



# **Commissioning of Omron LD90 AGV and TM Handling robot**

Adrià Espinet Dalmau

BACHELOR'S THESIS  
May 2021

Mechanical Engineering

## **ABSTRACT**

Tampereen ammattikorkeakoulu  
Tampere University of Applied Sciences  
Degree Programme in Mechanical Engineering

ESPINET DALMAU, ADRIÀ:  
Commissioning of Omron LD90 and TM Handling Robot

Bachelor's thesis 54 pages, appendices 7 pages  
May 2021

---

The objective of this thesis was to explain the integration of the MoMa robot to a production line, in order to conclude if the use of the combination of robots is better than using them separately.

In order to obtain an accurate result in the laboratory, a procedure similar to an industry one was designed and tested, thereafter analysing the capabilities of the robots.

In this thesis the basics of the robots and programs used during the thesis are explained, 3D design of workpieces and fixtures have been developed and used for the production line simulation, to follow the programming to obtain the thesis purpose is also explained and applied to the robots.

TMFlow and MobilePlanner were used to program the robots, and the 3D design was done with SolidWorks and Simplify3D.

The empirical part consists of implementing the designed procedure and improving the robot movements in order to make them as efficient as possible.

The main key results obtained from the development of the thesis are that implementing this combination of robots is profitable, and most of the times recommendable, the main benefits of implementing them is the reduction of time and monetary costs.

---

## CONTENTS

1	INTRODUCTION .....	5
2	Background and objectives .....	6
1.1	Background.....	6
1.2	Objectives .....	6
2	Theoretical background.....	7
2.1	MobilePlanner.....	7
2.2	TM Flow .....	9
2.3	Omron LD90 .....	16
2.4	Omron TM5 Handling robot.....	19
3	3D design, printing and CNC machining of the workpieces.....	24
3.1	3D design and printing. ....	24
3.1.0	Parameters when printing .....	27
3.2	3D design and CNC machining.....	27
4	Designing and programming robot actions.....	28
4.1	TM handling robot.....	28
4.2	Omron LD90 .....	35
4.3	Interaction between robots.....	40
5	Findings and proposals .....	43
	REFERENCES .....	44
	APPENDICES.....	47
	Appendix 1. Handle drawing .....	47
	Appendix 2. Placa Base drawing .....	48
	Appendix 3. TM Flow Program (Main Program).....	49
	Appendix 4. TM Flow Program (JOB 1).....	50
	Appendix 5. TM Flow Program (Job 2).....	51
	Appendix 6. TM Flow Program (Initialize).....	52
	Appendix 7. TM Flow Program (Move to Safe Position).....	53

**ABBREVIATIONS**

TAMK	Tampere University of Applied Sciences
cr	credit
AGV	Automated Guided Vehicle
ABS	Absolute
PTP	Point To Point
MoMa	Mobile Manipulator
ARAM	Advanced Robotics Automation Management
MARC	Mobile Adept Robot Controller
AIV	Autonomous Intelligent Vehicle
TM	TechMan

## 1 INTRODUCTION

For the realization of this thesis, we are using the combination of Automated Guided Vehicle (AGV), cobot (Robotiq, 2015) and machine vision systems.

The main purpose is the integration of both systems into one, finally obtaining a robot (Brittanica, 2020) more capable than both robots separately.

To do so we need to program both robots and then make them work together as one.

For programming the Omron LD90 AGV we will use MobilePlanner, and TMflow for the TM5 Handling robot, both are Omron products.

Omron is Japanese company dedicated to the sale and manufacturing of components, equipment and automatization systems. Also known for producing health care equipment and being the first one to develop automatic cashiers.

Omron is the brand of both robots, taking into account that Omron bought Tech Man (TM) and Adept Technologies, TM is the producer of TM5 handling robot and Adept Technologies producer of LD90 AGV.

(Wikipedia, 2021)



**PICTURE 1. Omron Logo (<https://www.omron.com/global/en/>).**

## **2 Background and objectives**

### **1.1 Background**

We can use AGV to transport heavy objects inside all kinds of industries, to carry dangerous material or to smartly organize material.

The main problem is that the vehicle by itself can only transport objects, not load and unload them, this can be solved by adding a conveyor belt on top of the vehicle, this solves the loading and unloading problem.

If the desired action is to pick and place relatively small objects carefully and precisely, the best option is to add a robotic arm on top of the AGV.

(6 River Systems, 2021)

Regarding that solution we can find the perfect combination on the Omron collaborative systems, were robotic arm (TM5 handling robot) and AGV ( LD90) can be found together as if it was one only robot, this contraption is called Mobile Manipulator (MoMa) in Omron terms.

(Omron, 2013, pp. 7)

The uses of this kind of robot are limitless, from automated processes where it can repeat the same procedure, over and over again, without losing precision or speed, on the other hand It can also accomplish the transportation of dangerous materials or objects that humans cannot do, this versatility is the main characteristic of this combination.

### **1.2 Objectives**

The main purpose of this thesis is to make an introduction to the capabilities of this robots, letting to know the potential of the AGV+Handling robot combination. To make so an introduction to the hardware and software of both robots is developed in this thesis, as well as, the 3D design and printing of the workpieces and fixtures, finally an explanation about the communication between both robots using the built in PLC is made.

## 2 Theoretical background

### 2.1 MobilePlanner

This program is designated to be used with the Omron LD90 AGV.

Floor planner is an interface designed with the aim to view and control individual robots or a fleet, control can be done from a tablet, computer or any mobile device.

Through the mobile device you can check the status of the robot (battery, position, job queue, alerts and travel speed), there are Android and IOS versions available

The laser guided robot can scan up to 1000 sq. ft. per min to create accurate maps of the location they are in, these maps can be customized as wanted, adding virtual walls, slow speed zones, etc.

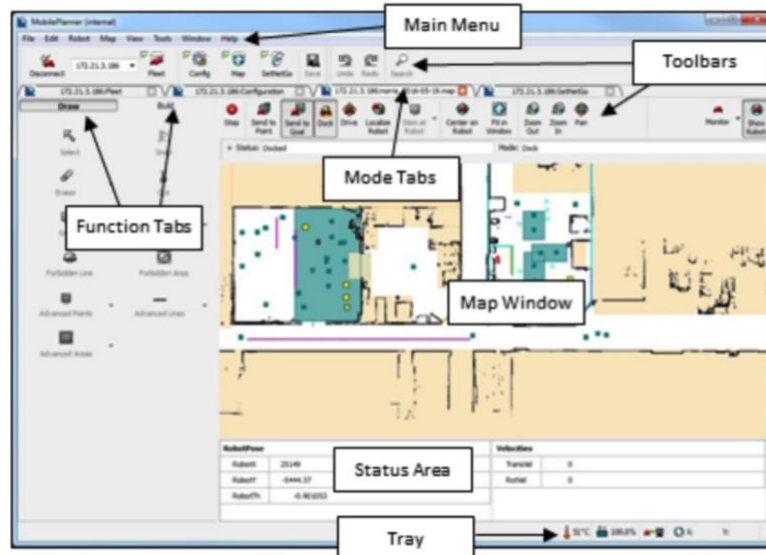
Communication with the robot can be done wirelessly or directly with an ethernet cable (RJ-45).

Using Mobile Planner, we can program from simple movements to making the robot say some words, in between we can also make virtual zones on the map, where the robot can only go slow or fast or go on a determined direction.

MobilePlanner can be used to do a wide variety of things including scanning robot environment, configuring the robot, creating and editing maps, etc.

(Omron Corporation, 2018)

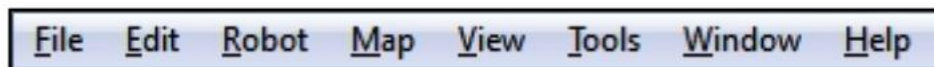
MobilePlanner interface is designed as it can be seen in PICTURE.2 the interface is designed to reduce the learning curve and the time used for deployment.



**PICTURE 2. MobilePlanner interface**

(Omron Corporation, 2018)

In order to get familiar with the software we need to understand everything that can be found on each window on the menu, PICTURE.3



**PICTURE 3. Menu bar**

The table found on PICTURE.4 Explains everything that can be found on all of the different windows in the tool bar.

Menu Option	Description
File	Allows you to open a local scan or map file, a map, scan, or configuration file on a mobile robot, save the map file, insert another map into the existing map file, and download or upload a files to/from a robot. For more details, see File Menu on page 61.
Edit	Allows you to undo or redo your last command. For more details, see Edit Menu on page 62.
Robot	Gives you control over various robot activities (stop, drive, dock/undock), allows you to monitor the robot dashboard and/or enterprise details; create custom commands; create an enterprise message of the day; and reload a configuration or stop the robot. For more details, see Robot Menu on page 63.
Map	Allows you work with and edit the displayed map file. For more details, see Map Menu on page 64.
View	Allows you to change the units used in the map, and turn the toolbar labels on and off. For more details, see View Menu on page 65.
Tools	Allows you to update map features and preferences. For more details, see Tools Menu on page 65
Window	Allows you to access the different files open in the workspace and change the way the windows in MobilePlanner are displayed. You can choose to tile them in the workspace or have them cascade. For more details, see Window Menu on page 66.
Help	Provides help on the MobilePlanner interface. For more details, see Help Menu on page 66.

**PICTURE 4. Menu bars selection description (Omron asset p.60)**



## 2.2 TM Flow

TM Flow main purpose is controlling the handling robots of Omron, programming the robot to make simple movements. Programming is very simple, the interface works using blocks, we can set points manually or through the program.

Manually you can move the robot to the desired position and then set a point, afterwards you can choose between different types of movements.

Straight line, 2 movements (if there's any obstacle in-between), or a combination of both, where the robot makes a fluid curves motion to go to the desired point.

In order to set those points, the first step is to put the robot into manual mode, after we have to use the buttons at the end of the robot head, mainly the Free button to move the robot freely, and to set the different points we will use the point button.

Then we will see the different boxes on the TM Flow program, we can change different options, the speed of the movement, the trajectory that it will follow, etc.

Before programming the procedure that the robot will follow, we need to think about all the specific steps that it will need to go through.

Firstly it's mandatory to set a starting point where it's safe to start the action, just to be sure that it doesn't collide with anything unexpected for being to the wrong starting position, later we need to keep in mind the desired position where the robot must pick up the piece.

With this kind of robot that is equipped with a high-tech vision system, we can set up a visual sign that is always at the same relative position to the pieces, therefore the vision system will know where all the pieces are.

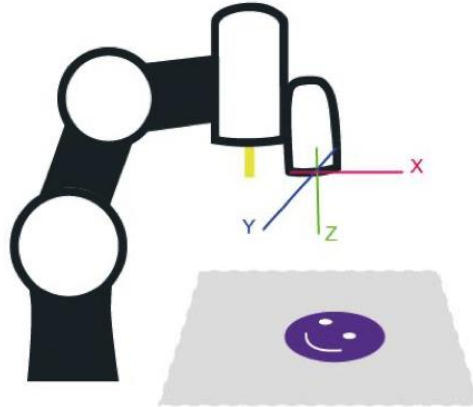
Another methodology to do the previously mentioned localization, is to make the robot learn the shape of the pieces or otherwise the color and shape.

Finally being more precise and effective when using the visual sign we don't ensure that the pieces are on place and all of them are available, avoiding extra movements and optimizing the movement.

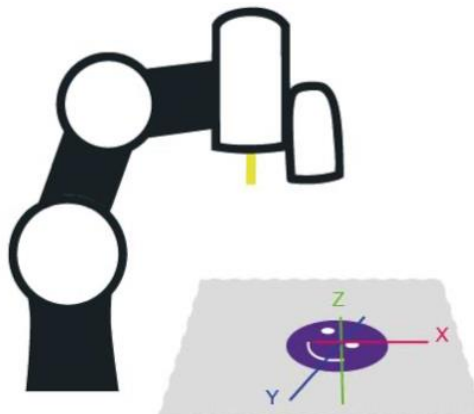
Using the vision positioning we can use 2 kind of bases, the first one being visual servoing positioning, the main idea is to approach the camera to the object, so the base is created on the camera, in the other hand, the second option is to use fixed-point positioning, using this method the coordinates are created on the

object so the relationship between the image coordinates and the robot are calculated using absolute coordinates.

(Omron Corporation & Techman Robot, pp.117 )



**PICTURE 5. Servoing base**



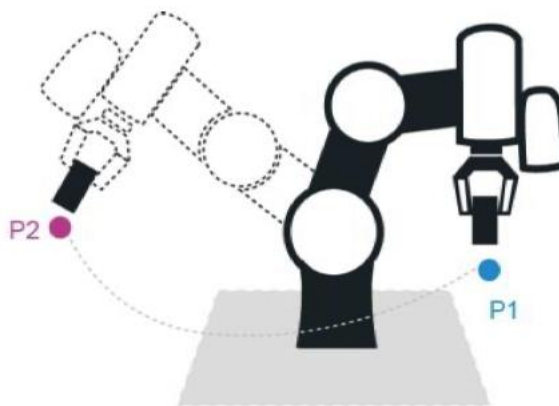
**PICTURE 6. Fix-point Base**

Once the positioning base is created and the starting point we need to design the movements that will make the robot reposition the workpieces, to do so we can set the different points where the robot has to go to via setting the robot to manual mode and then moving the hand freely using the FREE button located on the robot head.

(Omron Corporation & Techman Robot, pp.69 )

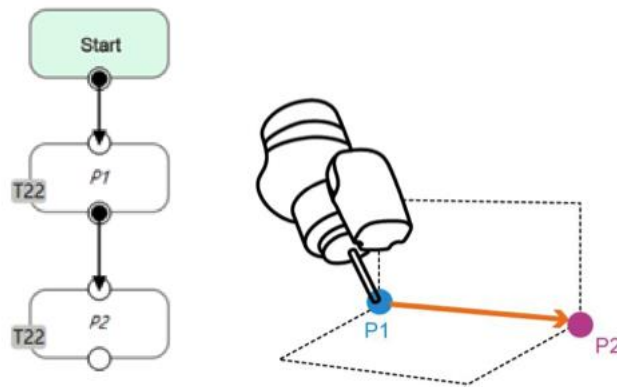
To proceed we need to place the robot to the next position, in this case is going to be in the location of the first workpiece, as soon as the robot is located on the desired position we need to press the POINT button located next to the free button, then to check if the point has been created we can take a look into the TM flow pc program and there we can observe that a new point has been created. On some of this points we will need to change the kind of movement, because if the robot moves making the shortest movement to get to the final position maybe there's an object midway and it runs over it, therefore we can chose from a set of different approaches that the robot can make to avoid the supposed collision:

- **Point-to-Point (PTP):** The robot will go from Point A to Point B calculating the fastest way to get there, that doesn't mind that is a straight line, the robot will calculate the easiest way regarding axis angular movement.  
(Omron Corporation & Techman Robot, pp.86 )



**PICTURE 7. PTP graphic representation**

- **Line:** As the name says the robot is going to move in a straight line to the desired final position.  
(Omron Corporation & Techman Robot, pp.87 )



**PICTURE 8. Line graphic representation**

- **Two steps motion (Way Point):** The movement that the robot will develop is going to be in 2 directions, the first part is going to be in the X and Y axis, then it's going to be located on the Way Point, after the only movement is going to be on the Z axis.

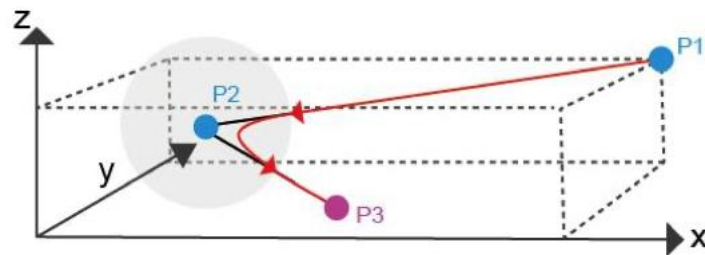
The first part of the movement is made using PTP and the second one using Line.

(Omron Corporation & Techman Robot, pp.88 )

After deciding which kind of movement, we need to apply we can change different settings:

- **Point-to-Point (PTP):** In the case of the PTP motion we can only change the speed of the motion, inside that settings we can modify the speed percentage and the time to top speed.
- **Line:** Line mode can cause in some cases joint speeding, that means that when you program 2-line movements in a row, the moment that the robot passes from the first line to the second one it may cause speeding in that particular zone, in this case we can modify speed percentage and absolute (ABS) speed in the Line speed settings.
- **Two steps motion (Way Point):** In this case we can modify different setting on each part of the movement, firstly we can change the speed percentage and time to top speed because is PTP movement, secondly, we can modify the speed percentage and the ABS as in Line movement.

Another important parameter during the programming of the movement is Blending, when you plan a track, the robot will try to follow it perfectly but the robot cannot, therefore we can use blending in order to make the movement more fluid and reduce the wear and robotic breaks, finally obtaining a shorter cycle time and longer useful lifetime, the 3D visualization of the blending can be like the one following:

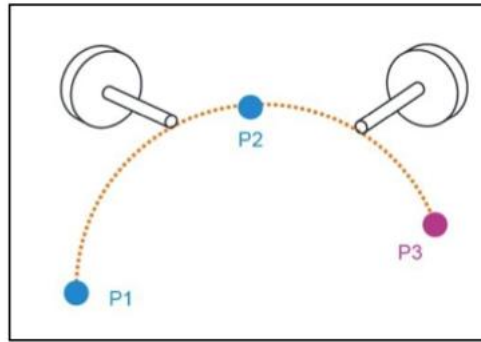


**PICTURE 9. Blending 3D representation**

(Omron Corporation & Techman Robot, pp.88 )

Another step to take account of is checking motion nodes, these are useful to do different things, we can create different kind of motion nodes:

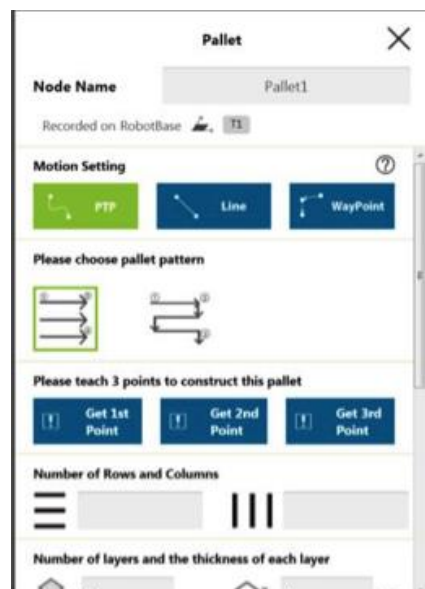
- **Point Node:** This kind of motion node can be set to motion node, blending, base shift or tool shift.
- **F-Point Node:** This node can do precise modifications to the X, Y and Z axis, up to a precision of  $\pm 10\text{mm}$  and Rx, Ry and Rz up to a precision of  $\pm 5^\circ$ .
- **Move Node:** This node can change the values from the Bases X, Y, Z, Rx, Ry, Rz or in the other hand six-axis angles J1-J6, in order to determine the movement that the robot has to perform (distance/angle), and then move to the relative position assigned from the current position.
- **Circle Node:** It's used to create a circular motion between 3 different points



**PICTURE 10. Circle Node graphic representation**

- **Path Node:** Used to read .path files, then the robot moves according that path.
- **Pallet Node:** The main purpose of this node is to be used in pallet placement applications, so this node is able to set three-point coordinates and row and column values in order to move between them.

As we can see on the image below, we can make the robot follow 2 kind of movements regarding pallets, the first one being parallel and the second one zig-zag, on both of them we have to set 3 points, 1<sup>st</sup> point being the starting point of the first row and the posture of the robot, 2<sup>nd</sup> being the end point of the first row and finally the 3<sup>rd</sup> being the end point of the last row.



**PICTURE 11. Pallet mode settings**

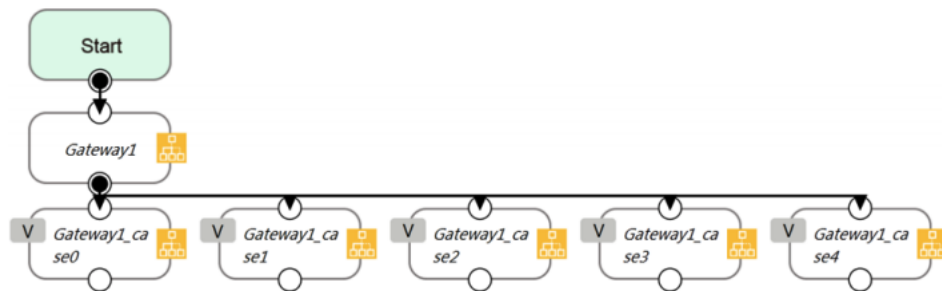
- **Listen Node:** We can execute expressions located on "Expression Editor".

**PICTURE 12. Listen node settings**

(Omron Corporation & Techman Robot, pp.92-102 )

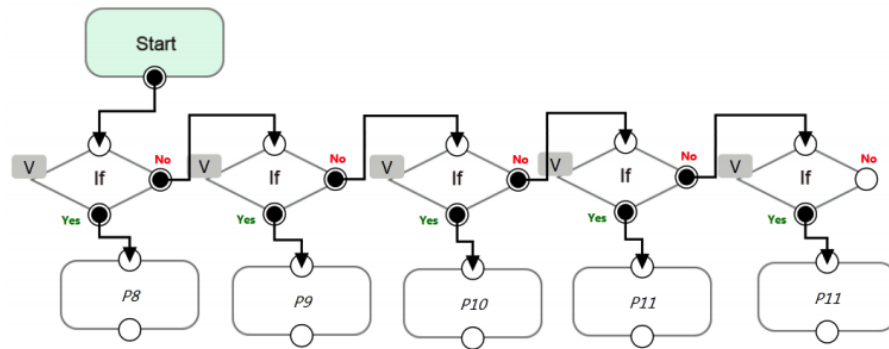
Another node that is going to be used on the project is the gateway node which is similar to using multiple IF nodes, this is going to be used to make the decision on what the robot has to do after what, in this gateway using the settings we can modify the numbers of sub-nodes that are going to be generated.

The program reads the conditions and when one condition is met the lecture of the following nodes is not going to be performed, in PICTURE.13's case we can see a gateway of 5 different conditions.



**PICTURE 13. Gateway Node with 5 different conditions**

Another way of visualizing the previously mentioned comparison, with the IF node is as shown in PICTURE.14



**PICTURE 14. IF nodes with five different conditions, comparison to Gateway**

(Omron Corporation & Techman Robot, pp. 111)

### 2.3 Omron LD90

Omron LD90 is a relatively small, lightweight and with high maneuverability robot, capable of multipurpose tasks, his chassis is made of aluminum and is IP40 certified, thing that makes it very durable.

The driving train is composed of 2 wheels capable of transmitting power and 2 other spring-loaded casters, in the front and in the rear, the driving wheels are equipped with a spring-suspension aiming for balance, the wheels are solid, full of foam, so they are puncture resistant.

LD90 as his name indicates is capable of moving payloads up to 90kg, it weights 60kg, his dimensions are 686x483x371mm (LenghtxWidthxHeight), the floor height clearance is 38mm, one interesting characteristic is that his turn radius is 0mm because both of the driving wheels are located in the center of the body, therefor it's easy for it to move in tight spaces.

Most of his capabilities are due the broad range of sensors that is equipped with:

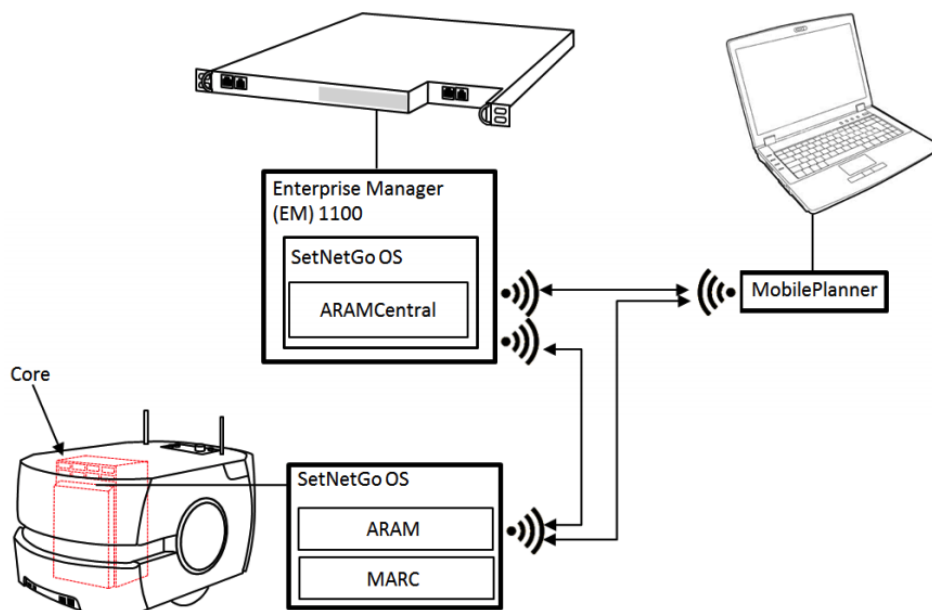


Description	Specification
<b>Sensors</b>	
Safety Scanning Laser	1 at front of platform 203 mm (8 in.) above floor 250°, 15 M range, Class 1, eye-safe PLd Safety per ISO-13849
Sonar Pairs (Each pair is one emitter and one receiver, working together)	2 at rear of platform, 2 M range
Position encoders	2 encoders (one each wheel) 2 Hall sensors (one each wheel)
Analog gyroscope (Core)	320 degree/sec max rotation
Bumper	1 at front of platform
Side Lasers (option)	2 on sides of payload structure, user-mounted
Low Front Laser	1 in bumper
Upward-facing camera (Acuity option)	1 on payload structure, user-mounted
Payload Structure bumpers (option)	6 inputs, user-designed and mounted sensors (3 front, 3 rear)

**PICTURE 14. Omron LD90 Sensor set**

(Omron Corporation, 2018)

The mobile robot is connected via Wifi to a computer, equipped with the MobilePlanner software and to a network appliance (Enterprise Manager), that is capable of handling and organizing fleets.



**PICTURE 15. All components working together**

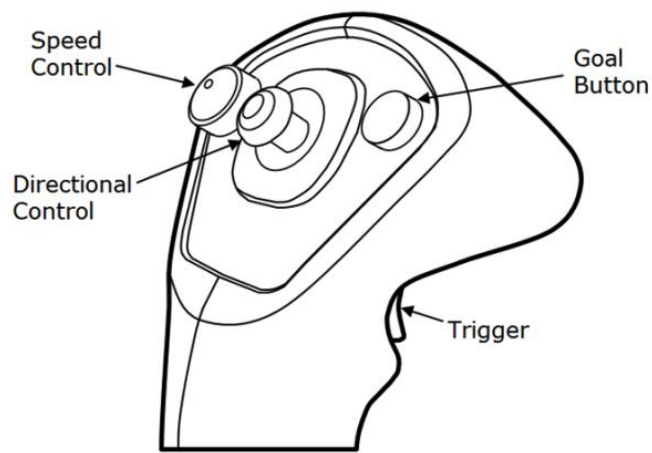
(Omron Corporation, 2018)

- **Enterprise Manager:** Enterprise Manager is a network appliance that is capable of coordinate up to 100 autonomous intelligent vehicles (Omron p.15)
- **ARAMCentral/ARAM:** This software runs in the Enterprise Manager, runs the Autonomous Intelligent Vehicle (AIV) core, operating the range of sensor, autonomous functions and sends commands to Mobile Adept Robot Controller (MARC).
- **MARC:** Resides on the AIV's core, it is the one that manages the speed and head sensor, sensor readings, emergency stop systems, bumpers, and joystick, also is the one that makes the odometry readings (X, Y and heading) and finally other low-level operations that it reports to the Advanced Robotics Automation Management (ARAM).
- **SetAndGo:** SetAndGo is the operating system that can be found on the robot and on the Enterprise Manager, in charge of establishing communication with the robot and, downloading debug information, and performing software upgrades, this software can be accessed via MobilePlanner interface.

The LD90 is capable of scanning rooms in order to generate maps, to do so the AIV uses the laser that is located on the front, capable of scanning a 250-degree arc, the scans are taken about 203 mm above floor level, so the table and chairs appear as 4 legs, a pedestal like a single pole, etc.

During the scan an important part is to locate the docks so the robot will be able to charge itself when necessary, we need to press the goal button on the robot's dock to locate them.

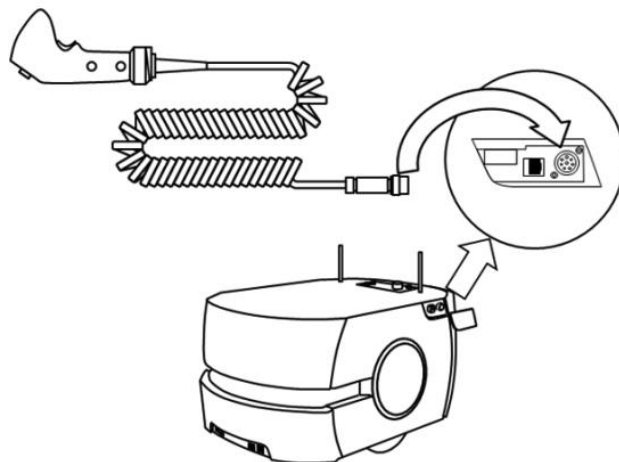
To move the robot carefully we can do it using the manual mode in MobilePlanner's interface or for a better use, the use of the joystick is the most recommended one, the joystick consists of what can we see on PICTURE.16



**PICTURE 16. LD90 control joystick**

(Omron Corporation, 2018)

In order to use it we need to connect it to the LD90 like it's shown on the PICTURE.17 And after waiting until MobilePlanner detects that is ready and it indicates so using the HMI, robot lights or interface.

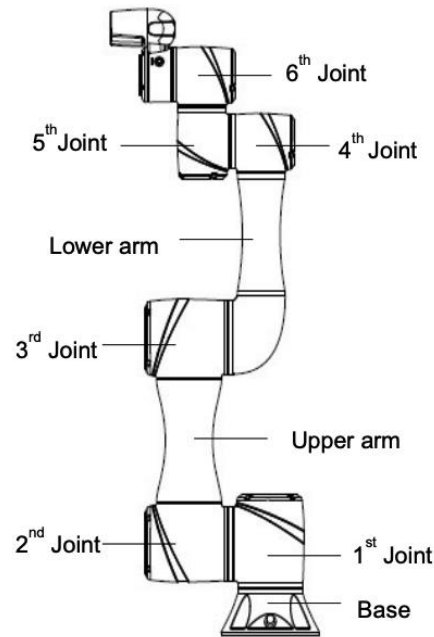


**PICTURE 17. Joystick connection to LD90**

(Omron Corporation, 2018)

## 2.4 Omron TM5 Handling robot

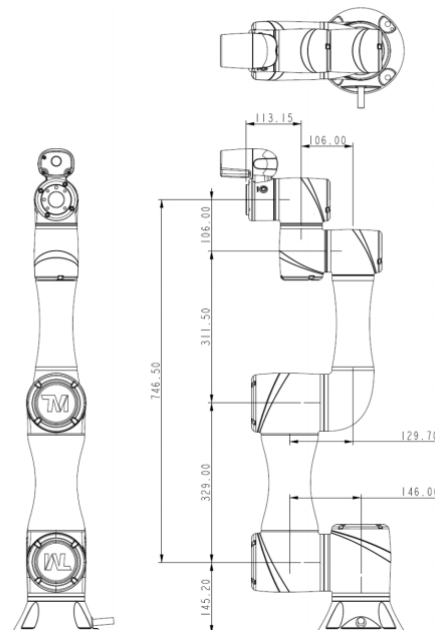
Omron TM5 Handling robot is basically a robotic arm with a gripper attached to the end, this particular model has 6 joints that can move freely.



**PICTURE 18. Omron TM5-700 Joints scheme**

(Omron Corporation, 2018)

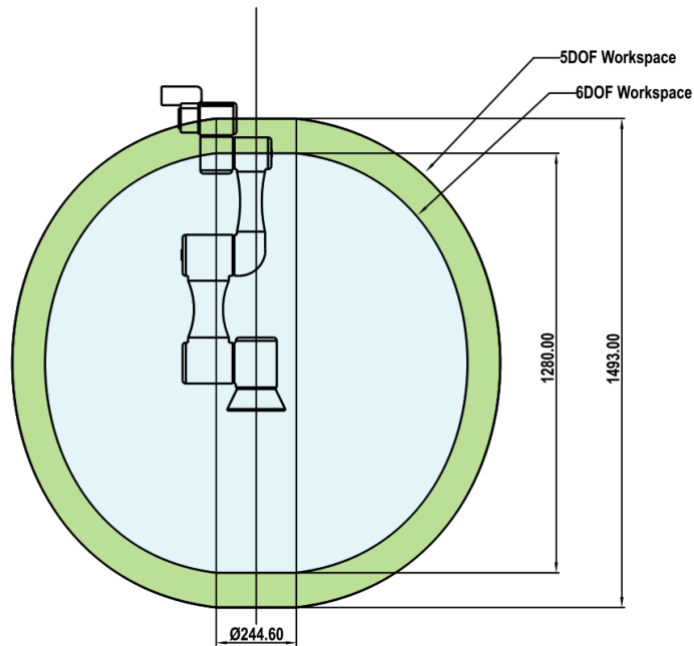
The complete robot consists of the arm itself, a control box and finally a robot stick, the dimensions of the arm are the ones showed on PICTURE.19, all the dimensions are indicated in millimeters.



**PICTURE 19. Dimension drawings of the robot**

(Omron Corporation, 2018)

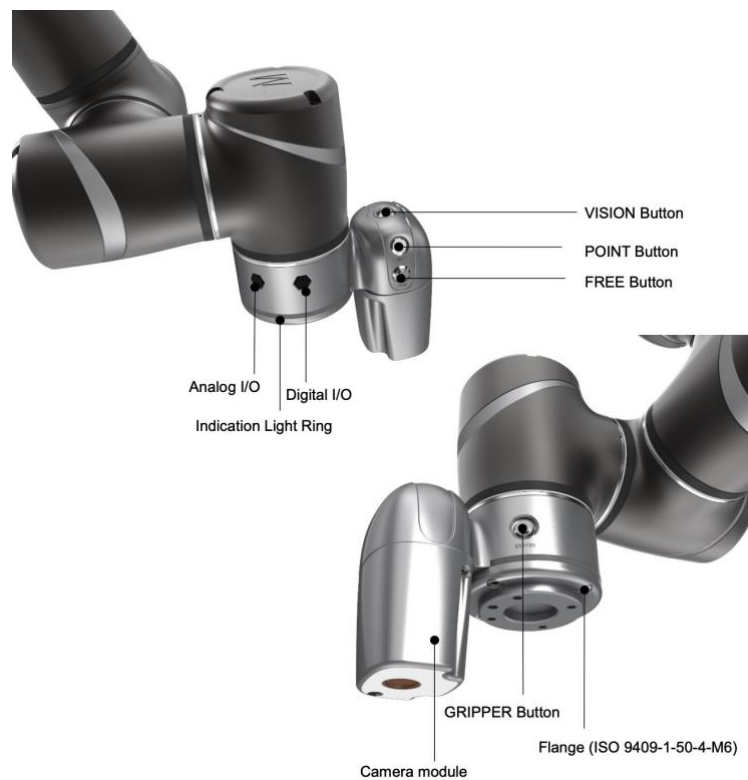
According to the previous dimensions and the hardware that is equipped with the movement range of the robot is the one shown in PICTURE.20



**PICTURE 20. Movement range diagram**

(Omron Corporation, 2018)

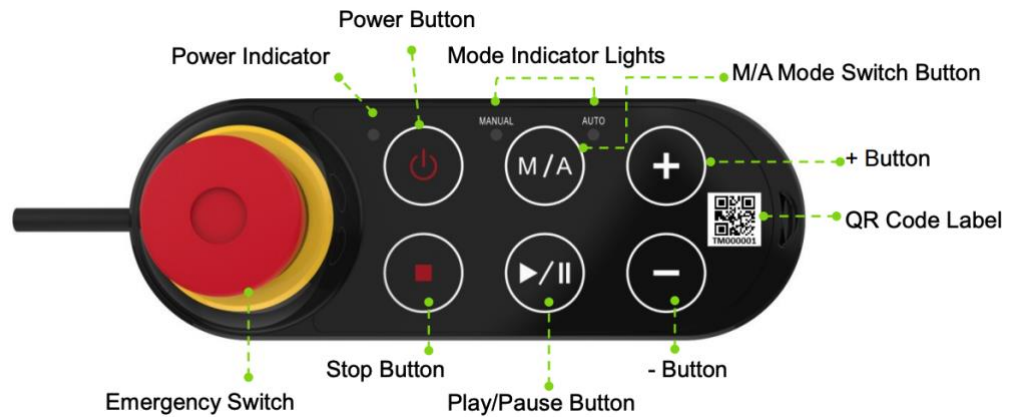
At the final point of the arm, we can find the gripper and a camera that is the most important part of the vision system that the robot is equipped with, also we can find different buttons at that zone, the ones that we will be using are the FREE and POINT, therefore we can find the buttons and switches shown in PICTURE.21



**PICTURE 21. Buttons and switches found in the final part of the arm**  
(Omron Corporation, 2018)

Another hardware part that is going to be used to control the robot is what Omron's call Robot Stick, a wired control stick with the buttons showed in PICTURE. We are going to use basically the play and pause, M/A, Stop and power buttons, during the normal use, if we need to stop the robot in case of emergency the Emergency Switch has to be pressed (E-Stop).

One thing to take into account to put the robot in automatic, we need to use the following sequence “ + - + + - “, in the other hand the play button is going to be used to start the project that is opened in TMFlow and has previously passed the step run.



**PICTURE 22. Robot stick scheme.**

(Omron Corporation, 2018)

Another part of the robot that is in our interest is the camera and its visual range, this parameter will help us understand the maximum range that the camera is able to see and so the limits of using vision nodes and landmark detection, as we can see (PICTURE.23) the field of view at a reasonable distance, like 300mm, is enough for our purpose.

Working distance (mm) \ Field of view (mm)	300	100
Width	281.6	96.9
Height	211.2	72.7

**PICTURE 23. Relation between working distance and field of view**

(Omron Corporation, 2018)

### 3 3D design, printing and CNC machining of the workpieces

#### 3.1 3D design and printing.

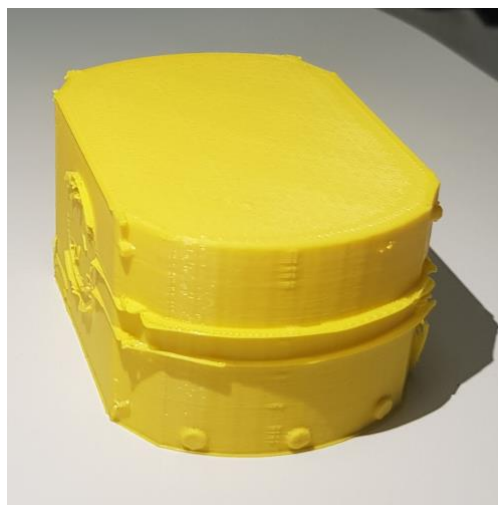
To start designing the pieces and fixtures, first of all, we need to know the limits of the 3D printer we will work with, to do so we can search up the specifications of the printer, in this case the printable area is a total of 400x200x200mm, more than enough for our main purpose, therefore we want to print 3 work pieces and 2 bases where to locate them.

Firstly, I designed the base taking into account the shape of the workpiece so it will fit perfectly once created, my main idea was to create the workpiece shaped similar as the Omron LD90 AGV robot.

The first 2 tries ended like what we can see in PICTURE.24 and PICTURE.25



**PICTURE 24. 1st 3D printed Omron LD90**



**PICTURE 25. 2nd 3D printed Omron LD90**



As we can see the final result is not as expected, to obtain a proper 3D printed object we will make some changes on the 3D design.

To start with the piece we are trying to print is a replica of the LD90, the 3D designed pieces can be downloaded from Omron's website, there we will find the SolidWorks file, this file is adequate for 3D design but not for 3D printing a small replica, because the design has all the details of the real robot, now to be able to print it we need to remove all the internal parts and reduce the complexity of some other parts.

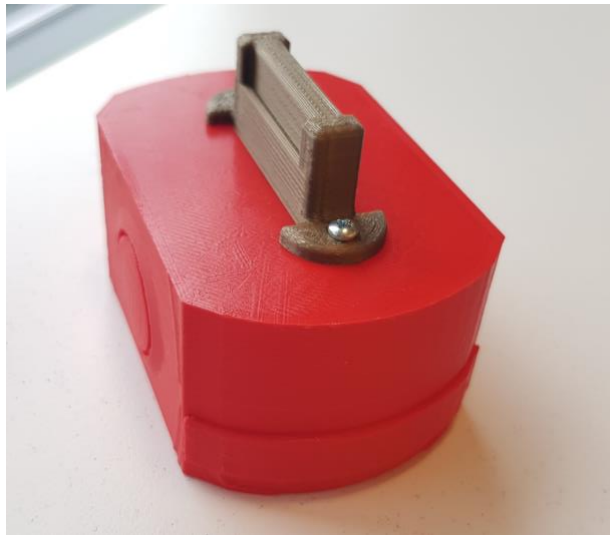
After the reduction and filling of the interior part has been done the final product has less imperfections.



**PICTURE 26. Final 3D printed LD90 model**

Once we obtained the desired final product, we can start printing the number of objects that we require, in this case the number of workpieces is 3, following the workpieces we need to print a handle, so the robot is able to lift up the pieces without problems.

These pieces were designed on SolidWorks and personalized for this particular case, therefore we obtain the final workpiece, LD90 scaled model + handle, both united using screws to obtain optimum solidity:



**PICTURE 27. Final 3D printed workpiece, LD90+Handle**

The other part that also needs to be printed is the fixture where the robot will take the workpieces from, this one has the negative shape of the LD90 scaled model base, so when the robot wants to catch the piece, we perfectly know that is going to be located inside the fixture, this piece has been also designed including the Omron logo, to make reference of the brand of the robots.

In the other hand the fixture has been designed with a tolerance and with chamfers in order to facilitate the robot putting the pieces in the correct place, regarding the fixture the first model was good enough that no changes were needed to do.



**PICTURE 28. 3D printed fixture**

### **3.1.0 Parameters when printing**

3D printing is based on melting different kinds of plastic and then making layers and layers of plastic in order to obtain a 3D printed object made of multiple 2D shapes, with little variations in between them, to do so a lot of 3D designing programs can be used, in this case the program that was used is SolidWorks.

After designing the pieces, we need to save them as .STL file, while are going through the saving screens we can change the quality of the .STL file and a lot of other parameters, but we have to focus on leaving the quality as high as possible in order the desired finished product.

To follow we also need a program that is capable of converting the 3D designed file into a Gcode file, Gcode is the language that 3D printers use, in order to know which path they have to follow to end up with the 3D printed piece, the file that we obtain after passing through the second program takes into account the settings applied as temperature of the bed and of the extruder, speed of the filament, speed of the extrusion, etc.

The previously mentioned settings have as main purpose to obtaining a solid 3D piece where all the layers are fused together and the quality of the print is as it was supposed, we can change the quality of the 3D printed object if we reduce the moving speed, and we reduce the height of the layers.

### **3.2 3D design and CNC machining**

The previously mentioned 3D designed, and printed pieces and fixtures need to be attached to a surface that is not the one direct on the robot, to eradicate any kind of problems regarding warranty and hardware issues, to solve this problem a metal base has been designed and will CNC machined by an external enterprise, this one is going to be used to attach the fixture on the robot's base and both workspaces, we can see a detailed dimensional drawing of the base and the handle in Appendix.1 and 2.

This metal plate has been designed and dimensioned using SolidWorks, the hole pattern was designed using "matrix" tool and "hole wizard", all the holes are M8 and countersunk, as seen on the dimensional drawing.

## **4 Designing and programming robot actions**

The actions that the robot will develop are considering the 3D printed pieces, the main idea is to make the robot move all the pieces from one of the fixtures that act like a base into the other, both of them located on different surfaces.

For accomplishing it we need to teach the Omron LD90 where and when to move, the action developed by it is conditioned by the previous movements the TM Handling robot will do.

So once the TM Handling has ended the picking up action, from the first fixture to the base on the robot then the LD90 has to change location in order to discharge the load, onto the final fixture and surface.

### **4.1 TM handling robot**

Considering all the previous knowledge acquired on the last part we can proceed to start the programming based on the purpose of this thesis, the main idea is to move some pieces from one place onto another using both robots, therefore to start we only need to think about the movement that the robot must do before and after the intervention of the other robot (LD90 AGV), so the procedure that the handling robot has to do as follows, a simplified version of it would be:

1. Move to initial position.
2. Vision node indicating where the pieces are on the first table.
3. Moving to the first piece.
4. Closing gripper.
5. Move piece to fixture on robot's base.
6. Open gripper and place piece.
7. Move to initial position (to ensure not collision to anything).
8. Move to second piece.
9. Close gripper.
10. Move piece to fixture on robot's base.
11. Open gripper and place piece.
12. Move to initial position.
13. Move to third piece.
14. Close gripper.

15. Move piece to fixture on robot's base.
16. Open gripper and place piece.
17. Move to initial position.
18. Move to safe position for moving.

// Intervention of LD90 AGV robot //

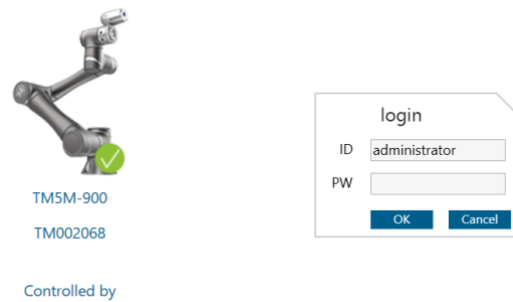
19. Move to initial position.
20. Vision node indicating where the pieces are.
21. Moving to the first piece, located on robot's base.
22. Closing gripper.
23. Move piece to fixture on second table base.
24. Open gripper and place piece.
25. Move to initial position (to ensure not collision to anything).
26. Moving to the second piece, located on robot's base.
27. Closing gripper.
28. Move piece to fixture on second table base.
29. Open gripper and place piece.
30. Move to initial position.
31. Moving to the third piece, located on robot's base.
32. Closing gripper.
33. Move piece to fixture on second table base.
34. Open gripper and place piece.
35. Move to initial position.
36. Move to safe position.

// Second intervention of LF90 AGV robot//

Now taking in consideration the process described previously we can start to transfer this information to the program in order to develop the desired program. To be able to develop the program first of all we need to connect the computer, where we will "write" the program, to do it we have to take the ethernet cable (RJ-45) and connect it to the toggle we will find on the robot, after we need to follow these steps to end up with a successful setup:

1. Turn on the robot

2. Wait until the motors on the joints are active (a clicking noise can be heard)
3. Taking a look in TMFlow we can see the robot that we want to connect to.
4. Double click on the robot.
5. A login screen will appear, leave ID: administrator and PW: "blank", click "OK".



**PICTURE 29. Login screen**

6. Now under the robot's name we can find the box "Get Control", click on it.



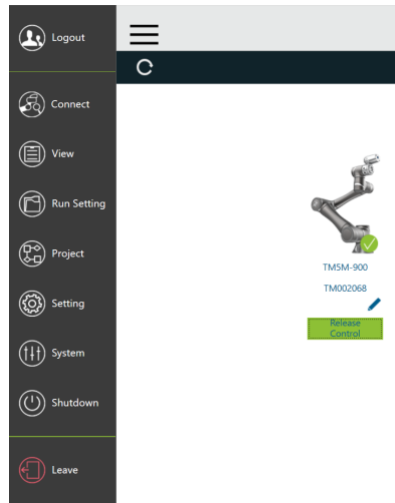
**PICTURE 30. Get control screen**

7. Now the box has changed to "Release Control"



**PICTURE 31. Control acquired screen**

8. At this point we have full control over the robot, we can start developing the program, going to the left side, clicking on the 3 lines and then entering “Project”.

**PICTURE 32. Open a project screen**

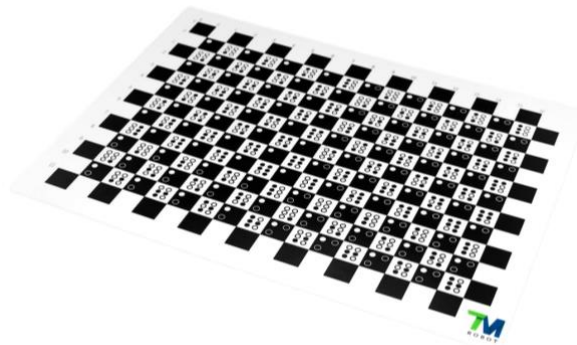
9. Now is time to start creating the procedure the robot will follow, we have to keep in mind that to do so, the robot has to be in “manual mode”, to put the robot in manual mode we have to take the control stick and maintain the M/A button pressed, until a beep is heard.

**PICTURE 33. Control stick**

10. Once inside the project panel we open a blank project, we will see a empty project only with the “Start” box.

Now we are connected to the robot and ready to start creating the procedure.

In order to make the project work we need to proceed like the following, first and most important is to calibrate the vision of the robot in order to be sure that all the movements that the robot makes are precise, to do so we need to find the base dedicated to calibrating.



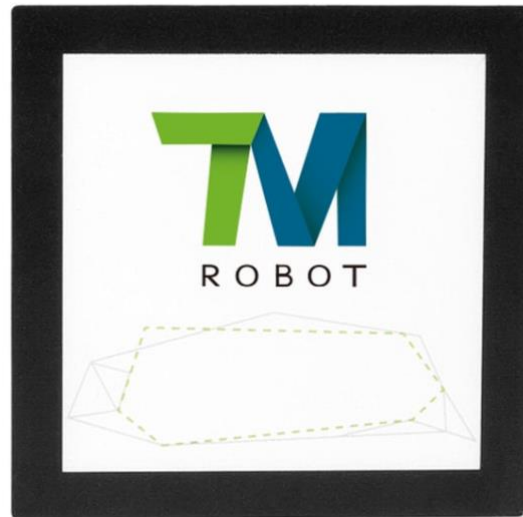
**PICTURE 34. Omron calibration base**

After we need to place it on the working space, the following steps is to proceed inside the vision node to calibrate using the automatic process.

After the vision system is calibrated, we can proceed to design the steps that the robot has to follow.

To start the program, we need to place on the workstation a landmark:





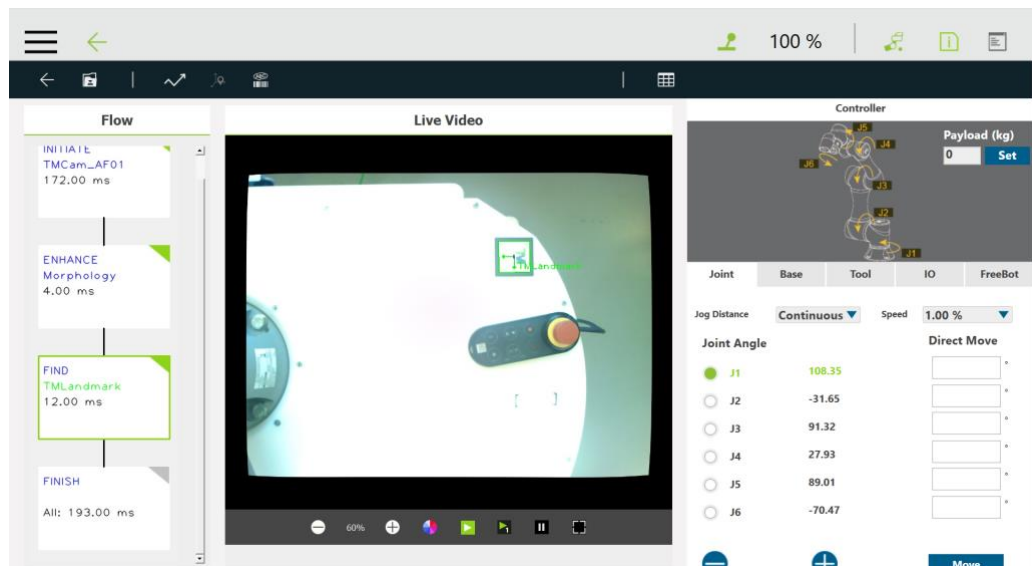
**PICTURE 35. TM Robot Landmark**

This landmark is known by the robot, so it's easy for the robot to recognize it, then when it recognizes the landmark it can start a procedure, to follow we need to create a starting point where is safe for the robot to start.

We can set the robot to manual mode as seen before, (if for some reason is in automatic mode) and then maintaining pressed the FREE button we move the arm to the desired position and click POINT, we can name the point as "startingpoint", going to settings of the point/Point manager/ change the name.

After this step is necessary to create a second point to use the gripper set component, and modify it, we enter: setting of the point>Initialize \_or\_not>Variable>change the text on the box to "true".

Now it's mandatory to create a landmark using a vision node, then edit it and create a vision job and name it "LandMark", click on "task designer", "landmark alignment", enter the fix option, make sure, using the preview screen that is showing, that the vision is capable of detecting it, to know if the detection is successful, we have to see the coordinates in the FIND section of the flow, inside the vision job.



**PICTURE 36. Landmark coordinates**

Next step, move the robot to the desired pick-up position and create a new point like before, click FREE, move the robot and click POINT, a new point should be created in the program, we can name the point as “PICK”, to proceed grab the gripper pick component (the one that has a gear), and create a MOVE node after the “OK” path on the gripper chart, enter the settings of the point and change Z to -200 mm.

Finally we need to move the arm and create a point, the “PlacePoint”, if we need to change the base we can enter the settings>point manager>Re-Record using another base>RobotBase.

Last point to create is a gripper one, this time the open gripper, and after as we saw we need to put a move node and change the Z value to -200 mm.

Using this method, we can change the position of the working pieces on the workspace as we want but not the relative position to the landmark because then robot is not going to be able to find the pieces.

Finally, it's time to test the program that we have been developing, first of all we have to know that the program is going to run under his limits, at 3% speed setting, to make sure you are capable of stopping it before colliding to anything. We will have to confirm the start of each section, this is why is called Step Run, we will have the control stick on our hand, to start each section the play button has to be pressed.

After the first run, using step run, is finished we can finally run the setup at the “normal” speed.

(Omron Industrial Automation EMEA, 2018, Video 1-8)

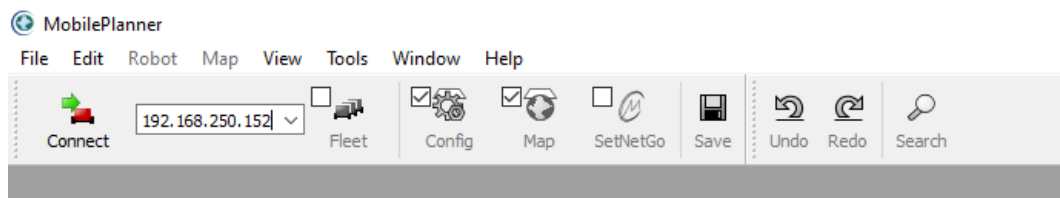
The final main program and subprograms created can be seen in Appendix 3,4,5,6 and 7.

## 4.2 Omron LD90

The Omron as we it’s said, needs MobilePlanner to be controlled and to facilitate the interaction between user and robot, using the interface first of all we need to establish connection between the computer and the robot, this can be done without any physical connection, it can be done using network connection, therefore we will need to know the IP of that concrete robot, in our case we only have two options:

- IP: 192.168.250.151 , this one is the LD90A
- IP: 192.168.250.152 , this one is MoMa’s LD90B

As we can see we need to introduce the desired IP and click connect, PICTURE.37

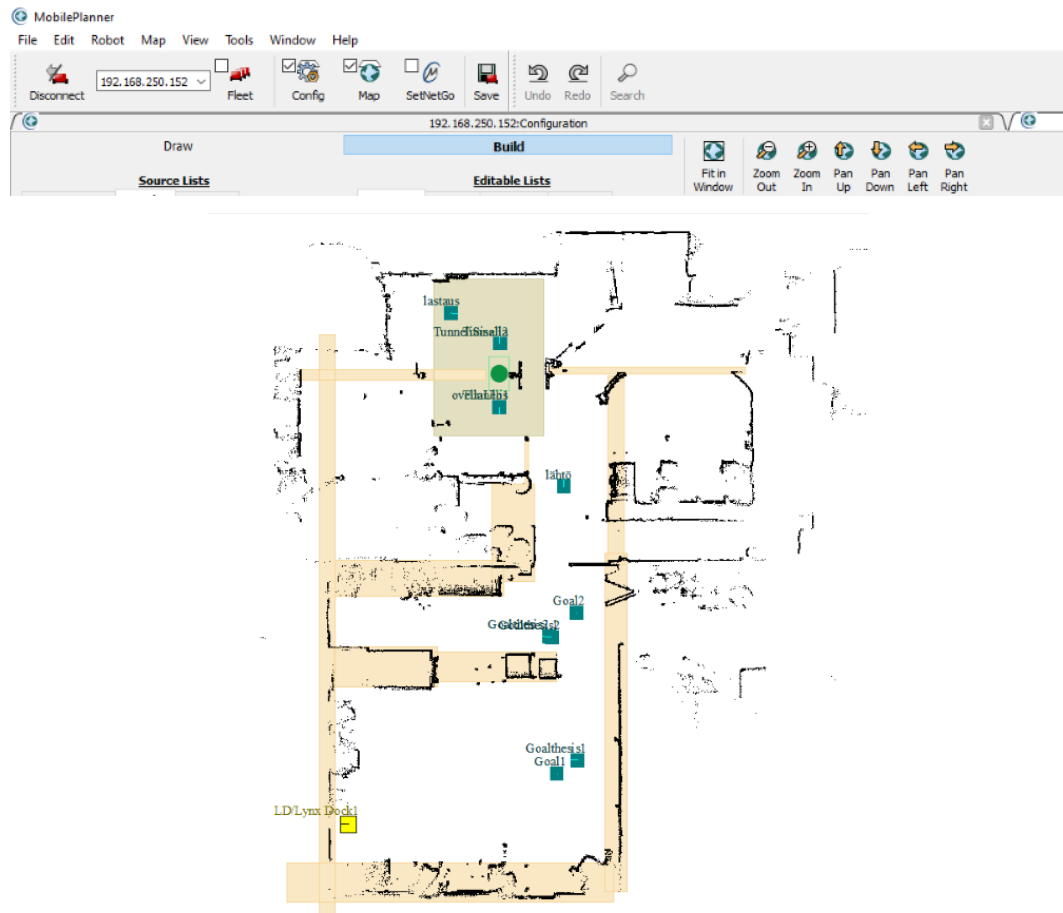


**PICTURE 37. Mobile Planner IP introduction and connection**

Now we have established connection with the robot, now we can take a look in the window that has this name: **192.168.250.152Field\_Lab\_2021\_2.map** (PICTURE.38), this indicates that the robot is connected because the IP is showing on the window name, and the words that follow are indicating the map that the robot has in his memory, in this case is a recent scan made to the FieldLab,

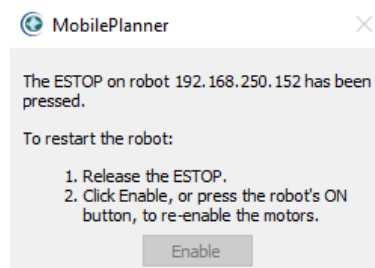
As we can also see there are some yellow zones that indicate restricted zones where the robot is not able to enter, due to software restrictions, this zone can be

deleted and modified, some of them indicate tables and other robots that are located on the FieldLab zone.



**PICTURE 38. Mobile Planner interface and FieldLab map.**

To follow the initialization, we will take a look a typical problem troubleshooting regarding the pop-up message shown on PICTURE.39



**PICTURE 39. Pop-up error message**

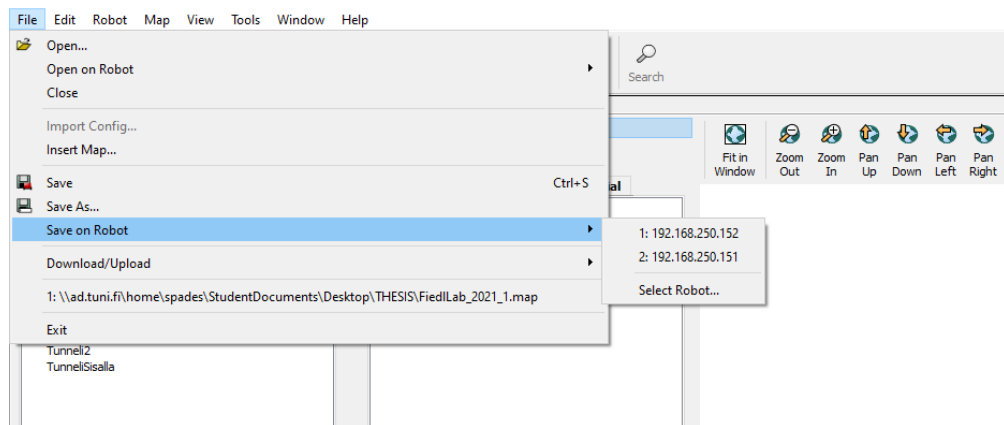
This error typically is due to the statement that is in the message, that the ESTOP has been pressed, or otherwise can be due one of the following things:

- The side box where the control stick is stored is not closed or not closed properly.
- The robot has started a procedure, for example in our case, the robot has received the signal that indicates that the TM5 has finished the task, and for some reason has stopped, the robot has the “ESTOP” pressed due to software, because after starting any task developed by the handling robot, the LD90 has to be in a stationary position.

In order to make sure if there's a problem with the ESTOP we can check it using MobilePlanner's interface, accessing the Robot window>Safety Commissioning> ESTOP, then follow the instructions that will appear on the screen.

After we have established the connection with the robot, and solved the ESTOP issue if applicable, we can continue with loading to the robot the .map file with the most recent version of the map of the zone that we will be using it, in our case the most recent one is Field\_Lab\_2021\_2.map, as we have seen before on the window's name.

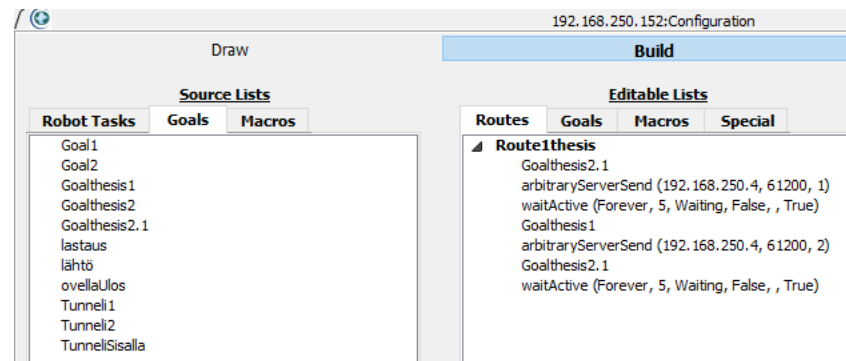
To load it to the robot we need to access the section shown on PICTURE.40



**PICTURE 40. Map load to MoMa**

There we will find two options, each one making reference to one of the robots, as said the 192.168.250.152 is the one that we are using, which is MoMa's IP, now the .map file has been saved to the robot and it's ready to run.

The program used in the thesis is based on setting 2 goals and sending signals to the TM5 once the task has been finished, therefore the program looks like PICTURE.41



**PICTURE 41. MobilePlanner Thesis program**

In order to “create” the program we need to Drag&Drop the different components from the Robot Tasks windows located on the left side.

Now, we will take a deeper look into each one of the steps that the program has, starting with Goalthesis 2.1 and Goalthesis1, this ones are the goals that the robot has to go to in order to load and unload the 3D printed workpieces, Goalthesis2.1 is the first one and Goalthesis1 is the second one.

After we also have the arbitraryServerSend(192.168.250.4, 61200, 1) parameter, this parameter is intended to send a signal to the TM5 once goalthesis2.1 has been reached in order to start the load of the work pieces.

This signal is sent to the **IP: 192.168.250.4, Port: 61200 and the string is 1**, the “1” is assigned to the Job1 in TMFlow, this can be modified or introduced in the window we can see in PICTURE.42

**Task "arbitraryServerSend" Description:**  
Sends the given text to a specified server and port.

**Class:** ExternalConnect

**Parameters:**

Parameter	Value	Min	Max
Server hostname	192.168.250.4		
Server port	61200	0	65536
String to send	1		

**Server hostname**  
Hostname of the server.

OK Cancel

**PICTURE 42. First arbitraryServerSend parameters**

In the other hand the second arbitraryServerSend(192.168.250.4, 61200, 2), makes reference to Job2 located also in TMFlow, therefore the parameters look like PICTURE.43, the only change is the string, in this case is 2.

**Task "arbitraryServerSend" Description:**  
Sends the given text to a specified server and port.

**Class:** ExternalConnect

**Parameters:**

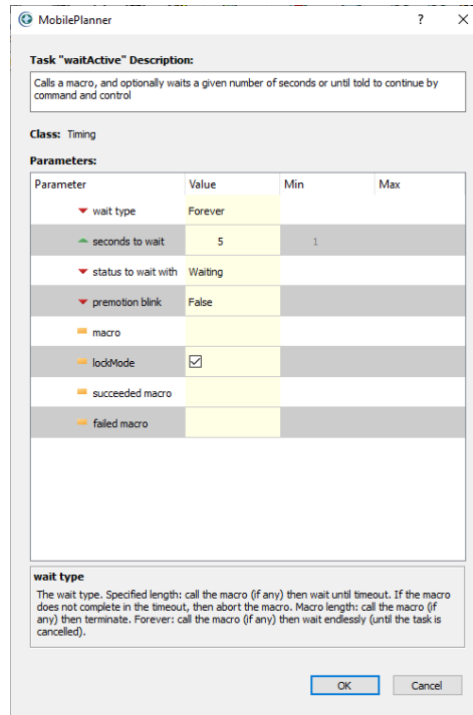
Parameter	Value	Min	Max
Server hostname	192.168.250.4		
Server port	61200	0	65536
String to send	2		

**Server hostname**  
Hostname of the server.

OK Cancel

**PICTURE 43. Second arbitraryServerSend parameters**

Finally we have to tasks that are completely the same, this make reference to the waiting while the TM5 is doing his part, the robot is going to wait “forever”, is going to be locked until the finish signal from TMFlow is sent, and the parameters on both of them are PICTURE.44



**PICTURE 44. waitActive parameters**

These tasks are all we need to make the robot move from Goalthesis2.1 to Goalthesis1 and establish connection between robots in order to load and unload the workpieces.

### 4.3 Interaction between robots

Programming the robots means making the robots do what you want to do, to do so every robot has a different program to understand what to do and to process orders, in this case we need to use 2 different software as previously mentioned, for the Omron LD90 we need to use Mobile Planner and to control the Omron TM Handling robot TM Flow.

After we have created both programs in the respective programs we need to use the built.in PLC, that we can find in the MoMa contraption, in order to



communicate the LD90 with the TM5 Handling robot, as a premise we know that the LD90 is de “master”.

After this is located on the starting position a signal has to be sent to the PLC so it indicates the TMFlow software that is time to start the procedure, and after we need to do the same but the other way, when the TM5 has finished moving the objects, and is located in the “safe to move” location, a signal has to be sent to the PLC, so later on the LD90 will know that can continue with the movement, and go back to the starting point.

To start with firstly we need to indicate 2 goals where the LD90 has to go, Goal 1 and Goal 2, making reference to:

- Goal 1: Position where the robots picks up the pieces from the workspace 1 to the fixture on the LD90.
- Goal 2: Position where the robot will unload the pieces to the second workspace located in a separated table.

After the LD90 reaches Goal 1 we have to send the following to TMFlow in order to start the movement of the arm:

```
arbitraryServerSend(192.168.250.4,61200,1)
```

```
waitActive(never,5,Waiting,False,,True)
```

the previous code will make the TMFlow program do the first task, the one that we have already prepared, and to follow it will wait until the task is finished, and then the LD90 will continue his route to Goal 2.

After the robot reaches Goal 2, we send:

```
arbitraryServerSend(192.168.250.4,61200,2)
```

```
waitActive(never,5,Waiting,False,,True)
```

So, the robot is located on the second location, next to the second workspace that is on top of the second table, after this it will wait until the unload of the pieces is done.

Therefore, the participation of the LD90 AGV has finished.

(Jasper Sjoblom, 2020, pp.42)

## 5 Findings and proposals

The main findings of the thesis are the versatility and easy use of the robot, from my point of view this robot will be used for developing some projects for companies that are interested on the robot, in the other hand having this robot in the university is a good value proposal for students that may want to join the university in the near future, not all universities are capable of having this kind of robots as well as all the other robots and facilities that TAMK has.

A secondary finding has been that an important factor, when using the landmark alignment, is the lighting environment because in my case we only had one original landmark in order to do everything, so in order to be able to accomplish everything we used paper printed copied. These copies are less solid and therefore they flex more and reflex the light in strange ways, which make the vision system incapable of detecting the landmark.

To solve this issue, the most recommendable solution is to purchase original Landmarks or activate the torch located at the end of the robotic arm, in order to light up the location and be able to read the landmark

My proposals for the near future about the implementation of the robots would be teach students the basic capabilities about the robots and then develop a industry related procedure, having this into mind a good value proposal would be implementing the versatility of this robot, that is capable of moving objects around, with the capabilities that fix robots have, for example making some robots collaborate, using the ABB robot that has 2 robotic arms to assemble to metallic pieces, afterwards the MoMa robot taking the pieces from that robot to the welding robot and after the pieces are welded together relocate them in a storage facility.

The previously mentioned example is one of the infinite procedures that all the robots at TAMK can develop, because seeing the future of the big 3D printer another collaboration between robots and the 3D printer could be created if someone orders a big 3D printed object that is made of different small parts, as seen recently with the pine cone printing, therefore a lot of man hours would be deleted from the project and possibly making the final product not as expensive as it may be.

## REFERENCES

Wikipedia ( 03 2021). Wikipedia Web Site. Web Site. Read 04.03.2021 from:  
<https://es.wikipedia.org/wiki/Omron>

Britannica (05 2020). Britannica Web Site. Web Site. Read 04.03.2021 from:  
<https://www.britannica.com/technology/robot-technology>

Omron Corporation & Techman Robot (10 2018). Omron Corporation Assets Web site. Manual. Read 04.03.2020 from:  
[https://assets.omron.eu/downloads/manual/en/v1/tm\\_flow\\_software\\_manual\\_in\\_stallation\\_manual\\_en.pdf](https://assets.omron.eu/downloads/manual/en/v1/tm_flow_software_manual_in_stallation_manual_en.pdf)

Omron Corporation (10 2018). Omron Corporation Automation Web site. Read 04.03.2021 from:  
<https://automation.omron.com/en/us/products/family/Omron%20TM%20Software>

Omron Corporation (10 2018). Omron Corporation Assets Web site. Manual. Read 08.03.2021 from:  
[https://assets.omron.eu/downloads/manual/en/v1/i611\\_mobile\\_robots\\_oem\\_-\\_ld\\_platform\\_users\\_manual\\_en.pdf](https://assets.omron.eu/downloads/manual/en/v1/i611_mobile_robots_oem_-_ld_platform_users_manual_en.pdf)

Omron Corporation (2013). Intechchennai Web site. Manual. Read 10.03.2021 from:  
[https://www.intechchennai.com/micro-site/omron-solutions/mobile\\_manipulator/pdf/Mobile-Manipulator-for-Digital-Industry.pdf](https://www.intechchennai.com/micro-site/omron-solutions/mobile_manipulator/pdf/Mobile-Manipulator-for-Digital-Industry.pdf)

Omron Corporation (2018). Omron Corporation Assets Web site. Brochure. Read 10.03.2021 from:  
[https://assets.omron.eu/downloads/brochure/en/v7/i834\\_fleet\\_manager\\_brochure\\_en.pdf](https://assets.omron.eu/downloads/brochure/en/v7/i834_fleet_manager_brochure_en.pdf)

Omron Corporation (2018). Omron Corporation Assets Web site. Manual. Read 10.03.2021 from:  
<https://assets.omron.com/m/69c1134242668d6d/original/Enterprise-Manager-2100-User-Manual.pdf>

Omron Corporation (03 2018). Omron Corporation Assets Web site. Manual. Read 10.03.2021 from:  
[https://assets.omron.eu/downloads/manual/en/v2/i623\\_collaborative\\_robots\\_hardware\\_installation\\_manual\\_en.pdf](https://assets.omron.eu/downloads/manual/en/v2/i623_collaborative_robots_hardware_installation_manual_en.pdf)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:  
[https://www.youtube.com/watch?v=mEg3o6u9GXY&t=1s&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=mEg3o6u9GXY&t=1s&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=q7I9BLfLO2E&t=186s&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=q7I9BLfLO2E&t=186s&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=7jGXwGiYV3g&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=7jGXwGiYV3g&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=EvNWMw73Uo8&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=EvNWMw73Uo8&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=iFaPP2xLVuM&t=249s&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=iFaPP2xLVuM&t=249s&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=b3cMp1r8skw&t=25s&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=b3cMp1r8skw&t=25s&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=YmVVqDfh1qs&t=2s&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=YmVVqDfh1qs&t=2s&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=S\\_28vHxjc44&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=S_28vHxjc44&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation [Omron Industrial Automation EMEA]. (11 2018). Youtube Web Site.Video. Read 14.03.2021 from:

[https://www.youtube.com/watch?v=A6dy7Ro0QW4&ab\\_channel=OmronIndustrialAutomationEMEA](https://www.youtube.com/watch?v=A6dy7Ro0QW4&ab_channel=OmronIndustrialAutomationEMEA)

Omron Corporation (2021). Omron Corporation Assets Web site. Web Site. Read 20.03.2021 from:

<https://web.omron-ap.com/ph/mobile-manipulator-digital/>

Robotiq (05 2015). Robotiq Web Site. Blog. Read 23.03.2021 from:

<https://blog.robotiq.com/what-is-a-cobot>

6 River Systems (03 2021). 6River Web Site. Web Site. Read 23.03.2021 from:

<https://6river.com/what-are-automated-guided-vehicles/#:~:text=Sometimes%20called%20self%2Dguided%20vehicles,an%20onboard%20operator%20or%20driver.>

Techman Robot(2019). TM Robot Web Site. Blog. Read 29.03.2021 from:  
<https://www.tm-robot.com/en/regular-payload/>

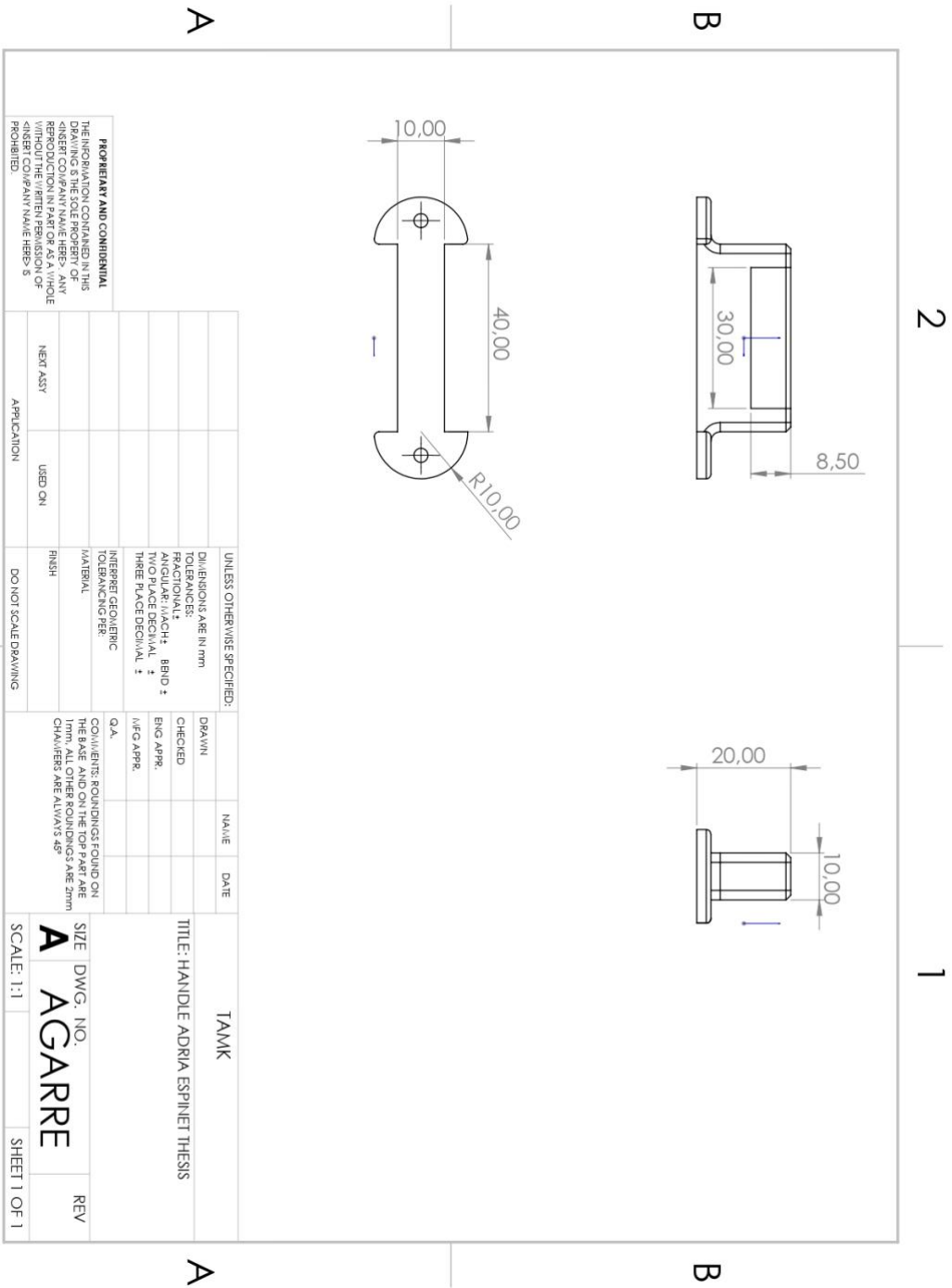
Omron Corporation (01 2017). Omron Corporation Edata Web site. Manual.  
Read 29.03.2021 from:  
<https://www.edata.omron.com.au/eData/Robotics/l617-E-01.pdf>

Atria Innovation (2021). Atria Innovation Web site. Manual. Read 29.03.2021  
from:  
<https://www.atriainnovation.com/agv/>

Tampereen ammattikorkeakoulu [TAMK]. (06 2020). Theseus Web Site.Thesis.  
Read 07.04.2021 from:  
[https://www.theseus.fi/bitstream/handle/10024/342192/Sjoblom\\_Jesper.pdf?sequence=3&isAllowed=y](https://www.theseus.fi/bitstream/handle/10024/342192/Sjoblom_Jesper.pdf?sequence=3&isAllowed=y)

APPENDICES

Appendix 1. Handle drawing



**PROHIBITAM Y CONFIDENCIAL**  
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF AND IS NOT TO BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS, WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> S (PROHIBITED)

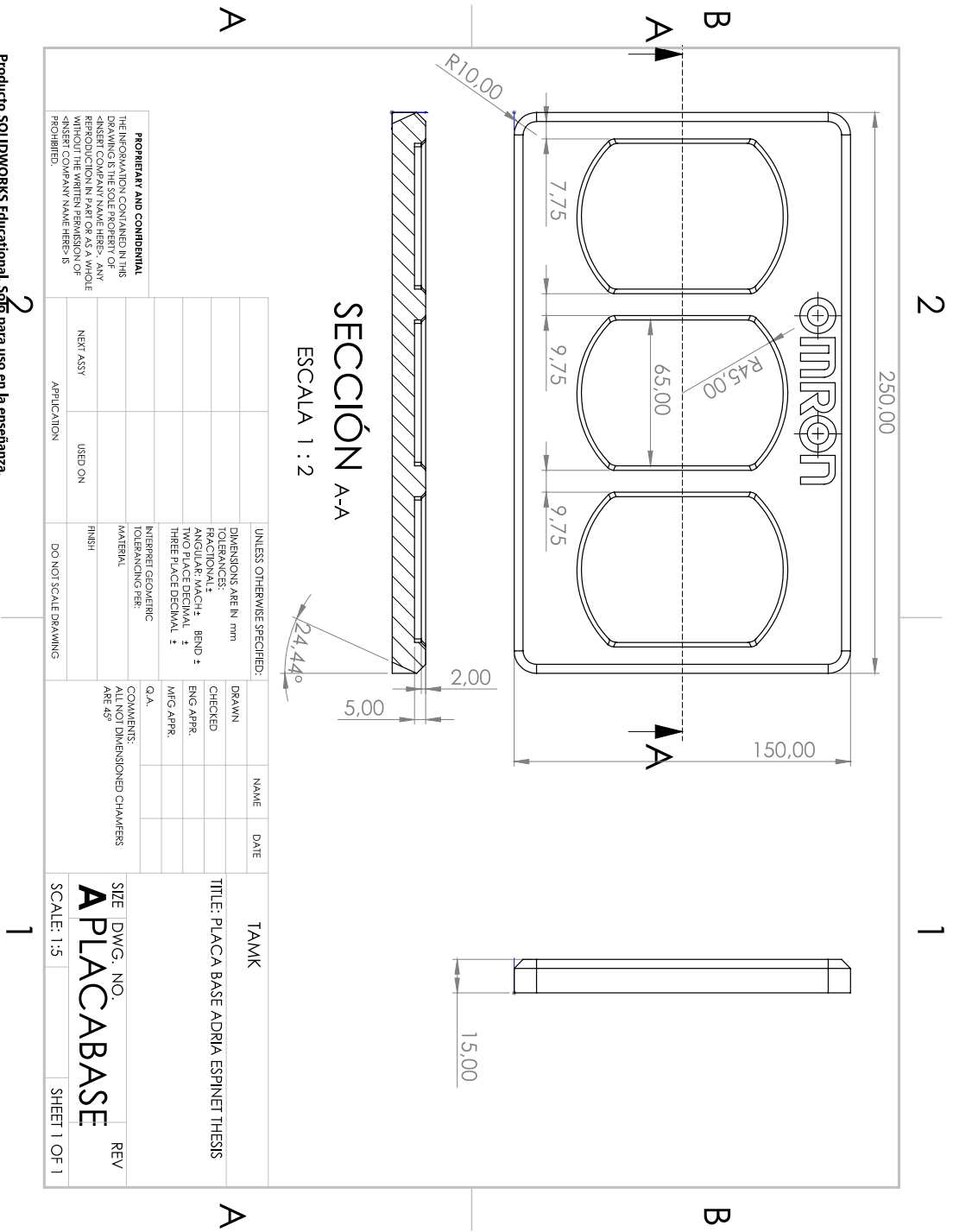
UNLESS OTHERWISE SPECIFIED:		DRAWN	NAME	DATE
DIMENSIONS ARE IN mm		CHECKED		
TOLERANCES:		ENG APPR.		
FRACTIONAL: ±0.10		INFO APPR.		
DECIMAL: ±0.10				
HOLE: ±0.05				
TWO PLACE DECIMAL: ±0.05				
THREE PLACE DECIMAL: ±0.02				
INTERPRET GEOMETRIC TOLERANCING PER:		Q.A.		
MATERIAL:		COMMENTS: ROUNDINGS FOUND ON THE BASE AND ON THE TOP PART ARE 1mm. ALL OTHER ROUNDINGS ARE 2mm		
FINISH:		CHAMFERS ARE ALWAYS 45°		
DO NOT SCALE DRAWING				
NEUT ASST	USED ON			
APPLICATION				

TAMK  
 TITLE: HANDLE ADRIA ESPINET TESIS

SIZE DWG. NO. **A**  
 SCALE: 1:1 **AGARRE**  
 SHEET 1 OF 1

Producto SOLIDWORKS Educativo, sólo para uso en la enseñanza.

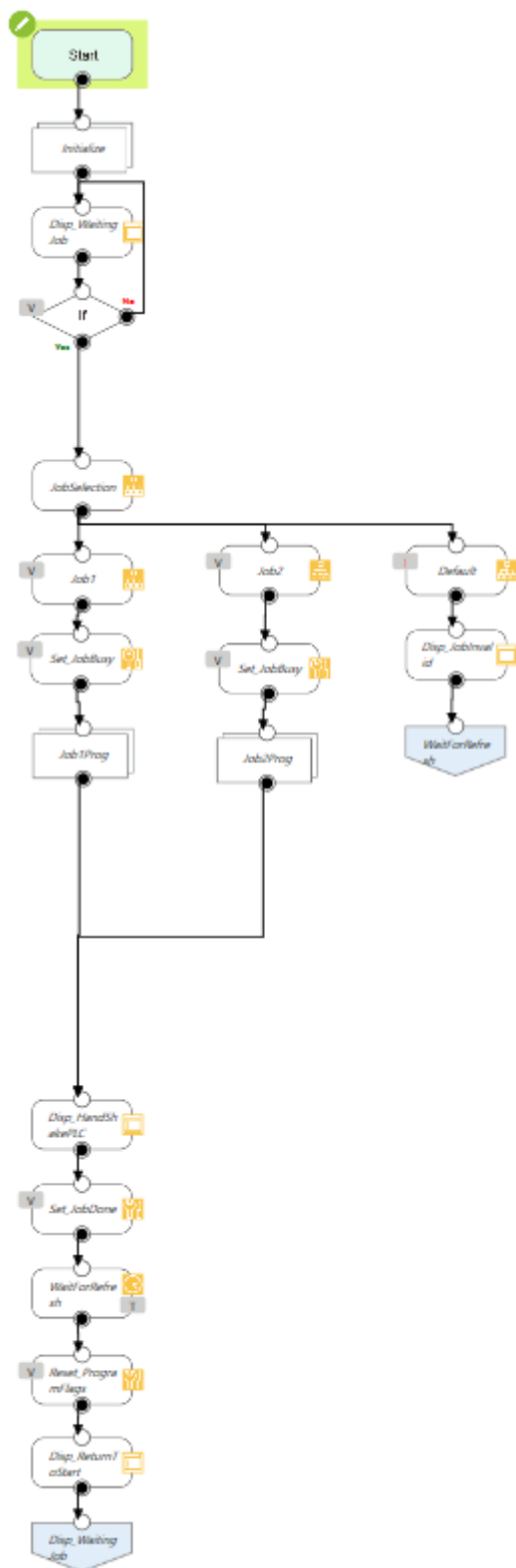
Appendix 2. Placa Base drawing



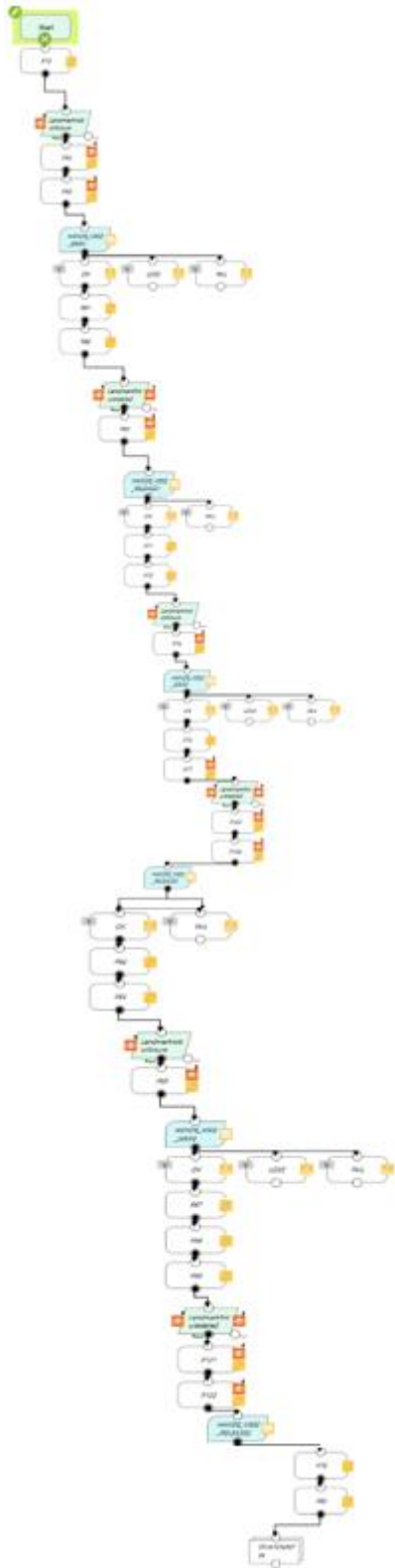
Producto SOLIDWORKS Educational. **2** sólo para uso en la enseñanza.



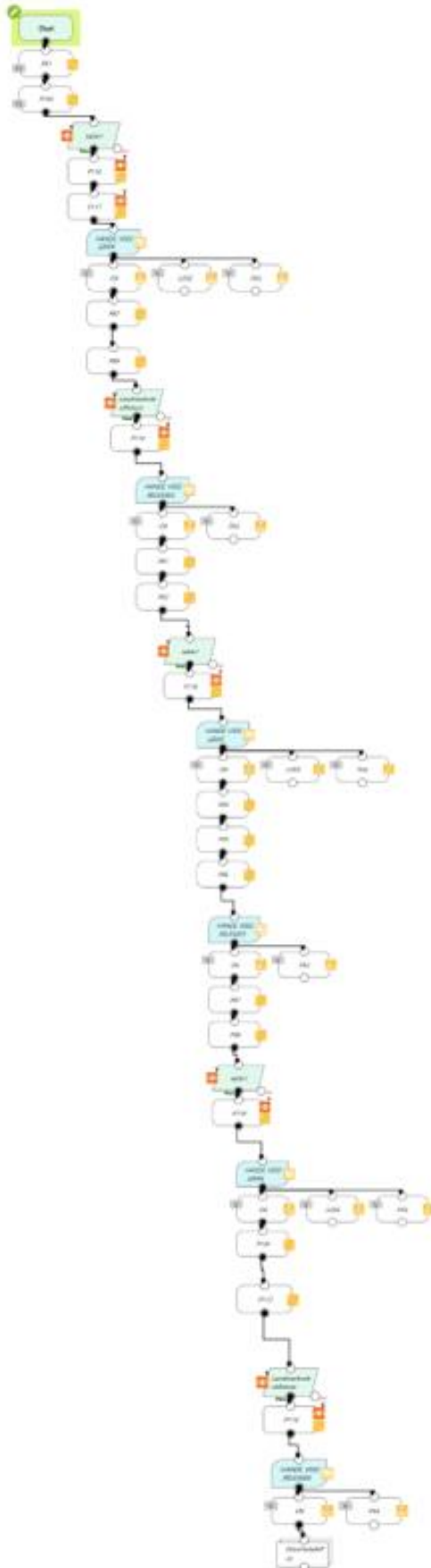
## Appendix 3. TM Flow Program (Main Program)



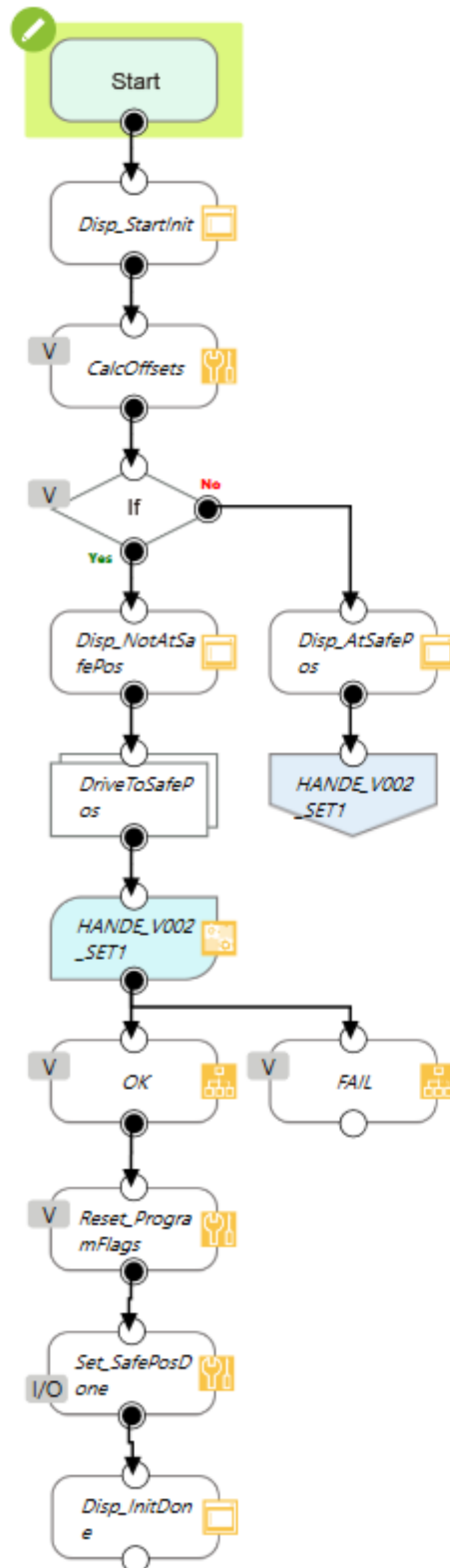
Appendix 4. TM Flow Program (JOB 1)



Appendix 5. TM Flow Program (Job 2)



## Appendix 6. TM Flow Program (Initialize)



## Appendix 7. TM Flow Program (Move to Safe Position)

