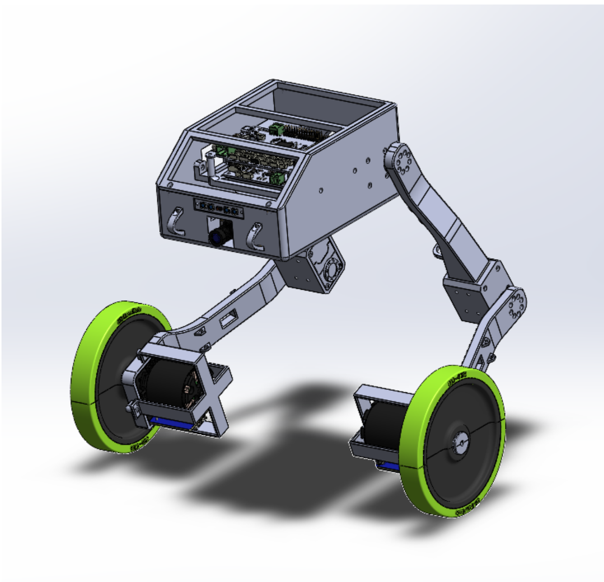
Institut für Mechatronische Systeme
imes Leibniz Universität Hannover

Leibniz Universität Hannover

# Design and Control of a Multi-Joint Two-Wheeled Inverted Pendulum Robot for Agile Motion Learning



Master's thesis M-MM/JJJJ-XXXXX

Mohamed Barakat
Matrikelnummer 10044038

Hannover, 14. January 2024

| | |
|---|---|
| First Examiner | Prof. Dr.-Ing. Thomas Seel |
| Second Examiner | Prof. Dr.-Ing. Firstname Lastname |
| Supervisor | Dustin Lehmann |

Institut für Mechatronische Systeme
imes Leibniz Universität Hannover

Leibniz Universität Hannover

**M-MM/JJJJ-XXXXX**

# Declaration of Authorship

| | |
|---|---|
| **Name:** | **Mohamed Barakat** |
| Registration no.: | 10044038 |
| Thesis title: | Design and Control of a Multi-Joint Two-Wheeled Inverted Pendulum Robot for Agile Motion Learning |
| Type of thesis: | Master's thesis |
| Study program: | Mechanical Engineering |
| Submission date: | 14. January 2024 |
| First examiner : | Prof. Dr.-Ing. Thomas Seel |
| Second examiner: | Prof. Dr.-Ing. Firstname Lastname |
| Supervisor: | Dustin Lehmann |

I, *Mohamed Barakat*, hereby affirm that the Master's thesis entitled *Design and Control of a Multi-Joint Two-Wheeled Inverted Pendulum Robot for Agile Motion Learning* was written independently, that no references and aids other than those indicated were used, that all passages of the thesis which were taken over literally or analogously from other sources are marked as such and that the thesis has not yet been presented to any examination board in the same or similar form.

I hereby agree to the transmission of my work also to external services for plagiarism checking by plagiarism software.

Hannover, 14. January 2024

(Mohamed Barakat)

# Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

| Ort, Datum | Mohamed Barakat |
|---|---|

**Abstract**

This thesis addresses the challenge of input trajectory transfer in heterogeneous multi-agent systems, where each agent has distinct system dynamics.
The concept of transfer learning has been applied to MAS to accelerate the learning process of a target agent by leveraging previously learned trajectories from a different source agent. However, directly transferring input trajectories between agents with dissimilar dynamics can lead to unwanted behaviour. This can be overcome by applying a mapping that transforms the input trajectory of the source system into an appropriate input trajectory for a target system so that their outputs match. Obtaining such an input transfer map from simple experiments is difficult, so previous research often focuses on output transfer, adaptive control or strong similarity assumptions. Those approaches either require extensive model knowledge or have limited usability.

This thesis provides a comprehensive understanding of trajectory transfer, distinguishing the input and output transfer cases and highlighting their similarities. Those insights are then used to propose a simple, data-driven method to estimate a dynamic input transfer map for SISO systems that addresses the aforementioned problem. Structural similarity in the systems is leveraged to simplify the estimation process further. The transfer map is estimated as a lifted system matrix and a transfer function. The performance of the estimated dynamic input transfer map is compared to a static map using a simple gain and to the direct transfer of the input trajectories.

It is shown that input and output transfer are equivalent under certain conditions. An input transfer map based on this assumption was able to significantly reduce the differences between a source and a target system in three simulated scenarios. This even includes scenarios where the systems have nonlinear, non-minimum phase dynamics.
Compared to a static map, the performance of a dynamic input transfer is superior, especially in cases where the estimated transfer map is applied to input trajectories which are not part of the training data. Furthermore, the dynamic input transfer map can have a model order lower than the theoretically expected order of the transfer system, given that the source and target systems are related. Therefore, the expected order only provides an upper limit for the order of an estimated transfer map, and not a recommended choice.
Nonetheless, this thesis alone does not fully explore the potential and limitations of the proposed method. Future research is necessary to address these topics further.

***Keywords***— input transfer, trajectory transfer, transfer learning, multi-agent systems

# Contents

# List of Figures

# List of Tables

# 1 Introduction

## 1.1 Motivation

This section should explore the underlying reasons for undertaking the research. It might include the importance of the topic, the gap in current knowledge, and the potential applications of the research findings.

Robotics has always been a cutting-edge field that combines sophisticated engineering, artificial intelligence, and a comprehension of human-environment interactions. This study is motivated by multiple important elements, all of which highlight the importance and relevance of the research.

**Advancements in Robotic Technologies:**

**Filling Knowledge Gaps:**

**Potential for Broad Impact:**

**Interdisciplinary Collaboration and Innovation:**

## 1.2 Explanation of the Goals and Requirements

# 2 Literature Review

## 2.1 Overview of Two-Wheeled Inverted Pendulum Robots

## 2.2 Prior Works and Advances in Multi-Legged Robotic Systems

This section should provide a comprehensive review of existing literature relevant to your research topic. It should cover key theories, models, experiments, and findings in the field, particularly focusing on works that directly relate to your research question or hypothesis. This review not only shows your understanding of the field but also how your work fits into and contributes to the existing body of knowledge.

## 2.3 State-of-the-Art in Robotic Control Systems

# 3 Mechanical Design

Modeling In this chapter, the details of the mechanical design are presented. where it goes from the initial conceptual design to the final design showing in the process the design decision-making for each critical point, This chapter emphasis the detailed description of the precise placement and alignment of different components such as the motors, wheels, battery, boards, and others to maintain the seamless integration of all the components into the robot body.
The mechanical design serves as an important pillar to define the new physical form, size, and shape of the TWIPR. Depending on how these criteria are defined, the robot would interact with the environment. taking into account that the design directly influences the center of gravity, which is crucial to consider in our robot due to the inverted pendulum nature to be able to balance and maintain the upright position. In addition to the impact that the design has on the maneuverability of the robot and how it would respond to the control signals to be able to execute a task.

## 3.1 Design Objectives and Requirements

The main objective is to come up with a new design for a multiple joints Robot to perform complex dynamic movements. This robot would be based on the Two wheeled inverted pendulum robots. The new design would add more degrees of freedom to enable the more complex movements. This enhances the robot's capabilities where it can execute more diverse scenarios. Initially, the main requirement was to add two more degrees of freedom where originally it used to have one degree of freedom in the wheels. The new design has three degrees of freedom, one in the wheels, one as a knee joint and one as a hip joint. The current design has two identical legs.

## 3.2 Conceptual Design

As for the initial design concepts the body was that main point of focus as shown in the figure 3.1.Three initial designs where considered mainly for the body. Symmetrical vertical body in figure A, symmetrical horizontal body in figure B and leaning forward body in figure C. For the three designs, two independent legs where considered. two designs where considered for the legs, the normal joint leg and the compliant leg

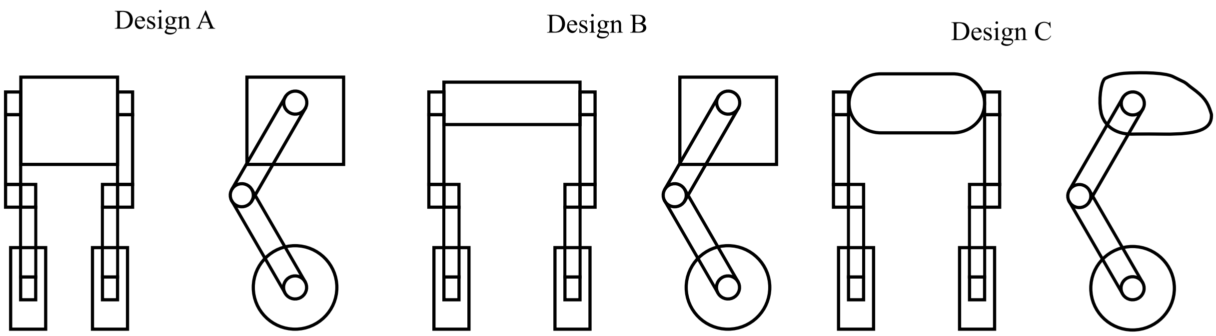



**Figure 3.1:** Three initial Design Concepts for the robot body

Two main designs where considered for the legs, the normal joint leg and the compliant leg. The compliant leg is more flexible and can be used to absorb the shock from the ground. The normal joint leg is more rigid and can be used to generate more torque.in addition, the normal is more relative to our use-case as it can precisely control the position of the leg.

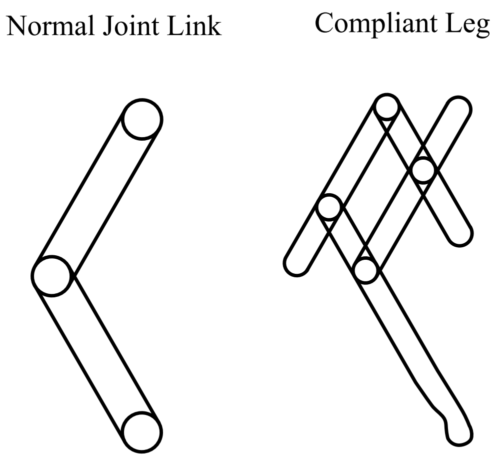



**Figure 3.2:** Leg Design Concepts

## 3.3 Schmatic Representation of the Robot

## 3.4 Initial Calculations

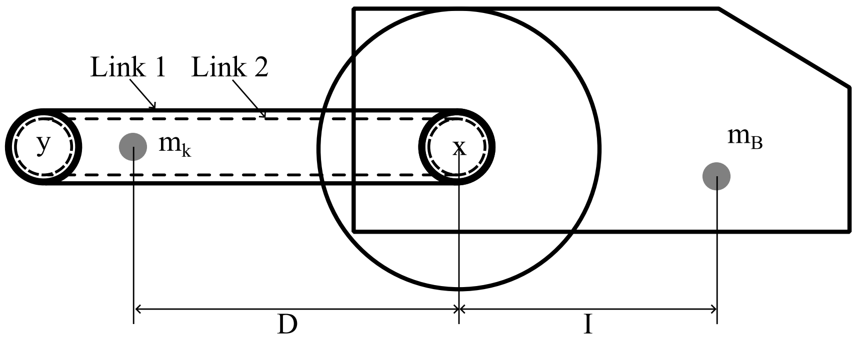



**Figure 3.3:** Initial Calculations

One of the main critical positions for the robot, as shown in the figure 3.3 is the position where the center of mass of the body and the center of mass of the legs are the furthest away from point x on the horizontal axis.It is important to make the initial torque calculations to be able to select the right motors for the robot. Starting from that position first, the robot should be able to balance itself and maintain the upright position. secondly, the robot should be able to change the knee angle to lift the body while maintaining its balance. The calculations are based on some assumptions and simplifications, Considering only half the body weight and one leg joint. As shown in the figure, The Wheel motor shaft is aligned with the hip joint, the minimum torque required to balance the robot is calculated as follows:

The summation of the torques around point x should be equal to zero to maintain the balance with minimum motor torque.

$$\sum_{i=1}^{n} \tau_i = 0 \tag{3.1}$$

$$\sum_{i=1}^{n} \tau_i = m_B * g * D - m_K * g * I \tag{3.2}$$

$$\sum_{i=1}^{n} \tau_i = 0.5 * 9.81 * 0.03 - 0.2 * 9.81 * 0.5 * 0.15 = 0Nm \tag{3.3}$$

The Lengths of D and I can be modified to make sure that the summation of the torques around point x is equal or approximately equal to zero. This will make sure that the robot can balance itself with minimum wheel motor torque. The torques can cancel each other out by readjusting the lengths of D or I and also by modefiing the weight distribution in the body and the legs.

The torque required to lift the body is calculated as follows:

$$\sum_{i=1}^{n} \tau_i = m_B * g * (D + I) - m_K * g * L_1 \tag{3.4}$$

$$\sum_{i=1}^{n} \tau_i = 0.5 * 9.81 * (0.03 + 0.15) - 0.1 * 9.81 * 0.5 * 0.15 = 0.9555Nm \tag{3.5}$$

0.9555 Nm is the minimum torque required to hold the body in position. The Knee motor should be able to generate more torque to be able to lift the body and change the knee angle.

**Table 3.1:** Initial Calculations Assumptions

| Parameter | Value | Description |
|---|---|---|
| $m_B$ | 0.5 kg | Mass of the body |
| $m_K$ | 0.2 kg | Mass of the Knee including the two legs |
| $L_1$ | 0.15 m | Length of Link 1 |
| $L_2$ | 0.15 m | Length of Link 2 |
| $d$ | 0.3 m | distance from the hip joint axis to the center of mass of the body |

**Note 3.1**

**To reduce the needed torque to lift we can:**

- Shorten the links
- Reduce the body weight
- Use gearbox to increase the torque

## 3.5 Detailed Design Development

Throughout the design process, modularity and ease of assembly were considered. The design was broken down into three main parts: the body, the hip-knee joint, and the knee-wheel joint. Print iterations were made to ensure that the parts fit together and that the robot could be assembled with ease. Modifications were made to simplify the printing process and reduce the support material used.

### 3.5.1 Body Design

The body design is the main part of the robot, it is the main structure that holds all the components together. The components that are mounted on the body are the hip motors, the battery, camera, sensor, a rack that includes the power distribution board, the microcontroller board attached to the Raspberry Pi.The optimization of the body design is crucial to be able to fit all the components in a compact form and at the same time distribute the weight equally to maintain the balance of the robot. Different design features were considered, such as the cable management, the fastening features specifically designed to easily mount the battery, organize the cable between the boards and the rest of the robot parts.

**Figure 3.4:** TOP view of the Body Design

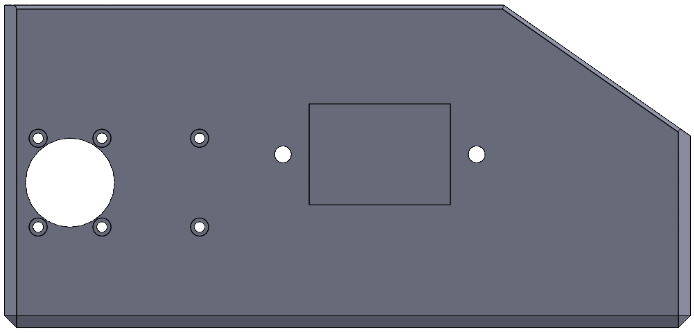


**Figure 3.5:** side view of the Body Design

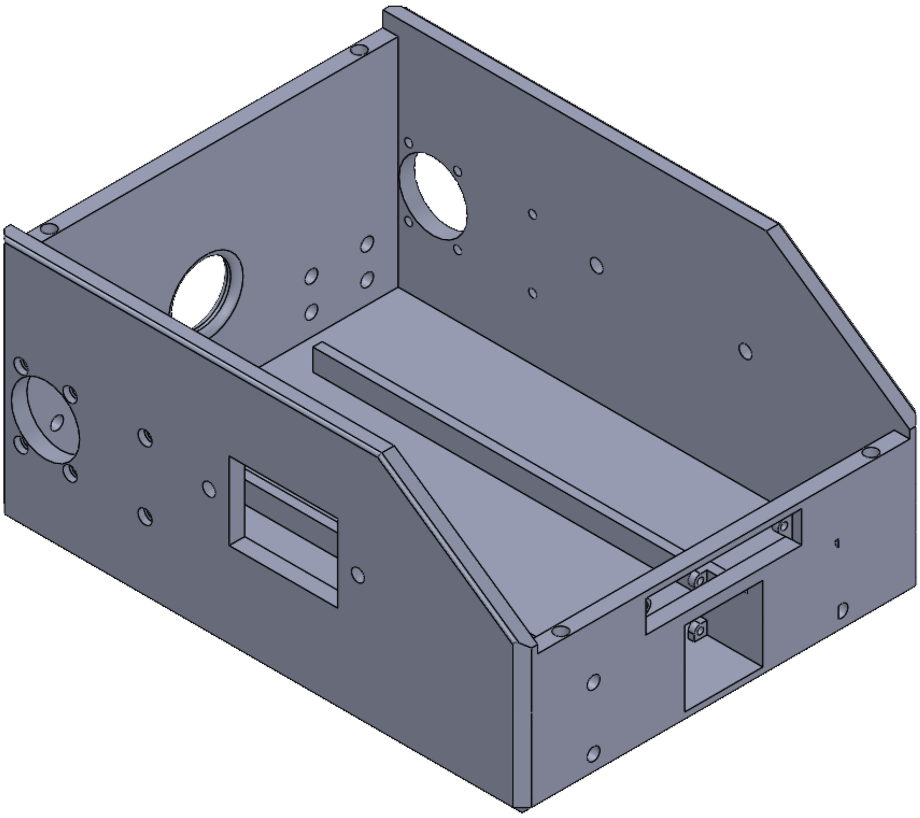


**Figure 3.6:** 3D view of the Body Design

### 3.5.2 Hip Knee Joint Design

This joint connects the hip body axis to the knee axis. The design of this joint includes the knee motor housing form one-side and a frame to attach to the output horn of the hip motor. The motor housing is designed to ensure the fixed placement of the motor inside the joint, As shown in the figure, 3.7 Cable management is also considered in the design of this joint to make sure that the cables are not interfering with the movement of the joint.The curvature of the joint creates enough clearance so that when the hip axis is aligned with the wheel axis, they would not touch with each other even when considering the elastic bending of the legs.

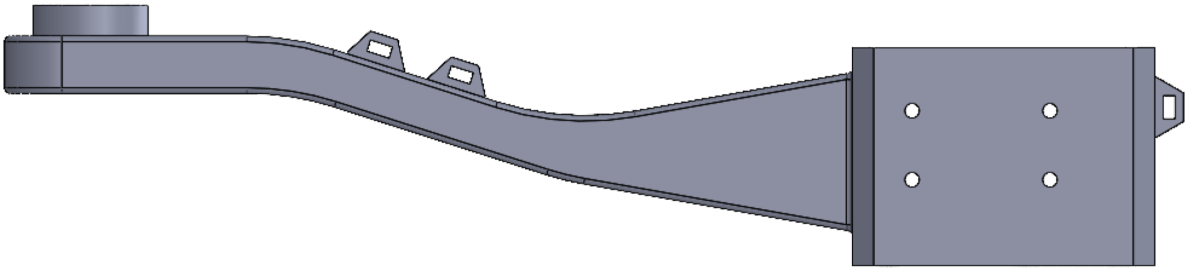


**Figure 3.7:** TOP view of the Body Knee Joint

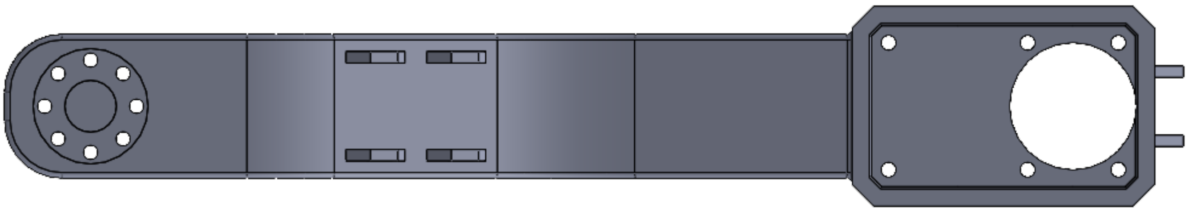


**Figure 3.8:** side view of the Body Knee Joint

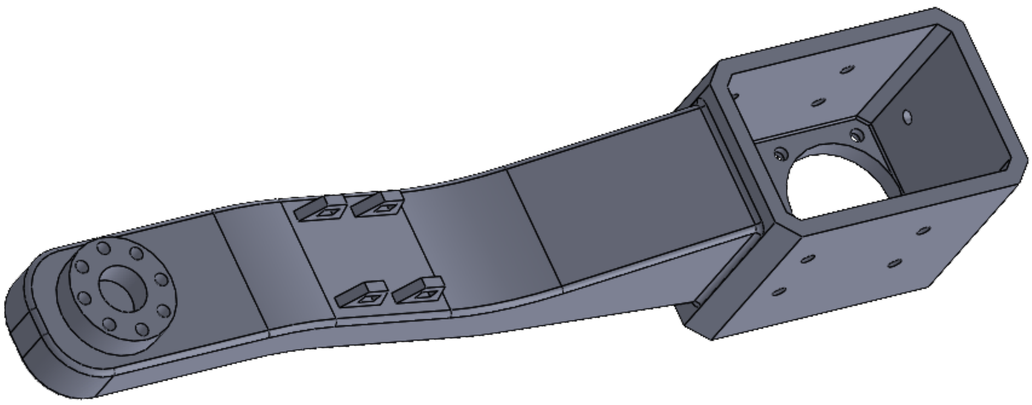


**Figure 3.9:** 3D view of the Body Knee Joint

### 3.5.3 Knee Wheel Joint Design

This joint connects the knee axis to the wheel axis. The design of this joint includes the wheel motor mounting form one-side and a frame to attach to the output horn of the knee motor from the other side.

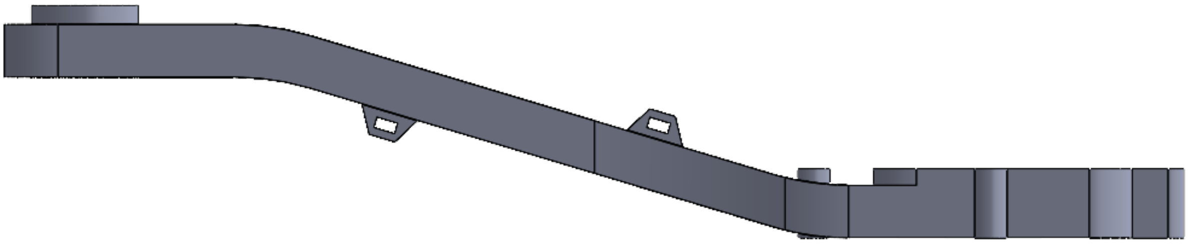


**Figure 3.10:** SIDE view of the Knee Wheel Joint

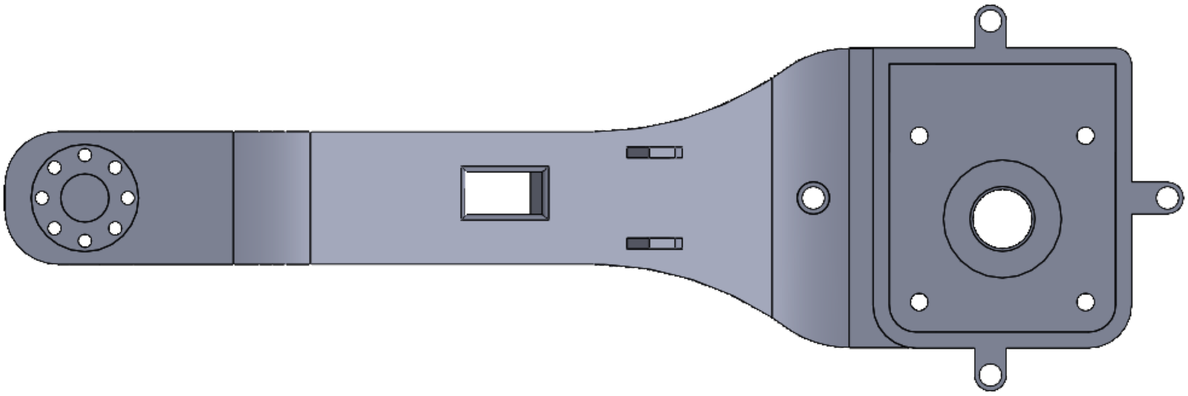


**Figure 3.11:** TOP view of the Knee Wheel Joint

### 3.5.4 Full Design

### 3.5.5 safety considerations

## 3.6 Design for Manufacturability and Assembly

## 3.7 Prototyping and Iterative Design

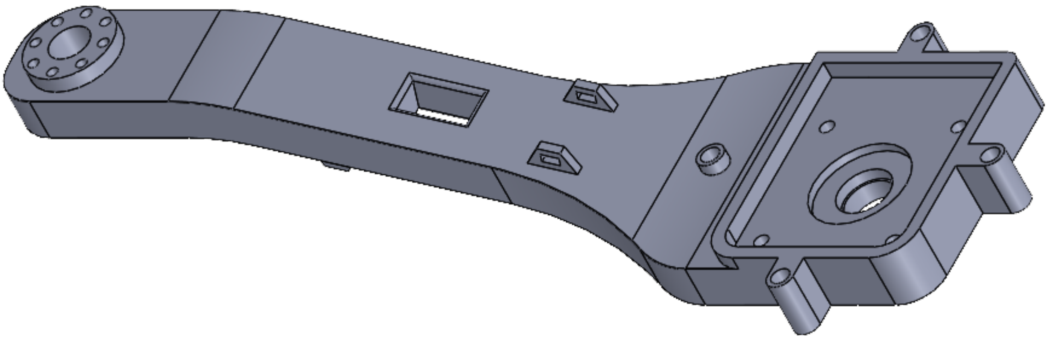


**Figure 3.12:** 3D view of the Knee Wheel Joint

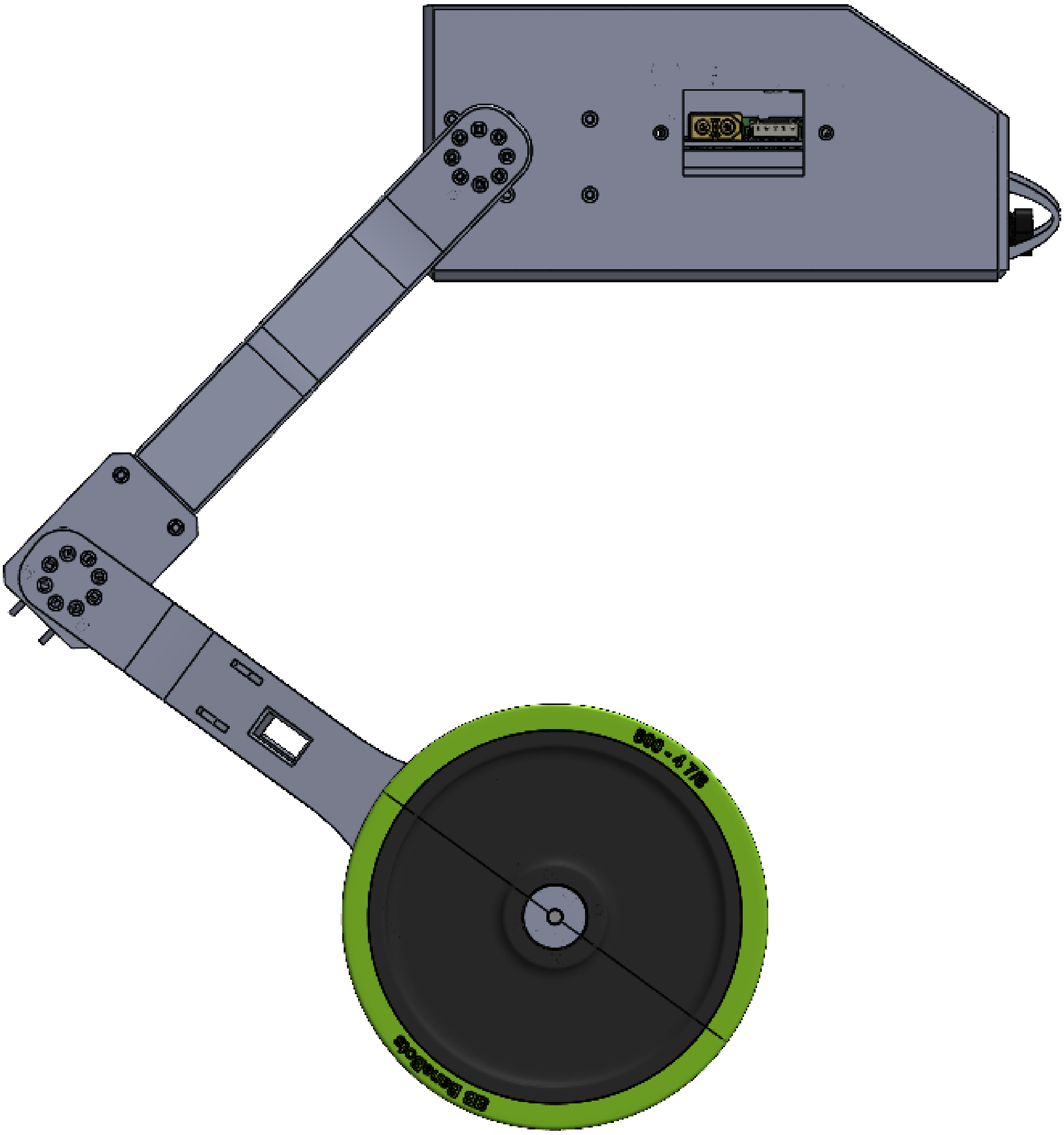


**Figure 3.13:** Side view of the Robot Assembly

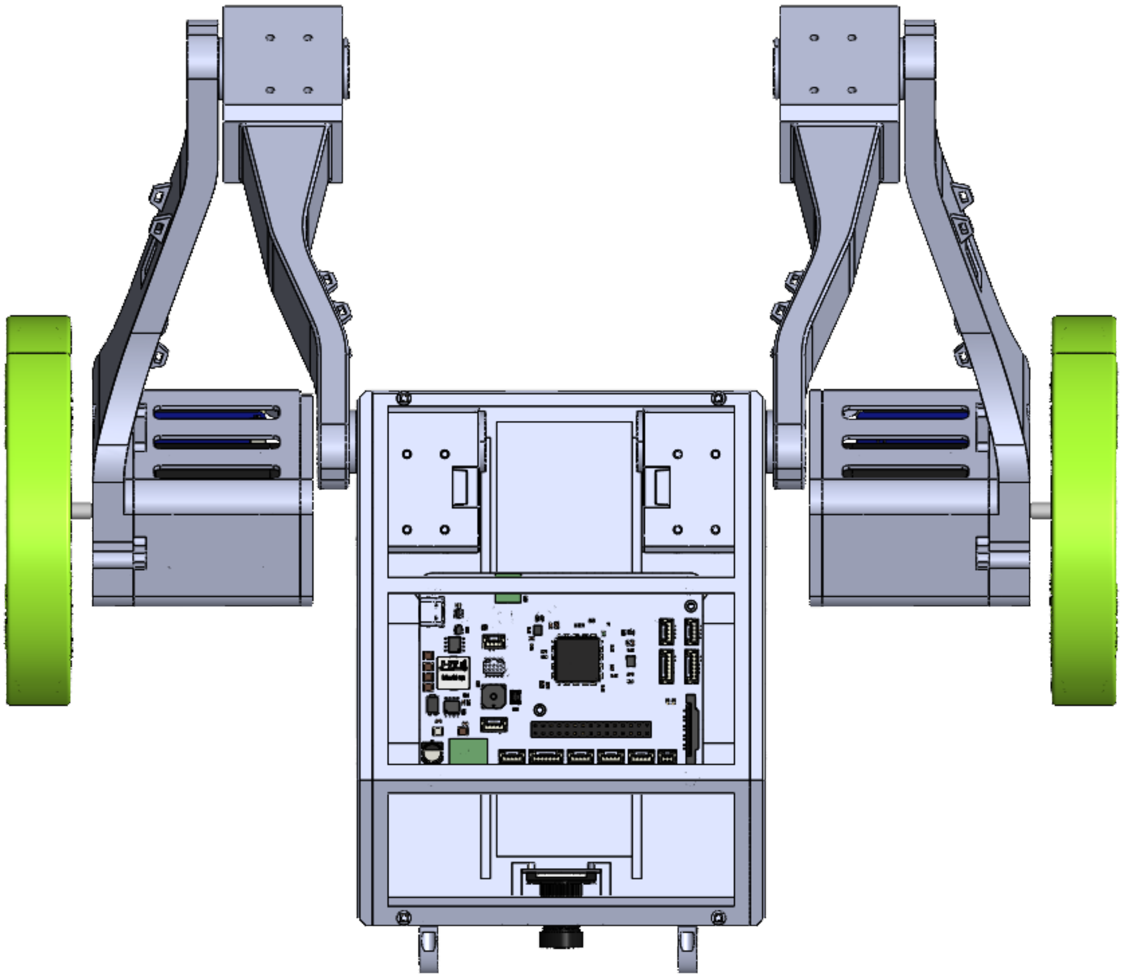


**Figure 3.14:** Front view of the Robot Assembly

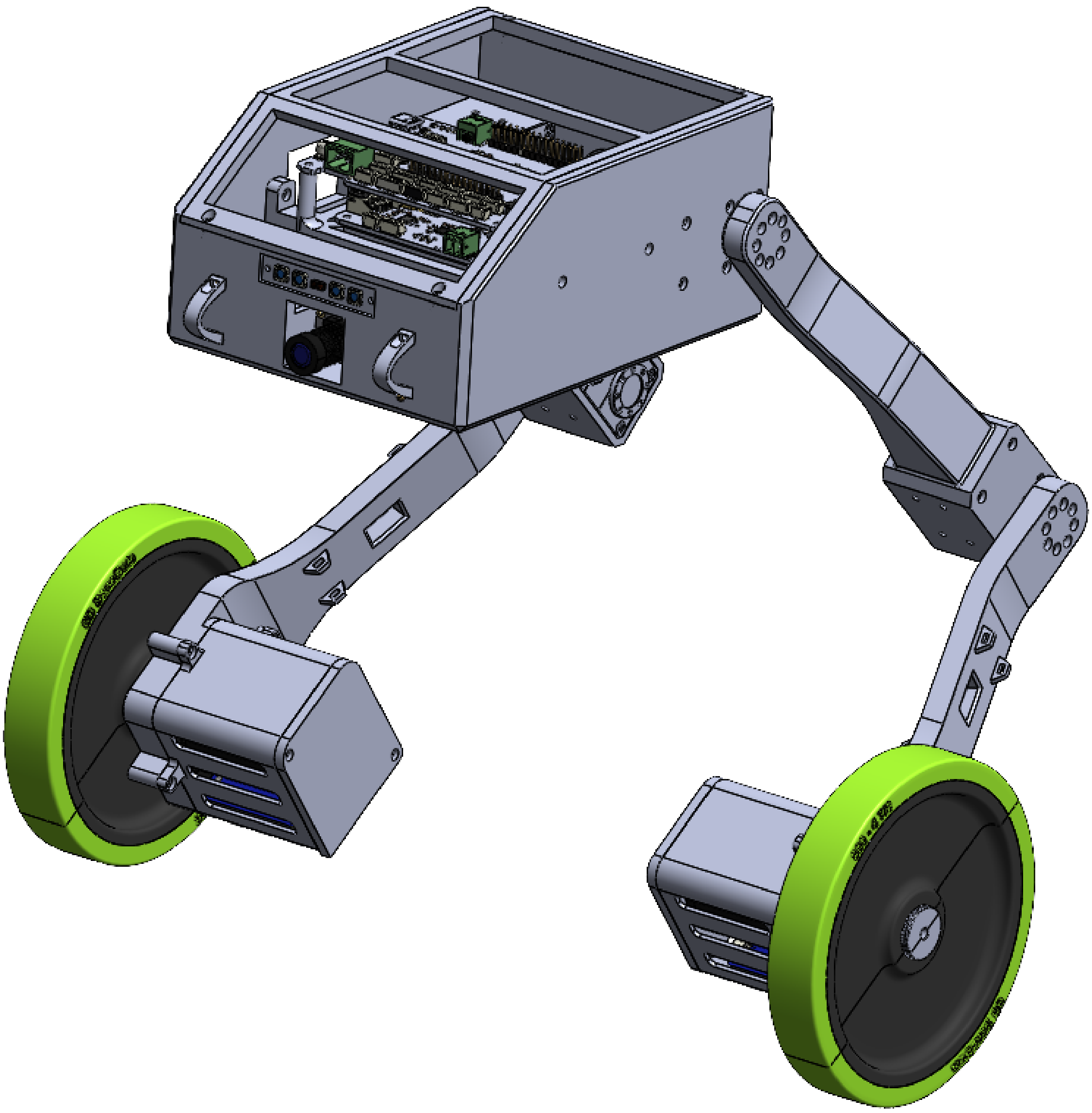


**Figure 3.15:** 3D view of the Robot Assembly

# 4 Electronic Design

In this chapter, the details of the electronic design are presented. Where it emphasizes the details of the components' technical specifications and the selection process. The chapter also discusses the circuit design and the PCB design. In addition, the chapter discusses the power management and the power distribution.

The electronic design serves as a critical link between the robitic conceptual framework and the physical implementation of the robot.The electronic design translates the abstract control algorithm into tangible action and responses. The electronic design directly influences the performance, responsiveness, adaptability to various scenarios.

## 4.1 Design Objectives and Constraints

The main objectives of the electronic Design are to connect the different components and enable them to perform the desired tasks. The electronic design is responsible for controlling the motors. Robust communication between the electronic components is needed to ensure reliable operation of the robot. The electronic design is constrained by the power requirements of the motors. The motors require a high current to operate, and the electronic design should be able to provide the required current. In addition, the electronic design is constrained by the size of the components and their placement in the robot body.

## 4.2 Component Selection

The component diagram is shown in a figure 4.1shows the different components and their relationship with each other. The main components are the microcontroller, rassberry pi, the motors, Distance sensor, Camera, the battery. The motors are chosen based on the calculations done in the mechanical design chapter. The microcontroller is chosen based on the complixity of the control algorithm and the processing power needed to run the control algorithm.

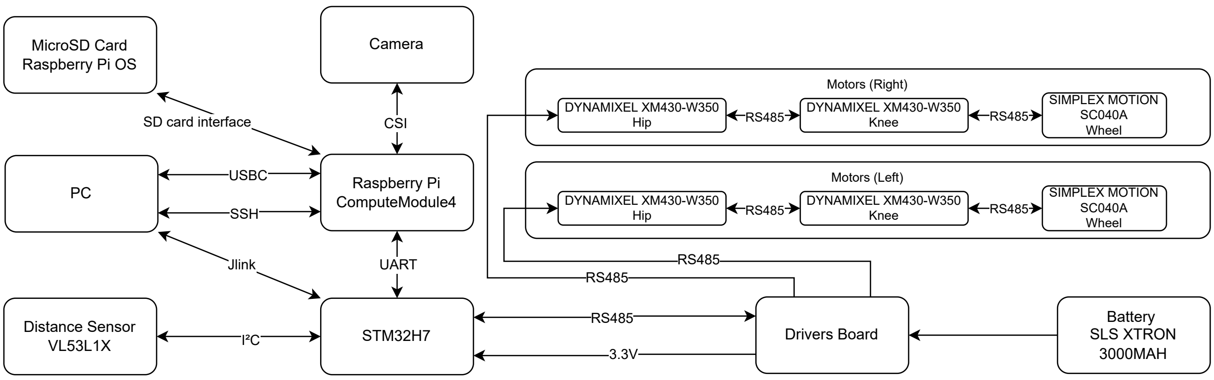



**Figure 4.1:** Components Diagram and there realationship with each other

### 4.2.1 Hip and Knee Motors

The DYNAMIXEL XM430-W350 is chosen as a motor for the hip joint and the knee joint. The motor is chosen because it has a high torque to weight ratio and it has a high resolution of 4096 steps per revolution. The motor has a built-in driver and it can be controlled using a serial communication protocol. The motor has a built-in encoder that can be used to measure the position of the motor. The motor has a maximum torque of 3.5 Nm and a maximum speed of 46 RPM. The motor has a maximum current of 2.1 A and a maximum voltage of 12 V. The motor has a weight of 82 g and a size of 28.5 x 46.5 x 34 mm.


**Figure 4.2:** DYNAMIXEL XM430-W350

### 4.2.2 Wheel Motors

The SIMPLEX MOTION SC040A is chosen as a motor for the wheels. The motor is chosen since it has a output of 120W and 280 mNm torque at 4000rpm. The motor has a built-in driver and it can be controlled using RS485 serial communication protocol. The motor has position and speed control with torque limit. The motor has a maximum torque of 800 mNm.

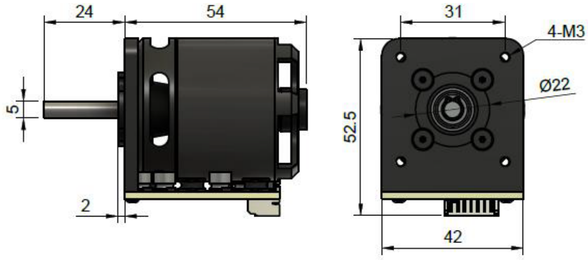



**Figure 4.3:** SIMPLEX MOTION SC040A

### 4.2.3 Raspberry Pi

The Raspberry Pi 4 compute module is chosen since it offers several advantages such as remote ssh connection via wifi. It can directly pass new control parameters to the microcontroller. It can be used to stream the video from the camera. The Raspberry Pi 4 compute module has a 64-bit quad-core ARM Cortex-A72 processor running at 1.5 GHz. It has 4 GB of LPDDR4-3200 SDRAM. The Raspberry Pi 4 compute module has a 32 GB eMMC Flash memory, a maximum current of 3 A and maximum voltage of 5.1 V.

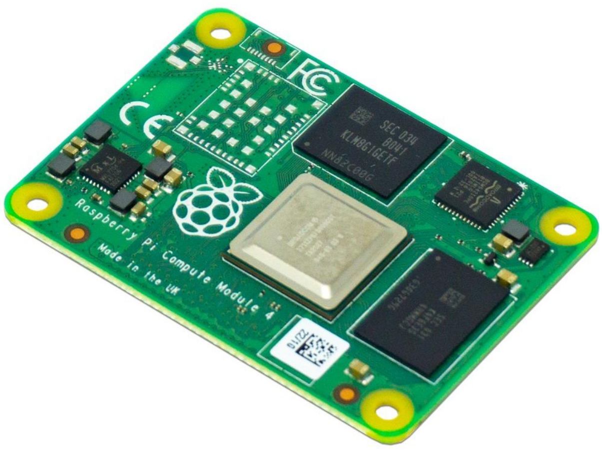



**Figure 4.4:** Raspberry Pi 4 compute module

### 4.2.4 Distance Sensor

The VL53L1X is chosen as a distance sensor for the robot. The sensor is chosen because it has a high accuracy and it has a high range. The sensor has a maximum range of 4 m. The sensor has a maximum accuracy of 1 mm. The sensor has a maximum field of view of 27 degrees. The sensor has a maximum current of 20 mA. The sensor has a maximum voltage of 3.6 V. The sensor has a weight of 1.6 g and a size of 4.4 x 2.4 x 1.0 mm.

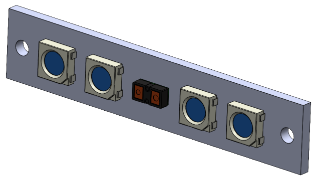


**Figure 4.5:** VL53L1X

### 4.2.5 Camera

The Raspberry Pi Camera Module V2 is chosen as a camera for the robot. The camera is chosen because it has a high resolution and it has a high frame rate. The camera has a resolution of 8 MP. The camera has a frame rate of 30 fps. The camera has a maximum current of 250 mA. The camera has a maximum voltage of 3.3 V. The camera has a weight of 3.4 g and a size of 25 x 23 x 9 mm.

**Figure 4.6:** Raspberry Pi Camera Module V2

### 4.2.6 Battery

The SLS XTRON 3000MAH 4S1P 14.8V 35C LIPO BATTERY is chosen as a battery for the robot. The battery is chosen because it has a high capacity and it has a high discharge rate. The battery has a capacity of 3000 mAh. The battery has a discharge rate of 35 C. The battery has a maximum current of 105 A. The battery has a maximum voltage of 16.8 V. The battery has a weight of 300 g and a size of 135 x 42 x 30 mm.


**Figure 4.7:** SLS XTRON 3000MAH 4S1P

## 4.3 Circuit Design

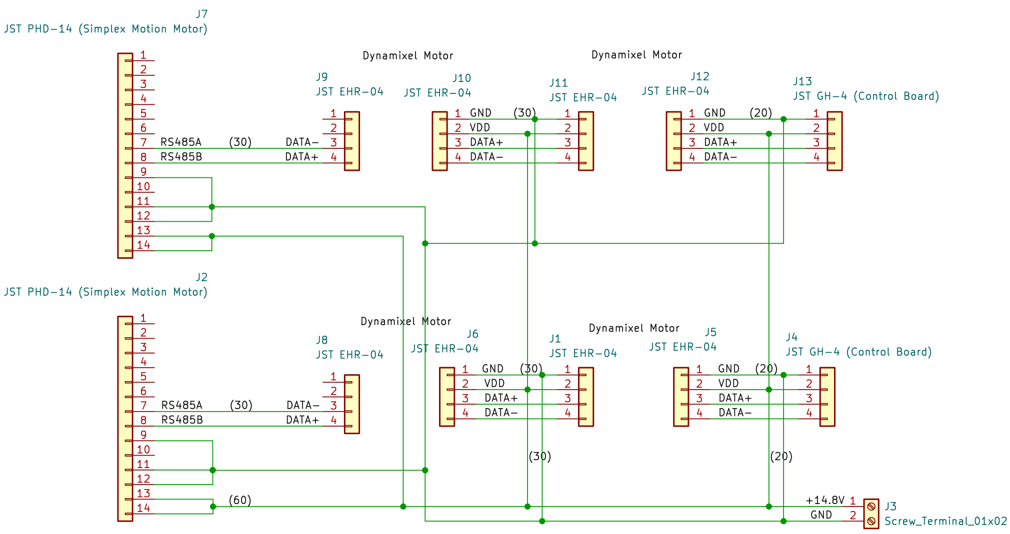



**Figure 4.8:** Electronic Design

The wiring tree of the electronic components is shown in figure 4.8 where it shows the connection between the different components and the microcontroller.The first motor of each leg is connected to the microcontroller and the rest of the motors are connected in series to the first motor of using daisy chain connection. the motors have the drivers board intigrated so they only need the control signal comimg from the microcontroller. The motor voltage is supplied from the power management board.

## 4.4 Robot Hub Board

The robot hub board is the main board that connects all the components together. The board includes the stm32h7 microcontroller that controls the motors and the sensors. The board also acts as a carrier board for the raspberry pi compute module, where different peripherals can be connected to the raspberry pi via the robot hub board. High-density edge connector is used to connect the raspberry pi to the robot hub board. UART is used to communicate between the microcontroller and the raspberry pi.

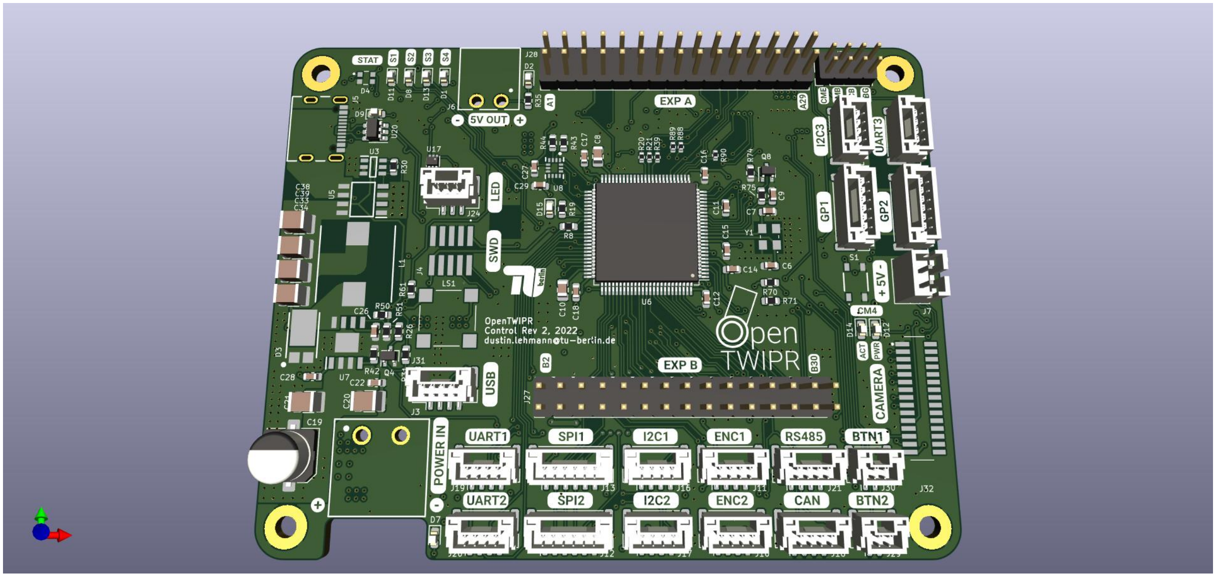



**Figure 4.9:** Robot Hub Board

## 4.5 Power Management

The power mangment board is responsible for managing the power distribution between the different components. The board has a stm32l4 microcontroller that monitors the battery voltage and the current consumption of the motors. The board has a 15 A fuse that protects the battery from over current. The board has a 4-cell battery undervoltage, overvoltage, overcurrent protection. The board has a cell balancing circuit that balances the voltage between the cells of the battery. The board has a high power MOSFET switches that controls the power distribution between the different components.

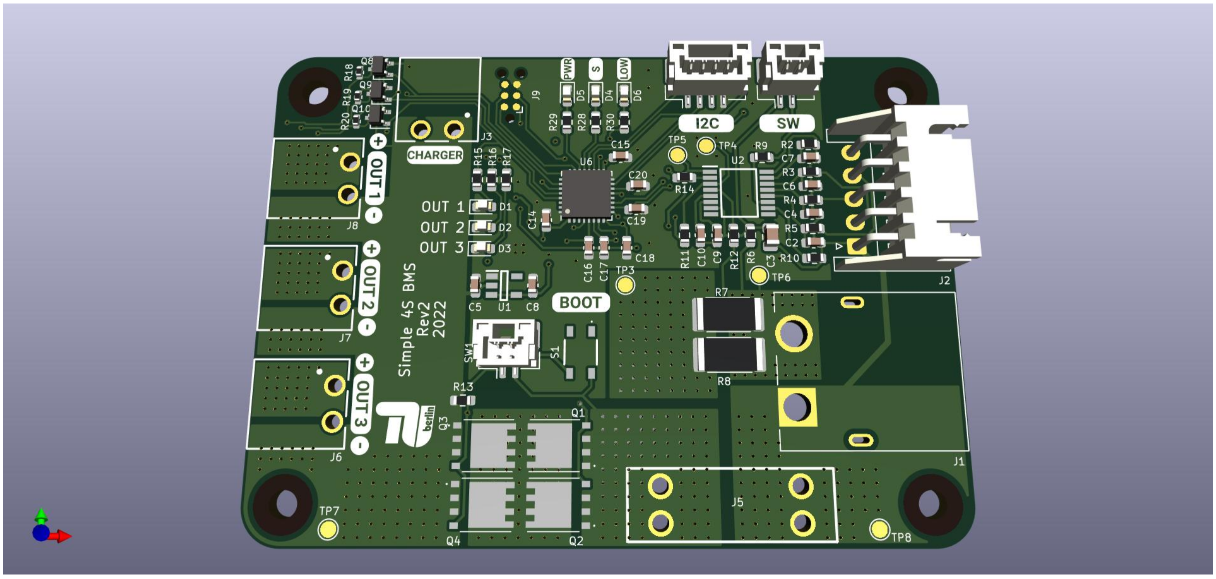


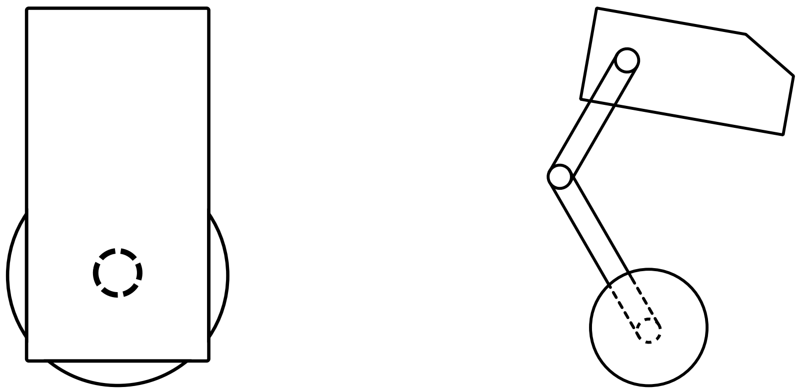

**Figure 4.10:** Power Management Board

# 5 Modelling and Simulation

In this pivotal chapter, we meticulously derive the center of gravity and the moment of inertia for the two-wheeled self-balancing robot. These parameters are the linchpins of our dynamic analysis, serving as the critical variables within the equations of motion that govern the robot's behavior. By calculating these values with precision, we can substitute them into our dynamic equations, thereby tailoring the model to reflect the true dynamics of the robot. This process not only enhances the accuracy of our simulations but also ensures that the control strategies developed are based on a robust and representative model of the robot's physical capabilities. The careful derivation of these parameters is a testament to the thoroughness of our approach, ensuring that the resulting model is both reliable and predictive of the robot's real-world performance.

## 5.1 Mathematical Modelling

The two-wheeled Inverted Pendulum robot model is as shown in the figure where it consists of two legs that include a hip and knee joints as well as wheels at the end of each leg. The robot needs to constantly adjust its posture to be able to maintain the balance, just like how the human being balances when standing on the feet. This new TWIPR model is the latest iteration in the evolution of its predecessor.



**(a)** Old model

**(b)** New model

**Figure 5.1:** Comparison between the old and new models

### 5.1.1 2D Dynamics

The scope of this project is limited to 2D dynamics. However, the model can be extended to 3D dynamics and controller synthesis for the 3D dynamics in the future. The 2D dynamics is considered for simplification purposes and to reduce the complexity of the model. The 2D dynamics modeling takes into account the robot's movement in the x-y plane and the rotation of the knee joint, hip joint, and the wheels around the z-axis. The 2D dynamics modeling considers one leg and half of the body mass.

### 5.1.2 Assumptions and Parameters

Many assumptions were made to simplify the model and reduce the complexity of the calculations.The weights and the lengths of the links are constant, and the angles $\phi_2$, $\phi_3$ are input variables that can be controlled to adjust the robot's posture.

**Table 5.1:** Parameters of the mechanical system

| Parameter | Value | Description |
|---|---|---|
| $L_1$ | 0.017 m | Length of the Wheel knee Link |
| $L_2$ | 0.017 m | Length of the Body knee Link |
| $L_{C1}$ | 0.013 m | Distance between the Wheel Joint and the center of mass of the Wheel knee Link |
| $L_{C2}$ | 0.013 m | Distance between the Knee Joint and the center of mass of the Body knee Link |
| $L_{C3}$ | 0.013 m | Distance between the Hip Joint and the center of mass of the body |
| $m_{L1}$ | 0.1 kg | Mass of the Wheel knee Link |
| $m_{L2}$ | 0.1 kg | Mass of the Body knee Link |
| $m_K$ | 0.1 kg | Mass of the Knee Joint |
| $m_B$ | 0.1 kg | Mass of the Body |
| $L_G$ | - | Distance between the center of gravity and the Wheel Joint |
| $\theta$ | - | Angle between the center of gravity and the Wheel knee Link |
| $\phi_1$ | - | Angle between the Body knee Link and the $y_B$ vertical axis of the body |
| $\phi_2$ | - | Angle between the Wheel knee Link and the Body knee Link |
| $\phi_3$ | - | Angle between the Body knee Link and the center of mass of the body |
| $\phi_4$ | - | Angle between the Body knee Link and the $X_B$ horizontal axis of the body |

### 5.1.3 Center of Gravity

The calculation of the center of gravity is crucial for the dynamic analysis of the robot. The center of gravity is calculated by taking into account the weights of the links, the knee joint, the body and the distances between these weights and the local coordinate system of the robot. The stability of the robot is directly impacted by the center of gravity position, which also affects how it reacts to forces and moments from the outside world. The following equations are used to calculate the center of gravity location referencing the local coordinate system of the robot.

$$x_{CG} = \frac{m_{L1} \cdot 0 + m_K \cdot 0 + m_{L2} \cdot L_{C2} \cdot \sin(\phi_2) + m_B \cdot (L_2 \cdot \sin(\phi_2) + L_{C3} \cdot \sin(\phi_2 + \phi_3))}{m_{L1} + m_{L2} + m_K + m_B} \tag{5.1}$$

$$y_{CG} = \frac{m_{L1} \cdot L_{C1} + m_K \cdot L_1 + m_{L2} \cdot (L_1 + L_{C2} \cdot \cos(\phi_2)) + m_B \cdot (L_1 + L_2 \cdot \cos(\phi_2) + L_{C3} \cdot \cos(\phi_2 + \phi_3))}{m_{L1} + m_{L2} + m_K + m_B} \tag{5.2}$$

$$L_G = \sqrt{x_{CG}^2 + y_{CG}^2} \tag{5.3}$$

$$\theta = \arctan\left(\frac{x_{CG}}{y_{CG}}\right) \tag{5.4}$$

### 5.1.4 Moment of inertia

Moment of inertia calculations

$$L_m = \sqrt{L_1^2 + L_{C2}^2 - L_1 L_{C2} \cos(180 - \phi_2)} \tag{5.5}$$

$$L_x = \sqrt{L_1^2 + L_2^2 - L_1 L_2 \cos(180 - \phi_2)} \tag{5.6}$$

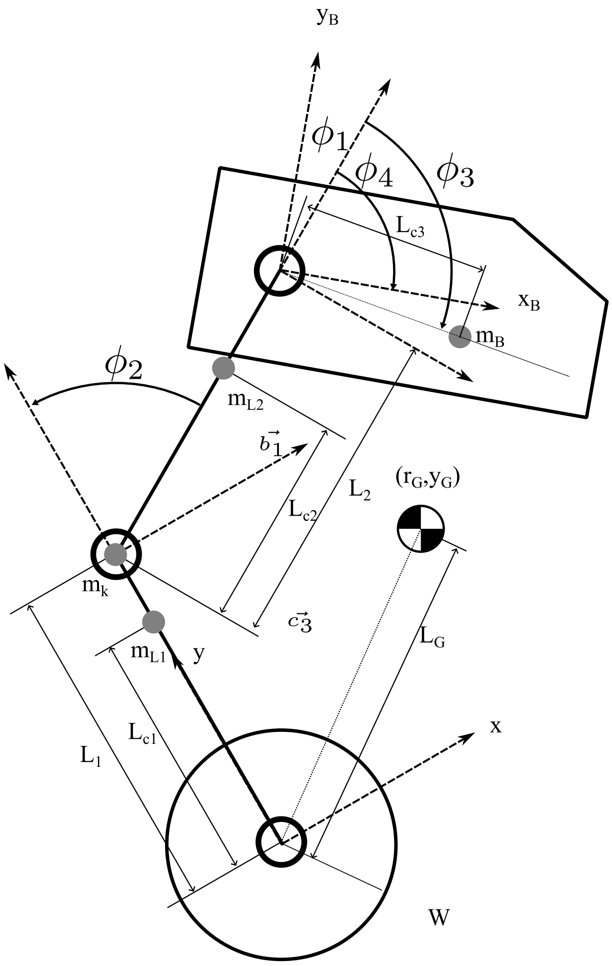



**Figure 5.2:** Figure illustrating the mechanical model with the local coordinate system to determine the center of gravity.

$$\alpha = \cos^{-1}\left(\frac{l_2^2 + l_x^2 - l_1^2}{2l_2 l_x}\right) \tag{5.7}$$

$$L_b = \sqrt{L_x^2 + L_{C3}^2 - LxL_{C3}\cos(180 - \alpha - \phi_3)} \tag{5.8}$$

$$I_{L1} = \frac{1}{12} m_{L1}(a_1^2 + b_1^2) \tag{5.9}$$

$$I_{L2} = \frac{1}{12} m_{L1}(a_2^2 + b_2^2) \tag{5.10}$$

$$I_K = \frac{1}{2} m_K R_m^2 \tag{5.11}$$

$$I_B = \frac{1}{12} m_B(a_B^2 + b_B^2) \tag{5.12}$$

$$I = I_{L1} + m_{L1}L_{C1}^2 + I_K + m_K L_1^2 + I_{L2} + m_{L2}L_m^2 + I_B + m_B L_b^2 \tag{5.13}$$

In order to predict the behavior of the robot under different settings, the modeling procedure entails constructing mathematical representations of the robot's dynamics and control systems.

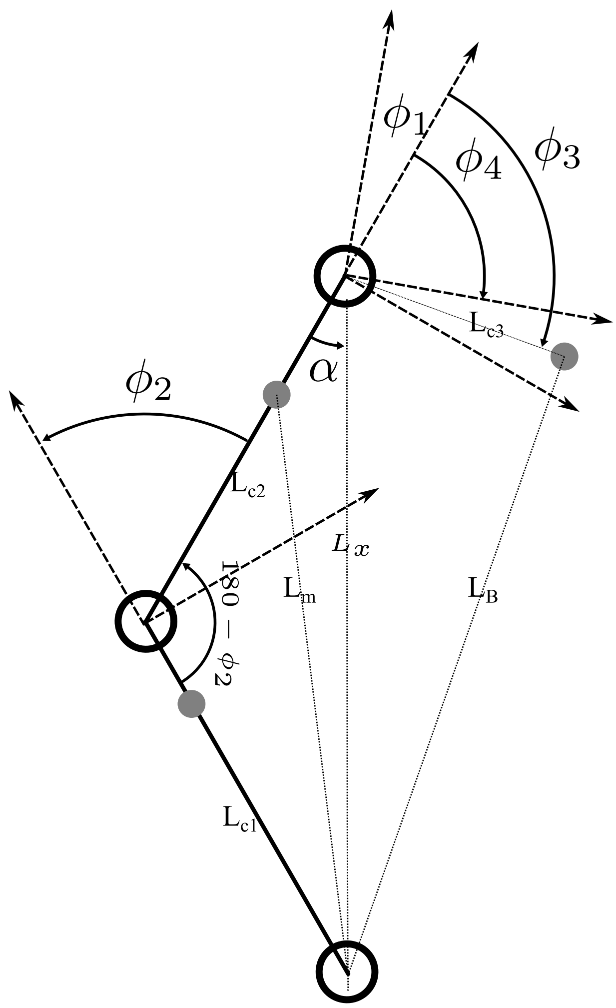



**Figure 5.3:** Schematic representation detailing the requisite angles and lengths for calculating the moment of inertia.

### 5.1.5 Equation of motion

Equation of motion

### 5.1.6 Dynamics of the Two-Wheeled Inverted Pendulum Robot

Given the functions $B_i : \mathbb{R} \to \mathbb{R}$, $C_{ij} : \mathbb{R} \to \mathbb{R}$, $D_{ij} : \mathbb{R} \to \mathbb{R}$, and $V_i : \mathbb{R} \to \mathbb{R}$, $i,j \in \{1,2,3\}$, the equations of motion are given by:

$$\ddot{s} = \frac{\sin(\Theta)}{V_1(\Theta)}\left(-C_{11}(\Theta)g + C_{12}\dot{\Theta}^2 + C_{13}(\Theta)\dot{\psi}^2\right) - \frac{D_{11}(\Theta)}{V_1(\Theta)}\dot{s} + \frac{D_{12}(\Theta)}{V_1(\Theta)}\dot{\Theta} + \frac{B_1(\Theta)}{V_1(\Theta)}(\tau_L + \tau_R) \tag{5.14}$$

$$\ddot{\Theta} = \frac{\sin(\Theta)}{V_1(\Theta)}\left(C_{21} - C_{22}(\Theta)\dot{\Theta}^2 - C_{23}(\Theta)\dot{\psi}^2\right) + \frac{D_{21}(\Theta)}{V_1(\Theta)}\dot{s} - \frac{D_{22}(\Theta)}{V_1(\Theta)}\dot{\Theta} - \frac{B_2(\Theta)}{V_1(\Theta)}(\tau_L + \tau_R) \tag{5.15}$$

$$\ddot{\psi} = \frac{\sin(\Theta)}{V_2(\Theta)}\left(C_{31}(\Theta)\dot{\Theta}\dot{\psi} - C_{32}(\Theta)\dot{\psi}\dot{s}\right) - \frac{D_{33}(\Theta)}{V_2(\Theta)}\dot{\psi} - \frac{B_3}{V_2(\Theta)}(\tau_L - \tau_R) \tag{5.16}$$

The equations of motion are derived in [44]. The functions $B_i : \mathbb{R} \to \mathbb{R}$, $C_{ij} : \mathbb{R} \to \mathbb{R}$, $D_{ij} : \mathbb{R} \to \mathbb{R}$, and $V_i : \mathbb{R} \to \mathbb{R}$, $i,j \in \{1,2,3\}$, are given by:

$$C_{11}(\Theta) = m_B^2 l^2 \cos(\Theta) g, \tag{5.17}$$

$$C_{12} = (I_2 + m_B l^2) m_B l, \tag{5.18}$$

$$C_{13}(\Theta) = (I_2 + m_B l^2) m_B l + m_B l (I_3 - I_1 - m_B l^2) \cos^2(\Theta), \tag{5.19}$$

$$C_{21} = (m_B + 2m_W + \frac{2J}{r^2}) m_B l, \tag{5.20}$$

$$C_{22}(\Theta) = m_B^2 l^2 \cos(\Theta), \tag{5.21}$$

$$C_{23}(\Theta) = m_B^2 l^2 + (m_B + 2m_W + \frac{2J}{r^2})(I_3 - I_1 - m_B l^2) \cos(\Theta). \tag{5.22}$$

$$C_{31}(\Theta) = 2(I_3 - I_1 - m_B^2) \cos(\Theta), \tag{5.23}$$

$$C_{31} = m_B l, \tag{5.24}$$

$$D_{11}(\Theta) = \frac{(I_2 + m_B^2) 2c_\alpha}{r^2} - \frac{m_B \cos(\Theta) 2c_\alpha}{r}, \tag{5.25}$$

$$D_{12}(\Theta) = \frac{(I_2 + m_B^2) 2c_\alpha}{r} - 2m_B \cos(\Theta) c_\alpha, \tag{5.26}$$

$$D_{21}(\Theta) = \frac{(m_B + 2m_W + \frac{2J}{r^2}) 2c_\alpha}{r} + \frac{m_B \cos(\Theta) 2c_\alpha}{r^2}, \tag{5.27}$$

$$D_{22}(\Theta) = \frac{(m_B + 2m_W + \frac{2J}{r}) 2c_\alpha + m_B \cos(\Theta) 2c_\alpha}{r}, \tag{5.28}$$

$$D_{33}(\Theta) = \frac{d^2}{2r^2 c_\alpha}, \tag{5.29}$$

$$B_1 = \frac{(I_2 + m_B^2)\frac{1}{r} + m_B \cos(\Theta)}{r}, \tag{5.30}$$

$$B_2 = \frac{m_B l}{r} - \cos(\Theta) + m_B + 2m_W + \frac{2J}{r^2}, \tag{5.31}$$

$$B_3 = \frac{d}{2r}, \tag{5.32}$$

$$V_1 = (m_B + 2m_W + \frac{2J}{r^2})(I_2 + m_B^2) - m_B^2 \cos^2(\Theta), \tag{5.33}$$

$$V_2 = I_3 + 2K + (m_W + \frac{J}{r^2})\frac{d^2}{2} - (I_3 - I_1 - m_B^2) \sin^2(\Theta). \tag{5.34}$$

**Table 5.2:** Parameters of the mechanical system

| Parameter | Value | Description |
|---|---|---|
| $m_B$ | 2.5 kg | mass of the pendulum body |
| $m_W$ | 0.636 kg | mass of a wheel |
| $l$ | 0.026 m | distance between the wheel axis and the pendulum's center of gravity |
| $d$ | - | distance between the two wheels |
| $J$ | $5.175e^{-4}$ kgm$^2$ | moment of inertia of a wheel w.r.t. Reference frame {C} in direction of $c_2$ |
| $K$ | - | moment of inertia of a wheel w.r.t. reference frame {C} in direction of $c_3$ |
| $I_1$ | - | moment of inertia of pendulum's body w.r.t. Reference frame {B} in direction of $b_1$ |
| $I_2$ | 0.0165 kgm$^2$ | moment of inertia of pendulum's body w.r.t. Reference frame {B} in direction of $b_2$ |
| $I_3$ | - | moment of inertia of pendulum's body w.r.t. Reference frame {B} in direction of $b_3$ |
| $c_\alpha$ | $4.630e^{-4}$ Nms | viscous friction coefficient |

## 5.2 Numerical Simulation

- **Discrete Numerical Simulation:** Elaboration on the process of discrete numerical simulation, including the discrete double integration method to arrive at the state vector.

## 5.3 Simulation Environment

- **Overview of the Simulation Environment:** A brief overview of the simulation environment, referencing David's Bachelor Thesis for details.
- **Integration of the Model:** Description of integrating the new TWIPR model into the simulation environment and a summary of the individual components of the model.

## 5.4 Controller Synthesis

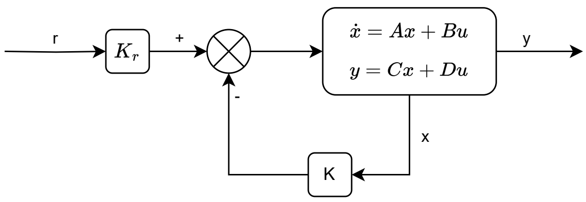



**Figure 5.4:** Block diagram of a state feedback controller

In the realm of modern control theory, State-Space controllers are a pivotal component for managing the behavior of complex systems.Among the most prominent of these controllers are the Linear Quadratic Regulator (LQR) and Pole-Placement controllers, both of which are robust and effective in various applications. This section provides an overview of these controllers and their application to the legged TWIPR robot.

### 5.4.1 Linear Quadratic Regulator (LQR)

The Linear Quadratic Regulator (LQR) is a state-space controller that uses a quadratic cost function to determine the optimal control input for a given system. It aims to minimize the cost function by adjusting the control input, thereby ensuring that the system's state converges to the desired state. The LQR controller is defined by the following equation:

$$u(t) = -Kx(t) \tag{5.35}$$

where $u(t)$ is the control input, $x(t)$ is the state vector, and $K$ is the gain matrix. The gain matrix is calculated using the following equation:

$$K = R^{-1}B^T P \tag{5.36}$$

where $R$ is the control weight matrix, $B$ is the input matrix, and $P$ is the solution to the Riccati equation:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{5.37}$$

where $A$ is the state matrix and $Q$ is the state weight matrix. The state matrix, state weight matrix, and control weight matrix are defined as follows:

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.38}$$

$$R = \begin{bmatrix} 1 \end{bmatrix} \tag{5.39}$$

The LQR controller is implemented in the simulation environment using the *Python Control Systems Library* [1].

### 5.4.2 Pole-Placement Technique

The Pole-Placement technique is a state-space controller that uses the Ackermann formula to determine the optimal control input for a given system. It aims to place the poles of the system at the desired locations by adjusting the control input, thereby ensuring that the system's state converges to the desired state.placing the poles is not initiutive for high order systems or systems with multiple actuators. The Pole-Placement controller is defined by the following equation:

$$u(t) = -Kx(t) \tag{5.40}$$

where $u(t)$ is the control input, $x(t)$ is the state vector, and $K$ is the gain matrix. The gain matrix is calculated using the following equation:

$$K = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix} \tag{5.41}$$

where $k_1$, $k_2$, and $k_3$ are the gains for the first, second, and third states, respectively. The gains are calculated using the following equation:

$$k_i = \frac{1}{b_i} \left( \sum_{j=0}^{n-1} a_{n-j} \alpha_{i+j} - \alpha_i \right) \tag{5.42}$$

where $a_i$ is the coefficient of the characteristic polynomial, $b_i$ is the coefficient of the denominator polynomial, and $\alpha_i$ is the desired location of the $i$th pole. The characteristic polynomial and denominator polynomial are defined as follows:

$$a_i = \begin{cases} 1 & i = 0 \\ 0 & i \neq 0 \end{cases} \tag{5.43}$$

$$b_i = \begin{cases} 1 & i = 0 \\ 0 & i \neq 0 \end{cases} \tag{5.44}$$

The Pole-Placement controller is implemented in the simulation environment using the *Python Control Systems Library* [1].

### 5.4.3 Configuration Specificity

These controllers are specific to one configuration for the knee angle and hip angle of the robot. Retuning the controllers is required when the configuration changes.

```
    def change_knee_angle(self):
```

```
        self.agent1.set_leg_angles(hip_angle=self.agent1.dynamics.model.hip_angle,
                                   knee_angle=self.agent1.dynamics.model.knee_angle + deg2rad(5))

    def change_hip_angle(self):
        self.agent1.set_leg_angles(hip_angle=self.agent1.dynamics.model.hip_angle + deg2rad(5),
                                   knee_angle=self.agent1.dynamics.model.knee_angle)
```

**Lst. 5.1:** Changing the knee and hip angles of the robot

where The code 5.1 Shows the functions that changes the knee and hip angles of the robot.

```
    def set_leg_angles(self, knee_angle: float, hip_angle: float):
        self.dynamics.model.knee_angle = knee_angle
        self.dynamics.model.hip_angle = hip_angle
        self.linear_dynamics = TWIPR_3D_Linear(self.dynamics.model, Ts, self.poles, self.eigenvectors)
        self.state_ctrl_K = np.hstack((np.zeros((2, 1)), self.linear_dynamics.K))
```

**Lst. 5.2:** Changing the knee and hip angles in the dynamics and retuning the controller

In the above code 5.2 It hows the function that changes the knee and hip angles in the dynamics and retunes the controller.

## 5.5 Simulation Analysis

### 5.5.1 Controller Responses

In order to analyze the performance of the controllers, we conducted a series of simulations to evaluate their responses to various inputs.

### 5.5.2 Impact of Non-Retuning

### 5.5.3 Influence of Leg Configuration

### 5.5.4 Controller Setting Comparisons

+

# 6 Mechanical Assembly

In this chapter, the mechanical assembly of the robot is discussed. The mechanical assembly is the process of putting together the mechanical components of the robot. The mechanical components include the chassis, the wheels, the motors, the encoders, the battery, the IMU, the camera, the Raspberry Pi, the Arduino, the motor controller, the motor driver, the motor encoder driver, the motor encoder, the motor encoder cable, the

## 6.1 Components Overview

- Detailed description of all mechanical components used in the robot.
- Source or method of fabrication for each component (e.g., machined parts, 3D printed elements).

## 6.2 Assembly Process

- Step-by-step explanation of the assembly process.
- Tools and techniques used in the assembly.
- Assembly sequence and rationale behind it.

## 6.3 Integration of Mechanical and Electronic Systems

- Discussion on how mechanical components interface with electronic systems.
- Challenges faced in integration and strategies employed to overcome them.

## 6.4 Troubleshooting and Problem Solving

- Discussion of any unexpected challenges or issues faced during assembly.
- How these issues were diagnosed and resolved.

## 6.5 Safety Considerations

- Safety measures taken during the assembly process.
- Design considerations for ensuring the operational safety of the robot.
- **Discrete Numerical Simulation:** Elaboration on the process of discrete numerical simulation, including the discrete double integration method to arrive at the state vector.

**Note 6.1**
the bolts used the problem to shorten them so that it wouldn't touch the motor

# 7 Firmware and Testing

- Overview of the chapter's content and its significance in the context of the overall project.
- Briefly state the objectives of firmware development and testing procedures.

## 7.1 Firmware Development

- Discuss the development environment and tools used for firmware programming.
- Detail the architecture and design of the firmware, including flowcharts or state diagrams if applicable.
- Explain the implementation of key functionalities such as control algorithms, sensor integration, and actuator management.

## 7.2 Programming Languages and Tools

- List and describe the programming languages used for firmware development.
- Mention any specific software tools, libraries, or frameworks employed.

## 7.3 Integration with Hardware

- Discuss how the firmware interacts with and controls the hardware components.
- Explain any challenges encountered in integration and how they were resolved.

## 7.4 Testing Framework and Methodology

- Outline the testing framework used to validate the firmware.
- Describe the methodology for functional testing, including unit tests, integration tests, and system-level tests.

## 7.5 Test Cases and Scenarios

- Present the results of the testing procedures.
- Analyze these results, highlighting successful areas and identifying any issues or bugs discovered.

## 7.6 Debugging and Troubleshooting

- Discuss the process of debugging and troubleshooting encountered problems.
- Explain how issues were diagnosed and resolved.

**Note 7.1**
the bolts used the problem to shorten them so that it wouldn't touch the motor

# 8 Safety

- hardware.
  - motors
    - Hip motor is enclosed in the body.
    - the knee motors are enclosed in cover.
    - the wheels motors have a
  - wiring
    - where the wires are neatly fastened in a predetermined route so that it wouldn't be caught in the robot movement which would cause damage to the robot and also to protect them from damage.
    - the correct type were used to avoid overheating upon the draw of current from the divers.
    - Rs-485 connection between motors where used to avoid additional cables.
  - proximity sensor
    - placed in the front and back side of the robot would help to detect crashing in trivial positions such as running into a wall and that could be easily prevented by proximity sensor monitoring the distance between the robot the and the obsticals in it direction.
  - body safety additional parts in case of impact to protect the internal components.
- software.

Questions to answer in each section

1. what is going to be in here?
2. how long or how elaborate?
3. what is the purpose (the take home message)?

Table of content Draft

1. Introduction
   a) Background and Motivation
      i. Discuss the evolution and significance of robotics in various industries.
      ii. Emphasize the need for advancements in robotic stability and mobility.
   b) Problem Statement
      i. Define the specific challenges in designing a legged self-balancing robot.
   c) Objectives(Outline the primary goals of the thesis)

2. Literature Review
    a) Overview of Robotics
    b) Previous Work in Self-Balancing Robots
    c) Control Strategies
        i. what is going to be in here? a comparison of the control strategies used.
        ii. how long or how elaborate? not too detailed but with more explanation of the control theory used in the project
        iii. what is the purpose (the take home message)? pros and cons of the different and why would we prefer one of them over the other depending on the applications
3. Design and Development of the Robot
    a) Mechanical Design
        i. Initial calculations
            A. what is going to be in here? -> torque initial calculations
            B. how long or how elaborate? -> two or three scenarios
            C. what is the purpose (the take home message)? for choosing the correct motors
        ii. design
            A. what is going to be in here?-> CAD design and the explanations of the challenges
            B. how long or how elaborate? detailed explanations of the reason behind the design decision
            C. what is the purpose (the take home message)? assembling the robot with fitting parts to match the new model requirements
        iii. Modeling
            A. what is going to be in here?-> the figures for the new model and the new COG and MOI calculations and the equations of motion.
            B. how long or how elaborate? 4 to 5 pages explaining the equations in details
            C. what is the purpose (the take home message)? showing the calculations for the new model and it would influence the equations of motion.
    b) Electrical Design
        i. what is going to be in here? Component diagram showing the choice of all the components and there intended use and why we chose each of these components
        ii. how long or how elaborate? detailed explanation of the requirement boards for operating the robot, the choice of components based calculations for the motors.
        iii. what is the purpose (the take home message)? show how the Electrical design is configured in the optimal way to operate the robot
    c) Software and Control
        i. Control Algorithm

A. what is going to be in here?->flowchart of the Control Algorithm

B. how long or how elaborate?-> detailed explanation of the used control theory

C. what is the purpose (the take home message)? -> how the control is implemented

ii. Firmware

d) Safety

i. what is going to be in here? different design changes for safety measures(motors covers, wire routing, body bumper, distance sensor , algorithm safety, electrical safety )

ii. how long or how elaborate? 1 or two pages max that include the

iii. what is the purpose (the take home message)? the safety measures taken to minimize crashes, failure

4. Experimental Setup and Methodology

a) Simulation Environment

b) Physical Prototype Testing

c) Data Collection and Analysis

5. Results and Discussion

a) Simulation Results

b) Real-world Performance

c) Comparison and Analysis

6. Conclusion and Future Work

a) Summary of Findings

b) Contributions

c) Recommendations for Future Research

content..

# 9 Random(Erase)

This comprehensive chapter unfolds the intricate details of the new design of our two-wheeled self-balancing robot, an advanced piece of engineering that incorporates additional degrees of freedom to enhance its movement capabilities. Through an exploration of the various motions the robot can perform, we delve into the intricate design considerations of each component, ensuring that they align with the overall functional and aesthetic vision.

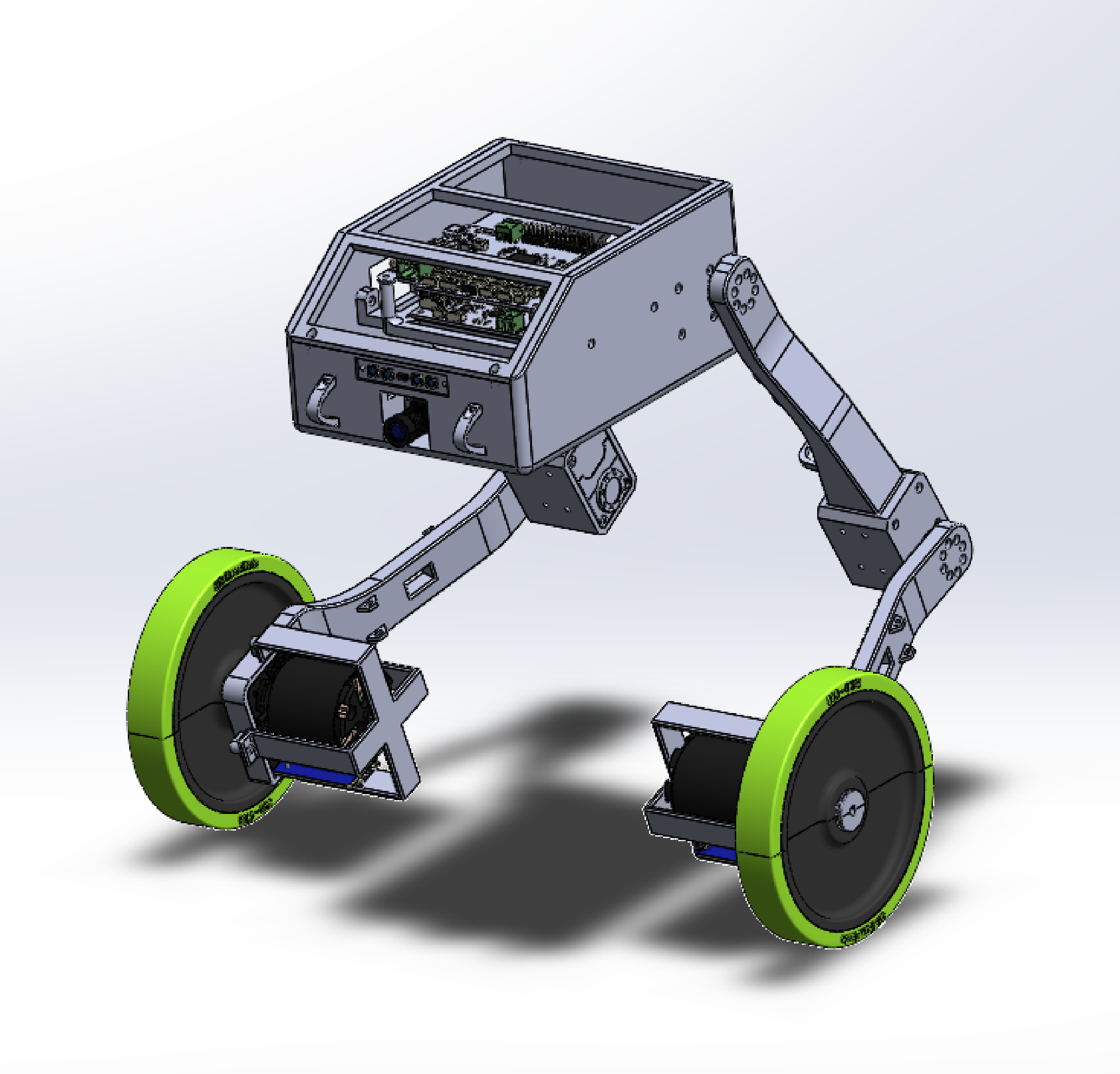


**Figure 9.1:** Schematic representation detailing the requisite angles and lengths for calculating the moment of inertia.

- The Robot new design.
- The added degrees of freedom.
- Different motions that can be performed.
- Discussing each component of the robot and things taken into account while designing it.
  - Body
    - overall design inspiration
    - For the body: the consideration for including all the necessary components in a compact form is to optimize the use of space and at the same time distribute the weight equally.
    - The body includes the hip motors, the battery, camera, sensor, a rack that includes the motors drivers board, the micro-controller board attached to the Raspberry Pi.
    - fastening features that were specifically designed in order to easily mount the battery, organize the cable between the boards and the rest of the robot parts.
    - Features for modular design and easy printing.
  - Thigh
    - curvature of that joint to give room for the motors
    - cable management
    - motor cover to insure its fixation.
  - Calf
    - curvature of that joint and the thigh joint combined make enough room for the wheel motor so the it have clearance from the body.
    - cable management
    - motor mount and additional frame.

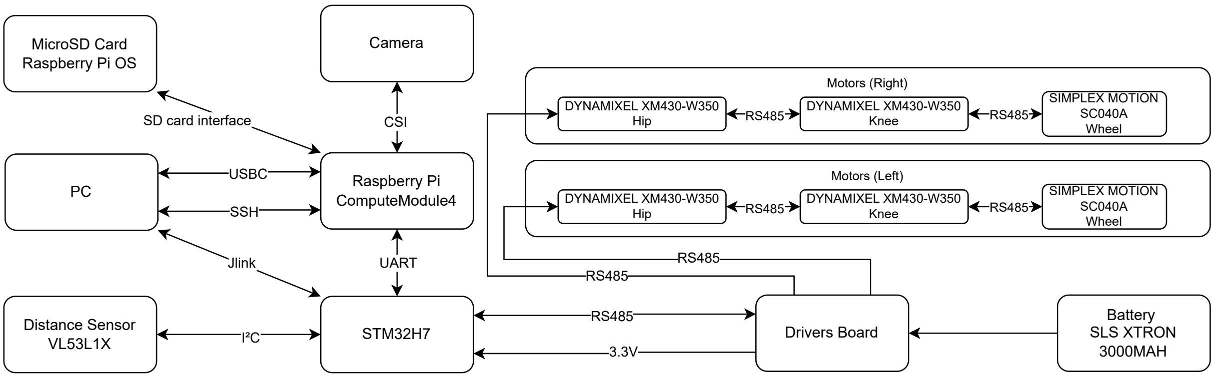



**Figure 9.2:** Schematic representation detailing the requisite angles and lengths for calculating the moment of inertia.

- electronic components
  - Motors taking into count the needed torque and speed
  - comparison between the BLDC and Geared Robotic
  - RS-485 communications protocol compared to others
    - for the knee and hip motors high torque and low speed is needed.(calculations the show the weights and the needed torques)
    - for the wheel motors high speed and low torque is needed.
  - Boards
    - STM high performance h7 constum board connected with rasberrypi
    - driver board that provide the power for 6 motors.
  - modules
    - camera module.
    - distance sensor.

# Bibliography

[1] S. Fuller, B. Greiner, J. Moore, R. Murray, R. van Paassen, and R. Yorke. The python control systems library (python-control). In *60th IEEE Conference on Decision and Control (CDC)*, pages 4875–4881. IEEE, 2021. URL https://python-control.org.