX1200i / ZX1300i / ZX1600i Specifications

Model		ZX1200i	ZX1300i	ZX1600i
Print Method		Thermal Transfer / Direct Thermal		
Resolution		203 dpi (8 dots/mm)	300 dpi (12 dots/mm)	600 dpi (24 dots/mm)
Print Speed		Up to 10 IPS (254 mm/s)	Up to 7 IPS (177 mm/s)	Up to 4 IPS(101.6 mm/s)
Print Width		4.09" (104 mm) Up to (108mm)	4.09" (104 mm) Up to (105.7mm)	4.09" (104 mm) Up to (105.6mm)
Print Length		Min. 0.16" (4 mm)** – Max. 180" (4572 mm)	Min. 0.16" (4 mm)** – Max. 85" (2159 mm)	Min. 0.16" (4 mm) ** – Max. 26" (660 mm)
Processor		32 bit RISC CPU		
Memory	Flash	128 MB Flash (60 MB for user storage)		
	SDRAM	32 MB		
Sensor Type		Adjustable reflective sensor and transmissive sensor, left aligned		
Media	Type	Continuous form, gap labels, black mark sensing, and punched hole; label length set by auto sensing or programming		
	Width	Tear : Min. 1" (25.4 mm) – Max. 4.64" (118 mm) Cutter : Max. 4.61" (117 mm) Dispenser / Rewind : Max. 4.64" (118 mm)		
	Thickness	Min. 0.003" (0.06 mm) – Max. 0.01" (0.25 mm)		
	Label roll diameter	Max. 8" (203.2 mm)		
	Core diameter	Min. 1.5" (38.1 mm) – Max. 3" (76.2 mm)		
Ribbon	Types	Wax, wax/resin, resin		
	Length	Max. 1476' (450 m)		
	Width	Min. 1.18" (30 mm) – Max. 4.33" (110 mm)		
	Ribbon roll diameter	3" (76.2 mm)		
	Core diameter	1" (25.4 mm)		
Printer Language		EZPL, GEPL, GZPL auto switch		
Software	Label design software	GoLabel (for EZPL only)		
	Driver	MAC , Linux , Windows 2000 / XP / VISTA / Windows 7 / Windows 8.1		
	DLL	Win CE , .NET , Windows Mobile, Windows 2000 / XP / VISTA / Windows 7 / Windows 8.1 / Android / Windows 10		
Resident Fonts	Bitmap Fonts	6, 8, 10, 12, 14, 18, 24, 30, 16X26 and OCR A&B Bitmap fonts 90°, 180°, 270° rotatable, single characters 90°, 180°, 270° rotatable Bitmap fonts 8 times expandable in horizontal and vertical directions		
	TTF Fonts	TTF fonts (Bold / Italic / Underline). 0°, 90°, 180°, 270° rotatable		
Download Fonts	Bitmap Fonts	Bitmap fonts 90°, 180°, 270° rotatable, single characters 90°, 180°, 270° rotatable		
	Asian Fonts	16x16, 24x24, Traditional Chinese (BIG-5), Simplified Chinese(GB2312), Japanese (S-JIS), Korean (KS-X1001) 90°, 180°, 270° rotatable and 8 times expandable in horizontal and vertical directions		
	TTF Fonts	90°, 180°, 270° rotatable		
Barcodes	1-D Bar Codes	Code 39, Code 93, EAN 8/13 (add on 2 & 5), UPC A/E (add on 2 & 5), I 2 of 5 & I 2 of 5 with Shipping Bearer Bars, Codabar, Code 128 (subset A, B, C), EAN 128, RPS 128, UCC 128, UCC/EAN-128 K-Mart, Random Weight, Post NET, ITF 14, China Postal Code, HIBC, MSI, Plessey, Telepen, FIM, GS1 DataBar, German Post Code, Planet 11 & 13 digit, Japanese Postnet, I2 of 5 with human readable check digit, Standard 2 of 5, Industrial 2 of 5 , Logmars, Code 11, Code 49, Cadablock		
	2-D Bar Codes	PDF417, Micro PDF417, Datamatrix code, Maxicode, QR code, Micro QR code and Aztec code		
Code Pages		Codepage 437, 850, 851, 852, 855, 857, 860, 861, 862, 863, 865, 866, 869, 737 Windows 1250, 1251, 1252, 1253, 1254, 1255, 1257 Unicode UTF8, UTF16BE , UTF16LE		
Graphics		Resident graphic file types are BMP and PCX, other graphic formats are downloadable from the software		
Interfaces		USB 2.0 (B-Type) Serial port: RS-232 (DB-9) IEEE 802.3 10/100 Base-Tx Ethernet port (RJ-45) 3 USB Host (A Type) 2 ports at the front panel, 1 port at the rear panel		
Control Panel		Backlight 3.2" touch screen LCD 1 Power on/off button with green color LED backlight 1 Control key : FEED / PAUSE / CANCEL with dual color LED backlight: Ready (Green); Error (Red) 1 Calibration button at rear panel		
Real Time Clock		Standard		
Power		Switching power 100-240VAC, 50-60Hz input		
Environment	Operation temperature	41°F to 104°F (5°C to 40°C)		
	Storage temperature	-4°F to 140°F (-20°C to 60°C)		
Humidity	Operation	20-85%, non-condensing		
	Storage	10-90%, non-condensing		
Agency Approvals		CE (EMC) · FCC Class A · CB · UL · cUL · CCC · KC		
Dimension	Length	18.30" (465 mm)		
	Height	12.13" (308.20 mm)		
	Width	10.65" (270.71 mm)		
Weight		30 lbs (13.6 Kg), excluding consumables		
Options		Cutter Parallel port adaptor module (Centronic female 36-pin) Bluetooth WiFi print server module (IEEE 802.11b/g/n) Applicator Interface (DSUB female 15-pin) External label rewinder Label Dispenser + Internal Rewinder		

* Specifications are subject to change without notice. All company and/or product names are trademarks and/or registered trademarks of their respective owners.

** Minimum print height and maximum print speed specification compliance can be dependent on non-standard material variables such as label type, thickness, spacing, liner construction, etc. Godex is pleased to test non-standard materials for minimum print height and maximum print speed capability.

Table 19 - Label Machine Data Sheet

5.2 IRB 1600 Robot Data Sheet

Specification

Robot version IRB	Reach (m)	Payloads (kg)	Armload (kg)
IRB 1600 -6/1.2	1.2	6	30.5
IRB 1600 -6/1.45	1.45	6	30.5
IRB 1600 -10/1.2	1.2	10	20.5
IRB 1600 -10/1.45	1.45	10	20.5
Number of axes	6+3 external (up to 36 with MultiMove)		
Protection	Standard IP54; opt. FoundryPlus 2 (IP 67)		
Mounting	Floor, wall, shelf, tilted, inverted		
Controller	IRC5 Single cabinet, IRC5 Dual cabinet, IRC5 Compact		

Technical information

Electrical Connections	
Supply voltage	200-600 V, 50-60 Hz
Energy consumption	0.58 kW
Physical	
Dimensions robot base	484 x 648 mm
Height	
IRB 1600-6/1.2	1069 mm
IRB 1600-10/1.2	1069 mm
IRB 1600-6/1.45	1294 mm
IRB 1600-10/1.45	1294 mm
Weight	250 kg

Performance

	Position repeatability RP (mm)	Path repeatability RT (mm):
IRB 1600 -6/1.2	0.02	0.13
IRB 1600 -6/1.45	0.02	0.19
IRB 1600 -10/1.2	0.02	0.06
IRB 1600 -10/1.45	0.05	0.13

IRB 1600-6/1.45

Axis movement	Working range	Axis max speed
Axis 1	+180° to -180°	180°/s
Axis 2	+150° to -90°	180°/s
Axis 3	+65° to -245°	185°/s
Axis 4	+200° to -200° def. +/-190° revolution	385°/s
Axis 5	+400° to -400° def.	400°/s
Axis 6	+/-288 revolution	460°/s

Table 20 - IRB 1600 Data Sheet

6. CAD Drawings

6.1 YuMi Drawings

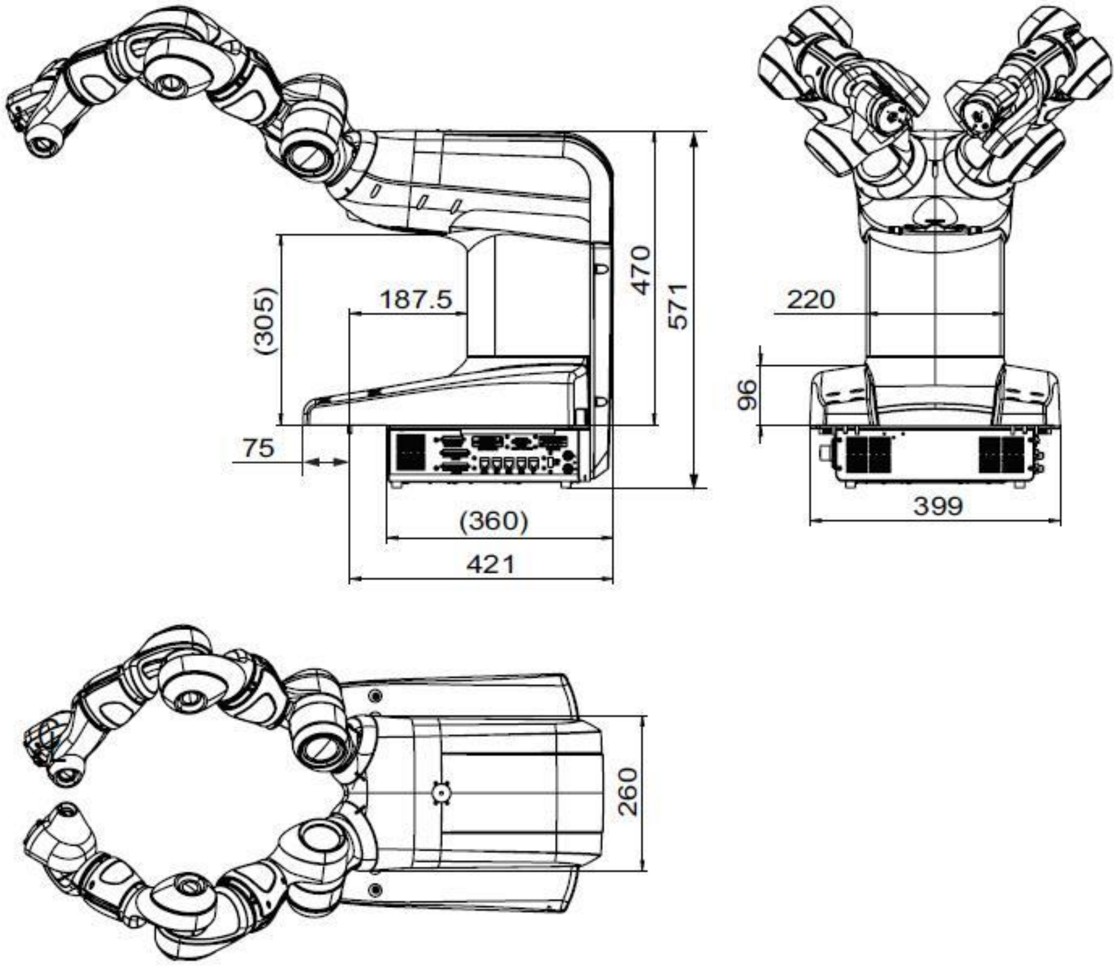


Figure 101 - Full YuMi Body Dimensions

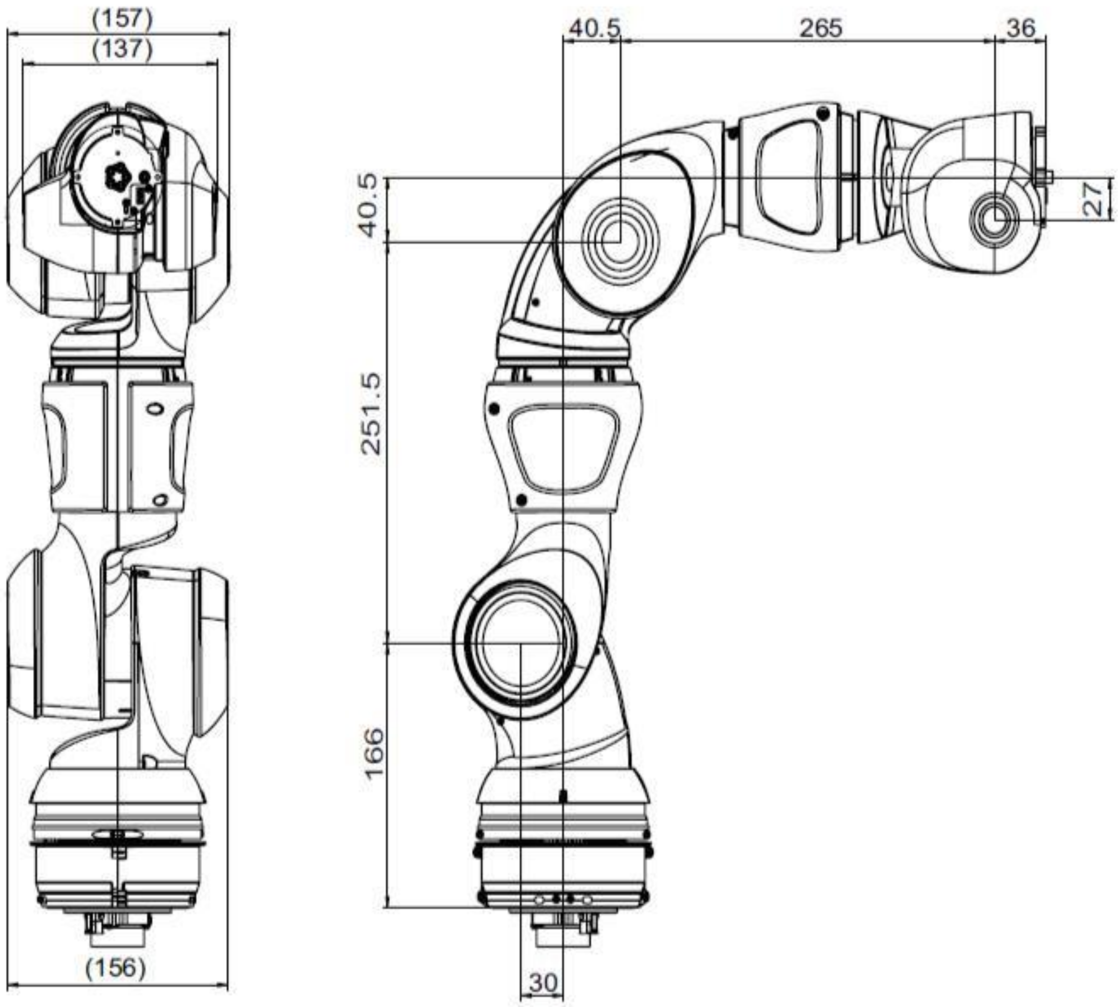


Figure 102 - YuMi Arm Dimensions

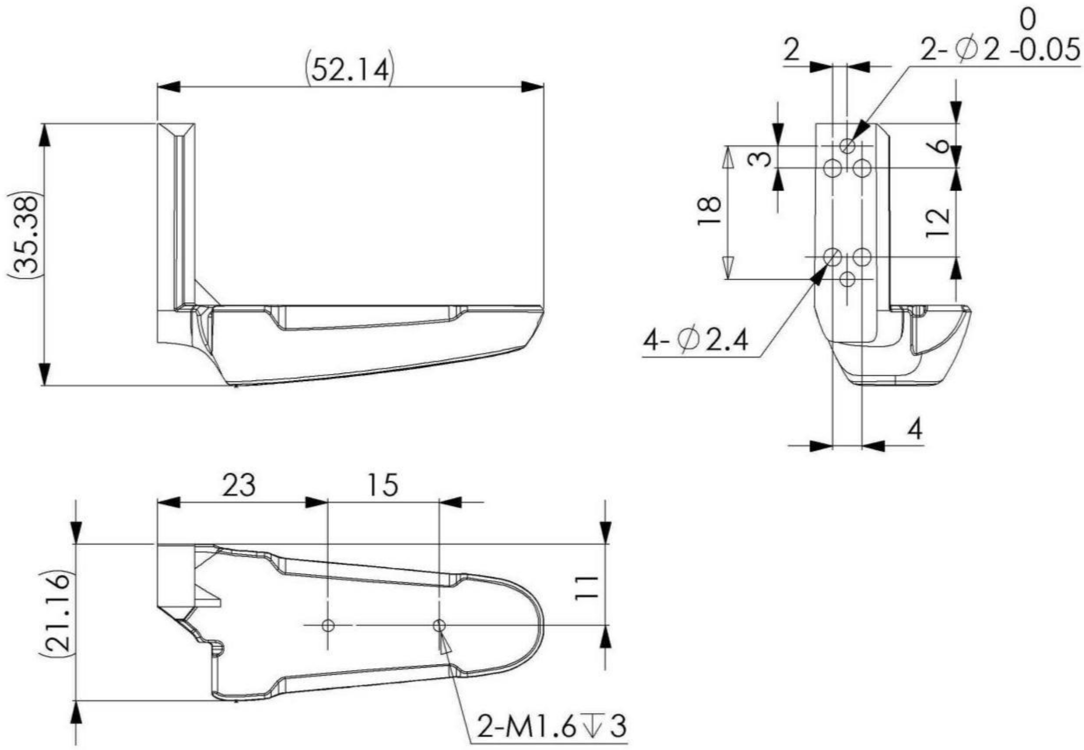


Figure 103 - YuMi Gripper Dimensions

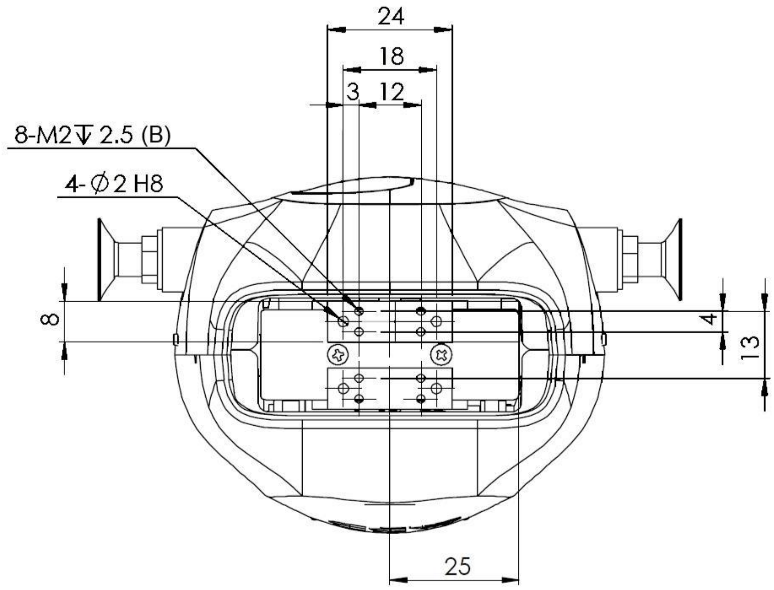


Figure 104 - YuMi Hand Dimensions

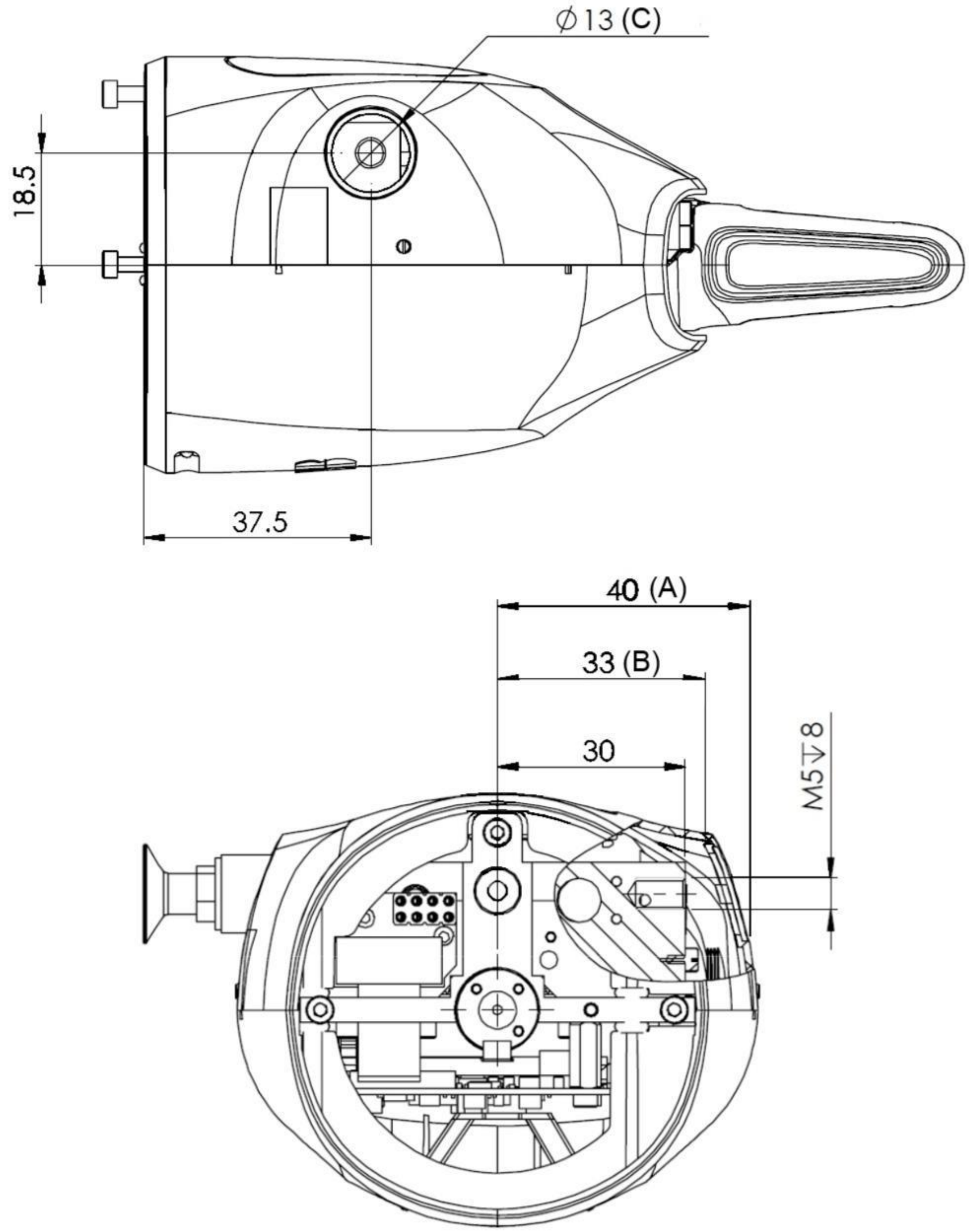


Figure 105 - YuMi Hand Dimensions 2

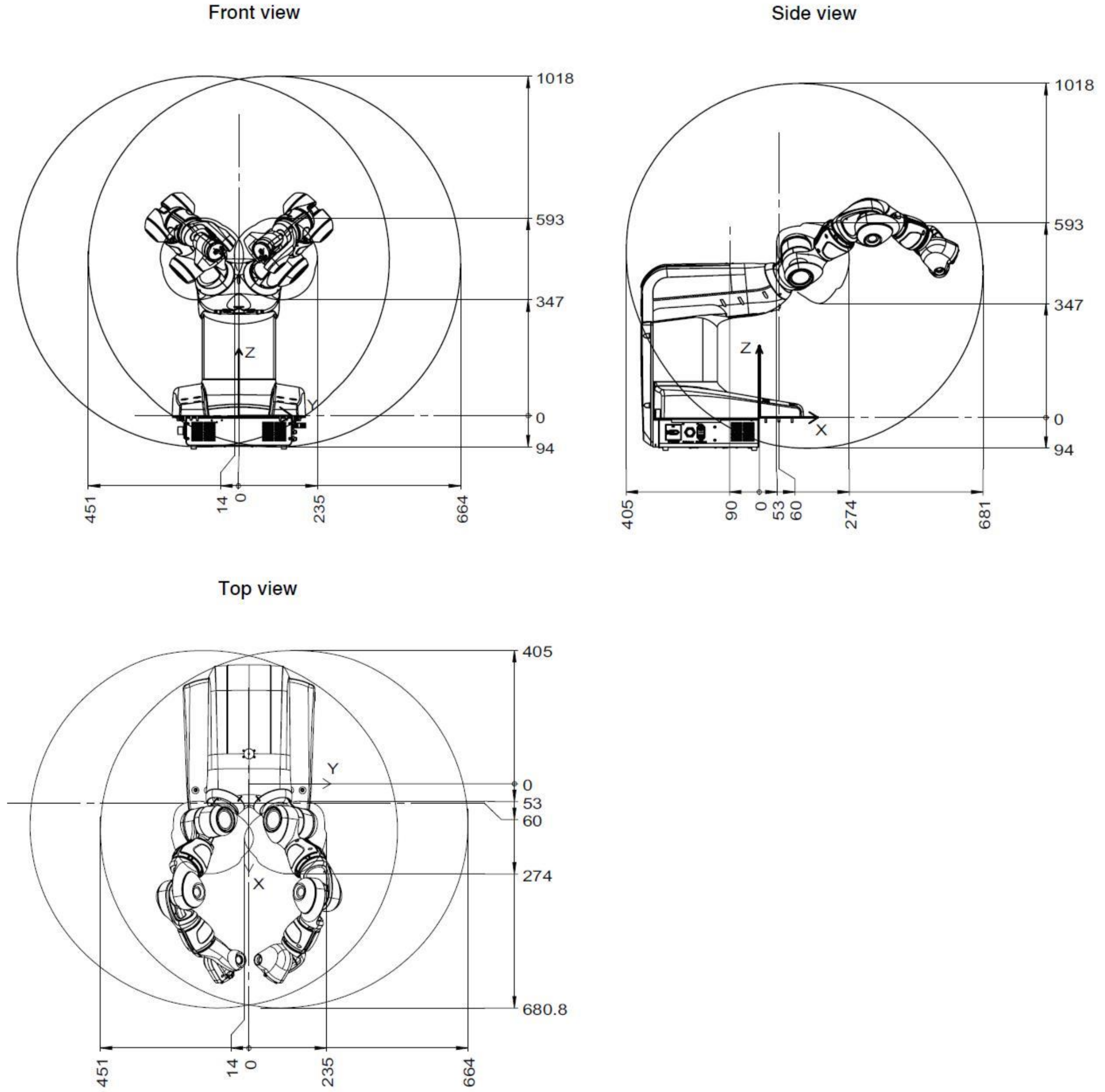


Figure 106 - YuMi Reach

6.2 Component Drawings

6.2.1 Relion Case 611and 615

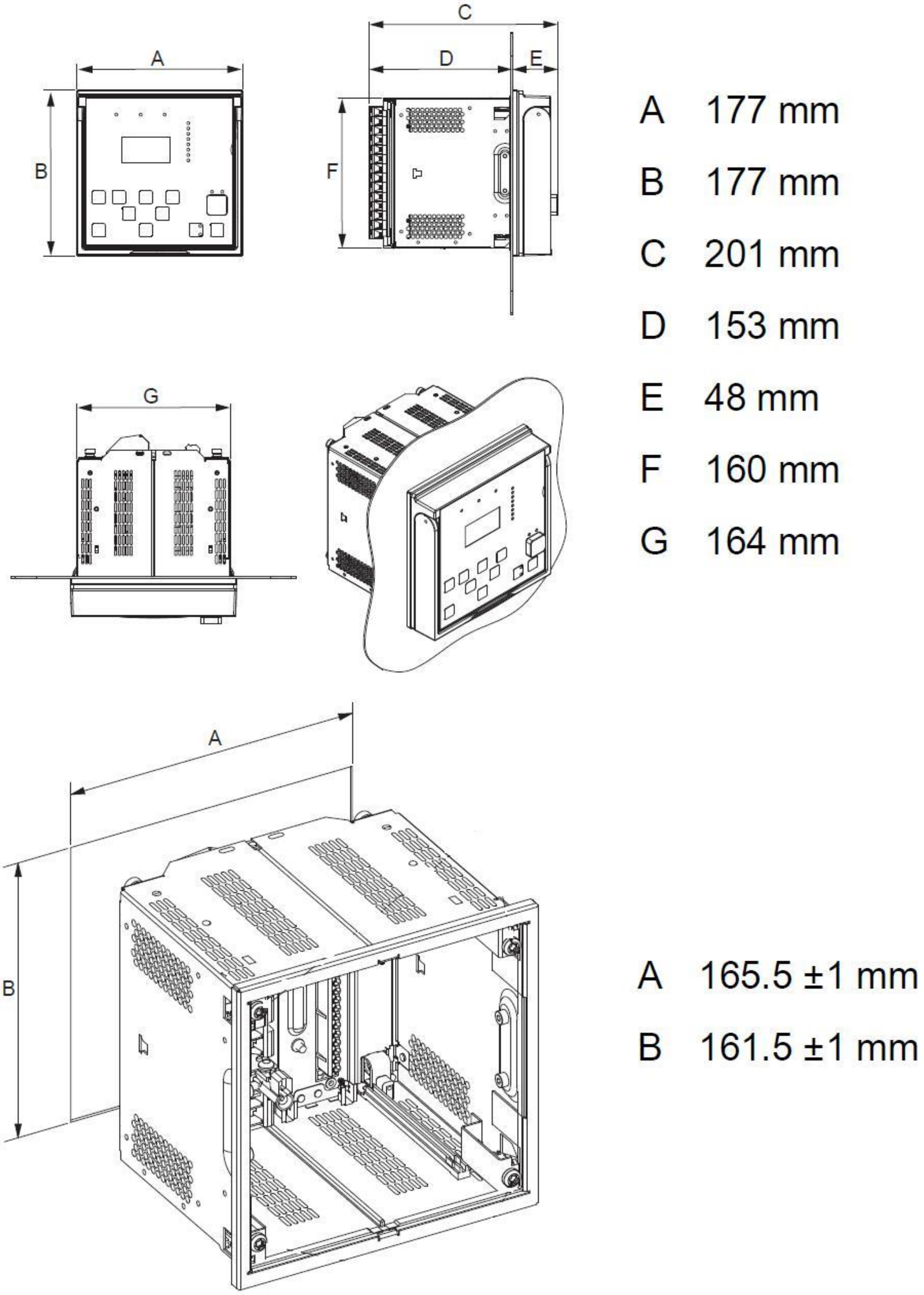


Figure 107 - Relion Case 611 and 615 Dimensions

6.2.2 Relion 620 Case

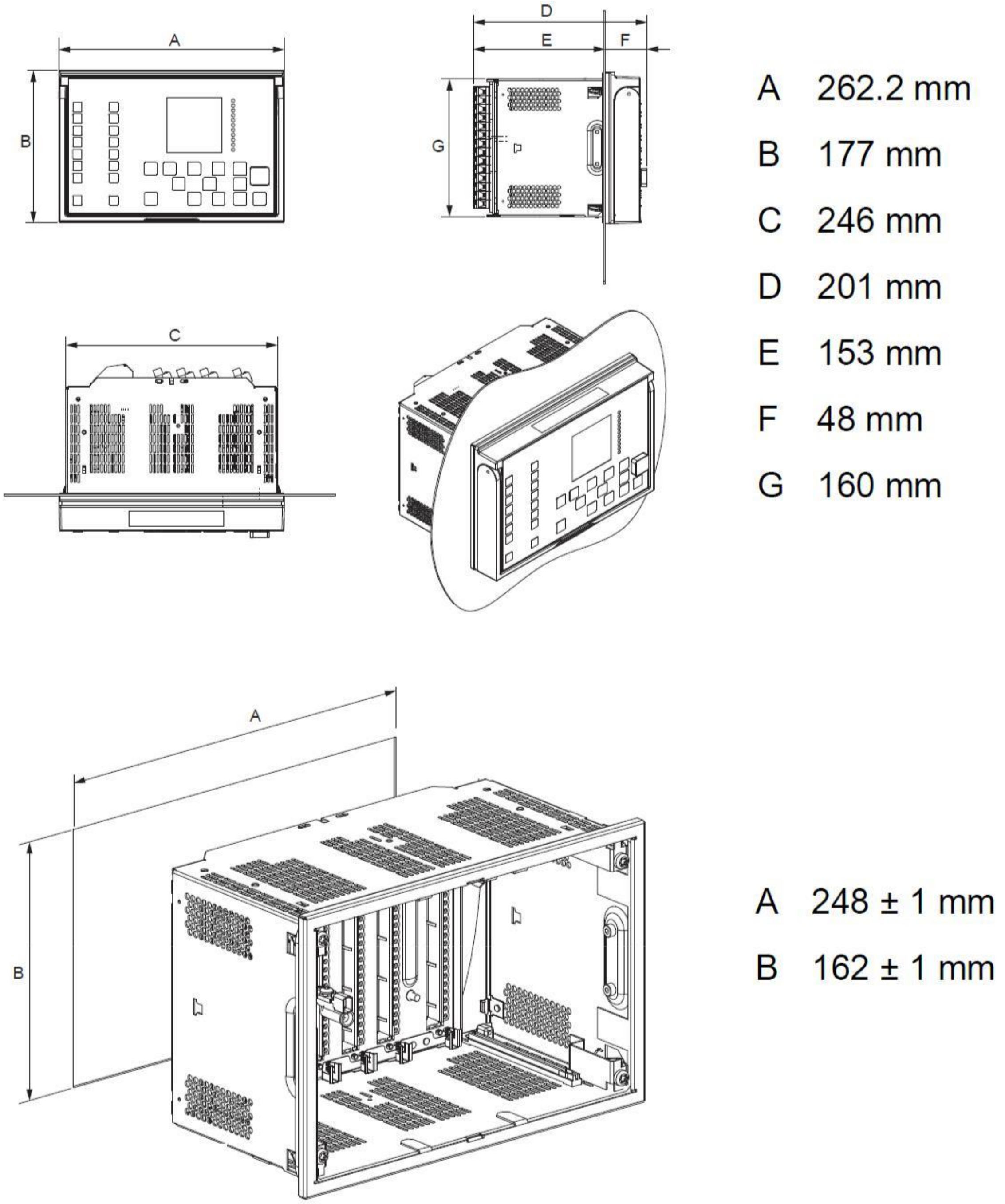


Figure 108 - Relion 620 Case Dimensions

6.2.3 Gripper Drawings

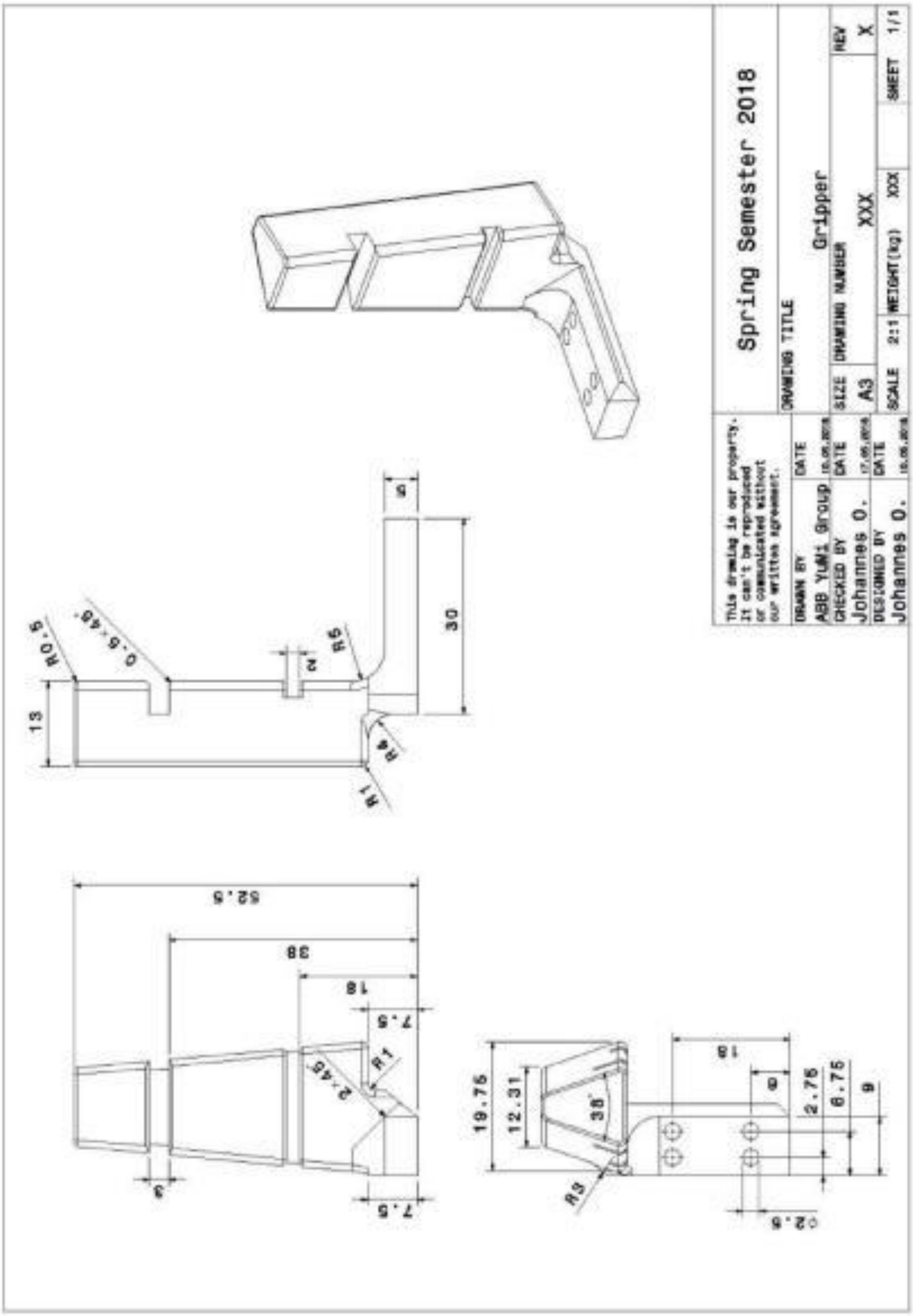


Figure 109 - ABB Gripper Design Drawings

6.2.4 Jig Drawings

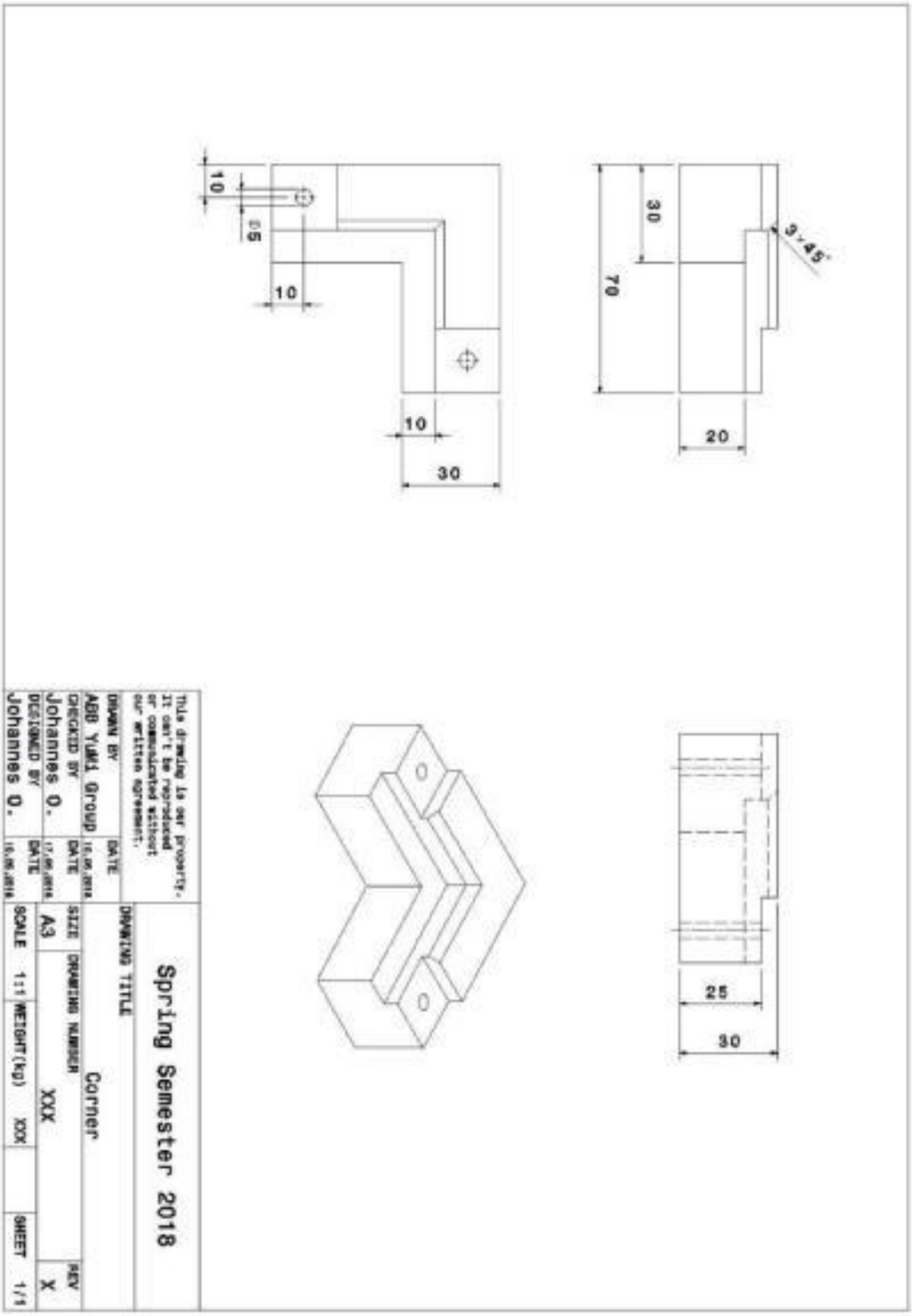


Figure 110 - Jig Corner Drawings

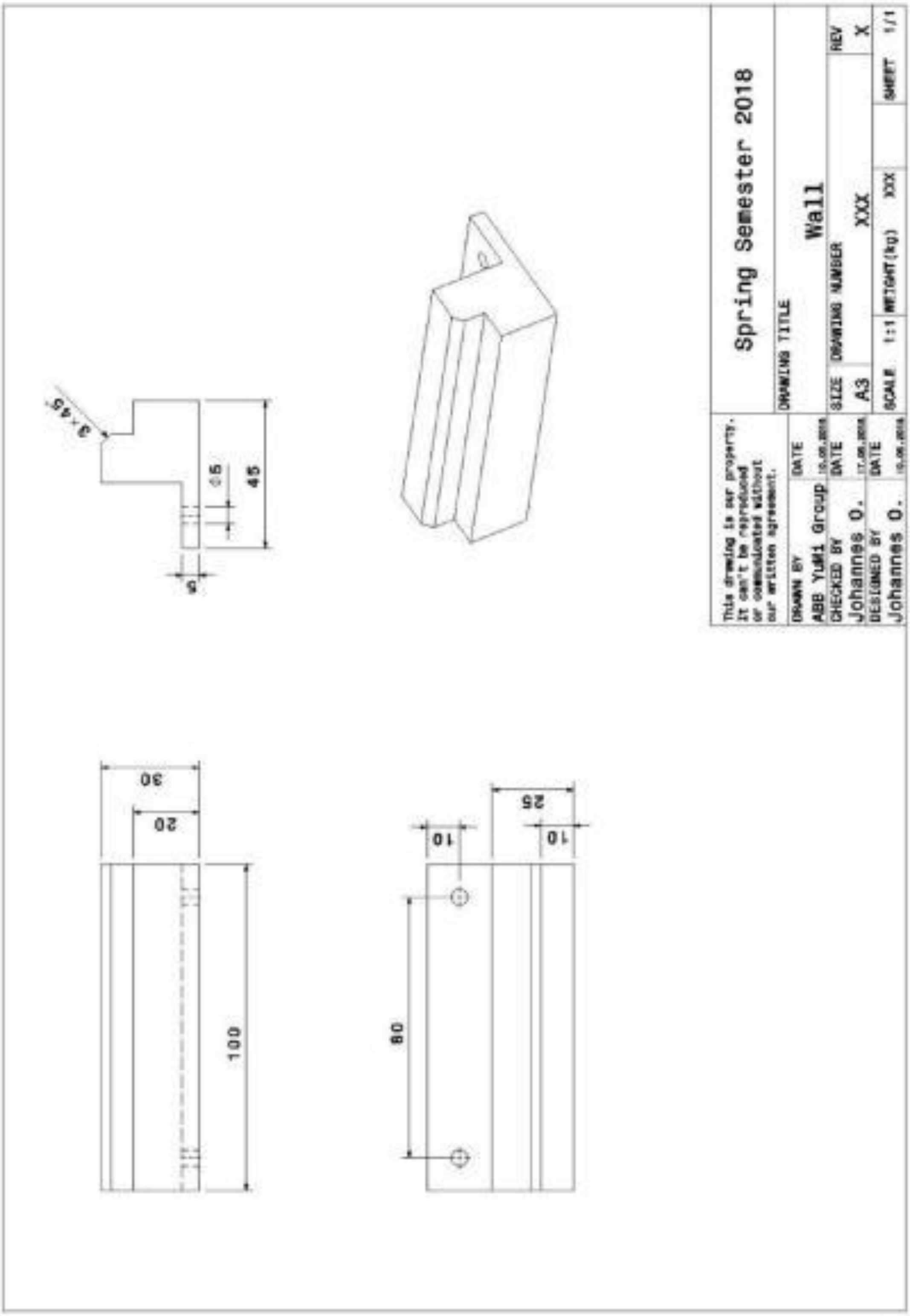


Figure 111 - Jig Wall Drawings

6.2.5 Support Drawings

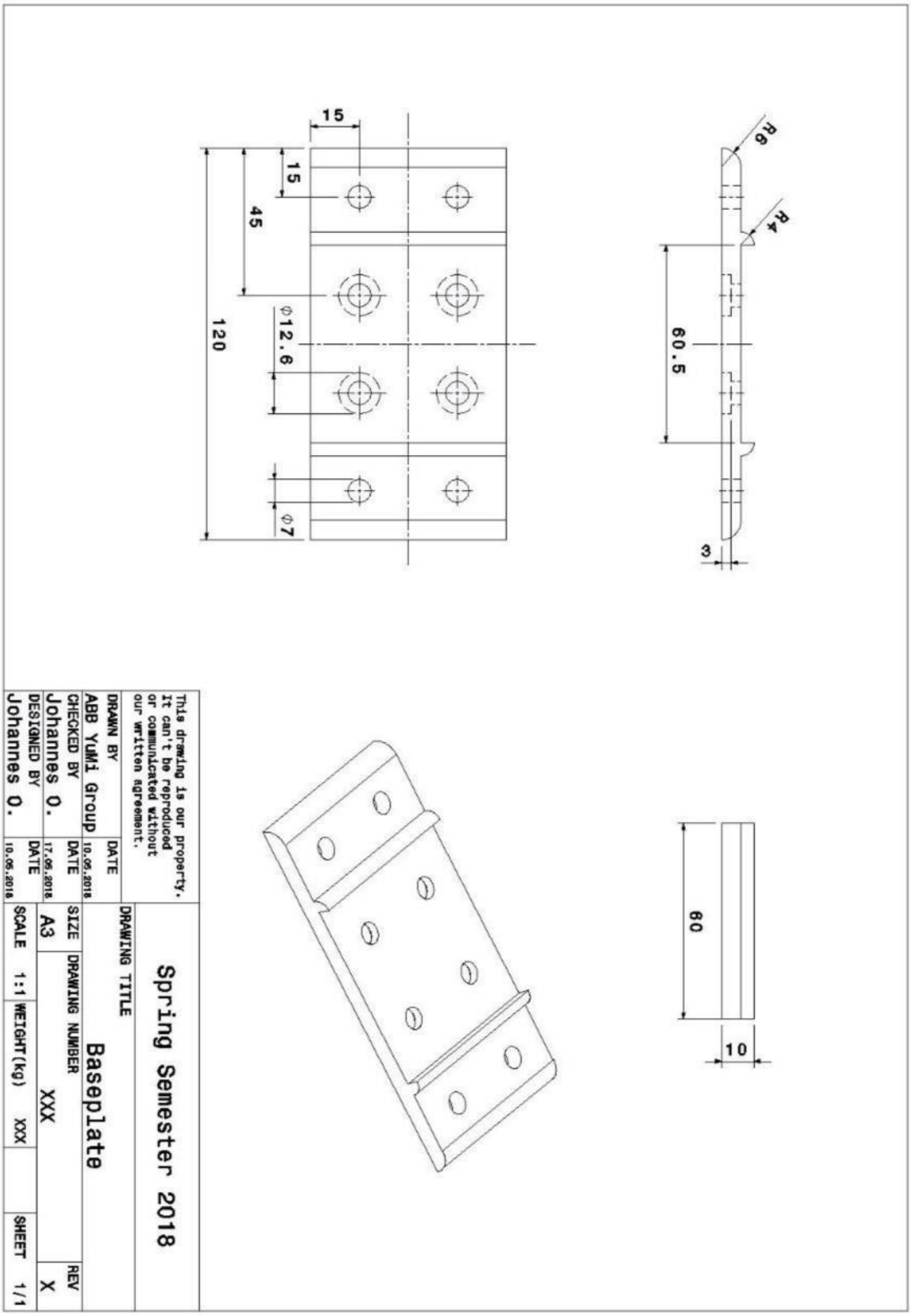


Figure 112 - Baseplate for Support Drawings

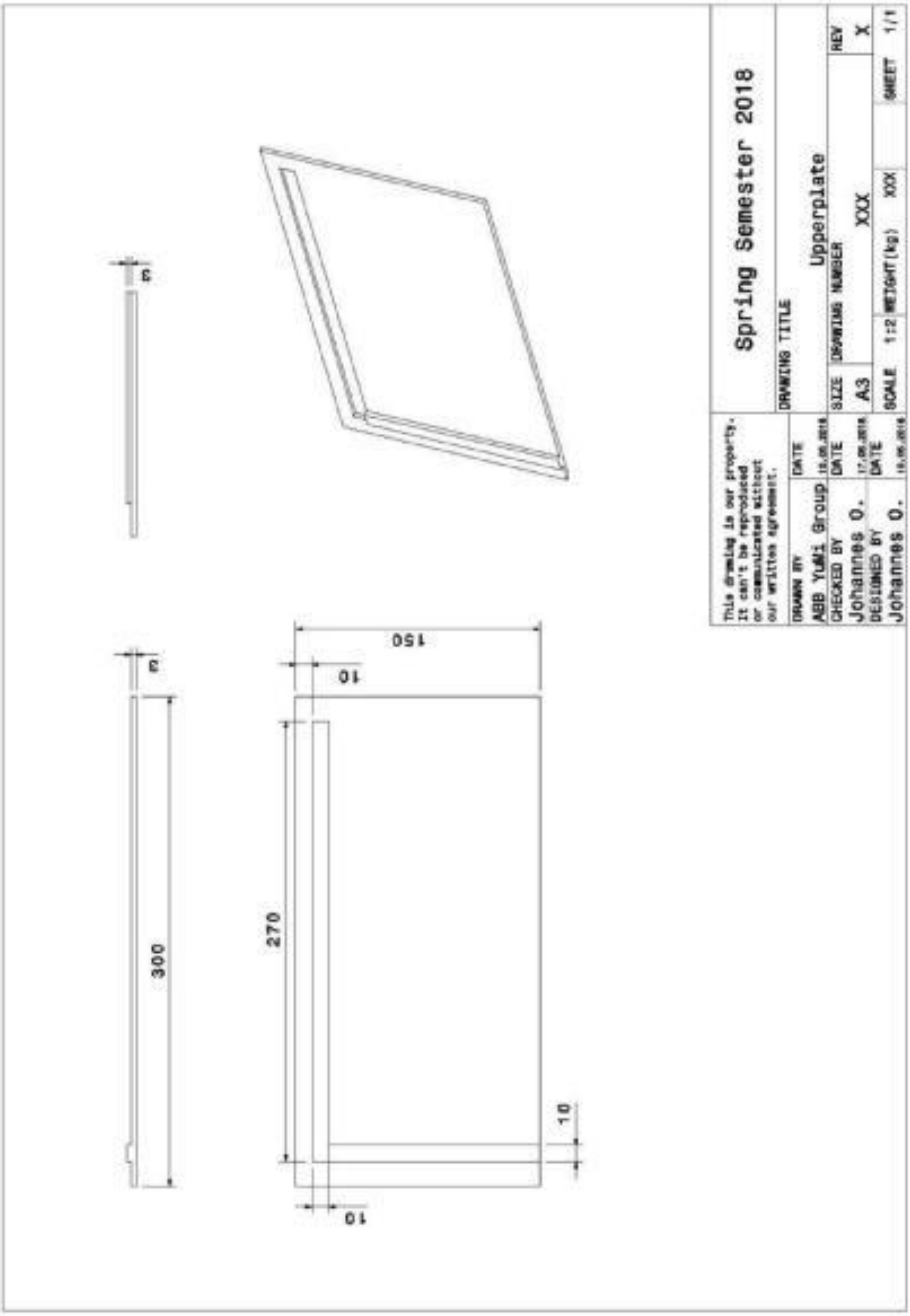


Figure 113 - Upperplate for Support Drawings

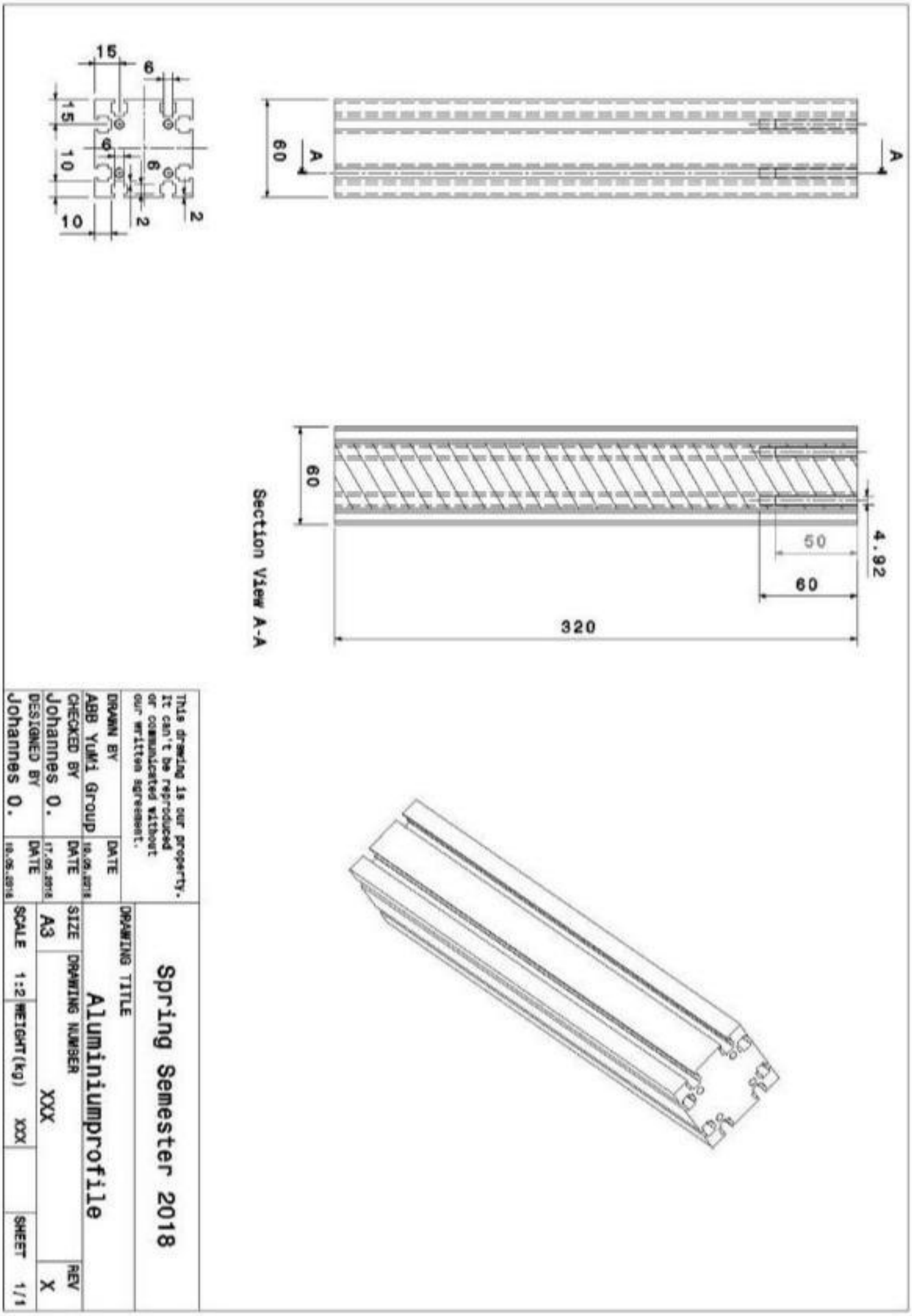


Figure 114 - Aluminium Profile for Support Drawings

6.2.6 Sticker and Suction Cup

The dimension of the YuMi default suction cup is Ø 16mm.

Figure 115 - Suction Cup Dimension. Source: ABB

The different models of yellow warning stickers have the same dimensions. The dimensions of the stickers are:

Figure 116 - Sticker Dimensions. Source: Own Elaboration

With all of that, the suction cup will be able to pick up the three different stickers

6.3 IRB 1600 Drawing

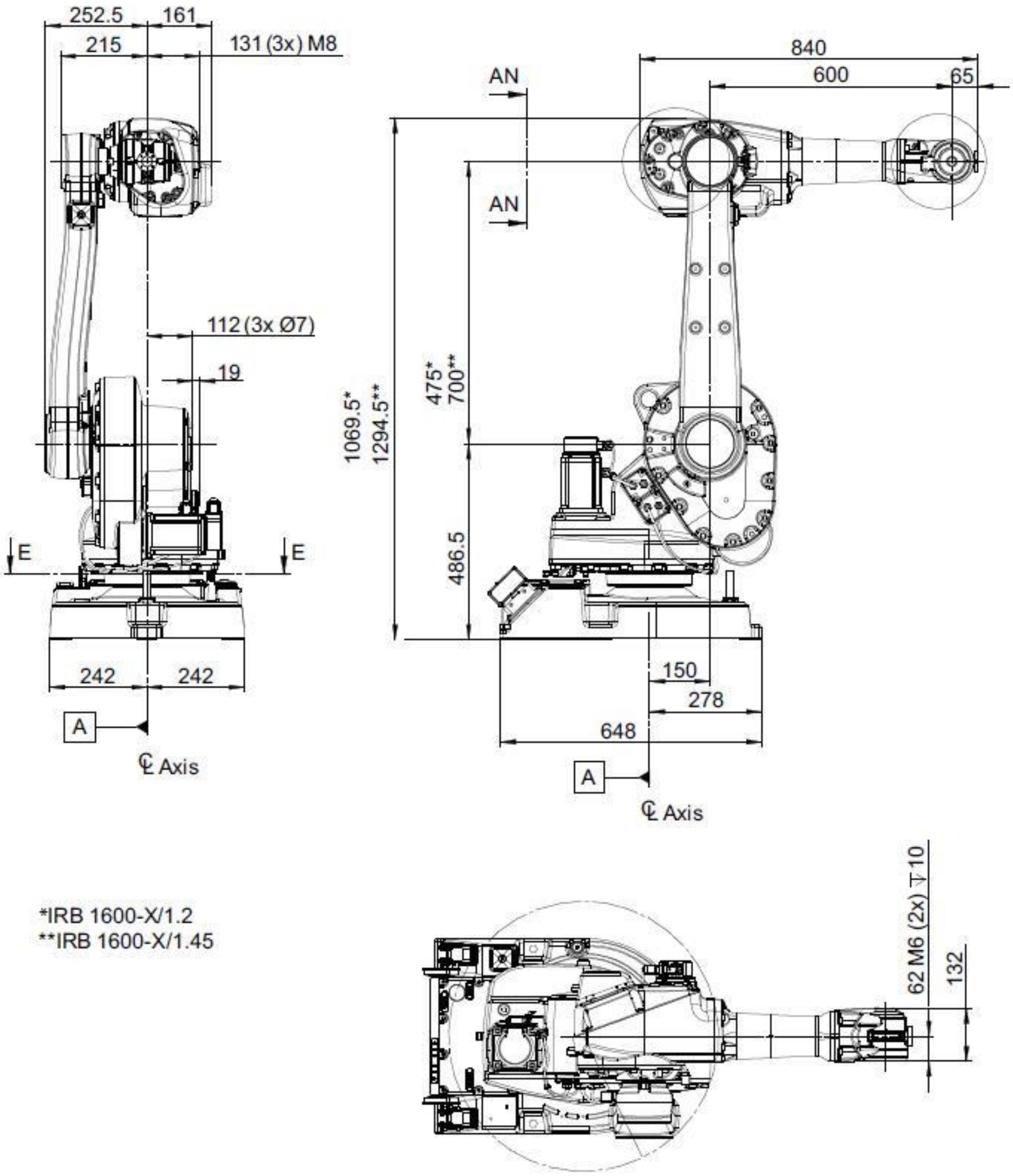


Figure 117 - IRB 1600 Drawings

IRB 1600-x/1.45

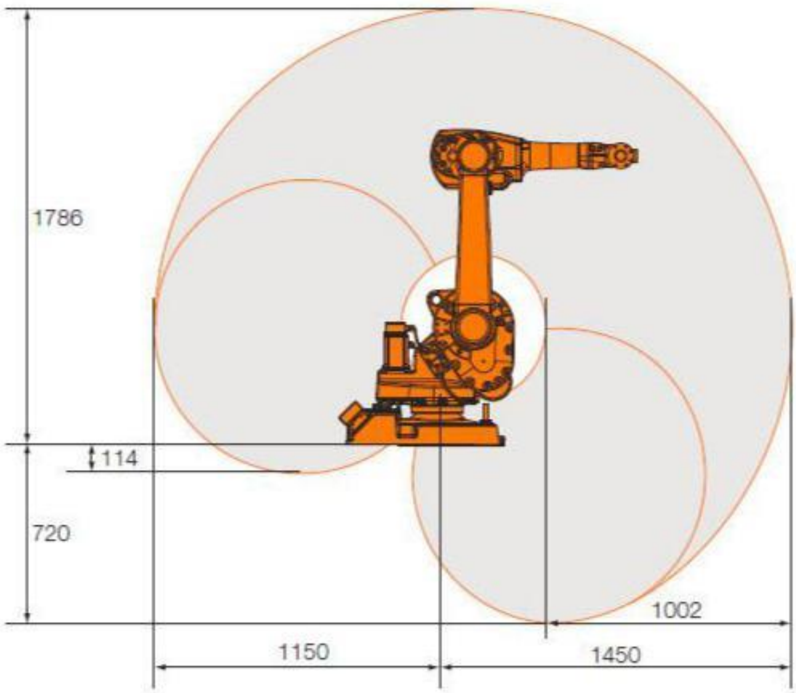


Figure 118 - IRB 1600 Reach

7. Coding

7.1 Large Box

7.1.1 Left Arm

```
MODULE MainModule
    PERS tasks taskslist1{2}:=[["T_ROB_L"],["T_ROB_R"]];
    VAR syncident sync1;
    VAR syncident sync2;
    VAR syncident sync3;
    VAR syncident sync4;
      CONST robtarget
pHome:=[[534.22,122.96,379.18],[0.526311,0.538683,0.564936,-
0.337144],[-1,0,1,11],[-170.038,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPlace:=[[534.42,125.99,64.67],[0.00704159,-
0.999774,0.0158698,-0.0122567],[-
2,1,1,11],[170.199,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pHomeConnector:=[[480.72,299.50,97.18],[0.000941248,0.99941,-
0.0058208,0.0338478],[-
1,1,1,11],[166.841,9E+09,9E+09,9E+09,9E+09,9E+09]];
    PERS loaddata Maxload:=[0.20,[0,0,1],[1,0,0,0],0,0,0];
    PERS loaddata Minload:=[0.02,[0,0,1],[1,0,0,0],0,0,0];
    PERS loaddata Connectorload:=[0.3,[0,0,1],[1,0,0,0],0,0,0];
    CONST robtarget
pHomeLabeling:=[[410.01,211.92,379.20],[0.626518,0.28075,0.685098,-
0.243504],[-1,-1,2,1010],[172.821,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPickLabel1:=[[456.40,145.43,168.32],[0.682835,0.069835,0.7
25676,-0.0474824],[-1,-2,2,11],[-
169.715,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPickLabel2:=[[456.40,175.43,168.32],[0.682835,0.069835,0.7
25676,-0.0474824],[-1,-2,2,11],[-
169.715,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel1:=[[575.01,91.75,163.20],[0.714789,0.0294322,0.698525,-
0.0165228],[-2,-2,2,11],[-175.139,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceLabel2:=[[769.60,10.93,327.12],[0.697766,0.0264131,0.715828,-
0.0039206],[-2,-2,2,11],[-133.863,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pStickLabel2:=[[776.11,17.92,292.81],[0.680013,0.0217463,0.732851,0
.00625558],[-2,-2,2,11],[-133.808,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pStickLabel1:=[[769.09,-
18.31,295.44],[0.68474,0.0342238,0.72672,-0.0428727],[0,-2,2,0],[-
120.299,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPlaceLabel2:=[[594.72,10.19,327.77],[0.698905,0.0254903,0.
714754,-0.00297448],[-1,-2,2,11],[-
133.92,9E+09,9E+09,9E+09,9E+09,9E+09]];
```

```
    CONST robtarget
pPickLabel2a:=[[709.21,120.05,159.45],[0.750701,0.00414234,0.660609
,0.00516011],[-2,-2,2,11],[178.067,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel2b:=[[584.91,118.92,119.52],[0.696426,-
0.0200795,0.717161,0.0163684],[-2,-
2,2,11],[177.79,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPickLabel3:=[[429.27,150.57,140.47],[0.708839,0.0355911,0.
704454,-0.00493695],[-1,-2,2,11],[-
170.886,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel3a:=[[591.60,138.37,139.24],[0.697272,0.00314585,0.713685 ,0.0667421],[-
2,-2,2,11],[-179.907,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel3b:=[[587.17,153.82,117.15],[0.703603,0.0217157,0.710129,
0.0137136],[-2,-2,2,11],[-179.358,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pTurnLabel3:=[[449.47,188.18,163.22],[0.424538,-0.520465,0.489666,-
0.555978],[-2,-2,1,11],[-126.417,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPlaceLabel3:=[[669.10,263.22,340.00],[0.492811,-
0.515145,0.474018,-0.516789],[-2,-2,0,11],[-
136.817,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceLabel3:=[[668.76,214.10,348.46],[0.446055,-
0.524034,0.569145,-0.449997],[-2,-2,1,11],[-
139.082,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPickout:=[[535.60,134.65,91.06],[0.0237871,-
0.999367,-0.0174926,-0.019847],[-1,1,1,11],[-
155.565,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPushConnector:=[[342.70,137.70,126.11],[0.0198139,0.998769,0.04109
41,0.0194694],[-1,1,1,11],[172.972,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPlaceout:=[[535.60,389.65,91.06],[0.0237871,-
0.999367,-0.0174926,-0.019847],[-1,1,1,11],[-
155.565,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pApproachPlaceLabel1:=[[594.15,-
14.76,322.87],[0.679389,0.0402471,0.730951,-0.0502113],[-2,-
2,2,0],[-121.885,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPlaceLabel1:=[[765.97,-
18.89,321.99],[0.678664,0.0417587,0.731557,-0.0499518],[0,-
2,2,0],[-121.195,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPickBox:=[[553.44,-
142.24,65.70],[0.00236511,0.999716,0.00226172,0.0236182],[-
2,1,1,11],[170.17,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceBox:=[[540.82,121.99,20.17],[0.0464419,0.996707,0.00264226,0.
0664245],[-2,1,1,11],[174.178,9E+09,9E+09,9E+09,9E+09,9E+09]];


    PROC main()
        MoveJ pHome,v1000, fine, GripperL;
```

```
 Hand_Initialize\maxSpd:=20\holdForce:=30\Calibrate;
 Hand_GripInward;
 Labelling;
 Move_Relayboxin;
 Connector;
 Move_Relayboxout;
ENDPROC


PROC labelling ()
     MoveJ pHomeLabeling, v1000, fine, tool0;
     MoveJ pApproachPickLabel1, v500, fine,
     tool0; MoveJ pPickLabel1, v500, fine, tool0;
     Hand_TurnOnVacuum1;
     MoveL Offs(pPickLabel1,0,0,-
     48),v50,fine,tool0; WaitTime 1;
     MoveL pPickLabel1, v500, fine, tool0;
     MoveJ pApproachPickLabel1,v500,fine,tool0;
     MoveJ pApproachPlaceLabel1, v500, fine, GripperL;
     MoveL pPlaceLabel1, v200, fine, GripperL;
     MoveL Offs(pPlaceLabel1,0,0,-
     30),v50,fine,GripperL; WaitTime 1;
     Hand_TurnOffVacuum1;
     MoveL Offs(pStickLabel1,5,3,0),v50,fine,GripperL;
     MoveL pApproachPlaceLabel1, v500, fine, GripperL;
     MoveJ pApproachPickLabel2, v500, fine, GripperL;
     MoveJ pPickLabel2a, v500, fine, GripperL;
     MoveL pPickLabel2b, v50, fine,
     tool0; Hand_TurnOnVacuum1;
     WaitTime 1;
     MoveL pPickLabel2a, v500, fine, GripperL;
     MoveJ pApproachPickLabel2, v500, fine, GripperL;
     MoveJ pApproachPlaceLabel2, v500, fine, GripperL;
     MoveL pPlaceLabel2, v200, fine, GripperL;
     MoveL Offs(pPlaceLabel2,0,0,-
     30),v50,fine,GripperL; WaitTime 1;
     Hand_TurnOffVacuum1;
     MoveL Offs(pStickLabel2,-3,-3,0),v50,fine,GripperL;
     MoveL Offs(pStickLabel2,3,3,0),v50,fine,GripperL;
     MoveL pPlaceLabel2, v100, fine, GripperL;
     MoveL pApproachPlaceLabel2, v500, fine, GripperL;
     MoveJ pApproachPickLabel3, v500, fine, tool0;
     MoveJ pPickLabel3a, v500, fine, tool0;
     Hand_TurnOnVacuum1;
     MoveL pPickLabel3b, v500, fine,
     tool0; WaitTime 2;
     MoveL pPickLabel3a, v500, fine, tool0;
     MoveJ pApproachPickLabel3, v500, fine,
     tool0; MoveL pTurnLabel3, v500, fine, tool0;
     MoveL pApproachPlaceLabel3, v500, fine, tool0;
     MoveL pPlaceLabel3, v50, fine, tool0; WaitTime
     1;
     Hand_TurnOffVacuum1;
     MoveL Offs(pPlaceLabel3,3,0,3),v50,fine,tool0;
     MoveL Offs(pPlaceLabel3,-3,0,-3),v50,fine,tool0;
```

```
            MoveJ pApproachPlaceLabel3, v500, fine, tool0;
            WaitDI custom_DI_0, 1;
            MoveJ pHome, v1000, fine, tool0;
            WaitSyncTask sync3,taskslist1;
        ENDPROC

            PROC Move_Relayboxin ()
            Hand_MoveTo 9;
            MoveJ pPickBox, v1000, fine, tFinger;
            SyncMoveOn sync1,taskslist1;
            MoveL Offs(pPickBox,0,0,-
73.5)\ID:=15,v50,fine,tFinger\WObj:=wobj0;
            WaitTime 1;
            Hand_GripInward;
            GripLoad Maxload;
            MoveL
Offs(pPickBox,0,0,0)\ID:=20,v50,fine,tFinger\WObj:=wobj0;
            MotionSup \Off;
            MoveL pApproachPlace\ID:=25, v200, fine, tFinger;
            MoveL pPlaceBox\ID:=30, v200, fine, tFinger;
            WaitTime 1;
            Hand_MoveTo 8;
            MotionSup \On;
            MoveL pApproachPlace\ID:=35, v50, fine,
tFinger\WObj:=wobj0;
            SyncMoveOff sync1;
            GripLoad Minload;

        ENDPROC
      PROC Connector()
            MoveL pHomeConnector, v500, fine,
            tFinger; Hand_GripInward;
            WaitSyncTask sync4,taskslist1;
            MotionSup \Off;
            MoveL pPushConnector, v500, fine,
            tFinger; SyncMoveOn sync4,taskslist1;
            MoveL Offs(pPushConnector,0,0,-
60)\ID:=40,v50,fine,tFinger\WObj:=wobj0;
            MoveL pPushConnector\ID:=45, v500, fine, tFinger;
            SyncMoveOff sync4;
            MotionSup \On;
    ENDPROC

    PROC Move_Relayboxout ()
        MotionSup \Off;
        WaitSyncTask sync2,taskslist1;
        Hand_MoveTo 7;
        MoveL pPickout, v500, fine,
        tFinger; WaitTime 1;
        SyncMoveOn sync3,taskslist1;
        MoveL Offs(pPickout,0,0,-
78)\ID:=50,v50,fine,tFinger\WObj:=wobj0;
        Hand_GripInward;
        GripLoad Maxload;
        MoveL pPickout\ID:=55, v200, fine, tFinger;
```

```
        MoveL
Offs(pPickout,0,255,0)\ID:=60,v50,fine,tFinger\WObj:=wobj0;
        MoveL Offs(pPlaceout,0,0,-
78)\ID:=65,v50,fine,tFinger\WObj:=wobj0;
        Hand_MoveTo 7;
        MoveL pPlaceout\ID:=70,v50,fine,tFinger;

       SyncMoveOff sync2;
       MotionSup \On;
    ENDPROC
ENDMODULE
```

7.1.2 Right Arm

```
MODULE MainModule
    PERS tasks taskslist1{2}:=[["T_ROB_L"],["T_ROB_R"]];
    VAR syncident sync1;
    VAR syncident sync2;
    VAR syncident sync3;
    VAR syncident sync4;
    PERS loaddata Maxload:=[0.20,[0,0,1],[1,0,0,0],0,0,0];
    PERS loaddata Minload:=[0.02,[0,0,1],[1,0,0,0],0,0,0];
    PERS loaddata Connectorload:=[0.3,[0,0,1],[1,0,0,0],0,0,0];
    CONST robtarget pHome:=[[540.00,-140.32,386.19],[0.31139,-
0.58401,-0.455915,-0.595071],[0,-1,0,11],[-
179.546,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickConnector1:=[[309.37,61.64,117.17],[0.0112673,0.99989,0.003076
74,0.00918799],[1,-2,-
1,11],[170.391,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceConnector1:=[[342,93,132.73],[0.0427047,0.998511,0.032159,0.0
108559],[1,-2,-1,11],[175.94,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPickConnector2:=[[386.48,-
416.41,111.42],[0.00143923,0.999458,0.030435,-0.0124962],[0,-1,-
3,11],[168.534,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceConnector2:=[[338.42,95.39,54.49],[0.00849555,0.999948,-
0.00288278,0.00492298],[1,-2,-
1,11],[177.05,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPushConnector:=[[339.81,40.15,124.35],[0.0330323,-0.999234,-
0.0152652,-0.0144258],[1,-2,-
2,11],[177.633,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPickout:=[[285.77,55.28,85],[0.00940332,-
0.999784,-0.0108484,-0.0150318],[1,-2,-
1,11],[171.505,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPlaceout:=[[285.77,310.28,85],[0.00940332,-
0.999784,-0.0108484,-0.0150318],[1,-2,-
1,11],[171.505,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPickBox:=[[301.08,-
195.02,65.61],[0.0219064,-0.999614,-0.00135194,-0.0170226],[0,-2,-
2,11],[167.461,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPlace:=[[280.50,62.34,64.56],[0.030809,0.999482,-
```

```
0.00767279,0.00530611],[1,-2,-
1,11],[166.731,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceBox:=[[283.38,59.18,13.82],[0.0170696,0.999802,0.00623936,-
0.00813713],[1,-2,-1,11],[163.369,9E+09,9E+09,9E+09,9E+09,9E+09]];

    PROC main()
      MoveJ pHome, v1000, fine, GripperR;
      Hand_Initialize\maxSpd:=20\holdForce:=30\Calibrate;
      Hand_GripInward;
      Labelling;
      Move_Relayboxin;
      Connector;
      Move_Relayboxout;
    ENDPROC
    PROC Labelling()
         WaitSyncTask sync3,taskslist1;
    ENDPROC

      PROC Move_Relayboxin ()
         Hand_MoveTo 9;
         MoveJ pPickBox, v1000, fine, tFinger;
         SyncMoveOn sync1,taskslist1;
         MoveL Offs(pPickBox,0,0,-
75.9)\ID:=15,v50,fine,tFinger\WObj:=wobj0;
         WaitTime 1;
         Hand_GripInward;
         GripLoad Maxload;
         MoveL
Offs(pPickBox,0,0,0)\ID:=20,v50,fine,tFinger\WObj:=wobj0;
         MotionSup \Off;
         MoveL pApproachPlace\ID:=25,v200, fine, tFinger;
         MoveL pPlaceBox\ID:=30, v200, fine, tFinger;
         WaitTime 1;
         Hand_MoveTo 8;
         MotionSup \On;
         MoveL pApproachPlace\ID:=35,v50,
fine, tFinger\WObj:=wobj0;
         SyncMoveOff sync1;
         GripLoad Minload;

        ENDPROC
  PROC Connector()
         MoveL pPickConnector1,v1000, fine, tFinger;
         MoveL pPickConnector2,v1000, fine, tFinger;
         Hand_Gripoutward;
         MoveL Offs(pPickConnector2,0,0,-
206),v50,fine,tFinger\WObj:=wobj0;
         Hand_GripInward;
         GripLoad Maxload;
         MoveL  pPickConnector2,v200,  fine,  tFinger;
         MoveL pPlaceConnector1,v200, fine, tFinger;
         MoveL pPlaceConnector2, v50, fine, tFinger;
         MoveL Offs(pPlaceConnector2,0,0,-
10),v50,fine,tFinger\WObj:=wobj0;
```

```
        WaitTime 1;
        Hand_Gripoutward;
        MoveL pPlaceConnector1 ,v50, fine, tFinger;
        Hand_GripInward;
        WaitSyncTask sync4,taskslist1;
        MotionSup \Off;
        MoveL pPushConnector, v500, fine,
        tFinger; SyncMoveOn sync4,taskslist1;
        MoveL Offs(pPushConnector,0,0,-
60)\ID:=40,v50,fine,tFinger\WObj:=wobj0;
        MoveL pPushConnector\ID:=45, v500, fine, tFinger;
        SyncMoveOff sync4;
        MotionSup \On;
    ENDPROC

    PROC Move_Relayboxout ()
        MotionSup \Off;
        WaitSyncTask sync2,taskslist1;
        Hand_MoveTo 7;
        MoveL pPickout, v500, fine, tFinger;
        WaitTime 1;
        SyncMoveOn sync3,taskslist1;
        MoveL Offs(pPickout,0,0,-
76.5)\ID:=50,v50,fine,tFinger\WObj:=wobj0;
        Hand_GripInward;
        GripLoad Maxload;
        MoveL pPickout\ID:=55, v200, fine, tFinger;
        MoveL
Offs(pPickout,0,255,0)\ID:=60,v50,fine,tFinger\WObj:=wobj0
        ; MoveL Offs(pPlaceout,0,0,-
78)\ID:=65,v50,fine,tFinger\WObj:=wobj0;
        Hand_MoveTo 7;
        MoveL pPlaceout\ID:=70,v50,fine,tFinger;
        SyncMoveOff sync2;
        MotionSup \On;
    ENDPROC
ENDMODULE
```

7.2 Small Box

7.2.1 Left Arm

```
MODULE MainModule
    PERS tasks taskslist1{2}:=[["T_ROB_L"],["T_ROB_R"]];
    VAR syncident sync1;
    VAR syncident sync2;
    VAR syncident sync3;
    VAR syncident sync4;
      CONST robtarget
pHome:=[[534.22,122.96,379.18],[0.526311,0.538683,0.564936,-
0.337144],[-1,0,1,11],[-170.038,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlace:=[[466.26,130.06,95.01],[0.028082,0.999604,-
0.0011032,0.0015903],[-
1,1,1,11],[167.01,9E+09,9E+09,9E+09,9E+09,9E+09]];
```

```
    CONST robtarget pHome1:=[[466.26,300,95.00],[0.00155178,-
0.999976,0.00312532,0.00604186],[-
1,1,1,11],[167.132,9E+09,9E+09,9E+09,9E+09,9E+09]];
    PERS loaddata Maxload:=[0.15,[0,0,1],[1,0,0,0],0,0,0]; PERS
    loaddata Minload:=[0.02,[0,0,1],[1,0,0,0],0,0,0]; PERS
    loaddata Connectorload:=[0.3,[0,0,1],[1,0,0,0],0,0,0];
    CONST robtarget
pHomeLabeling:=[[410.01,211.92,379.20],[0.626518,0.28075,0.685098,-
0.243504],[-1,-1,2,1010],[172.821,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPickLabel1:=[[456.40,145.43,168.32],[0.682835,0.069835,0.7
25676,-0.0474824],[-1,-2,2,11],[-
169.715,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPickLabel2:=[[456.40,175.43,168.32],[0.682835,0.069835,0.7
25676,-0.0474824],[-1,-2,2,11],[-
169.715,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel1:=[[575.01,91.75,163.20],[0.714789,0.0294322,0.698525,-
0.0165228],[-2,-2,2,11],[-175.139,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPlaceLabel1:=[[750.89,-
48.32,342.22],[0.69987,0.0183306,0.713018,-0.0380932],[-2,-
2,2,11],[-137.536,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPlaceLabel2:=[[750.89,-
18.32,342.22],[0.69987,0.0183306,0.713018,-0.0380932],[-2,-
2,2,11],[-137.536,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pStickLabel2:=[[767.89,-
18.32,297.22],[0.69987,0.0183306,0.713018,-0.0380932],[-2,-
2,2,11],[-137.536,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pStickLabel1:=[[765.89,-
68.32,297.22],[0.69987,0.0183306,0.713018,-0.0380932],[-2,-
2,2,11],[-137.536,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pApproachPlaceLabel1:=[[577.77,-
60.65,343.77],[0.693912,0.0250903,0.718463,-0.0408335],[-2,-
3,2,11],[-128.555,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pApproachPlaceLabel2:=[[577.77,-
30.65,343.77],[0.693912,0.0250903,0.718463,-0.0408335],[-2,-
3,2,11],[-128.555,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel2a:=[[709.21,120.05,159.45],[0.750701,0.00414234,0.660609
,0.00516011],[-2,-2,2,11],[178.067,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel2b:=[[584.91,118.92,119.52],[0.696426,-
0.0200795,0.717161,0.0163684],[-2,-
2,2,11],[177.79,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPickLabel3:=[[429.27,150.57,140.47],[0.708839,0.0355911,0.
704454,-0.00493695],[-1,-2,2,11],[-
170.886,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel3a:=[[591.60,138.37,139.24],[0.697272,0.00314585,0.713685 ,0.0667421],[-
2,-2,2,11],[-179.907,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickLabel3b:=[[587.17,153.82,117.15],[0.703603,0.0217157,0.710129,
0.0137136],[-2,-2,2,11],[-179.358,9E+09,9E+09,9E+09,9E+09,9E+09]];
```

```
    CONST robtarget
pTurnLabel3:=[[449.47,188.18,163.22],[0.424538,-0.520465,0.489666,-
0.555978],[-2,-2,1,11],[-126.417,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pApproachPlaceLabel3:=[[669.10,263.22,340.00],[0.492811,-
0.515145,0.474018,-0.516789],[-2,-2,0,11],[-
136.817,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceLabel3:=[[670.07,135.27,324.98],[0.438965,-
0.510418,0.509029,-0.536352],[-2,-2,1,11],[-
145.301,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPickout:=[[451.60,130,85],[0.0179469,0.998636,0.0400029,0.0283664]
,[-1,1,1,11],[165.344,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPick:=[[466.27,-
124.92,95],[0.0280404,0.999605,-0.00110664,0.00161972],[-
2,1,1,11],[167.069,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPushConnector:=[[342.70,137.70,126.11],[0.0198139,0.998769,0.04109
41,0.0194694],[-1,1,1,11],[172.972,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlaceout:=[[451.60,384.97,85.00],[0.0180005,0.998635,0.0400051,0.0
283627],[-1,1,1,11],[165.246,9E+09,9E+09,9E+09,9E+09,9E+09]];




    PROC main()
         MoveJ pHome,v1000, fine, GripperL;
         Hand_Initialize\maxSpd:=20\holdForce:=30\Calibrate;
         Hand_GripInward;
         Labeling;
         Move_Relaybox;
         Connector;
         Move_Relayboxout;
        ENDPROC

        PROC labeling ()
             MoveJ pHomeLabeling, v1000, fine, tool0;
             MoveJ pApproachPickLabel1, v500, fine,
             tool0; MoveJ pPickLabel1, v500, fine, tool0;
             Hand_TurnOnVacuum1;
             MoveL Offs(pPickLabel1,0,0,-
             48),v50,fine,tool0; WaitTime 1;
             MoveL pPickLabel1, v500, fine, tool0;
             MoveJ pApproachPickLabel1,v500,fine,tool0;
             MoveJ pApproachPlaceLabel1, v500, fine, GripperL;
             MoveL pPlaceLabel1, v200, fine, GripperL;
             MoveL Offs(pPlaceLabel1,15,-15,-
             45),v50,fine,GripperL; WaitTime 1;
             Hand_TurnOffVacuum1;
             MoveL Offs(pStickLabel1,5,3,0),v50,fine,GripperL;
             MoveL pPlaceLabel1, v200, fine, GripperL;
             MoveL pApproachPlaceLabel1, v500, fine, GripperL;
```

```
            MoveJ pApproachPickLabel2, v500, fine,
            GripperL; MoveJ pPickLabel2a, v500, fine,
            GripperL; MoveL pPickLabel2b, v50, fine, tool0;
            Hand_TurnOnVacuum1;
            WaitTime 1;
            MoveL pPickLabel2a, v500, fine, GripperL;
            MoveJ pApproachPickLabel2, v500, fine, GripperL;
            MoveJ pApproachPlaceLabel2, v500, fine, GripperL;
            MoveL pPlaceLabel2, v200, fine, GripperL;
            MoveL Offs(pPlaceLabel2,17,0,-45),v50,fine,GripperL;
            WaitTime 1;
            Hand_TurnOffVacuum1;
            MoveL Offs(pStickLabel2,-3,-3,0),v50,fine,GripperL;
            MoveL Offs(pStickLabel2,3,3,0),v50,fine,GripperL;
            MoveL pPlaceLabel2, v100, fine, GripperL;
            MoveL pApproachPlaceLabel2, v500, fine, GripperL;
            MoveJ pApproachPickLabel3, v500, fine, tool0;
            MoveJ pPickLabel3a, v500, fine, tool0;
            Hand_TurnOnVacuum1;
            MoveL pPickLabel3b, v500, fine,
            tool0; WaitTime 2;
            MoveL pPickLabel3a, v500, fine, tool0;
            MoveJ pApproachPickLabel3, v500, fine,
            tool0; MoveL pTurnLabel3, v500, fine, tool0;
            MoveL pApproachPlaceLabel3, v500, fine, tool0;
            MoveL pPlaceLabel3, v50, fine, tool0; WaitTime
            1;
            Hand_TurnOffVacuum1;
            MoveL Offs(pPlaceLabel3,5,5,0),v50,fine,tool0;
            !MoveL Offs(pPlaceLabel3,-5,-5,0),v50,fine,tool0;
            MoveJ pApproachPlaceLabel3, v500, fine, tool0;
            WaitDI custom_DI_0, 1;
            MoveJ pHome, v1000, fine, tool0;
            WaitSyncTask sync3,taskslist1;
        ENDPROC

          PROC Move_Relaybox ()
           Hand_MoveTo 9;
           MoveJ pPick, v1000, fine, tFinger;
           SyncMoveOn sync1,taskslist1;
           MoveL Offs(pPick,0,0,-
103)\ID:=15,v50,fine,tFinger\WObj:=wobj0;
           WaitTime 1;
           Hand_GripInward;
           GripLoad Maxload;
           MoveL
Offs(pPick,0,0,0)\ID:=20,v50,fine,tFinger\WObj:=wobj0;
           MotionSup \Off;
           MoveL
Offs(pPick,0,255,0)\ID:=20,v200,fine,tFinger\WObj:=wobj0;
           MoveL Offs(pPlace,-11,0,-
78)\ID:=35,v50,fine,tFinger\WObj:=wobj0
;
           WaitTime 2;
           Hand_MoveTo 8;
           MotionSup \On;
```

```
            MoveL pPlace\ID:=40, v50, fine,
            tFinger\WObj:=wobj0; SyncMoveOff sync1;
            GripLoad Minload;

          ENDPROC
      PROC Connector()
            MoveL pHome1, v500, fine, tFinger;
            Hand_GripInward;
            WaitSyncTask sync4,taskslist1;
            MotionSup \Off;
            MoveL pPushConnector, v500, fine,
            tFinger; SyncMoveOn sync4,taskslist1;
            MoveL Offs(pPushConnector,0,0,-
60)\ID:=70,v50,fine,tFinger\WObj:=wobj0;
            MoveL pPushConnector\ID:=75, v500, fine, tFinger;
            SyncMoveOff sync4;
            MotionSup \On;
    ENDPROC

    PROC Move_Relayboxout ()
        MotionSup \Off;
        WaitSyncTask sync2,taskslist1;
        Hand_MoveTo 7;
        MoveL pPickout, v500, fine, tFinger;
        SyncMoveOn sync3,taskslist1;
        MoveL Offs(pPickout,0,0,-
71.5)\ID:=45,v50,fine,tFinger\WObj:=wobj0;
        WaitTime 2;
        Hand_GripInward;
        GripLoad Maxload;
        MoveL pPickout\ID:=50, v200, fine, tFinger;
        MoveL
Offs(pPickout,0,255,0)\ID:=60,v50,fine,tFinger\WObj:=wobj0;
       MoveL Offs(pPlaceout,0,0,-
74)\ID:=60,v50,fine,tFinger\WObj:=wobj0;
       Hand_MoveTo 7;
       MoveL pPlaceout\ID:=65,v50,fine,tFinger;

       SyncMoveOff sync2;
       MotionSup \On;
    ENDPROC
ENDMODULE
```

7.2.2 Right Arm

```
MODULE MainModule
    PERS tasks taskslist1{2}:=[["T_ROB_L"],["T_ROB_R"]];
    VAR syncident sync1;
    VAR syncident sync2;
    VAR syncident sync3;
    VAR syncident sync4;
      CONST robtarget pHome:=[[540.00,-140.32,386.19],[0.31139,-
0.58401,-0.455915,-0.595071],[0,-1,0,11],[-
179.546,9E+09,9E+09,9E+09,9E+09,9E+09]];
```

```
      CONST robtarget pPick1:=[[306,-180,113.80],[0.0701091,-
0.996613,-0.0112179,-
0.0414917],[0,0,0,1010],[175.614,9E+09,9E+09,9E+09,9E+09,9E+09]];
    PERS loaddata Maxload:=[0.15,[0,0,1],[1,0,0,0],0,0,0];
    PERS loaddata Minload:=[0.02,[0,0,1],[1,0,0,0],0,0,0];
    PERS loaddata Connectorload:=[0.3,[0,0,1],[1,0,0,0],0,0,0];
    CONST robtarget
pPlace1:=[[309.37,61.64,117.17],[0.0112673,0.99989,0.00307674,0.009
18799],[1,-2,-1,11],[170.391,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlace3:=[[342,93,132.73],[0.0427047,0.998511,0.032159,0.0108559],[
1,-2,-1,11],[175.94,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPick:=[[300.78,-
178.05,94.98],[0.0120484,0.998944,0.0396648,0.0198355],[0,-2,-
2,11],[161.933,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlace:=[[300.77,76.94,94.98],[0.012056,0.998943,0.0396765,0.019828
3],[1,-2,-1,11],[161.967,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPlace2:=[[386.48,-
416.41,111.42],[0.00143923,0.999458,0.030435,-0.0124962],[0,-1,-
3,11],[168.534,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPlace4:=[[335.42,93.38,73],[0.0149008,0.999843,-
0.00916815,0.00262671],[1,-2,-
1,11],[177.062,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget
pPushConnector:=[[339.81,40.15,124.35],[0.0330323,-0.999234,-
0.0152652,-0.0144258],[1,-2,-
2,11],[177.633,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPickout:=[[285.77,55.28,85],[0.00940332,-
0.999784,-0.0108484,-0.0150318],[1,-2,-
1,11],[171.505,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget pPlaceout:=[[285.73,310.27,84.97],[0.0093794,-
0.999785,-0.0107642,-0.0150441],[1,-2,-
1,11],[171.448,9E+09,9E+09,9E+09,9E+09,9E+09]];

    PROC main()
      MoveJ pHome, v1000, fine, GripperR;
      Hand_Initialize\maxSpd:=20\holdForce:=30\Calibrate;
      Hand_GripInward;
      Labeling;
      Move_Relayboxin;
      Connector;
      Move_Relayboxout;
    ENDPROC
    PROC Labeling()
         WaitSyncTask sync3,taskslist1;
    ENDPROC

      PROC Move_Relayboxin ()
         Hand_MoveTo 9;
         MoveJ pPick, v1000, fine, tFinger;
         SyncMoveOn sync1,taskslist1;
         MoveL Offs(pPick,0,0,-
105.5)\ID:=15,v50,fine,tFinger\WObj:=wobj0;
```

```
        WaitTime 1;
        Hand_GripInward;
        GripLoad Maxload;
        MoveL
Offs(pPick,0,0,0)\ID:=20,v50,fine,tFinger\WObj:=wobj0;
        MotionSup \Off;
        MoveL
Offs(pPick,0,255,0)\ID:=20,v200,fine,tFinger\WObj:=wobj0;
        MoveL Offs(pPlace,-11,0,-
78)\ID:=35,v50,fine,tFinger\WObj:=wobj0;
        WaitTime 2;
        Hand_MoveTo 8;
        MotionSup \On;
        MoveL pPlace\ID:=40,v50, fine, tFinger\WObj:=wobj0;
        SyncMoveOff sync1;
        GripLoad Minload;

       ENDPROC
  PROC Connector()
        MoveL pPlace1,v1000, fine, tFinger;
        MoveL pPlace2,v1000, fine, tFinger;
        Hand_Gripoutward;
        MoveL Offs(pPlace2,0,0,-
        206),v50,fine,tFinger\WObj:=wobj0; Hand_GripInward;
        GripLoad Maxload;
        MoveL pPlace2,v200, fine, tFinger;
        MoveL pPlace3,v200, fine, tFinger;
        MoveL pPlace4, v50, fine, tFinger;
        MoveL Offs(pPlace4,0,0,-
        30),v50,fine,tFinger\WObj:=wobj0; WaitTime 1;
        Hand_Gripoutward;
        MoveL pPlace3,v50, fine, tFinger;
        Hand_GripInward;
        WaitSyncTask sync4,taskslist1;
        MotionSup \Off;
        MoveL pPushConnector, v500, fine,
        tFinger; SyncMoveOn sync4,taskslist1;
        MoveL Offs(pPushConnector,0,0,-
60)\ID:=70,v50,fine,tFinger\WObj:=wobj0;
        MoveL pPushConnector\ID:=75, v500, fine, tFinger;
        SyncMoveOff sync4;
        MotionSup \On;
   ENDPROC

   PROC Move_Relayboxout ()
        MotionSup \Off;
        WaitSyncTask sync2,taskslist1;
        Hand_MoveTo 7;
        MoveL pPickout, v500, fine, tFinger;
        SyncMoveOn sync3,taskslist1;
        MoveL Offs(pPickout,0,0,-
74.5)\ID:=45,v50,fine,tFinger\WObj:=wobj0;
        WaitTime 2;
        Hand_GripInward;
        GripLoad Maxload;
```

```
        MoveL pPickout\ID:=50, v200, fine, tFinger;
        MoveL
Offs(pPickout,0,255,0)\ID:=60,v50,fine,tFinger\WObj:=wobj0
        ; MoveL Offs(pPlaceout,0,0,-
74)\ID:=60,v50,fine,tFinger\WObj:=wobj0;
        Hand_MoveTo 7;
        MoveL pPlaceout\ID:=65,v50,fine,tFinger;
        SyncMoveOff sync2;
        MotionSup \On;
    ENDPROC
ENDMODULE
```