Automation with HORST600

Project Work by
Jires Donfack Voufo

In Partial Fulfillment of the Requirements for the
Degree of
Bachelor of Engineering - ELECTRONICS

HOCHSCHULE
HAMM-LIPPSTADT

HOCHSCHULE HAMM-LIPPSTADT
First University Examiner: **Prof Dr. Ali Hayek**

ACKNOWLEDGEMENTS

I would like to Thanks Prof Dr Ali Hayek from initiating this Project.He assisted with the installation and commissioning of the robot.He helped with all the issues related to exchanging the robot,getting the logging details and a Fruitcore employee to assist us. I am grateful for the opportunity and the knowledge that I gain during this project.

ABSTRACT

Robotic arm are used in automation for tasks that need to be repeated with a high level of precision and accuracy.In some cases like in the automotive industry they are used to lift very heavy car parts that no human will otherwise be able to.Robotic arms also helps companies to meet their production target when the demand for its product is high.Given the level of influence and importance this has in our modern society, that is the reason we decided to take a look at it in this project.We started this report by giving a brief history of industrial automation and how it had evolve over the years.We then deep into our theme which is industrial robots arms.We described industrial robot,the different types that exist and the way they operate.We introduce Fruitcore robotics which is the startup from which we bought the robot for this project.We described the robot that we used for our project and the software that came with it.Safety was also an important point,given that we needed to make sure no injuries occurs during operation.Using the horstFX software and its build in simulation capabilities and digital twin,We did 3 examples for illustration purpose.We gave a result of the simulations and explained the challenges that we faced during this project.To conclude we spoke about a way forward.

TABLE OF CONTENTS

Acknowledgements ii
Abstract iii
Table of Contents iv
List of Illustrations v
Chapter I: Introduction vi
Chapter II: Safety xi
Chapter III: Use Cases for HORST 600 xiii
Chapter IV: Results and Evaluation xxi
Chapter V: Conclusion xxiv
Chapter VI: Appendix xxv
Chapter VII: Declaration of originality xxviii
Bibliography xxix

LIST OF ILLUSTRATIONS

Number *Page*

1.1 Steps leading to Industry 4.0 vi
1.2 Solution Kit 1[4] viii
1.3 Solution kit 2[4] viii
2.1 Connection of the Laser Scanner[3] xi
2.2 Connection E-Stop for 2 Robots[3] xi
2.3 Mounting mechanical stop 1[3] xii
2.4 Mounting mechanical stop 2[3] xii
3.1 HORST axes[3] xiii
3.2 HORST components[3] xiii
3.3 code 1 xvii
3.4 code 2 xvii
3.5 Palette Settings 1 xix
3.6 Palette Settings 2 xix
3.7 result 1 xx
3.8 result 2 xx
4.1 result 1 xxi
4.2 result 2 xxi
4.3 rel Palettizing Point xxii
4.4 Drop spot xxii
4.5 rel Dep & approach Point xxii
4.6 Palette completed xxii
4.7 Overview of the Word "his" xxiii
4.8 Robot running through each letter xxiii
6.1 I/O Interface [3] xxv
6.2 Side view of working area [3] xxv
6.3 Top view of working area [3] xxv
6.4 Digital Input Connection [3] xxvi
6.5 Sensor connection on DI [3] xxvi
6.6 Control via robot axes xxvi
6.7 Control via coordinate values or orientation xxvii
6.8 Robot Axes limits xxvii

Chapter 1

INTRODUCTION

Mass production systems started been produced at the end of the 18th century.The machines used in this era were mostly power by steam and water, the demand for human workforce was therefore decreased.This also marked the advent of the textile railway -industry.The next stop into this revolution can be situated at the end of the 19th century.This can also be named 2nd industrial revolution. Here machines were powered by electricity.Conveyor belt was used for the first time. More systems were also automatized, and the production output was drastically increased.The 3rd industrial revolution started early in the years 1970.Here we can talk about revolutionary technology like the internet, semiconductors and powerful software and computers.PLC(programmable logic controllers) came into service for the first time .At the moment the world is in the middle of the 4th industrial revolution which is driven my CPS[7].The technology behind industrial automation helps to continuously increase efficiency,productivity and minimise human input in jobs that require repetition.In some cases the aim is to have robot that can work alongside(or collaborate)with human or to completely replace the workforce for long term cost saving reasons.Industrial robots arms count amount the solution employ in the field of automation technology.They can be use for:welding,spray painting,material handling and palletizing.To pick the right robot for the application we have to take in consideration:weight to be carry,the reach,task to be automated,cycle time and the working environment.The picture below illustrate the different steps leading to the current state of industrial automation.

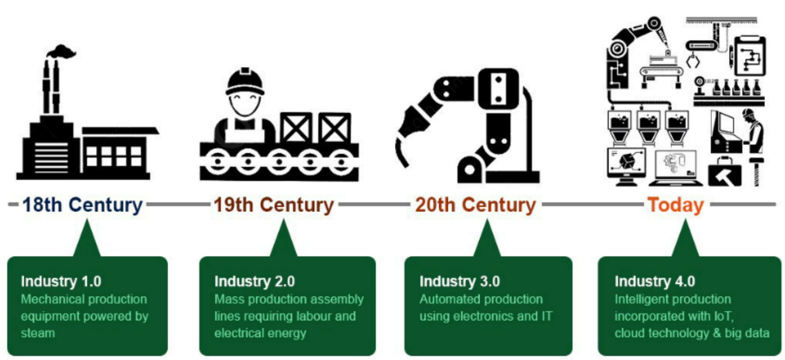

Figure 1.1: Steps leading to Industry 4.0

1.0.1 What is an industrial robot

An industrial robot is an automatically controlled, multi-active, versatile and programmable system that consists of three or more axes.The market for industrial robotic arm is very competitive.The 4 biggest manufactures are ABB,KUKA,FANUC and Yaskawa.They control up to 75% of this business.They offer different types like:Delta,SCARA and cartesian robot systems.Companies like Universal Robots and Fruitcore robotics are more specialise in lightweight,flexible robots.Meanwhile FANUC and Yaskawa are better known for the ones in the automotive industry.
There are two types of robots[1]:

- Cartesian robots
  Also called Gantry robots,they move according to an orthogonal (X,Y and Z) coordinate system.They follow a straight line motion with the help of linear actuators and 3 sliding joints.They are well suited for tasks that required high level of accuracy a repeatability.

- poly-articulated robot
  They are equipped with rotary joints that make them capable of mimicking 3D human arm motions. It usually has 6 axes and can have up to a maximum of 10 joints.Because poly.articulated robot can bend and twist in very thigh space,it make them suitable for tasks like surgery.

The five components of an industrial robots are[5]:

- Manipulators
  These are joints or mechanical linkages that make it possible for the robot to operate all sort of movements.They can be powered by pneumatic,hydraulic or electric drive systems(servo/stepping motors).Gear systems or chains can also be use to power the motion.

- End Effectors
  This are components that can be mounted at the end of the robot arm.They can be grippers or process tools.There are active and passive grippers.As example we can mention:Magnetic grippers,vacuum/suction grippers,2 or Multiple finger grippers.They can be powered by pneumatic, hydraulic or electricity.

- Feedback Devices
  They are used to monitor and check the state of the process.As example we can name sensors.

- Controllers
  This is the computer equipped with hardware and software that is in charge of the decision making making and the motion of the robot.The controller can be couple with external devices and the control program can be written in languages like programming by learning, graphics,or python.This also depend on the brand of robot we are dealing with.

1.0.2 Who is Fruitcore Robotics

Fruitcore robotics is a company founded in 2017 and based in Konstanz,Germany that offers industrial automation solutions.It develops and manufacture digital robots. Fruitcore robotics solution differs from the competition by the use innovative software and artificial intelligence to make its robots intelligent.It offers standard interfaces to make it possible for them to be seamlessly integrated in the existing process.Fruitcore robotics technology offers digital services like predictive maintenance,online-backups,optimisation of the code.This helps makes the production more efficient,and continuously optimise all processes in a short amount of time. Fruitcore robotics aims to offer the best return-on-investment on the product purchase by the client[2].The pictures bellow illustrate some of the solution kits from Fruitcore robotics.

Figure 1.2: Solution Kit 1[4]

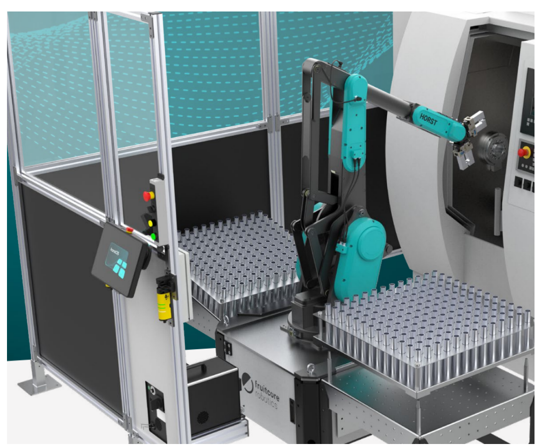

Figure 1.3: Solution kit 2[4]

1.0.3 HORST Robot System [3]

Every HORST robot system comes with the following components:

- HORST robot
  Fruitcore robotics offers the following three robots: HORST 600,1000,1400 and the new comer 1500.HORST stands for highly optimized robotic systems technology.Based on the model, they can carry a weight between three and twelve kilograms.They have a reach between 584 and 1425 millimeter.They are all rated IP54, which mean they offer protection against dust and splashed water from all directions.They all have 6 axis of motion.

- horstControl Cabinet
  This is considered as the brain of the whole system.It has six safety inputs,eight safety outputs.added to that are 20 more inputs and 18 outputs that can be configured in the horstFX software .Sensors,actuators and other devices can be connected via the I/O or Profinet and Ethernet port.I/0 can be configure for example to start/stop a program ,acknowledge signals related to emergency stop or simply driving robot with low speed.

- horstPanel
  13" FUll-HD control panel with a fast response time.It has 2 USB-Ports to connect mouse,keyboards or a external memory to upload a code into the system.It also has an emergency button and enabling switch.

- Mobile Roboter Basis
  This is where the HORST get mounted.

- Software horstFX
  In horstFX the robot can be programmed using graphics and text.There is also a build-in digital twin for virtual simulation of the code

The five communication interfaces are as follow:

- Digital inputs and outputs
  Standard digital 24V.It can be use for applications like pick and place

- Profinet
  This is the industry standard communication for PLC

- Modbus
  Low data transfer rate and safety protocol.Master slave concept.

- TCP/IP
  connection for external devices that use Ethernet.A camera or PC can be connected through it.

- XML-RPC
  This makes the remote execution of methods possible and the XML formatted data transfer is done via http.

The other frameworks and standards that are supported:OPC UA, ROS, SILA

1.0.4 Interfaces Overview [3]

The following interfaces are available on the horstIO

- 7 safety-critical inputs and 6 safety-critical outputs
  It includes 3 dedicated inputs for:Emergency stop input,Safety stop input and Internal enabling switch.There are also 4 configurable outputs (external emergency stop, acknowledgement signals).On the safety-critical outputs side we have 4 configurable outputs, push-pull and 2 configurable, potential-free outputs (two relay contacts each)

- Test signal generation TA/TB
  OSSD signal for safety-critical inputs

- 28 general digital inputs and 30 general outputs

- Shared GND and +24 V

- RS-485 interface
  This can be used for future extension

- External bridge for +24 V:
  When routed via the safety relay ,it enable the safety criticall switch off of digital outputs 1-16

Chapter 2

SAFETY

2.0.1 Laser Scanner & Emergency Button

One of the safety device that came with the robot is the laser scanner. It is suppose to stop the robot once the define protected area has been violated. There are two emergency stop buttons,one at the top right-hand side of the Panel and a second one is mounted on the robot basis .There is also a possibility to connect additional Emergency stop buttons to the system.For the robot to continue operating after E-stop activation,We need to unlock that E-stop and acknowledge the warning message in the software.A safety PLC or relay can also be wired to the safety in and output of the interface.No emergency stop devices should be connected to general digital or safety stop inputs. The pictures bellow show how the safety scanner.In the one next to it we can see how one E-stop can be wired for 2 robots at the same time.

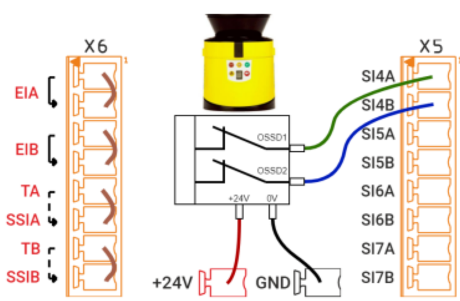

Figure 2.1: Connection of the Laser Scanner[3]

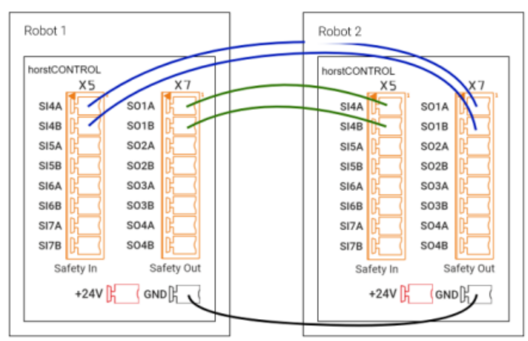

Figure 2.2: Connection E-Stop for 2 Robots[3]

2.0.2 Operation Mode & Enabling Switch

Operation Mode The system has 3 operation modes:

- Teaching mode T1
  manual operation with reduced speed of max 250 mm/s.To move the robot, the enabling switch need to be pressed in the middle position.It is also known as programming mode
- Teaching mode T2
  manual operation at high speed above 250 mm/s.In this mode the robot also

need the enabling switch to be moved.This mode is mostly used for verification purpose.

- A
  This is for the automatic execution of the program.Here there is no need for the enabling switch for operation.All safety functions are also still active.

The Enabling Switch has 3 different positions.When it is pressed in middle position,the robot can be operated.

2.0.3 Movement restriction

The robot can be supply with mechanical stopping devices.They are there to restrict the movement in the working environment.Once they are install,the operation space must also be restricted in the software to avoid damage to the robot.

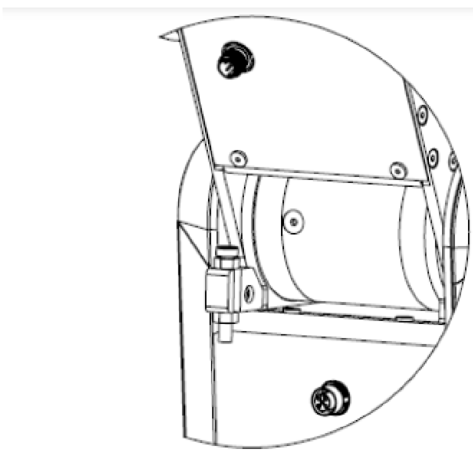

Figure 2.3: Mounting mechanical stop 1[3]

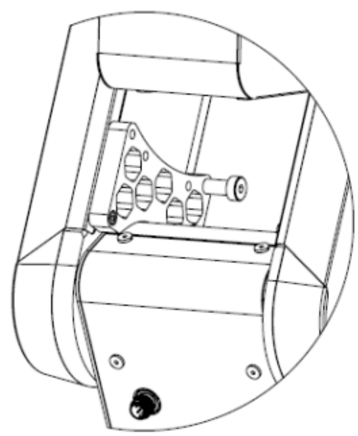

Figure 2.4: Mounting mechanical stop 2[3]

2.0.4 Safe Shot down

After we done using the robot,we have to turn it off in a safe manner.To do so we have to initially cancel any running program.We then have to check if the robot is in a safe position and that the gripper is not holding any object.In case needed,we can move the robot in a safe position using the "free navigation" tool in the horstFX software.The next step is to go into the main menu of the software,tap the "Exit horstFX",select the "shut down system" option and validate with "OK".Additionaly to what we mention before,to avoid any risk to the robot:we have to hold it by the swivel arm and not move the axes by force.The figure bellow shows the working area of the robot.

Chapter 3

USE CASES FOR HORST 600

3.0.1 Robot Description[3]

This robot was especially design for loading/unloading CNC machines and pick/ place tasks.It can carry a weight of up to 3 Kilos.The movement of the HORST600 are optimize with artificial intelligence.Each axes has a high acceleration rate that reduces the cycle time.This makes a pick and place task more efficient.It has a reach of 584mm,repetition accuracy of +/- 0.05mm and weights almost 30kg.The robot can be program using graphical and Textual method The picture bellow depict the robot use in this project which is a HORST 600.A1 to A6 shows all the different axes in which the robot can move

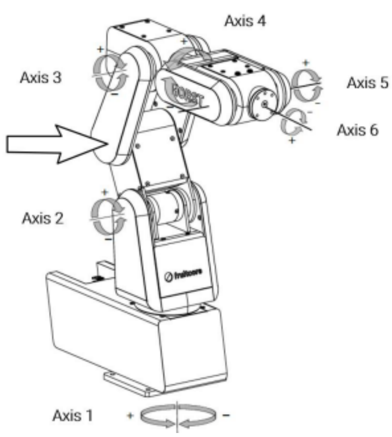

Figure 3.1: HORST axes[3]

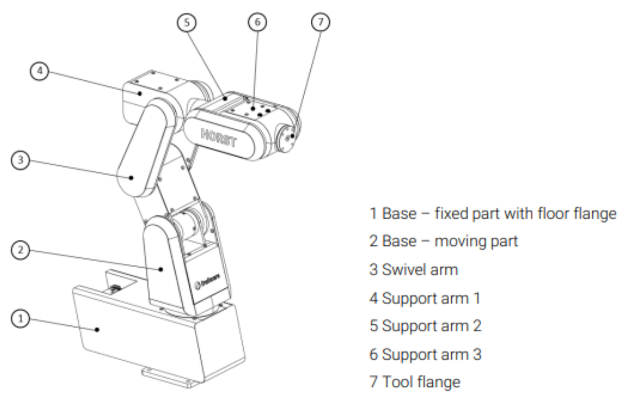

Figure 3.2: HORST components[3]

3.0.2 How to control the Robot using HorstFX

The robot can be controlled using the following two ways:

- Control via robot axes To move an axis, the desired axis must first be selected.We do this by either tapping the axis marker in the 3D world or by tapping the axis symbol between the plus and minus buttons. By pressing and holding these buttons, the robot rotates around the selected axis in the corresponding direction.

- Control via coordinate values or orientation
  Here the movement can be done either by a linear motion (translation) along a certain axis or by a rotation around a that axis.The desired axis is first selected by tapping the axis symbol between the plus and minus controls. Alternatively, the corresponding axis can be tapped in the coordinate system visualized in the 3D world. By holding down the plus and minus buttons that are then activated, the robot performs the corresponding movement.

The HorstFX software uses two types of coordinates system.The base coordinate system follows the global world coordinate system and always remains fixed, no matter how the robot is moved.The TCP coordinate system depends on the orientation of the tool and can therefore be transverse in space depending on the position of the robot.When at least one axis comes close to its limit,the robot model is displayed as a wire frame model.There is also a flashing red ball that is displayed on the corresponding axis.

3.0.3 Robot Installation and Commissioning

The package was delivered with the robot itself ,user manual,control panel,touch screen display with its holding bracket ,cables, some screws.The robot platform came separately with and emergency button and safety laser already mounted to it. The installation steps of the robot were as follow:

- Steps 1
  We started by removing the four M10*16 screws holding the robot to the transport palette.It was then lifted by hand placed and aligned in the middle of the robot basis.Four M8*20 screws were used to bolt it down and fix it to the platform.We control that the robot was well secured.

- Steps 2
  We connected the horstPanel and the robot and the horstControl panel.We turn the mode switch into the T1 position.We made sure all connecting cables were safely routed and they will not be on the way during the operation of the robot.

- Steps 3
  Turn On the power to the system.On the control display,we put in the Admin username and password.The robot have to be reinitialise every time it is turn

on.From the"initialise robot " menu of the horstFX software we press and hold onto "Auto Init"to initialise all 6 axes .There is an initialisation status progress bar in the software that shows when the process is completed.

- Steps 4
  we check how far the robot will stretch to sure there was enough room for the robot arm to move freely and no obstacles was in the working area.We then check the dimensions of the safety cage.We then lifted it and placed it over the robot and on top of the surface of basis.

3.0.4 Illustration Example 1

In this first example 1 we simulated the machining of work piece.The robot carries a cutting tool as End effector.It start at a define initial point and move around all four sides of the work piece.

- Start/configuration
  After creating a project and giving it a name,We go to the "Tool & Coord"Tab.Here we set the cutting tool to be use in our program.We will approach all our waypoints with this tool,given that it is the only one that we need.under the "Weight"Tab the maximum weight to be carried is already set as 3Kg and cannot be changed for the Horst600.Under the "3D world "window,we set our environment to "Training_WORLD".We then apply the changes.

- Waypoint 1
  We named it in out program as "the Start point".It defines the relative approaching point to the work piece

- Cutting Tool Initialisation
  A folder was created in which we set 2 outputs to start and stop the cutting tool.Those outputs are selected in a drop down list as:"TOOL_OUTPUT_1" ,"TOOL_OUTPUT_2" and are set as 1.0

- Cutting Tool Start
  Here the cutting tool get started before we approach the first corner of the work piece.We do so by setting the corresponding output "TOOL_OUTPUT_2" to 1.0 after a delay of 100 ms

- Waypoint 2
  This define the first corner of the work piece and it has the following coordi-

nate: X=304.96 mm ,Y=-93.62 mm, Z=42.67 mm.It is named "Corner_1" in the program

- Waypoint 3
  This define the second corner of the work piece and it has the following coordinate:X=461.34 mm ,Y=-93.62 mm, Z=42.67 mm.It is named "Corner_2" in the program

- Waypoint 4
  This define the third corner of the work piece and it has the following coordinate:X=461.34 mm ,Y=-205.02 mm, Z=42.67 mm.It is named "Corner_3" in the program

- Waypoint 5
  This define the fourth corner of the work piece and it has the following coordinate:X=30183 mm ,Y=-205.02 mm, Z=42.67 mm.It is named "Corner_4" in the program

- Waypoint 6
  This is the first corner which is also the starting point.It is named "Corner_5" in the program

- Waypoint 7
  this point named "End Point" tells the robot to return to its initial start position after the job is done

- Cutting Tool Stop
  Here the cutting tool is stopped after returning to the starting point. We do this by setting the "TOOL_OUTPUT_2" to "1.0".

- Loop
  With this loop we can go around the work piece until we obtain the desired result.So the value of the passes can be increased or decreased depending on the need.

From "Corner_2" to "Corner_5" the operation speed was set to 20% and it can also be adjusted based on the application.The figures 3.3 and 3.4 bellow illustrate some lines of the graphical code.

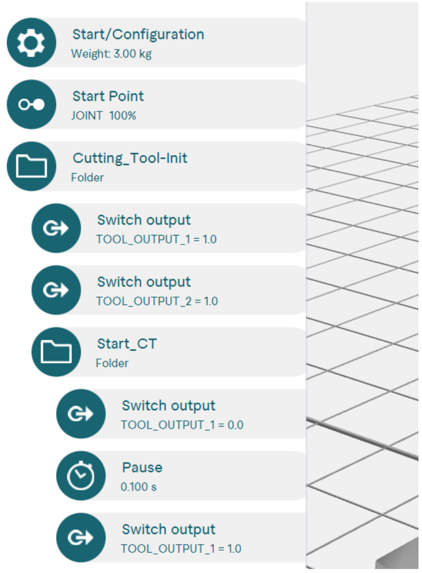

Figure 3.3: code 1

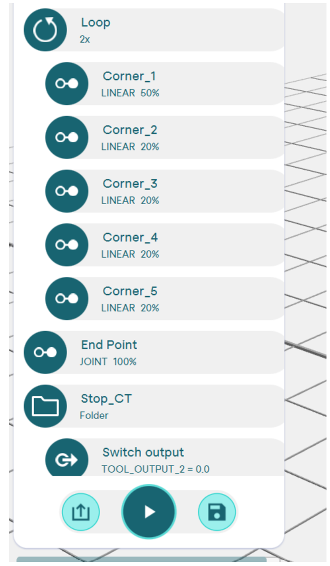

Figure 3.4: code 2

3.0.5 Illustration Example 2

Here we are simulating one of the most common use case for the robot in industry which is palletizing.The aim is to pick a group of items and drop them in a define location.

- Start/Configuration
  Amongst several settings that can be defined,we go to the "Tool & Co-ord"Tab.Here we set the gripper as the tool to be use in our program.We will approach all our waypoints with this tool,given that it is the only one that we need.under the "Weight"Tab the the maximum weight to be carried is already set as 3Kg and cannot be changed for the Horst600.Under the "3D world "window,we set our environment to "EducationPackage_L".We then apply the changes.

- Start Point
  Here we define the movement of the robot to a certain position or to a target point.We have the option to configure the velocity,type of movement, tolerances (looping) and target point of the robot.In our case we name our first waypoint "Start point".The velocity at which the waypoint is approached is set here to 100%.For the type of movement to the waypoint we chose joint instead of linear.The Target points are defined we the values:X=265 mm,Y=130mm and Z=100 mm.There are also options like " Switching outputs","Stop-conditions

" and " Set registers" that we did not touch because it was not needed in this case.

- Gripper Initialisation
  A folder was created in which we set 2 outputs to open and close the gripper.Those outputs are selected in a drop down list as:"TOOL_OUTPUT_1" ,"TOOL_OUTPUT_2" and are set as 1.0

- Opening of the Gripper
  Here we are suppose to open the Gripper before we approach the pallet.To do so we start by creating another folder.In that folder,We check first that the gripper is close by setting "TOOL_OUTPUT_1" with a value of "0.0".A delay of 100 ms is made.Then we set "TOOL_OUTPUT_1" to "1.0".

- Creation of a palette
  The pallet function is used to perform a repetitive action like picking multiple items in a define position.The pallet function comes already with some program blocks whose order cannot be changed.Those blocks are:"Start pallet ","End of pallet","Rel. approach point","Rel. palletizing point", "Rel. departure point" and "If condition." The "Rel. approach point" and "Rel. departure point" define the path travelled by the robot to go pick an item and also back to go drop.Those were set to be identical and as:X=265,Y=130,Z=100.The "Rel. palletizing point" is automatically calculated via the pallet definition.After the "Rel. palletizing point" we set "TOOL_OUTPUT_2" to a value of 1.0 after a delay of 100ms to close the gripper.Another waypoint with a name of "Drop Spot" was created right after "Rel. departure point" to specify where the picked items are suppose to be dropped.When the "Rel. departure point"is reached, we open the gripper again after another delay of 100ms.The "If condition." block was not used and remain untouched.The figure bellow illustrate some settings of the palette function.
  On the figures 3.5 and 3.6 bellow we can observe some settings of the palette functions as used in our code.

- Loop
  We create an 16 times loop that we place over the palette function.This tells the robot to do 16 palletizing pass to pick all items.The value of the loop can be adjusted according to the size of the pallet.

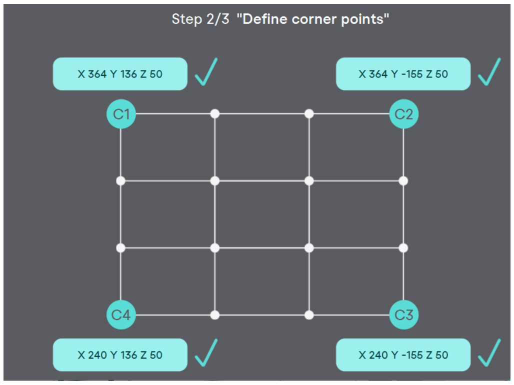

Figure 3.5: Palette Settings 1

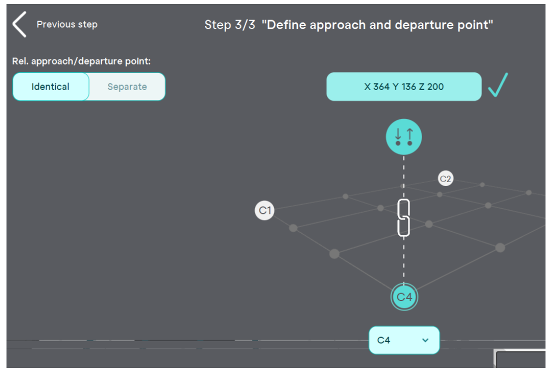

Figure 3.6: Palette Settings 2

3.0.6 Illustration Example 3

In this last example, we used the "Concatenation" function to combine spline paths.The aim was to simulate how we can use the robot to write the word "his".The graphical code follows the following steps.

- Start/Configuration
  After creating a project and giving it a name,We go to the "Tool & Co-ord"Tab.Here we set the gripping tool to be use in our program to "Zimmer_EducationPackage".Under the "Weight"Tab the maximum weight to be carried is already set as 3Kg and cannot be changed for the Horst600.Under the "3D world "window,we set our environment to "EMPTY_WORLD".We then apply the changes.

- Start Point
  Here we define from were the robot will start writting.The type of Movement is set as "joint", the velocity is kept at 100%.The coordinates are: X=386.54 mm,Y=132.69 mm,Z=73.64 mm.This waypoint is named "Start_Point"

- Concatenation
  Use of the "concatenation"function.Here we use the Basic instead of the Advanced option of this function.We set the path connection type to "Connect automatically".We also gave it the name "Concatenation 1"

- SplinePath 1,2 &3
  All this 3 different spline paths were used to write the letter "h".They are all set to be of type "CUBIC".Each of them also have 3 control points

- SplinePath 4&5
  They were used to write the letter "i".They are all set to be of type "CUBIC".Each of them also have 3 control points

- CirclePath 1
  This was used to write the letter "s".The circle type was set to type "Plane". The angle is "0°" and the orientation is "Fixed orientation".

On figure 3.7 we see the lines of the graphical code.fig 3.8 also shows the overview of the line define by Spline 1.

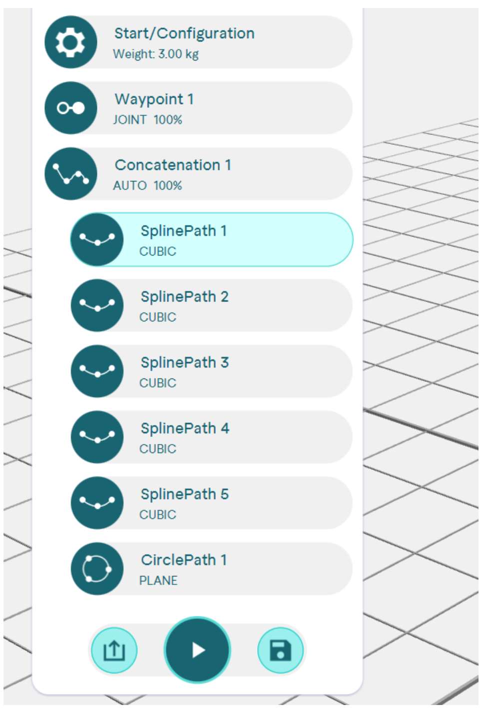

Figure 3.7: result 1

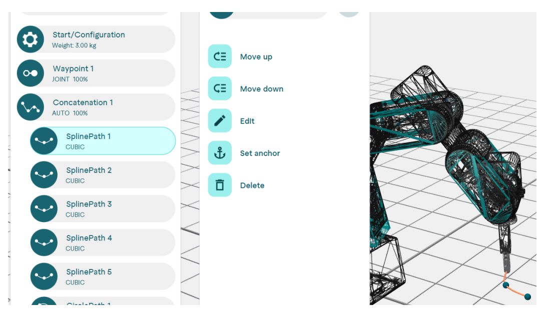

Figure 3.8: result 2

Chapter 4

RESULTS AND EVALUATION

4.0.1 Illustration Example 1

The first issue that we came across this case was the cutting tool. There was no appropriate cutting tool available in the horstFX software.Just for testing purpose we picked a random one that at least look like what we need.During the simulation although it is possible to see the twin robot moving across the 4 faces of the robot,it is not possible to see the tool spinning or the work piece being machined.To be able to choose all the different coordinates that the robot has to follow,we have to visually approach the work piece and save the corresponding coordinate.They can still be adjusted if needed while we do the real world testing on the robot itself.Figures 4.1 and 4.2 show some results of the simulation using the twin robot.On figure 4.1,the orange line shows the path followed to approach the first side of the work piece.On figure 4.2,we see the position of the robot as it reaches the 3rd side of the work piece.The gray side of the line shows the path that has already been covered and the orange is the one that is still to come.The initial idea that i had in this example was to simulate the robot going to the housing as shown in fig 4.1,picking the yellow object and dropping it in the red tray.The issue that i had was that based on the position of the robot on the work bench,it was never possible to reach that object.Beside that,I did not come across any major challenge in this example.

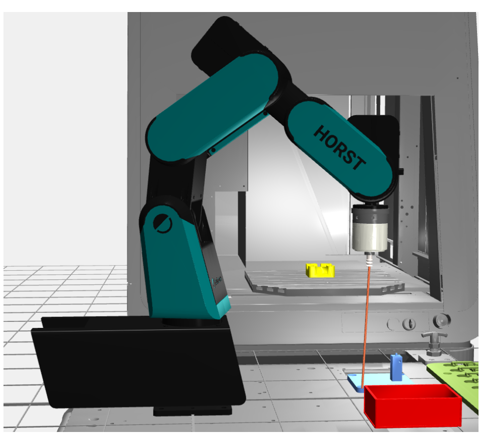

Figure 4.1: result 1

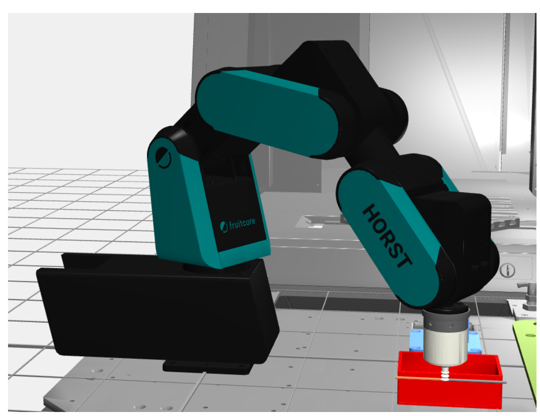

Figure 4.2: result 2

4.0.2 Illustration Example 2

Here we still have no possibility to see the gripper opening or closing.Due to the limitation of the horstFX software it is only possible to test that in real life..The first challenge that we faced was an error stating that the dimension of our palette was out of the workspace.To solve this problem. we redefine all 4 points of the palette by using rotation around the robot 1st axes and translation movement along X,Y coordinate with the value of Z remaining unchanged.The 2nd issue was the program crushing on the 12th palette point.This issue was also solved using the aforementioned solution related to the 1st error.After the 1st simulation trial,i was not satisfy with the position of the "Rel approach point" and "Rel departure point".To solve this issue i adjusted their Z values to be the same as that of the location where the object needed to be dropped.The figures bellow are the results of the simulation.Fig 4.6 shows the robot successfully cycle through all 16 points of the palette.

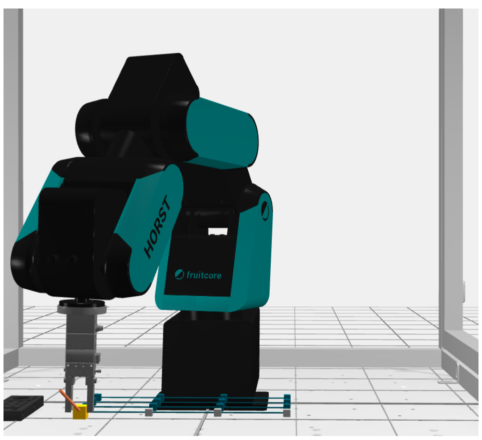

Figure 4.3: rel Palettizing Point

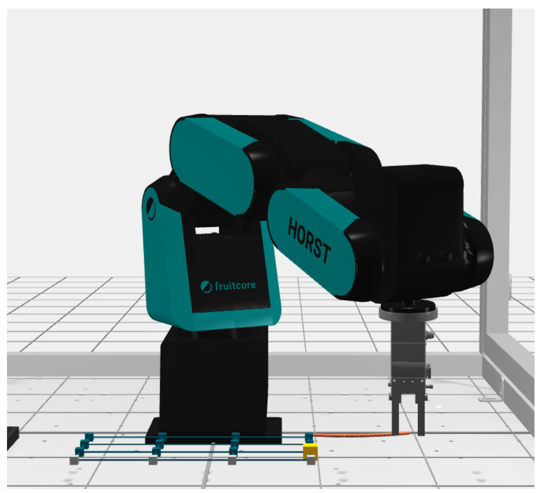

Figure 4.4: Drop spot

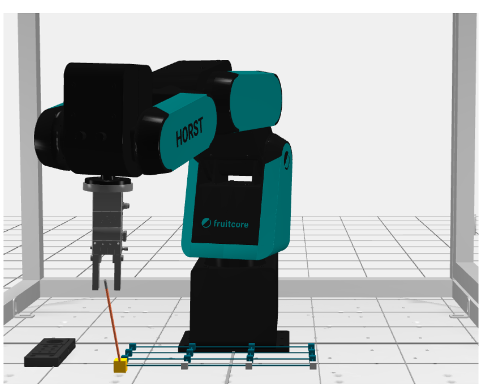

Figure 4.5: rel Dep & approach Point

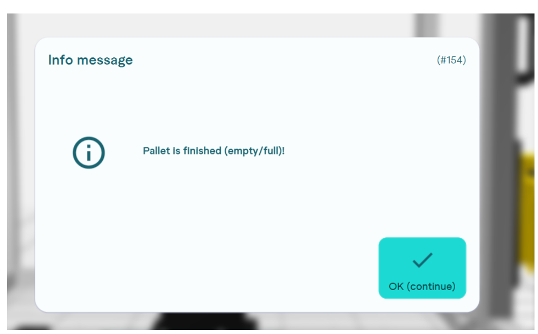

Figure 4.6: Palette completed

4.0.3 Illustration Example 3

After creating a spline path,i had to run to see if it write the part of the letter as intended.On Fig 4.7, we can see a highlight of all spline paths combined together.In Fog 4.8 we can see how during the writing process the robot runs through each letter.To make things easy in this task, i used as starting point of the next spline the end point of the previous one.Every time a spline path did not look as desired,i had to delete it and redefine all the 3 control points again.The letter "s" could also have been written using the Spline Path function. I used the Circle Path function instead to show that this could also have been done in another way.

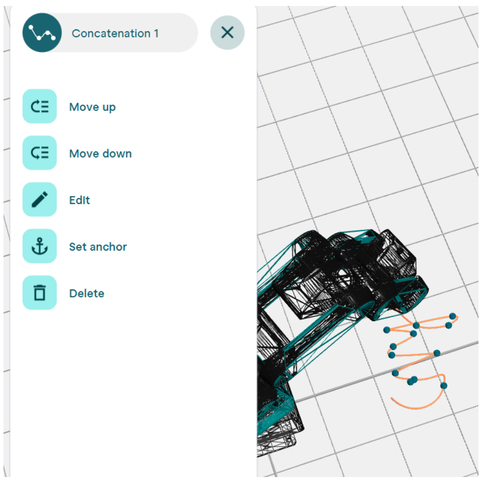

Figure 4.7: Overview of the Word "his"

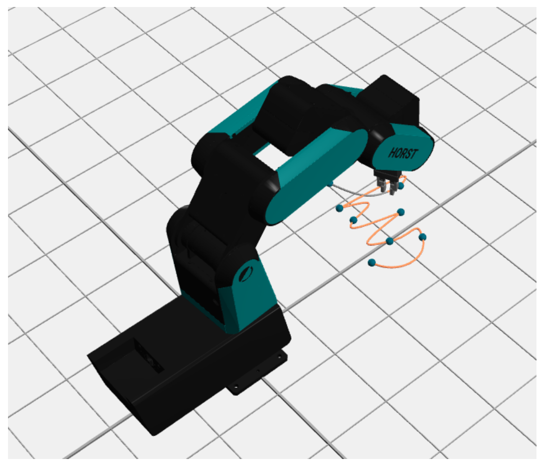

Figure 4.8: Robot running through each letter

Chapter 5

CONCLUSION

With the advancement in artificial intelligence,industrial robotic arm will be capable in the future to make decision in how to improve the way they complete a task.This knowledge could be shared via the cloud with other robots.There is also a trend toward cobot(collaborative robots),remote operation(operate on patient remotely),3D printing and additive manufacturing[6].For some future work we can explore the horstCOSMOS,which is the IOT-platform of Fruitcore robotic.Additionally we can connect more external devices like a PLC to our horst1000.However the lack of available training material or community(for exchange of information) related to programming the robot poses a major challenge related to complexity of a project we would like to do next.This is partly related to the fact that Fruitcore robotics is not very popular on the market for industrial robot arm.

Chapter 6

APPENDIX

6.0.1 Appendix A

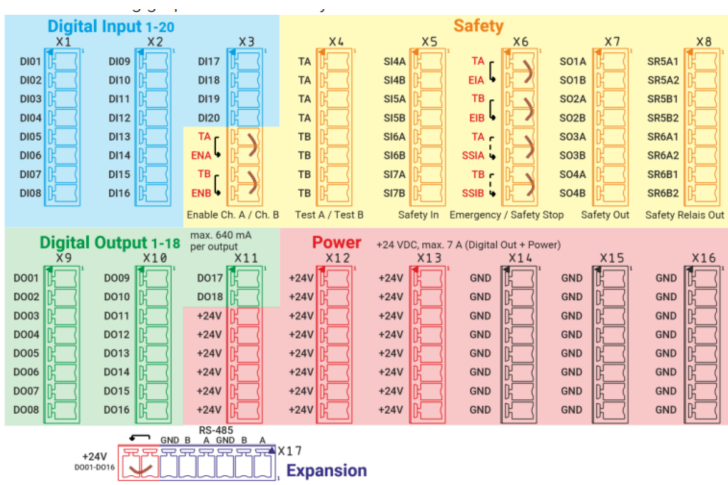

Figure 6.1: I/O Interface [3]

6.0.2 Appendix B

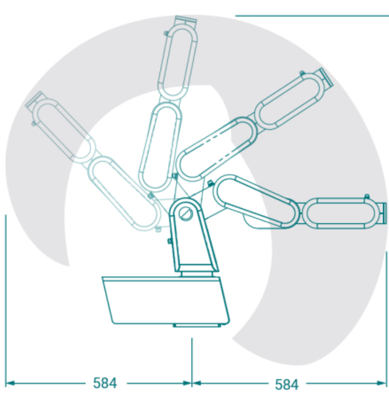

Figure 6.2: Side view of working area [3]

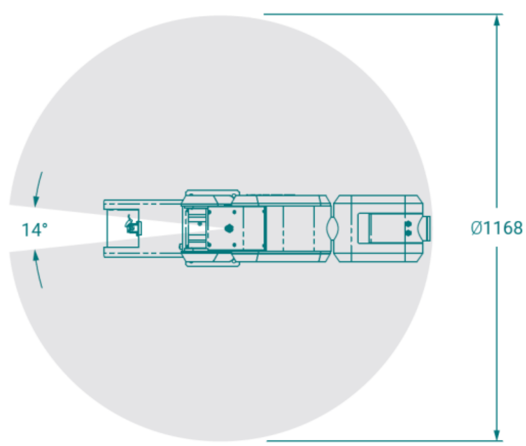

Figure 6.3: Top view of working area [3]

6.0.3 Appendix C

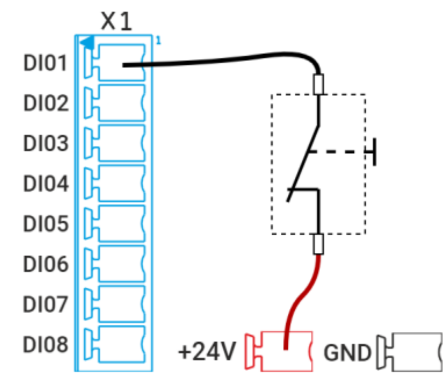

Figure 6.4: Digital Input Connection [3]

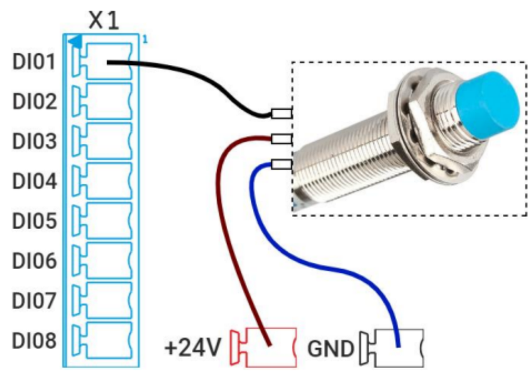

Figure 6.5: Sensor connection on DI [3]

6.0.4 Appendix D

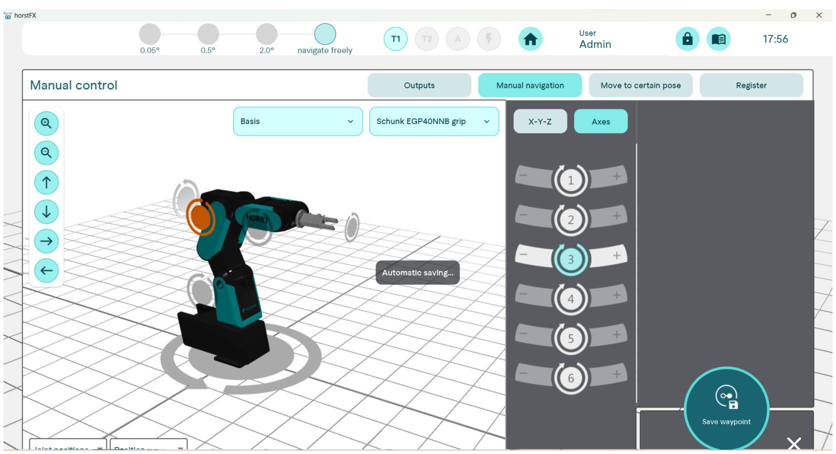

Figure 6.6: Control via robot axes

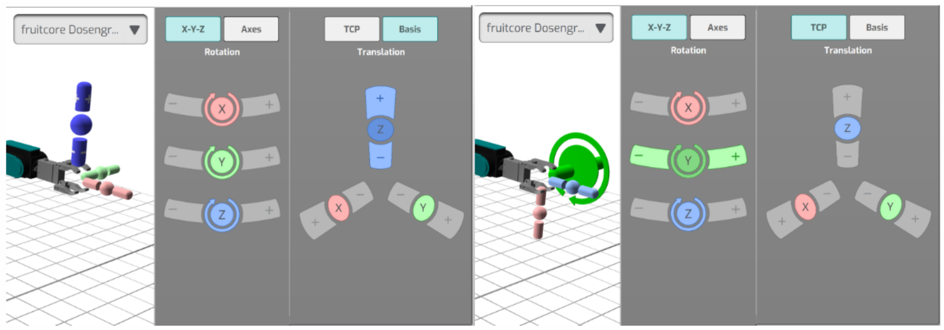

Figure 6.7: Control via coordinate values or orientation

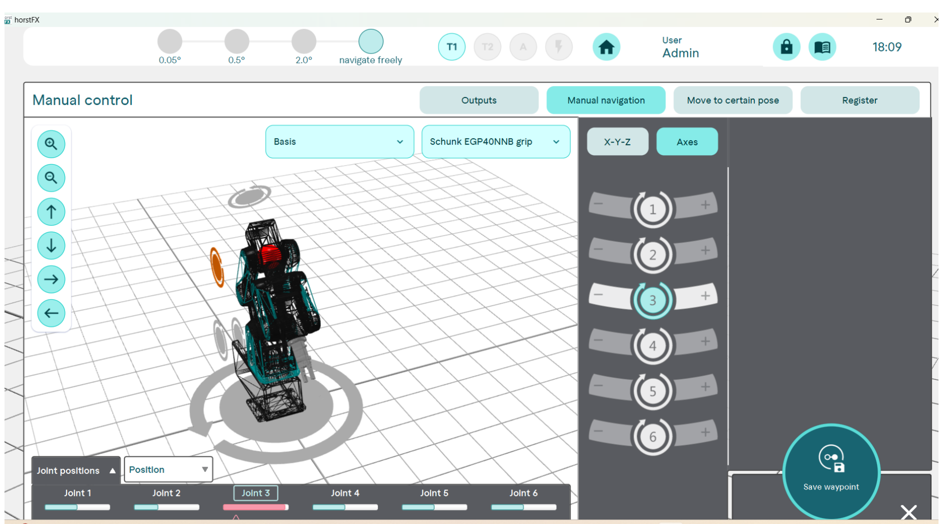

Figure 6.8: Robot Axes limits

Chapter 7

DECLARATION OF ORIGINALITY

I, Jires Donfack Voufo, herewith declare that I have composed the present paper and work by myself and without the use of any other than the cited sources and aids. Sentences or parts of sentences quoted literally are marked as such; other references with regard to the statement and scope are indicated by full details of the publications concerned. The paper and work in the same or similar form have not been submitted to any examination body and have not been published. This paper was not yet, even in part, used in another examination or as a course performance. I agree that my work may be checked by a plagiarism checker.

BIBLIOGRAPHY

[1] KUKA. The industrial robot: How does it work? *https://www.kuka.com/en-us/products/robotics-systems/good-to-know-industrial-robots*.

[2] Fruicore Robotics. Meilensteine mit horst: Die geschichte von fruitcore robotics. *https://www.fruitcore-robotics.com/unternehmen*.

[3] Fruicore Robotics. Horst1000 user manual. *https://horstcosmos.com/downloads/Betriebsanleitungen*, 2023.

[4] Fruitcore Robotics. Fruitcore flyer solution kit. *https://www.fruitcore-robotics.com/hubfs/Flyer/Brosch*

[5] UNIVERSAL ROBOTS. Robotic arm. https://www.universal-robots.com/in/blog/robotic-arm/*, 2022.*

[6] SMC. Robots: the next generation. https://industrie-wegweiser.de/von-industrie-1-0-bis-4-0-industrie-im-wandel-der-zeit/.

[7] Thomas. Von industrie 1.0 bis industrie 4.0-industrie im wandel der zeit. Von https://industrie-wegweiser.de/von-industrie-1-0-bis-4-0-industrie-im-wandel-der-zeit/*, 2014.*