LMI Re-store

Final report

28-01-2019

This page has been removed

Preface

This report is written in the context of the LMI Re-Store System project. The project is a part of the 7th semester (4th year) of mechatronics at the Fontys University of Applied Sciences. This project was initially started last year and was undertaken by a group of 8 students for their minor. After finishing the project successfully each student will be rewarded with 10 ECTS (credits).
In this part of the project another group of 8 students will work in collaboration with the Fontys Lectorate M&R. This report is written for those interested in the LMI-restore system or the Mechatronics smart manufacturing environment.

The first chapter details the project goals set by the project group and the recommendations made by the previous group and the starting position of the project. The User requirements, system requirements, and Preconditions can be found in chapter 2. The third chapter depicts what details considerations and choices were made concerning the improvements. The fourth chapter is about testing the whole storage to see if all the requirements are made.

The goal of the project is to develop an automated storage system that is part of a smart manufacturing environment in which robots from different manufacturers are able to take goods from the storage and deliver them to a robot cell.

We want to thank our tutor Wilco Pancras for the guiding and support he gave us.

Eindhoven 28-01-2019

Summary

A project group of Fontys students created a working smart storage system during the minor "Adaptive Robotics". This system is able to store 100 containers, pick them up and hand them over to an AGV and the other way around. Small components such as bolts and nuts could be stored in those containers. Time, space and picking mistakes will be reduced while using this system. The system did not work optimal at the start of the project. The goal of this project is implement improvements and add extra functionality to the smart storage.

One of the extra functionalities that are added to the system is, queuing. Three queue places are created in the storage to form a buffer for incoming and outgoing containers. The queue increases the lead time (from ordering the container till receiving the container from the AGV) and the time an AGV needs to stand before the storage will decrease. The places in the queue are also able to weigh the containers in the queue. If a container is empty, the container will not leave the storage and this information will be saved in a database. The database could now be used to order new components.

The next functionality that is added to the system is: detecting content of containers. If a user wants to order a container from the smart storage, the system needs to know what the contents of the containers are in the storage. A barcode scanner, together with barcodes on the containers will add this new feature. The barcodes of the containers are saved in a database.

The customer wanted a storage system where humans could take content out of the container themselves. To make this possible the front plate of the storage was replaced by a new one. With this new front plate it is not possible to remove the container by hand. Humans can grab some components out of the container, but not the whole container. This is important because if humans could pick containers out of the storage the database will be incorrect.

Besides those extra functionalities, some improvements were added to the system:

- *Adding brake on Y-axis*: This prevents the arm from falling down when the power cuts off.
- *Replacing microswitches:* Limit switches were swapped for photoelectric sensors to prevent damage on the limit switches.
- *Replacing bearing on R-axis*: This reduces the play in the arm for rotation.
- *Creating electrical schematic*: A complete and clear electrical schematic has been made in EPLAN.
- *Motion planner:* With the motion planner it is now possible to move multiple axis at the same time.
- *CAD database:* This establishes a clear overview of all Solidworks parts, assemblies, drawings, DXF and other 3D files in excel.
- *GUI:* A touchscreen with a user interface is added in the door of the electrical container, to make it easier for users to use the storage system.

Recommendations for next project groups are designing and implementing a new arm. A new arm could make the system faster and more reliable. Besides this, programming a routine scan to check if the database correspond with the actual system.

Contents

1. Introduction

In a manufacturing environment, components such as bolts and nuts are usually stored in containers which are then manually retrieved by workers. This process is labor intensive and very prone to errors because of the ever-existing human factor. Medium to large engineering companies also have a different logistics location for storing materials that are used in the production of machinery and systems. Taking this into account together with the human factor a significant amount of time and space can be saved, while picking mistakes will be reduced with the use of a smart storage system and an AGV delivering the materials.

The lectorate of Fontys University of Applied Sciences wishes to participate in Industry 4.0 in the near future. This means that the different systems and applications used within the lectorate are connected and able to communicate with each other through certain protocols. The client within the lectorate reckons it would be convenient to have a certain automated smart storage already available once the other systems or applications are made available.

Given the circumstances wherein the system should operate, the client would like a complete separate system which can function on itself. No additional add-ons should be required, neither should any software modification on AGV's (or fleet managers) be necessary. An AGV, or any fleet manager in particular, should only request a certain container, drive towards the storage, and retrieve the container. [1]

1.1 Smart Storage (previous project)

Prior to this project, another project group of 6 Fontys students created a working smart storage system during the minor “Adaptive Robotics” (Figure 1). This system consists of a hundred storage cells were containers (also referred to as containers) with items in them could be stored. The robotic arm in front of the storage was able to pick up a container and deliver it to a predefined location.

Figure 1 The smart storage system from the Adaptive robotics minor.

After the minor, most parts of the machine were powder coated for aesthetic purposes. This created some problems with the connections and alignments of the mechanical components of the storage. And because the storage had to be taken apart to be coated, all the electronics were disconnected.

The starting point for the LMI Re-store group was no working storage, due to the coating process. All the electronics and software were not documented well, so the electronics could not be reconnected right away and testing the machine was not possible.

2. Objectives and requirements

To describe the goal of the project, the objectives and the derived requirements are listed in tables in this chapter.

2.1 Objectives

The project goals that were presented by the customer (Michiel van Osch) are listed below in Table 1 [2].

Project goals	
PG-1	Queueing containers to be picked up and received at any place in front of the storage system.
PG-2	Detecting containers and reading content.
PG-3	Placing a container somewhere else and memorizing the storage location if a place is occupied.
PG-4	Detecting empty containers.
PG-5	Sorting containers to the storage cells.

Table 1 Project goals

The project group from the adaptive robotics minor (the previous group) had the following recommendations (Table 2) [1]:

Recommendations	
R-1	Electrical / mechanical brake when the voltage drops off, when the voltage is dropping at the moment the platform falls down and parts can be damaged.
R-2	The storage is a little bit too deep, now the containers can get too far in the storage so that the robot cannot grab anymore.

Table 2 Recommendations

2.2 User requirements

Together with the customer, the project goals [2] and the recommendations from the previous project group [1] were bundled into the following user requirements (see Table 3).

User requirements	
UR-1	The storage system should have a queue to quickly receive and deliver multiple containers.
UR-2	The storage system should detect if a container is present in a storage cell.
UR-3	The storage system should be able to determine the content of the containers.
UR-4	The storage system should relocate a container if the storage space is occupied.
UR-5	The storage system should detect empty containers
UR-6	The storage system should place containers into the storage cells.
UR-7	Humans should be able to retrieve containers from the storage system in a safe manner without using the robot arm.
UR-8	Add electrical / mechanical brake when the voltage drops off, to prevent the platform from dropping and damaging parts.

Table 3 User requirements

2.3 System requirements

From the user requirements the following system requirements were derived. These requirements are prioritized using the MoScoW method [3] and are listed in Table 4.

The system requirements are divided into multiple categories:

- SR 1.x – queuing
- SR 2.x – location of containers
- SR 3.x – contents of containers
- SR 4.x – Interaction with environment
- SR 5.x – Safety
- SR 6.x – Improvements

System requirements	
SR-1.1	The system must be able receive 3 containers into a queue, that are later placed into an empty storage cell.
SR-1.2	The system must be able to pick up 3 containers from the storage, bring them to the queueing area. Later, these containers will be handed over to the AGV when it is ready for pickup.
SR-2.1	The system must be able to detect in which storage cell a container is present.
SR-2.2	The system must relocate the container to another storage cell if the storage cell is occupied.
SR-3.1	The system must determine the content of the container delivered by the AGV.
SR-3.2	The system must know the content of every container in the storage.
SR-3.3	The system must detect an empty container.
SR-4.1	The system should give a notification to the user/fleet manager when a container is empty.
SR-4.2	The system should grab a container from an AGV and place it in an empty storage cell.
SR-4.3	Human operators should be able to access the contents of the containers within the storage unit without using the robotic arm.
SR-5.1	The system must prevent human fingers from getting stuck in the moving parts.
SR-5.2	The robotic arm should not fall down in the case of a power outage.
SR-6.1	Hitting the end stops of the X,Y,R and Z axis must not damage or wear out the sensor in any way.
SR-6.2	The play in the arm allows the arm to tilt no more than 2 degrees in any direction
SR-6.3	The arm should be able to reach at least 40mm into the storage cell
SR-6.4	An electrical scheme could be present in/on the electrical container.
SR-6.5	A database could be available containing all CAD files.

Table 4 System requirements

2.4 Preconditions

During this project, some preconditions are applicable. These include:

- The storage unit itself doesn’t have to be built. This has already been done by the previous project group as part of their minor.
- The communication through the fleet manager is being done by the K.I.M. (Keep IT Moving) project which runs parallel with the LMI re-store project.

3 System design and realization

In this chapter the design and implementations of the various components and improvements of the LMI re-store system are discussed. Each paragraph contains a solution linked to the system requirements of §2.3.

3.1 Physical Queue (SR-1.1 and 1.2)

To form a solution for system requirements 1.1 and 1.2, a buffer for outgoing and incoming containers (a queue) had to be created.

Multiple concepts for this problem were generated, these ideas were ranked using the morphological chart (Figure 2).

		Low costs	Low complexity	High speed	hoge robustness	High flexibility	Total score
Queing	Turntable external	3	2	6	4	4	19
	Second storage unit	4	4	2	5	6	21
	Designated places in storage	6	6	4	6	5	27
	Elevator	1	2	3	2	1	9
	Turntable on arm	2	3	5	3	3	16
	Stack	4	1	1	1	2	9

Figure 2 Morphological chart for the queueing system.

Making some existing places of the storage system designated places for queuing seemed to be the best overall solution. Because the system already has a hundred places for storage containers, taking a few places away for queueing seemed to be a small adjustment.

The final design for the queue has place for 3 containers and is positioned on the 5th row near to the electro container (Figure 3). This position was chosen because the AGV that will pick up the container will positioned on the right side (close to the electronic connection container. In this way, a short travel distance between the queue and the AGV is created.

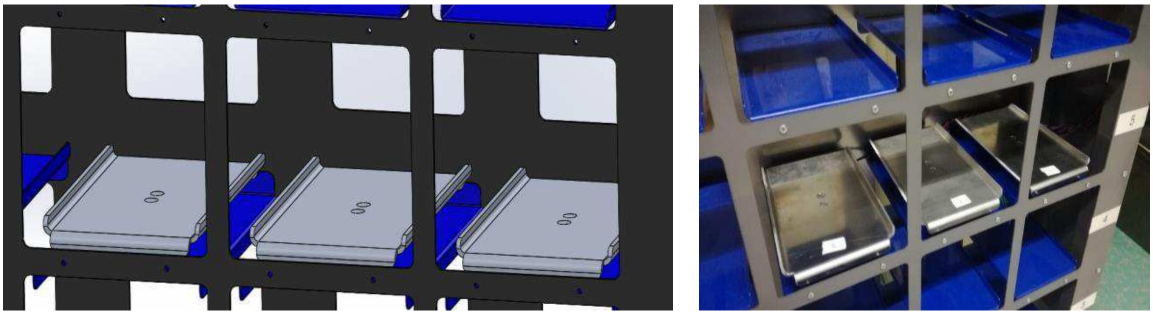


Figure 3 End product of the queueing module

While containers are in the queue their mass is measured to determine if they are almost empty or not, this is explained in more detail in §3.3.

3.2 Presence and content of a container (SR-2.1, 2.2, 3.1 and 3.2)

The content of containers delivered to the system and inside the system must be known at all times (SR-3.1 and 3.2). The presence of the container in a storage cell should also be measured, so it can be determined if a place is occupied (SR-2.1 and 2.2).

Since humans can’t take out an entire container themselves (see §3.5 for more info), the system only needs to recognize the contents of the container when the container is collected from the storage or delivered to the AGV.

For detection of the containers and their contents there were several options to choose from (see Figure 4).

		Low costs	Low complexity	High speed	hoge robustness	High flexibility	Total score
Detecting content of box	Vision	2	1	2	2	1	8
	QR/barcode	5	4	3	4	4	20
	RFID/NFC	3	3	4	3	3	16
	Visual inspection by human operator	1	2	1	1	2	7
Detecting presence of box	Vision	3	2	7	2	1	15
	QR/barcode	6	5	4	7	3	25
	RFID/NFC	4	4	3	6	4	21
	Visual inspection by human operator	1	1	1	1	2	6
	Ultrasonic	7	7	5	3	5	27
	Laser sensing	5	6	6	4	7	28
	Feeler probe	2	3	2	5	6	18
	Database	8	8	8	8	8	40

Figure 4 Morphological chart for the detecting contents and presence of containers.

To determine the content of a container (SR 3.1 and 3.2), QR/barcode scanning seemed the best option. Barcodes can be cheaply printed with a label printer and scanned from a small distance with a barcode scanner. Since RFID/NFC requires tags that need to be bought and needs a very close distance to the scanner in order to work, a barcode scanner seemed to be a more flexible and cheaper option than RFID/NFC. Of course storing the values of the barcode scanner will be done in the database.

To detect the presence of a container (SR-2.1 and 2.2), using a database was by far the most logical option. Because of the elimination of the human factor (humans taking out containers) the storage system is the only entity that is allowed to move the containers. Therefore solely adding a database will offer an impermeable solution that requires no extra sensors or other physical modifications to the system.

The barcode scanner and the database are used together to save the exact location of ,for example, a container filled with M5 bolts.

In the final design the barcode scanner was implemented on the robotic arm of the storage system (Figure 5). This was done to have the fastest and most robust system. If the scanner would have been placed at a fixed point, then the arm needs to come to the scanner every time a container will be entered into the system which will lose more time.

The scanner is a module from Waveshare [4]. It has an USB port where it emulates a keyboard, typing the codes that are read when the user presses the on-board button. Full control over the module is provided over the UART interface. Here, a host can send commands to control the scanner, for example to start scanning. The scanner sends data back with the code of the barcode. This specific scanner was chosen because it was the only scanner available from the suppliers that wasn’t a handheld device.

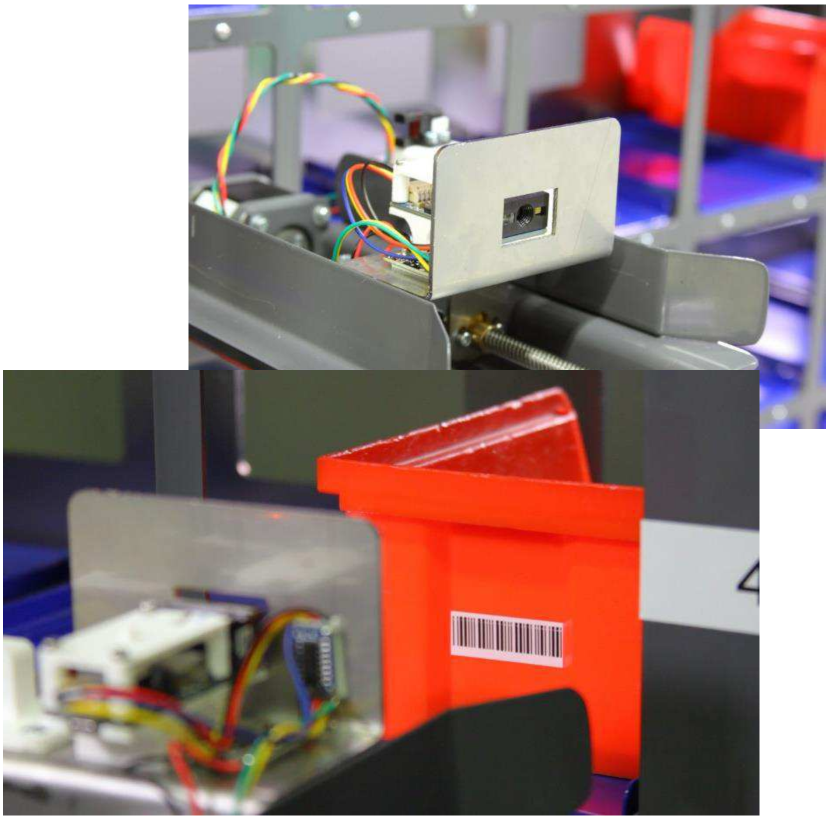


Figure 5 The barcode scanner on the robotic arm.

The database that was created for remembering the location and content of all the containers of the storage system, is a MySQL database with the layout seen in Figure 6. The database contains only one table with the following columns:

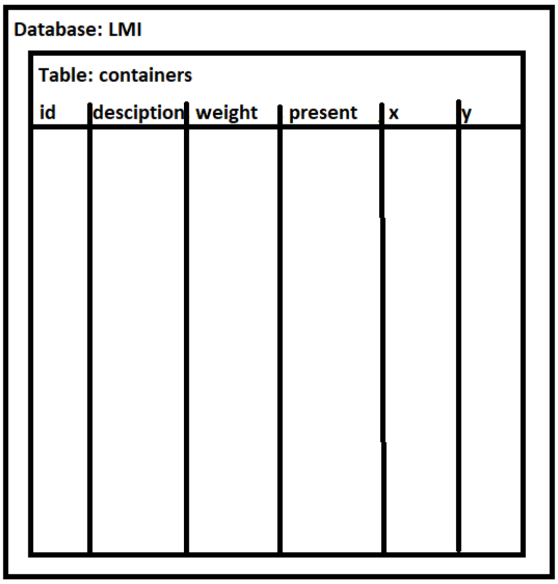


Figure 6 Database LMI

ID, The identification number (id) is a unique numeric code which is the same code that is printed on the barcode of the individual containers. This number is also stored first in the database as it is the number which interface and the software on the raspberry pi use to search for certain containers.

Description, The description is a short explanation of the goods inside the containers. This description is inserted using the interface when storing a container. A description can for instance be:"M5 bolts" or "Voltage regulator 12V" to inform others what's inside the containers

Weight, This value will be the weight after storing or after obtaining a container in grams. By knowing the weight of the component and the weight of the container, one can see the possibility of it being (almost) empty. The sensing of the weight (and thereby the detection of empty containers) of a container will be discussed in §Detecting empty containers (SR-3.3)3.3.

Present, this value is used by the raspberry pi to check what containers are present in a storage cell.

X, The X value will be the column in which the container is or must be placed. When x is 1, the position of the container is in column 1.

Y, The Y value will be the row in which the container is or must be placed. When Y is 1 the position of the container is in row 1.

Commands

To access, select and alter the values of the database. Several commands can be used. Because the main program is written in Python 3*1, the commands will look like the commands in Figure 7. This piece of code show the connection of the database by asking for the host, username, password and database name of the MySQL-server. After that it will define the cursor and below that the MySQL command is defined.

```
Mydb = mysql.connector.connect(
        host="localhost",
        user="LMI",
        passwd="***",
        database="LMI"
)

Mycursor = mydb.cursor(buffered=True)

Sql = 'SELECT x,y FROM containers where present=1'
mycursor.execute(sql)
taken = mycursor.fetchall()
```

Figure 7 Python code for acessing the database.

3.3 Detecting empty containers (SR-3.3)

In the queue (see §3.1) the weight of the containers is measured to detect if they are almost empty or not. A load cell with a maximum payload of 5 kg will be used for weight sensing Figure 8.

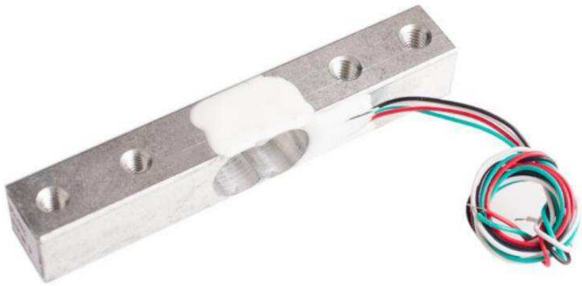


Figure 8 load cell

On top of the weight sensor a 3 mm aluminum plate is placed to support a container Figure 9. The existing plate that was used as the bottom of the storage cells is flipped to lower the weight sensing system (Figure 2). This way the entire weight sensor will only be 5,30 mm higher compared to the other work cell.

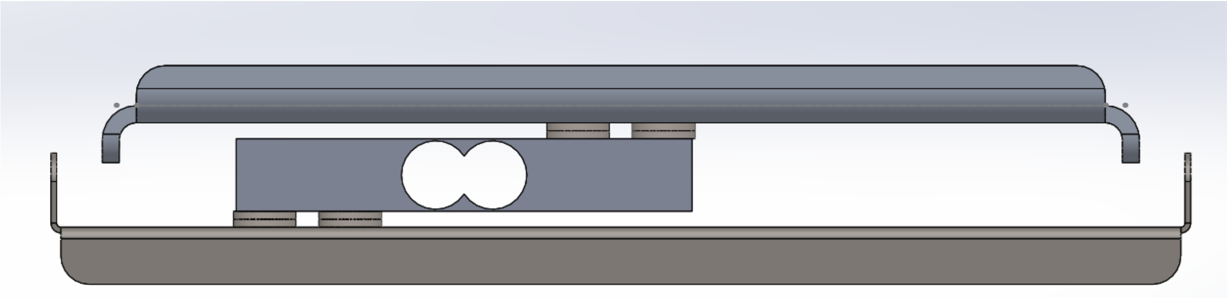


Figure 9 The load cell integrated.

3.4 GUI (SR-4.1 and 4.2)

To give a notification to the user/fleet manager if a container is empty (SR-4.1) some sort of feedback system had to be designed. Also it had to be possible for an AGV to add a container to the storage system (SR-4.2). Since communicating with an AGV was not possible in the project time, it was decided (in consultation with the customer) to implement an user interface. With this user interface storing and retrieving containers is done in the exact same way as it would be when using an AGV. The delivering and pick-up point is the same as it would be for an AGV, but the actions of the system are triggered by a human and not an AGV.

When a human operator requires a product from the storage system they have two options: collect their product from the containers on the human side of the storage system (see §3.5 for more info), or acquire a container from the robot side.

Retrieving products from the human side is done by hand, on the robot side, directly retrieving a container is not possible due the safety risks associated with the movement of the robot. Thus a graphical user interface (GUI) is used for safely retrieving a full container from the robot side as a human operator.

The GUI is displayed on the touchscreen that is located at the door of the electro container (Figure

Figure 10 The GUI is located on the electro container.

10)

Via the home menu of the GUI a user can add a container, request a container or perform a routine scan (Figure 11).

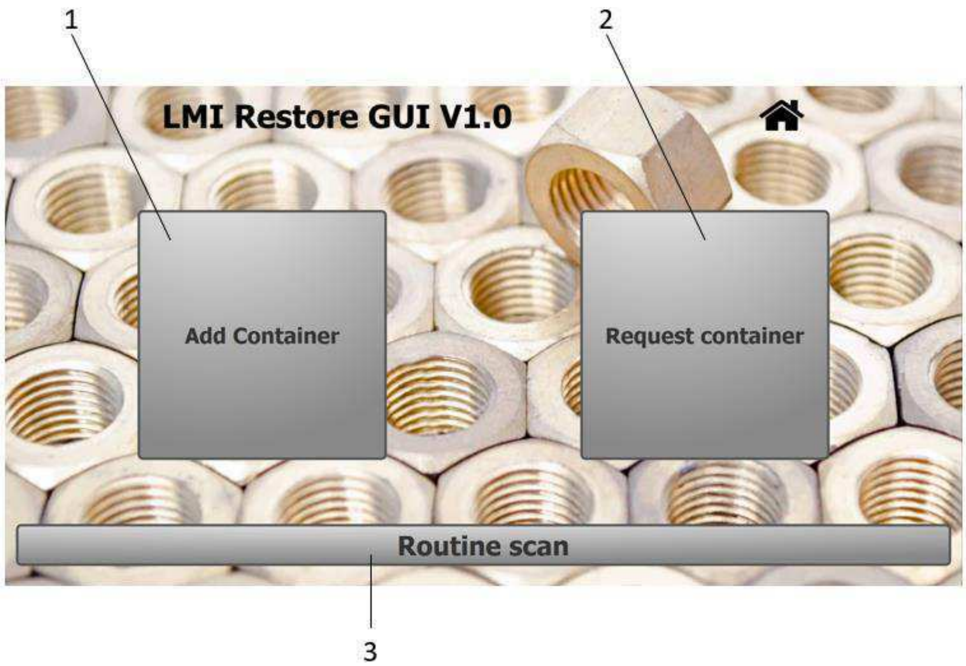


Figure 11 The home menu of the GUI. With (1) add container button, (2) request container button and (3) routine scan button.

After clicking the “Add container” button from the home menu, the following steps need to be taken (Figure 12).

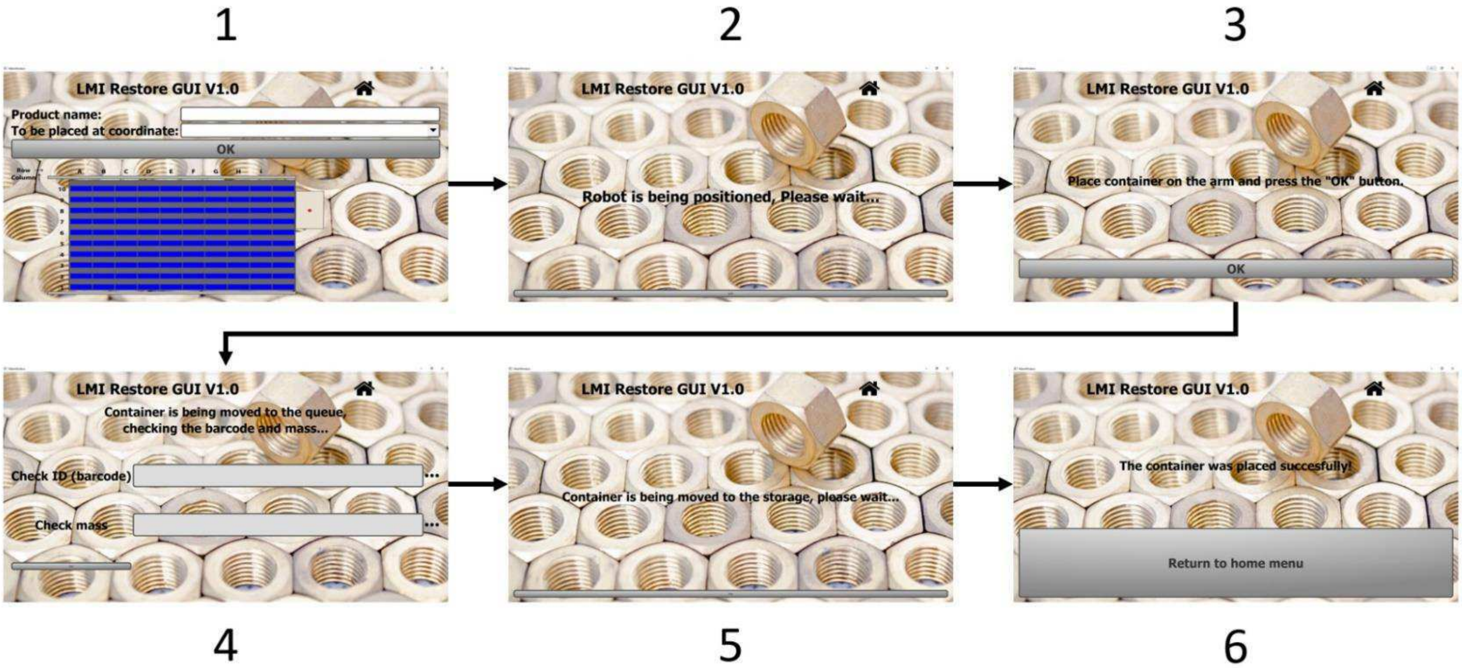


Figure 12 Steps/screens of adding a container.

(1) The user needs to insert the product name (for example M5x30 bolt) and the coordinate (for example A1) where the container needs to be placed. The coordinate is chosen from a list of available coordinates (corresponding to the row and column of the storage unit) via a dropdown list.

(2) The robot will be positioned nearby the user. The user is requested to wait.

(3) When the robot has stopped moving, the interface notifies the user that the container can be placed on the arm. The user needs to press the “OK” button after the container is placed on the arm.

(4) After the user has acknowledged that the container is placed on the arm, the robot will move the container to the queue. Here the barcode and mass are being checked, the values are being displayed.

(5) The container will be moved to the coordinate that was filled in. The user is requested to wait.

(6). The container was placed successfully, the user can return to the home menu.

After clicking the "Request container" button from the home menu, the following steps need to be taken (Figure 13).

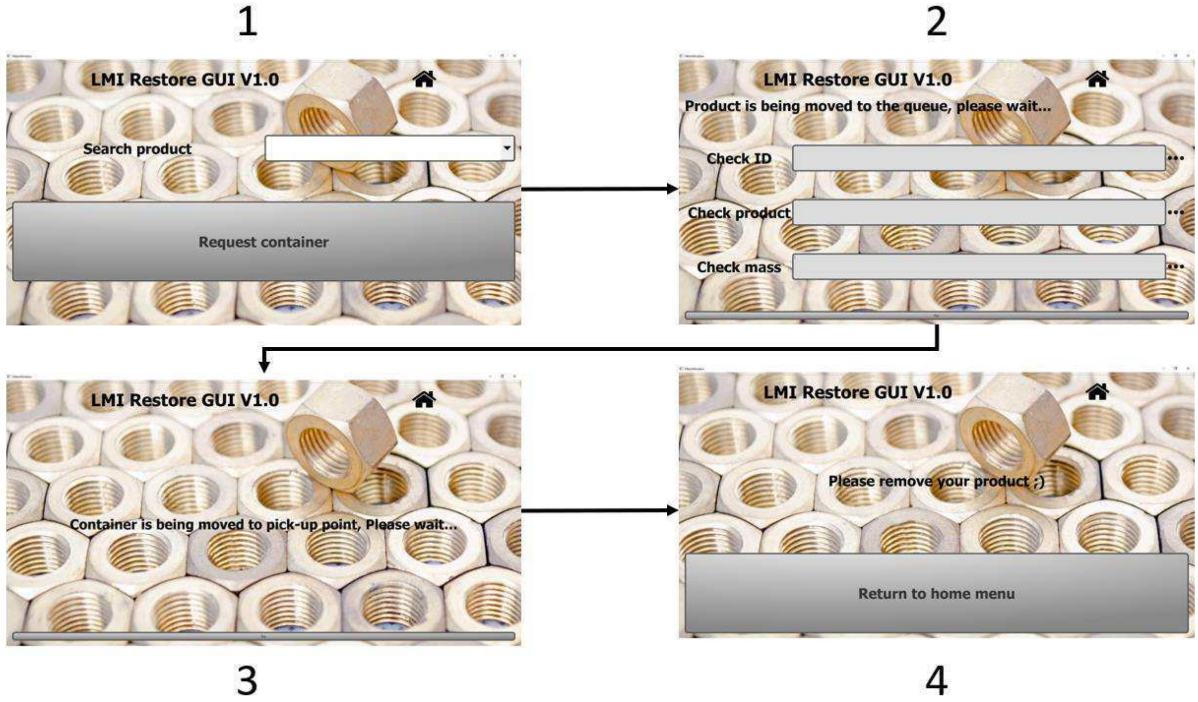


Figure 13 Steps/screens of requesting a container.

(1) The name of the product must be filled in by the user. This can be entered via an editable dropdown menu (editable in the way that the product name can either be typed in or chosen from the dropdown list).

(2) When a product is chosen, the corresponding container is moved to the queue. There, the ID (number of the barcode), the product name and the mass are being checked.

(3) The container will move to the pick-up point. This is the same place were containers can be placed on the arm during the add container process.

(4) When the robot has stopped moving the GUI notifies the user that the product can be removed. The user can return to the home screen after the product is removed.

A routine scan checks all the storage cells of the storage system for their ID (barcode) and adds these values to the database. After clicking the “Routine scan” button from the home menu, the following steps need to be taken (Figure 14).

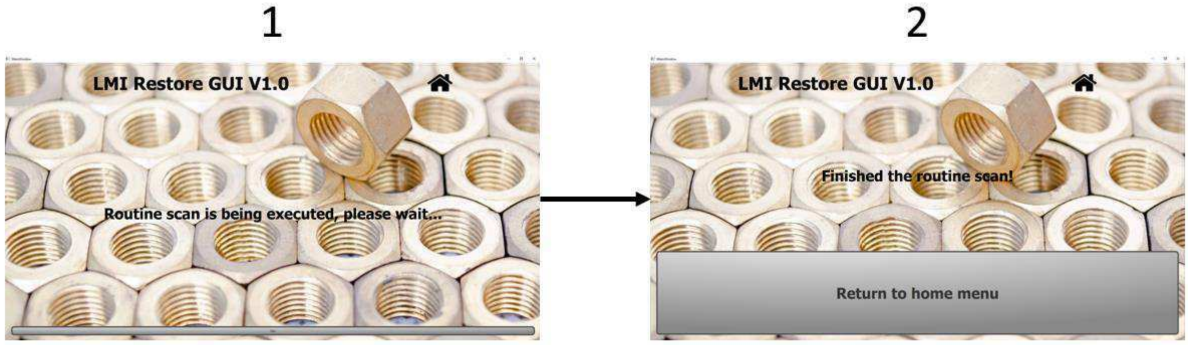


Figure 14 Steps/screens of the routine scan.

(1) The scan will be executed and requires the user to wait.

(2) When the routine scan is finished, the user can return to the home screen.

3.5 Human interaction (SR-4.3 and 5.1)

The customer wants a storage system where humans can take the contents of the containers out themselves. For this reason, the customer requested a system where all the containers could be taken out at any time by a human operator.

To comply with what the customer wants a discussion with the customer was started to get to his real intentions. The outcome was that the storage should be accessible for anyone at any time, so nobody has to wait for the machine to get a container. It was decided that it would be best if everyone could still fetch the contents of the containers but without taking the entire container. To make this possible, the side of the storage unit that is not used by the robot will be used by humans (Figure 15).

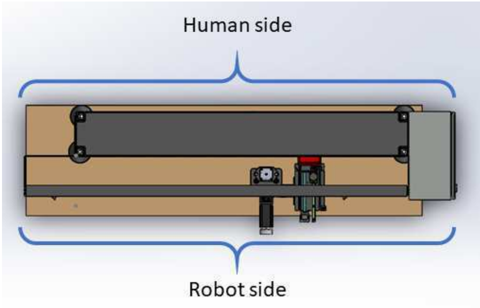


Figure 15 Top of the storage with the human side and the robot side.

To make sure that humans can't remove an entire container a new front plate was added on the "human side"(Figure 16). The holes in this plate are adjusted to only fit a hand and not the whole container.

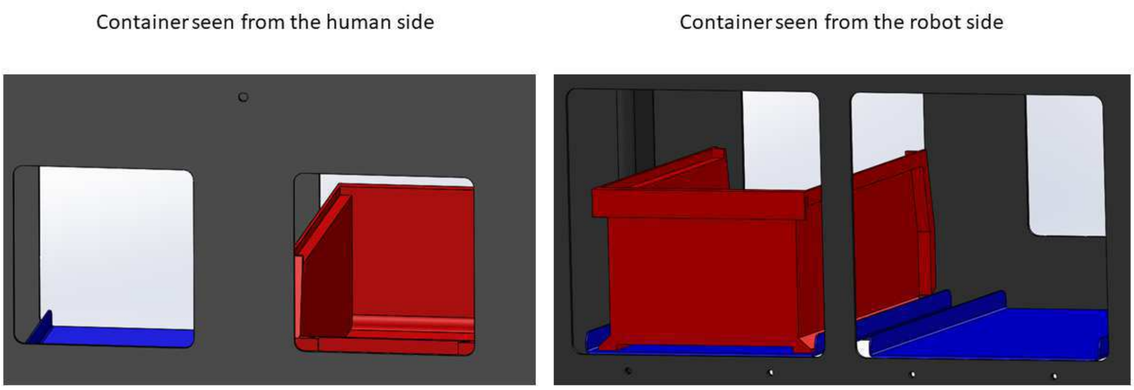


Figure 16 placement of container

To improve safety, it's been decided to use the safety stickers (see Figure 17). This choice was made because of the low cost, low complexity, and high flexibility. High flexibility as the stickers won't impede human interaction. The moving parts of the storage do not have to be accessible for humans, it will be placed close to the wall. Nobody will be able to touch the pulleys or the timing belt, therefore nobody will be harmed when the storage is in operation. And if the storage would be placed differently there will be stickers warning the people before they want to touch the storage. It was decided that using stickers would be the best solution of the morphological map considering that humans are not allowed to retrieve containers from the robot side (Figure 17).

		Low costs	Low complexity	High speed	hoge robustness	High flexibility	Total score
Safety (moving parts)	Warning sticker (ground tape)	4	4	0	1	4	13
	Enclousure entire system	2	3	0	4	1	10
	Covers around moving parts	3	2	0	3	2	10
	laser curtain	1	1	0	2	3	7

Figure 17 Morphological map, safety (moving parts).

3.6 Failsafe Brake (SR-5.2)

In the previous project the arm could fall down when the power was cut-off. This could cause potential damage to the arm, sensors and nearby humans. To improve this, a way of preventing the arm from falling down when there is no power applied to the system had to be created. There were a few solutions for this problem, a damper at the bottom of the Y-axis, a motor brake, counter mass, centrifugal brake and a rail stopper.

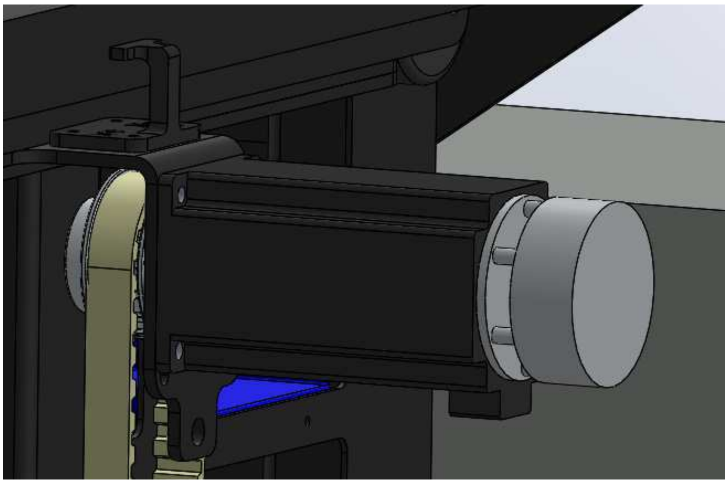


Figure 18 motor Y axis

In the morphological map in appendix 1 the choice was made for the motor brake because this one had the most points. And particularly it has the highest robustness. The motor brake is attached to the shaft of the Y-axis stepper motor Figure 18.

To make sure that the brake is able to hold the arm and a full container, the maximum braking torque should be calculated.

The arm weighs at max 5 kg without the container on it. This would mean that the total maximum weight would be 10 kg. The torque needed on the motor will be equal to the force from the arm with the container times the arm that is in between. So this would be, 98.1N over an arm of 0,018m. therefor the torque needed in the motor break would be, F/m = 1.7 N/m. The chosen motor has a torque of 2N/m, which should be more than sufficient. See appendix 2 for a full calculation with visualization.

Because the motor brake and the motor had to be bought in one package, the torque of the motor was increased from 3,5N/m to 4N/m.

It was possible to also make a counterweight, this would make the motor turn more efficient, but because the motor had enough torque for the job, and the project time is limited the choice was made not to use a counterweight in this application.

3.7 Limit switches (SR-6.1)

The function of the limit switches is to home the motors on the different axis and stopping the motor if the maximum position is reached.

The problem with the microswitches from the previous project, was that the arm needs to touch the sensor physically. When this happens, it could occur that the arm falls or pushes though the microswitch, which would mean that the microswitch will break. To avoid this problem another type of sensor has been chosen.

The limit switches have been replaced by a non-impact variant, this was chosen because these sensors cannot be damaged by the system. The chosen sensor is a u-shaped photoelectric sensor (also referred to as infrared gate sensor). These sensors will give a high value when something passes through the light beam and thereby stop the motor.

There are some alternatives to the infrared gate sensor like an Inductive sensor, a capacitive sensor. The conductive sensor is very sensitive to changes in the environment [5] , therefore this sensor will not be used. The inductive one has one drawback compared to the light gate sensor, it can only detect metallic materials. This limits the designing options [6]. Also, the inductive sensors could influence the motors we use, or the motor on the sensor due to EMC. Therefore, the choice for the infrared light gate sensor has been made.

The infrared gate sensor used are Panasonic PM-series photelectric sensor (Figure 19). This specific one has been chosen because it shows when it is being trigger or not by turning the LED on the device itself on or off. This will help during programming. If the sensor shows that it is detecting an object passing through, but the machine does not stop, there will most likely be an error in the software. Also, the gate has a fairly big width of 6mm. This means that a 3D printed part can easily be used to pass through the gate without this printed part being too thin and thereby fragile. Also there is less possibility of collision between the sensor and the sensing object with this big gate. Lastly it has a short circuit protection build into the sensor. [7]

Figure 19 Panasonic PM-series infrared gate sensor

The sensors are placed on the X, Y, Z and R axis of the system (Figure 20).

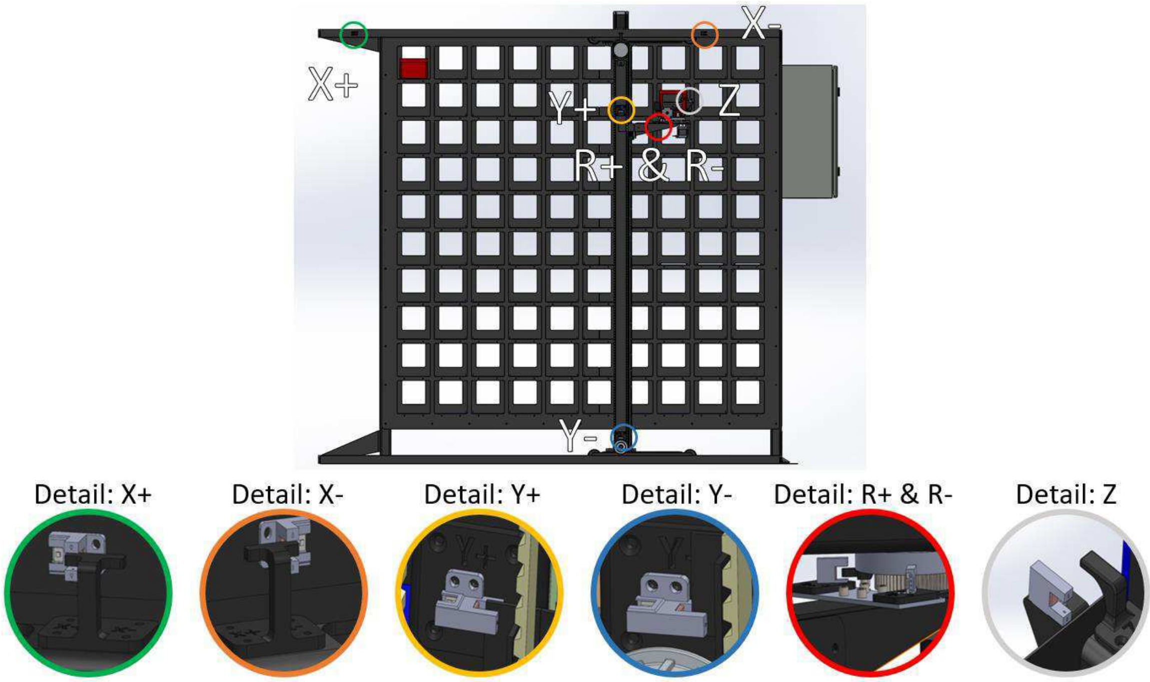


Figure 20 placement of all sensors

3.8 Arm design (SR-4.2, 6.2 and 6.3)

Part of the objective of this project was to improve the arm of the LMI-restore system (see §2.1 for more information). Operation used to be noisy because of the misalignment, and if the trays were situated too far back in the storage cells the arm could not reach them. Table 5 is the initial table which shows the problems in the first column, the possible causes in the second column, and the potential solutions conceived at the start of the project in the third column. Table 6 shows the problems that were encountered when the project began in the first column, the solutions initially considered to solve those problems in the second column, and the implemented solution in the third column. As there were many problems with the arm, replacing it with a completely new design was considered (see appendix 3). Ultimately the original arm design was used. However, some changes were made to eliminate some of the problems it was experiencing.

Problems/required improvements	**Causes**	**Potential solutions**
Slow extension of the arm (Z direction)	The lead screw has a very small pitch. So the stepper has to make a lot of revolutions to make its complete movement	- Replace the lead screw by one with a higher pitch. Possibly even a ball screw to have it run smoother. - Replace the lead screw with a different form of transmission. Possibly a belt. - Place faster motor
Insufficient reach (Z direction)	The depth of the storage cells is higher than the reach of the arm. So it’s possible that the arm can’t reach the Containers.	- Increase the reach (scope) of the leadscrew. (Z direction) - Limit the depth of the storage cells, using bumpers for instance.
High friction (Z direction)	The entire system was design without the coating in mind. So the coating caused misalignment in the system.	- Redesign parts that suffer from the misalignment. - Slightly alter the design to make it adjustable.
Slow rotation arm (R direction)	Low maximum speed of the stepper and/or a high gear ratio.	- Increase motor speed - Decrease gear ratio
Unstable (R direction)	The rotating part of the arm is positioned on a single bearing which isn’t rated against axial loads.	- Use bearings that are rated for axial loads. (tapered roller bearing) - Increase axel length and use multiple bearings.
Noise when operating (R direction)	Friction, Motor noise, etc.	- Lubricate the moving parts. Upgrade motors/drivers. - Replace the lead screw with a ball screw. - Re-align the components

Table 5 problems with arm

Problems/required improvements	Potential solutions	Solution
Slow extension of the arm (Z direction)	- Replace the lead screw by one with a higher pitch. Possibly even a ball screw to have it run smoother. - Replace the lead screw with a different form of transmission. Possibly a belt.	Changed motor settings to make the motor turn faster, to decrease time to take a container.
Insufficient reach (Z direction)	- Increase the reach (scope) of the leadscrew. (Z direction) - Limit the depth of the storage cells, using bumpers for instance.	Replace pick plate with modified plate. This modified pick plate has a longer reach due to an extended horizontal plane(see figure X1 and figure X2).
High friction (Z direction)	- Redesign parts that suffer from the misalignment. - Slightly alter the design to make it adjustable.	Increased the diameter of motor mounting holes. This allows the motor to be moved to a more suitable position. This eliminates the misalignment.
Slow rotation arm (R direction)	- Increase motor speed - Decrease gear ratio	Motor speed has been increased.
Unstable (R direction)	- Use bearings that are rated for axial loads. (tapered roller bearing) - Increase axel length and use multiple bearings.	Increased the hole diameter of the bearing slot to 35mm to create new fitting. Size of the fitting is H5, 35mm diameter roller bearing is inserted. Play has been reduced but not eliminated due to play in axel-bearing connection.
Noise when operating (R direction)	- Lubricate the moving parts. Upgrade motors/drivers. - Replace the lead screw with a ball screw. - Re-align the components	Components have been re-aligned.

Table 6 problems with solutions

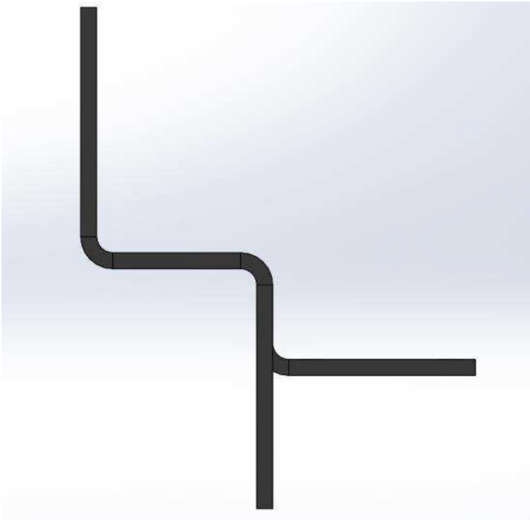


Figure 21 old pickplate

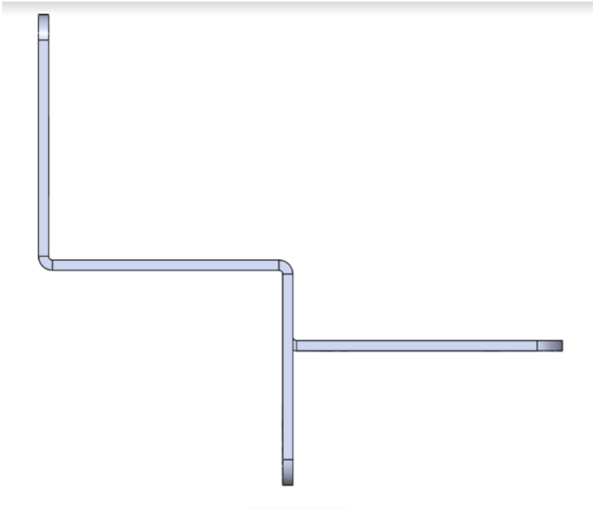


Figure 22 new pickplate

In conclusion, the reach has been increased by replacing the old pickplate (Figure 21) with a new pickplate (Figure 22) which has a longer horizontal piece. The friction in the arm that was allegedly caused by the coating has mostly been eliminated by re-aligning the components. Re-alignment was achieved by increasing the size of the holes of the fixes, this allowed the components in the moving part of the arm to move around as the picking assembly moved along the Z-axis.

The previous team had inserted the old bearing, at 32mm outer diameter, that connected the moving part of the arm to the stationary part of the arm into a fitting that was too large. This caused a lot of play in the arm. In the improved arm a larger bearing, at 35mm outer diameter, was used. The new fitting is approximately 35,010mm (it is an H5 fitting) diameter into which the new 35mm bearing is inserted. This eliminated play in the fitting. Finally, the leadscrew has been cleaned and lubricated to remove another source of friction.

3.9 Electrical schematic (SR-6.4)

The project started with a working electro container. The main components in this container were power supplies, stepper motor drivers, Raspberry pie, Arduino Mega and a DIN rail with terminal blocks. Outside the electro container, the stepper motors and limit switches were connected to the other side of the terminal blocks in the electro container. All these components were wired to each other. However, there was no documentation available of how these components were wired together. In the report of the former group that worked on this project was an electrical diagram, but this diagram was incomplete and unclear. Because of this reason the team decided to create a new, complete and clear electrical diagram.

To create this diagram, the choice for the used software is important. The final choice fell on using EPLAN, because of clarity, ease of use and costs. EPLAN has an educational version that students can use for free.

The main advantage of using EPLAN instead of draw.io or eagle is that EPLAN uses breakpoints with references and has the functionality to add several pages. This creates a clear schematic where you can see directly where the wires are connected to. Figure 23 represents the connection of the stepper motor that drives the X-axis. The four breakpoint on the left have numbers in front of them. (20.3) The number before the dot (20) refers to the page were the breakpoint comes from and the number after the dot (3) refers to the column the breakpoint is on that page. If you want to know where the wire will continue, you go to page 20 and look for that same breakpoint name in column 3.

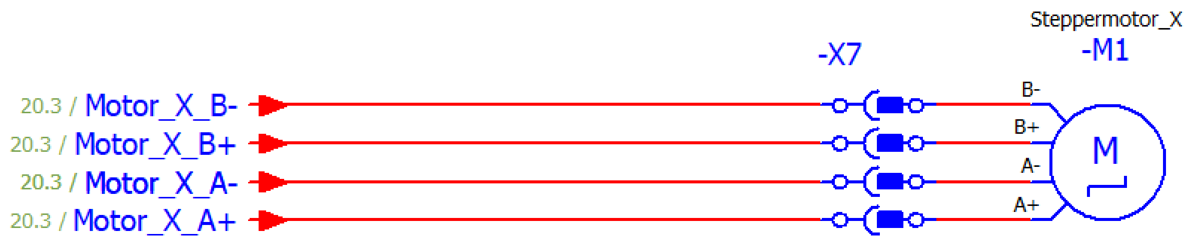


Figure 23 example of using breakpoint in EPLAN

The complete electrical schematic has been designed in EPLAN and can be found at Appendix 4. The advantages of this complete diagram are that electrical adjustments can be implemented faster and new project groups that are going to work on this project can understand the working principle of the system faster.

To control multiple drivers at the same time a STM32 Nucleo-144 (Figure 24) was used because the processing power of the Arduino Mega was not enough. Only the signal for the light switches had to be shifted from 5V. These signals are now shifted to 2.5V by a voltage divider.

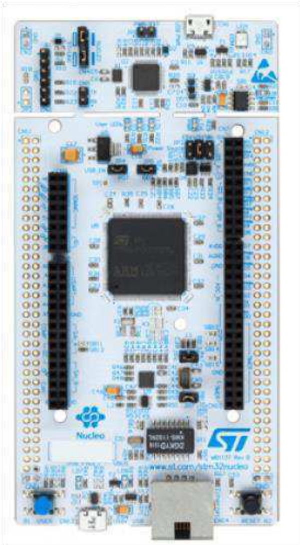


Figure 24 Nucleo-144 STM32

3.10 CAD database (SR-6.5)

The LMI-restore project contains a lot of CAD files. Finding a part, Assembly or drawing can be hard if they are not named and stored in a structured way. In the beginning of the project all the CAD files from the previous project group had names like: “side plate”, “tray”, etc.

It might also not be clear what the latest version of a part or drawing is.

To establish a clearer overview of all the Solidworks parts, assemblies, drawings, DXF and other 3D files an Excel database was created (Appendix 5). This database uses drawing numbers to indicate a specific part or assembly. By filling in properties of the part or assembly, such as the drawing format and revision number (see Figure 25) the drawing name is generated. This number is then assigned to the specific part or assembly. The meaning of the generated number is described with an example in Figure 26.

A	B	C	D	E	F	G	H	I	J	K
Tekeningnummer:	Omschrijving:	K/M	CAD Model:	Drawi	3D machining file (Bijv. STL of STEP):	Custom:	Hoogste revisie:	Samenstelling	Link:	Bestandsnaam:
00122	Eindplaat rails (4mm dik)	M	JA	DXF	NEE	JA	00	00001		T00122-DXF-R00
00123	y-as staaf Ø 16mm	M	JA	A3	NEE	JA	01	00001		T00123-0A3-R01

Figure 25 example of generated names in the CAD database

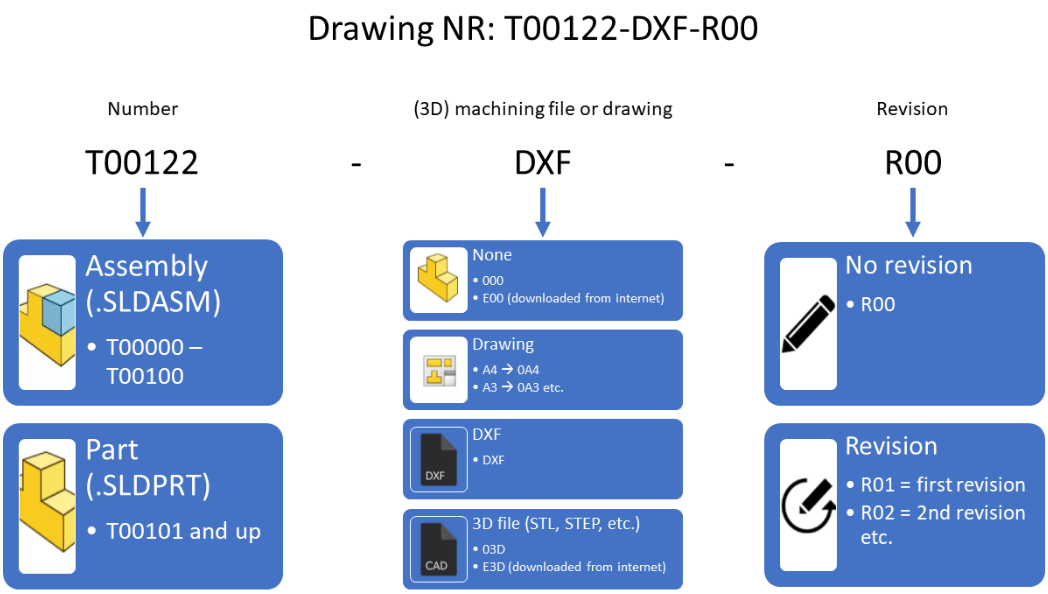


Figure 26 layout database

Using the database has made the following improvements:

- Clear overview of the latest version (=revision) of a part or assembly.
- It can be quickly determined if a part has a drawing, DXF or 3D file.
- For parts that have been bought online, the supplier has been listed in the description and the link to the website has been added. This makes replacing the part easy.
- Assembly files are listed at the top of the directory (because they are named somewhere between T00000 – T00100, they will be on top of the page).

3.11 Motion planner

To smoothly move the stepper motors of the system, a motion planner is used.

The motion planner generates a 2nd order motion profile, consisting of 3 phases, as can be seen in Figure 27. It starts with a phase where the velocity increases linearly towards the target velocity. When this velocity is reached, the profile has a constant velocity, with the position increasing towards the setpoint. At the end of the profile, the velocity is decreased again to stand still at the target position.

A second order setpoint is chosen to have a smooth motion without making the profile more complex than needed. When a 1st order setpoint is used, the motors don't have time to accelerate, and will skip steps at the start of the motion, causing a loss in position accuracy. With a 3rd order setpoint, the calculations for generating the setpoint will be more complex, without increasing the performance significantly.

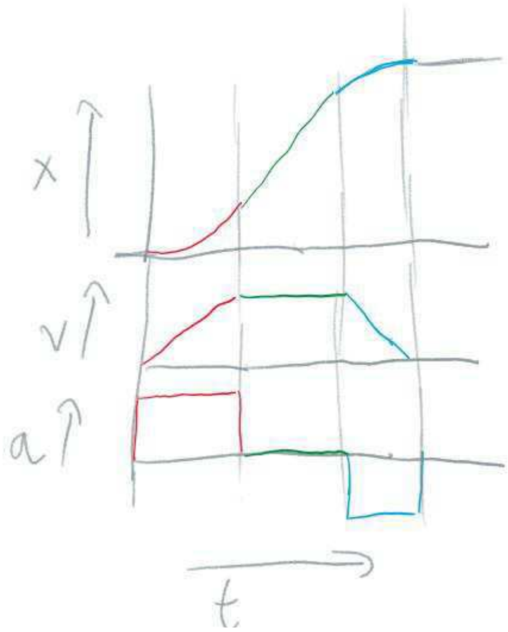


Figure 27 Phases of the motion profile

The timing for the motion planner is done by a single, 32-bit timer running at 1MHz. This timer has 4 compare channels active that generate an interrupt when the timer counter reaches a value that is stored in the compare register. Each axis has its own compare channel assigned, meaning that all axes can run independently of each other. When an interrupt is fired, the interrupt service routine checks for which axis the interrupt request is and executes the stepping method belonging to that specific axis.

Phase 1

During the first phase of the motion profile (the red part in Figure 28, the acceleration is a constant positive value. This results in the velocity increasing linearly until the target velocity is reached.

$$v = a * t \left[\frac{st}{s^2} * \mu s\right] \rightarrow \left[\frac{\mu st}{s}\right]$$

$$v = \frac{a * t}{10^6} \rightarrow \left[\frac{st}{s}\right]$$

$$\Delta t = \frac{1}{v}\left[\frac{s}{st}\right]$$

$$\Delta t = \frac{10^6}{v}\left[\frac{\mu s}{st}\right]$$

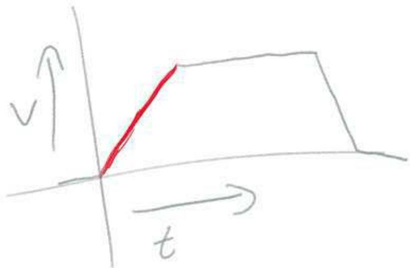


Figure 28 Motion planner phase 1

Because, initially, $t = 0$, $v = 0$ and $\Delta t = 1/v = 1/0 = \infty$. To be able to set the time for the next interrupt, the velocity at the next step ($x = 1$) is taken as initial velocity.

$$x(t) = \frac{1}{2}at^2 \ [st]$$

$$x(t_0) = \frac{1}{2}at_0^2 = 1 \ [st]$$

$$t_0 = \sqrt{\frac{2}{a}} \ [s] = \sqrt{\frac{2 * 10^6}{a}} \ [\mu s]$$

This formula is only dependent on the acceleration, meaning that t_0 only needs to be recalculated when the acceleration is changed.

Phase 2

In phase 2 (green part in Figure 29), the velocity of the movement is constant, so the Δt is the same between each pulse. With every step, it only needs to set the timer compare value to a new value at each step pulse.

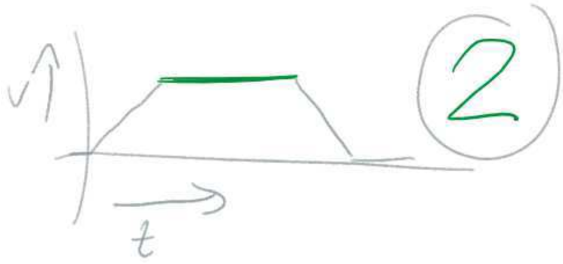


Figure 29 Motion planner phase 2

Phase 3

At phase 3, the motion planner decelerates with at a constant rate until it is standing still (the blue part in Figure 30). Calculating Δt works the same as in phase 1, but with the time going backward.

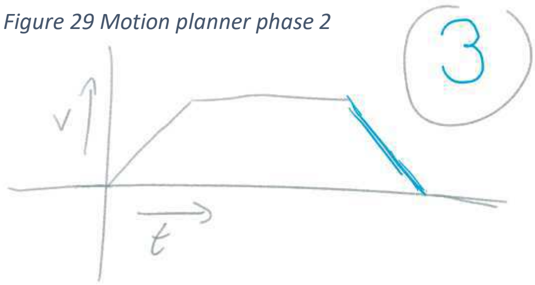


Figure 30 Motion planner phase 3

Real-time controller

To generate The Atmel chip on the Arduino mega was found to not be powerful enough for controlling the system. It could only generate pulses at a maximum of 4kHz. This means that the only way to make the axes move faster, is to set the stepper drivers to a lower microstepping factor. This, however, results in the motor moving a bigger angle with each step, resulting in a jerkier movement.

To solve this, the real-time controller that generates the motion profiles of the motors has been changed from an Arduino Mega to a NUCLEO-F429ZI. The main advantages of this microcontroller are an increased core clock speed and an upgrade in the architecture. The STM32F429 (which the Nucleo that is used is based on) runs at a frequency of 168MHz in the current configuration. This is about 10x faster compared to the 16MHz of the Arduino. The Nucleo is based on a 32-bit architecture. This means that it can calculate bigger numbers without using multiple instructions.

With the higher performance of the new real-time controller, multiple axes can now be moved simultaneously, and with a higher velocity while keeping the smoothness of microstepping.

3.12 Software communication

The software communication is a very important aspect of the storage system. Communication is required to control the movement of the robotic arm and process the user input from the GUI.

The main code of the storage unit is written on a Raspberry Pi. Because python is a common language used on a Raspberry Pi with loads of support compared to other languages python seemed to be the code of choice.

Architecture, As seen in Figure 31, The server main code is connected to the client which is also written in python and connects through each other by program “0MQ” the total flow chart can be found in Appendix 6 . This is a tested working extension for running multiple python scripts and have them communicate between each other by IP. The advantage of this is that there are several commands which the server can recognize so when adding a secondary client (a robot asking for a request for instance), no changes have to be made to the main code. A further guide on how to use 0MQ can be found online. [8]

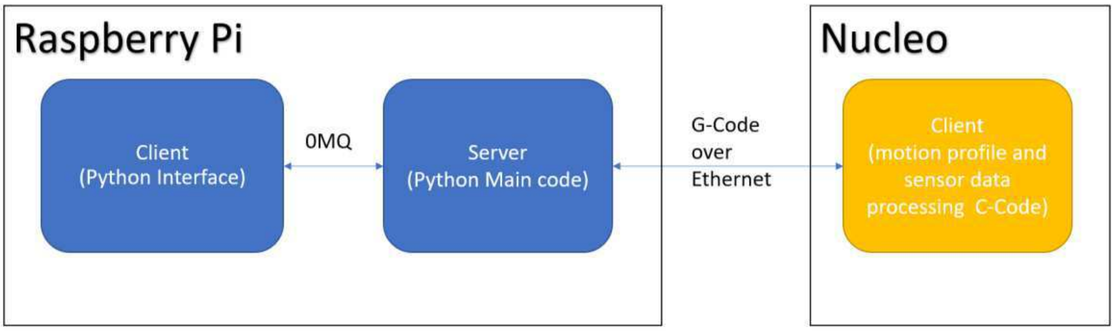


Figure 31: Architecture of the programming code

Commands, Several commands can be sent to the server to ask for the collection or positioning of a container. These commands can be recognized by comparing text as seen in Figure 32. [9] This code defines the part that needs to be recognized plus a number and compares that to the message. If the message contains the part to be recognized, the function request_container(number) will be executed.

```
Import re

Re_requestcontainer = re.compile(r’Request Container (\d+)’)

If re_requestcontainer.match(message)
        container = re_requestcontainer(message).group(1)
        request_container(container)
```

Figure 32: recognition codes

4. Testing

For validation of functions from the system, the functions of the system need to be predefined and tested when the system is operational. These functions can be anything from moving a certain axis till human interaction with the storage. The layout of the test documents was used from the source www.cdc.gov. [10] The documents contain all the information about the testing conditions, risks, approach, environment, entry conditions and when it will pass/fail. When the test was passed, there are some deliverables for the validation of the test. These have to be submitted to guarantee the validity of the test.

Not all the tests were finished when this report was handed in. The only test that is finished is the human interaction plate. This one has been done when the plate was placed on the storage. This test was done with two students who could each try to take a bolt from the storage, and try to take the whole om the storage.

The test documents can be found in appendix 7.

5. Conclusion and recommendations

The conclusion for the project comes down to a couple of points. First the part of human interaction with the smart storage system is solved by restricting the way humans can interact to only picking the contents from the containers instead of the whole container. This way the system knows at all times where all the containers are in the storage system. For the queuing the decision was made to use three cells as the queue. These spots are able to weight the container that is placed in the queue by using the build in load cell. By weighing the containers the system can determine if the container is almost empty and can send out a message to fill the container if needed. For the part of recognizing the contents of the container the arm that picks and places the containers has a build-in barcode scanner. All the containers are provided with a unique barcode which is coupled to the system's database. We took the safety into account by making sure no humans have to be on the side where the arm operates. This is made possible by using one side of the storage system for the arm and the other side for human interaction. Another safety feature is the new motor brake on the Y-axis which make sure when the emergency button is used or in case of a power outage the arm on the Y-axis is hold in place instead of falling down and possible braking some parts.

As for improvements to the system, first the micro switches on all the axis were replaced by infrared gate sensor to prevent them breaking in case the arm would not stop on time. Secondly the arm wasn't able to reach the container if it was placed all the way in the back of the cell so the arm has been made longer so it can reach the container even if it is placed at the back of the cell. Another thing which has been improved is that the system can move more than one motor at the same time. This way we can save a lot of time with just software without speeding up the motors. At last we improved on the documentation by first reorganizing and rewiring the electrical container and making a complete electrical schematic. Second an excel database is made to have a clear overview of all the CAD-drawings made for the entire system.

There are a couple of things we, as project group, can recommend to improve the storage system even further. We recommend to have the system make some sort of routine scan of all the containers in the system at the beginning of a workday to make sure the database corresponds with the actual system. It shouldn't be possible to move containers without the system knowing about it but the routine scan can be an extra check. Our last recommendation is about what kind of things are put in the system. If the client always has the same items in the storage system it is possible to label the side where humans can get parts to make searching for the correct part easier. It is also possible that the client changes the parts in the system on a regular bases. If so, we recommend a GUI where the worker can easily search the database for the part he or she needs and can see the current position of that part in the system.

References

[1] S. Achaoui, R. van Hoof, N. van Kasteren, N. van de Gevel, T. Ummenthum and R. Looijmans, "Technical report Smart Storage," Eindhoven, 2018.

[2] M. v. Osch, "Let's move it_description_Restore," [Online]. Available: https://connect.fontys.nl/instituten/FHEngAut/opleidingen/mech/S7M/_layouts/15/WopiFrame.aspx?sourcedoc=/instituten/FHEngAut/opleidingen/mech/S7M/Documents/Projects%201819/8%20-%20let%27s%20move%20it_description_Restore.pdf&action=default. [Accessed 20 September 2018].

[3] C. Remmers and M. Laarhoven van, "Reader System engineering 2 (SYE2) 2015 2016," Eindhoven, Fontys engineering, 2016, pp. 44-48.

[4] Waveshare, "Barcode scanner module," [Online]. Available: https://www.waveshare.com/wiki/Barcode_Scanner_Module. [Accessed 8 October 2018].

[5] RF wireless world, "Advantages of Capacitive sensor | Disadvantages of Capacitive sensor," [Online]. Available: http://www.rfwireless-world.com/Terminology/Advantages-and-disadvantages-of-Capacitive-Sensor.html. [Accessed 26 January 2019].

[6] RF wireless world, "Advantages of Inductive sensor | Disadvantages of Inductive sensor," [Online]. Available: http://www.rfwireless-world.com/Terminology/Advantages-and-disadvantages-of-Inductive-Sensor.html. [Accessed 26 January 2019].

[7] Channel Panasonic - Official, "Youtube," Panasonic, 18 January 2016. [Online]. Available: https://www.youtube.com/watch?v=MvqT8lfSyUo. [Accessed 25 January 2019].

[8] ZeroMQ, "Learning 0MQ," ZeroMQ, 2018. [Online]. Available: https://learning-0mq-with-pyzmq.readthedocs.io/en/latest/. [Accessed 12 1 2019].

[9] A. K. <amk@amk.ca>, "Regular Expression HOWTO," Python software foundation, 2018. [Online]. Available: https://docs.python.org/2/howto/regex.html. [Accessed 25 1 2019].

[10] U.S. Department of Health & Human Services, 12 2018. [Online]. Available: https://www2a.cdc.gov/cdcup/library/templates/CDC_UP_Test_Plan_Template.doc.

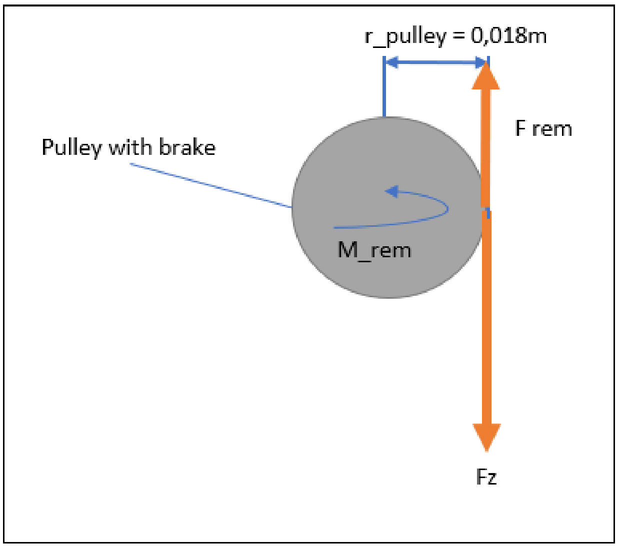
Appendix 1: Morphological chart

		lage kosten	lage complexiteit	hoge snelheid	hoge robuustheid	hoge flexibiliteit	Totaal
Queing	Turntable external	3	4	6	4	4	21
	Second storage unit	5	5	2	5	6	23
	Designated places in storage	6	6	4	6	5	27
	Elevator	1	2	3	2	1	9
	Turntable on arm	2	3	5	3	3	16
	Stack	4	1	1	1	2	9
Detecting empty boxes	Vision	2	3	4	3	3	15
	Weight sensing in all cells	3	1	6	4	5	19
	Weight sensing in queue	5	5	5	6	2	23
	Weight sensing in arm	6	2	1	5	6	20
	Conductance	4	4	2	1	4	15
	Visual inspection by human operator	1	6	3	2	1	13
Detecting content of box	Vision	2	1	2	2	1	8
	QR/barcode	4	4	3	4	3	18
	RFID/NFC	3	3	4	3	4	17
	Visual inspection by human operator	1	2	1	1	2	7
Detecting presence of box	Vision	3	2	7	2	1	15
	QR/barcode	6	5	4	7	3	25
	RFID/NFC	4	4	3	6	4	21
	Visual inspection by human operator	1	1	1	1	2	6
	Ultrasonic	7	7	5	3	5	27
	Laser sensing	5	6	6	4	7	28
	Feeler probe	2	3	2	5	6	18
	Database	8	8	8	8	8	40
Safety (moving parts)	Warning sticker (ground tape)	4	4	0	1	4	13
	Enclousure entire system	2	3	0	4	1	10
	Covers around moving parts	3	2	0	3	2	10
	laser curtain	1	1	0	2	3	7
Safety (platform falls down)	Damper	5	5	1	3	0	14
	Motor brake	4	4	5	5	0	18
	Counter mass	3	3	3	2	0	11
	Centrifugal brake	2	1	2	1	0	6
	Rail stopper	1	2	4	4	0	11

Appendix 2: Motor brake calculation

Rem berekenening		
UITKOMSTEN		
Grootheid	Waarde	Eenheid
Fz	88,29	N
F rem	111,11	N
INVULVAKKEN		
Grootheid	Waarde	Eenheid
m_arm	9	kg
r_pulley	0,018	m
r_pulley	18	mm
M_rem (Rated)	2	Nm

Veiligheidsfactor: 1,3



Appendix 3: New arm design

During the process of improving the arm, various courses of action were taken into consideration. The first course of action was to solve the problems individually by redesigning or editing specific components of the original design. The second course of action was to design a completely new arm from scratch. It was, however, deemed that this approach would have taken considerably more time compared to the first one, as all problems with the original design can be addressed at once. Additionally, the shortcomings of the original design wouldn't constrain de redesign choices when redesigning the entire arm.

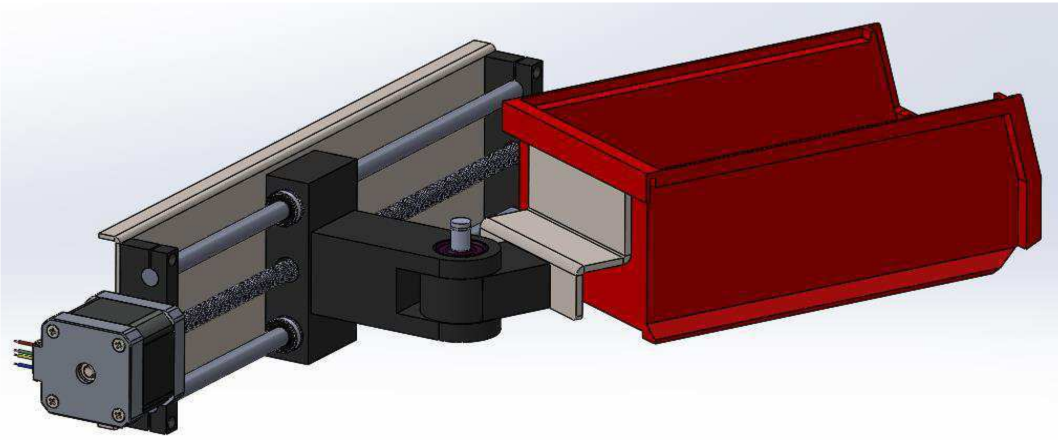


Figure 33 new arm design

Figure 33 displays the first take on a design for a completely new arm. The main focus for the redesign was to swap the position of the R- and Z-axis, because the bearing in the R-axis of the original design couldn't take the weight of the Z-axis. The following improvements would be realized using a final version of the redesigned arm.

- The redesign would have enough reach to pick-up the containers.
- The introduction of a second bearing for the R-axis would reduce the play.
- Fixing the Z-axis to the carriage of the Y-axis reduces the load on the actuator of the R-axis, thus increases the lifetime of said actuator.
- The redesign was designed coating in mind so it wouldn't introduce any friction and other complications afterwards.
- The redesign has a lower mass compared to the original design.

Despite these potential improvements, it was decided to not go with a redesign. The reason for this was the additional time necessary to realize a completely new design. As such, the current arm was improved.

Appendix 4: EPLAN Electrical Schematic

0	1	2	3	4	5	6	7	8	9

F26_001

EPLAN Software & Service
GmbH & Co. KG

An der alten Ziegelei 2
40789 Monheim am Rhein
Tel +49 (0)2173 - 39 64 - 0

Bedrijf / klant	
Projectbeschrijving	Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats
Tekeningnummer	IEC_tpl001
Opdrachtgever	EPLAN
Fabrikant (bedrijf)	EPLAN Software & Service GmbH & Co. KG
Pad	EPLAN voorbeeldproject
Projectnaam	PRJ7
Fabrikaat	
Type	
Installatieplaats	
Projectleider	
Deelbijzonderheid	

Gemaakt op 28-11-2018

Bewerkt op 25-1-2019 Van (afkorting) NIEKV Aantal pagina's 13

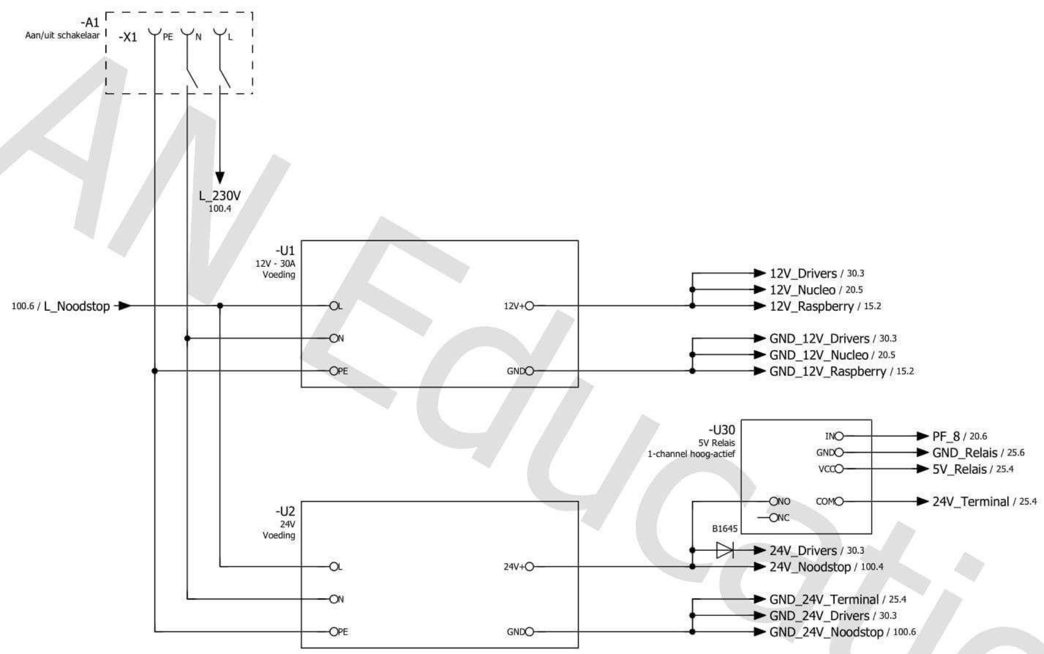
2

			Datum	23-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Titelblad / voorblad	= Besturingskast
			Bew	NIEKV				+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats		IEC_tpl001	Blad 1
Wijziging	Datum	Naam	Oorspr		Vervanging van / Vervangen door			Pagina 1 / 13

Inhoudsopgave

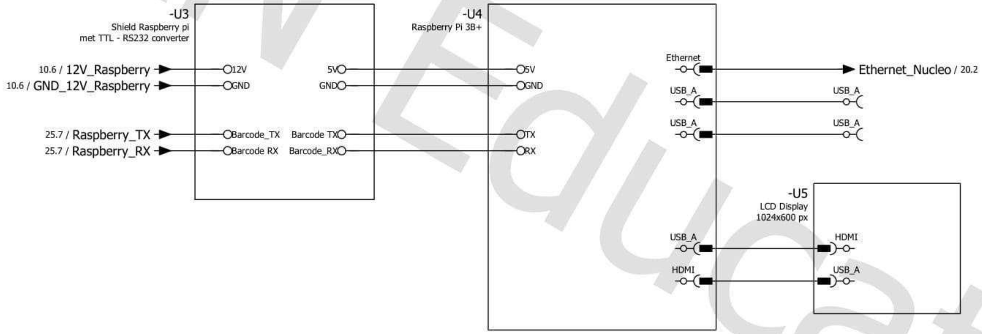
Kolom X: een automatisch gegenereerde pagina is handmatig bewerkt

F06_001



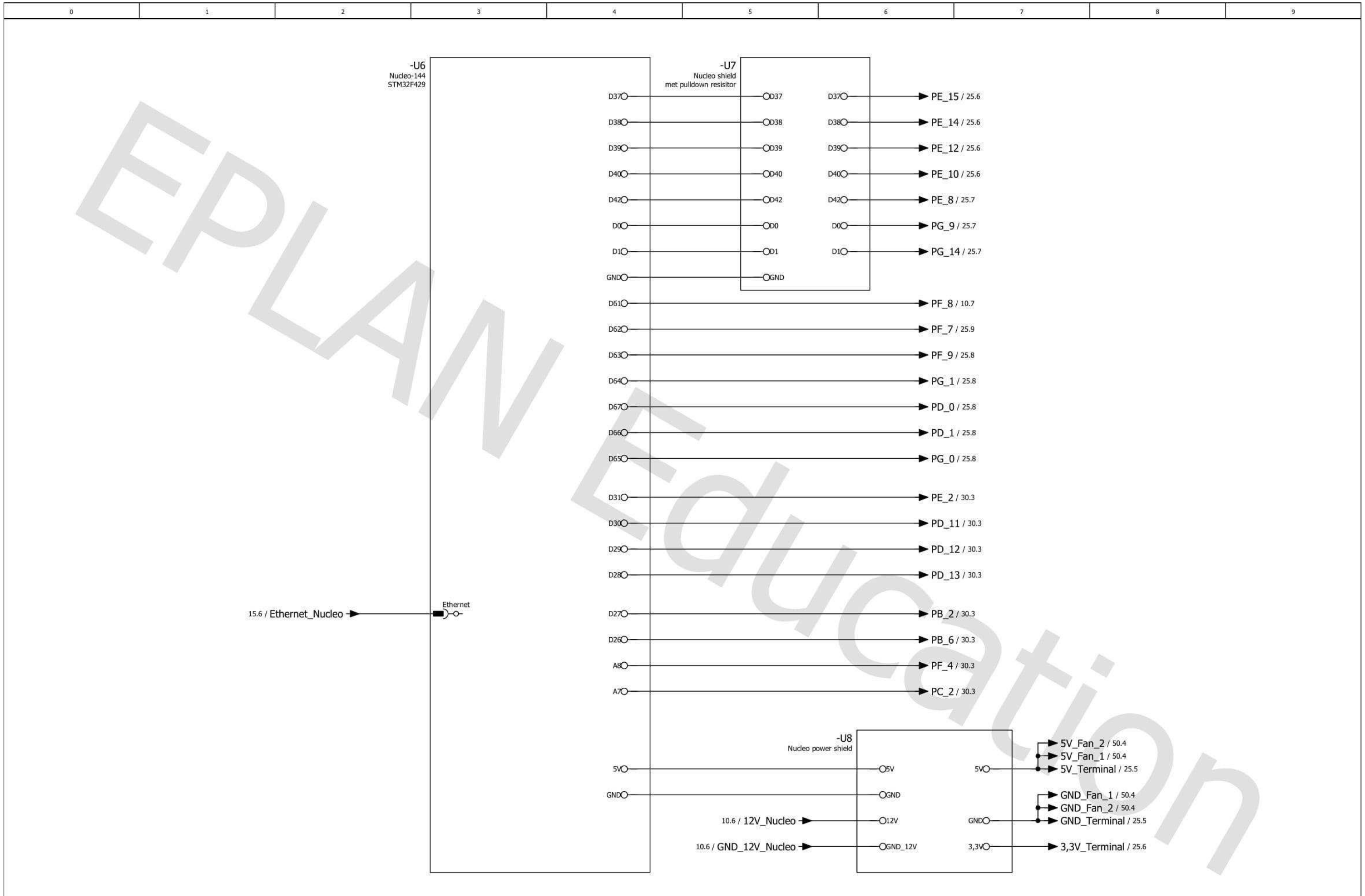
			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Voeding		= Besturingskast
			Bew	NIEKV					+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats			IEC_tpl001	Blad 10
Wijziging	Datum	Naam	Oorspr		Vervanging van / Vervangen door				Pagina 3 / 13

0	1	2	3	4	5	6	7	8	9



10 20

			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Raspberry pi 3B+		= Besturingskast
			Bew	NIEKV					+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats			IEC_tpl001	Blad 15
Wijziging	Datum	Naam	Oorspr		Vervanging van	Vervangen door			Pagina 4 / 13



15 25

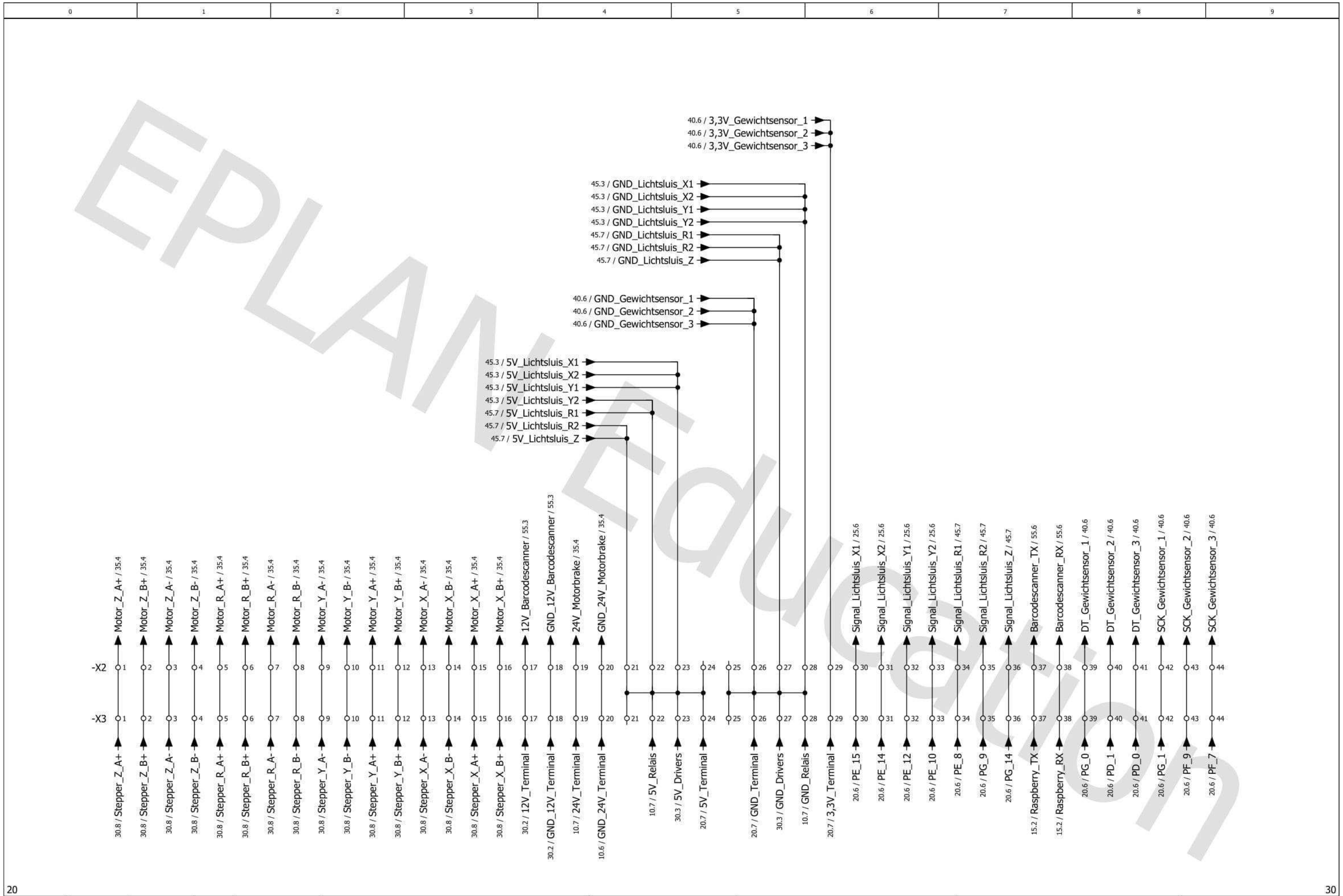
			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG
			Bew	NIEKV	Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats	
			Gecontr			
Wijziging	Datum	Naam	Oorspr		Vervanging van / Vervangen door	

Nucleo-144

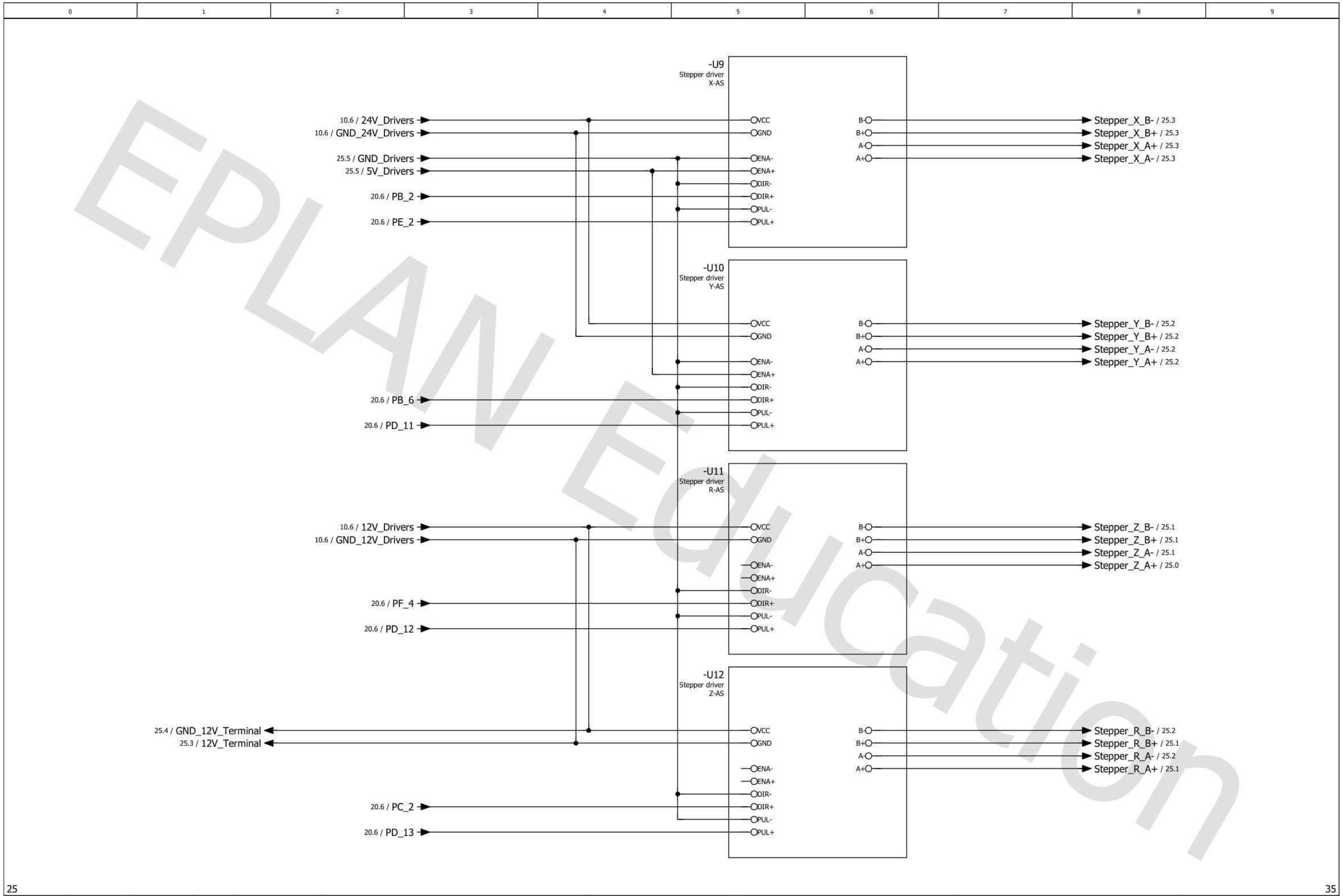
= Besturingskast
+ 10

IEC_tpl001

Blad 20

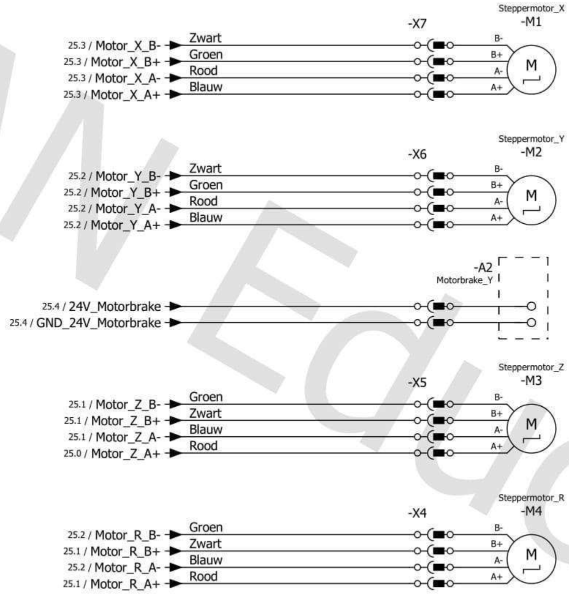


			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Terminals	= Besturingskast
			Bew	NIEKV				+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats		IEC_tpl001	Blad 25
Wijziging	Datum	Naam	Oorspr		Vervanging van	Vervangen door		Pagina 6 / 13



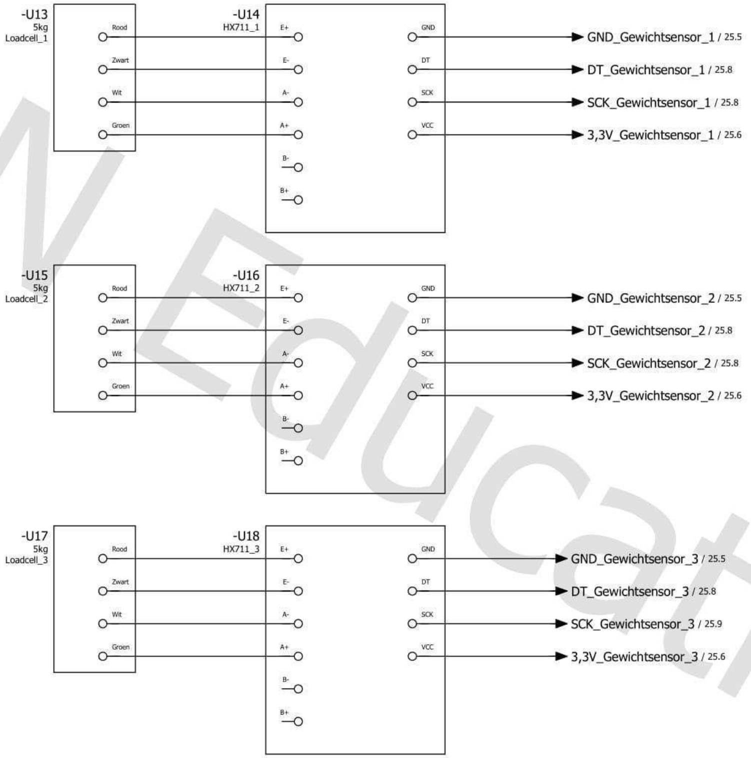
			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Drivers		= Besturingskast
			Bew	NIEKV					+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats			IEC_tpl001	Blad 30
Wijziging	Datum	Naam	Oorspr		Vervanging van / Vervangen door				Pagina 7 / 13

0	1	2	3	4	5	6	7	8	9



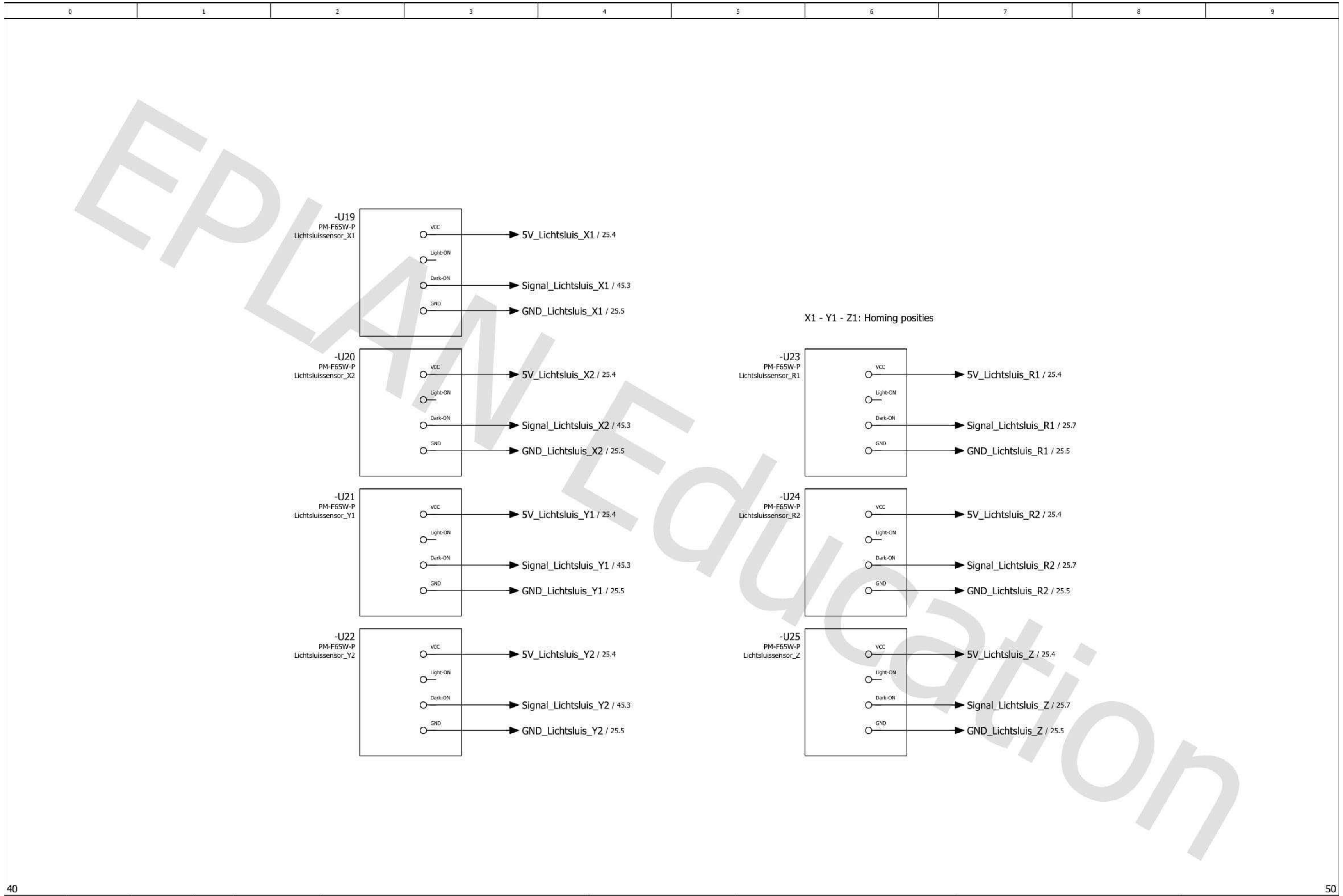
30 40

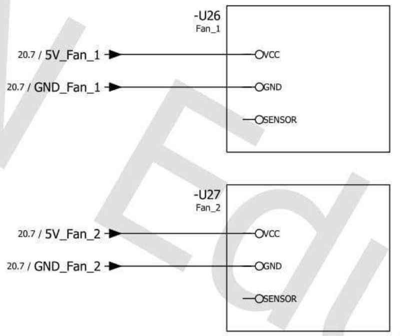
			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Stepper motoren	= Besturingskast
			Bew	NIEKV				+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats		IEC_tpl001	Blad 35
Wijziging	Datum	Naam	Oorspr		Vervanging van	Vervangen door		Pagina 8 / 13



35 45

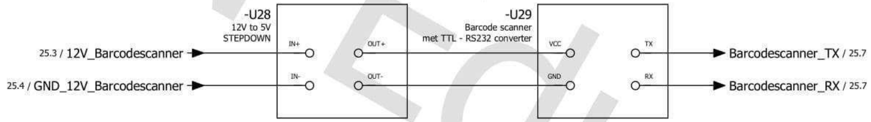
			Datum	23-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Gewicht sensoren	= Besturingskast
			Bew	NIEKV				+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats		IEC_tpl001	Blad 40
Wijziging	Datum	Naam	Oorspr		Vervanging van	Vervangen door		Pagina 9 / 13


0
1
2
3
4
5
6
7
8
9
-U19
PM-F65W-P
Lichtsluissensor_X1
VCC
Light-ON
Dark-ON
GND
5V_Lichtsluis_X1 / 25.4
Signal_Lichtsluis_X1 / 45.3
GND_Lichtsluis_X1 / 25.5
-U20
PM-F65W-P
Lichtsluissensor_X2
5V_Lichtsluis_X2 / 25.4
Signal_Lichtsluis_X2 / 45.3
GND_Lichtsluis_X2 / 25.5
-U21
PM-F65W-P
Lichtsluissensor_Y1
5V_Lichtsluis_Y1 / 25.4
Signal_Lichtsluis_Y1 / 45.3
GND_Lichtsluis_Y1 / 25.5
-U22
PM-F65W-P
Lichtsluissensor_Y2
5V_Lichtsluis_Y2 / 25.4
Signal_Lichtsluis_Y2 / 45.3
GND_Lichtsluis_Y2 / 25.5
X1 - Y1 - Z1: Homing posities
-U23
PM-F65W-P
Lichtsluissensor_R1
5V_Lichtsluis_R1 / 25.4
Signal_Lichtsluis_R1 / 25.7
GND_Lichtsluis_R1 / 25.5
-U24
PM-F65W-P
Lichtsluissensor_R2
5V_Lichtsluis_R2 / 25.4
Signal_Lichtsluis_R2 / 25.7
GND_Lichtsluis_R2 / 25.5
-U25
PM-F65W-P
Lichtsluissensor_Z
5V_Lichtsluis_Z / 25.4
Signal_Lichtsluis_Z / 25.7
GND_Lichtsluis_Z / 25.5
40
50



			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Fans	= Besturingskast
			Bew	NIEKV				+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats		IEC_tpl001	Blad 50
Wijziging	Datum	Naam	Oorspr		Vervanging van	Vervangen door		Pagina 11 / 13

0	1	2	3	4	5	6	7	8	9

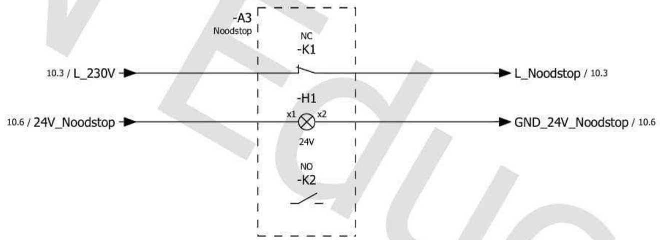


50

100

			Datum	23-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Barcode scanner		= Besturingskast
			Bew	NIEKV					+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats			IEC_tpl001	Blad 55
Wijziging	Datum	Naam	Oorspr		Vervanging van	Vervangen door			Pagina 12 / 13

0	1	2	3	4	5	6	7	8	9



55

			Datum	25-1-2019	EPLAN	EPLAN Software & Service GmbH & Co. KG	Noodstop	= Besturingskast
			Bew	NIEKV				+ 10
			Gecontr		Projectsjabloon met coderingsstructuur volgens IEC-Norm: paginastructuur met groep en inbouwplaats			IEC_tpl001 / Blad 100
Wijziging	Datum	Naam	Oorspr		Vervanging van / Vervangen door			Pagina 13 / 13

Appendix 5: CAD Database

Txx bestand

Txx nummer Uitleg:

T(5 cijferig Tekeningnummer)-(3 cijferig typenummer)-R(revisienummer)

Voorbeelden:
T00011-0A3-R00 (revisie: 0,Bevat een A3 werktekening) -> Model: JA, Drawing:(type), stl of STEP file: NEE; Custom: JA
T00012-000-R00 (revisie: 0,Bevat geen werktekening,) -> Model: JA, Drawing: NEE, stl of STEP file: NEE; Custom: JA
T00013-000-R01 (revisie: 1,Bevat geen werktekening,) -> Model: JA, Drawing: NEE, stl of STEP file: NEE; Custom: JA
T00014-E00-R00 (revisie: 0,Bevat geen werktekening,) -> Model: JA, Drawing: NEE, stl of STEP file: NEE; Custom: NEE
T00015-03D-R03 (revisie: 3,Wordt bijv. ge-3-dprint/heeft een stl of STEP file,) -> Model: JA, Drawing: NEE, stl of STEP file: JA; Custom: NEE
T00016-E3D-R00 (revisie: 0,Wordt bijv. ge-3-dprint/heeft een stl of STEP file,) -> Model: NEE, Drawing: NEE, stl of STEP file: JA; Custom: NEE
T00014-DXF-R00 (revisie: 0,Bevat een DXF file,) -> Model: JA, Drawing: DXF, stl of STEP file: NEE; Custom: JA of NEE

LET OP! enige combinatie die niet mogelijk is een CAD file met drawing en 3D machining file!!!

K/M

K= kooponderdeel
M = maakonderdeel
K/M= koop/maakonderdeel

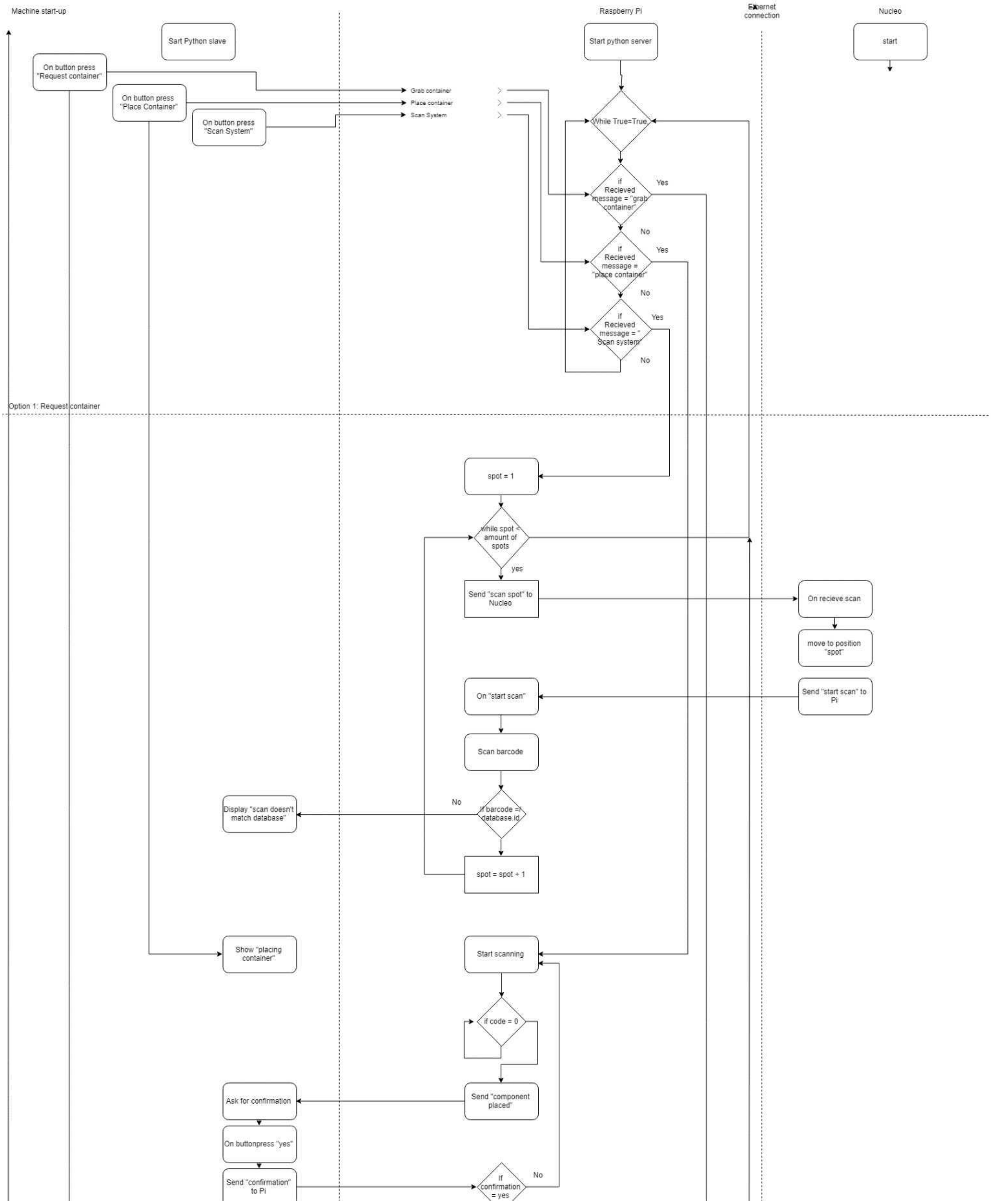
Tekeningnummer:	Omschrijving:	K/M	CAD Model:	Drawir	3D machining file (Bijv. STL of STEP):	Custom:	Hoogste revisie:	Samenstelling:	Link:	Bestandsnaam:
00000	LMI Re-store system hoofdsamenstelling	K/M	JA	NEE	NEE	JA	01			T00000-000-R01
00001	XY-carriage samenstelling	K/M	JA	NEE	NEE	JA	01			T00001-000-R01
00002	Magazijn samenstelling	K/M	JA	NEE	NEE	JA	00			T00002-000-R00
00003	Arm samenstelling	K/M	JA	NEE	NEE	JA	01	00001		T00003-000-R01
00004	Weegschaal samenstelling	K/M	JA	NEE	NEE	JA	00	00002		T00004-000-R00
00005										
00006										

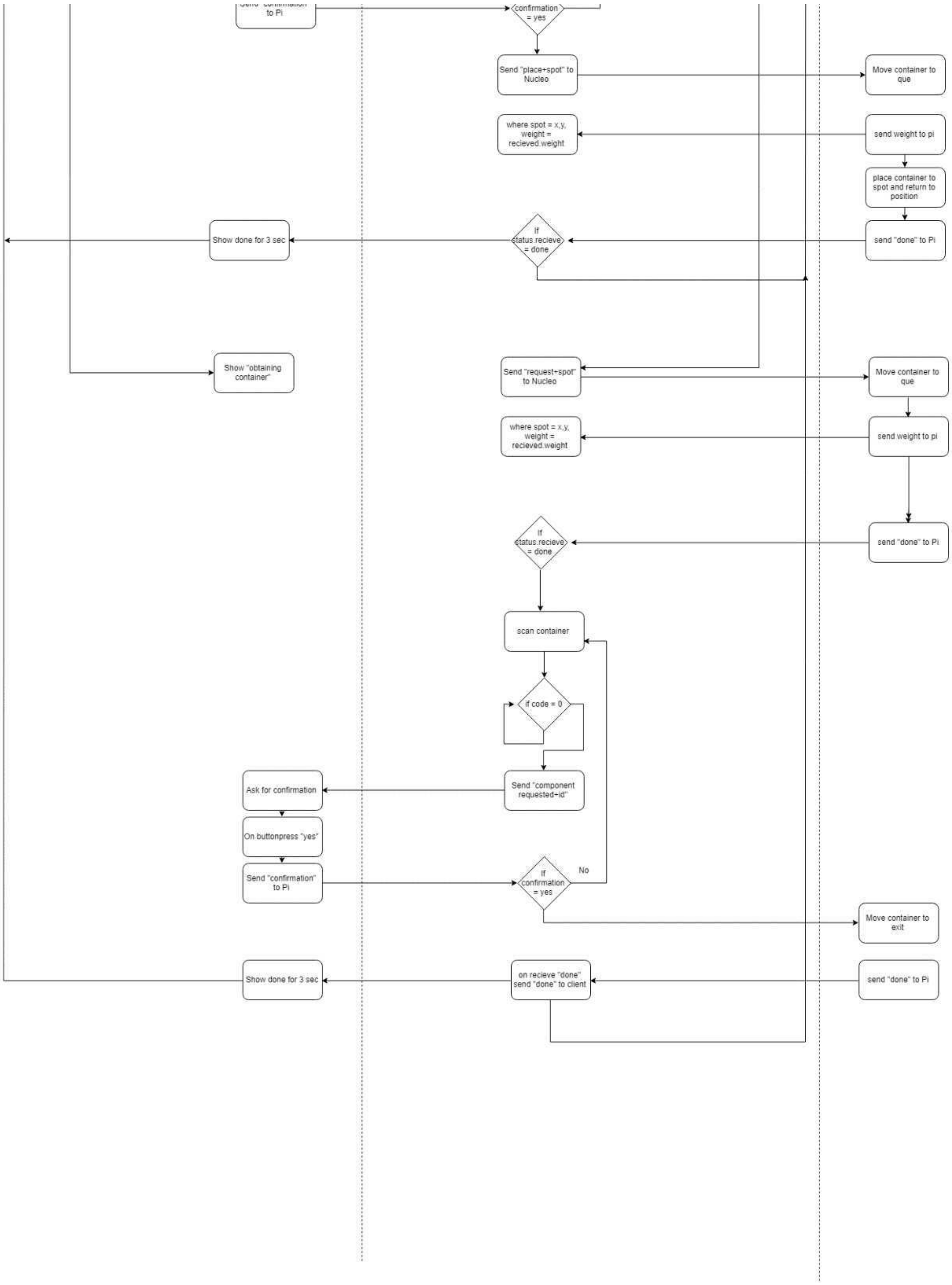
Tekeningnummer:	Omschrijving:	K/M	CAD Model:	Drawir	3D machining file (Bijv. STL of STEP):	Custom:	Hoogste revisie:	Samenstelling:	Link:	Bestandsnaam:
00100	Flenslager binnenØ 15mm	K	JA	NEE	NEE	NEE	00	00001		T00100-E00-R00
00101	Reely Kogellager radiaal Chroomstaal Binnen binnenØ 8mm, buitenØ 15mm(Conrad	K	JA	NEE	NEE	NEE	00	00001	Link	T00101-E00-R00
00102	Timing pulley T10 Ø 15mm	K	JA	NEE	NEE	NEE	00	00001	Link	T00102-E00-R00
00103	Blickle 346601 wiel binnenØ 6mm, BuitenØ 50 mm(Conrad artnr: 826519)	K	JA	NEE	NEE	NEE	00	00001	Link	T00103-E00-R00
00104	Afstandsbus 20x12x25mm	K	JA	NEE	NEE	NEE	00	00001		T00104-E00-R00
00105	Afstandsbus klein	K	JA	NEE	NEE	NEE	00	00003		T00105-E00-R00
00106	Stepper Motor DCNC-NEMA23-2.5Nm(RepRapworld.nl artnr: 8719345005800)	K	JA	NEE	NEE	NEE	00	00001	Link	T00106-E00-R00
00107	Stepper Motor DCNC-NEMA17-0.5Nm(RepRapworld.nl arnr: 8719345003981)	K	JA	NEE	NEE	NEE	00	00003	Link	T00107-E00-R00
00108	Carriage HG15	K	JA	NEE	NEE	NEE	00	00003		T00108-E00-R00
00109	Rail HGR15R l=1530mm	K	JA	NEE	NEE	NEE	00	00001		T00109-E00-R00
00110	Rail HGR15R l=270mm	K	JA	NEE	NEE	NEE	00	00003		T00110-E00-R00
00111	Draaispindel l=240mm	K	JA	NEE	NEE	NEE	00	00003		T00111-000-R00
00112	Rechthoekig buisprofiel 50x30x2,6x1540mm (hoogte x breedte x dikte x lengte)	M	JA	NEE	NEE	JA	00	00001		T00112-000-R00
00113	Vierkant buisprofiel 60x60x3,2x1600mm (hoogte x breedte x dikte x lengte)	M	JA	NEE	NEE	JA	00	00002		T00113-000-R00
00114	Voorplaat (2mm dik)	M	JA	DXF	NEE	JA	00	00002		T00114-DXF-R00
00115	Achterplaat (2mm dik)	M	JA	DXF	NEE	JA	01	00002		T00115-DXF-R01
00116	La magazijn (1mm dik)	M	JA	A4	NEE	JA	00	00002		T00116-0A4-R00
00117	u-profiel 30x50x3	M	JA	NEE	NEE	JA	00	00001		T00117-000-R00
00118		M	JA	NEE	NEE	JA	00	00001		
00119	Strip wielbevestiging (3mm dik)	M	JA	DXF	NEE	JA	00	00001		T00119-DXF-R00
00120	Bevestigingsplaat Y as (4mm dik)	M	JA	DXF	NEE	JA	00	00001		T00120-DXF-R00
00121	Bevestiging Nema Y as (4mm dik)	M	JA	A4	NEE	JA	00	00001		T00121-0A4-R00
00122	Eindplaat rails (4mm dik)	M	JA	DXF	NEE	JA	00	00001		T00122-DXF-R00
00123	y-as staaf Ø 16mm	M	JA	A3	NEE	JA	01	00001		T00123-0A3-R01
00124	Bevestigingsplaat motor x-as (4mm dik)	M	JA	DXF	NEE	JA	00	00001		T00124-DXF-R00
00125	X-as onderkant	M	JA	A4	NEE	JA	00	00001		T00125-0A4-R00
00126	Oppikplaat (3mm dik)	M	JA	NEE	NEE	JA	00	00003		T00126-000-R00
00127	lager binnenØ 15mm DIN 625 6002	K	JA	NEE	NEE	NEE	00	00003		T00127-E00-R00
00128	As rotatie r richting	M	JA	A4	NEE	JA	00	00003		T00128-0A4-R00
00129	Zegering	K	JA	NEE	NEE	NEE	00	00003		T00129-E00-R00

Tekeningnummer:	Omschrijving:	K/M	CAD Model:	Drawir	3D machining file (Bijv. STL of STEP):	Custom:	Hoogste revisie:	Samenstelling:	Link:	Bestandsnaam:
00130	Draaiflens	M	JA	A4	NEE	JA	00	00003		T00130-0A4-R00
00131	Reely Stalen tandwiel Boor-Ø = 5mm (Conrad artnr: 240311)	K	JA	NEE	NEE	NEE	00	00003		T00131-E00-R00
00132	Reely polyacetaal tandwiel Boor-Ø = 8mm (Conrad artnr: 1515292)	K	JA	NEE	NEE	NEE	00	00003	Link	T00132-E00-R00
00133	Plateau stappenmotor R-as (3mm)	M	JA	A4	NEE	JA	00	00003		T00133-0A4-R00
00134	Zijplaat plateau (3mm)	M	JA	DXF	NEE	JA	00	00003		T00134-DXF-R00
00135	Plaat draaiplateau (3mm)	M	JA	DXF	NEE	JA	00	00003		T00135-DXF-R00
00136	Meenemer tandriem (3mm)	M	JA	DXF	NEE	JA	01	00003		T00136-DXF-R01
00137	Oplasplaat plateau stappenmotor	M	JA	NEE	NEE	JA	00	00003		T00137-000-R00
00138	Oplasplaat plateau stappenmotor gespiegeld	M	JA	NEE	NEE	JA	00	00003		T00138-000-R00
00139	Zijplaat magazijn (2mm)	M	JA	DXF	NEE	JA	00	00002		T00139-DXF-R00
00140	Top- en onderplaat magazijn (2mm)	M	JA	A4	NEE	JA	00	00002		T00140-0A4-R00
00141	Voetje	K	JA	NEE	NEE	NEE	00	00002		T00141-E00-R00
00142	Zijplaat rechtsonder (4mm)	M	JA	A4	NEE	JA	00			T00142-0A4-R00
00143	Zijplaat links (4mm)	M	JA	DXF	NEE	JA	00			T00143-DXF-R00
00144	Zijplaat rechtsboven (4mm)	M	JA	A4	NEE	JA	00			T00144-0A4-R00
00145	Reely Kogellager radiaal Chroomstaal Binnen binnenØ 8mm, buitenØ 22mm(Conrad	K	JA	NEE	NEE	NEE	00			T00145-E00-R00
00146	Rittal AE 1045.500 schakelkast 400x500x210mm (Conrad artnr:531738)	K/M	JA	NEE	NEE	NEE	00			T00146-E00-R00
00147	Vloer [referentie]		JA	NEE	NEE	NEE	00			T00147-E00-R00
00148	Bakje	K	JA	NEE	NEE	NEE	00	00003		T00148-E00-R00
00149	Panasonic PM-Y65-P Infrarood gate sensor (Conrad artnr: 1547442)	K	JA	NEE	NEE	NEE	00	00001	Link	T00149-E00-R00
00150	Eindstop X-as links (X+) en rechts (X-)	M	JA	NEE	JA	JA	00	00001		T00150-03D-R00
00151	Bracket eindstop Y boven (Y+)	M	JA	NEE	JA	JA	00	00001		T00151-03D-R00
00152	Bracket eindstop Y onder (Y-)	M	JA	NEE	JA	JA	00	00001		T00152-03D-R00
00153	Panasonic PM-F65W-P infrarood gate sensor (Conrad artnr: 1547476)	K	JA	NEE	NEE	NEE	00	00001		T00153-E00-R00
00154	Barcode Scanner Module - 1D/2D Codes Reader (Kiwi electronics artnr:WS-14810)	K	JA	NEE	NEE	NEE	01		Link	T00154-E00-R01
00155	Onderplaat weegschaal	M	JA	A4	NEE	JA	00			T00155-0A4-R00
00156	Bovenplaat weegschaal	M	JA	NEE	NEE	JA	00			T00156-000-R00
00157	Load cell 5kg (Tinytronics artnr: 000951)	K	JA	NEE	NEE	NEE	00		Link	T00157-E00-R00
00158	Achterplaat 1v2 (2mm dik)	M	JA	DXF	NEE	JA	00	0155		T00158-DXF-R00
00159	Achterplaat 1v2 (2mm dik)	M	JA	DXF	NEE	JA	00	0155		T00159-DXF-R00

Tekeningnummer:	Omschrijving:	K/M	CAD Model:	Drawir	3D machining file (Bijv. STL of STEP):	Custom:	Hoogste revisie:	Samenstelling:	Link:	Bestandsnaam:
00160	Nema 24 Stappenmotor 4.0Nm(566 oz.in) w/ Rem Wrijvingskoppel 2.0Nm(283oz.in) (	K	JA	NEE	NEE	NEE	00			T00160-E00-R00
00161	Bracket eindstop R as (R+ en R-)	M	JA	A4	NEE	JA	00	00003		T00161-0A4-R00
00162	Eindstop R as (R+ of R-)	M	JA	NEE	JA	JA	03			T00162-03D-R03
00163	Eindstop Z	M	JA	NEE	JA	JA	01			T00163-03D-R01
00164	Eindstop houder R+	M	JA	NEE	JA	JA	00			T00164-03D-R00
00165	Eindstop houder R-	M	JA	NEE	JA	JA	00			T00165-03D-R00
00166	HabaSYNC T10 Conveyer timing belt (Misumi)	K	JA	NEE	NEE	NEE	00	00001	Link	T00166-E00-R00
00167	Bevestigingsplaat connectoren	M	JA	A4	NEE	JA	00			T00167-0A4-R00
00168	Hoeksteun eindstop Z	M	JA	NEE	JA	JA	00	00003		T00168-03D-R00
00169	7 inch HDMi Display-c 1024x600 pixel USB touch (Tinytronics)	K	JA	NEE	NEE	NEE	00		Link	T00169-E00-R00
00170	Cover 7 inch Display	M	JA	NEE	JA	JA	00	00000		T00170-03D-R00

Appendix 6: flowchart




confirmation = yes
Send "place+spot" to Nucleo
Move container to que
where spot = x,y, weight = recieved.weight
send weight to pi
place container to spot and return to position
Show done for 3 sec
If status.recieve = done
send "done" to Pi
Show "obtaining container"
Send "request+spot" to Nucleo
Move container to que
where spot = x,y, weight = recieved.weight
send weight to pi
If status.recieve = done
send "done" to Pi
scan container
if code = 0
Ask for confirmation
Send "component requested+id"
On buttonpress "yes"
Send "confirmation" to Pi
If confirmation = yes
No
Move container to exit
Show done for 3 sec
on recieve "done" send "done" to client
send "done" to Pi

Appendix 7: Test plans

A.

LMI Re-store HI PLATE

Test Plan

Version *1.0*

05/12/18

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	*Stan Heemskerk*	*<mm/dd/yy>*	*<name>*	*<mm/dd/yy>*	Initial Unit Testing

UP Template Version: 12/31/07

Introduction

1.1 Purpose of The Test Plan Document

The Test Plan document documents and tracks the necessary information required to effectively define the approach to be used in the testing of the project's product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager and project team. Some portions of this document may on occasion be shared with the client/user and other stakeholder whose input/approval into the testing process is needed.

2 Functional Testing

2.1 Test Risks / Issues

The bin can still be pulled through the hole. Or not possible to take bolts via the new plate.

2.2 Items to be Tested / Not Tested

Item to Test	Test Description	Test Date	Responsibility
Does the bin fit through the hole?	A person should never be able to take the hole bin in in what ever position. So try to take it out!	23-1-2018	Stan
Are you able to take bolt?	Person should be able to take the nuts and bolts placed in the bins try to take bolts	23-1-2018	Stan

2.3 Test Approach(s)

A random person will be asked to take some nuts and bolts from the storage. He/she can try to take them. After he/she can try to take out the whole bin.

2.4 Test Regulatory / Mandate Criteria

Test must be done with at least 2 students in order to control the test conditions.

2.5 Test Pass / Fail Criteria

The placement of the bin must:

- The bin cannot be taken out → pass
- The person can take the bolts out → pass

2.6 Test Entry / Exit Criteria

The bin will be a almost empty one, and almost full once for testing if they are able to take the bolts.

2.7 Test Deliverables

Documentation about the test

2.8 Test Suspension / Resumption Criteria

Before the testing, if the code on the Arduino is of importance, the Arduino code must be saved. During testing, no-one else can program or use any of the electronics on the system. After testing, the original program, if the code is of importance, must be re-uploaded.

2.9 Test Environmental / Staffing / Training Needs

An engineer who worked on the HI plate must be present at any point of the testing sequence.

Test Plan Approval

The undersigned acknowledge they have reviewed the *LMI Restore Arm Rotation* **Test Plan** document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Signature:		Date:	
Print Name:			
Title:			
Role:	Project leader		

Signature:		Date:	
Print Name:			
Title:			
Role:	Software Engineer		

Signature:		Date:	
Print Name:			
Title:			
Role:	Arm Redesign Engineer		

B.

LMI Re-store carrot test

Test Plan

Version *1.0*

05/12/18

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	*Stan Heemskerk*	*<mm/dd/yy>*	*<name>*	*<mm/dd/yy>*	Initial Unit Testing

UP Template Version: 12/31/07

Introduction

2.1 Purpose of The Test Plan Document

The Test Plan document documents and tracks the necessary information required to effectively define the approach to be used in the testing of the project's product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager and project team. Some portions of this document may on occasion be shared with the client/user and other stakeholder whose input/approval into the testing process is needed.

3 Functional Testing

3.1 Test Risks / Issues

There can be a risk of people getting hurt by the storage. To know for sure if the storage would ever hurt somebody. The carrot test will be done. This means putting a carrot between every moving part to insure a human finger would survive. Else the

3.2 Items to be Tested / Not Tested

Item to Test	Test Description	Test Date	Responsibility
Person gets hrut when finger is between bin and outer plate	A carrot will be placed between a the bin and the outerplate, then the Z-axis motor will be actuated and we will look for damage on the carrot		
Person gets hurt when finger gets between the belt of Y-axis motor	A carrot will be placed between the belt and the pulley then actuate the Y-axis motor, and look for damage on the carrot		

3.3 Test Approach(s)

A piece of Arduino code will be written that will use the motor driver for the Y and Z axis of the arm and the infrared sensors belonging to these axis. The arms will be actuated when the carrot is in place. Afterwards the carrot will be inspected to look for damage.

3.4 Test Regulatory / Mandate Criteria

Test must be done with at least 2 students in order to easily shut off the system when things appear to go wrong, breaking a axis for example.

3.5 Test Pass / Fail Criteria

The carrot may not:

- Break in half.
- Be cut for more than half.

3.6 Test Entry / Exit Criteria

The arm starts with the bin and will bump in to the carrot immediately.

3.7 Test Deliverables

Arduino code of the test. The damage is filmed by a camera.

3.8 Test Suspension / Resumption Criteria

Before the testing, if the code on the Arduino is of importance, the Arduino code must be saved. During testing, no-one else can program or use any of the electronics on the system. After testing, the original program, if the code is of importance, must be re-uploaded.

3.9 Test Environmental / Staffing / Training Needs

An engineer who worked on the HI plate and a n extra person must be present at any point of the testing sequence.

Test Plan Approval

The undersigned acknowledge they have reviewed the *LMI Restore Arm Rotation* **Test Plan** document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Signature: ______________________ Date: __________

Print Name: ______________________

Title: ______________________

Role: Project leader

Signature: ______________________ Date: __________

Print Name: ______________________

Title: ______________________

Role: Software Engineer

Signature: ______________________ Date: __________

Print Name: ______________________

Title: ______________________

Role: Arm Redesign Engineer

C.

LMI Re-store weighing

Test Plan

Version *1.0*

05/12/18

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	*Stan Heemskerk*	*<mm/dd/yy>*	*<name>*	*<mm/dd/yy>*	Initial Unit Testsing

UP Template Version: 12/31/07

3.10 Purpose of The Test Plan Document

The Test Plan document documents and tracks the necessary information required to effectively define the approach to be used in the testing of the project's product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager and project team. Some portions of this document may on occasion be shared with the client/user and other stakeholder whose input/approval into the testing process is needed.

4 Functional Testing

4.1 Test Risks / Issues

The bin will be placed to much to the left/right and will hit the storage frame. This damages to the drivetrain of the arm.

4.2 Items to be Tested / Not Tested

Item to Test	Test Description	Test Date	Responsibility
Placement of bin on desired place	The bin must be places 30 times in a row without missing the loadcell. 15 times with load and 15 times without		

4.3 Test Approach(s)

A piece of Arduino code will be written that will only use the motor driver for the X,Y and Z axis of the arm and the infrared sensors belonging to these axis. The bin starts from 30 different positions within the storage. And will bring the bin to the weighing sensor. The bins will be filled for 15 times and emptied for the remaining 15 test runs.

4.4 Test Regulatory / Mandate Criteria

Test must be done with at least 2 students in order to easily shut off the system when things appear to go wrong, breaking arm due to wrong placement.

4.5 Test Pass / Fail Criteria

The placement of the bin must:

- Stop when the sensor is activated with and without load.
- Place the bin within the loadcells bottom plate.

4.6 Test Entry / Exit Criteria

The arm starts with the bin and will travel to the desired place, from where it will go to *the weighing area.*

4.7 Test Deliverables

Arduino code of the test. The placement filmed by a camera.

4.8 Test Suspension / Resumption Criteria

Before the testing, if the code on the Arduino is of importance, the Arduino code must be saved. During testing, no-one else can program or use any of the electronics on the system. After testing, the original program, if the code is of importance, must be re-uploaded.

4.9 Test Environmental / Staffing / Training Needs

A software engineer and an engineer who worked on the weighing cell must be present at any point of the testing sequence.

Test Plan Approval

The undersigned acknowledge they have reviewed the *LMI Restore Arm Rotation* **Test Plan** document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Signature: ______________________ Date: ____________

Print Name: ______________________

Title: ______________________

Role: Project leader

Signature: ______________________ Date: ____________

Print Name: ______________________

Title: ______________________

Role: Software Engineer

Signature: ______________________ Date: ____________

Print Name: ______________________

Title: ______________________

Role: Arm Redesign Engineer

D.

LMI Re-store Arm Reach

Test Plan

Version *1.0*

05/12/18

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Glenn Smulders	<mm/dd/yy>	<name>	<mm/dd/yy>	Initial Unit Testsing

UP Template Version: 12/31/07

4.10 Purpose of The Test Plan Document

The Test Plan document documents and tracks the necessary information required to effectively define the approach to be used in the testing of the project's product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager and project team. Some portions of this document may on occasion be shared with the client/user and other stakeholder whose input/approval into the testing process is needed.

5 Functional Testing

5.1 Test Risks / Issues

The arm might not be able to reach the containers. Therefore adjustments have been made to extend the arm.

5.2 Items to be Tested / Not Tested

Item to Test	Test Description	Test Date	Responsibility
Arm - Motor stops when the infrared sensor is activated	Arm must stop when the end sensor is activated		
Arm - Arm reaches container when standing in the back.	Arm must reach and grab the container when placed 0 mm of the back plate.		

5.3 Test Approach(s)

A piece of Arduino code will be written that will only use the motor driver for the translation Z of the arm and the infrared sensor belonging to the translating movement of the arm.

5.4 Test Regulatory / Mandate Criteria

Test must be done with at least 2 students in order to easily shut off the system when things appear to go wrong. (Motor not stopping)

5.5 Test Pass / Fail Criteria

The translation Z of the arm must:

- Stop when the sensor is activated with and without load.
- Reach and grab a container when it's move all the way to the back.

5.6 Test Entry / Exit Criteria

The arm must be still and the translational speed must be zero. After the test the arm should also be still and the arm itself may not be damaged. The Container must be *grabbed by the arm.*

5.7 Test Deliverables

Arduino code of the test. Translation of the arm filmed by a camera.

5.8 Test Suspension / Resumption Criteria

Before the testing, if the code on the Arduino is of importance, the Arduino code must be saved. During testing, no-one else can program or use any of the electronics on the system. After testing, the original program, if the code is of importance, must be re-uploaded.

5.9 Test Environmental / Staffing / Training Needs

A software engineer and the engineer responsible of the arm redesign team must be present at any point of the testing sequence.

6 Test Plan Approval

The undersigned acknowledge they have reviewed the *LMI Restore Arm Reach* **Test Plan** document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Signature: ______________________ Date: __________

Print Name: ______________________

Title: ______________________

Role: Project leader

Signature: ______________________ Date: __________

Print Name: ______________________

Title: ______________________

Role: Software Engineer

Signature: ______________________ Date: __________

Print Name: ______________________

Title: ______________________

Role: Arm Redesign Engineer

E.

LMI Re-store Arm Rotation

Test Plan

Version *1.0*

05/12/18

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	*Glenn Smulders*	*<mm/dd/yy>*	*<name>*	*<mm/dd/yy>*	Initial Unit Testsing

UP Template Version: 12/31/07

7 Introduction

7.1 Purpose of The Test Plan Document

The Test Plan document documents and tracks the necessary information required to effectively define the approach to be used in the testing of the project's product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager and project team. Some portions of this document may on occasion be shared with the client/user and other stakeholder whose input/approval into the testing process is needed.

8 Functional Testing

8.1 Test Risks / Issues

When over rotating, the arm can damage the system. Components that can break are the sprocket and infrared sensor.

8.2 Items to be Tested / Not Tested

Item to Test	Test Description	Test Date	Responsibility
Rotational movement of the arm	Arm must move within 0 and 270 degrees.		
Stopping of the rotational movement	The rotational movement of the arm must be stopped by the activation of the infrared sensor		

8.3 Test Approach(s)

A piece of Arduino code will be written that will only use the motor driver for the rotational movement R of the arm and the infrared sensor belonging to the rotational movement of the arm.

8.4 Test Regulatory / Mandate Criteria

Test must be done with at least 2 students in order to easily shut off the system when things appear to go wrong. (over rotating)

8.5 Test Pass / Fail Criteria

The rotation of the arm must:

- Stop when the sensor is activated with and without load.
- Rotate 270 degrees with and without load.

8.6 Test Entry / Exit Criteria

The arm must be still and the rotational speed must be zero. After the test the arm should also be still and the arm itself may not be damaged.

8.7 Test Deliverables

Arduino code of the test. The rotation of the arm filmed by a camera.

8.8 Test Suspension / Resumption Criteria

Before the testing, if the code on the Arduino is of importance, the Arduino code must be saved. During testing, no-one else can program or use any of the electronics on the system. After testing, the original program, if the code is of importance, must be re-uploaded.

8.9 Test Environmental / Staffing / Training Needs

A software engineer and the engineer responsible of the arm redesign team must be present at any point of the testing sequence.

Test Plan Approval

The undersigned acknowledge they have reviewed the *LMI Restore Arm Rotation* **Test Plan** document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Signature: ______________________________ Date: ______________

Print Name: ______________________________

Title: ______________________________

Role: Project leader

Signature: ______________________________ Date: ______________

Print Name: ______________________________

Title: ______________________________

Role: Software Engineer

Signature: ______________________________ Date: ______________

Print Name: ______________________________

Title: ______________________________

Role: Arm Redesign Engineer