Tallinn University of Technology

Department of Computer & Systems Engineering

Building a Digital Twin of YuMi Robot in Virtual Reality

MD Sakhawat Hossain 194274IASM

Master's Thesis

Supervisor:

Saleh Ragheb Saleh Alsaleh, MSc

Early-Stage Researcher

Co-supervisor:

Aleksei Tepljakov, PhD

Research Scientist

Tallinn 2021

Tallinna Tehnikaülikool

Arvuti- ja Süsteemitehnika Osakond

YuMi roboti digitaalse kaksiku loomine virtuaalreaalsuses

MD Sakhawat Hossain 194274IASM

Magistritöö

Juhendaja:

Saleh Ragheb Saleh Alsaleh, MSc

Teadur

Kaasjuhendaja:

Aleksei Tepljakov, PhD

Teadur

Tallinn 2021

Author’s Declaration of Originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature, and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: MD Sakhawat Hossain

10.05.2021

Acknowledgments

The author of this thesis would like to sincerely express his kind appreciation to his supervisor Saleh Ragheb Saleh Alsaleh, MSc, and co-supervisor Dr. Aleksei Tepljakov for their encouragement, responsiveness, instruction, and supervision throughout the whole thesis work.

This study was performed in Alpha Control Lab, which is within the Centre for Intelligent Systems, Department of Computer Systems, Tallinn University of Technology. My big appreciation goes to the people working in Alpha Control Lab for making things easier to learn.

List of Symbols and Abbreviations

API	Application Programming Interface
AR	Augmented Reality
AI	Artificial Intelligence
GPU	Graphics Processing Unit
IoT	Internet of Things
HMD	Head Mounted Display
MR	Mixed Reality
PC	Personal Computer
RE	Real Environment
VR	Virtual Reality
VW	Virtual World
CAD	Computer-aided design
UE	Unreal Engine
DOF	Degrees of Freedom
DH	Denavit–Hartenberg
IK	Inverse Kinematics
PS5	PlayStation 5
PLM	Product Lifecycle Management
IRC5	Industrial Robot Controller 5

TALLINN UNIVERSITY OF TECHNOLOGY

MASTER`S THESIS TASK SPECIFICATION

Date: 21/11/20

Student name: MD Sakhawat Hossain

Student code: 194274 IASM

Topic: Building a Digital Twin of YuMi Robot in Virtual Reality

Topic background: In the background of robotics pedagogy, digital twins (DT) of robotic processes can play a significant role in delivering efficacious learning experiences. Digital twins (DT) are seen as an environment, where the learning process can be simplified. Moreover, with the DT technology, the consumption and significance of virtual reality (VR) technology have been developing at a bourgeoning percentage and being functional to many other fields as it is rich with a lot of yet-to-be-determined possibilities. Administering virtual reality is an amalgamation of equally hardware and software. VR acknowledges us to experience the real world with a simulation of vision by creating an immersive 3D environment. It enables us to look throughout the virtual artificial reality, make real-time interaction movements around it. This procedure has already or being amalgamated in some fields mentioned below:

- Medical Training, Public Safety

- Retails and Logistics
- Military
- Repair and Maintenance
- Design and Modeling
- Media
- Game Engine based Development

The enthusiasm in the making of a digital twin (DT) of the YuMi (ABB) robot in virtual reality (VR) can allow us to develop a simple but efficient, safe, more human friendly and collaborative for end-users and various industrial use cases, as an example, one can create a virtual representation of YuMi robot model to gain safety, education, testing and training of the robot and make sure the conscientiousness for the manufacturing companies.

Supervisor: Saleh Ragheb Saleh Alsaleh, MSc, Early-Stage Researcher

Co-supervisor: Aleksei Tepljakov, PhD, Research Scientist

Additional specifications: By the end of this thesis project, it is expected that a system is developed to detect issues earlier on and portend the outcomes as precisely as possible for manufacturing companies like ABB, ENICS.

Issues to be resolved according to the following learning outcomes:

A. System aspects:

    a. implementation of the existing system to predict the consequences

    b. working on the process of making digital twin through AR user interface

c. trustable and cheap controlling system for any manufacturing company

B. Software aspects:

a. using accessible data and start processing it (chosen tools: Unreal Engine, SolidWorks, MATLAB)

C. Hardware aspects:

a. process of using virtual reality which includes a computer processor, GPS, Display, Microphone, controller

b. Virtual reality glasses (Microsoft Hololens 2/ Magic Leap One/ Vuzix Blade AR etc)

c. IoT

Exceptional conditions:

There is a possibility that my thesis project will not be completed fully as planned, specifically if further COVID19 rules are announced that prevent me to work directly on my project. Consequently, it can happen like:

a. I will need more time to complete my thesis project

b. My thesis will highlight all the research, findings, outcomes, and conclusions which was possible to make in between the given time frame

Student's signature: MD Sakhawat Hossain

(Digitally signed)

Abstract

In this paper, we introduce a part of a collaboration between industry and research area for the development of the YuMi robot by showing a user-friendly alternative. The necessity to build a physical prototype which is time-consuming and very costly at the same time, keeping this in mind, our goal is to provide a safe alternative environment for the YuMi robot by ensuring the maximized output and highest quality using digital twin (DT) technology in VR (virtual reality) user interface that can be used for the testing and training of the robot for industrial purposes. Additionally, this will help the manufacturing companies like ABB by providing real-time and predictive analytic that measure and optimize the operation of engineered systems and processes.

Keywords

Digital twin, Robotics, Collaborative robots, YuMi, Unreal Engine, Virtual Reality

The thesis is in English and contains 60 pages of text, 7 chapters, 30 figures, 3 tables.

Table of Contents

List of Figures

List of Tables

1. Introduction

The new era is on the crossover of a scientific transformation that will essentially change our associations with others and the way we live and work. Currently, processes and machines are so complex that the risks of failure or disruption from experimenting with different approaches become too high or unacceptable. To use an old analogy, it's difficult to change the wheels on a moving train. And that can be frustrating when new designs might provide significant benefits to existing systems. These new designs, their significance, possibility, and involvedness are unalike to what civilization has known until now. We cannot say surely what will happen, but one thing is certain: our reaction must involve all stakeholders at the global level: the public sector, the private sector, the academic world, and civil society. In our industry sector, we already have different detection systems, predictive maintenance, improved decision making in real-time. These constant improvements help to optimize the production tools.

Manufacturing robots are expended to accomplish identical tasks 24/7 during the day. Any mistake in programming will consequence in malfunctioning products which is an enormous damage to the business. It needs extra attention to do the same task repeatedly for many hours with full accuracy. Robots in manufacturing, support creating jobs by reestablishing more manufacturing work [1]. Predetermining industrial robots encompasses mainly regulating the 3D settings of the workstation. The complete development of programming a manufacturing robot is finished through the testing and training process.

The modern era is covered with manufacturing technologies including artificial intelligence, advanced robotics, and the internet of things (IoT). It is very important for the manufacturing companies, industries with need of special precise worker to get the best production quality. Industrial collaborative robots are the best and easiest solution for this as it delivers the precise and reliable results. It is also beneficial because of its concentrated cycle times. An industrial robot boosts up the manufacturing processes by operating without any break throughout the day and night [2]. The rapidity and dependability of robots eventually decrease cycle time and amplifies the result very largely. Additionally, the safety of the workers can be maintained very easily with the help of an industrial robot. Some tasks are very risky for humans but as an industrial collaborative robot, it can do the task without any issue, especially when engineering must take place underneath antagonistic environments. All these exclusive advantages can be deeply developed by testing and training the industrial collaborative robot for manufacturing purposes.

Life without technology is powerless in today's progressive world. Technology, which delivers mechanisms to promote improvement, use, and information exchange, has as its main objective of making tasks easier and solving many problems of mankind. Technologies such as the Microsoft HoloLens or Oculus Rift propose the capability to communicate and acquire various holographical representativeness [28]. Doctors can understand how the aortal valve in a heart works, engineers can investigate how the movements of the joints of a robot arm are structured through virtual reality (VR) technology. Apprentices can discover topography not only by understanding or inspecting a cinematographic but also throughout being submerged in it. It gives a highly processed motional response to the whole understanding, and this is the way where students learn more holistically.

The present thesis provides a safe alternative environment of the YuMi robot by building a digital twin (DT) of it inside Unreal Engine (UE) which can be used later for testing and training of the robot for industrial purposes.

1.1. Chapters of Thesis Subject Development

The thesis is ordered in a way so that each subsequent part provides a description of a new aspect of development.

The first chapter describes the introduction of the thesis, background, problem statement, currently available system, and further development process that can be used for the thesis. It also includes the thesis outline.

The second chapter talks about the research gaps.

The third chapter describes the concept of mixed reality (MR), augmented reality (AR), virtual reality (VR), and their use in robot study including in our thesis project. Here also we characterize the state of the art, where we look at some of the existing technologies such as Digital Twin (DT), UE (Unreal Engine). It also holds the introduction of our desired robot YuMi, its working ranges from different angles, axes. Also, it represents the hypothesis of kinematics.

The fourth chapter characterizes the main motivation behind our work.

The fifth chapter describes the methodology what have been used during the thesis work and to achieve the desired result. It includes the transformation of the YuMi robot model from SolidWorks to Unreal Engine (UE), implementation of the blueprints in UE, setting up the movements, limitations of the arm comprising joints, grippers, fingers. Additionally, it introduces the controller T Flight Hotas X and illustrates the inverse kinematics (IK) of the YuMi robot. It also embodies the virtual reality environment with Oculus Rift.

The sixth chapter gives us an overview of the initial findings, implementations, and discussions of our thesis work.

The seventh chapter designates the conclusions, it summarizes everything that has been achieved throughout the development. Moreover, it gives us an overall outlook of the future procedures of this thesis work.

2. Identified Research Gaps

Presently, developments and technologies are enough complicated that the possibilities of malfunction or interruption from investigating with unlike methodologies result unexpected and unpleasant.

If we think about the old times when technology did not introduce us to the concept of virtual reality, it was difficult to make any transformation, modification on any system. It can be frustrating when innovative organizations might stipulate momentous reimbursements to presented systems.

The necessity to build a substantial prototype which is time overwhelming and expensive at the same time.

3. State of The Art

3.1. Mixed Reality (MR)

Teaching and learning with machinery, until we got introduced to mixed reality, was mainly limited to complementary collaboration tools for communication such as learning management systems and electronic texts [22]. The equipment was an automatic aid to old-fashioned educational training, complementing a little up-to-the-minute muscle to the same ways we have communicated and absorbed for decades. We are now experiencing a modern era that is full of technologies and where machinery does the training, stipulating an entirely new apparatus for learning.

Mixed Reality (MR) has a rising character in the manufacturing background. MR is an amalgamation of real and virtual worlds obtainable together within a single display, which makes it exceedingly advantageous for the industry especially in those areas where the pre-visualization of the operation plays a critical role in the anticipation of hardships [23]. Within the engineering opportunity, MR has various applications.

In MR, spectators are not inadequate to perceiving only what exactly the developer or the scriptwriter wants us to watch. Rather than that, usually, it can be concentrating on their visualization in 360 degrees transversely a complete astronomical and in every track [27]. We can go wherever we want; it can take us to the moon in a second and by the touch of a click we can start walking in Rome. Especially, when we talk about our YuMi robot then we can see the robot's position, arms movements in high-definition quality through MR. To give an example; it is also possible to be acknowledged by the help of MR to see a beating heart as blood flows through it. Hundreds of years ago, we absorbed momentous settings like the Roman Colosseum by understanding a textbook with a 2x2-inch pixilated picture. There was no other way rather than to imagine how space looked like from our planet earth. Today we can easily watch videos, see movies, share understandings with the help of AR/VR/MR (Figure 1). This modern technology allows several students to work collaboratively on a holographical project, seeing and cooperating from innumerable vintage themes.

The Real Environment (RE) represents exactly what its name says, so humans can cooperate with it without any complexity where the Virtual Reality (VR) is a 3D computer engendered world where humans can interact with it with the help of hand controllers. Irreversibly, with Mixed Reality (MR), we can play a virtual video game, grab our real-world water bottle, and smack an imaginary character from the game with the bottle. This makes it very easier for humans to interact with computer-generated information virtually. An exceptional case of MR is Augmented Reality (AR) which lets people superimpose digital content over a real-world environment.

3.2. Augmented Reality (AR)

Augmented reality (AR) is a piece of innovative machinery that includes the intersection of computer illustrations of the actual world. Nowadays the technologies are getting more and more advanced and to be updated with them, we need to use the facilities provided by augmented reality to make sure a secure and beneficial future. The multifaceted nature of construction, manufacturing and industrialized composition progressively improves the requirement for the use of new alternative machinery.

To develop their manufacture and mechanization improvements; industries are affecting concerning the implementation of sophisticated machinery such as Digital Twin (DT), Augmented reality (AR), and Virtual reality (VR) [3]. The significance to gather these new technologies through the improvement of new systems, machinery, and performances that can expand the opportunity of manufacturing [4].

A classification of Augmented reality presentation is the assembly, maintenance, and repair of complex technology. Guidelines could be easier to appreciate if they were obtainable, not as instructions with text and pictures; but rather as 3D representations overlaid upon the concrete equipment, screening step-by-step responsibilities that need to be done and how to do them [5]. These overlaid 3D illustrations can be enthusiastic, making the guidelines even more clear.

When scheming an AR system, three characteristics need to be in concentration:

(1) Amalgamation of real and simulated worlds;

(2) Interactivity in real-time;

(3) Cataloguing in 3D.

3.3. Virtual Reality (VR)

Virtual Reality (VR) has been the “next big thing” for several years if we consider checking back our scientific discoveries and inventions, but fortunately, the time has conclusively come to produce realistic images, impacts, and impressions that put us in the middle of an outstanding fantasy world. Augmented Reality (AR), which adds virtual entities to our real-world environment, is subsidizing the thrill, and both AR, VR technologies should become a big part of our future. Imagination and reality have never been so amalgamated [26].

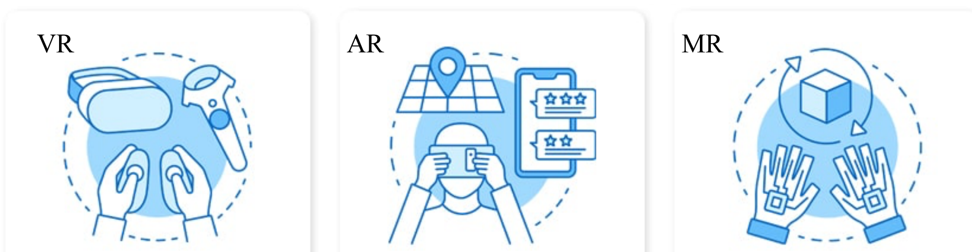


Figure 1: VR (Virtual Reality), AR (Augmented Reality), MR (Mixed Reality) [30].

Virtual reality is an appropriate instrument to absorb how to accomplish progressions without demanding an approach to the real workspace, being able to make the workers well trained and experienced without damaging the real machinery. VR gives the user an experience of a virtual environment, it is also used to teach the manipulator and accomplish reproductions of hypothetically false responsibilities. VR is the most extensively acknowledged out of all these (MR, AR, VR) technologies [29]. It is completely immersive, which tricks our brains into virtuality whether we are in a different atmosphere separately from the real world. Using a head-mounted display (HMD) or headset, we can easily get familiarized with a visual world that is full of imaginations, movies, and sounds where we can influence substances and move everywhere using actual controllers while connected to a PC, Xbox, or PS5 (PlayStation 5).

3.4. Digital Twin (DT)

The progress of our new generation which is based on technology requires robotics and automation. The future of such newborn technologies and equipment widely depends on the proper modeling of digital twin technology and by forecasting the possible outcomes based on the database. The process of modeling such technologies is called digital twin (DT). In general, digital twins are the virtual duplications of physical devices that can be used to determine the flexibility, restrictions, and limitations by running the simulations before the actual physical device is manufactured and implemented [6].

The scheme of a simulated digital technology to an extensive invention or digital twin (DT) was announced in 2002 by Dr. Grieves of the University of Michigan underneath the perception of Product Lifecycle Management (PLM) [7]. When he established his characterization, the realistic presentation of the knowledge of the Internet of Things (IoT) was in its beginning, and so the implementation persisted theoretically. New machinery was industrialized several years later. Therefore, we had to accept an unpredictable delay for years before entering a developed characterization. Since 2015, unique characterizations were correlated to DT to expand its involvement and to demonstrate the importance of this perception in a distinctive industrialized area. This way, DT helps to develop a prolonged designation in the framework of Industry 4.0 and the industrial Internet of Things (IoT) [8].

During the creation of a digital twin, the simulated model of an actual physical object remains the same, but it gets updated from time to time only when changes happen in the behavior according to the needs of the physical object. These advanced enthusiastically restructured simulated models can uninterruptedly perform the process of receiving new measurement statistics from instruments of the control device, from the environment, and from the whole system to manage the recently received as well as the past data also to transform parameters of the model.

The availability of the practice of digital twin is unbounded. Corporations can accomplish better understandings on consequence implementation and improve customer service, also can lower the risks of failure by ensuring improved functioning and calculated statements through the proper practicing of DT [9].

DT is supposed to be an up-to-date and detailed representation of all including elements of an actual physical device or object. Digital twins integrate machinery including artificial intelligence (AI), machine learning, extrapolative analytics, and sensor telemetry to generate digital twins of existent and linear implementation of substantial technologies and exaggerated digital replica that developed based on the information composed from real-world occurrences [10].

3.5. Collaborative Robot YuMi

A collaborative robot which can be called a Cobot as well is a robot in our modern times that has the power and necessary functionalities to learn real-time multiple tasks. The main motivation behind developing such a robot to make the assistant easier for humankind. These autonomous robots are built in such a way that they can continuously perform one task without any signs of failure, it can work independently and stay immobile because of their very rich user interface. Currently, collaborative robots are used in numerous perpendicular businesses, including engineering, manufacturing, quantity chain administration, and healthcare [11]. They normally have subordinate influence necessities than their enormous, independent equivalents, they are frequently transportable and use impact recognition to respond to an alarm in case of emergency or mechanical error to their human coworkers and other robots.

Figure 2: ABB YuMi Collaborative robot [12].

A Swiss-Swedish multinational corporation worldwide manufacturing company ABB successfully developed a collaborative robot called YuMi (Figure 2). ABB's YuMi robot which stands for You and Me is a twin arm robot full of innovative functionality which collaborates with humans, for example in small parts congregation, where humans and robots work together on the same tasks [12]. YuMi has achieved the goal of working together with humans and robots by removing both the physical and psychological barriers separating people from robots. The reason behind YuMi's worldwide popularity has been its friendly design that imitates the size and movements of a human operator, and its ultra-low weight including soft padded arms, which allow it to safely move faster compared to other cobots [49].

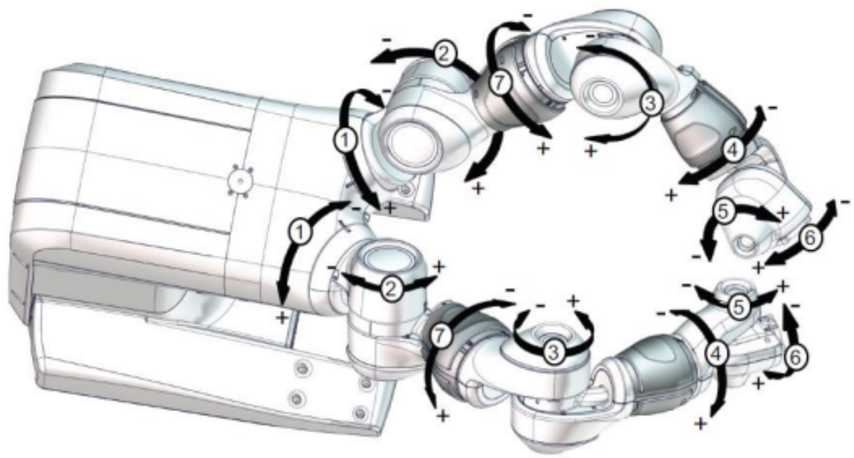


Figure 3: ABB YuMi Robot Axis overview [18].

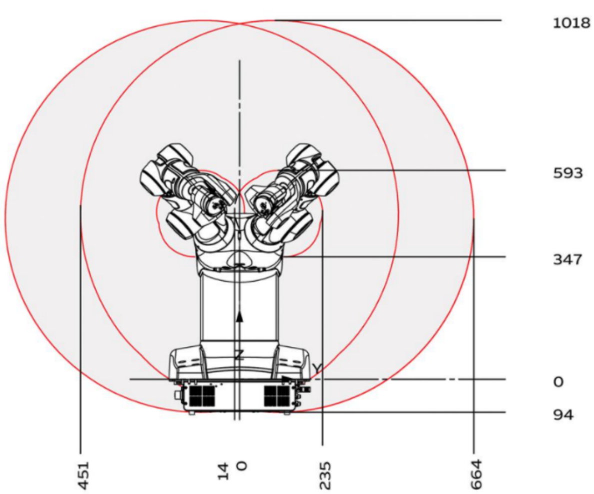
Figure 3 above, gives us a clear visual representation of the dual-arm 7 DOF (degree of freedom) of the modern collaborative robot YuMi. It is shown in the figure by indicating numbers from 1 to 7. Each DOF has a different working range including a unique axis maximum speed which can be seen from Table 1 below.

Axis movement	Working range	Axis max. speed
Axis 1 arm rotation	-168.5° to +168.5°	180°/s
Axis 2 arm bend	-143.5° to +43.5°	180°/s
Axis 7 arm rotation	-168.5° to +168.5°	180°/s
Axis 3 arm bend	-123.5° to +80°	180°/s
Axis 4 wrist rotation	-290° to +290°	400°/s
Axis 5 wrist bend	-88° to +138°	400°/s
Axis 6 flange rotation	-229° to +229°	400°/s

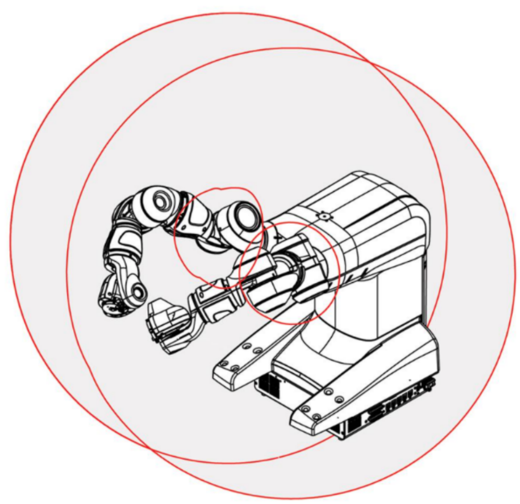
Table 1: ABB YuMi Robot Axis movement, the scale of work, and axis maximum speed [32].

3.5.1. YuMi Arm Working Range Views

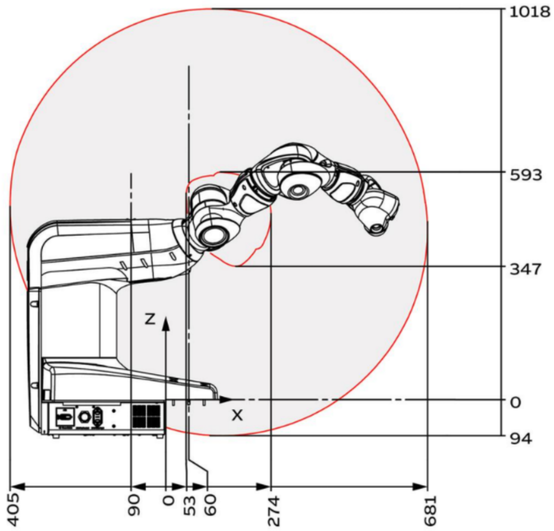
On its fundamental, YuMi is the safest compared to other collaborative robots [13]. YuMi arm is built similarly to a human arm. The human's arm has a skeleton sheltered with muscles, similarly, YuMi has lightweight magnesium alloy arms with soft padding to avoid pinch points. This function engages the arm in case of any unexpected events to a very high response [21]. Like the human arm, YuMi has no touchpoints so that delicate supplementary parts cannot be crumpled between two contrasting exteriors as the axes open and close. It has an easily understandable scheme, handy grippers, penetrating strength controller reaction, flexible software, and built-in protection structures that mutually tolerate software design through training rather than coding.



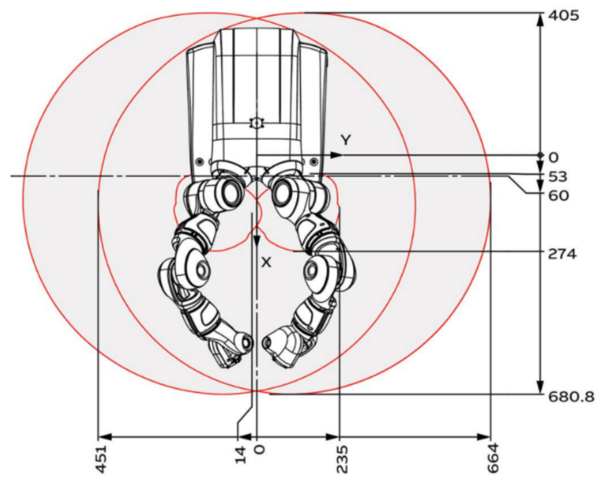
A-Front View



B-Isometric View



C-Side View



D-Top View

Figure 4: Representation of the YuMi robot from different angles [12].

The working range for each arm of the Yumi robot is shown in Figure 4. Additionally, in figure 4; A, B, C, and D represent a front view, isometric view, side view, and top view of YuMi correspondingly. Each arm has movements of approximately 664 mm along the Y-axis, 1018 mm along the Z-axis, and 681 mm along the X-axis.

3.6. Kinematics

Forces and torques are the Kinematics which are considered as the training of motion without reflecting the cause of the motion. There are two types of kinematics. Inverse kinematics is the practice of kinematic equations to control the motion of a robot to reach a preferred position [16]. For example, to perform automated assembly of the car parts or move the desired products into the specific location and position, to perform automated bin picking, robotic arms used in a manufacturing line what needs accurate motion from an initial position to the desired position between car parts, products, bins, and manufacturing machines. The acquisitive end of a robot arm is nominated as the end-effector. The robot confirmation is a list of joint positions that are within the position limits of the robot model and do not violate any constraints of the robot.

Forward kinematics and inverse kinematics are two elementary complications in robot kinematics. Forward kinematics which regulates the posture of the end-effector comparative to the orientation synchronize system rendering to the combined variables of the robot is comparatively informal, and its explanation is systematic, deterministic, and exceptional. Despite the fact, in inverse kinematics, to gain the combined variables from the posture of the end-effector is a composite system of nonlinear calculations with the robust connection of variables [17]. Therefore, it is a considerably more problematic system than forward kinematics.

As with all things, a little bit of arrangement goes a long way. Making sure that we have an idea of what we need in our IK setup to do. For our case, it is the robot's arm. During the process of performing a task, the different joints of the arm should move simultaneously that it makes the gripper properly hold things. Fortunately, with the power of Unreal Engine's Blueprint visual scripting, it is easier to add functionality to our YuMi robot model.

3.6.1. Inverse Kinematics

There are analytic, iterative, and geometrical methods to solve the inverse kinematics problem. It is shown in research papers how to solve the kinematics problem for a humanoid robot arm with a graphical method [39]. This method will be shown with a simplified example robot.

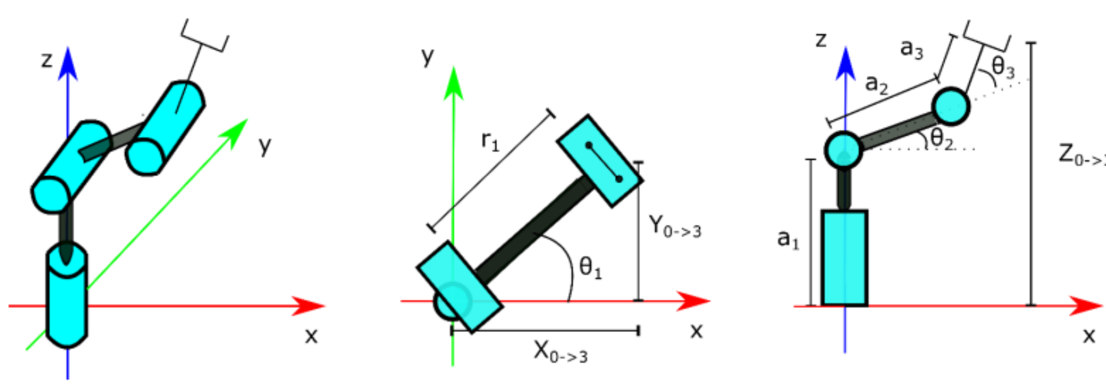


Figure 5: Example robot arm with 3 joints [40].

Figure 5 demonstrates the example robot from three different views. The top and side views are needed to determine the inverse kinematics solution. The top view can be used to describe the distance of the end effector from the origin. This example refers to [40].

$$\theta_1 = tan^{-1} \frac{y_3^0}{x_3^0} \tag{1}$$

$$r_1 = \sqrt{(x_3^0)^2 + (y_3^0)^2} \tag{2}$$

Figure 6 shows the insertion of the red rectangular triangle which connects the second joint with the end effector.

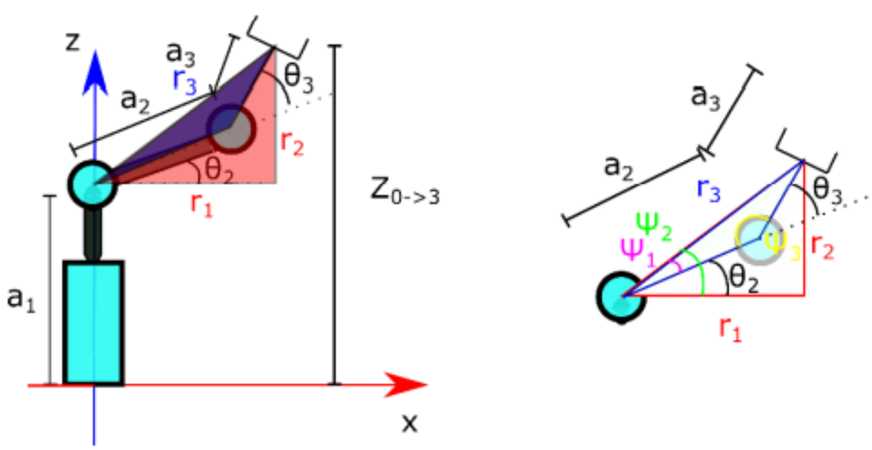


Figure 6: Parallelograms for interpretation [40].

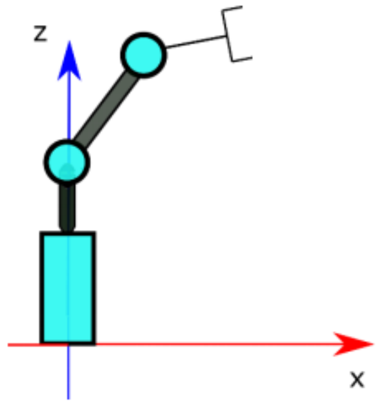


Figure 7: Replacement alignment [40].

$$r_2 = z_3^0 - a_1 \tag{3}$$

The distance, r_2 can be calculated by subtracting the distance, a_1 from the distance between the base and the end-effector frame in the z-axis direction. On the adjacent side, r_1 and the opposite side, r_2 the rectangular triangle angle, ψ_2 can be calculated.

$$\psi_2 = tan^{-1}(\frac{r_2}{r_1}) \tag{4}$$

$$r_3 = \sqrt{r_1^2 + r_2^2} \tag{5}$$

$$a_3^2 = a_2^2 + r_3^2 - 2a_2r_3.\cos(\psi_1) \tag{6}$$

$$\psi_1 = cos^{-1}(\frac{a_3^2 - a_2^2 - r_3^2}{-2a_2r_3}) \tag{7}$$

Equation (6), (7) shows the calculation of cosines and formula conversion from the angle ψ_1.

$$\theta_2 = \psi_2 - \psi_1 \tag{8}$$

$$\theta_3 = 180° - \psi_3 \tag{9}$$

$$r_3^2 = a_2^2 + a_3^2 - 2a_2a_3.\cos(\psi_3) \tag{10}$$

$$\psi_1 = cos^{-1}(\frac{r_3^2 - a_2^2 - a_3^2}{-2a_2a_3}) \tag{11}$$

The above-shown example is one of the ways to calculate the angles of the joints for a defined end-effector position. The perspectives which result from the derived formulas are for the cause of Figure 7. On the other hand, the objective arrangement can be achieved with another set of approaches for the joints as well. This is demonstrated in Figure 7. The difficulty of inverse kinematics grows with the number of joints due to increasing joint structure potentials. Especially as the YuMi robot model has 7 DOF (degrees of freedom) so in this case, it gets more complicated.

3.7. Unreal Engine (UE)

Unreal Engine (Figure 8), accompanying all prevailing systems including the next-generation consoles PlayStation 5 and Xbox Series X is a widespread and worldwide accepted game engine established by Epic Games [35]. The main motivation behind the creation of such engine was to ensure customizability, multiplatform capabilities, and the ability to create high-quality games for game developers. Games like battle royale shooter Fortnite or other standard modern AAA games like Psyonix's 'Rocket League' are mainly developed in UE. Developing any project in Unreal Engine is very straightforward and simple for beginners and educational purposes. Students can easily get familiar with UE with few instructions. The main development functionalities of this modern engine are the blueprints. By using the Blueprints Visual Scripting system, one can generate or build the entire project (in our case the YuMi robot model) without writing a single line of code. Because of the combination with an easy-to-use user interface, one can briefly build a model and start the necessary or desired developments.

Figure 8: Name and the logo of Unreal Engine [37].

The intensification in attractiveness of VR (virtual reality) in recent years is not ignorable, therefore it is advantageous to acquire knowledge to an engine that can easily help us to construct projects, games, and other necessary contents for the platforms incorporating Oculus Rift.

The main user interface of UE consists of [36]:

Content Browser: This section contains a folder structure that displays all our project files. Because of the great functionalities of UE, by default, it has a lot of built-in folders which is indeed very helpful when it comes to creating a new project.

World Outliner: It represents all the items in the current scene. One has the ability to toggle the appearance of each item as well as to create groups and folders to organize the world outliner as necessary.

Modes: This panel lets us choose between implements such as the Landscape Tool and the Foliage Tool. The Place Tool is the nonappearance tool. It acknowledges us to move and place different types of objects into our level such as lights, cameras, characters.

Details: Any object we select from the world outliner will have its detailed properties demonstrated here. Predominantly, this panel is used mostly to edit the settings of the object. Transformations created will only imitate that occurrence of the object.

Viewport: This is the view of our level. We can look around by holding right-click and moving the mouse.

Blueprint Class: A Blueprint Class, regularly abbreviated as Blueprint, is a strength that acknowledges users to straightforwardly add functionality and property on top of presented gameplay classes. In the content, package blueprints are generated inside of Unreal Editor and saved as assets visually. These fundamentally explain a new class or type of Actor which can then be positioned into maps as illustrations that perform like any other type of Actor.

Parent Classes: In Table 2, we can see the most common Parent Classes used when creating a new Blueprint [37].

Class Type	**Description**
Actor	An object that can be positioned or reproduced in the world
Pawn	A Pawn is an Actor that can be "controlled" and obtain a response from a Controller
Character	It’s a Pawn that incorporates the capability to walk, run, jump, and more
Player Controller	A Player Controller is an Actor conscientious for monitoring a Pawn expended by the user
Game Mode	A Game Mode delineates the game being performed, its documentation, recording, and other appearances of the game type

Table 2: Parent classes and their roles.

4. Motivation

Virtual reality (VT) and digital twin (DT) technology produce analytical extensions and clarifications for manufacturing industries [20]. These new technologies help manufacturing companies to maintain the development process. This thesis is an introductory involvement of that development process. Digital twin technology in the virtual reality user interface can give a clear overview regarding controlling the desired product also the testing and development of it which is in our case the collaborative robot YuMi [14]. The existing system of the YuMi robot is well developed but to improve its controllability, creativity, speed, and performance we need to make some development in its behavior and system. To do that, the real-time testing by just building or adding something into the robot is very risky as if it does not suit the robot or can't match the internal system with the newly introduced testing environment then the consequences could be worse. It can destroy the whole system of the robot and we end up with nothing. So, this can easily be avoided with the help of digital twin technology where we do not risk our product (YuMi) and at the same time, it's cheaper to perform. Updating a new system or any new development of the system can be tested to make sure the maximized output and highest quality by optimizing the existing system in YuMi through virtual reality. Additionally, manufacturing companies will get the best-trusted results before starting the production from real-time developments and predictive analytics that measure and enhance the operation of engineered systems and processes. This will also include the detection of issues earlier on and portend the outcomes as precisely as possible for manufacturing companies.

5. Methodology

This thesis is based on practical approach and the proposed goal. In scope of the present work, much attention is paid to development of creating an alternative environment. In addition, a certain amount of analysis of underlying concept is included. The main components of the developed system can be seen below (figure 9).

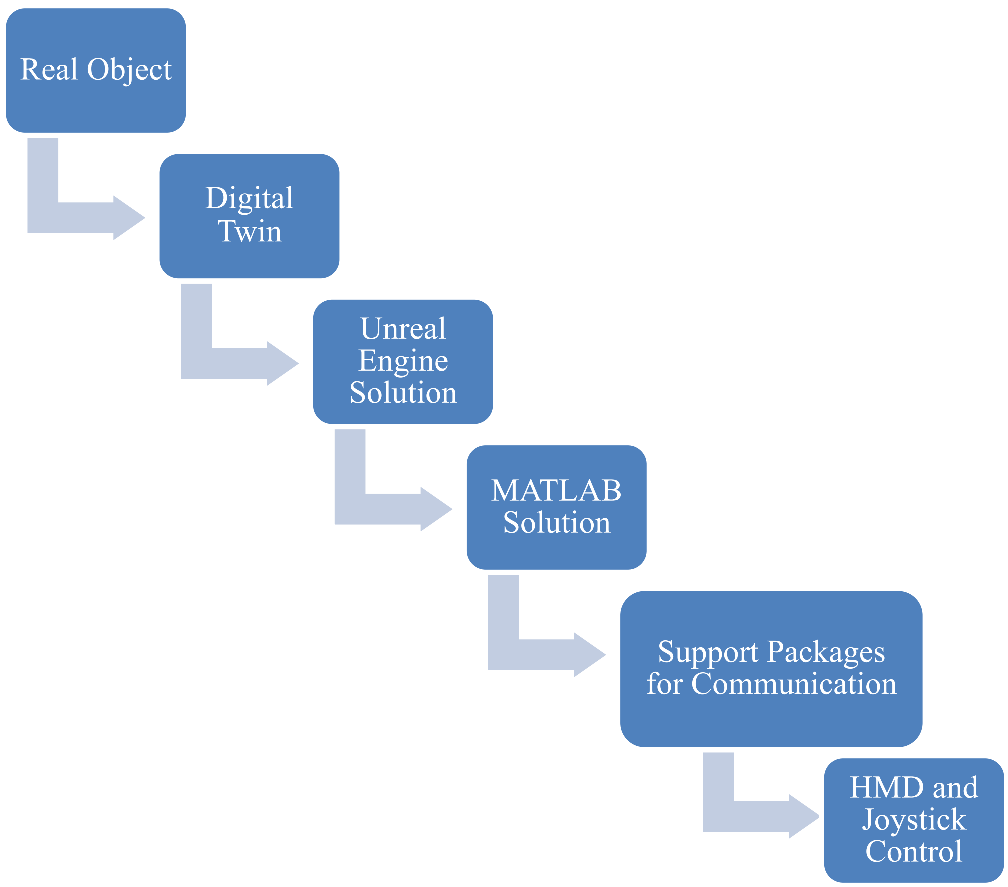


Figure 9: Overview of the developed system.

5.1. 3D Modelling of the Collaborative Robot

The sensitivity of robot movement plays a huge role in building a collaborative robot. The special word collaborative stands for working together simultaneously with the new updated technologies which result in the collaboration of humans and robots. As of preliminary statues, the research of making the 3D modeling of the YuMi collaborative robot has been developed and it is finalized to use SolidWorks 3D CAD (computer-aided design).

5.1.1. Initial Modelling with SolidWorks 3D CAD (Computer-Aided Design)

SolidWorks is a computer-aided design (CAD) based software that uses the theory of parametric construct and engenders three kinds of interlocked files: the component, the fabrication, and the sketching [46]. Solidworks 3D CAD delivers an easier way for engineers, designers, and manufacturers with the design, simulation, and manufacturing. It is influential and industry-proven [47].

We started with the existing 3D Model of our desired YuMi robot [48]. The present model of the robot needed to configure which has been done in SolidWorks. The sketch consists of geometries and materials. Where geometries give us all the necessary components such as parts, joints, fingers, grippers.

5.1.2. Plugin to connect to Unreal Engine

The transformation of our SolidWorks 3D CAD model has been achieved through using a plugin called DATASMITH which takes the ready preconfigured SolidWorks files and then converts them into Unreal Engine. Figure 10 illustrates a clear concept of this process.

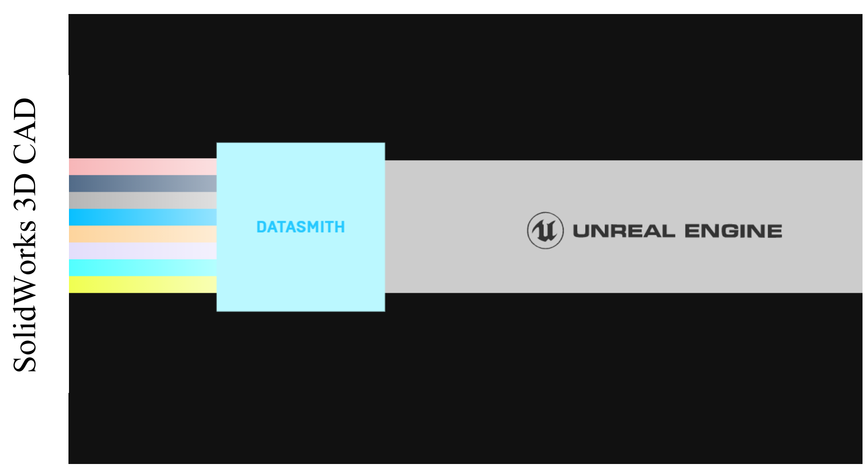


Figure 10: Conversion of the 3D CAD model to UE through DATASMITH plugin.

5.1.3. Unreal Engine Modelling

As our main goal is to optimize the controlling of the DT of the YuMi model, UE is the best choice to start with the designing and modeling of the robot. Our initially developed YuMi robot model inside SolidWorks 3D CAD has been now imported to UE for further implementations.

Regarding the starting approach of making the SolidWorks model working, we had to change the origins of each component, therefore, it was performed in SolidWorks after that the completed model has been introduced into Unreal Engine through the datasmith plugin. However, SolidWorks and UE (Unreal Engine) are not using the same way of illustrating coordinates. So, when we select a part in UE, we can just rotate it only around the X-axis. It is not necessary to change any other variables.

Unreal engine is made up of different types of components which make sure the creation will deliver the best result [19]. Controlling the YuMi 3D model in UE gets easier because of the enormous system tools and editors that allow us to systematize the parts of the robot and control them according to the requirements.

5.2. Unreal Engine Mappings

Unreal provides two methods to create key bindings [24]:

Action Mapping: These can only be in two states: pressed or not pressed. Action events will only trigger once when it is pressed or released the key. It is mainly used for actions that do not have an in-between state, such as firing a gun but in this thesis work also Action mapping has been used largely to move around different parts of the YuMi robot arm.

Axis Mapping: These outputs a numerical value called an axis value. Axis occurrences will fire every frame. Conventionally used for actions that require a thumbstick or mouse. Regarding the movement of the fingers of the YuMi robot model, axis mapping plays a huge role. As the gripper movement was controlled by action mapping and to match with the fingers which are based on the location transform, axis mapping has been used for controlling the fingers. Here to control the fingers 'T Flight Hotas X' used.

An axis value is a mathematical value that is concluded by the type of input and how we use it. Buttons and keys output 1 when pressed. Thumbsticks output a value between -1 and 1 depending on the direction and how far we push it (Figure 11).

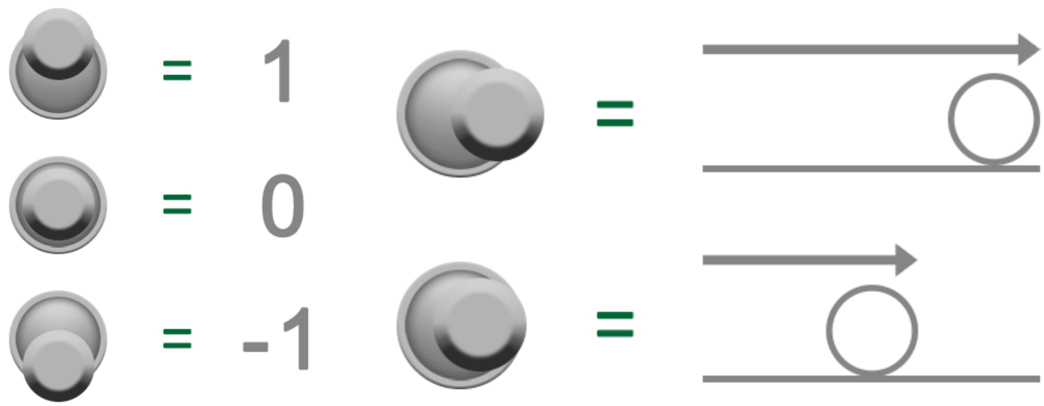


Figure 11: Thumbstick functionality.

We can use the axis value to control a Pawn's speed. If we push the thumbstick to the edge, the axis value will be 1. If we push it halfway, it will be 0.5. By multiplying the axis value with a speed variable, we can adjust the speed with the thumbstick (Figure 11).

5.2.1. Blueprints UE

Most of the functionality is developed using blueprints which is a flexible visual scripting system inside UE. Configuration of movement of the arm base rotation, different parts of the arm rotation, grippers, fingers are configured and structured through different functionalities using the available built-in functions of UE. Here in Figure 12 and Figure 13, it is seen the implementation of blueprints for controlling the left arm of the YuMi robot model. Additionally, to control the right arm, parallel functioning has been used.

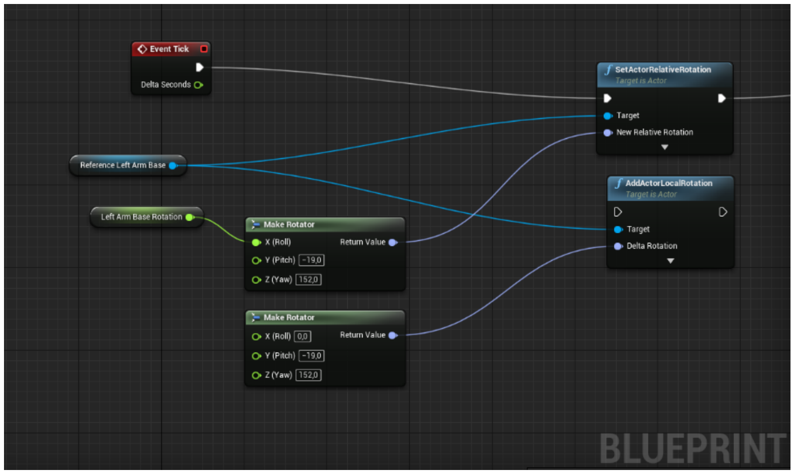


Figure 12: Left-arm base rotation blueprint.

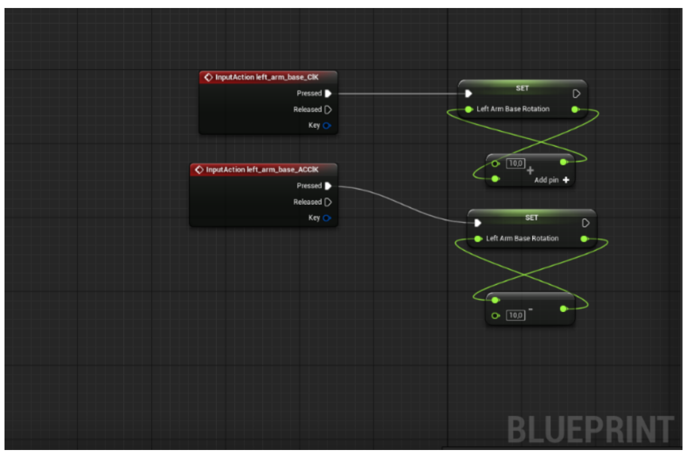


Figure 13: Left-arm base rotation controller blueprint.

Other actors and their blueprints can be seen from the Appendix.

5.3. T Flight Hotas X

T flight hotas X (Figure 14) has the throttle and joystick that are entirely made of plastic, though the bases feel strong and weighty enough not to move mid-flight. A single USB cable hooks up the HOTAS to a PC or PlayStation 3, and the throttle and joystick can be functioned apart or substantially associated, which is accomplished by a straightforward button and fold screw correlation [43]. It is possible also if someone prefers them pulled together, while others may choose a more natural experience. Thrustmaster uses a sound-cable supervision system beneath the throttle unit that keeps everything looking tidy.

Figure 14: T Flight Hotas X.

When we talk about the throttle, there is a single axis with a central neutral setting. A total of four thumb buttons are located on the front, with a further three situated on the base. On the tail of the throttle, there are two interrelated shoulder triggers and additional two buttons. It's possible to operate frequent functions because of its ergonomic layout.

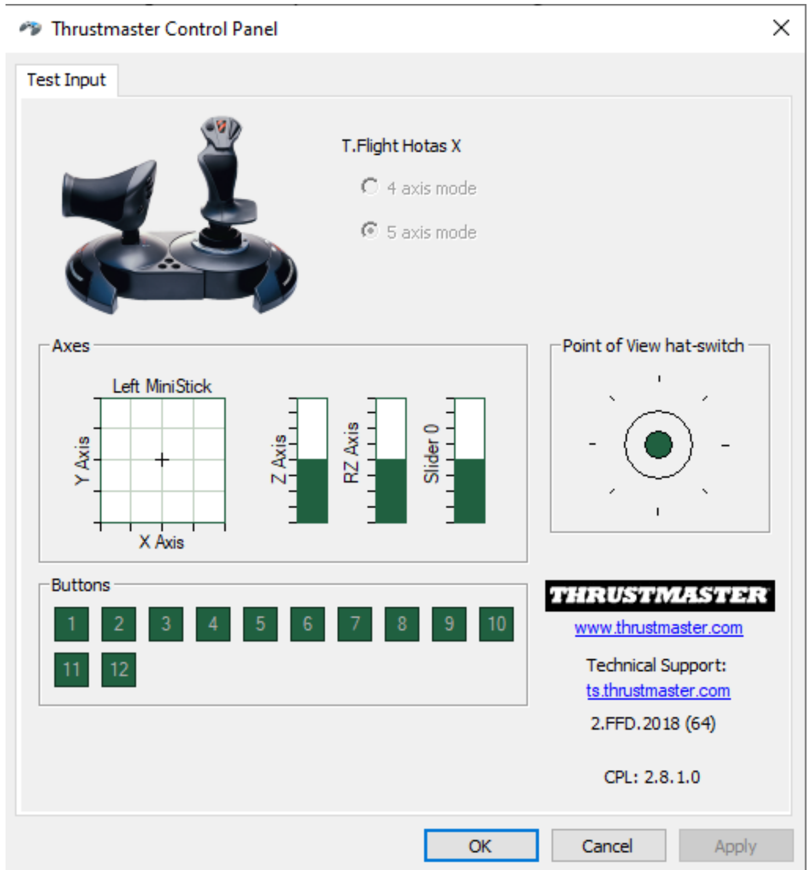


Figure 15: Controlling areas of Hotas X.

Figure 15 indicates the working areas including the buttons and axis. The joystick is largely useful regarding our project as it has some features that include a third axis achieved by twisting. A single spark is available on the rear, alongside a secondary knob and at the same time, additional two buttons are located on the front, next to a thumb pad. Together, there are 12 buttons in total that are completely configurable, and achieving full use of supplementary transfer facilities, it's conceivable to map a good 30 or so functions.

In our case we have not used all the functions of this controller, only to control the fingers of our desired YuMi robot model, the Axes functionalities of Hotas X were used.

5.4. Inverse Kinematics of the YuMi Robot

Robotic Arm Kinematics: YuMi is the first safe collaborative industrial robot with two 7 DOF (Degree of Freedom) arms. This Dual-arm ABB robot is ideal for assembling small parts with a load capacity of 500g for each arm [32]. It is controlled with the ABB IRC5 (Industrial Robot Controller 5) control system. The two arms of the robot are the same manipulator of 7 DOF (Degree of freedom) with two different bases. For this reason, it is sufficient to identify the parameters of a single arm. Unlike other dual-arm robots, each arm of YuMi has 7 DOF with joint offsets which enhance the creativity of the robot and consequently very appropriate for small part assembly.

In the first step of defining the frames for each joint, we used the DH (Denavit-Hartenberg) parameter [31]. This is shown in Figure 16. The corresponding Denavit-Hartenberg (DH) parameters of each axis are presented in Table 3. The inverse kinematic (IK) therefore can be realized.

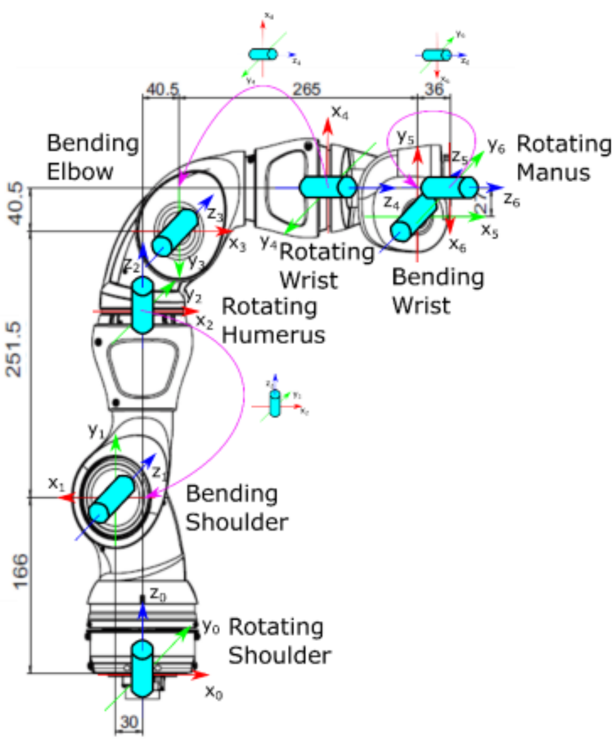


Figure 16: Frames of the Joints [31].

i	θ_i	d_i/mm	a_i/m	α_i
1	θ_1	166	30	$-\pi/2$
2	$\pi+\theta_2$	0	30	$-\pi/2$
3	θ_3	251.5	40.5	$\pi/2$
4	$\pi+\theta_4$	0	40.5	$\pi/2$
5	$\pi+\theta_5$	265	27	$\pi/2$
6	$\pi+\theta_6$	0	27	$\pi/2$
7	θ_7	120	0	0

Table 3: DH parameters of YuMi Dual-arm robot [31].

The inverse kinematics (IK) problem for a robot manipulator is defined as the problem of obtaining the values of the joint angles for a given position and orientation of the end-effector and the values of the geometric link parameters. The IK technique is widely used for manipulating robot manipulators. It has a finite number of solutions if the number of degrees of freedom (DOF) of the arm is adequate to accomplish a task in its accessible space. If the sum of DOF of the manipulator is less than the expected to accomplish a task by the end-effector, then the solutions are not available. Nevertheless, if the quantity of DOF of the manipulators is bigger than the mandatory to accomplish a task such as for the YuMi robot model in this project, then there is a solution.

5.5. Oculus Rift

Oculus Rift (Figure 17) is a VR (virtual reality) Ski-Masked Designed Goggle device that works along with computers or mobiles. Other VR headsets have the problem of gesticulation conditions. But here is a new technology which entitlements to have solved this problem of motion condition and lightheadedness post the usage [33]. This new equipment is Oculus Rift. Regarding the VR environment, Oculus Rift plays a very important role as it gives the feel of being in the 3D virtual world. It is a headset for users to wear on the head with fastening onto it to step into a virtual artificial world. Oculus VR the makers of the oculus rift headset originally funded as a Kickstarter project in 2012 and engineered with the help of John Carlmark became the leader company of VR for video games company [34].

Figure 17: Oculus Rift [25].

As we can see from Figure 18 the interior construction of the Oculus Rift consists of a cover, circuit board, HD (high definition) display, dial functionalities that gives us to control the display to move back and forth, lenses that come in different sizes for the special persons who are nearsighted and a foam padding for the comfort.

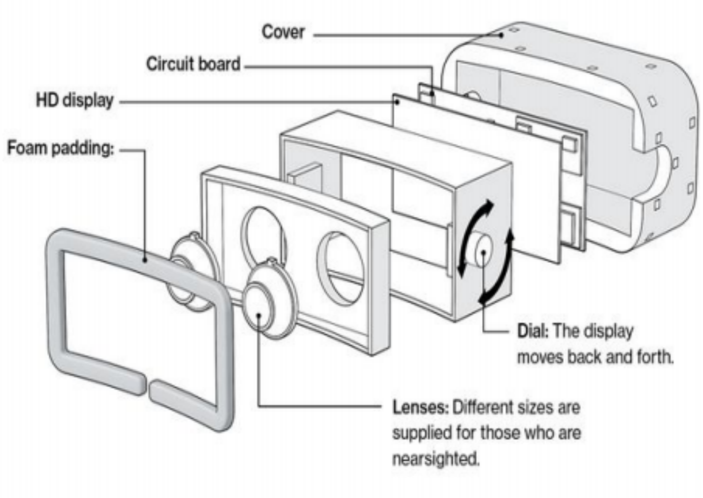


Figure 18: The internal structure of the oculus rift headset [45].

For our thesis work regarding controlling the digital twin of the YuMi robot model in UE through VR technology using Oculus Rift and its initial connection has been made in a general way. That means we used the built-in HDMI cable, USB cable, and Headset cable.

5.5.1. Connection Between Oculus Rift and UE

To make the relationship between Unreal Engine and Oculus Rift, we used the Virtual Reality Unreal Engine default plugin StreamVR. After that necessary settings have been simplified to get it to work with the YuMi project. Figure 19 shows how the implementation happened in UE Blueprint.

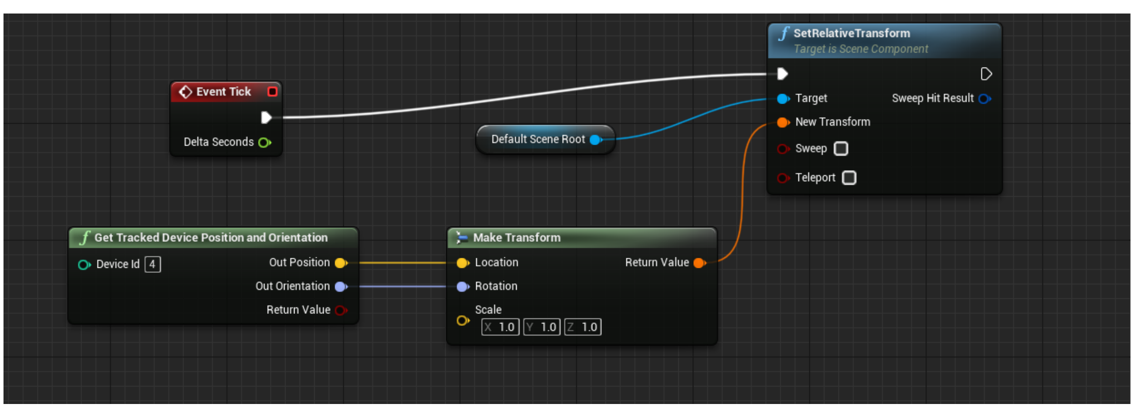


Figure 19: Setting Up Blueprint.

6. Implementations and Discussions

First, after creating the actors and blueprint class for them, different joints of the arm were successfully able to control but regarding the arm gripper and the fingers, to keep the fingers inside the gripper and make the axis movement limited we needed to use a different way of input settings. For the actors, Left_Arm_Base, L1, L2, L3, L4, L5, L6, Left_Gripper_Base, L_finger_L, and L_finger_R reference blueprint class (Appendix 1) have been created and put into the World Outliner to implement the blueprint functionalities using Event graph.

6.1. SetActorRelativeRotation Function

Figure 20 is demonstrating the Unreal Engine interface with the complete 3D model of the YuMi robot including fully controlled movements of the arm, joints, gripper, and fingers.

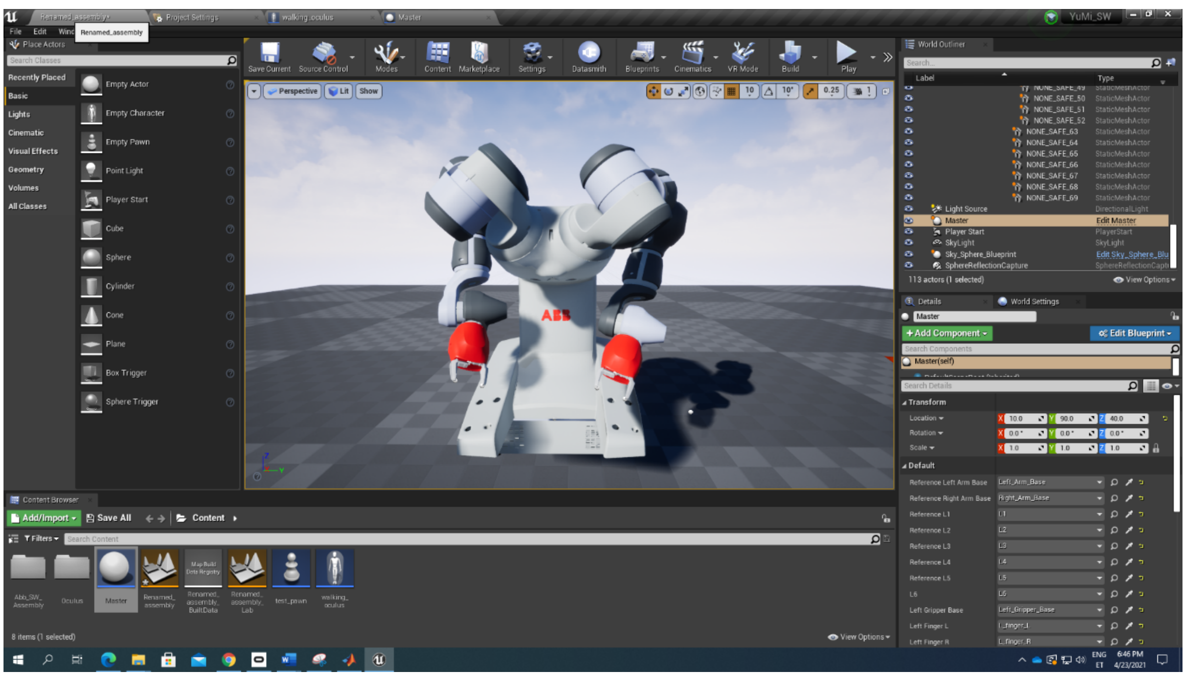


Figure 20: YuMi inside UE.

To control separate joints of the arm two main functions have been used *SetActorRelativeRotation* and *AddActorLocalRotation*. *SetActorRelativeRotation* takes an actor after that sets the relative rotation on the actor to its parent. In our blueprint class (Appendix 1), each joint of the arm is set as an actor therefore relative rotation on them towards their parent worked successfully. As relative is comparing to its parent what exactly is going to be the rotation. Subsequently, if the parent has a rotation of nothing and we declare that the child or Target rotates itself by 10 degrees on the X-axis then it will end up rotating to 10 degrees on the X-axis. But the function *SetActorRelativeRotation* is worthy to use especially in the case when the parent may change. So, if I set the rotation of 20 degrees on the X-axis then, it is always going to be 20 degrees on the X-axis based on the World. On the other hand, when I set the rotation based on relative then it does not matter where I move my actor to, it is going to be relative to its parent.

6.2. AddActorLocalRotation Function

AddActorLocalRotation also works mostly similar to our previous function but here additionally it takes a delta rotation. Here we are adding the rotation to our actor not setting like our previous function. Moreover, we are doing it based on the local coordinate space.

If we talk about one of our actors such Left_Arm_Base where we used *AddActorLocalRotation* and *Make Rotator* returns value to the local rotator's delta rotation. Here in the *Make Rotator,* we used the Roll, Pitch, and Yaw which are X, Y, and Z coordinates respectively, and their values 79.83 degrees, -19.0 degrees, and 152.0 degrees correspondingly. So, when we play our project and start controlling the actor it rotates according to these values based on their local rotation.

6.3. Visual Representation of YuMi with Oculus and in UE

From Figure 21 we can see the actual view of YuMi inside Oculus Rift versus how it looks in real life. The presented figures give us a detailed demonstration of how easy, comfortable and precisely it is possible to control and work with the digital twin of the YuMi robot model regarding testing, training of the robot itself. Additionally, the DT model makes it very handy to control the robot according to our needs and apply necessary safety protocols.

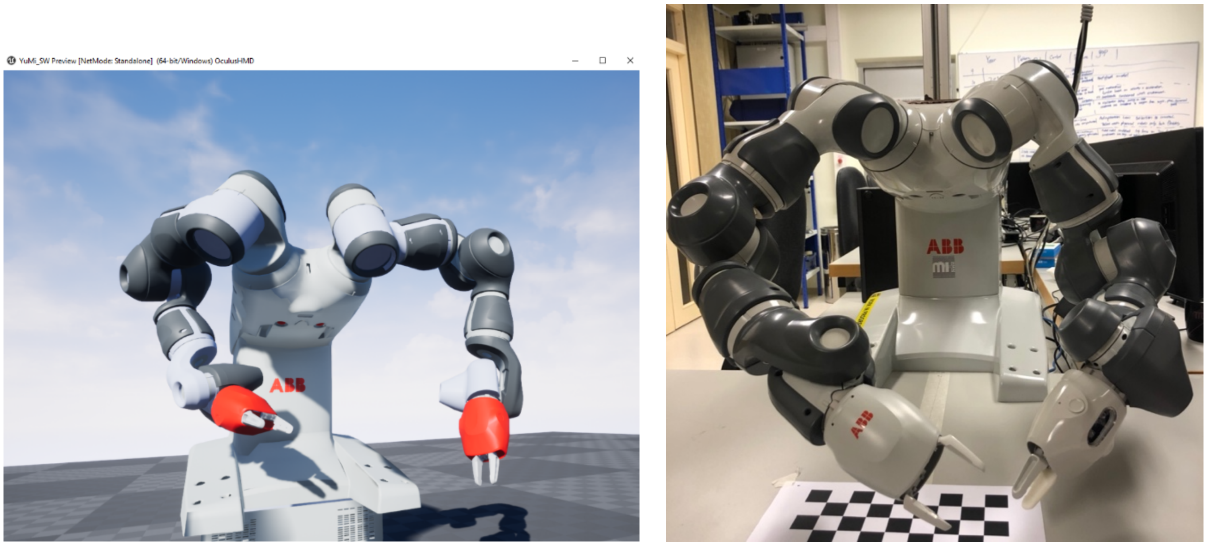


Figure 21: YuMi with the Oculus HMD (Head Mounted Display) and in real life.

Figures 22 and 23 illustrate the working view of the YuMi robot during controlling the robot through Unreal Engine.

Figure 22: High-quality view of YuMi in UE during controlling the arm.

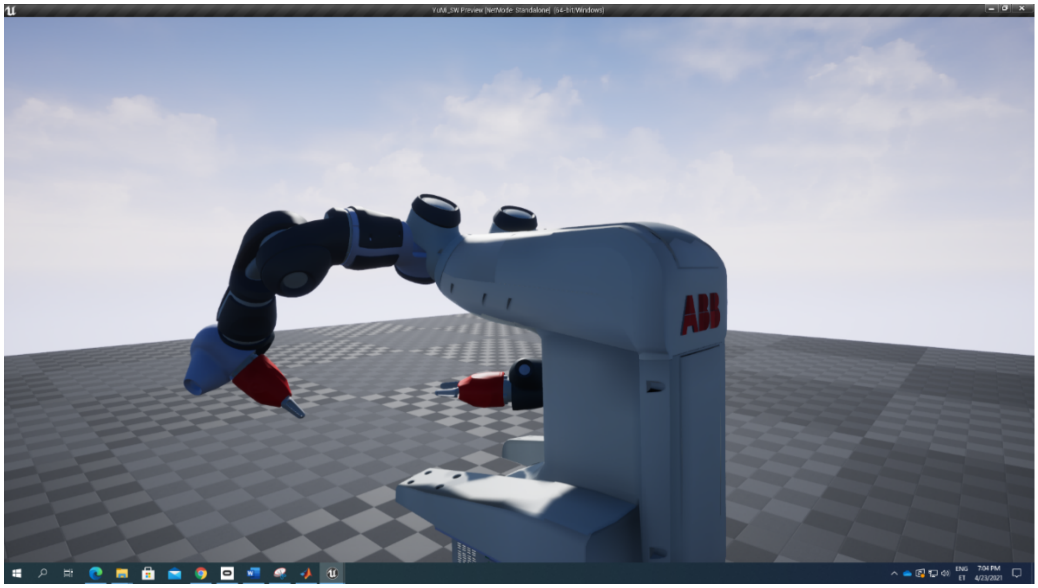


Figure 23: High-quality view of YuMi (from the backside) in UE during controlling the arm.

Additionally, the graphical interpretation of the complete execution together with the datasmith, StreamVR input, and Oculus VR mode can be witnessed from figure 24.

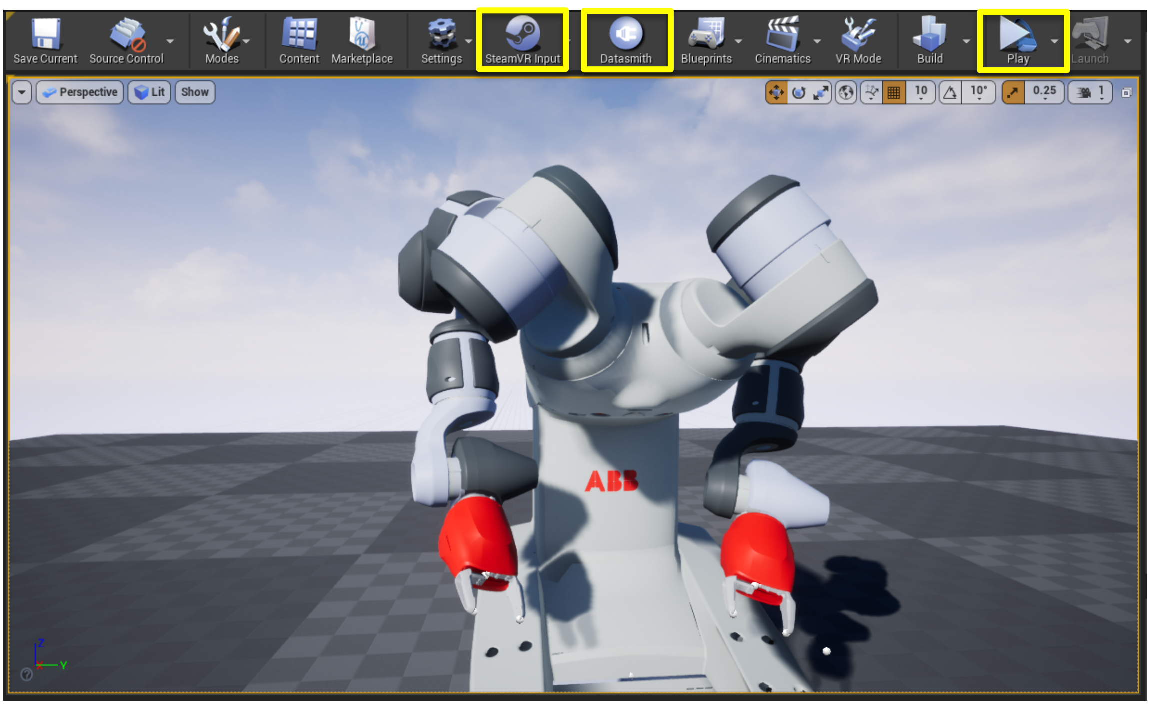


Figure 24: Exemplifies the full connection with yellow markings.

6.4. Inverse Kinematics Solution Using MATLAB

The inverse kinematics system works by creating an inverse kinematic (IK) solver to calculate joint patterns for a preferred end-effector pose based on a specified rigid body tree model. Here below we can see the implementation of IK [41].

```
% Inverse Kinematic
p_ot=dk(1:3,4); % End-effector's position
pw=p_ot-d1*a; % Wrist's position
pw_x=dk(1,4); % Vector's components that represents the wrist's position
pw_y=dk(2,4);
pw_z=dk(3,4);
c3=(pw_x^2+pw_y^2+pw_z^2-a2^2-a3^2)/(2*a2*a3); % cos(teta3)
s3=-sqrt(1-c3^2); % sin(teta3)
teta3=atan2(real(s3),real(c3));
c2=(sqrt(pw_x^2+pw_y^2)*(a2+a3*c3)+pw_z*a3*s3)/(a2^2+a3^2+2*a2*a3*c3); %
cos(teta2)
s2=(pw_z*(a2+a3*c3)-sqrt(pw_x^2+pw_y^2)*a3*s3)/(a2^2+a3^2+2*a2*a3*c3); %
sin(teta2)
teta2=atan2(real((a2+a3*c3)*pw_z-
a3*s3*sqrt(pw_x^2+pw_y^2)),real((a2+a3*c3)*sqrt(pw_x^2+pw_y^2)+a3*s3*pw_z
));
teta1=atan2(pw_y,pw_x);
R3_0=[cos(teta1)*cos(teta2+teta3) -cos(teta1)*sin(teta2+teta3)
sin(teta1);
sin(teta1)*cos(teta2+teta3) -sin(teta1)*sin(teta2+teta3) -cos(teta1);
sin(teta2+teta3) cos(teta2+teta3) 0];
R6_3=R3_0*R; % Matrix for the Euler's angle of spherical wrist
% Inverse kinematic for the spherical wrist
teta4=atan2(R6_3(2,3),R6_3(1,3));
teta5=atan2(sqrt((R6_3(1,3))^2+(R6_3(2,3))^2),R6_3(3,3));
teta6=atan2(R6_3(3,2),R6_3(3,1));
q=[teta1 teta2 teta3 teta4 teta5 teta6]
```

6.4.1. Connection Between MATLAB and Unreal Engine Using Simulink

Here to use the transmission between MATLAB and Unreal Engine, MATLAB version R2021a and Unreal Engine version 4.26.1 have been used.

Automated Driving Toolbox Interface for Unreal Engine 4 Projects is a support package that allows us to modify events in the Unreal editor and utilize them in Simulink.

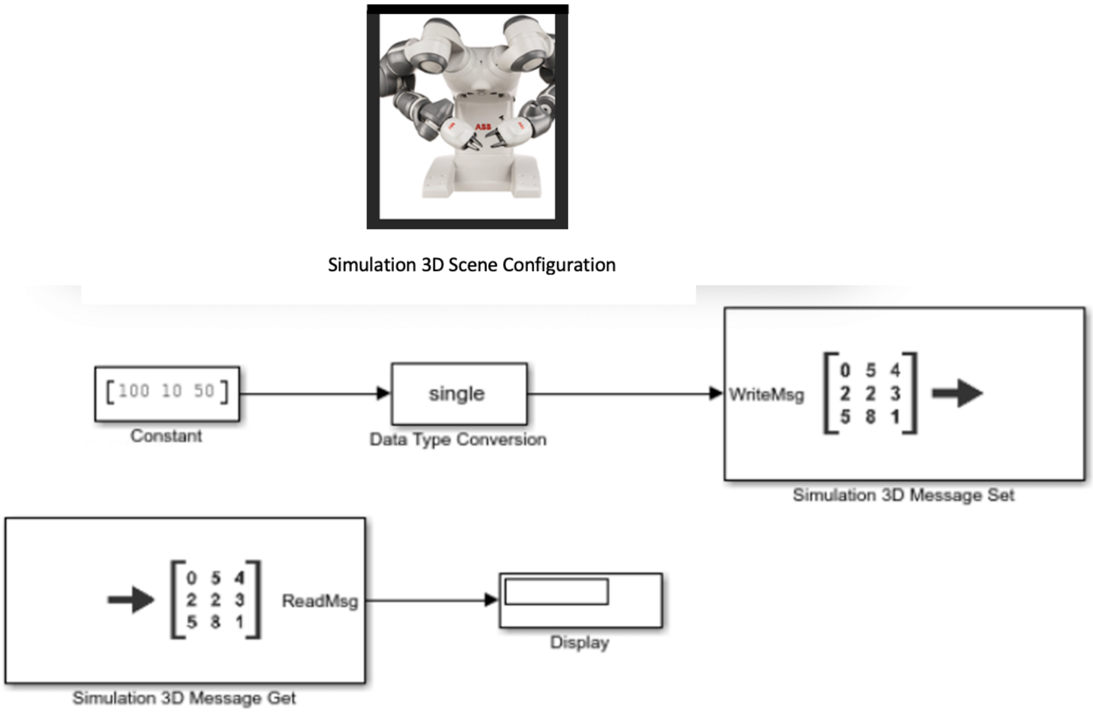


Figure 25: Setting up the Simulink model to send and receive data [38].

C^{++} Workflow: After setting up the sending and receiving model in Simulink we set up the C^{++} roadmap. *Simulation 3D Message Get (Sim3DGetFloat1)* (Figure 26) receives data from Unreal Engine editor C^{++} actor class and *Simulation 3D Message Set (Sim3DSetFloat1)* (Figure 26) sends data to UE editor C^{++} actor class. To do the execution of C^{++}, Visual Studio® 2019 was used [42].

To receive data, C++ syntax:

```
void *StartSimulation3DMessageReader(const char* topicName,uint32
maxDataSize);
int ReadSimulation3DMessage(void *MessageReader, uint32 dataSize,
void *data);
int StopSimulation3DMessageReader(void * MessageReader);
```

To send data, C++ syntax:

```
void *StartSimulation3DMessageWriter(const char* topicName, uint32
maxDataSize);
```

```
int WriteSimulation3DMessage(void * MessageWriter, uint32 dataSize,
void *data);
int StopSimulation3DMessageWriter(void *MessageWriter);
```

We simulated custom scenes instantaneously from both the Unreal editor and Simulink (Figure 25). By using this co-simulation structure, we added the arm, joints, grippers, and fingers of YuMi to a Simulink model and then ran this simulation in our custom scene.

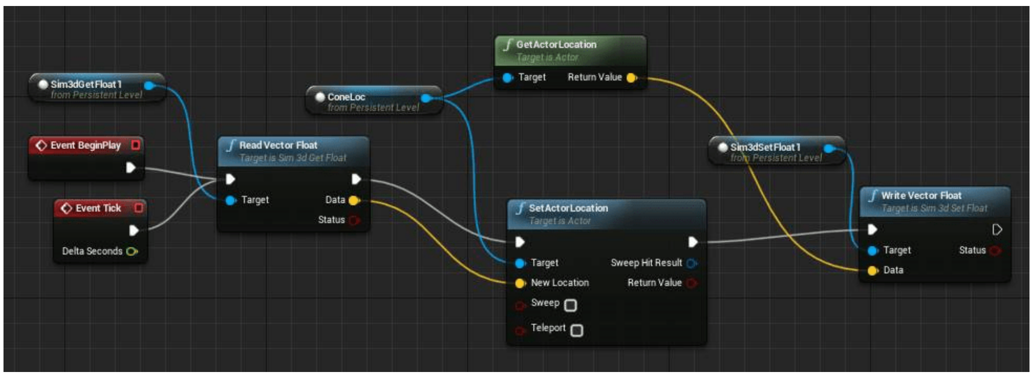


Figure 26: Making the connections in UE level blueprint [42].

Vehicle Dynamics Blockset Interface for Unreal Engine 4 Projects is another support package used to simulate and visualize the YuMi robot modeled in Simulink [44]. From Figure 26 we can see the established connection for MATLAB.

6.4.2. Trajectory Plots and the Visual Representation of YuMi in MATLAB

To set the initial position of the robot arm, followings have been performed.

```
%Define initial state of the robot position, velocity, and acceleration
of each joint.
% Define initial state
q0 = configSequence(10:18,1)'; % Position
dq0 = zeros(size(q0)); % Velocity
ddq0 = zeros(size(q0)); % Acceleration

% Set the initial and final end-effector pose.
jointInit = currentrobotRJConfig;
taskInit = getTransform(robotR,jointInit,endEffector);
```

To get the current state of the end effector,

```
%get current position of end effector
cuurentpos=tform2trvec(getTransform(robotR,currentrobotRJConfig,'gripper_
r_base'))
```

The joint object in MATLAB has a *PositionLimits* property where the limits can be set for each joint. The robotics package provides an *interactiveRigidBodyTree(robotR)* function for the *RigidBodyTree* class. The function gets a tree model, a configuration of the joint angles and a target rigid body of the tree model as an input and calculates the transformation matrix from the base to the target body with the given joints. Grippers are set as the end effectors and the target bodies, for the calculation.

From figure 27, we can observe the plots of the trajectory of the end effector as the robot arm moves from initial position to another.

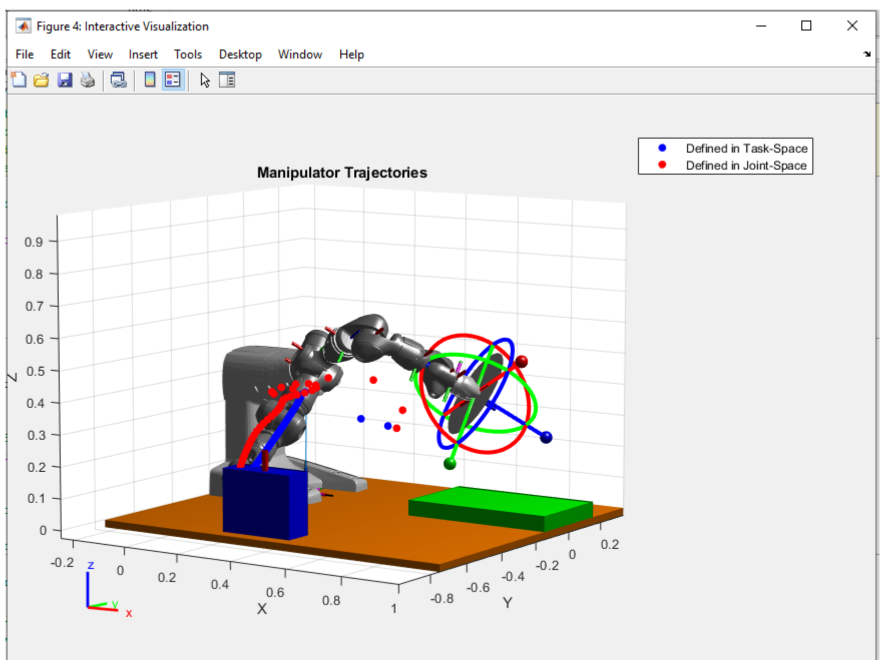


Figure 27: Manipulator Trajectories plots.

Moreover, Figure 28 gives a photographic interpretation of the YuMi robot in MATLAB.

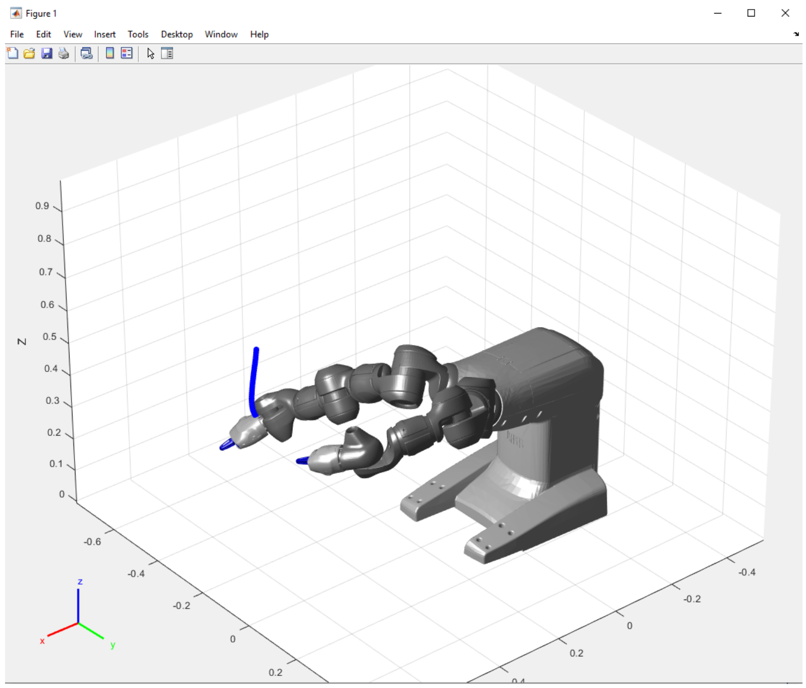


Figure 28: MATLAB Inverse Kinematics of YuMi.

6.5. Use of the Digital Twin of YuMi Robot in VR

For manufacturing robotics companies like ABB, it is very important to do the continuous development of their products. One of the main categories of this development process is testing and training the product. This thesis work is based on ABB's one of the most popular products YuMi robot. The digital twin of the YuMi robot model has been created in VR using Unreal Engine user interface. From figure 29, we can see a general way of how DT of the YuMi robot in VR can be used for the development of the robot.

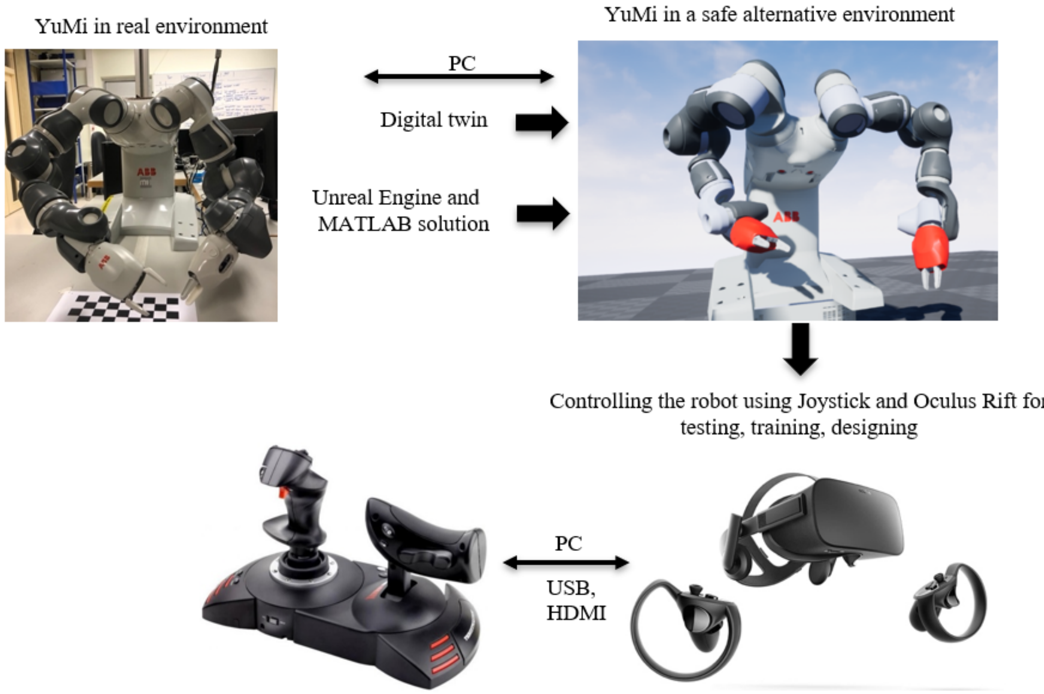


Figure 29: Overview of the way of testing the robot.

With the created digital twin, robotics companies can design various combinations of the robot so that they can offer personalized products and services to their customers. It enables faster installations and easier setups, allowing for higher adoption of robots within industrial settings. To monitor and analyze detailed functionalities of a robot, DT helps developers to see which part of the robot functions are defective or have lower performance than intended. Robotics engineers can use predictive analytics to foresee any future problem involving the joints, arms, grippers, fingers, or other components to ensure the safety of the robot. The above-described development processes cover the main aspects of the robot that can be used by robot manufacturers to ensure unexpected damages and injuries.

7. Summary

It’s been predicted by scientists that most likely by 2050, most of our learning platforms, training materials, military practices, medical solutions, and robotics training will be done through the digital twin (DT) technology in virtual reality (VR). Keeping that in mind, doing a project such as constructing a digital twin of the modern collaborative robot YuMi in VR is a gamechanger. The focus of our thesis work was to provide a safe alternative environment for testing and training the robot using the DT in VR and further development of the robot. We have accomplished our goal by establishing a safe alternative environment by creating the DT of YuMi in VR.

It is well known how important it is for manufacturing companies like ABB to test their products before launching to the real market. But physically building a prototype is time overwhelming plus pricy. Additionally, to maintain robot safety, DT opens many doors for us to implement collaborative robot behavior.

To conclude, it can be said that real-time communications along with DT of the YuMi robot model in VR have been successfully executed, which is designated by reasonable results of the assessment of theoretical expectations and actual performance. Consequently, an immersive environment for testing and training of the digital twin of innovative YuMi collaborative robot for industrial and educational purposes has been effectively generated in UE.

It is planned that in future the detailed investigation will be performed of this thesis work which includes the augmented reality (AR) implementations. It is possible to make a lot of future developments but because of the time limit and the COVID-19 situation, this is all we could do by this time. It is also proposed to implement human interaction with the robot regarding ensuring safety measurements.

References

1. Kumičáková, Darina & Tlach, Vladimír & Císar, Miroslav. (2016). Testing the Performance Characteristics of Manipulating Industrial Robots. Transactions of the VŠB - Technical University of Ostrava, Mechanical Series. 62. 39-50. 10.22223/tr.2016-1/2009.
2. Golnazarian, Wanek & Hall, Ernest. (2002). Intelligent Industrial Robots. 10.1201/9780203908587.ch6.5.
3. R. Azuma. A survey of virtual reality. ACM SIGGRAPH, 1-38, 1997.
4. S. Rabah, A. Assila, E. Khouri, Florian M., Fakreddine A., V. Bourny, P. Maier, F. Mérienne, "Towards improving the future of manufacturing through digital twin and virtual reality technologies", 28th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM2018), June 11-14, 2018, Columbus, OH, USA.
5. Mathieu B ́elanger-Barrette, "Robotiq Collaborative Robot Ebook" Robotiq, 2016, 6th edition.
6. Malykhina, G. F, "Digital Twin Technology as A Basis of The Industry In Future", 2018. DOI: 10.15405/epsbs.2018.12.02.45.
7. Negri, E., Fumagalli, L., Macchi, M., 2017.A review of the roles of digital twin in cps-based production systems. Procedia Manufacturing 11, 939–948.
8. R ́ıos, J., Herna ́ndez, J.C., Oliva, M., Mas, F., 2015. Product avatar as digital counterpart of a physical individual product : Literature review and implications in an aircraft., in: ISPE CE, pp. 657–666.
9. Boschert, S., & Rosen, R. X. (2016). Digital Twin—The Simulation Aspect. In Hehenberger P., Bradley D. (Eds.) Mechatronic Futures. Cham: Springer.
10. A. Fuller, Z. Fan, C. Day and C. Barlow, "Digital Twin: Enabling Technologies, Challenges and Open Research," in IEEE Access, vol. 8, pp. 108952-108971, 2020, doi: 10.1109/ACCESS.2020.2998358.
11. E. Gambao, M. Hernando, D. Surdilovic, "A new generation of collaborative robots for material handling", 2012.
12. ABB Robotics, "YuMi® - Creating an automated future together.," [Online]. Available: https://new.abb.com/products/robotics/industrial-robots/irb-14000-yumi.
13. S. Giordani, M. Lujak, F. Martinelli "A distributed multi-agent production planning and scheduling framework for mobile robots" Computers and Industrial Engineering, 64 (2013), pp. 19-30.
14. Grieves, M., Vickers, J., 2017. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems, in: Transdisciplinary, Perspectives on Complex Systems. Springer, pp. 85–113.
15. Greg Rauhoeft, Markus Leyrer, William B. Thompson, Jeanine K. Stefanucci, Roberta L. Klatzky, and Betty J. Mohler. Evoking and assessing vastness in virtual environments. In *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception*, SAP '15, 51–54. New York, NY, USA, 2015.
16. S. Kucuk and Z. Bingul, "Inverse kinematics solutions for industrial robot manipulators with offset wrists," Applied Mathematical Modelling, vol. 38, no. 7-8, pp. 1983–1999, 2014.

17. X. H. Xie, S. M. Fan, X. Y. Zhou, and Z. Y. Li, “Inverse kinematics of manipulator based on the improved differential evolution algorithm,” Robot, vol. 41, no. 1, pp. 50–57, 2019, in Chinese.
18. ABB Robotics, Product specification IRB14000, Västerås: ABB, 2018.
19. Boschert, S., & Rosen, R. X. (2016). Digital Twin—The Simulation Aspect. In Hehenberger P., Bradley.
20. D. (Eds.) Mechatronic Futures. Cham: Springe.
21. ISO. 15066: Robots and robotic devices–collaborative robots. Geneva, Switzerland: International Organization for Standardization, 2016.
22. J. Collins, H. Regenbrecht, T. Langlotz, Visual coherence in mixed reality: A systematic enquiry, Presence Teleoperators and Virtual Environments, 26 (1) (2017), pp. 16-41.
23. P. Milgram and F. Kishino, ‘A Taxonomy of Mixed Reality Visual Displays’, IEICE Trans Inf. Syst., vol. E77-D, no. 12, pp. 1321–1329, Dec. 1994.
24. Z. Ya-Lun, X. Jia-Wei, Y. Lu-Jing, H. Yin-Dong, Sun Shadow Positioning and Timing Based on Unreal Engine 4, 2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC).
25. Oculus Rift, Product Visualization, © Facebook Technologies, LLC, 2021.
26. L.P. Berg, J.M. Vance, Industry use of virtual reality in product design and manufacturing: A survey, Virtual Reality, 21 (1) (2016), pp. 1-17.
27. P. Milgram, F. Kishino, A taxonomy of mixed reality visual displays, IEICE Transactions on Information and Systems, 77 (12) (1994), pp. 1321-1329.
28. F. Bonetti, G. Warnaby, L. Quinn, Augmented reality and virtual reality in physical and online retailing: A review, synthesis and research agenda, T. Jung, M. tom Dieck (Eds.), Augmented reality and virtual reality. Progress in IS, Springer, Cham (2018), pp. 119-132.
29. Greenlight, 2015 virtual reality consumer report, Retrieved from https://goo.gl/Lpfi94 (2015), Accessed 14th Jan 2021.
30. Kathryn Jeffords, "Virtual and Augmented Reality: Changing the Game in Healthcare," SMASH: Science Media Awards & Summit in the Hub (website), June 29, 2016.
31. J. Krüger, G. Schreck, D. Surdilovic. Dual arm robot for flexible and cooperative assembly. CIRP Annals - Manufacturing Technology, 2011, 60(1):5-8.
32. ABB Robotics, Product specification IRB14000, Affolternstrasse 44, Zürich, Switzerland: ABB, 2021.
33. Oculus Best Practices 2016. Version 310-30000-02, http://static.oculus.com/documentation/pdfs/intro-vr/latest /bp.pdf.
34. Oculus VR. EVE: Valkyrie, Open Source Hardware, and the Best Practices Guide. Feb. 5, 2014. (Feb. 11, 2014) http://www.oculusvr.com/blog/eve-valkyrie-open-source-hardwareand-the-best-practices-guide/.
35. Andrew Sanders, An Introduction to Unreal Engine 4, CRC Press, 2016, pp. 4-13.

36. J. Busby, Z. Parrish, J. Wilson, Mastering Unreal Technology, Volume II: Advanced Level Design Concepts with Unreal Engine 3, Sams Publishing, 2009.
37. Unreal Engine documentation, Epic games, USA, 2021.
38. The MathWorks Documentation, MATLAB, USA, 2021.
39. D. H. Song and S. Jung, "Geometrical analysis of inverse kinematics solutions and fuzzy control of humanoid robot arm under kinematics constraints," Proc. 2007 IEEE Int. Conf. Mechatronics B Autom. ICMA 2007, pp. 1178–1183, 2007.
40. A. Sodemann, "Inverse Kinematics of Articulated Manipulator," 2017. [Online]. Available: Inverse Kinematics of Articulated Manipulator. (Accessed May 1st, 2021).
41. "Inverse Kinematics - Mathworks.com" https://se.mathworks.com/help/robotics/ug/ interactively-build-a- trajectory-abb-yumi.html - Kinematics (Accessed May 3rd, 2021).
42. "MathWorks Automated Driving Toolbox - Mathworks.com" https://se.mathworks.com/matlabcentral/fileexchange/74555-automated-driving-toolbox-interface-for-unreal-engine-4-projects - MathWorks (Accessed May 3rd, 2021).
43. "T Flight Hotas X- Thrustmaster.com" http://www.thrustmaster.com/products/tflight-hotas-x - PC / PlayStation®3 (Accessed May 3rd, 2021).
44. "Vehicle Dynamics Blockset Interface - Mathworks.com" https://se.mathworks.com/matlabcentral/fileexchange/65966-vehicle-dynamics-blockset-interface-for-unreal-engine-4-projects - MathWorks (Accessed May 5th, 2021).
45. "Oculus Rift – Oculus.com" https://www.oculus.com/rift/ - Oculus (Accessed May 5th, 2021).
46. F. Ionescu, F. Chojnowski and G. Constantin, 2002. "Virtual Reality in Mechanical Engineering, Modelling and Simulation with Solid Dynamics", ARA-Journal, vol. 2002. No. 27.
47. "SolidWorks – Solidworks.com" https://www.solidworks.com/3dexperience-works - 3D CAD (Accessed May 6th, 2021).
48. "ABB YuMi – ABB.com" https://new.abb.com/products/robotics/collaborative-robots/irb-14000-yumi/irb-14000-yumi-data - Technical data IRB 14000 YuMi (Accessed May 6th, 2021).
49. "ABB – New.abb.com" https://new.abb.com/news/detail/63763/yumi-five-years - Zurich, Switzerland (Accessed May 6th, 2021).

Appendix

Figure 30 describes the blueprint implementation of *actors* from the *world outliner* to create collision in the joints of the arm, and the use of input functions to control the *actors* through a keyboard or the joystick.

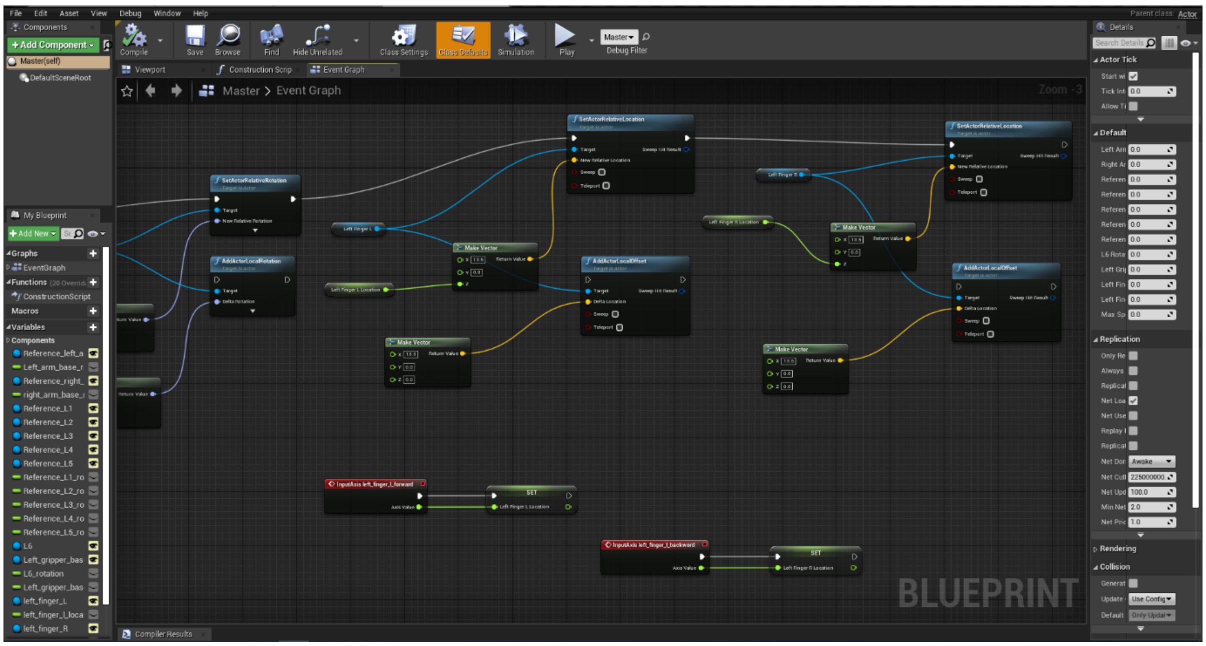


Figure 30: Blueprint of Controlling the YuMi arm.