DEVELOPMENT OF A STATIONARY UPPER AND LOWER EXTREMITIES REHABILITATION SYSTEM

CHONG YING HANG

UNIVERSITI TUNKU ABDUL RAHMAN

DEVELOPMENT OF A STATIONARY UPPER AND LOWER EXTREMITIES REHABILITATION SYSTEM

CHONG YING HANG

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Biomedical Engineering with Honours

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

May 2025

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Name : CHONG YING HANG

ID No. : 020318-01-0419

Date : 2/5/2025

COPYRIGHT STATEMENT

This final year project report is submitted in partial fulfilment of the requirements for the degree of Biomedical Engineering with Honours at Universiti Tunku Abdul Rahman (UTAR). This final year project report represents the work of the author, except where due acknowledgement has been made in the text. No part of this final year project report may be reproduced, stored, or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the author or UTAR, in accordance with UTAR's Intellectual Property Policy.

ACKNOWLEDGEMENTS

I would like to express my gratitude to Mr. Chong Yu Zheng as my research supervisor and Dr. Chan Siow Cheng as my research co-supervisor for their invaluable advice, guidance and enormous patience throughout the development of the research.

Besides, I would also like to thank the faculty and the departmental members from Lee Kong Chian Faculty of Engineering and Science and Department of Mechatronic & Biomedical Engineering, for creating a pleasant working environment throughout my years in UTAR.

Lastly, I would like to express my heartfelt gratitude to my dear friends Jun Han, Yan Xing, Yen Mei, Jian Le, Xian Wei and all the participants of the functionality test for their invaluable help, friendship, and unwavering support, which have been instrumental to the success of this research project.

ABSTRACT

Rehabilitation for individuals with upper and lower limb impairments such as those resulting from stroke, spinal injuries, and neurological conditions remains a prolonged, resource-intensive process. Traditional therapy methods often rely heavily on the availability of skilled therapists and in-person clinical visits, leading to high treatment costs, inconsistent therapy sessions, and limited access, especially in low- and middle-income regions. Furthermore, current robotic rehabilitation systems are often prohibitively expensive, and many lack the flexibility to accommodate both upper and lower extremity rehabilitation in a single platform. These gaps highlight a critical need for a cost-effective, accessible, and versatile rehabilitation system that delivers programmable, repeatable, and safe therapeutic exercises. To address these challenges, this project presents the development of a stationary rehabilitation system designed to support the rehabilitation of both upper and lower limbs. The system utilizes stepper motors, linear actuators, and interchangeable limb supports to facilitate joint movements, while an ESP32 microcontroller enables system control and IoT connectivity. An MPU6050 sensor module (accelerometer and gyroscope) was integrated for real-time monitoring of acceleration and angular velocity during rehabilitation exercises. The prototype was fabricated using CNC machining and 3D printing and tested on eight healthy participants performing exercises at different speeds and across varied range of motion (ROM) profiles for both elbow and knee joints. Validation of motor rotational speed showed a low overall error of 2.82% across high (29.25 °/s) and low (13.50 °/s) speeds. ROM evaluations confirmed system repeatability, with a 4.75% average difference across varying speeds. Joint motion tracking using the MPU6050 sensor showed an average percentage error of 8.79% when validated against a Trigno IMU, demonstrating acceptable accuracy with slight variations depending on movement dynamics. In addition to ROM and motion tracking, dynamic kinematic data (acceleration and angular velocity) were captured for real-time motion analysis and rehabilitation performance assessment. The complete system was developed at an estimated cost of RM1600, providing a significantly lower cost alternative to commercial robotic rehabilitation devices. This project demonstrates promising potential to deliver accessible, quantifiable,

and remotely monitored therapy for upper and lower extremity impairments, suitable for use in clinical settings and at home.

Keywords: Rehabilitation, Biomechanics, Upper Extremities and Lower Extremities, Extension and Flexion, Accelerometer and Gyroscope.

Subject Area:
R856-857 Biomedical engineering. Electronics. Instrumentation

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

τ	Torque, Nm
r	Radius from axis of rotation to point of application of the force, m
F	Applied force, N
M	Moment, Nm
g	Gravity force, m/s^2
r_ℓ	Center of gravity of limb, m
m_s	Limb support structure mass, kg
m_ℓ	Limb mass, kg
CoM_ℓ	Center of mass of limb, m
τ_{min}	Minimum torque required by stepper motor.
$\tau_{BS\,min}$	Minimum torque required by stepper motor to lift body segment.
$\tau_{S\,min}$	Minimum torque required by stepper motor to lift support structure.
$\tau_{BS\,min}$	Minimum torque required by stepper motor to lift body segment, Nm
F_x	horizontal force, N
R_x	horizontal recaction force, N
F_y	vertical force, N
R_y	vertical recaction force, N
W_{F+H}	forearm and hand weight, kg
W_{S+F}	shank and foot weight, kg
CoM_F	center of mass location for forearm, m
CoM_S	center of mass location for shank, m
M_E	the moment at elbow joint, N
M_{knee}	the moment at knee joint, N
m_{F+H}	forearm and hand mass, N
m_{S+F}	shank and foot mass, N
EMG	Electromyography
IMU	Inertia Measurement Unit

CNC	Computer Numerical Control
IDE	Integrated Development Environment

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 General Introduction

Rehabilitation is a multidisciplinary process aimed at helping individuals regain optimal functioning following a health condition, injury, or disability (World Health Organization, 2024). This process can involve various forms of therapy, such as physical, cognitive, occupational, or psychosocial interventions, depending on the specific needs of the individual. The main objective of rehabilitation is to improve the individual's quality of life by enhancing independence and enabling them to reintegrate into daily activities and society (Wade, 2020). Rehabilitation services can be provided in hospitals, outpatient settings, or even at home, depending on the severity of the condition and the patient's preferences (World Health Organization, 2023).

Rehabilitation for the upper extremities focuses on restoring the functional use of the arms and hands through various therapeutic approaches. Key techniques include occupational therapy, robot-assisted rehabilitation, and the use of virtual reality (VR) systems. These methods are designed to help patients regain the ability to perform essential daily tasks, enhance range of motion, and improve fine motor skills (Saikaley Bsc et al., n.d.). In parallel, rehabilitation for the lower extremities targets the restoration of walking ability, balance, and overall mobility. This is achieved through interventions such as physical therapy, specialized gait training, and advanced technologies like exoskeletons. These approaches aim to strengthen the muscles, enhance coordination, and support functional leg movements for improved mobility and independence (Sarvenaz, M. et al., n.d.).

The demand for rehabilitation technologies is increasing due to several key factors. As the global population ages, the incidence of age-related conditions such as arthritis and stroke is on the rise, leading to a greater need for rehabilitation services (Hatem et al., 2016). Additionally, the prevalence of strokes and injuries, including those from accidents and sports, is becoming more common, making effective rehabilitation crucial for patients to regain their independence. Chronic conditions, such as Parkinson's disease and multiple sclerosis, also contribute to this growing demand, as these patients require

ongoing rehabilitation to manage symptoms and maintain their quality of life (Mang and Peters, 2021). Furthermore, advancements in rehabilitation technology, including robotic systems and virtual reality (VR), are enhancing the effectiveness and accessibility of rehabilitation, offering new opportunities for patient recovery (Hatem et al., 2016). Overall, rehabilitation systems play a critical role in helping individuals with motor impairments improve their quality of life, and the growing need for these technologies highlights the importance of continued innovation and investment in the field.

1.2 Importance of the Study

The development of a stationary upper and lower extremities rehabilitation system is essential for several reasons. Currently, conventional rehabilitation systems are often prohibitively expensive, limiting patients' ability to afford personal rehabilitation devices. As a result, patients are typically required to visit rehabilitation centers, which adds to their financial burden (e.g., the high costs associated with conventional systems like robotic therapy devices). Creating a cost-effective, stationary rehabilitation system that is affordable for patients is, therefore, vital. In addition to cost-effectiveness, such systems must support a wide range of conditions. By incorporating interchangeable features, a rehabilitation system can accommodate various needs, addressing both upper and lower extremity issues. For example, the system can assist in stroke recovery, as well as movement disorders such as cerebral palsy (Chong et al., 2016).

Another key benefit of developing this system is the potential to reduce the workload of healthcare professionals. With advancements in technology, these systems can automate routine rehabilitation tasks, allowing healthcare providers to focus on more complex cases, thus improving overall efficiency in rehabilitation centers. Furthermore, making the system affordable and accessible will enhance its availability across various rehabilitation centers, reducing the need for extensive travel for patients. Those living in remote areas could also benefit by purchasing an at-home system, enabling them to receive rehabilitation without traveling long distances. Finally, with the integration of advanced technology, these systems can offer enhanced monitoring of rehabilitation activities, ensuring more effective patient progress tracking.

1.3 Problem Statement

Rehabilitation for patients with upper and lower extremity impairments, such as those caused by stroke, spinal cord injuries, or other neurological conditions, is often a prolonged and resource-intensive process. Traditional methods heavily depend on skilled therapists and require significant physical effort from both patients and therapists. This reliance can lead to inconsistent therapy sessions, limited access to rehabilitation services, and suboptimal recovery outcomes. Furthermore, the lack of advanced stationary rehabilitation systems capable of delivering comprehensive, repetitive, and controlled exercises for both upper and lower extremities exacerbate these challenges. The financial burden of rehabilitation is a significant concern for both healthcare systems and patients. Globally, around 2.4 billion people live with health conditions that could benefit from rehabilitation, and this need is expected to grow due to an aging population and the rise in chronic diseases (World Health Organization, 2023). In many low- and middle-income countries, over 50% of individuals do not receive the rehabilitation services they require (World Health Organization, 2023). In Asia, rehabilitation costs are also considerable. Given these challenges, there is a critical need to develop an innovative stationary rehabilitation system that can support the effective rehabilitation of both upper and lower extremities. Such a system would improve patient outcomes and alleviate the strain on healthcare providers while making rehabilitation more accessible and affordable.

1.4 Aim and Objectives

This project aims to develop an affordable stationary rehabilitation system for both upper and lower extremities. The objectives of this project are as follows:

1. To design an affordable and customizable rehabilitation system that supports diverse mobility levels and provides a wide range of therapeutic exercises.
2. To incorporate the accelerometer and gyroscope module (MPU6050) sensor for real-time kinematic data acquisition, enabling measurement of joint movement during rehabilitation exercises.

3. To integrate technology for remote monitoring, enabling users' progress to be tracked and adjustments to be made without the need for frequent in-person interactions.

1.5 Scope and Limitation of the Study

This study focuses on the development of a stationary upper and lower extremities rehabilitation system designed for patients with impairments caused by conditions such as stroke, spinal cord injuries, and movement disorders. The system aims to integrate advanced technologies, including robotics, stepper motors, and sensors such as accelerometer and gyroscope sensor module, to provide controlled, repetitive exercises. It also includes features for remote monitoring and feedback, allowing to track rehabilitation progress and adjust rehabilitation plans accordingly. The system is intended to be customizable, with interchangeable components to cater to different patient needs, body types, and mobility levels. Moreover, it aims to be cost-effective, making rehabilitation accessible to a wider population, particularly in low- and middle-income regions. By providing a system that can be used in both clinical and home settings, the study seeks to reduce the need for frequent travel to rehabilitation centers and lighten the workload of healthcare professionals.

However, there are several limitations to this study. While the system is adaptable, it may not be suitable for patients with complex or severe impairments that require more specialized care. Another limitation is ensuring patient compliance, particularly in home settings where there may be no direct supervision from healthcare providers, leading to inconsistent use and suboptimal outcomes. Additionally, customization may also present challenges, as the system may not perfectly fit all patients, especially those with unique body types or specific rehabilitation needs. Furthermore, the integration of sensors for real-time monitoring raises concerns about data privacy and security, which may be difficult to fully address in the initial stages of development. Lastly, the system’s long-term maintenance and durability, especially for home users, could be a barrier, as regular upkeep may be required to ensure optimal functioning. By addressing both the scope and limitations, this study aims to provide a practical yet innovative solution for rehabilitation while recognizing the potential challenges in its development and implementation.

1.6 Contribution of the Study

This study provides a significant contribution to the field of rehabilitation engineering by developing a stationary system that supports therapy for both the upper and lower extremities. Unlike many existing systems that are limited in scope or too costly for widespread adoption, the proposed rehabilitation system is designed to be affordable, user-friendly, and versatile. Its dual-limb focus allows patients recovering from a wide range of neuromuscular conditions, including stroke and spinal cord injuries, to perform rehabilitative exercises within a single integrated system. The modularity of the design, with adjustable limb supports and customizable movement parameters, further enhances its adaptability for different body sizes and conditions.

Another major contribution lies in the system's integration of real-time data acquisition and remote monitoring features. By using the accelerometer and gyroscope module (MPU6050) sensor and the ESP32 microcontroller, the system can capture motion-related data such as joint angles and angular velocity and transmit this information wirelessly for remote access. This allows therapists or clinicians to track a patient's rehabilitation progress over time without the need for frequent in-person consultations, which is particularly beneficial in tele-rehabilitation contexts and for patients in rural or underserved regions.

The project also demonstrates how anthropometric data can be applied in the development of rehabilitation devices. Human limb segment lengths, masses, and center-of-mass locations were calculated based on established anthropometric models (de Leva, 1996), ensuring that the limb supports, and motorized components align with biomechanical principles. This ensures patient safety and improves the effectiveness of movement therapy by mimicking natural joint motion.

Finally, this study promotes innovation by demonstrating how off-the-shelf components such as stepper motors, linear actuators, and microcontrollers can be combined with 3D printing and custom circuitry to produce a functioning prototype at a relatively low cost. This not only benefits the healthcare system by offering a more scalable solution for rehabilitation but also serves as a foundation for future development, including the potential integration of

electromyography (EMG), artificial intelligence, or machine learning for adaptive therapy. The prototype and system architecture developed in this project thus serve both practical and academic purposes, paving the way for further research in affordable and intelligent rehabilitation technologies.

1.7 Outline of the Report

This report is structured into five chapters, each addressing a core component of the research and development process. Chapter 1 introduces the background of the project, emphasizing the growing need for accessible rehabilitation technologies. It outlines the research problem, objectives, and scope, focusing on the development of a cost-effective, dual-limb rehabilitation system to address current limitations. Chapter 2 presents a literature review covering medical conditions that require rehabilitation, such as musculoskeletal and neurological disorders. It examines traditional therapy methods and their shortcomings, and explores modern techniques like biomechanical analysis, motor-assisted therapy, and sensor-based tracking. The chapter also reviews potential actuators, sensors, and microcontrollers suitable for rehabilitation devices.

Chapter 3 describes the methodology used to design, develop, and test the prototype. It details the mechanical structure, choice of motors and actuators, and integration of the ESP32-based control system. Software architecture, electrical layout, communication protocols, calibration procedures, cost estimation, and testing strategies are also discussed. Chapter 4 presents the results from development and testing. It includes prototype images, validation of sensor accuracy, performance of actuators, and analysis of joint angle data captured by the accelerometer and gyroscope module (MPU6050) sensor. The chapter also covers the user interface, control mechanisms, and safety features implemented to prevent overextension or erratic motion. Chapter 5 concludes the report by summarizing key findings and contributions. It discusses the system’s potential for clinical and home-based rehabilitation, acknowledges current design limitations, and proposes future improvements such as adaptive feedback systems and broader sensor integration. References and technical appendices are provided at the end.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The literature review explores various rehabilitation techniques and technologies used to address musculoskeletal and neurological conditions that impair motor functions. These conditions, such as osteoarthritis, rheumatoid arthritis, stroke, and Parkinson's disease, often require comprehensive rehabilitation programs to restore mobility, strength, and daily functioning. The review examines both traditional and modern therapeutic approaches, highlighting the importance of biomechanics in designing effective rehabilitation strategies. Additionally, it discusses the integration of mechanical components and advanced sensors and microprocessor in rehabilitation systems, emphasizing their role in delivering precise, customizable therapies to support patient recovery and improve quality of life.

2.2 Musculoskeletal System Diseases Required Rehabilitation

Musculoskeletal conditions, such as osteoarthritis, rheumatoid arthritis, bone fractures, and sarcopenia, affect both the upper and lower extremities and often require comprehensive rehabilitation to restore function and improve quality of life. One of the most prevalent of these conditions is osteoarthritis (OA), a degenerative joint disease that affects millions of people worldwide. OA occurs when the cartilage that cushions the joints deteriorates over time, leading to pain, stiffness, and reduced mobility, especially in weight-bearing joints like the knees and hips (Mayo Clinic, 2021). As depicted in Figure 1, the stages of knee osteoarthritis illustrate the progressive degeneration of cartilage, which can lead to significant joint dysfunction. While OA primarily affects the lower extremities, it can also impact the hands and shoulders, causing similar limitations in function. Rehabilitation for OA typically involves physical therapy focused on enhancing joint mobility, strengthening muscles surrounding the joint, and improving flexibility. These interventions are vital for reducing pain and restoring the patient’s ability to carry out daily tasks effectively (Garrick, 2017).

Another musculoskeletal condition that often necessitates rehabilitation is rheumatoid arthritis (RA). RA is an autoimmune disorder characterized by inflammation in the joints, leading to painful swelling. Unlike OA, which is caused by wear and tear, RA results from the immune system mistakenly attacking the body's tissues. RA commonly affects smaller joints, such as those in the hands and feet, but it can also impact larger joints like the knees and elbows. Rehabilitation for RA focuses on managing pain and preserving joint function through a combination of physical and occupational therapy. These therapies aim to maintain the range of motion, strengthen muscles, and help patients adapt to the limitations imposed by joint pain and stiffness (Mayo Clinic, 2023).

Fractures, particularly in the extremities, also require extensive rehabilitation to fully restore mobility and strength. Fractured bones in the arms, legs, or hands typically require immobilization during the healing process, followed by physical therapy to regain movement and strength. The rehabilitation process is crucial for restoring flexibility, improving muscle strength, and ensuring proper bone healing. Depending on the severity of the fracture, recovery can take months, with therapy targeting the muscles surrounding the injury to help patients regain the range of motion necessary for performing daily activities (Cleveland Clinic, 2022a).

In older adults, sarcopenia, the age-related loss of muscle mass and strength is a growing concern. Sarcopenia can severely affect both mobility and dexterity, increasing the risk of falls and frailty. Rehabilitation for sarcopenia often involves resistance training exercises aimed at rebuilding muscle strength and improving balance. Such exercises are essential for slowing the progression of muscle loss and enabling individuals to continue performing everyday tasks. Nutritional interventions also play a key role in supporting muscle mass retention and overall health during rehabilitation (Cleveland Clinic, 2022b).

Globally, musculoskeletal conditions represent a significant health burden, affecting approximately 1.71 billion people in 2020 (Hanifa Bouziri et al., 2023). These conditions are the leading cause of disability worldwide, with low back pain being the most prevalent. Additionally, upper extremity disorders, such as carpal tunnel syndrome and repetitive strain injuries, are common, particularly in occupations involving repetitive motions. The prevalence of

these disorders varies widely across different populations, with studies indicating rates between 1.6% and 53% (Huisstede et al., 2006). Given the sheer number of individuals affected and the significant impact on quality of life, effective rehabilitation programs are essential for managing symptoms and improving long-term health outcomes.

2.3 Nervous System Diseases Required Rehabilitation

Nervous system diseases often result in significant impairments, particularly in the upper and lower extremities. These conditions disrupt the body's ability to control movement, sensation, and coordination. As a result, a structured and individualized rehabilitation approach becomes essential. Rehabilitation not only aids in restoring function but also enhances the quality of life and fosters independence. For example, a stroke, which occurs when blood flow to the brain is interrupted, frequently leads to hemiplegia, or paralysis on one side of the body. Consequently, both the arm and leg on the affected side are typically impaired, resulting in weakness, spasticity, and a loss of coordination (Bartels et al., 2016). These motor impairments make it difficult for individuals to perform daily tasks. Thus, rehabilitation for stroke patients emphasizes physical therapy to restore strength and movement, as well as occupational therapy to help them regain the ability to perform tasks such as dressing, eating, and grooming (Bartels et al., 2016).

Similarly, Multiple Sclerosis (MS) is a chronic autoimmune disorder that damages the protective covering of nerves. This leads to muscle weakness, fatigue, spasticity, and coordination difficulties, particularly in the extremities. As the disease progresses, patients often face increasing challenges with walking and fine motor tasks. Therefore, rehabilitation in MS is designed to manage these symptoms, helping patients maintain functional independence for as long as possible (Iezzoni, 2010). In addition, Peripheral Neuropathy, which involves damage to the peripheral nerves, leads to weakness, numbness, tingling, and pain, particularly in the extremities. This condition is often linked to diabetes, nerve trauma, or the side effects of certain medications. Consequently, rehabilitation focuses on improving strength, sensation, and mobility in the affected areas to enhance the patient's ability to perform everyday tasks (Armada-Da-Silva, 2014).

Furthermore, a Traumatic Brain Injury (TBI) can result in a wide range of motor impairments that affect both the arms and legs, depending on the area of the brain that is injured. As a result, TBI patients frequently experience difficulties with balance, coordination, and strength, which severely limit their mobility and ability to function in daily life. For this reason, rehabilitation for TBI patients is often intensive, involving physical therapy aimed at rebuilding muscle strength and restoring range of motion. Repetitive task training and functional electrical stimulation are commonly used to facilitate recovery (Vos and Diaz-Arrastia, 2015). Moreover, Parkinson's Disease is a progressive neurodegenerative disorder that primarily affects movement. It causes symptoms such as tremors, muscle stiffness, and balance problems. Over time, these motor impairments make walking, writing, and other fine motor tasks increasingly difficult. As a result, rehabilitation in Parkinson's Disease focuses on improving mobility, reducing tremors, and preventing falls to help patients maintain as much independence as possible (Ross and Singer, 2024).

In summary, neurological conditions require tailored rehabilitation programs focused on symptom management and functional recovery, playing a vital role in improving patients' quality of life. Neurological disorders are among the leading causes of death and disability globally, with their burden increasing, particularly due to aging populations. The 2016 Global Burden of Disease study identified neurological disorders as the second most common cause of death and the leading cause of disability-adjusted life years (DALYs). This burden is expected to rise, especially in regions like Asia, where aging populations and socioeconomic disparities contribute to the higher prevalence and mortality of these diseases (Kang et al., 2022). Targeted health policies are essential to address these challenges.

2.4 Conventional Rehabilitation Techniques

Rehabilitation systems today use various modern techniques to help individuals recover from conditions that affect movement, such as musculoskeletal injuries (like joint or muscle damage) and neurological issues (such as stroke or spinal cord injuries). These methods focus on restoring motor function and improving patients' ability to manage physical challenges. Below are some of the key rehabilitation techniques currently used. One of the most common techniques is

motor learning and repetition. This involves practicing movements repeatedly to strengthen the body's ability to perform them correctly. The idea is that by doing the same movements over and over, patients can relearn how to use their muscles effectively. In one study, patients participated in repetitive goal-directed active movements for 60 minutes a day over two weeks, leading to significant improvements in reaching (Cirstea et al., 2006). Research shows that repetitive training is highly effective in improving motor functions, especially in areas like the arms and legs.

Another widely used technique is task-oriented therapy, where patients practice activities that are similar to what they would do in their daily life. This method is important because it helps people regain the ability to do functional tasks, like walking, lifting objects, or using their hands. For instance, in rehabilitation for musculoskeletal conditions such as recovering from a knee replacement, patients might practice walking or climbing stairs to regain mobility. Similarly, stroke patients might focus on everyday actions like reaching for objects to improve coordination (Barnes and Good, 2014). Constraint-induced movement therapy (CIMT) is a specialized technique designed for patients who have one limb more affected than the other. In CIMT, the stronger or unaffected limb is restricted, forcing the patient to use the weaker or injured limb more often. This technique has been shown to help people with stroke or injuries like shoulder or arm damage rebuild strength and regain function in the affected limb (Barnes and Good, 2014). It is especially useful for those recovering from injuries where one side of the body becomes dominant, and the other needs encouragement to move and strengthen.

Robotic-assisted therapy is becoming increasingly popular in rehabilitation, particularly for patients with severe impairments. In this approach, robotic devices help patients perform repetitive movements that they might struggle to do on their own. For example, robots are often used to help people recovering from stroke or serious musculoskeletal injuries, like after a major surgery, by assisting them with exercises such as walking or moving their arms. These robots can adjust the level of support based on the patient's ability, allowing them to engage in rehabilitation activities more intensively and safely (Bertani et al., 2017). Electrical stimulation is another advanced rehabilitation technique that uses electrical impulses to stimulate muscles and help them move.

This is particularly useful in cases where the muscles have weakened or lost function due to injury or lack of use. Functional electrical stimulation (FES) can be applied to specific muscle groups, like in the hand or leg, to promote movement. In musculoskeletal rehabilitation, FES is often used after surgeries like knee or shoulder repairs, to help prevent muscle atrophy and speed up recovery. In neurological conditions, such as stroke or spinal cord injury, it can help patients regain some control over paralyzed or weakened muscles (Barnes and Good, 2014).

Mental practice and motor imagery is a technique that allows patients to mentally rehearse movements, even when they cannot physically perform them. This method activates the same brain pathways used during actual movement, which can help patients improve their motor skills. For example, patients who are recovering from surgery or a stroke can benefit from mentally practicing movements, helping to maintain their brain's connection to the muscles involved. Studies have shown that this technique can enhance recovery when combined with physical therapy (Page et al., 2005). Mirror therapy is a simple but effective technique used mainly in patients with paralysis or musculoskeletal injuries. In this therapy, a mirror is placed in such a way that the patient sees the reflection of their healthy limb, making it look like the affected limb is moving. This visual trick helps the brain reconnect with the impaired limb, which can speed up recovery. Mirror therapy has been particularly useful for stroke patients and those recovering from fractures or other injuries that affect one side of the body (Barnes and Good, 2014).

Overall, these modern rehabilitation techniques including repetitive practice, task-based exercises, robotic assistance, electrical stimulation, mental imagery, and mirror therapy offer a range of solutions to help patients recover from both musculoskeletal and neurological conditions. By combining different methods based on the patient's specific needs, rehabilitation programs are more effective in helping individuals regain strength, movement, and independence.

2.5 Current Rehabilitation Robot Technology

Rehabilitation robots have become essential tools in supporting patients with motor impairments by enhancing recovery, restoring movement, and improving quality of life. These robots are broadly categorized into assistive

robots, prosthetic and orthotic robots, and therapeutic or training robots. Each category serves distinct purposes within the rehabilitation process and offers specialized functionalities tailored to patient needs.

Assistive robots are designed to help individuals with physical disabilities carry out daily activities independently. Stationary assistive robots, such as Handy 1 and Neater Eater, are used primarily at fixed locations like a bedside or table. Their functionality includes helping users with feeding, grooming, or using communication devices, thereby improving their self-care capabilities. Mobile assistive robots, including intelligent wheelchairs and robots like Manus and Raptor, combine robotic arms with mobility platforms to help users interact with their environment. These robots assist with navigation, object manipulation, and mobility in dynamic settings, allowing users to function more independently in real-world environments (Li *et al.*, 2017).

Prosthetic and orthotic robots aim to replace or support lost limb function. Prosthetic robots, such as the Utah/MIT arm or Southampton hand, function by detecting muscle signals (EMG) or neural activity to mimic natural limb movements. Their main functionality is to restore voluntary motor control and allow users to perform complex tasks such as grasping, lifting, or pointing. Orthotic robots, particularly wearable exoskeletons like ReWalk and HAL, function as external frameworks that support and enhance limb movement. These devices assist with walking, posture, and balance by detecting user-initiated movements and providing real-time mechanical support to complete the action, enabling individuals with paralysis or lower limb weakness to walk again (Li *et al.*, 2017).

Therapeutic and training robots are primarily used in clinical rehabilitation to support motor recovery, particularly after stroke or spinal cord injury. Upper limb training robots like InMotion, ARMeo, and ReoGo function by guiding the patient's arm through repeated movements, either actively or passively, while providing visual and haptic feedback. This repetitive training helps in re-establishing neural pathways and improving coordination, strength, and range of motion. For lower limbs, systems like Lokomat, KineAssist, and Tibion provide body-weight-supported gait training. These robots function by moving the patient’s legs in natural walking patterns while allowing therapists to adjust support levels, resistance, and feedback. They improve walking ability,

balance, and endurance through safe and intensive rehabilitation sessions (Li *et al.*, 2017).

In summary, each type of rehabilitation robot brings a distinct functionality to the rehabilitation process—assistive robots enhance independence in daily tasks, prosthetic and orthotic robots restore motor capabilities, and therapeutic robots promote physical recovery through targeted training. Together, these technologies play a crucial role in advancing patient outcomes and enabling greater participation in daily life (Li *et al.*, 2017).

2.6 Biomechanics on Human Upper Extremities

The human upper limb is a highly complex structure composed of bones, joints, muscles, and other anatomical components that enable a wide range of motion and functionality. It consists of the shoulder girdle and the arm. The shoulder girdle includes the clavicle (collarbone) and scapula (shoulder blade), which provide structural support and connect the upper limb to the torso. The arm is composed of the humerus, radius, and ulna, along with the bones of the hand, including the carpal bones in the wrist, metacarpal bones in the palm, and the phalanges in the fingers. In total, there are 37 bones in the upper limb. The upper limb contains several major joints that allow for different movements. The shoulder joint is a ball-and-socket joint that connects the humerus to the scapula, allowing for multi-directional motion, including abduction, adduction, flexion, extension, and rotation. The elbow joint is a hinge joint made up of three smaller joints, providing flexion and extension of the arm, as well as supination and pronation of the forearm. The wrist and hand joints allow for intricate movements needed for grasping and manipulating objects (Jaworski et al., 2016).

A complex network of 43 muscles controls the upper limb's movements, dividing into groups that govern the shoulder girdle, arm, forearm, and hand. These muscles enable not only basic functions such as lifting and holding objects but also fine motor skills such as writing and precision tasks. The biomechanics of the upper limb allow for a broad range of motion, with the shoulder capable of flexion and extension (150°-180° and 40°-50°, respectively), abduction and adduction (180° and 30°-40°), and internal and external rotation (70°-95° and 40°-70°). The elbow permits flexion and extension (135°-140° and 0°) and forearm rotation through supination and pronation (85°-90° and 70°-

90°). The wrist has a range of motion in flexion/extension (73°/70°) and radial/ulnar deviation (27°/27°). Overall, the upper limb is a biomechanical marvel, functioning as an open kinematic chain that allows for significant rotational and linear movements, essential for everyday tasks like lifting, reaching, and interacting with the environment. Its sophisticated structure not only facilitates mobility and strength but also enables fine motor control, making it a crucial part of human anatomy (Jaworski et al., 2016).

2.7 Biomechanics on Human Lower Extremities

The human lower limb is a specialized structure designed to support body weight, enable movement, and maintain balance. It consists of the hip, knee, ankle, and foot, each of which plays a vital role in various movements such as walking, running, and standing. The primary bones include the femur (thigh bone), tibia and fibula (lower leg bones), and the tarsals, metatarsals, and phalanges in the foot. The femur, the longest bone in the body, connects the pelvis to the knee, while the tibia and fibula form the lower leg, with the foot's bones providing further support and mobility. The hip joint is a ball-and-socket joint, allowing for a wide range of movements, including flexion, extension, abduction, adduction, and internal and external rotation. The knee joint functions mainly as a hinge, enabling flexion and extension, with some rotational movement when bent. The ankle joint allows for dorsiflexion (raising the foot upward) and plantarflexion (pointing the foot downward), along with inversion and eversion, which help the foot adjust to different surfaces (Monk et al., 2016).

The lower limb has a significant range of motion. The hip can flex up to 120°-130° and extend by 10°-30°. It can abduct (move away from the body) by 40°-50° and adduct (move toward the body) by 20°-30°. Internal rotation ranges between 30°-40°, while external rotation is 40°-60°. The knee can flex to around 135°-150°, with full extension to 0°, though slight hyperextension is possible in some individuals. The ankle moves within 10°-20° of dorsiflexion and 25°-30° of plantarflexion, providing the flexibility needed for walking and running (CDC, 2023). Biomechanically, the lower limb plays a crucial role in absorbing and distributing forces during movement. The hip joint experiences forces of up to 300% of body weight during walking and up to 500% during

jogging. The knee joint bears significant loads, especially during weight-bearing activities. Misalignments, such as genu varum (bow-legged) or genu valgum (knock-kneed), can shift the load distribution, leading to stress on specific areas and potentially causing joint degeneration over time (Monk et al., 2016).

The ankle and foot are essential for stability and forward propulsion. The ankle has a large weight-bearing surface that distributes forces over a broad area, minimizing stress on the joint. The subtalar joint, located beneath the ankle, converts rotational forces from the tibia into foot movements like inversion and eversion, helping the foot adapt to uneven surfaces. The transverse tarsal joint provides both flexibility and rigidity, essential for stabilizing the foot during gait. The plantar fascia, a thick tissue band on the sole of the foot, tightens during the push-off phase of walking, stiffening the foot to create a rigid lever for propulsion. In summary, the lower limb's bones, joints, and muscles work together to provide stability and mobility, allowing for efficient locomotion and balance. Its biomechanics are crucial for absorbing forces, facilitating movement, and protecting the body during physical activities (Monk et al., 2016).

2.8 Human Anthropometry

According to Casadei and Kiel (2022), anthropometry involves the non-invasive, quantitative measurement of the human body, including parameters such as height, weight, body circumferences, skinfold thickness, and body mass index (BMI). These measurements are essential for evaluating the nutritional and health status of both children and adults. In pediatric populations, they help assess growth patterns, nutritional adequacy, and overall health, while in adults, they are used to evaluate body composition, diagnose conditions like obesity, and predict future health risks. Anthropometric data is also crucial for monitoring the well-being of pregnant women and athletes, supporting interventions that enhance health outcomes. Beyond healthcare, such data is widely used by manufacturers to design products that accommodate the physical diversity of global populations (Dawal et al., 2015). In the context of rehabilitation system design, anthropometric data plays a vital role in ensuring ergonomic compatibility with users of varying body types. For this project, user-specific anthropometric data is based on average body measurements of the

Malaysian population. Anthropometric data for various body segments, detailing mass distribution, center of mass locations, and segment dimensions across the sagittal, transverse, and longitudinal planes for both males and females are used in the theoretical calculations.

2.9 Rotational Actuator Design

Rotational actuators convert electrical energy into rotational motion and are widely used in robotics, automation, and consumer devices. The three common types are stepper motors, servo motors, and DC motors, each offering different levels of precision, control, and performance. Stepper motors operate in precise steps without feedback, making them suitable for low-cost, accurate positioning tasks like 3D printing and CNC machines. Servo motors use feedback for continuous, precise motion control and are ideal for dynamic applications such as robotics and automation systems. DC motors, while less precise, are simple, reliable, and commonly used in appliances and automotive systems. Table 2.1 compares these motor types based on key features including operation, precision, control complexity, torque and speed, maintenance, and typical applications.

Table 2.1: Comparison of Rotational Actuator Types.

Feature	Stepper Motors	Servo Motors	DC Motors
Operation	Discrete steps via sequentially energized electromagnets (open loop) (Dietrich, 2022)	Continuous motion with feedback control (closed loop) (Dietrich, 2022)	Continuous rotation from magnetic and electrical interaction (Helen, 2019)
Precision	High (fixed step increments, no feedback) (Dietrich, 2022)	Very high (real-time position correction via feedback) (Dietrich, 2022)	Low to moderate (no precise position control) (Helen, 2019)

Table 2.1 (Continued)

Control Complexity	Simple (open-loop control, no feedback) (Helen, 2019)	Complex (requires encoders, feedback loops) (Dietrich, 2022)	Simple (basic power supply or PWM) (Helen, 2019)
Torque and Speed	Moderate torque, lower speed; resonance issues possible (Helen, 2019)	High torque and speed; adaptable under load (Dietrich, 2022)	Moderate torque and speed; efficiency varies by type (Helen, 2019)
Types	Permanent magnet, variable reluctance, hybrid (Dietrich, 2022)	AC servo, brushed and brushless DC servo (Dietrich, 2022)	Brushed and brushless DC (Helen, 2019)
Maintenance	Low	Moderate to low depending on type (Dietrich, 2022)	Brushed require maintenance; brushless low maintenance (Helen, 2019)
Typical Applications	3D printing, CNC, robotics (Helen, 2019)	Robotics, automation, conveyors (Dietrich, 2022)	Household appliances, toys, automotive (Helen, 2019)

2.10 Linear Actuator Design

Linear actuators are essential components in systems requiring controlled, straight-line motion. In rehabilitation and medical applications, three primary types of actuators are commonly used: telescopic lifting columns, screw-type actuators, and linear electric actuators—each offering unique advantages based on performance, size, and complexity. Telescopic lifting columns are compact, self-contained units that integrate the motor, drive, and electronics. They

provide a high extension ratio and stable operation with minimal noise. These actuators are ideal for space-constrained applications such as adjustable hospital beds, therapy tables, and ergonomic office systems. Their low maintenance and aesthetic appeal make them suitable for both medical and industrial use (Casillo, 2023).

Screw-type actuators include lead screws, ball screws, and planetary roller screws. Lead screws are the most economical but wear out faster due to friction between threads, making them suitable for light-duty applications. Ball screws offer better efficiency, reduced friction, and greater precision, making them ideal for applications requiring accurate positioning. Planetary roller screws provide the highest thrust, durability, and operational life, but are also the most expensive and complex (GlobalSpec, n.d.). Linear electric actuators convert electrical energy directly into linear motion without mechanical linkages. These actuators offer high precision, fast response, and low maintenance due to fewer moving parts. They are widely used in robotics, medical imaging systems, and automation, although they require more sophisticated control systems and are typically more costly (Boldea and Nasar, 2024). To aid in selection, Table 2.2 summarizes the key characteristics of each actuator type, highlighting their precision, load capacity, durability, noise level, cost, and ideal applications.

Table 2.2: Comparison of Linear Actuator Types.

Type	Precision	Load	Durability	Noise	Cost	Best Use
Telescopic Column	Moderate	High	High	Low	Medium	Compact lifting systems
Lead Screw	Low	Low	Low	Low	Low	Simple, low-load applications
Ball Screw	High	High	High	Medium	Medium	Accurate rehab/medical devices
Planetary Roller Screw	Very High	Very High	Very High	High	High	Heavy-duty, high-precision tasks
Linear Electric Actuator	Very High	High	High	Low	High	Robotics, automation, imaging

2.11 Sensor Design

2.11.1 Inertial Measurement Unit

Inertial Measurement Units (IMUs) are key sensors used to track motion, orientation, and position in fields like robotics, VR, and autonomous vehicles. IMUs typically combine a 3-axis accelerometer, 3-axis gyroscope, and sometimes a 3-axis magnetometer to provide real-time data on acceleration, angular velocity, and orientation (Gunasekaran, 2019). Modern IMUs use MEMS (Micro-Electro-Mechanical Systems) technology for compactness and energy efficiency. MEMS accelerometers detect linear acceleration via changes in capacitance caused by proof mass displacement, while MEMS gyroscopes detect angular motion using the Coriolis effect (Joseph, 2022). One popular IMU, the MPU6050, integrates a 3-axis accelerometer and gyroscope with a Digital Motion Processor (DMP) for on-chip sensor fusion. It supports I2C and SPI communication, with configurable measurement ranges and internal buffering, though it may lack precision for high-accuracy tasks (Tripathi & Kumar, 2023). The MPU9250 expands upon the MPU6050 by adding a magnetometer, enabling full 9-axis tracking suitable for advanced navigation. It also features a DMP and flexible sampling rates, making it versatile for low-power or real-time systems (Aniket Fasate, 2023). Table 2.3 summarizes the key features and differences of these IMU modules, including sensor types, ranges, interfaces, and ideal applications.

Table 2.3: Comparison of IMU Sensor Modules.

Feature	**MPU6050**	**MPU9250**
Sensor Components	3-axis accelerometer, 3-axis gyroscope	3-axis accelerometer, gyroscope, magnetometer
Motion Processor	Digital Motion Processor (DMP)	Digital Motion Processor (DMP)
Measurement Ranges	±2g to ±16g (accel), ±250°/s to ±2000°/s (gyro)	Same as MPU6050
Magnetometer	No	Yes (3-axis)
ADC Resolution	16-bit	16-bit

Table 2.3 (Continued)

Communication Interface	I2C (400 kHz), SPI (1 MHz)	I2C, SPI
Applications	Motion tracking, gesture recognition	3D orientation, navigation systems

2.11.2 Electromyography Sensor

Electromyography (EMG) sensors are essential tools for capturing electrical signals generated by muscle activation, offering a non-invasive means to monitor muscle activity. They are widely used in electrodiagnostic medicine, robotics, rehabilitation, motion analysis, and prosthetics due to their ability to reveal insights into motor control and biomechanics (Kamavuako, 2022). Technological advancements have made EMG sensors more compact and efficient, enabling integration with machine learning to improve prosthetic and assistive device control. These integrations facilitate smoother, more intuitive human-machine interactions.

In rehabilitation, EMG sensors help decode user intent, enhance prosthetic functionality, and assess muscle coordination for developing biomechanical models. Clinically, they aid in evaluating conditions like dysphagia, where EMG can differentiate swallowing types and monitor hydration in elderly patients (Ye-Lin et al., 2022). In sports and physical therapy, EMG is used to monitor muscle fatigue, prevent injuries, and customize rehabilitation programs. As wearable technologies evolve, EMG sensors are increasingly embedded in wearables for real-time muscle activity monitoring, benefiting users with motor impairments or neurodegenerative conditions.

There are two main types of EMG sensors: surface EMG, which uses skin-placed electrodes for general muscle activity monitoring in non-invasive applications; and intramuscular EMG, which involves needle electrodes inserted into muscles for more precise diagnostics. The choice depends on the required data precision and acceptable invasiveness (Zhao, 2015).

2.12 Microprocessor

Microcontrollers play a central role in controlling sensors, actuators, and managing wireless communication in rehabilitation and IoT systems. The

ESP32, developed by Espressif Systems, is a powerful SoC microcontroller that integrates Wi-Fi, Bluetooth 4.2, and dual-core processing up to 240 MHz. It features 36 GPIO pins, 16 PWM channels, 4MB flash memory, and supports multiple protocols including SPI, I2C, UART, and CAN. Hardware-accelerated encryption and a low-power design make it ideal for real-time and secure medical applications (Babiuch et al., 2019). In this project, the ESP32 is chosen for its real-time performance, wireless connectivity, motor control capabilities, and compatibility with various sensors and actuators.

In contrast, the Arduino Uno WiFi Rev2 uses an ATmega4809 microcontroller clocked at 16 MHz with 48 KB flash memory and 6 KB SRAM. It includes the NINA-W102 module for Wi-Fi and Bluetooth, and an onboard LSM6DS3TR motion sensor. It offers 14 digital I/O pins, 6 analog inputs, and 5 PWM outputs, and supports secure communication through the ATECC608 chip. Though user-friendly and well-supported by the Arduino ecosystem, its limited processing power and fewer I/O pins make it less suitable for complex or high-performance applications (Arduino, 2024). Table 2.4 compares the ESP32 and Arduino Uno Rev2, highlighting their specifications and suitability. Overall, the ESP32 offers greater performance, flexibility, and scalability for this rehabilitation system (Tan & Vendel, 2021).

Table 2.4: Comparison of ESP32 and Arduino Uno WiFi Rev2.

Feature	**ESP32**	**Arduino Uno Rev2**
Processor	Dual-core, 240 MHz	ATmega4809, 16 MHz
Memory	4MB flash	48 KB flash, 6 KB SRAM
Wireless	Wi-Fi 802.11 b/g/n, Bluetooth 4.2	Wi-Fi 802.11 b/g/n, Bluetooth 4.2 (NINA-W102)
GPIO / PWM	36 GPIO pins, 16 PWM channels	14 digital I/O, 5 PWM outputs
Communication Protocols	SPI, I2C, UART, CAN	SPI, I2C, UART
Onboard Sensor	–	LSM6DS3TR (3-axis IMU)

Table 2.4 (Continued)

Security Features	AES, SHA-2, RSA, ECC, RNG	ATECC608 chip
Applications	IoT, medical devices, robotics	Prototyping, basic IoT systems

2.13 Summary

The literature review addresses the rehabilitation needs for musculoskeletal and nervous system diseases, focusing on conditions like osteoarthritis, rheumatoid arthritis, stroke, and Parkinson's disease. Rehabilitation techniques such as physical therapy, task-oriented exercises, robotic-assisted therapy, and electrical stimulation are essential for restoring mobility, strength, and functionality in patients. These methods are particularly important for improving the quality of life for individuals with impaired motor functions in the upper and lower extremities. Additionally, the biomechanics of human limbs play a crucial role in guiding rehabilitation approaches, ensuring that therapies are tailored to support movement and recovery.

Key mechanical components such as linear and rotational actuators play a crucial role in modern rehabilitation systems. Linear actuators, including telescopic lifting columns and screw type mechanisms, deliver the precise and stable motion required for medical applications. Rotational actuators such as stepper motors, servo motors, and DC motors provide controlled movements essential for therapeutic exercises. In addition, advanced sensors like Inertial Measurement Units (IMUs) and Electromyography (EMG) sensors monitor motion and muscle activity, offering real-time feedback that enhances the effectiveness of therapy. Microcontrollers are also incorporated to manage system control and ensure coordinated functionality. Collectively, these technologies enable rehabilitation systems to deliver accurate, adaptive therapies that support motor recovery and promote user independence.

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction

This chapter outlines the methodology and workplan for the complete development of the stationary upper and lower extremities rehabilitation system. The components and materials used are clearly described in the following sections. In addition, this chapter covers the methods applied in the design, preparation, fabrication, and assembly of all mechanical parts, including the prototype's structural frame, actuators, circuit system, and user interface. It also details the data validation and functionality testing, providing a comprehensive understanding and visualization of the entire process undertaken to build the prototype.

3.2 Prototype Development Process

Figure 3.1 illustrates the overall methodology flow for the development of the stationary upper and lower extremities rehabilitation system. The process was clearly structured into three parallel pathways: mechanical design, circuit design, and user interface design. It began with a literature review to establish foundational knowledge related to the research area, specifically the stationary rehabilitation system focused on the human upper and lower extremities. For the mechanical design, the development started by creating the design using SOLIDWORKS software. Materials were then prepared based on the research findings. The purchased materials were converted into the respective prototype parts. Equipment such as Computer Numerical Control (CNC) machines, drilling machines, and cutting machines was used to fabricate the structural frame and larger components. 3D printing was used to produce smaller components, such as the circuit box for the accelerometer and gyroscope module (MPU6050) sensor. At the same time, the electrical design process, which included circuit design and component procurement, was carried out. Once all components were prepared, the circuit assembly was performed. In parallel, the user interface (UI) development was conducted without the need for laboratory equipment. This involved designing the UI using FlutterFlow and integrating it

with the database, in this case Firebase, for data storage and retrieval. Once all three parts were completed, the full assembly of the prototype was conducted. The completed prototype then underwent validation and functionality testing to confirm its safety and performance. Finally, data analysis was performed on the collected data to effectively present the obtained results.

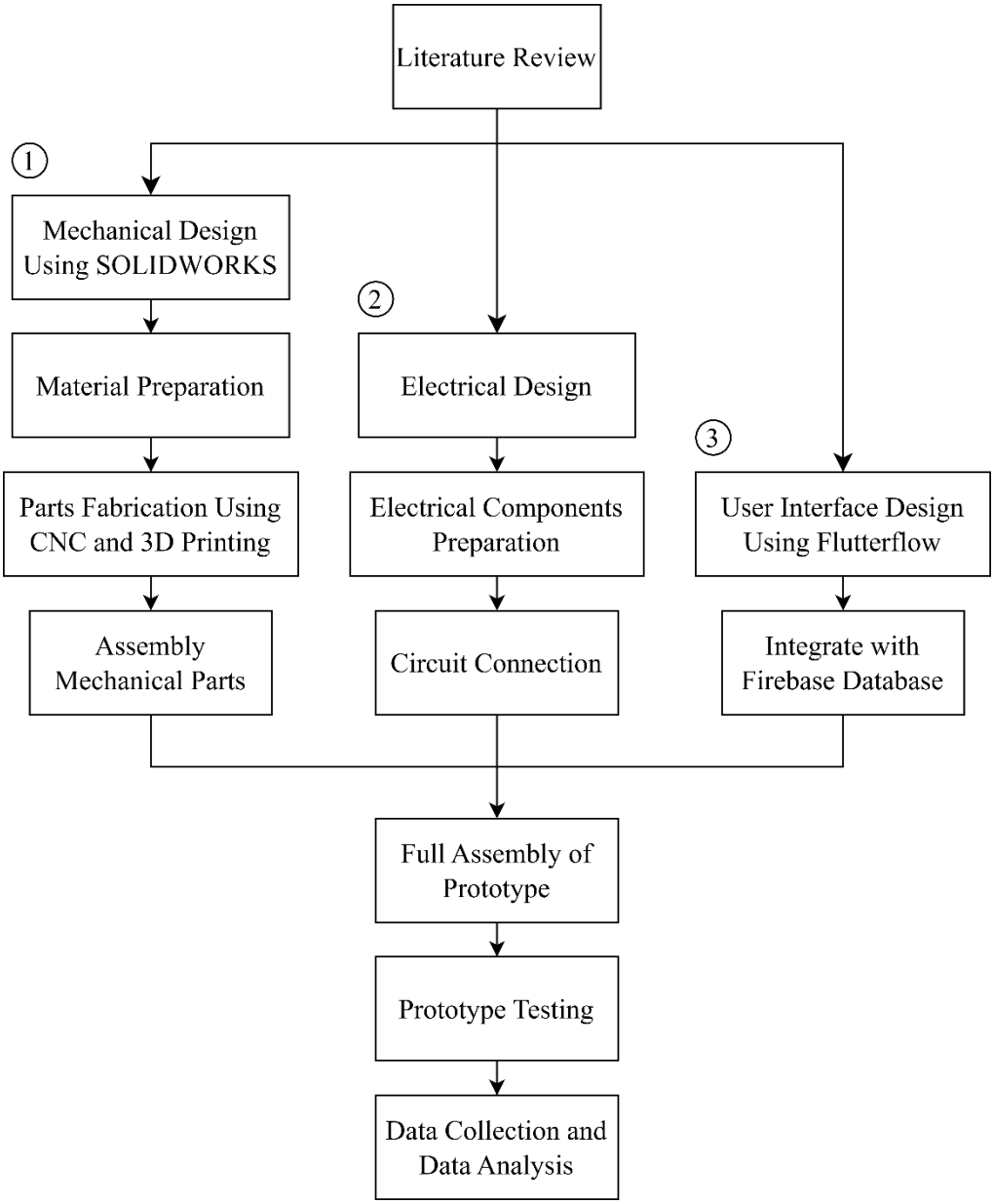


Figure 3.1: Prototype Development Process.

Figure 3.2 and Figure 3.3 further describe the project activities carried out over 2 semesters. All milestones that were set were successfully achieved within the established timeline.

Gantt Chart Part-1														
No.	Project Activities	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
M1	Project Initiation and Problem Statement														
M2	Literature Review and Background Research														
M3	Conceptual Design														
M4	Report Writing and Presentation														

Figure 3.2: Gantt Chart Final Year Project (FYP) Part 1.

Gantt Chart Part-2															
No.	Project Activities	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
M1	Prototype Parts Fabrication, Circuit Design, and User Interface Development														
M2	Prototype Full Assembly														
M3	Data Collection and Analysis														
M4	FYP Poster														
M5	FYP2 Report Writing and Oral Presentation														

Figure 3.3: Gantt Chart Final Year Project (FYP) Part 2.

3.3 System Architecture

The system architecture depicted in Figure 3.4 demonstrates the integration of mechanical components, electronic circuits, and the user interface. It is divided into two main segments: the main system, shown in Figure 3.4(a), which consists of the physical prototype, and the sensor system, illustrated in Figure 3.4(b), which includes sensors that interact with the user interface. Each segment is independently controlled by its own ESP32 microcontroller. In the main system, the microcontroller is powered by a portable 5V power supply, while a separate 3-pin plug power supply drives the actuators, including the stepper motor and linear actuator. To enhance operational safety, an emergency stop switch is placed between the microcontroller and the 3-pin power supply. The microcontroller also functions as a signal processor, receiving input via an

IR receiver, decoding signals from the remote, and executing the corresponding commands based on the received hexadecimal codes to control the mechanical components. In the sensor system, the microcontroller which also powered by portable 5V power supply collects data from the accelerometer and gyroscope module (MPU6050) sensor, which provides both accelerometer and gyroscope readings. This data is then uploaded to a Firebase database, where it can be accessed and visualized through a user interface developed using the FlutterFlow platform. Overall, this architecture ensures reliable system functionality, user-friendly interaction, and real-time data monitoring for efficient rehabilitation performance.

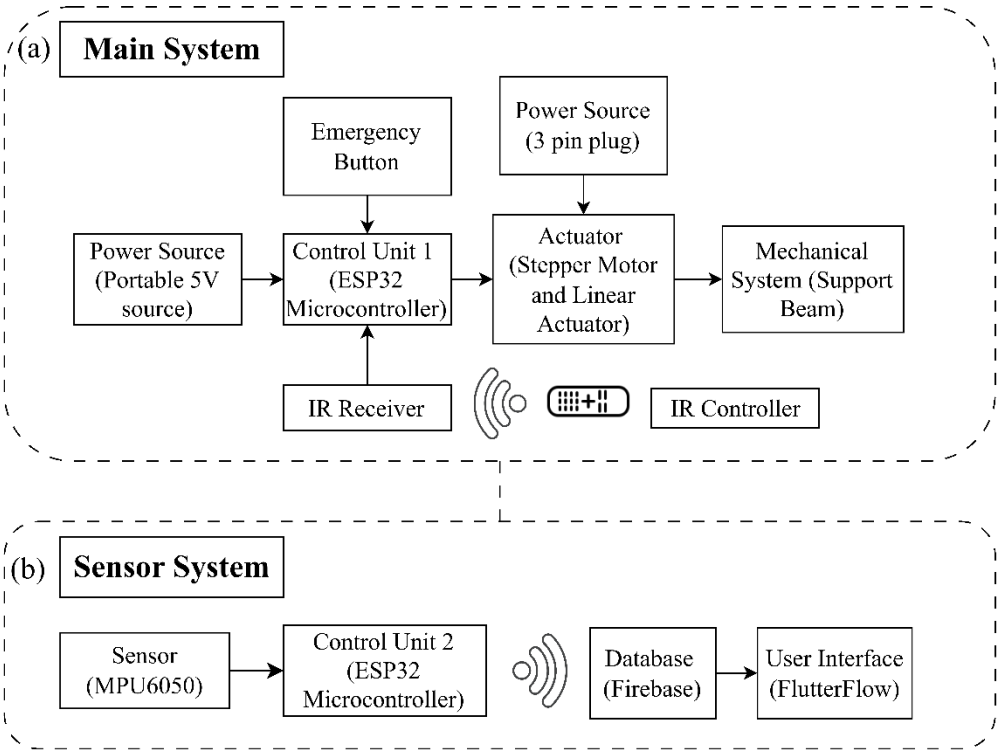


Figure 3.4: Prototype System Architecture for (a) Main System (b) Sensor System.

3.4 Mechanical Design

The overall mechanical system, as shown in Figure 3.5, consists of two major components: the prototype frame and the actuator system, which provides both linear and rotational motion to specific parts. The main frame is constructed

using aluminium profiles, providing the primary structural support and forming the system's base. All aluminium profiles are securely connected using zinc L-brackets to ensure strong and rigid joints. Inside the frame, a plywood circuit platform serves as the mounting base for electronic components and wiring, safely enclosed and supported by the aluminium structure. The frame is equipped with four caster wheels, allowing smooth mobility and easy repositioning of the unit. Hollow mild steel components are used for the limb support section to ensure sufficient strength and rigidity. Additionally, custom 3D-printed parts using Polylactic Acid (PLA) such as connectors, the linear actuator base connector, and the movable Velcro-secure plate are integrated at key connection points to meet specific functional and geometric requirements. A detailed engineering of the overall prototype frame is provided in Appendix A for better understanding.

For actuation, the system features a stepper motor with planetary gear reduction, mounted on a mild steel platform. This motor serves as the primary actuator and delivers controlled rehabilitation motion. A linear actuator supports the motor platform and provides vertical adjustment to accommodate different user positions, allowing the system height to be set from sitting knee level to sitting elbow level. The limb support is directly attached to the motor shaft and features a hand grip along with a movable Velcro-secure plate to hold the limb securely during operation. The motor driver and control electronics are mounted on an upper plate above the actuators, keeping the system compact, organized, and easy to maintain. Overall, the design offers a well-balanced combination of modularity, strength, and adjustability. This makes it highly suitable for rehabilitation applications that require precise and programmable linear and rotational movements, while ensuring seamless integration between mechanical and electrical components.

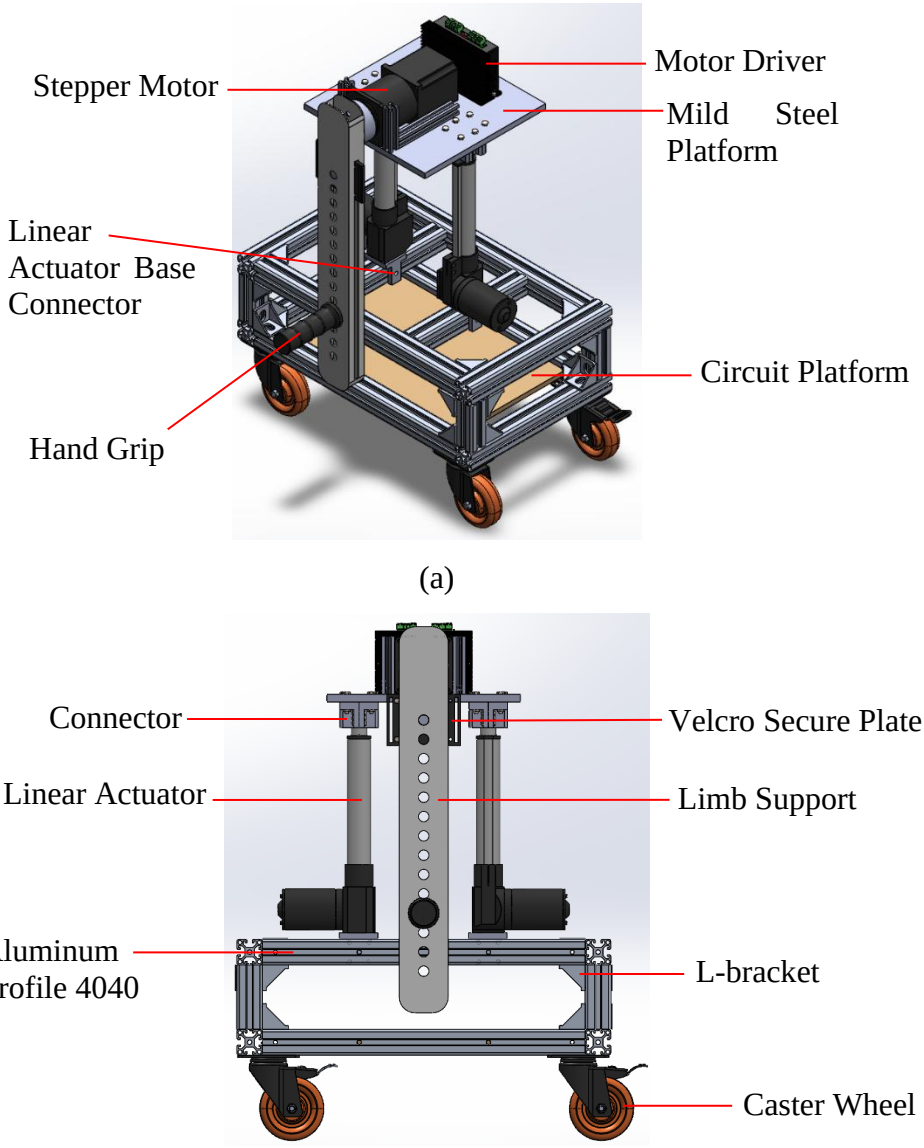


Figure 3.5: Conceptual Design of Stationary Upper and Lower Extremities Rehabilitation System (a) Isometric View (b) Front View.

Figure 3.6 shows a custom-designed cover, intended to be 3D printed using PLA, for housing the accelerometer and gyroscope module (MPU6050) sensor. The purpose of this development is to ensure the sensor is securely mounted in a position that provides measurements comparable to commercial sensor devices. It is specifically designed for placement on the forearm and calf to capture acceleration and gyroscope data accurately. The hollow interior allows space for the necessary wiring, while the small circular hole is designed to hold a magnet, enabling the top cover to close securely and remain in place during use. Additionally, the linear hollow bar at the base is designed to

accommodate a Velcro strap, making it easy to attach the sensor housing directly to a user's limb.

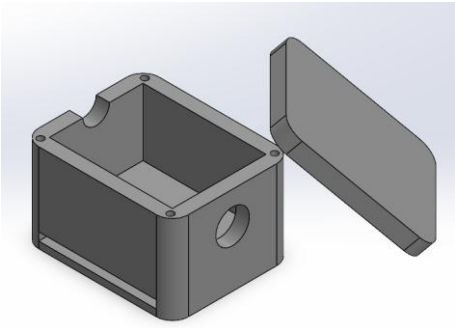


Figure 3.6: Accelerometer and Gyroscope Module (MPU6050) Sensor Cover.

3.4.1 Limb Supports

The main movable components of the prototype are the limb supports, as illustrated in Figure 3.7. Each limb support is designed with a length of 600 mm, determined based on Malaysian anthropometric data collected from 1,007 individuals, comprising 516 males and 491 females. According to the data, the average sitting knee height is 513.56 mm for males and 470.36 mm for females, while the average forearm-hand length is 460.67 mm for males and 421.53 mm for females (Mohamad et al., 2010). Therefore, a 600 mm limb support length was selected to accommodate the majority of users. To enhance adaptability, a movable Velcro-secured plate is placed within the track of the limb support, allowing adjustment according to the user's limb length and providing secure support during specific rehabilitation phases. Additionally, interchangeable components have been designed for the hand grip and calf support to accommodate different rehabilitation modes. When the hand grip is attached, the user can hold it during elbow flexion and extension exercises. Alternatively, when the calf support cushion is in place, the user can rest their leg on it during lower limb rehabilitation sessions. In addition, a 3D-printed coupler, as shown in Figure 3.7, is designed at the end of the limb support to securely attach it to the motor shaft, ensuring efficient motion transmission during rehabilitation exercises.

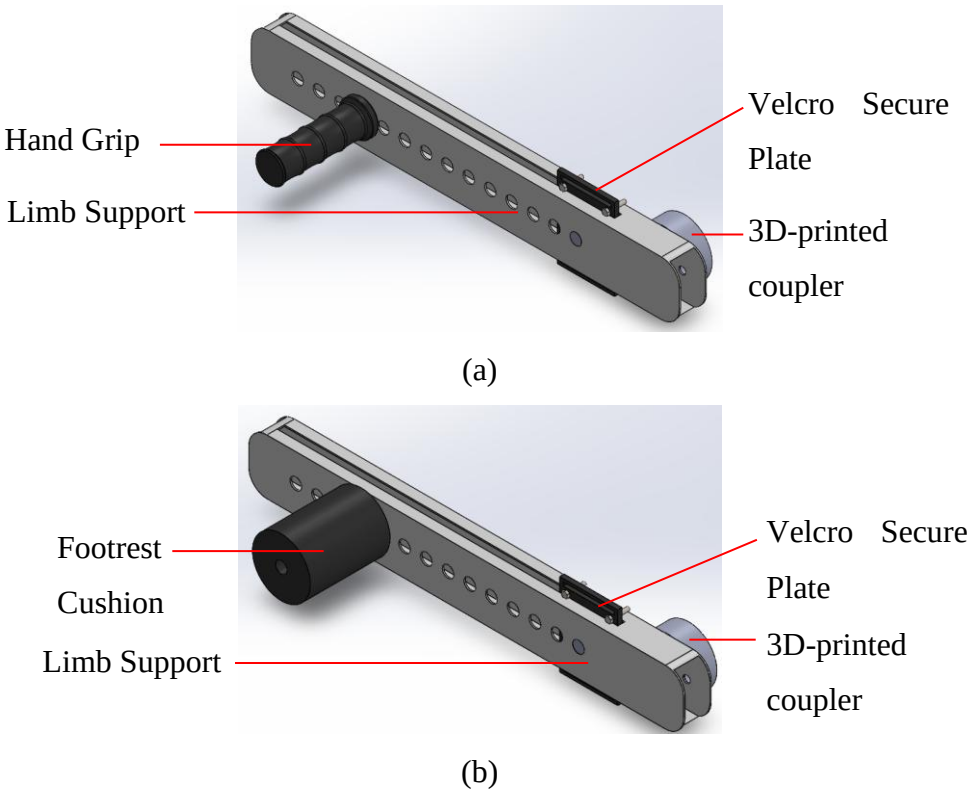


Figure 3.7: Limb Support (a) Forearm Support (b) Shank Support.

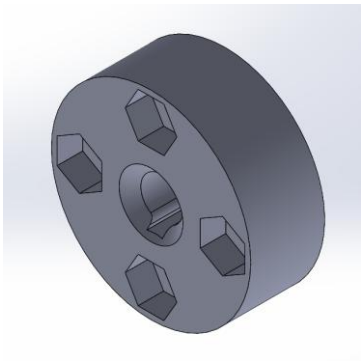


Figure 3.8: 3D-Printed Coupler.

In this design, mild steel hollow bar was selected as the material for the limb support due to its lightweight nature and high durability, allowing it to be driven effectively by the stepper motor while resisting deformation over long-term use. The customized coupler for the limb support is planned to be 3D printed to ensure a precise fit with the shaft and to prevent any dislodgement during operation. The handgrip is adapted from a bicycle handle grip, the Velcro securing plate is 3D printed, and the footrest cushion is made from foam, which is wrapped around the bar.

3.4.2 Stepper Motor

The stepper motor was intentionally selected as the rotational actuator for this rehabilitation system due to its ability to provide precise and reliable control of angular motion. This makes it well-suited for enabling controlled movement of the upper and lower extremities during therapy, as illustrated in Figure 3.9. Determining the required torque is essential for selecting a stepper motor with appropriate specifications. To achieve this, the minimum torque needed to lift the limb and meet performance requirements must be accurately calculated, beginning with the fundamental torque equation presented in Equation 3.1.

$$\tau = r \cdot F \tag{3.1}$$

where

τ = Torque, Nm

r = Radius from axis of rotation to point of application of the force, m

F = Applied force, N

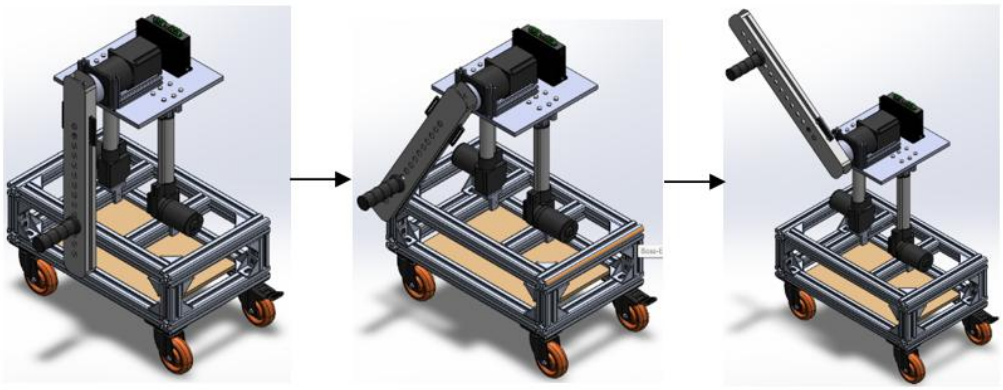


Figure 3.9: Rotational Motion of Limb Support.

In this context, torque is required to overcome the gravitational force acting on both the limb support structure and the human limb itself, allowing the limb to move through its intended range of motion during flexion and extension of the elbow and knee joints. Equation 3.2 expands upon the initial calculation by incorporating the combined torque contributions from the support structure and the limb. The weight and length of the limb segments are estimated

using anthropometric data derived from a study involving 1,006 Malaysian individuals, ensuring realistic and population-specific parameters.

$$\tau_{\min} = \tau_{\text{BS min}} + \tau_{\text{S min}} \tag{3.2}$$

where

τ_{min} = Minimum torque required by stepper motor.

$\tau_{BS\,min}$ = Minimum torque required by stepper motor to lift body segment.

$\tau_{S\,min}$ = Minimum torque required by stepper motor to lift support structure.

An essential factor in torque calculation is the angle of rotation, as it directly affects the moment arm and the resulting gravitational load. For this analysis, the target joint ranges are 0–90 degrees for the elbow and 0–90 degrees for the knee, which represent optimal therapeutic motion goals. The initial step in determining the required torque involves evaluating the torque necessary to lift each body segment. Figure 3.10 illustrates the reaction forces acting on the human forearm and elbow joint, as well as the shank extension and forearm flexion. This analysis leads to Equation 3.3, which defines the net torque required to achieve controlled movement of the specific joint. The derivation of Equation 3.3 is provided in the Appendix D.

$$\tau_{\text{BS min}} = \text{M} = (\text{m}_\ell \text{g})\text{CoM}_\ell \tag{3.3}$$

where

$\tau_{BS\,min}$ = Minimum torque required by stepper motor to lift body segment, Nm

M = Moment, Nm

m_ℓ = Limb mass, kg

g = Gravity force, m/s^2

CoM_ℓ = Center of mass of limb, m

The mass of the limbs is estimated as a percentage of total body weight. According to anthropometric data by (de Leva 1996), the forearm and hand constitute 1.62% and 0.61% of total body weight in males, and 1.38% and

0.56% in females, respectively. Similarly, the shank and foot represent 4.33% and 1.37% of body weight in males, and 4.81% and 1.29% in females. These values are summarized in Table 3.1.

Table 3.1: Mass Percentage of Limb for Male and Female.

	Percentage of Mass (%)			
	Forearm	**Hand**	**Shank**	**Foot**
Male	1.62	0.61	4.33	1.37
Female	1.38	0.56	4.81	1.29

These percentages were applied to the average body weights of Malaysian males (66.64 kg) and females (60.40 kg), obtained from a research study, to calculate the corresponding limb masses (Mohamad et al., 2010). The resulting weights, expressed as forces due to gravity, are presented in Table 3.2. For males, the combined weight of the forearm and hand was calculated as 14.59 N, while the shank and foot weighed 37.26 N. For females, the corresponding values were 11.50 N for the forearm and hand, and 36.14 N for the shank and foot.

Table 3.2: Calculated Limb Weight for Male and Female.

	Body Weight, m_ℓ (kg)	**Forearm and Hand Weight, W_{F+H} (N)**	**Shank and Foot Weight, W_{S+F} (N)**
Male	66.64	14.58	37.26
Female	60.40	11.49	36.14

The center of mass (CoM) of each limb segment is estimated as a percentage of the total limb length. According to anthropometric data from de Leva (1996), the CoM of the forearm is located at 79.00 % of its length in males and 74.74 % in females. Similarly, the CoM of the shank is positioned at 44.59 % of its length in males and 44.16 % in females. These percentages are measured from the proximal joint of each limb segment, specifically from the

elbow joint for the forearm and from the knee joint for the shank. These values are summarized in Table 3.3.

Table 3.3: Length Percentage of Center of Mass (CoM) for Male and Female.

	Percentage of Length (%)	
	Forearm	**Shank**
Male	79.00 %	44.59 %
Female	74.74 %	44.16 %

These percentages were applied to the forearm and shank lengths of Malaysian males (460.67 mm for the forearm and 513.56 mm for the shank) and females (421.53 mm for the forearm and 470.36 mm for the shank), obtained from a research study (Mohamad et al., 2010), to calculate the corresponding center of mass (CoM) positions. The resulting CoM values are presented in Table 3.2. For males, the CoM of the forearm was calculated as 363.93 mm and the CoM of the shank as 229.00 mm. For females, the corresponding values were 315.05 mm for the forearm and 207.71 mm for the shank.

Table 3.4: Calculated Limb Center of Mass (CoM) for Male and Female.

	Forearm Length (m)	**Shank Length (m)**	**CoM of Forearm, CoM_F (m)**	**CoM of Shank, CoM_S (m)**
Male	0.46	0.51	0.36	0.23
Female	0.42	0.47	0.31	0.21

Using the calculated mass and center of mass of each limb, the torque required to lift the forearm and shank to a 90-degree angle, as illustrated in Figures 3.10(a) and 3.10(b) respectively, was determined and is presented in Table 3.5.

Table 3.5: Minimum Torque Required to Lift Human Limb.

	Minimum Torque (Nm)	
	Forearm	**Shank**
Male	5.25	8.57
Female	3.68	7.59

Determining the torque required to lift the limb support structure is a crucial step in calculating the minimum torque needed by the stepper motor to raise the combined weight of the limb and its support. The support structures are illustrated in Figure 3.12, where Figure 3.12(a) shows the forearm support structure and Figure 3.12(b) shows the shank support structure. A red pivot point is marked on each support structure to indicate the location where the moment is generated when the rotational mechanism is activated. The torque required to lift the support structure is described by Equation 3.4. The center of mass is estimated to be located at three-quarters of the distance from the pivot point, due to the placement of the handgrip and footrest cushion near the end of the support structures, which increases the load at the distal end. The weight of each support structure is estimated to be 3 kg. The calculated minimum torque required to lift the limb support structures is 8.82 Nm for the forearm support structure and 13.24 Nm for the shank support structure, as summarized in Table 3.6.

$$\tau_{S\,min} = M = \ m_S\, gr \tag{3.4}$$

where

$\tau_{S\,min}$ = Minimum torque required by stepper motor to lift limb support structure, Nm

M = Moment, Nm

m_s = Limb support structure mass, kg

g = Gravity force, m/s^2

r = Center of gravity of limb, m

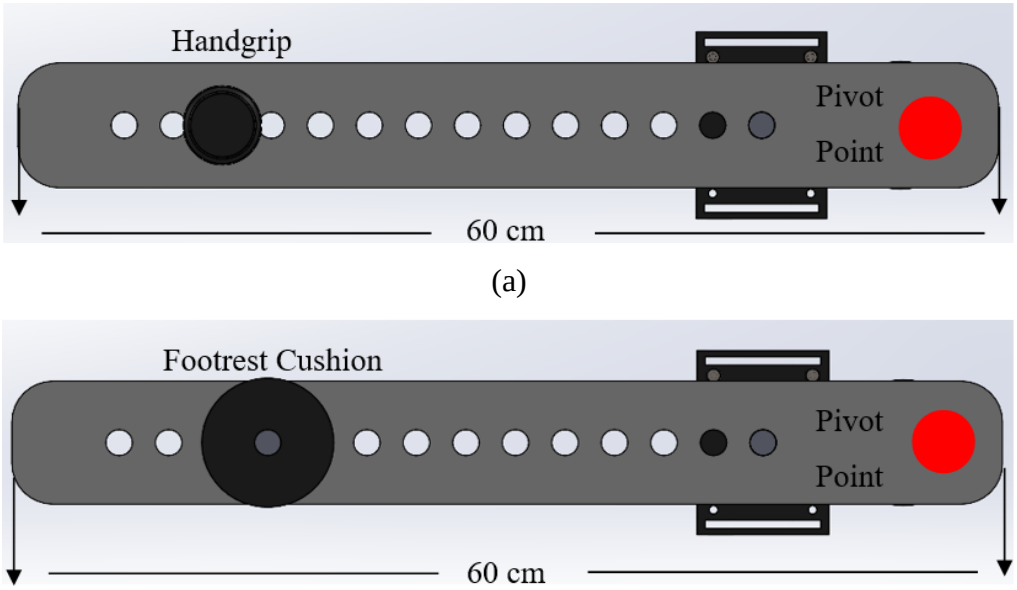


Figure 3.10: Limb Support Structure (a) Forearm Support Structure (b) Shank Support Structure.

Table 3.6: Minimum Torque Required to Lift Limb Support Structure.

	Minimum Torque (Nm)
Forearm Support Structure	8.83
Shank Support Structure	13.24

After calculating the torque required to lift both the limbs and their respective support structures, the total minimum torque that must be generated by the stepper motor to ensure proper system functionality is summarized in Table 3.7. Based on these calculations, it is determined that the stepper motor should be capable of generating a minimum torque of 21.78 Nm. To meet this requirement, a stepper motor with a rated torque of 4.5 Nm is selected, paired with a planetary reduction gear featuring a 50:1 reduction ratio, resulting in an output torque of approximately 220 Nm. The safety factor of the stepper motor in this system is calculated by taking the ratio of the available torque (220 Nm) to the required minimum torque (21.78 Nm). The theoretical safety factor is approximately 10.1, indicating that the selected motor and gear combination provides a substantial margin of safety for the intended application.

Table 3.7: Total Required Minimum Torque for Stepper Motor.

	Minimum Torque (Nm)	
	Forearm and Forearm Support Structure	**Shank and Shank Support Structure**
Male	14.13	21.78
Female	12.45	20.75

3.4.3 Linear Actuator

The linear actuator is one of the moving components in the system and is intended to adjust the height of the stepper motor to accommodate rehabilitation activities at varying user heights, as shown in Figure 3.12. Since individuals have different sitting heights as well as different sitting elbow and knee joint positions, a height-adjustable mechanism, as illustrated in the figure, facilitates easier transitions between rehabilitation modes for example, switching from knee joint to elbow joint exercises while in a seated position. This adaptability also allows the system to be used by multiple users with varying sitting heights. According to research, the average sitting heights for Malaysian individuals are 846.68 mm for males and 792.86 mm for females, with specific joint sitting heights summarized in Table 3.8. The linear actuator is therefore designed with a range of motion that covers the required height adjustments. In addition to height variability, the actuator must also be capable of supporting the weight of the stepper motor platform, which is a critical factor in selecting an appropriate actuator.

Table 3.8: Anthropometric Data for Sitting Position.

Anthropometric Data	**Mean (mm)**	
	Male	**Female**
Sitting Height	846.68	792.86
Sitting Shoulder Height	554.36	515.84
Sitting Knee Height	513.56	470.36

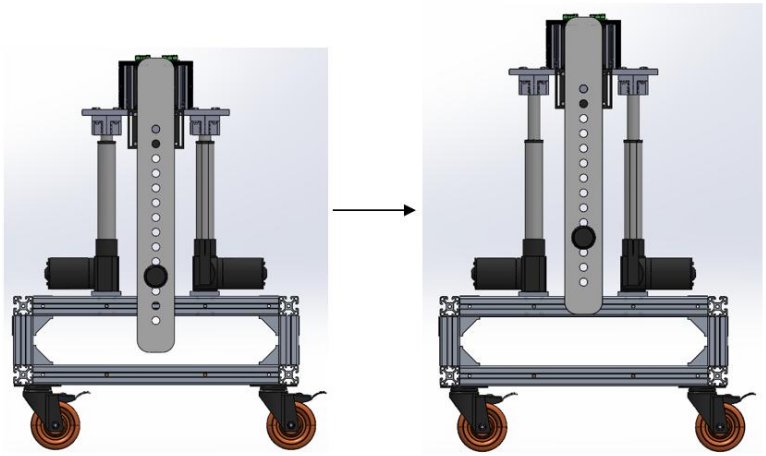


Figure 3.11: Linear Motion of Stepper Motor Platform.

The linear actuators are designed to support the stepper motor assembly, which comprises approximately 5 kg for the stepper motor and planetary reduction gear, around 3 kg for the limb support structure, and an additional 1 kg for the mild steel plate and motor driver, contributing to a total load of approximately 9 kg. The selected linear actuators are each capable of sustaining a load of 600 N, providing a combined capacity of 1200 N when two actuators are used. The total system load is approximately 88.3 N. By dividing the total actuator capacity of 1200 N by the total load of 88.3 N, the safety factor is approximately 13.6. This indicated that the chosen linear actuators offer strong and reliable support, ensuring safe and stable operation well within the system's mechanical limits.

3.5 Electrical and Electronic Design

Figure 3.12 shows the schematic diagram of the complete main circuit design of the developed prototype. The system can be divided into several key components, including the power supply system with an emergency stop, the stepper motor system, the linear actuator system, the IR controller system, and the LCD screen display system. The entire system is controlled by a microcontroller, the ESP32 that powered by a constant 5V source.

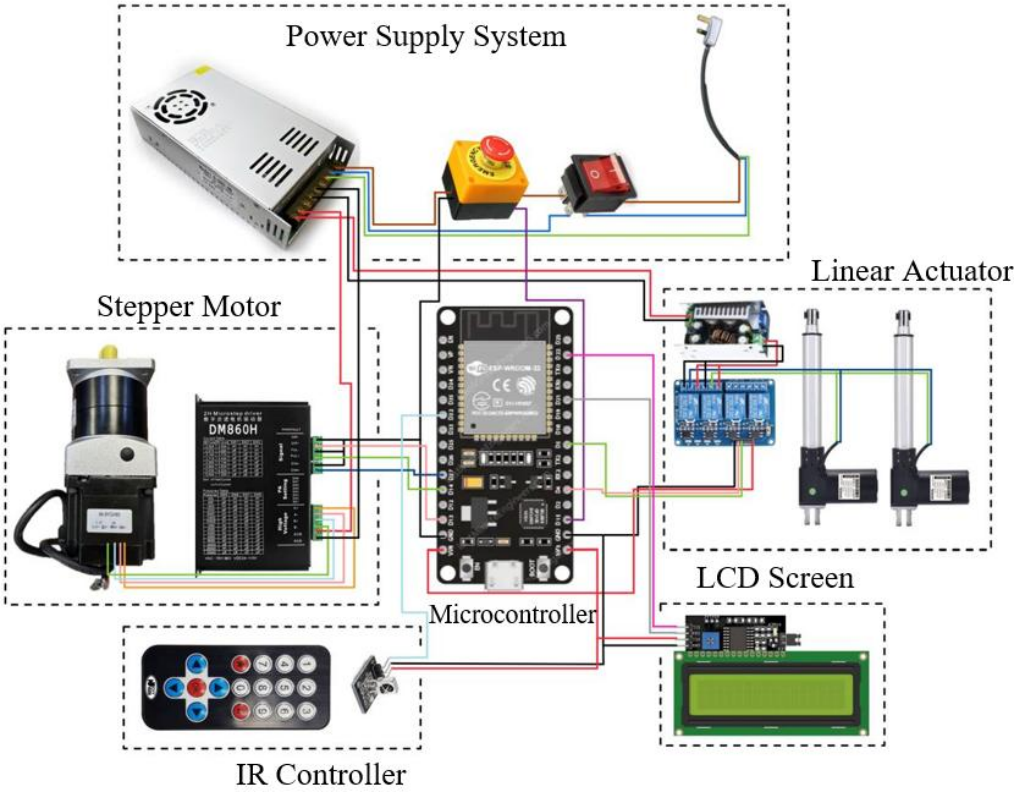


Figure 3.12: Schematic Diagram of Main System Circuit Design.

The ESP32 was chosen as the microcontroller because of its advanced features, particularly its Wi-Fi connectivity, which allows communication with a cloud-based data storage. Its excellent performance in executing tasks is another reason for its selection. Integrated with the programming code, the ESP32 serves as the main control unit, managing communication, task execution, and feedback mechanisms for the overall prototype. By decoding the IR signals sent by the IR remote control, the ESP32 executes tasks corresponding to specific predefined hex codes in the program, such as rotating the stepper motor to a specific angle, moving the linear actuators up and down, and displaying information on the LCD screen.

Figure 3.13 shows the sensor system, which operates independently from the main system. The accelerometer and gyroscope module (MPU6050) sensor serves as the primary sensor for analysing the performance of rehabilitation activities. To ensure smoother task execution and more reliable data transmission to cloud storage, allowing data retrieval without lag, the

sensor system is kept separate from the main control system. It primarily collects dynamic kinematic data, including linear acceleration and angular velocity.

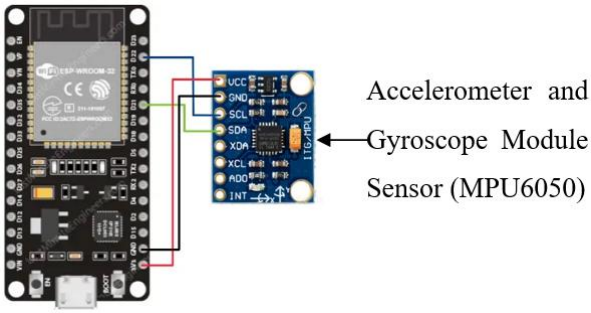


Figure 3.13: Schematic Diagram of Sensor System Circuit Design.

3.5.1 Power Supply System with Emergency Stop

Figure 3.14 shows the circuit connection of the power supply system, which includes a 3-pin plug wire connected to an rocker switch, integrated with an emergency stop, and finally linked to the AC to 48V DC power supply unit. The ON/OFF button controls the power supply to the entire system, allowing the user to easily start or stop the device. It enhances safety by preventing accidental activation and helps conserve energy when the system is not in use. Additionally, it can serve as a simple way to reset the system if needed. The emergency stop is a critical safety feature designed to immediately halt system operation in case of an emergency or malfunction. It is also connected to the ESP32 to reset the program after the emergency stop is pressed, ensuring a full system reset. This mechanism allows users to quickly disconnect power or stop motion to prevent accidents, damage, or injury, ensuring the system can be safely and rapidly deactivated in hazardous situations. The AC to 48V DC power supply unit converts standard AC mains voltage into a stable 48V DC output. It delivers the necessary power for components that operate at 48V DC, such as the stepper motor, ensuring reliable and safe energy delivery for consistent system performance.

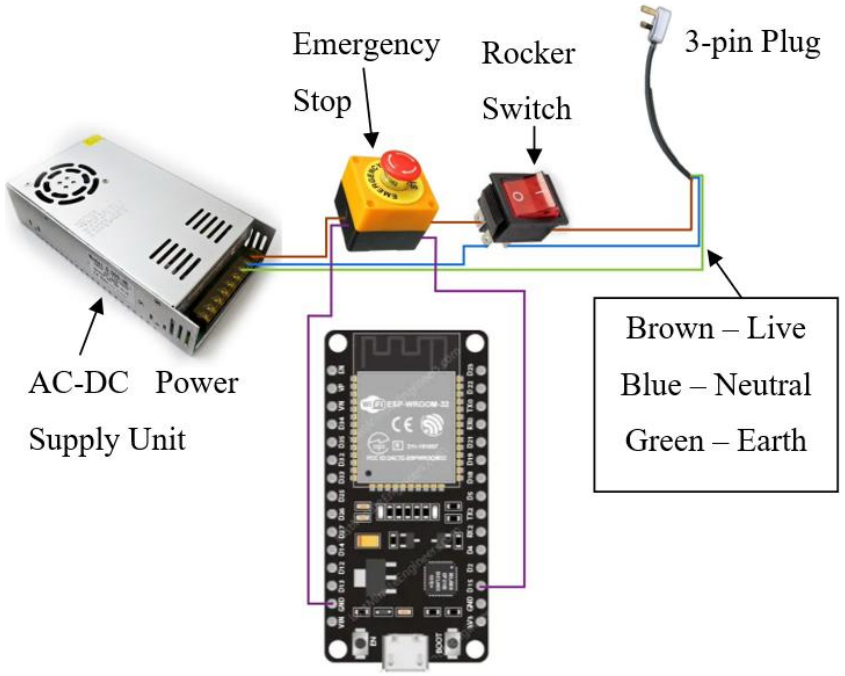


Figure 3.14: Circuit Connection of Power Supply System with Emergency Stop.

3.5.2 Stepper Motor System

Figure 3.15 shows the connection of the selected 4.5 Nm stepper motor with a planetary reduction gear, powered by the 48V DC power supply and connected to the motor driver. The motor driver is interfaced with the ESP32 microcontroller. Three main pins are connected between the motor driver and the ESP32: DIR (Direction), PUL (Pulse), and ENA (Enable). The DIR pin on the motor driver is the direction control pin that determines the rotation direction of the stepper motor. When the ESP32 sends a high signal to this pin, the motor rotates in one direction (e.g., clockwise); when a low signal is sent, it rotates in the opposite direction (e.g., counterclockwise). This allows precise directional control through the code. The PUL pin is the pulse input that receives step signals from the ESP32. Each time the ESP32 sends a pulse to this pin, the motor advances by one step. A continuous series of pulses results in continuous rotation, with the frequency of pulses controlling the speed and the total number of pulses determining the rotation angle or distance. The ENA pin is the enable input that controls whether the motor driver is active or disabled. When the ESP32 sends an active signal (logic high or low, depending on the driver), the motor is enabled and ready to operate. When the pin receives a deactivation signal, the motor driver output is disabled, stopping the motor or putting it into

a high-impedance state for safety and power saving. The ESP32 microcontroller acts as the main controller, managing the speed, acceleration, direction, and rotation angle of the stepper motor based on program logic and external inputs.

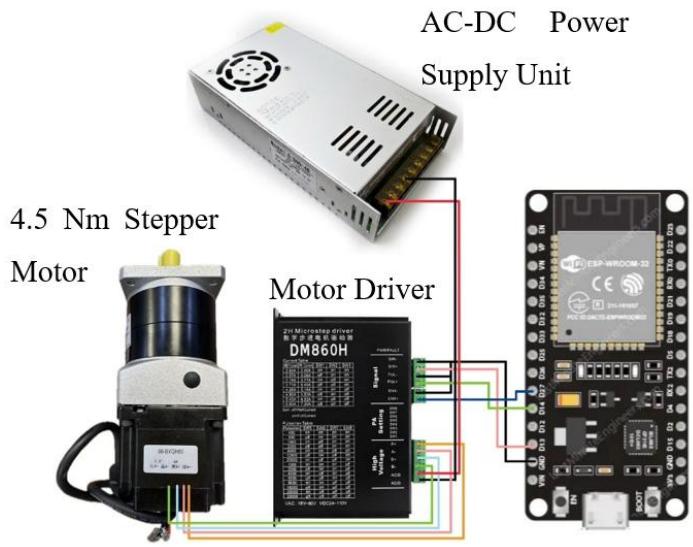


Figure 3.15: Circuit Connection of Stepper Motor System.

3.5.3 Linear Actuator System

Figure 3.16 shows the circuit connection of the selected linear actuator system. A step-down transformer is used to reduce the 48V supply to 12V, which matches the operating voltage of the linear actuator. A relay board is employed to receive low-voltage control signals from the ESP32 microcontroller and switch the high-voltage power to operate the linear actuator. When the ESP32 receives a signal from the IR controller, it sends a low-voltage control signal to the relay board, which in turn activates the linear actuator to move up or down as required.

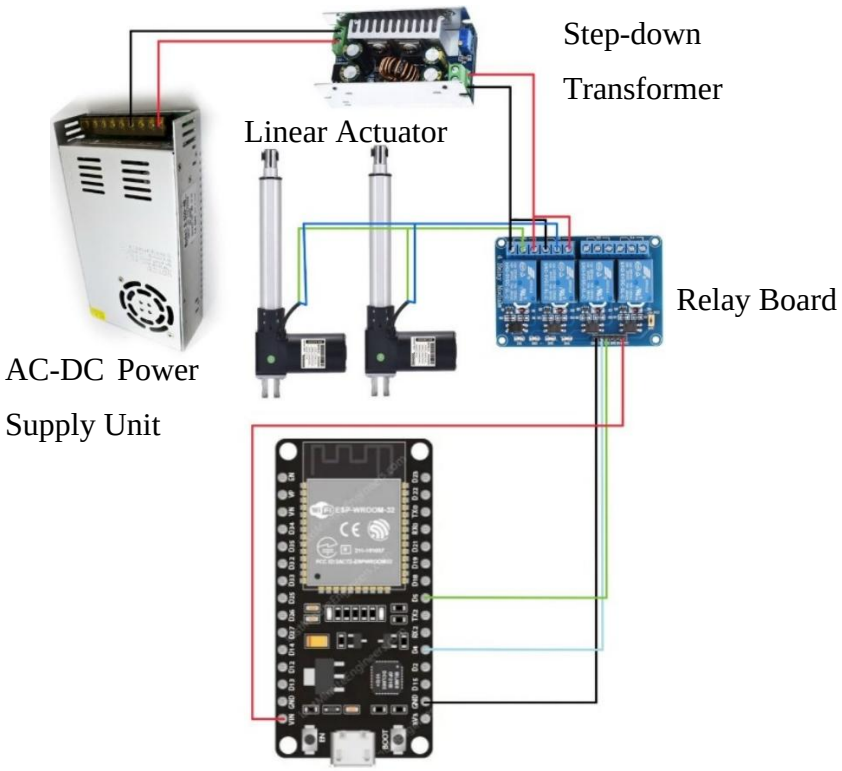


Figure 3.16: Circuit Connection of Linear Actuator System.

3.5.4 Gyroscope and Accelerometer Module System

Figure 3.17 illustrates the sensor system setup, which includes an ESP32 microcontroller and an MPU6050 module, which functions as both a gyroscope and accelerometer. This sensor system is primarily used to collect dynamic kinematic data during motion, specifically flexion and extension of the upper and lower extremities. The MPU6050 was mounted on the system cover and on the user's limb, as shown in Figure 3.18, to collect data such as range of motion, acceleration, and angular velocity.

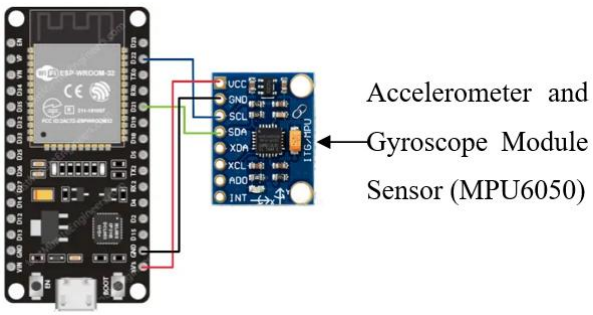


Figure 3.17: Circuit Connection of Gyroscope and Accelerometer Module (MPU6050) Sensor.

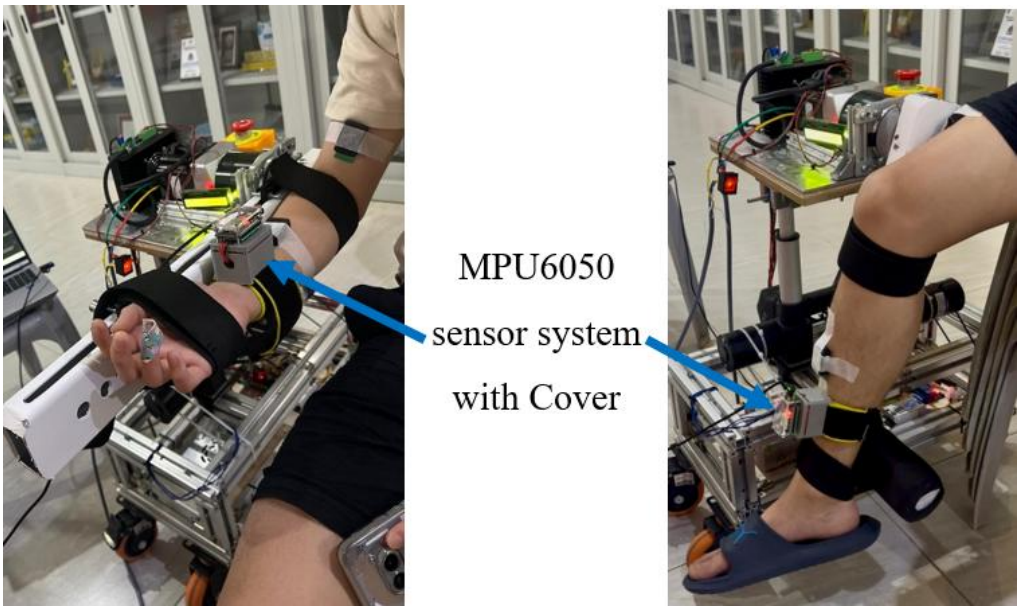


Figure 3.18: MPU6050 Sensor Allocation on Forearm and Shank during Rehabilitation Activity.

3.5.5 IR Remote Control

Figure 3.19 shows the circuit connection between the infrared (IR) receiver and the ESP32. When a button on the IR remote control is pressed, it emits an IR signal that is received by the IR receiver. The receiver then decodes this signal into a hexadecimal code, which the ESP32 interprets to execute specific tasks such as moving the stepper motor within a defined range of motion, raising or lowering the linear actuator, or displaying messages on the LCD screen.

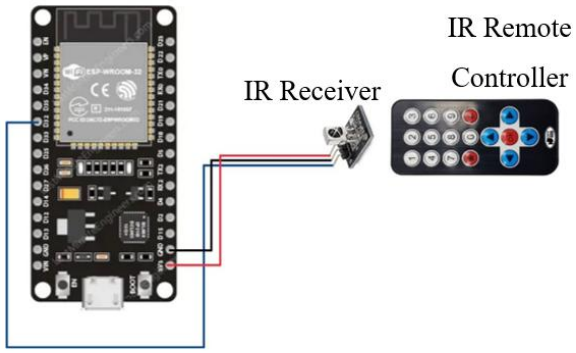


Figure 3.19: Circuit Connection of IR Receiver.

3.5.6 LCD Screen

Figure 3.20 displays the circuit connection between the liquid-crystal display (LCD) screen and the ESP32 microcontroller. Upon receiving an IR signal, the

ESP32 decodes the command and performs the corresponding action. Simultaneously, it updates the LCD display to show the executed step or current system status. This visual feedback enhances user awareness and interaction, providing informative prompts or updates throughout the system's operation, thereby improving usability and ensuring smoother control.

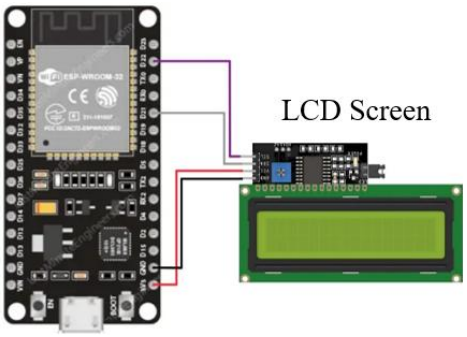


Figure 3.20: Circuit Connection of LCD Screen.

3.6 Internet of Things System Design

The architecture of the Internet of Things (IoT) system is illustrated in Figure 3.21, comprising four key components that work together to collect, store, and retrieve motion sensor data. At the user end, the custom-designed application developed using FlutterFlow serves as the main interface, allowing users to log in and access their data. FlutterFlow was selected for its advantages, including ease of use and minimal coding requirements. Upon login, the user's unique ID is submitted to Firebase through the app. This ID plays a critical role in data storage, as the ESP32 microcontroller retrieves the current logged-in user ID and stores the measured data under the corresponding account. This approach ensures data privacy for each patient and simplifies user-specific data tracking. In addition to data storage, the app allows users to retrieve their motion records and related information at any time.

The Firebase Firestore database functions as the central cloud-based storage system. It securely stores user IDs submitted by the FlutterFlow app and acts as a communication bridge between the app and hardware. When the ESP32 control unit operates, it first retrieves the user ID from Firebase to ensure that any collected sensor data is tagged and stored under the correct user's account. Besides managing user identification, Firebase also stores the actual motion data,

making it available for real-time updates and future retrieval. The ESP32 microcontroller-based control unit is a critical hardware component of the system. After obtaining the user ID from Firebase, the ESP32 continuously collects motion data from the connected accelerometer and gyroscope module (MPU6050) sensor. This sensor provides accelerometer and gyroscope readings, capturing detailed information on movement and orientation. Once captured, the ESP32 uploads the data to Firebase, associating it with the respective user ID to maintain data isolation and accuracy.

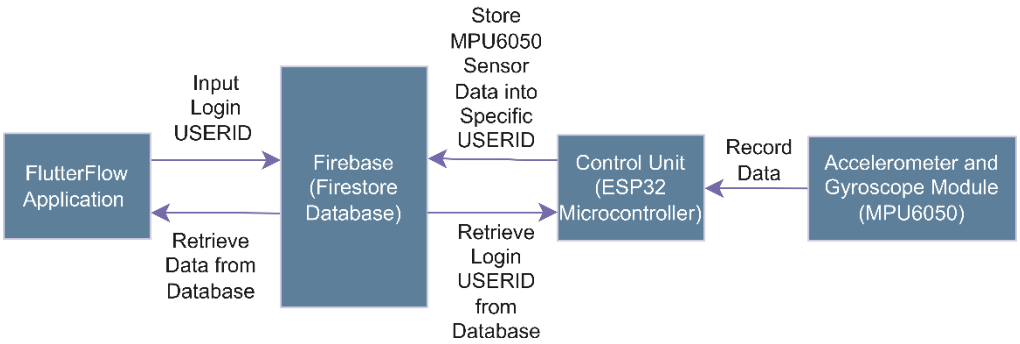


Figure 3.21: Internet of Things (IoT) System Architecture.

The MPU6050 plays a vital role in capturing raw motion data. It integrates an accelerometer and gyroscope to measure linear acceleration and angular velocity, providing a comprehensive picture of the user's movements. This data is transmitted to the ESP32, which processes and uploads it to the cloud. To illustrate the data collection and upload process implemented within the Arduino environment, a flowchart is included as Figure 3.22. This chart visually outlines the sequence of operations carried out by the ESP32, starting with retrieving the user ID from Firebase, collecting motion data from the MPU6050, and uploading the processed data back to Firebase. It highlights essential steps such as establishing a Wi-Fi connection, initializing the sensor, reading accelerometer and gyroscope data, formatting the readings, and storing them under the correct path in Firebase. The full Arduino code is shown in Appendix B. This visual representation clarifies the interaction between hardware and cloud services, demonstrating how the system ensures data integrity and real-time synchronization.

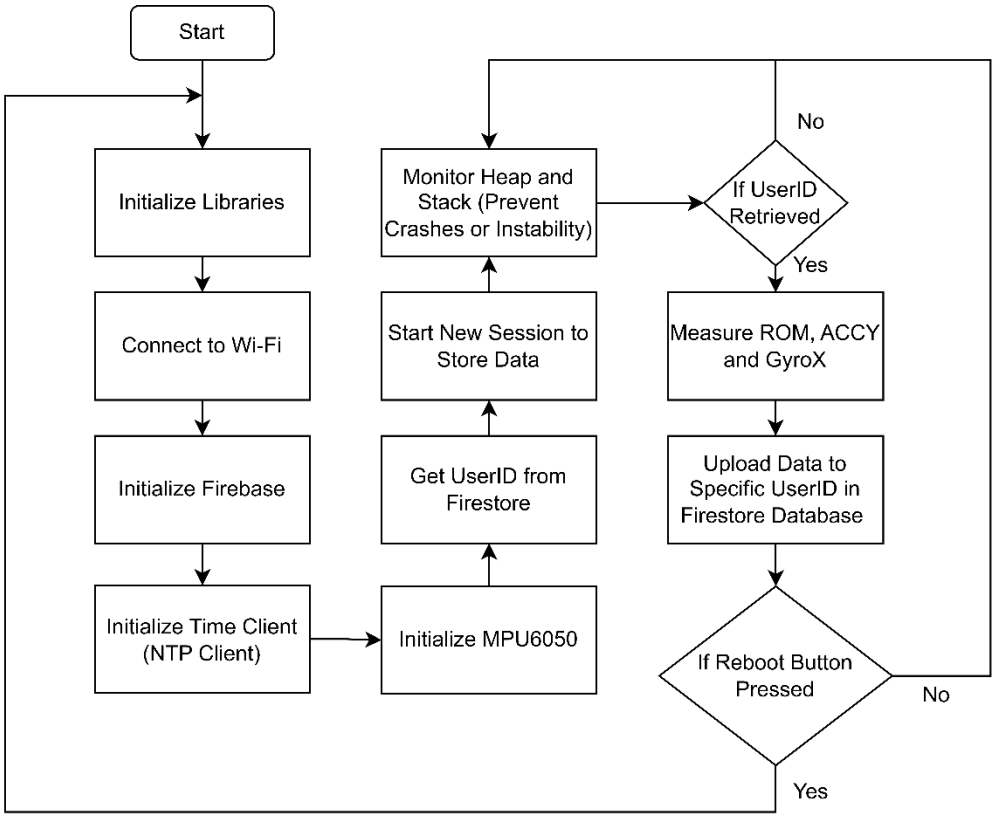


Figure 3.22: Process Flowchart for Sensor Data Collection and Firebase Storage.

In summary, this IoT system architecture provides a seamless and efficient workflow from the sensor hardware to the user interface. By leveraging Firebase as the central database, the system ensures secure data storage and easy access. The integration of FlutterFlow, Firebase, ESP32, and the MPU6050 sensor forms a cohesive framework that supports real-time data collection, cloud storage, and user interaction, making it highly suitable for motion tracking and similar applications.

3.7 System Workflow

The following section outlines the workflow of the completed system, which integrates the Arduino IDE code as shown in Appendix C, a physical Arduino board, and the mechanical components. As shown in Figure 3.23, the process begins with system initialization. This includes establishing serial communication for debugging, configuring the IR receiver for remote inputs, and setting up the stepper motor with predefined speed, acceleration, and control pins. The linear actuator is also prepared by setting its control pins as outputs, while the LCD display is initialized to provide real-time user feedback. Additionally, the emergency stop button is configured to ensure user safety, and the system records the initial position of the stepper motor as a reference. After initialization, the system continuously monitors the emergency stop button. If the button is not pressed, the system proceeds; however, if it is activated, all operations are immediately halted. The system remains in a paused state until the emergency stop is released, after which it resumes normal operation.

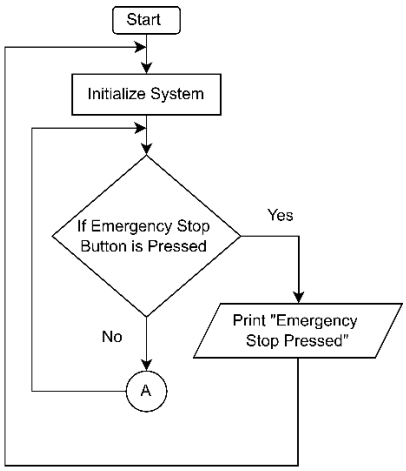


Figure 3.23: Initial Step.

After confirming that the emergency stop is not pressed, the system proceeds to receive an infrared (IR) signal from the remote control. If no signal is detected, the system continuously reads data from the IR receiver port until a valid signal is received. Once a signal is captured, it is decoded into a hexadecimal code. Each button on the IR remote corresponds to a specific hexadecimal code. These codes can be identified by printing them out upon reception. Each valid code is mapped to a specific task. If the received code does

not match any predefined command from the controller, the system returns to waiting for a new signal. If the code is valid, the system continues to check for another conditions.

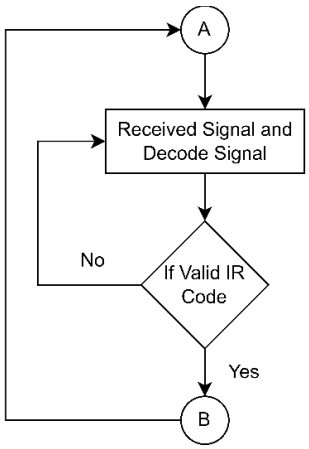


Figure 3.24: IR Signal Reception and Decoding.

The received valid signal is then cross-checked against predefined commands, each mapped to a specific task. This comparison is performed using an if-else conditional structure. The first condition checks whether the signal corresponds to the clockwise function, which triggers the stepper motor to rotate in the clockwise direction. If the signal does not match, the system then checks whether it corresponds to the counterclockwise function, which commands the motor to rotate in the opposite direction. Figure 3.25 illustrates the flowchart for both clockwise rotation and counterclockwise rotation.

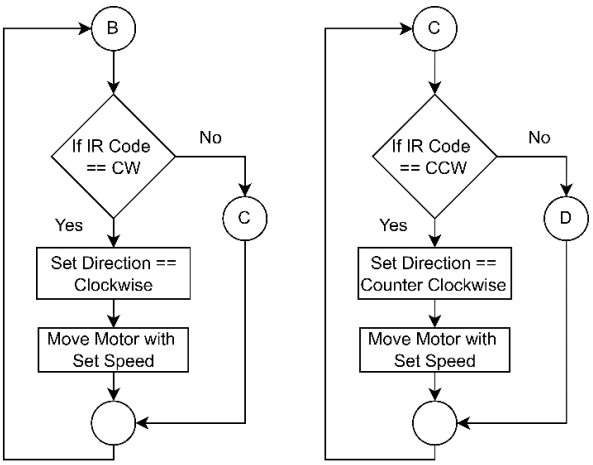


Figure 3.25: Manual Motor Rotation Control.

If the signal does not match clockwise and counterclockwise rotation, the third condition proceeds to check for the linear actuator’s up and down functions, as illustrated in Figure 3.26. If the decoded signal matches the 'UP' command, the linear actuator extends upward. Similarly, if the signal matches the 'DOWN' command, the actuator retracts downward. If neither signal is matched, the system continues to the next condition check.

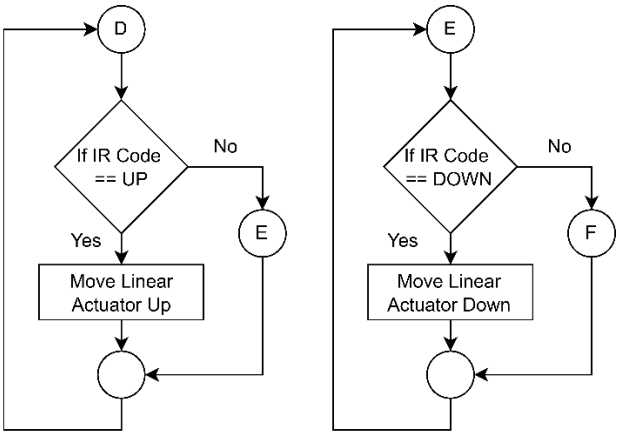


Figure 3.26: Manual Linear Actuator Control.

The next condition check determines the rehabilitation mode of the system, identifying whether it is intended for the forearm or the shank. Modes 1, 2, and 3 are predefined for forearm rehabilitation, while Modes 4 and 5 are designated for the shank. Once the received signal matches a specific mode, the stepper motor moves to the ready position, allowing the user to secure either the forearm or shank on the support structure for proper sensor attachment and calibration. Figure 3.27 illustrates the flowchart detailing the condition check for each rehabilitation mode.

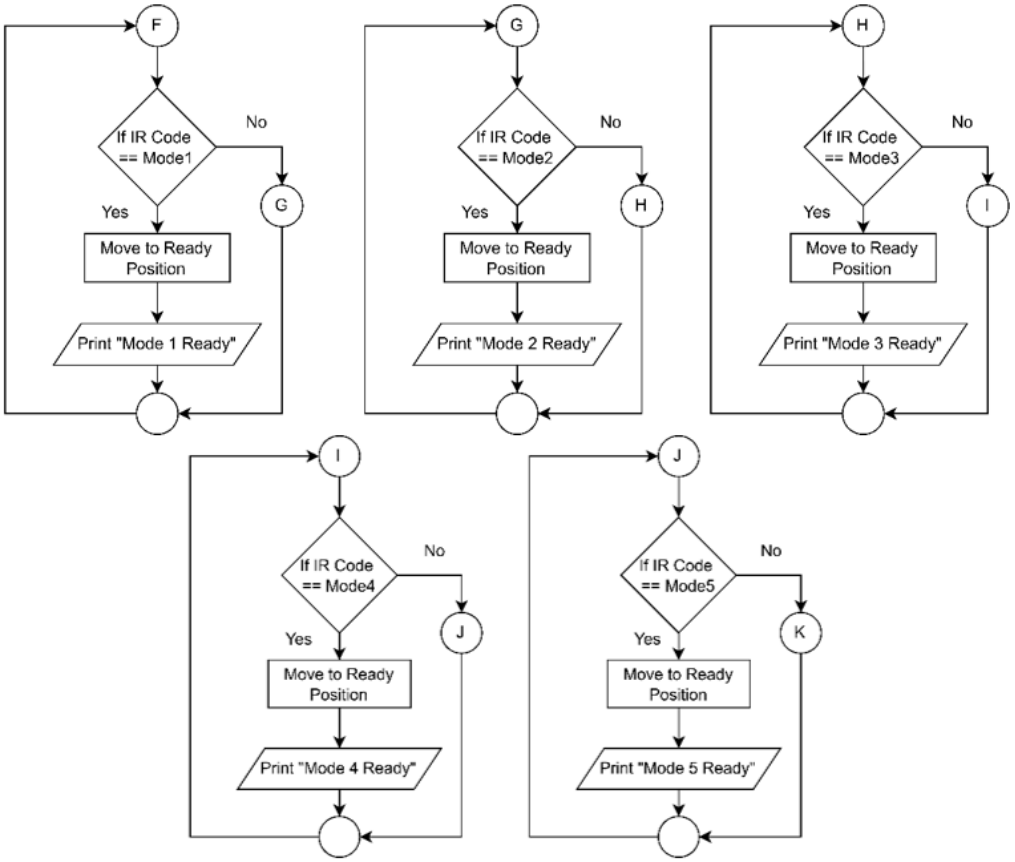


Figure 3.27: Rehabilitation Mode Selection.

If the decoded valid signal does not match any prior conditions, the system proceeds to check for the 'Start' function, as illustrated in Figure 3.28. Execution of this function requires an AND condition which a rehabilitation mode must already be in the ready state. If this condition is satisfied, the system identifies the active mode. For instance, if Mode 1 is ready, the system initiates the start function by toggling the motor between 20° and 90°, as shown in Figure 3.29. Throughout each movement cycle, the system continuously monitors the stop button. If pressed, the motor stops immediately, and the system exits the current loop, as depicted in Figure 3.30.

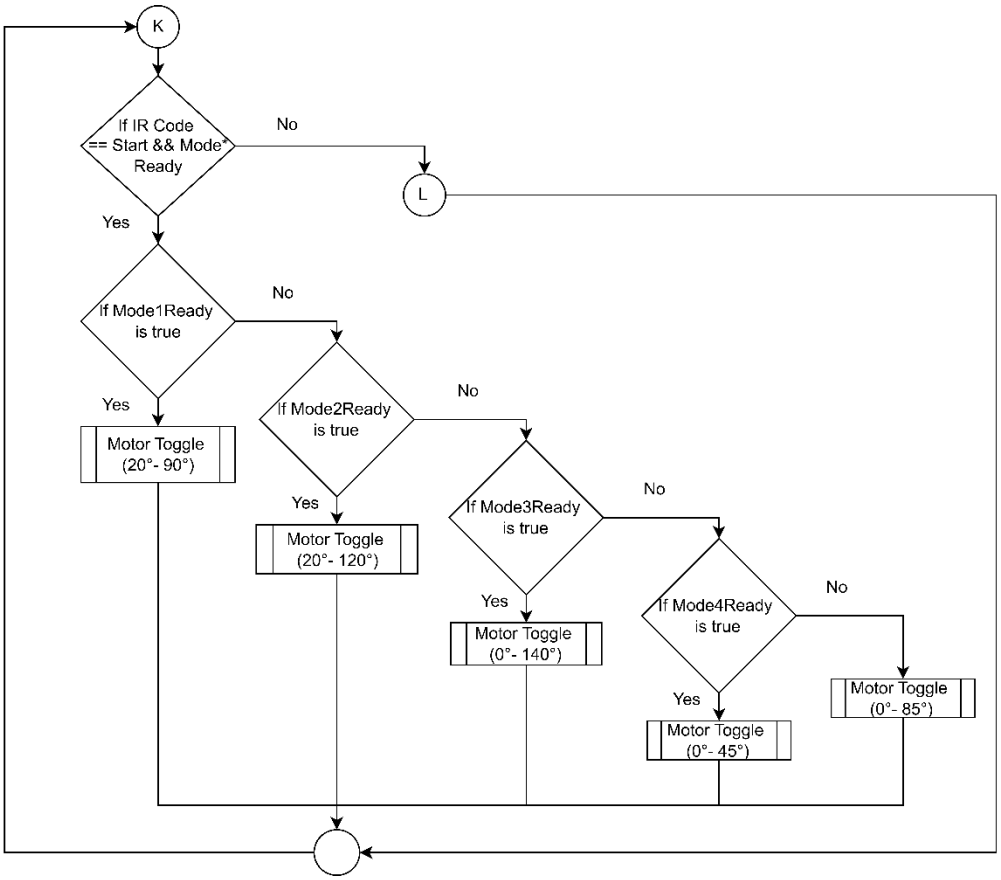


Figure 3.28: Start Ready Mode.

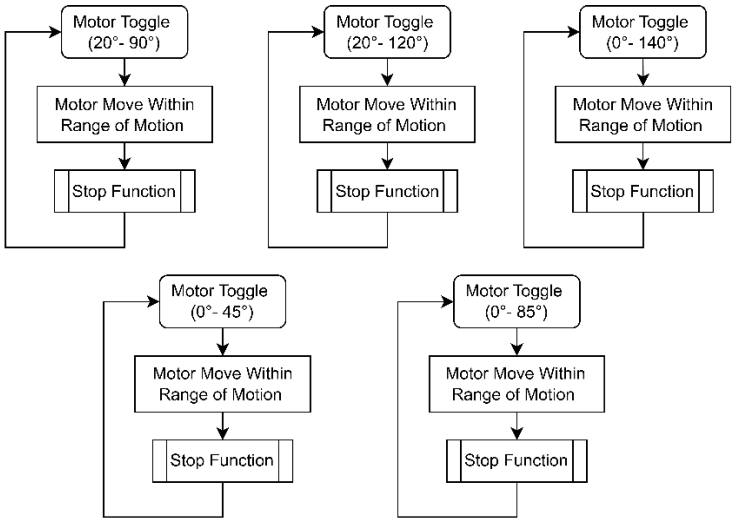


Figure 3.29: Motor Toggle Function for Each Mode.

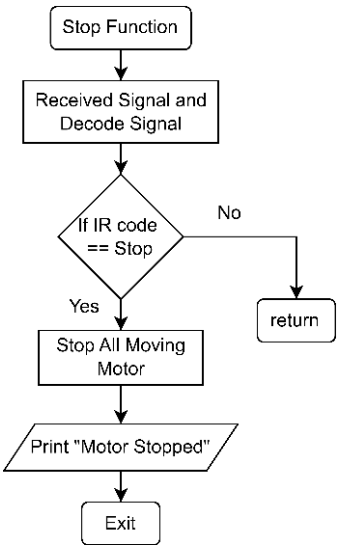


Figure 3.30: Stop Function.

If the start button is not pressed, the system proceeds to check additional conditions. The next condition involves the 'Complete' button, as shown in Figure 3.31. If a valid complete signal is received, the system commands the motor to return to the zero position from its current stopped position which essentially bringing it back to its original starting point.

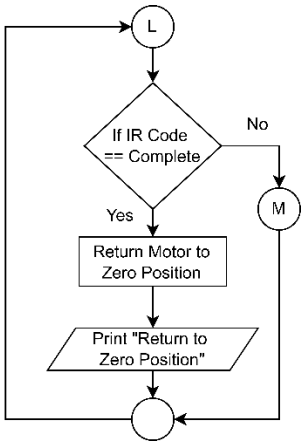


Figure 3.31: Return to Zero Position.

The final condition involves checking for a repeated code, which corresponds to the long-press function. This function is only applicable in manual control mode, allowing the motor to turn continuously in either the clockwise or counterclockwise direction. When the user long-presses a button on the IR remote, a decoded repeat signal of 0x0 (hexadecimal) is received. If

this signal is detected, the system then verifies whether the last valid code was for clockwise or counterclockwise rotation. If it was neither, the repeated signal is ignored. However, if the last valid code corresponds to clockwise rotation, the motor will continue moving in that direction. The same applies for counterclockwise movement. The flow of this repeated signal process is illustrated in Figure 3.32.

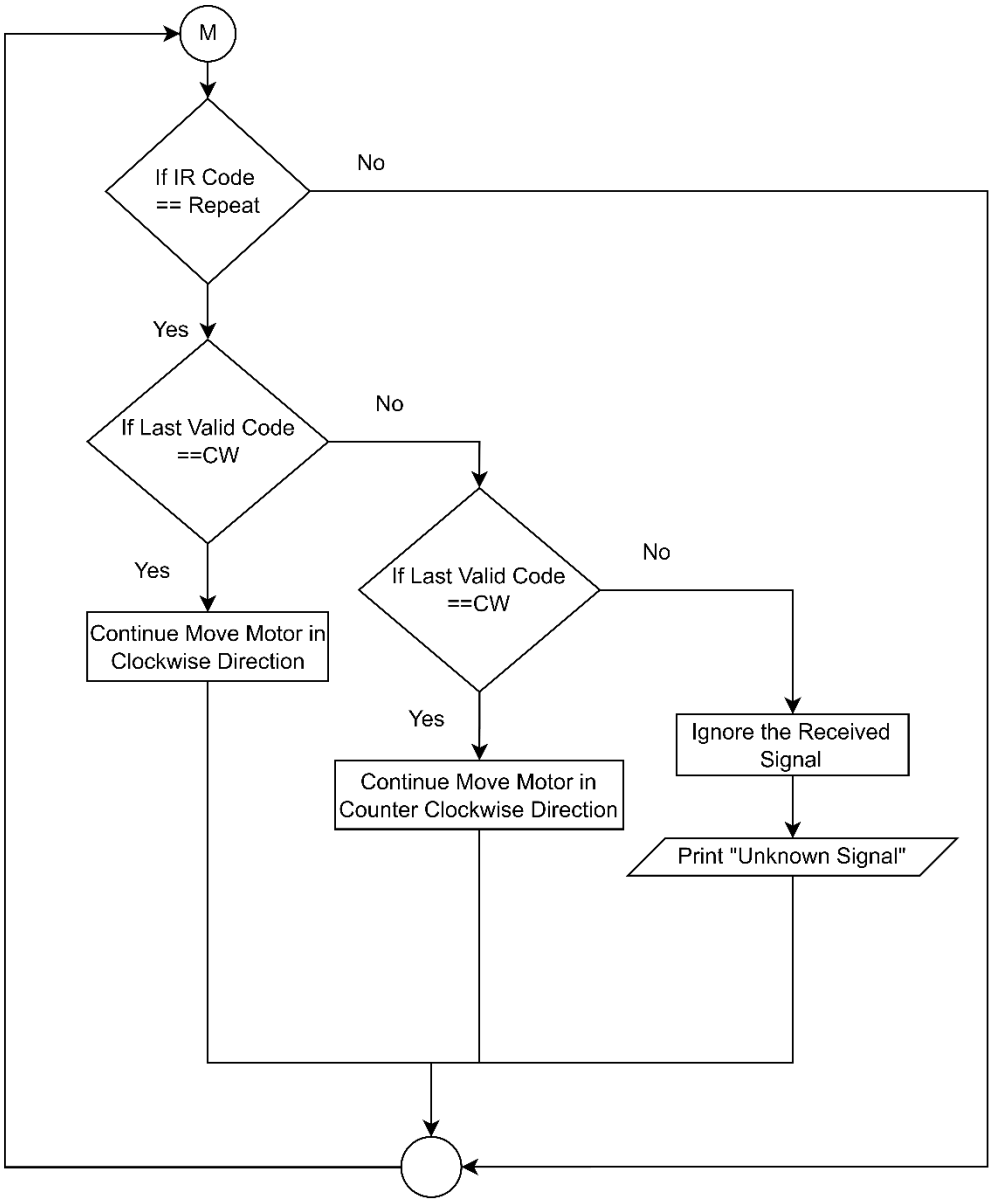


Figure 3.32: Repeat Signal Control.

3.8 Cost of Material Used

Table 3.9 presents the costs of all materials used in developing the stationary upper and lower extremities rehabilitation system, with a total cost of RM 1,587.34, highlighting its cost-effectiveness.

Table 3.9: Material Costs for Prototype Development.

No.	Item Name	Quantity	Total Cost (RM)
1.	Nema 34 Stepper Motor 4.5 Nm	1	190.00
2.	Stepper Motor Driver DM860H	1	54.84
3.	Planetary Gear Reduction	1	212.00
4.	Linear Actuator	1	250.00
5.	Aluminium Profile	1	232.00
6.	L Bracket	24	95.04
7.	Caster Wheel	4	120.00
8.	AC-DC 48V Power Supply	1	97.89
9.	8-60V to 1-36V Step Down Transformer	1	22.33
10.	ESP32	2	50.00
11.	ESP32 Base	1	8.90
12.	Relay Board	1	4.95
13.	Rocker Switch	1	3.80
14.	Emergency Stop	1	9.80
15.	3D Print	1	100.00
16.	Mild Steel Hollow Bar	1	8.24
17.	Mild Steel Plate	1	29.03
18.	Plywood	1	9.00
19.	Hand Grip	1	6.99
20.	Stainless Steel Rod	1	23.03
21.	Velcro Strap	1	5.00
22.	MPU6050 Sensor	1	8.29
23.	Mini Breadboard	1	1.29
24.	IR Controller	1	5.90
25.	LCD Screen	1	8.69

Table 3.9 (Continued)

26.	3-pin Plug	1	1.65
27.	3-pin Wire	1	6.30
28.	Jumper Wire	1	2.59
29.	Soft Silicone Tinned Copper Wire	1	19.79
		Grand Total Cost (RM)	1587.34

3.9 Calibration of Rotational Angle for Stepper Motor

The precise control of the rotational angle of the stepper motor is a crucial part of the rehabilitation activities. Any miscalculation of the motor steps may lead to over-flexion or over-extension of the human limb, potentially causing injury. Therefore, the rotational angle of the stepper motor is carefully determined through both mathematical calculation and implementation in the Arduino code. The stepper motor used in this project is a typical model NEMA 34, which has a default step angle of 1.8°. This means that without any micro stepping, the number of steps needed to complete one full revolution of 360 degrees is calculated by dividing 360° by 1.8°, resulting in 200 full steps per revolution. However, the motor driver (DM860H) supports micro stepping, a method that divides each full step into smaller micro steps for smoother and more precise control. With 1/8 micro stepping enabled, each full step is divided into 8 micro steps. This increases the total number of steps per revolution to 1600 micro steps.

This is where the 1600 steps per revolution value originates. In this setup, although the motor itself requires 1600 steps to complete one full rotation, a 50:1 gear system is attached. This means that the output shaft of the motor completes one full rotation only after the motor itself has rotated 50 times. Therefore, the total number of steps required to achieve a full 360-degree rotation at the output shaft is 80,000 steps per output shaft revolution.

To determine how many steps are needed for one degree of rotation at the output shaft, the total number of steps, 80,000 is divided by 360 degrees, which gives 222.22 steps per degree. Using this steps-per-degree value, the number of steps needed for common rotation angles can be pre-calculated as in Table 3.10 and implemented in the code.

Table 3.10: Steps-per-degree Required to Achieve the Rotational Angle.

Angle (°)	Steps Required
30	6666.60
40	8888.80
45	9999.90
50	11111.00
70	15555.40
85	18888.70
90	19999.80
100	22222.00
140	31110.80

The pre-calculated step values are integrated into the program and utilized when selecting specific movement modes via an IR remote control. Each mode commands the stepper motor to rotate a defined number of steps in either the clockwise or counterclockwise direction to achieve a desired angle. Figure 3.33 illustrates the manual calibration process for the stepper motor angle. The structural support is first adjusted to the position shown in Figure 3.33(b), and a levelling measurement app on a smartphone is used to ensure the surface is parallel to the floor. This position is then defined as the zero reference point in the Arduino IDE code. All subsequent rotations are performed relative to the zero position.

(a) (b)

Figure 3.33: Manual Adjustment of Zero Position (a) Top View (b) Front View.

3.10 Functionality Test and Data Analysis

Functionality testing is a crucial phase in validating the performance of the developed prototype. This process involves assessing the accuracy of the accelerometer and gyroscope module (MPU6050) sensor by comparing its measurements with those of the embedded IMU in the Trigno EMG sensor. Range of motion (ROM) data from the MPU6050 are cross-referenced with the Trigno IMU to evaluate measurement consistency and ensure reliable motion tracking throughout each rehabilitation cycle. Subsequently, the system's ability to capture limb ROM, acceleration, and angular velocity is analyzed under varying speeds and ROM conditions to assess biomechanical performance and sensor accuracy.

A total of eight healthy participants (four males and four females), with a mean age of 23.4 ± 0.69 years and a BMI distribution of 5:2:1 across normal, underweight, and overweight categories, were recruited. Inclusion criteria required no history of upper or lower limb injuries or neurological disorders. All tests were conducted on the participants' right limbs and were carefully supervised to ensure safety. An emergency stop button was incorporated into the system to allow immediate shutdown in the event of discomfort, malfunction, or other unforeseen issues. The testing protocol was designed as below to ensure both accurate data acquisition and participant well-being.

(i) Mode 1 is selected using the remote control. The support structure automatically moves to the ready position, with the rotation speed preset to 13.5°/s.

(ii) The participant is seated on a chair, ensuring the elbow joint is aligned parallel to the pivot point of the support structure.

(iii) The participant's forearm is secured with a Velcro strap, and the hand is positioned on the handle.

(iv) The MPU6050 is attached to the participant's forearm, aligned along the same axis as the Trigno EMG sensor, which is also placed on the forearm.

(v) The calibration button on the MPU6050 is pressed to calibrate the sensor.

(vi) The recording button on the serial monitor is pressed to begin data collection. The Trigno EMG system automatically calibrates at the start of recording.

(vii) The support structure passively moves the forearm through 3 full cycles of flexion and extension.

(viii) Upon completion, the stop button on the remote control is pressed to end data recording.

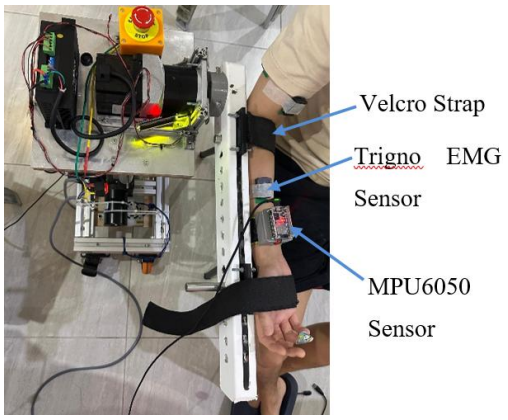
(ix) The complete button on the remote control is pressed to return the support structure to the starting position.

(x) Steps (i)–(ix) are repeated for Mode 2 and Mode 3, each with a different range of motion.

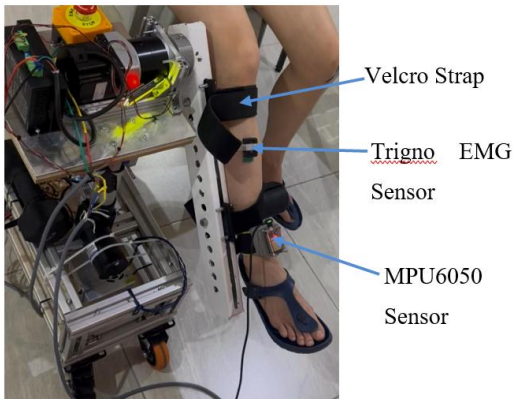
(xi) After all 3 modes are completed at 13.5°/s, the experiment is repeated with the next preset speed of 29.25°/s.

(xii) After the upper extremity tests are completed, the experiment is continued with the lower extremities (Mode 4 and Mode 5), ensuring the knee joint is aligned parallel to the pivot point.

The Figure 3.34 illustrates both the upper and lower extremities during testing, showing the positioning and setup for data collection. It describes how each limb is securely fastened to the support structure using adjustable Velcro straps to ensure stability and prevent unwanted movement. For the upper extremity, the forearm is strapped to the support with the hand firmly positioned on the handle. For the lower extremity, the leg is secured with the knee joint aligned parallel to the pivot point of the support structure to ensure accurate joint movement tracking. The sensor placement is also explained. The MPU6050 is attached directly to the limb segment being moved, positioned on the forearm for upper extremity tests and on the lower leg for lower extremity tests. The Trigno EMG sensor is placed along the same axis as the MPU6050, targeting the primary muscle group responsible for the movement, to ensure synchronized kinematic and muscle activity data collection.



(a)



(b)

Figure 3.34: Limb Secure on Developed Prototype. (a) Upper Extremities (b) Lower Extremities.

Additionally, the Table 3.11 provides a clear description of each mode, specifying the movement type, range of motion, and the corresponding limb for each mode tested. Meanwhile, Figure 3.35 illustrates the zero position of the support structure.

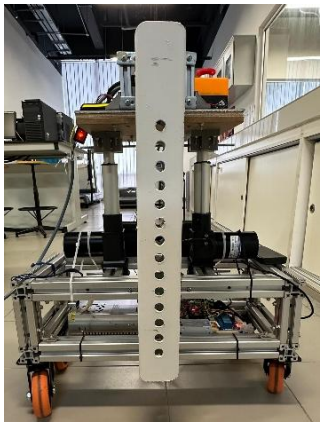


Figure 3.35: Zero Position of Support Structure.

Table 3.11: Range of Motion (ROM) of Each Mode.

Human Limb	**Mode**	**Range of Motion (ROM)**
Forearm	1	20° - 70°
	2	20° - 120°
	3	0° - 140°
Shank	4	0° - 45°
	5	0° - 85°

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This section provides a comprehensive discussion of the developed prototype of the stationary upper and lower extremities rehabilitation system. It includes a detailed description of the prototype, accompanied by labelled diagrams and a demonstration of the user interface. A user manual is also provided to guide system operation. Functionality tests were conducted to validate the accelerometer and gyroscope module (MPU6050) sensor, as well as the stepper motor. Performance was evaluated by analysing sensor data collected in various operating modes from eight healthy individuals. Furthermore, safety features and the system's relevance to the Sustainable Development Goals (SDGs) are discussed.

4.2 Complete Prototype

Figure 4.1 shows the front view of the final completed prototype, weighing approximately 32.76 kilograms. The design features a CNC-fabricated support structure that is securely connected to a stepper motor via a 3D-printed coupler. The motor is mounted on a robust mild steel platform, which is further reinforced with a wooden plate to enhance stability and support. For mobility and ease of handling, the prototype is equipped with four caster wheels, each capable of supporting up to 100 kilograms—well above the total weight of the system. The electronic circuit board is neatly enclosed within the aluminium profile frame to ensure protection and organization. Additionally, two linear actuators are installed beneath the platform to provide support and assist in maintaining the alignment of the motor assembly during rehabilitation activity.

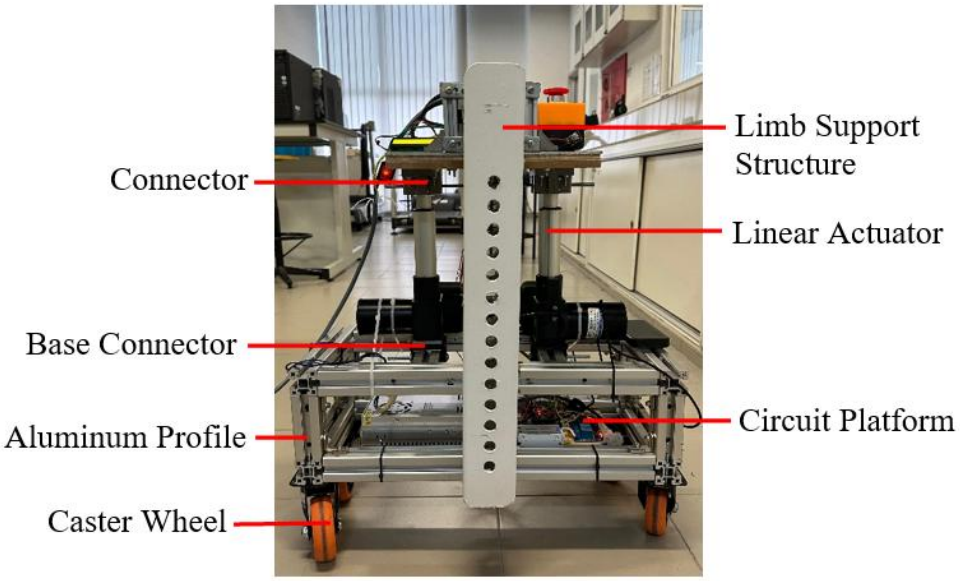


Figure 4.1: Complete Prototype (Front View).

The circuit connection is illustrated in the accompanying Figures 4.2 and 4.3. The ESP32 microcontroller is powered by a stable 5V supply. The main power supply delivers energy to both the stepper motor and the linear actuators. A step-down transformer is used to reduce the 48V input from the power supply to 12V, which matches the operating voltage of the linear actuators. A rocker switch is integrated to control the flow of electricity from the main socket to the power supply. For safety, an emergency stop button is also included, allowing the system to be halted immediately in case of any malfunction. The stepper motor is controlled via a motor relay, which is linked to the ESP32 and managed through the developed software. Additional relays are used to control the upward and downward motion of the linear actuators. An IR receiver is connected to the ESP32, enabling remote control of the entire system. Lastly, an LCD screen is used to display the current operational status of the system.

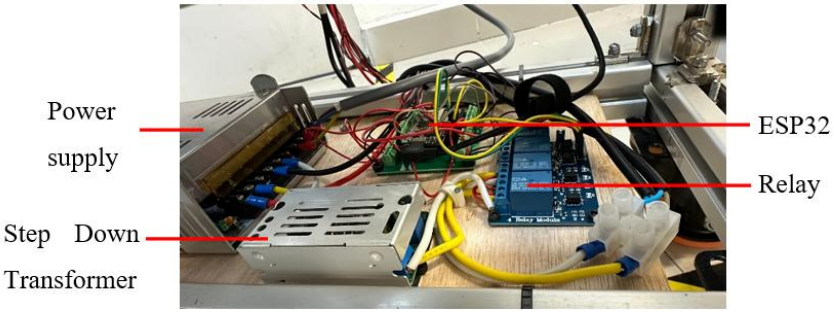


Figure 4.2: Circuit Connection on Plywood Platform.

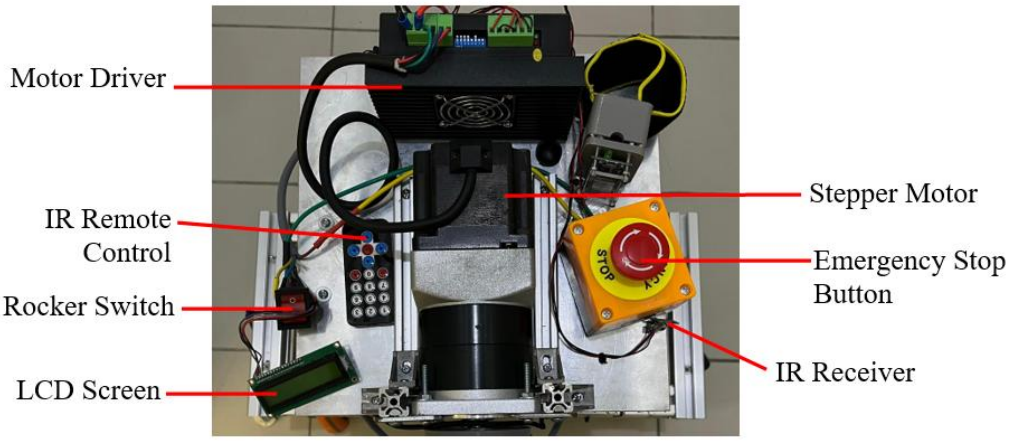


Figure 4.3: Circuit Connection on Mild Steel Platform.

Figures 4.4 and 4.5 illustrate the accelerometer and gyroscope module (MPU6050) sensor, which is enclosed in a custom-designed 3D-printed casing. The ESP32 microcontroller is mounted on the top cover of the casing, while a Velcro strap is sewn onto the enclosure to ensure secure attachment during use.

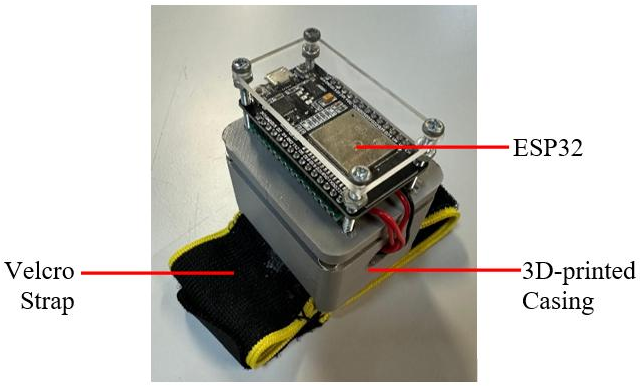


Figure 4.4: Sensor System.

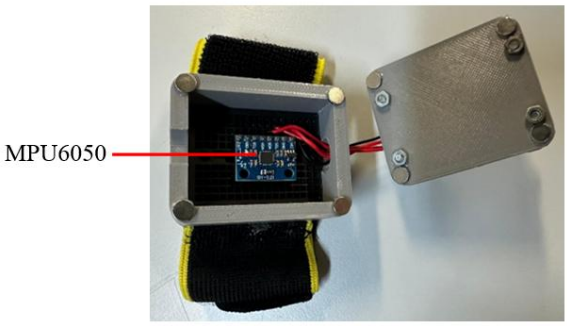


Figure 4.5: Accelerometer and Gyroscope Module (MPU6050) Sensor with Casing.

4.3 User Interface Design

A user interface has been developed to display measured data to the user in a clear and organized manner. Figure 4.6 illustrates the sign-up and login pages of the interface, which are designed to ensure that only authorized users can access the system. This feature enhances data privacy and security by restricting access to personal rehabilitation records.

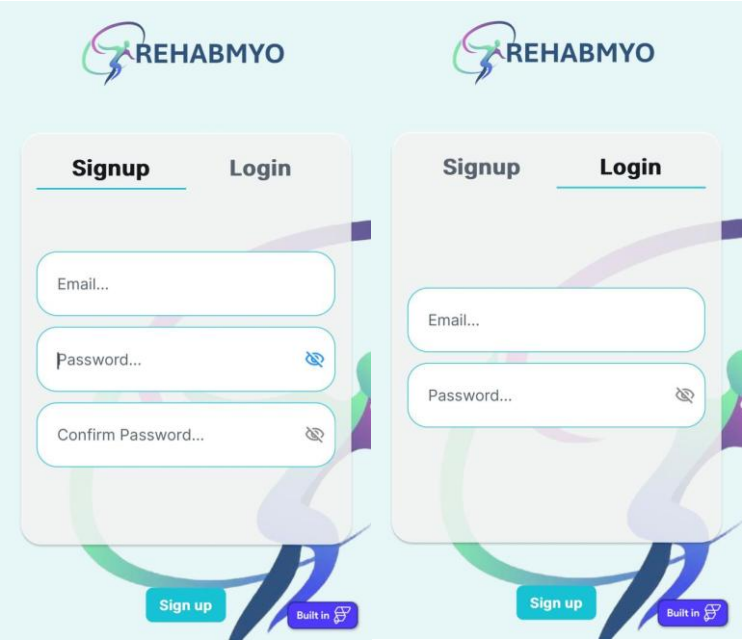


Figure 4.6: Signup and Login Page.

The Figure 4.7 further illustrates the main interface of the developed user interface which consists of a Summary Page and an Account Page. The Summary Page includes two icons and selecting either one displays the corresponding results of the rehabilitation activity retrieved from database as shown in the Figure 4.8. Three graphs are presented to visualize the data, including Range of Motion (ROM), acceleration, and gyroscope measurements. The Account Page displays the user's personal information such as profile picture, name, email, weight, height, and date of birth.

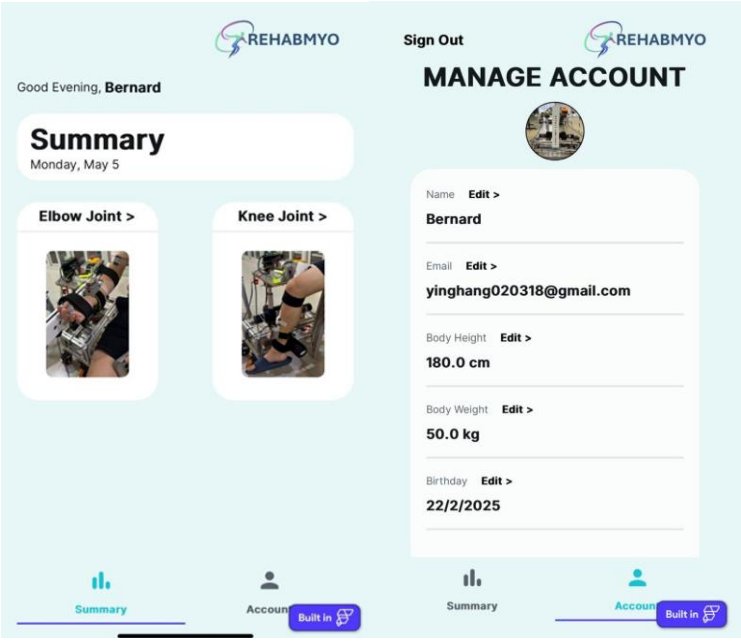


Figure 4.7: Summary and Account Page.

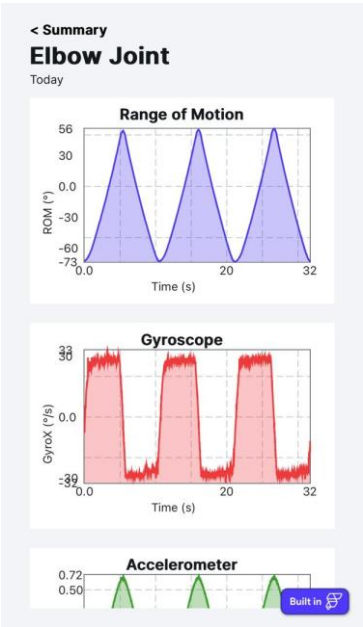


Figure 4.8: Result Page.

4.4 User Manual

In this section, a detailed description of the overall functional process of the rehabilitation system is presented in Table 4.1, outlining the rehabilitation activities for both the upper and lower extremities. This serves as a general user manual for operating the system. Additionally, Figure 4.9 illustrates the button

layout and corresponding functions of the IR remote control used to interact with the system.

Figure 4.9: IR Remote Control – Button Layout and Labels.

Table 4.1: User Manual.

Activity	Instruction
Perform Rehabilitation Activity 1: Elbow Extension and Flexion (20° - 90°)	1. Press ① to select Rehabilitation Activity 1. 2. Wait for the support structure to move to the ready position. 3. Sit beside the machine and ensure your elbow joint is aligned with the pivot point of the support structure. 4. Secure your forearm onto the support structure, as shown in the Figure 4.10 (a). 5. Press ⑦ to begin the rehabilitation session. Press ⑧ to stop the rehabilitation session at any time. 6. Carefully remove your forearm from the support structure. 7. Press ⑥ to return the support structure to the zero position, as shown in the Figure 4.11.

Table 4.1 (Continued)

Perform Rehabilitation Activity 2: Elbow Extension and Flexion (20° - 90°)	1. Press ② to select Rehabilitation Activity 2. 2. Wait for the support structure to move to the ready position. 3. Sit beside the machine and ensure your elbow joint is aligned with the pivot point of the support structure. 4. Secure your forearm onto the support structure, as shown in the Figure 4.10 (a). 5. Press ⑦ to begin the rehabilitation session. 6. Press ⑧ to stop the rehabilitation session at any time. 7. Carefully remove your forearm from the support structure. 8. Press ⑥ to return the support structure to the zero position, as shown in the Figure 4.11.
Perform Rehabilitation Activity 3: Elbow Extension and Flexion (0° - 140°)	1. Press ③ to select Rehabilitation Activity 3. 2. Wait for the support structure to move to the ready position. 3. Sit beside the machine and ensure your elbow joint is aligned with the pivot point of the support structure. 4. Secure your forearm onto the support structure, as shown in the Figure 4.10 (a). 5. Press ⑦ to begin the rehabilitation session. 6. Press ⑧ to stop the rehabilitation session at any time. Carefully remove your forearm from the support structure. 7. Press ⑥ to return the support structure to the zero position, as shown in the Figure 4.11.

Table 4.1 (Continued)

Perform Rehabilitation Activity 3: Knee Extension and Flexion (0° - 45°)	1. Press ④ to select Rehabilitation Activity 4. 2. Wait for the support structure to move to the ready position. 3. Sit beside the machine and ensure your knee joint is aligned with the pivot point of the support structure. 4. Secure your shank onto the support structure, as shown in the Figure 4.10 (b). 5. Press ⑦ to begin the rehabilitation session. 6. Press ⑧ to stop the rehabilitation session at any time. 7. Carefully remove your shank from the support structure. 8. Press ⑥ to return the support structure to the zero position, as shown in the Figure 4.11.
Perform Rehabilitation Activity 5: Knee Extension and Flexion (0° - 85°)	1. Press ⑤ to select Rehabilitation Activity 5. 2. Wait for the support structure to move to the ready position. 3. Sit beside the machine and ensure your knee joint is aligned with the pivot point of the support structure. 4. Secure your shank onto the support structure, as shown in the Figure 4.10 (b). 5. Press ⑦ to begin the rehabilitation session. 6. Press ⑧ to stop the rehabilitation session at any time. 7. Carefully remove your shank from the support structure. 8. Press ⑥ to return the support structure to the zero position, as shown in the Figure 4.11.

Table 4.1 (Continued)

Move up or down of support structure.	1. Press ⑨ to move up the support structure. 2. Press ⑩ to move down the support structure.
Turn clockwise or counterclockwise of support structure.	1. Press ⑪ to move the support structure counterclockwise. 2. Press ⑫ to move the support structure clockwise.

(a) (b)

Figure 4.10: Proper Limb Positioning on the Support Structure (a) Forearm (b) Shank.

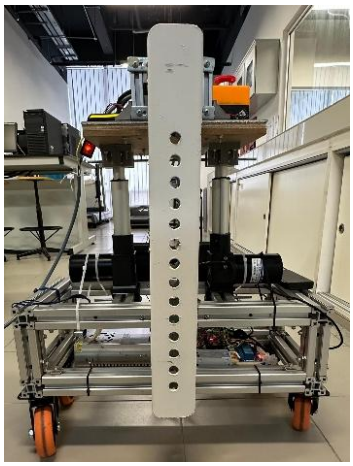


Figure 4.11: Zero Position of Support Structure.

4.5 Functionalities and Performance Evaluation of Completed Prototype.

This section presents the results of assessments conducted to evaluate the prototype's ability to perform elbow and knee joint movements, specifically flexion and extension across various ranges of motion (ROM). Tests were carried out on the right-side limbs of subjects with different BMI levels to examine the system's adaptability. The Table 4.2 summarizes the activities completed during functionality testing. Additionally, the accuracy of the accelerometer, gyroscope, and motor speed was analysed to validate system performance.

Table 4.2: Functionality Test Activities.

Activities	Range of Motion, ROM (°)
Elbow Extension and Flexion	20° - 90°
	20° - 120°
	0° - 140°
Knee Extension and Flexion	0° - 45°
	0° - 85°

4.5.1 Validation of Accelerometer and Gyroscope Module Sensor

Table 4.3 and the accompanying Figures 4.12 and 4.13 present the validation results of the accelerometer and gyroscope module (MPU6050) sensor by comparing its performance with the embedded IMU in the Trigno EMG sensor. Data were collected from eight participants performing elbow and knee flexion and extension movements across specified ranges of motion.For each activity, the mean and standard deviation of the recorded range of motion values were calculated for both sensors. As shown in Table 4.2, the MPU6050 consistently recorded lower range of motion values compared to the Trigno IMU across all tasks. The largest percentage error, 15.73 percent, was observed during knee flexion in the 0°–45° range, highlighting reduced accuracy in detecting smaller joint movements. In contrast, the smallest error, 5.25 percent, occurred during the 0°–85° knee range of motion task, indicating improved agreement in larger flexion activities. For elbow joint movements, the error ranged from 6.54

percent to 8.89 percent, suggesting moderate accuracy. Overall, the average percentage error across all measured activities was 8.79 percent, indicating that the MPU6050 delivers reasonably accurate joint movement data, although its precision varies depending on the joint and the range of motion involved.

Table 4.3: Percentage Error of MPU6050 Sensor.

Extension and Flexion		**Range of Motion, ROM (°)**				**Percentage Error (%)**
		MPU6050		**Trigno (IMU)**		
		Mean	**STD**	**Mean**	**STD**	
Elbow Joint	**20°-90°**	61.32	±3.53	67.30	±5.17	8.89
	20°-120°	90.93	±4.37	98.37	±6.32	7.56
	0°-140°	120.08	±6.74	128.48	±10.06	6.54
Knee Joint	**0°-45°**	49.44	±3.83	42.72	±4.17	15.73
	0°-85°	74.44	±6.48	78.56	±5.39	5.25
Overall Percentage Error (%):						**8.79**

Figures 4.12 and 4.13 provide visual comparisons of the ROM profiles captured by both sensors over normalized time. In the forearm analysis, three elbow flexion activities were examined, which are 20°–90°, 20°–120°, and 0°–140°. Across all these conditions, the MPU6050 consistently underestimated range of motion compared to the Trigno IMU. This underestimation became more pronounced in larger range of motion tasks. However, the motion trajectories' shape and timing remained closely aligned between the two sensors, indicating that while the MPU6050 reported lower amplitude values, it effectively captured the overall motion pattern.

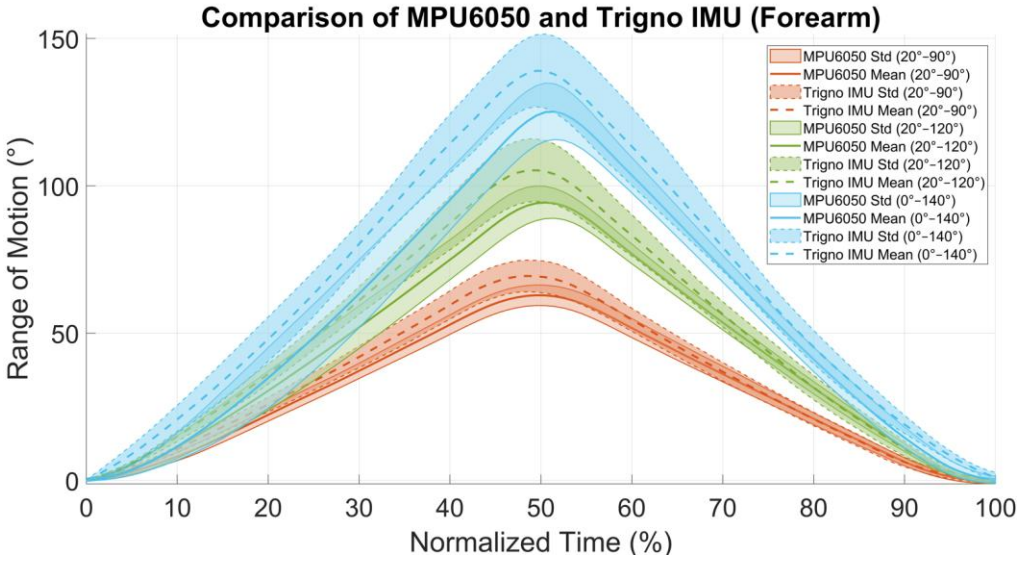


Figure 4.12: Comparison of Elbow ROM Captured by MPU6050 and Trigno IMU Over Normalized Time.

For the lower limb analysis, shown in Figure 4.13, two knee flexion tasks were assessed: 0°–45° and 0°–85°. Similar to the elbow results, the MPU6050 underreported range of motion, particularly in the smaller 0°–45° condition where the deviation from the Trigno IMU was more significant. In contrast, the 0°–85° task showed much closer alignment between the sensors.

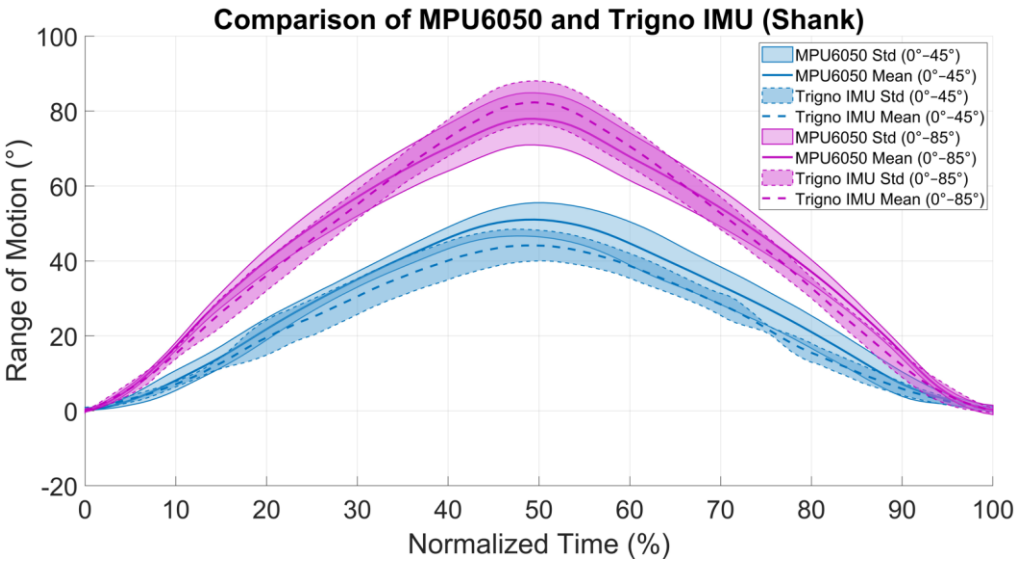


Figure 4.13: Comparison of Knee ROM Captured by MPU6050 and Trigno IMU Over Normalized Time.

Several factors may contribute to the discrepancies observed, including the influence of gravitational forces at higher flexion angles, especially in the forearm, as well as sensor attachment stability, vibrations, and potential electromagnetic interference. These elements may disproportionately affect the accuracy of the MPU6050. In summary, the validation results demonstrate that the MPU6050 is capable of capturing joint movement profiles with acceptable accuracy for many applications. While it tends to underestimate range of motion, especially in smaller movement ranges, it remains suitable for functional joint monitoring in rehabilitation settings. Incorporating calibration or compensation techniques may further enhance its performance, particularly in contexts that demand higher measurement precision.

4.5.2 Validation of Stepper Motor Setting

Proper configuration of the stepper motor is a crucial aspect in the development of this rehabilitation prototype, as it functions as the primary actuator responsible for driving controlled movement in both the upper and lower limbs. One of the key configurable parameters of the stepper motor is its rotational speed. In this system, two rotational speeds were implemented, 29.25 °/s as high speed and 13.50 °/s as low speed. These theoretical speeds were programmed in the Arduino microcontroller by specifying a fixed number of steps per second, with detailed calculations provided in the Appendix E. To evaluate the accuracy of the implemented speeds, actual measurements were taken using the accelerometer and gyroscope module (MPU6050) sensor. The comparison revealed percentage errors of 2.22% and 3.41% for high and low speeds, respectively, with an overall error of 2.82% as shown in Table 4.4. These small deviations confirm the system’s capability to control motor speed accurately and consistently.

Table 4.4: Comparison of Theoretical and Measured Rotational Speeds of the Stepper Motor.

	Theoretical Speed (°/s)	**Practical Speed (°/s)**	**Percentage Error (%)**
High Speed	29.25	28.60 ± 0.11	2.22
Low Speed	13.50	13.04 ± 0.12	3.41
	Overall Percentage Error (%)		**2.82**

Beyond speed accuracy, it is essential to assess the consistency of the motor's performance across different operational conditions. A primary consideration in rehabilitation is ensuring that the range of motion (ROM) remains stable, regardless of changes in motor speed. Consistent ROM ensures that therapeutic exercises are delivered safely and effectively, avoiding unintended variations that could compromise the rehabilitation process or increase the risk of injury. To evaluate this, Table 4.5 presents the ROM achieved at both speed settings across several elbow and knee movements. The results show relatively small differences in ROM between high and low speeds, with an overall percentage difference of 4.75%. This indicates that the motor maintains a consistent output despite speed changes. For example, the elbow joint over a 0°–140° range shows only a 3.73% difference, while the knee joint over a 0°–85° range exhibits just a 1.45% difference. The largest observed discrepancy is 8.72% for the knee joint at a smaller ROM (0°–45°), likely due to mechanical loading effects or reduced angular resolution at smaller movements. Nevertheless, these results support the conclusion that the motor delivers repeatable and stable joint motion across a range of functional speeds.

Table 4.5: Percentage Difference in Range of Motion (ROM) Between Two Rotational Speeds for the Same Activity.

Extension and Flexion		Range of Motion (°)				Percentage Difference (%)
		High Speed (29.25 °/s)		Low Speed (13.5°/s)		
		Mean	STD	Mean	STD	
Elbow Joint	**20°-90°**	61.32	±3.53	59.46	±1.51	3.08
	20°-120°	90.93	±4.37	84.99	±1.70	6.75
	0°-140°	120.08	±6.74	115.68	±6.29	3.73
Knee Joint	**0°-45°**	49.44	±3.83	45.31	±4.65	8.72
	0°-85°	74.44	±6.48	75.53	±6.61	1.45
		Overall Percentage Difference (%):				**4.75**

In addition to evaluating speed and consistency, the system's ability to achieve precise rotational control was also assessed by comparing the preset ROM values with the actual angles measured by the accelerometer and gyroscope module (MPU6050) sensor. The overall percentage error across all tested motions was found to be 11.78% as shown in Table 4.6. The highest individual error occurred in the elbow joint for a 0°–140° ROM (15.70%), potentially due to sensor drift or accumulated error over a larger angular range. In contrast, the lowest error (5.69%) was recorded at the knee joint for a 0°–45° ROM, where motion is more constrained and predictable. These discrepancies highlight the influence of external factors such as sensor placement, human limb movement, and dynamic instability. Unlike a fixed mechanical setup, attaching the sensor to a human limb introduces variability through motion artifacts and potential misalignment, which may reduce measurement accuracy—particularly during wider or more dynamic movements. These findings emphasize the importance of not only programming accurate motor angles but also validating real-time sensor feedback to ensure safety and precision.

Table 4.6: Percentage Error in Range of Motion (ROM) Between Measured MPU6050 Values and Programmed Preset Values.

Extension and Flexion		Range of Motion (°)		Percentage Error (%)
		MPU6050 [Average ± SD]	Stepper Motor Preset ROM	
Elbow Joint	**20°-90°**	60.46±2.93	70	13.63
	20°-120°	87.88±4.49	100	12.12
	0°-140°	118.03±6.89	140	15.70
Knee Joint	**0°-45°**	47.56±4.65	45	5.69
	0°-85°	74.98±6.37	85	11.79
		Overall Percentage Error (%):		**11.78**

In conclusion, the stepper motor demonstrated reliable speed control, ROM consistency, and reasonable rotational accuracy, all of which are essential for effective rehabilitation therapy. While minor deviations were observed, they fall within acceptable limits for functional use, reinforcing the motor's suitability for clinical or at-home rehabilitation systems. Ongoing calibration and improvements in sensor placement and signal processing can further enhance system precision and user safety.

4.5.3 Accelerometer Data

Acceleration is one of the dynamic parameters that can be obtained from the accelerometer and gyroscope module (MPU6050) sensor. Figure 4.14 shows the three axes of the accelerometer, clearly illustrating how the module defines directional orientation. Each axis, X, Y, and Z, corresponds to a specific direction, allowing the sensor to detect changes in movement and position. Figure 4.15 demonstrates the response of the MPU6050 to acceleration during various ranges of motion. In this setup, the sensor is aligned with the sagittal plane. The Z axis is facing upward while the Y axis is positioned perpendicular to it. The acceleration detected along each axis is measured in units of gravitational force, commonly referred to as G. When the module is in its initial upright position, the Z axis reads approximately 1 G due to the effect of gravity. As the module rotates around the X axis, the Z axis value changes relative to its

orientation with gravity. It decreases from 1 G to 0 G and eventually reaches negative 1 G as it turns upside down.

For the Y axis, when the Z axis is facing upward, it is perpendicular to the direction of gravity and initially shows a reading of approximately 0 G. As the module rotates clockwise, the Y axis reading decreases from 0 G to negative 1 G before returning to 0 G. In contrast, during counterclockwise rotation, the Y axis value increases from 0 G to 1 G and then returns to 0 G. During motion in the sagittal plane, the X axis is not the primary direction of movement. Therefore, changes in acceleration along the X axis are usually minimal. Any small variations are likely caused by motion artifacts, slight disturbances, or improper sensor attachment due to the sensor not being securely fastened to the limb.

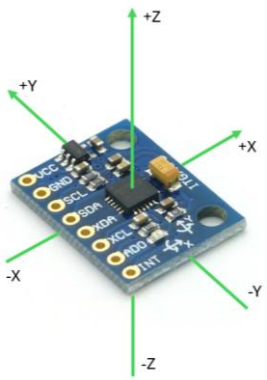


Figure 4.14: MPU6050 Orientation of Axes.

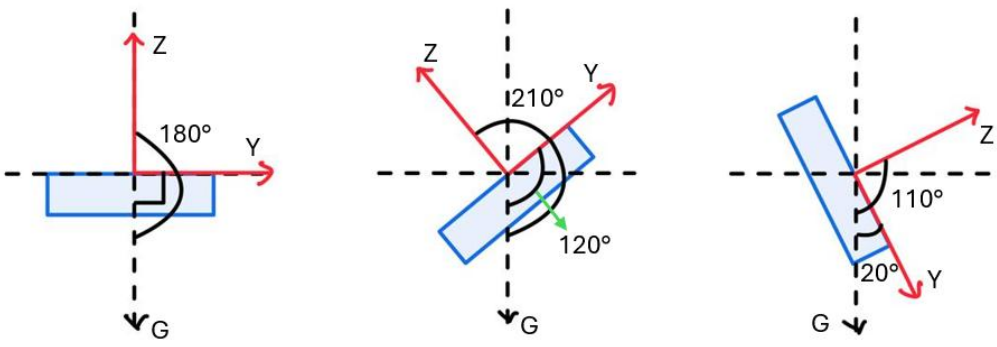


Figure 4.15: Orientation of Y and Z Axes Relative to Gravity.

The Figure 4.16 presents accelerometer data recorded during elbow flexion-extension movements. For elbow motion, forearm moves from approximately 20° to 90° in a counterclockwise direction. This motion is

reflected in the Y-axis acceleration, which increases from -1 G to 0 G during flexion, and then returns from 0 G to -1 G during extension. Similarly, the Z-axis acceleration begins at approximately 0.5 G during flexion, rises to 1 G at peak flexion, and decreases back to around 0.5 G during extension. The X-axis acceleration remains close to zero throughout the movement, indicating minimal lateral deviation and confirming that the motion occurs primarily within the sagittal plane.

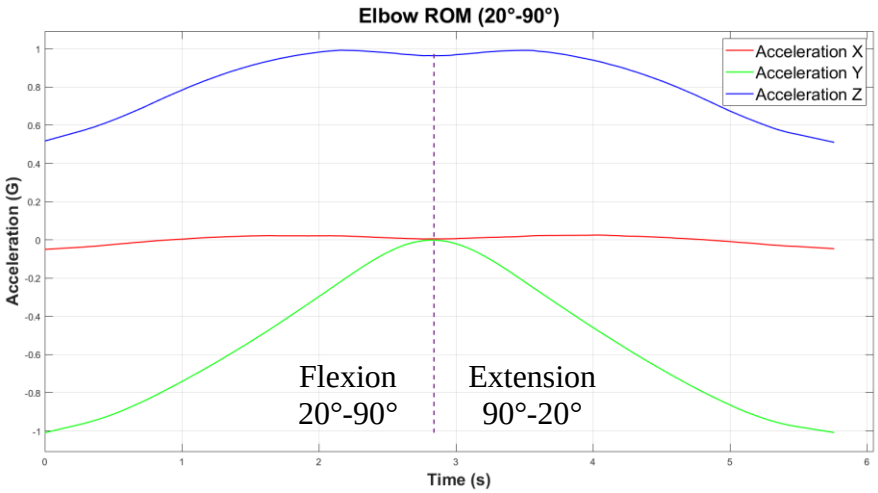


Figure 4.16: Acceleration vs. Time for X, Y, and Z Axes During Elbow Flexion and Extension.

Similar to the elbow flexion-extension, the knee extension-flexion movement as depicted in Figure 4.17 exhibits a consistent trend in the accelerometer data. The Y-axis and Z-axis accelerations form a bell-shaped curve, indicating smooth and controlled motion throughout the range of movement. These trends further support the reliability and stability of the system. Additionally, the X-axis acceleration remains close to zero during the entire movement, reinforcing that the motion is primarily constrained to the sagittal plane with minimal lateral displacement.

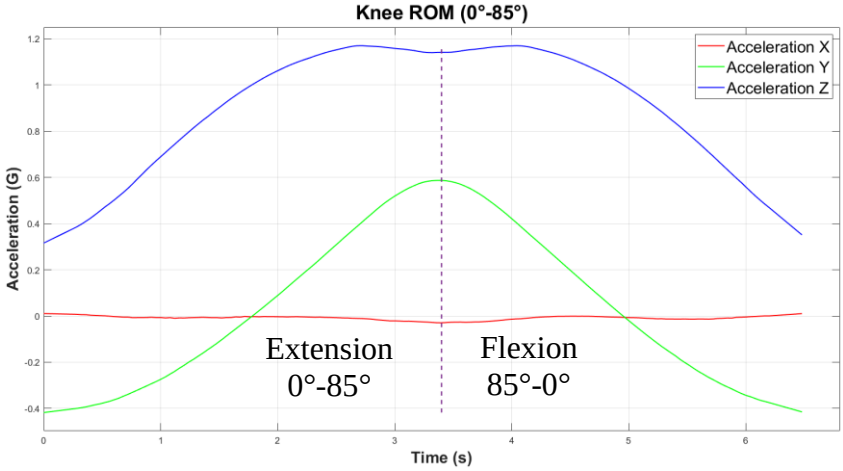


Figure 4.17: Acceleration vs. Time for X, Y, and Z Axes During Knee Flexion and Extension.

The data presented in the Figure 4.18 demonstrate that across varying speeds, the peak acceleration or G force remains relatively consistent. However, the time required to complete one full movement cycle, whether elbow extension followed by flexion or flexion followed by extension, varies depending on the speed setting. Higher speeds result in shorter cycle durations, while lower speeds correspond to longer cycle times. This observation confirms that the system operates with stable functionality under different rotational speeds, indicating reliable performance across a range of motion dynamics.

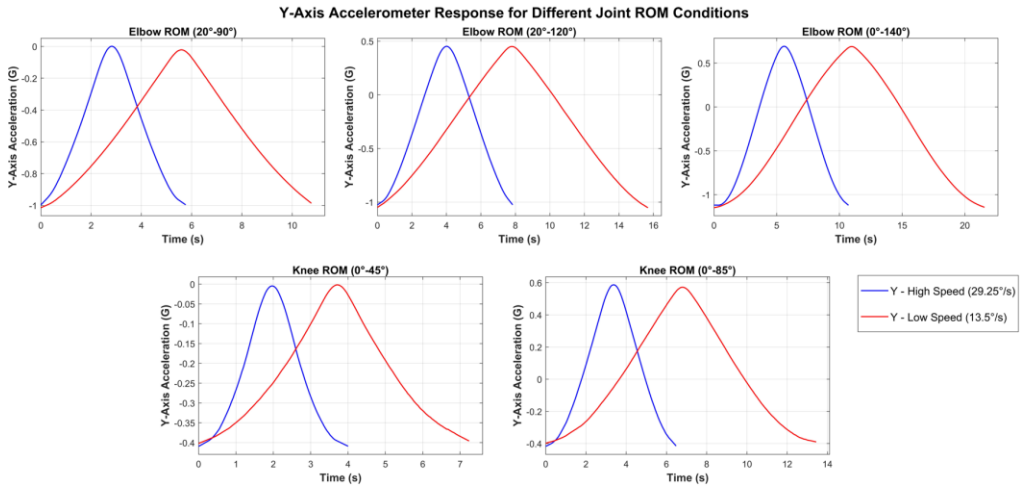


Figure 4.18: Acceleration vs. Time for Y-Axis at Varying Speeds During Elbow and Knee Flexion and Extension.

The Figure 4.19 illustrate the peak-to-peak mean acceleration measured in G-force along the Y-axis and Z-axis for both the forearm and shank during flexion-extension movements at two different angular speeds (13.5°/s and 29.25°/s) and various ranges of motion (ROM). For the forearm which described in the left chart, it is evident that increasing the ROM from 20°–90° to 0°–140° results in a noticeable rise in peak-to-peak mean acceleration, particularly along the Y-axis. This trend is attributed to both the greater angular displacement and the associated changes in gravitational influence on the MPU6050 as the sensor undergoes a wider arc. The acceleration along the Z-axis remains relatively stable across ROMs and speeds, suggesting a more consistent vertical component of movement or less variation in that axis.

In contrast, the shank data as in right chart shows a less dramatic increase in acceleration with increased ROM. While higher ROM (0°–85°) does lead to greater peak-to-peak values compared to the lower ROM (0°–45°), the difference is not as pronounced as in the forearm. Additionally, varying the speed between 13.5°/s and 29.25°/s does not significantly impact the G-force readings in either segment. This observation supports the idea that acceleration peaks are more sensitive to changes in ROM than to movement speed, especially when the movement is smooth and consistent. Overall, the data demonstrate that greater ROM contributes to higher acceleration peaks due to larger motion arcs and increased gravitational effects on the MPU6050, while speed changes have a comparatively smaller effect under controlled movement conditions.

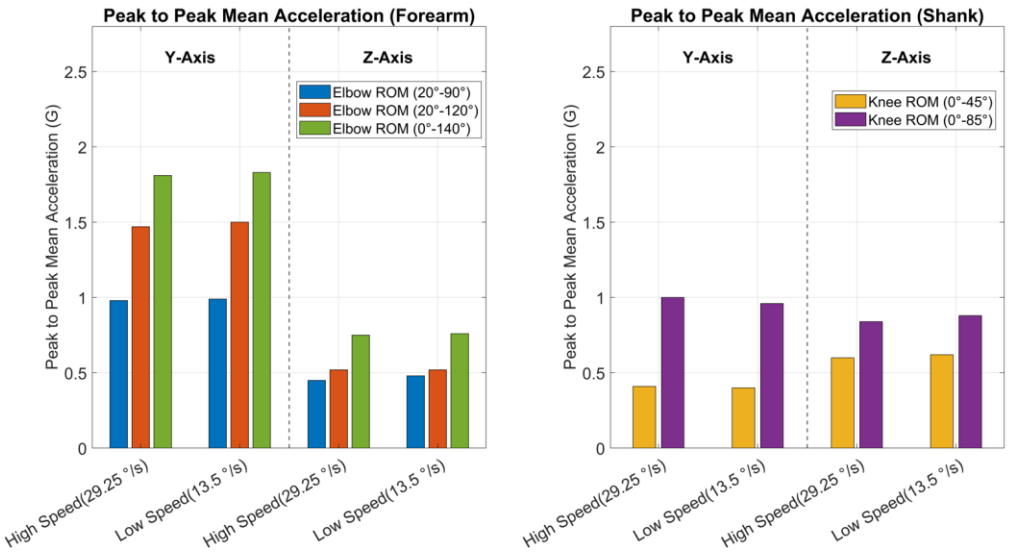


Figure 4.19: Comparison of Peak-to-Peak Mean Acceleration for Forearm and Shank at Different Ranges of Motion and Speeds.

4.5.4 Angular Velocity

The Figure 4.20 demonstrates how the rotation of the accelerometer and gyroscope module (MPU6050) sensor affects the data collected along each axis. As the sensor is placed on the limb, as shown previously, movements within the sagittal plane such as elbow flexion and extension and knee flexion and extension primarily influence the X-axis rotation. During these movements, the angular velocity along the X-axis increases and decreases in response to clockwise and counterclockwise rotations. These variations in angular velocity correspond to the rotational changes in the limb's motion, with the X-axis capturing the most significant data fluctuations due to the nature of the sagittal plane movement.

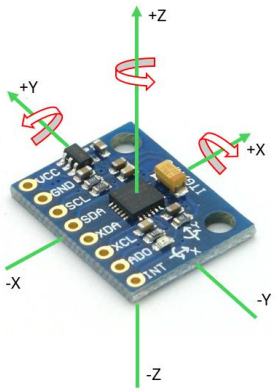


Figure 4.20: MPU6050 Gyroscope Axis Rotation.

The Figure 4.21 presents gyroscope data captured along the X, Y, and Z axes during elbow flexion and extension. Since elbow flexion and extension occur on the sagittal plane, the X-axis is particularly significant, exhibiting the most prominent changes and clearly reflecting the primary rotational movement of the elbow joint during the rehabilitation cycle. A transition from positive to negative values in the X-axis curve is observed, indicating a change in the direction of rotation as the movement shifts between flexion and extension. In contrast, the Z-axis shows only minor variations, suggesting that slight rotational deviations occur around this axis, possibly due to natural arm adjustments or minor compensatory motions during the exercise. The Y-axis readings remain close to zero throughout the movement, indicating minimal or no rotation in that direction. This stability along the Y-axis supports the consistency and controlled nature of the system, confirming that the motion is primarily constrained to the sagittal plane.

Similar results were observed in knee joint extension and flexion, as shown in Figure 4.22. Like the elbow, the X-axis exhibits the most significant changes, reflecting the primary rotational movement of the knee joint in the sagittal plane, with minimal variations in the Y and Z axes.

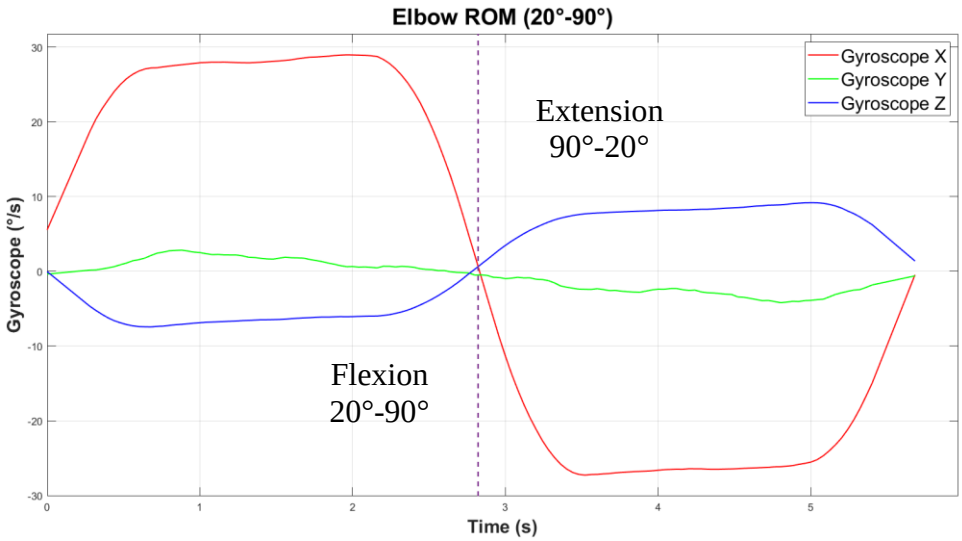


Figure 4.21: Gyroscope vs. Time for X, Y, and Z Axes During Elbow Flexion and Extension.

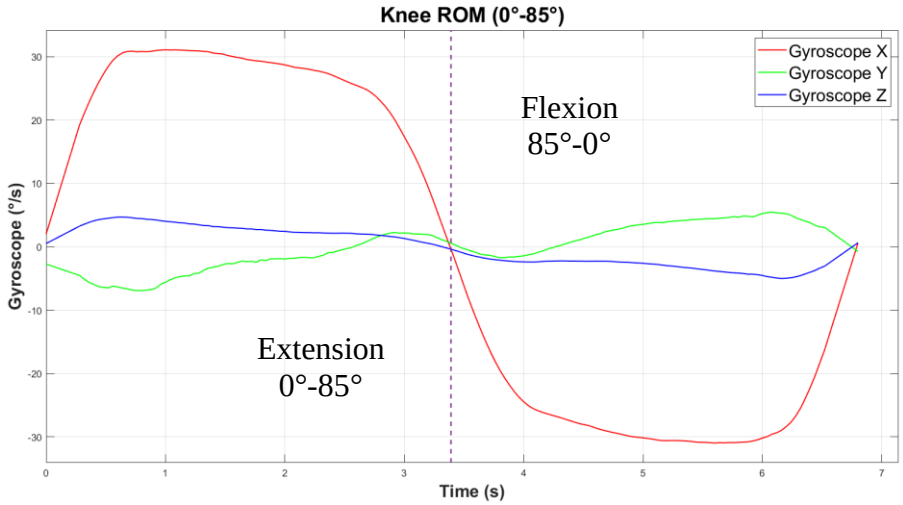


Figure 4.22: Gyroscope Reading vs. Time for X, Y, and Z Axes During Knee Flexion and Extension.

The Figure 4.23 illustrates the gyroscope X-axis readings across different rotation speeds. As anticipated, higher-speed rotations produce greater gyroscope X-axis values compared to lower-speed movements. Although both movements cover the same range of motion (ROM), the lower-speed motion requires a longer duration to complete a full cycle. Quantitatively, the high-speed motion is approximately twice as fast as the low-speed counterpart. Further details are provided in the Table 4.7, which presents the angular velocity of the X-axis under different motion conditions. Notably, flexion and extension demonstrate comparable peak angular velocities, despite one being assisted by gravity and the other performed against it. The percentage difference between these motions is minimal, at just 1.07%, indicating consistent system performance. A similar observation is made at lower speeds, with a percentage difference of 3.68%. These findings suggest that the system maintains a high

degree of stability and reliability in generating controlled rotational movements, regardless of speed or gravitational influence.

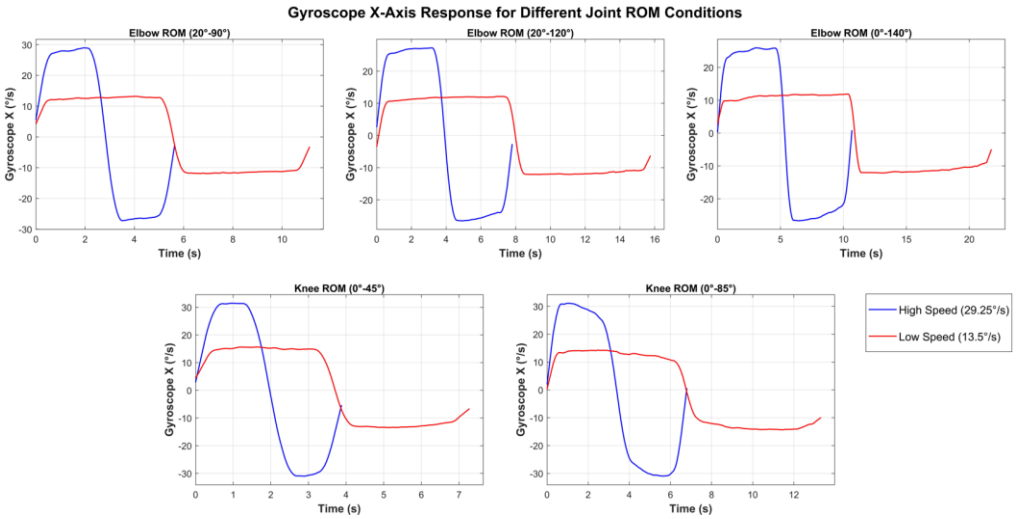


Figure 4.23: Gyroscope Readings vs. Time for X-Axis at Varying Speeds During Elbow and Knee Flexion and Extension.

Table 4.7: Comparison of Gyroscope X-Axis Peak Angular Velocities During Elbow and Knee Flexion and Extension at Different Speeds.

	Gyroscope X (°/s)							
	High Speed				**Low Speed**			
	Flexion Peak		**Extension Peak**		**Flexion Peak**		**Extension Peak**	
Mode	**Mean Peak**	**STD**	**Mean Peak**	**STD**	**Mean Peak**	**STD**	**Mean Peak**	**STD**
1	28.64	0.14	-27.16	0.06	13.04	0.10	-12.10	0.48
2	26.92	0.16	-26.49	0.08	12.01	0.07	-12.10	0.05
3	25.78	0.05	-26.70	0.03	11.73	0.05	-12.09	0.05
4	31.35	0.05	-31.13	0.08	15.41	0.21	-13.30	0.11
5	31.10	0.07	-30.78	0.18	14.21	0.10	-14.41	0.07
Average	28.76		-28.45		13.28		-12.80	
	Percentage Difference (%)		**1.07**		**Percentage Difference (%)**		**3.68**	

4.6 Safety considerations

The stationary upper and lower extremities rehabilitation system has been engineered with comprehensive safety considerations, covering electrical, mechanical, and software domains to ensure secure operation, especially in home-based settings. In the electrical system, a 48V 12.5A main power supply is used to drive both the stepper motor and the linear actuators. The stepper motor, requiring 48V and rated at 6A, results in a safety factor of approximately 2.08, confirming the power supply's adequacy. For the linear actuators, power is delivered through a 200W DC to DC buck converter rated for 8 to 60V input and 1 to 36V output at up to 15A. The actuators operate at 12V and 1A each, totaling 2A for both actuators. With the buck converter capable of delivering up to 15A at 12V, the electrical safety factor is 7.5, meaning the converter can handle over seven times the load required. Although the converter is rated for higher current than the power supply's 12.5A output, it only draws the amount of current needed, and thus no overloading occurs. This configuration not only ensures reliable actuator operation but also protects against overcurrent and thermal overload. All wiring is properly sealed with high-quality insulators to minimize the risk of shorts or contact hazards. A physical emergency stop button is provided to immediately disable all actuators and motors during malfunction or user distress, serving as a crucial safety feature.

To ensure robust mechanical safety, the calculated torque required to lift the limb and support structures is 21.78 Nm, representing the highest anticipated load. To meet this requirement, a stepper motor with a rating of 4.5 Nm is paired with a planetary gear featuring a 50:1 reduction ratio. This configuration produces an output torque of approximately 220 Nm, resulting in a mechanical safety factor of 10.1. This ensures the system can effectively handle unexpected loads, patient variability, or misalignments without risk of failure. The total weight of the upper assembly, including the motor, support structure, and control components, is approximately 9 kg, generating a combined load of around 88.3 N. Each linear actuator is capable of supporting up to 600 N, providing a combined load capacity of 1200 N. This gives a safety factor of 13.6, indicating that the actuators are operating well within their mechanical limits, even under dynamic or shifting loads. The high safety margin enhances system reliability and ensures safe operation during rehabilitation.

Software Safety Measures further enhance the reliability and safety of the system. A dedicated user interface displays real-time data and progress tracking, while an infrared remote control enables contactless control for added convenience and safety. To enhance user experience, the interface is designed to be intuitive and provide immediate feedback on limb movements, allowing users to track their rehabilitation progress. Data Privacy is a top priority for the system. Any sensitive information stored locally or in databases is encrypted to protect it from unauthorized access. User authentication is implemented to ensure that only authorized users can access their personal data, with role-based access control allowing restricted access to sensitive data based on user roles such as medical professionals and patients. In summary, the rehabilitation system incorporates a robust multi-layered safety approach. The electrical system includes safety factors of 2.08 for the motor and 7.5 for the actuators. Mechanically, the system offers safety factors of 10.1 for torque requirements and 13.6 for linear actuator support. These are further supported by intelligent software safeguards and an emergency stop mechanism, making the system safe, reliable, and well-suited for use in both home and clinical environments.

4.7 Sustainable Development Goals

The developed stationary upper and lower extremities rehabilitation system aligns with several United Nations Sustainable Development Goals (SDGs), particularly in the areas of health, innovation, and equality. Firstly, the system supports SDG 3: Good Health and Well-being by enabling targeted rehabilitation for users suffering from joint problems, whether due to physical injury or neurological conditions. Designed specifically for knee and elbow joints, the system facilitates controlled, repeatable therapy sessions that enhance recovery and improve functional mobility. Key features such as adjustable range of motion, real-time monitoring using accelerometer and gyroscope sensors, and a user interface for data logging help ensure effective rehabilitation outcomes. The inclusion of an emergency stop mechanism further enhances user safety during unsupervised or home-based use.

The project also contributes to SDG 9: Industry, Innovation and Infrastructure by demonstrating the integration of modern technologies such as the ESP32 microcontroller, motion sensors, and IR-based control systems into

a practical healthcare solution. Its design emphasizes adaptability and modularity, with customizable limb supports suitable for different limb weights and lengths, and an adjustable frame to accommodate various user postures. The mobile design using caster wheels enhances usability in home settings, especially for users who cannot easily access clinical rehabilitation services. In alignment with SDG 10: Reduced Inequalities, the system is developed as a low-cost solution (approximately RM1600) to increase accessibility for individuals in low-resource or rural environments. Its affordability and ease of use reduce barriers to rehabilitation, allowing more patients to undergo consistent therapy in the comfort of their own homes. This helps bridge the gap in healthcare access and supports inclusive rehabilitation practices.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

The development of a stationary upper and lower extremities rehabilitation system has been successfully achieved, resulting in a functional prototype capable of delivering controlled rehabilitation exercises. The system's design incorporates stepper motors for reliable speed control, crucial for consistent therapeutic movement. Validation of the motor's rotational speed showed an overall error of only 2.82% at high (29.25 °/s) and low (13.50 °/s) speeds. Range of motion assessments also confirmed the system's repeatability, with an overall percentage difference of 4.75% across different speeds. The integration of the MPU6050 sensor enables effective monitoring and interpretation of limb movements through acceleration and angular velocity measurements. Validation against a Trigno IMU revealed an average percentage error of 8.79%, indicating reasonable accuracy in joint motion tracking, though slight variations occurred depending on the specific movement and range. Overall, the system demonstrates significant potential to provide controlled, quantifiable, and monitored rehabilitation therapy for both upper and lower extremities. The validation of key components, including motor speed and sensor accuracy, supports the feasibility of this design for assisting rehabilitation.

Despite the promising results, several limitations were identified during the development and testing phases of the rehabilitation system. One significant limitation is the durability of 3D-printed components. The mechanical parts, printed with standard PLA material, showed signs of wear after approximately 100 hours of continuous operation, indicating that the prototype's lifespan may be limited, especially under long-term or high-frequency use. Additionally, while the system includes wireless data logging, it currently lacks a robust communication platform to facilitate efficient interaction between healthcare providers and users. This means that therapists cannot remotely monitor progress or adjust settings during home rehabilitation sessions, limiting the effectiveness of the system for continuous therapy. Another limitation is the limited sensor suite. While the system uses the MPU6050 sensor to monitor limb movements, the addition of more advanced

sensors, such as muscle sensors (EMG), would provide more detailed insights into muscle activity and fatigue, improving the precision of rehabilitation monitoring and therapy customization.

To address these limitations and enhance the system’s performance, several improvements are recommended. First, to improve durability, the 3D-printed parts should be replaced with more robust materials, such as aluminum or carbon fiber composites. This would significantly extend the system's lifespan and reliability, especially for long-term or high-frequency use. Additionally, to improve the communication between healthcare providers and users, it is recommended to develop a more integrated system that enables real-time data sharing and remote adjustments. A dedicated platform for therapists to monitor user progress, provide feedback, and modify settings would enhance the overall therapy process, ensuring more personalized and efficient treatment. Another important enhancement would be the integration of muscle sensors (EMG). These sensors would enable the system to track muscle effort, fatigue, and recovery, providing more detailed data that could inform real-time adjustments to the rehabilitation program. Lastly, to further enhance data access, the system should allow therapists to easily review and analyze user data, ensuring that treatment plans can be continually adapted based on user progress and specific needs.

REFERENCES

Aniket Fasate (2023) *'Mastering Motion Sensing: a Comprehensive Guide to the MPU6050 and MPU9250', Medium*. Medium. Available at: https://medium.com/@fasateaniket5/motion-sensors-mpu-6050-and-mpu9250-e86467e9a237 (Accessed: 12 May 2025).

Arduino (2024) *UNO WiFi Rev2, Arduino.cc*. Available at: https://docs.arduino.cc/hardware/uno-wifi-rev2/#tech-specs (Accessed: 12 May 2025).

Babiuch, M., Foltynek, P. and Smutny, P. (2019) 'Using the ESP32 Microcontroller for Data Processing', *2019 20th International Carpathian Control Conference (ICCC)* (pre-proof). Available at: https://doi.org/10.1109/carpathiancc.2019.8765944 (Accessed: 12 May 2025).

Barnes, M.P. and Good, D.C. (2025) *Neurological Rehabilitation*. 1st edn, *Ebsco.com*. 1st edn. Edinburgh: Elsevier, p. 970. Available at: https://publications.ebsco.com/public/rma-pfapi/v1/pf/external-link/a36df4f2-bbdf-4b2c-8902-dea5ba517918/0ed692e9-3ca1-44ef-9515-2c093cf74e70 (Accessed: 12 May 2025).

Bartels, M.N., Duffy, C.A. and Beland, H.E. (2016) 'Pathophysiology, Medical Management, and Acute Rehabilitation of Stroke Survivors', *Stroke Rehabilitation*, pp. 2–45. Available at: https://doi.org/10.1016/b978-0-323-17281-3.00001-0.

Bertani, R. *et al.* (2017) 'Effects of robot-assisted Upper Limb Rehabilitation in Stroke patients: a Systematic Review with meta-analysis', *Neurological Sciences*, 38(9), pp. 1561–1569. Available at: https://doi.org/10.1007/s10072-017-2995-5.

Boldea, I. and Nasar, S.A. (1999) 'Linear Electric Actuators and Generators', *IEEE Transactions on Energy Conversion*, 14(3), pp. 712–717. Available at: https://doi.org/10.1109/60.790940.

Casadei, K. and Kiel, J. (2023) *Anthropometric Measurement*, *PubMed*. Treasure Island (FL): StatPearls Publishing. Available at: https://www.ncbi.nlm.nih.gov/books/NBK537315.

Casillo, D. (2024) *Why Telescopic Lifting Columns Are the Perfect Choice for Your Linear Motion Application, Isotech, Inc*. Available at: https://www.isotechinc.com/telescoping-lifting-columns-for-linear-motion-application (Accessed: 12 May 2025).

CDC (2023) *Joint Range of Motion Study, Centers for Disease Control and Prevention*. Available at: https://archive.cdc.gov/www_cdc_gov/ncbddd/jointrom/index.html (Accessed: 12 May 2025).

Chong, Y.Z. *et al.* (2016) 'Development of low cost upper and lower extremities rehabilitation system with interactive feedback for children with movement disorders', *2016 IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES).* [Preprint]. Available at: https://doi.org/10.1109/iecbes.2016.7843556.

Cirstea, C.M., Ptito, A. and Levin, M.F. (2006) 'Feedback and Cognition in Arm Motor Skill Reacquisition After Stroke', *Stroke*, 37(5), pp. 1237–1242. Available at: https://doi.org/10.1161/01.str.0000217417.89347.63.

Cleveland Clinic (2022a) *Bone Fractures, Cleveland Clinic*. Cleveland Clinic. Available at: https://my.clevelandclinic.org/health/diseases/15241-bone-fractures (Accessed: 12 May 2025).

Cleveland Clinic (2022b) *Sarcopenia (Muscle Loss): Symptoms & Causes, Cleveland Clinic*. Available at: https://my.clevelandclinic.org/health/diseases/23167-sarcopenia (Accessed: 12 May 2025).

Dawal, S.Z.M. *et al.* (2015) 'Determination of the Significant Anthropometry Dimensions for user-friendly Designs of Domestic Furniture and Appliances – Experience from a Study in Malaysia', *Measurement*, 59, pp. 205–215. Available at: https://doi.org/10.1016/j.measurement.2014.09.030.

de Leva, P. (1996) 'Adjustments to Zatsiorsky-Seluyanov's Segment Inertia Parameters', *Journal of Biomechanics*, 29(9), pp. 1223–30. Available at: https://doi.org/10.1016/0021-9290(95)00178-6.

Dietrich, S. (2022) *Servo Motor vs Stepper Motor: Understanding the Differences, Control.com*. Available at: https://control.com/technical-articles/servo-motor-vs-stepper-motor-understanding-the-differences (Accessed: 12 May 2025).

Garrick, N. (2023) *Osteoarthritis, National Institute of Arthritis and Musculoskeletal and Skin Diseases*. Available at: https://www.niams.nih.gov/health-topics/osteoarthritis/diagnosis-treatment-and-steps-to-take (Accessed: 12 May 2025).

GlobalSpec (2025) *'linear actuators' | GlobalSpec, Globalspec.com.* Available at: https://www.globalspec.com/search/products?term=linear+actuators (Accessed: 12 May 2025).

Gunasekaran, B. (2019) *IMU Sensors: Everything You Need to Know!, Embedded Inventor*. Available at: https://embeddedinventor.com/what-is-an-imu-sensor-a-complete-guide-for-beginners (Accessed: 12 May 2025).

Hanifa Bouziri *et al.* (2023) 'Temporal and Spatial Distribution of Musculoskeletal Disorders from 1990 to 2019: a Systematic Analysis of the Global Burden of Disease', *BMJ Public Health*, 1(1), pp. e000353–e000353. Available at: https://doi.org/10.1136/bmjph-2023-000353.

Hatem, S.M. *et al.* (2016) 'Rehabilitation of Motor Function after Stroke: a Multiple Systematic Review Focused on Techniques to Stimulate Upper Extremity Recovery', *Frontiers in Human Neuroscience*, 10(442). Available at: https://doi.org/10.3389/fnhum.2016.00442.

Helen (2019) *DC Motor Vs Stepper Motor Vs Servo Motor - Which to choose?*, *Latest News from Seeed Studio*. Available at: https://www.seeedstudio.com/blog/2019/04/01/choosing-the-right-motor-for-your-project-dc-vs-stepper-vs-servo-motors (Accessed: 12 May 2025).

Huisstede, B.M. *et al.* (2006) 'Incidence and Prevalence of upper-extremity Musculoskeletal disorders. a Systematic Appraisal of the Literature', *BMC Musculoskeletal Disorders*, 7(1). Available at: https://doi.org/10.1186/1471-2474-7-7.

Iezzoni, L.I. (2010) *Multiple Sclerosis*. Bloomsbury Publishing USA.

Jaworski, Ł., Karpiński, R. and Dobrowolska, A. (2016) 'Biomechanics of the Upper Limb.', *Journal of Technology and Exploitation in Mechanical Engineering*, 2(1), pp. 56–59. Available at: https://doi.org/10.35784/jteme.517.

Joseph, J. (2022) *How Does the MPU6050 Accelerometer & Gyroscope Sensor Work and Interfacing It with Arduino*, *circuitdigest.com*. Available at: https://circuitdigest.com/microcontroller-projects/interfacing-mpu6050-module-with-arduino (Accessed: 12 May 2025).

Kamavuako, E.N. (2022) 'On the Applications of EMG Sensors and Signals', *Sensors*, 22(20), p. 7966. Available at: https://doi.org/10.3390/s22207966.

Kang, S. *et al.* (2022) 'Burden of Neurological Diseases in Asia from 1990 to 2019: a Systematic Analysis Using the Global Burden of Disease Study Data', *BMJ Open*, 12(9), e059548. Available at: https://doi.org/10.1136/bmjopen-2021-059548.

Li, B. *et al.* (2017) 'A Review of Rehabilitation Robot', *32nd Youth Academic Annual Conference of Chinese Association of Automation (YAC), Hefei, China*, pp. 907–911. Available at: https://doi.org/10.1109/yac.2017.7967538.

Luks, H. (2021) *10 Recommendations to Thrive with Osteoarthritis of the Knee — Physio Network, Physio Network — Physiotherapy CPD Courses Online*. Available at: https://www.physio-network.com/blog/10-recommendations-to-thrive-with-osteoarthritis-of-the-knee (Accessed: 12 May 2025).

Mang, C.S. and Peters, S. (2021) 'Advancing Motor Rehabilitation for Adults with Chronic Neurological Conditions through Increased Involvement of kinesiologists: a Perspective Review', *BMC Sports Science, Medicine and Rehabilitation*, 13(1). Available at: https://doi.org/10.1186/s13102-021-00361-6.

Mayo Clinic (2021) *Osteoarthritis - Symptoms and Causes, Mayo Clinic*. Available at: https://www.mayoclinic.org/diseases-conditions/osteoarthritis/symptoms-causes/syc-20351925 (Accessed: 5 May 2025).

Mayo Clinic (2023) *Rheumatoid Arthritis - Symptoms and Causes, Mayo Clinic*. Available at: https://www.mayoclinic.org/diseases-conditions/rheumatoid-arthritis/symptoms-causes/syc-20353648 (Accessed: 5 May 2025).

Mohamad, D. *et al.* (2010) Development of a Malaysian Anthropometric Database, ResearchGate. World Engineering Congress 2010, 2nd – 5th August 2010, Kuching, Sarawak, Malaysia Conference on Manufacturing Technology and Management. Available at: https://www.researchgate.net/publication/288877992_Development_of_a_Malaysian_anthropometric_database (Accessed: 5 May 2025).

Monk, A.P. *et al.* (2016) 'Biomechanics of the Lower Limb', *Surgery (Oxford)*, 34(9), pp. 427–435. Available at: https://doi.org/10.1016/j.mpsur.2016.06.007.

Page, S.J., Levine, P. and Leonard, A.C. (2005) 'Effects of Mental Practice on Affected Limb Use and Function in Chronic Stroke', *Archives of Physical Medicine and Rehabilitation*, 86(3), pp. 399–402. Available at: https://doi.org/10.1016/j.apmr.2004.10.002.

Ross, J.A. and Singer, S.S. (2024) 'Parkinson's disease.', *Salem Press Encyclopedia of Health* [Preprint]. Available at: https://research-ebsco-com.libezp2.utar.edu.my/c/qdh7q6/viewer/html/n4taaa5pb5 (Accessed: 13 May 2025).

Saikaley Bsc, M. *et al.* (no date) *Chapter 10 UPPER EXTREMITY MOTOR REHABILITATION INTERVENTIONS*. Available at: http://www.ebrsr.com/sites/default/files/Ch.%2010%20Upper%20Extremity%20Motor%20Interventions_v20.pdf (Accessed: 13 May 2025).

Sarvenaz, M. *et al.* (no date) *Lower Extremity Interventions | EBRSR - Evidence-Based Review of Stroke Rehabilitation*, *www.ebrsr.com*. Available at: http://www.ebrsr.com/evidence-review/9-lower-extremity-interventions.

Tan, C. (2021) *Esp32 Vs Arduino: the Differences Simply Explained, All3DP*. Available at: https://all3dp.com/2/esp32-vs-arduino-differences/ (Accessed: 12 May 2025).

Tripathi, J. and Kumar, B. (2023) 'Machine Learning Assisted MPU6050-Based Road Anomaly Detection', *Lecture Notes in Electrical Engineering*, pp. 419–433. Available at: https://doi.org/10.1007/978-981-99-0189-0_32.

Vos, P. and Diaz-Arrastia, R. (2014) *Traumatic Brain Injury*. John Wiley & Sons.

WEEBLY (2024) *Rheumatoid Arthritis, Differential Diagnosis of the Knee*. Available at: https://differentialdiagnosisknee.weebly.com/rheumatoid-arthritis.html (Accessed: 12 May 2025).

World Health Organization (2024) *Rehabilitation, World Health Organisation*. World Health Organization: WHO. Available at: https://www.who.int/news-room/fact-sheets/detail/rehabilitation (Accessed: 13 May 2025).

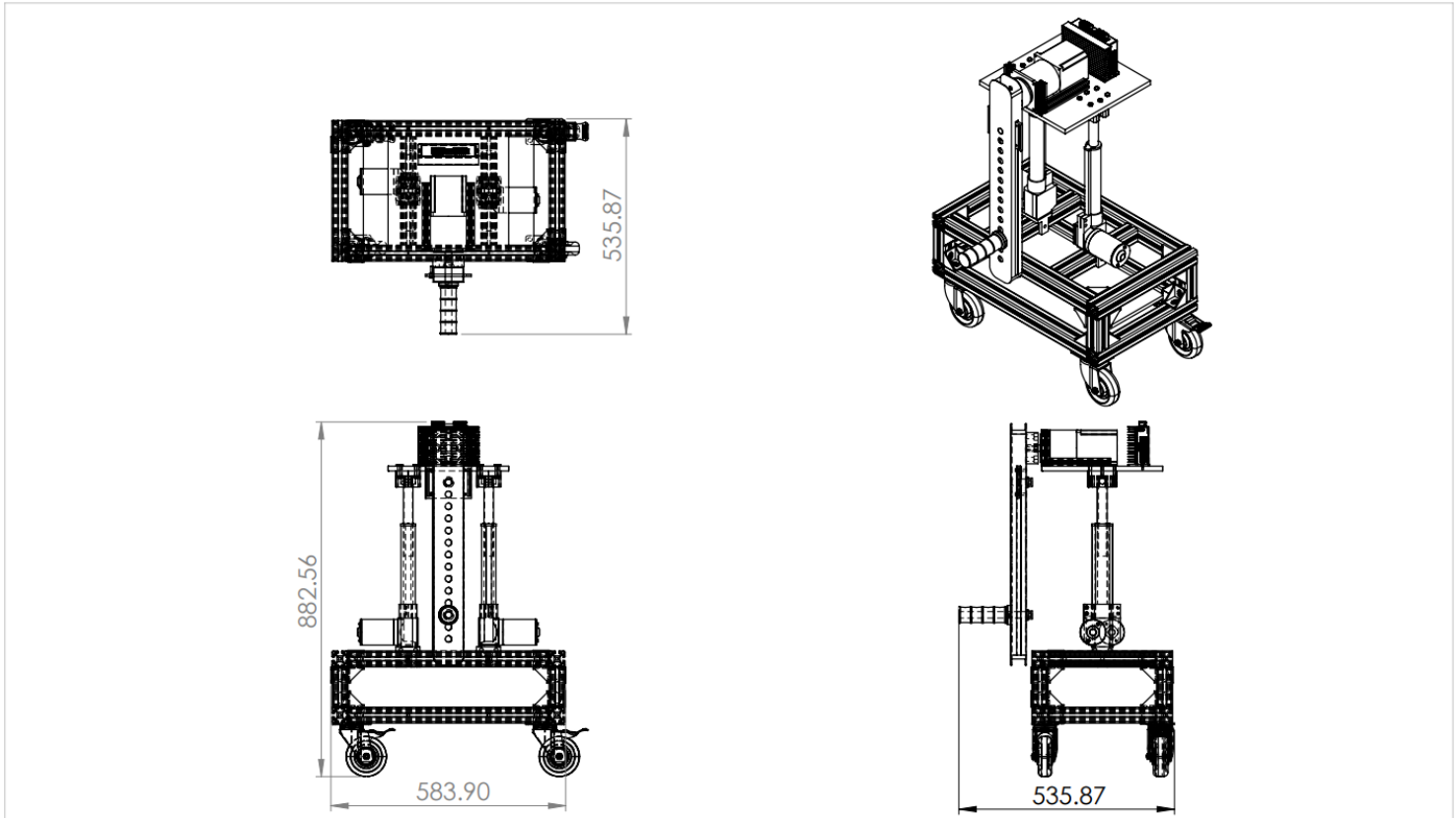
Ye-Lin, Y. *et al.* (2022) 'Directed Functional Coordination Analysis of Swallowing Muscles in Healthy and Dysphagic Subjects by Surface Electromyography', *Sensors*, 22(12), p. 4513. Available at: https://doi.org/10.3390/s22124513.

Zhao, H. (2015) *EMG Sensors | Muscle Activation Devices, Arduino EMG Sensors, PCB Assembly, PCB Manufacturing, PCB Design - OURPCB*. Available at: https://www.ourpcb.com/emg-sensors.html. (Accessed: 12 May 2025).

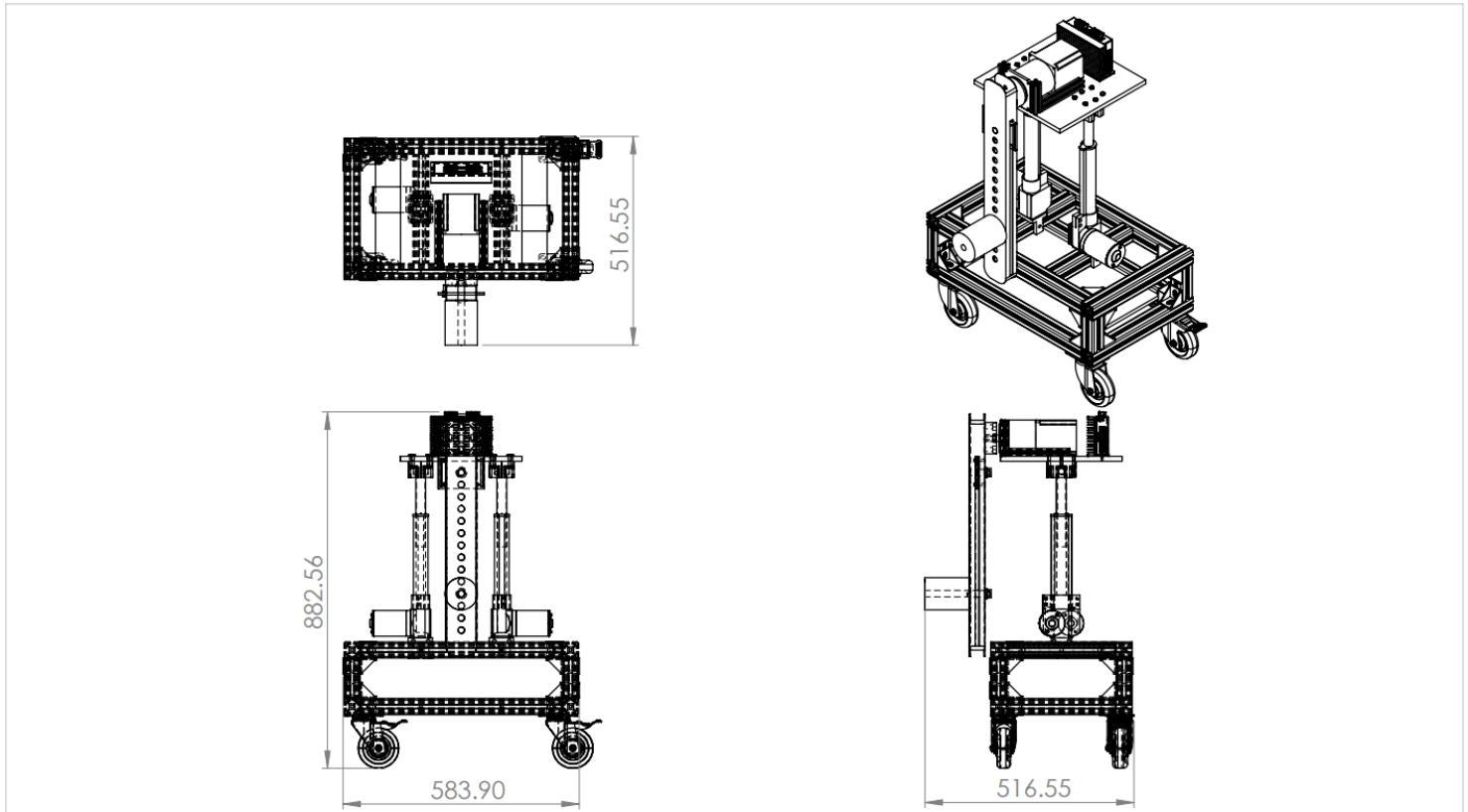
APPENDICES

Appendix A: Engineering Drawing.

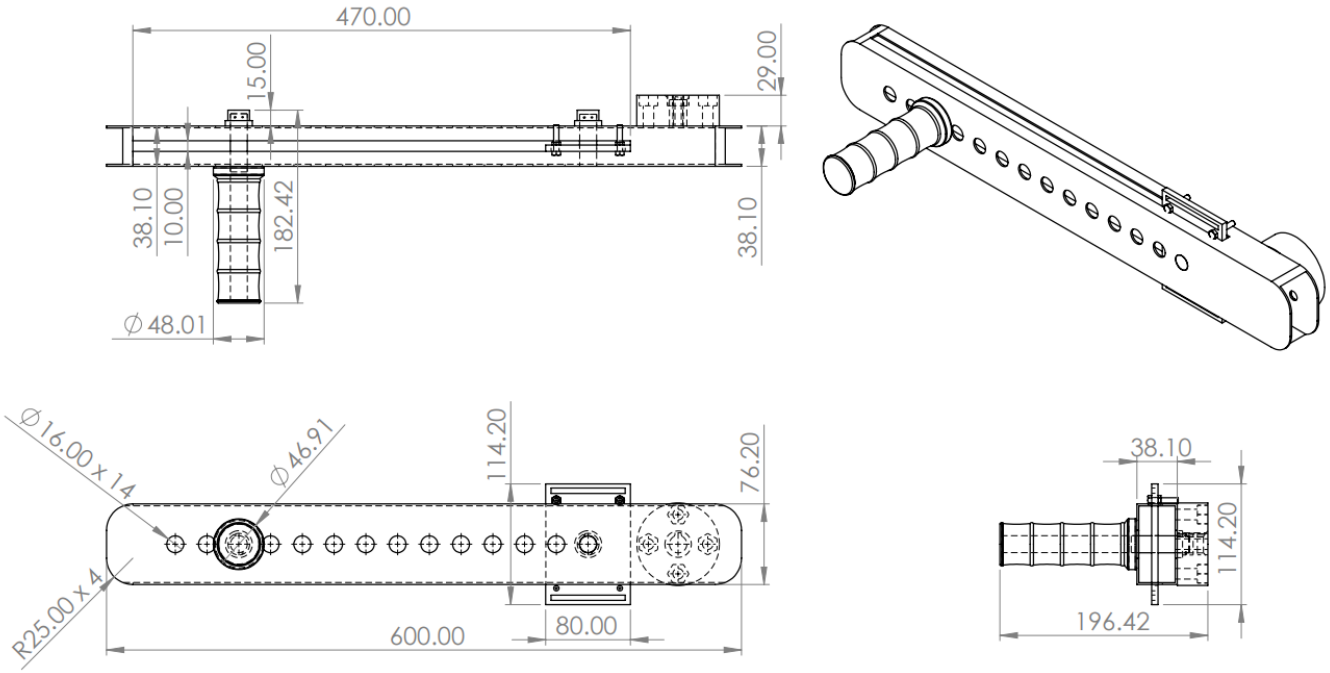
Drawing No.	Engineering Drawing
1.	Full Assembly (Forearm)
2.	Full Assembly (Shank)
3.	Forearm Support Structure
4.	Shank Support Structure
5.	Hand Grip
6.	Footrest Cushion
7.	Support Structure
8.	Stopper
9.	3D-printed Coupler
10.	Velcro Secure Plate
11.	Connector
12.	Base Connector
13.	Mild Steel Platform



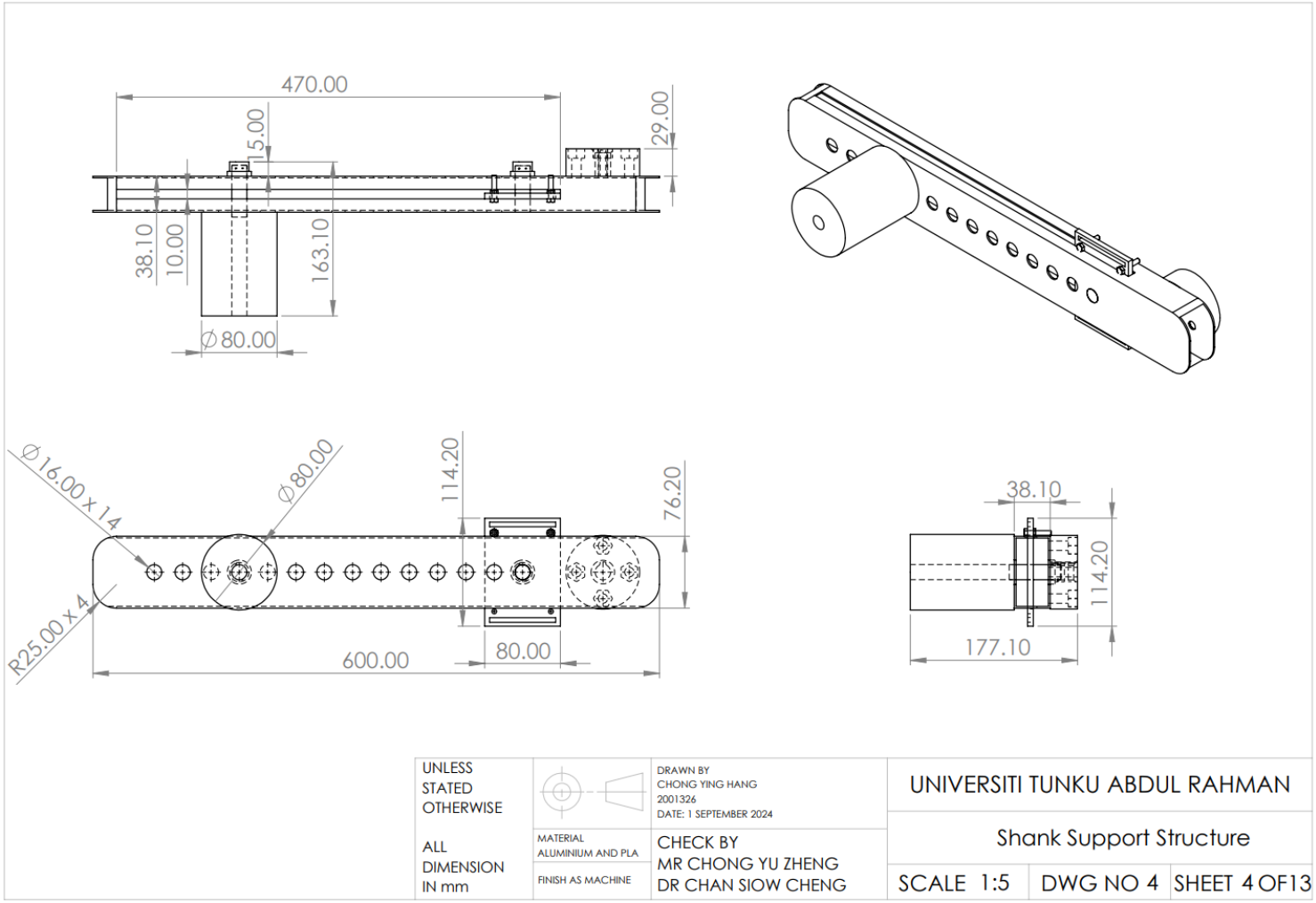
UNLESS STATED OTHERWISE ALL DIMENSION IN mm		DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024	UNIVERSITI TUNKU ABDUL RAHMAN
	MATERIAL ALUMINIUM AND PLA	CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	Full Assembly (Forearm)
	FINISH AS MACHINE		SCALE 1:13 | DWG NO 1 | SHEET 1 OF13

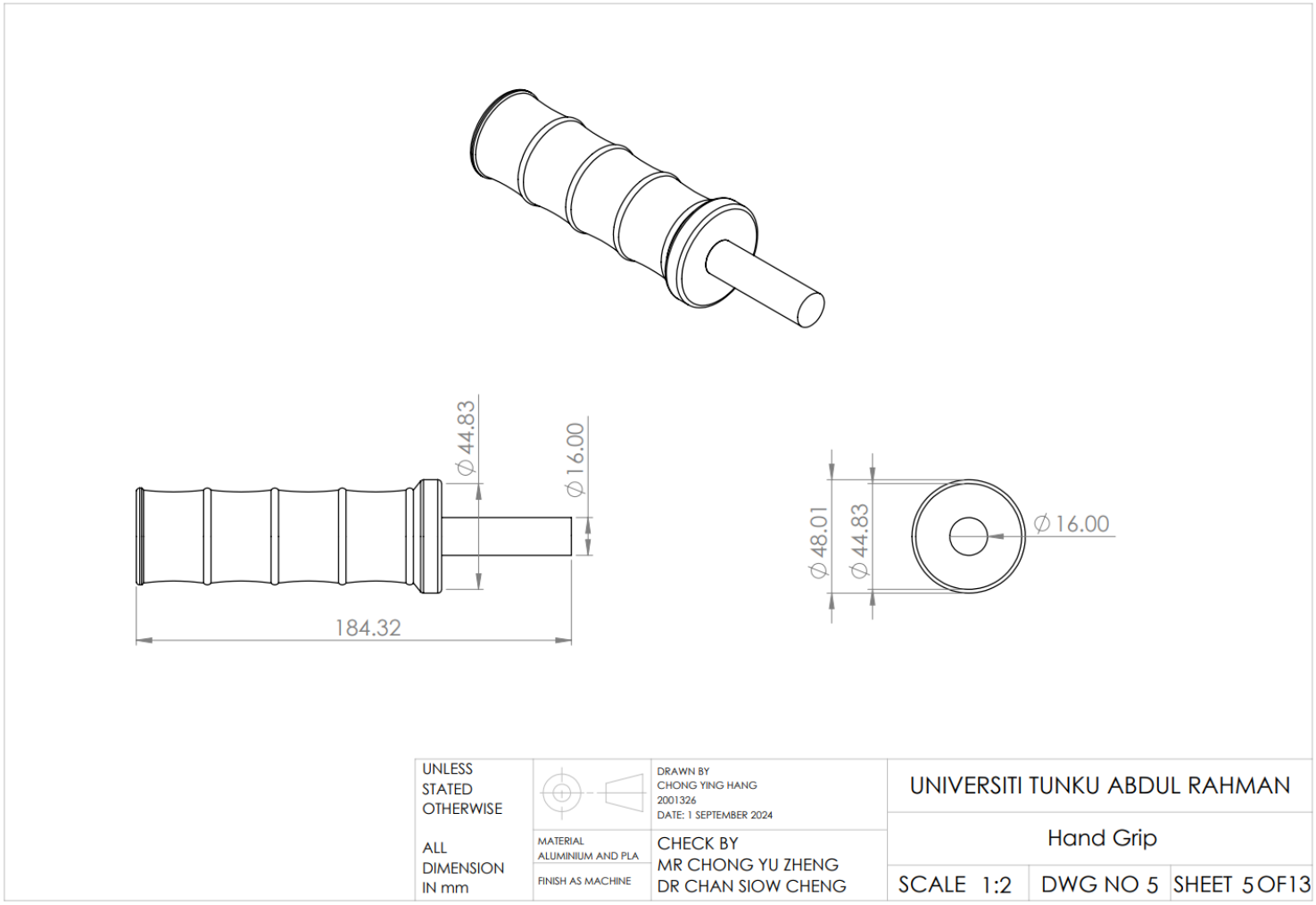


UNLESS STATED OTHERWISE ALL DIMENSION IN mm	MATERIAL ALUMINIUM AND PLA FINISH AS MACHINE	DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024 CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	UNIVERSITI TUNKU ABDUL RAHMAN Full Assembly (Shank) SCALE 1:13 | DWG NO 2 | SHEET 2 OF 13

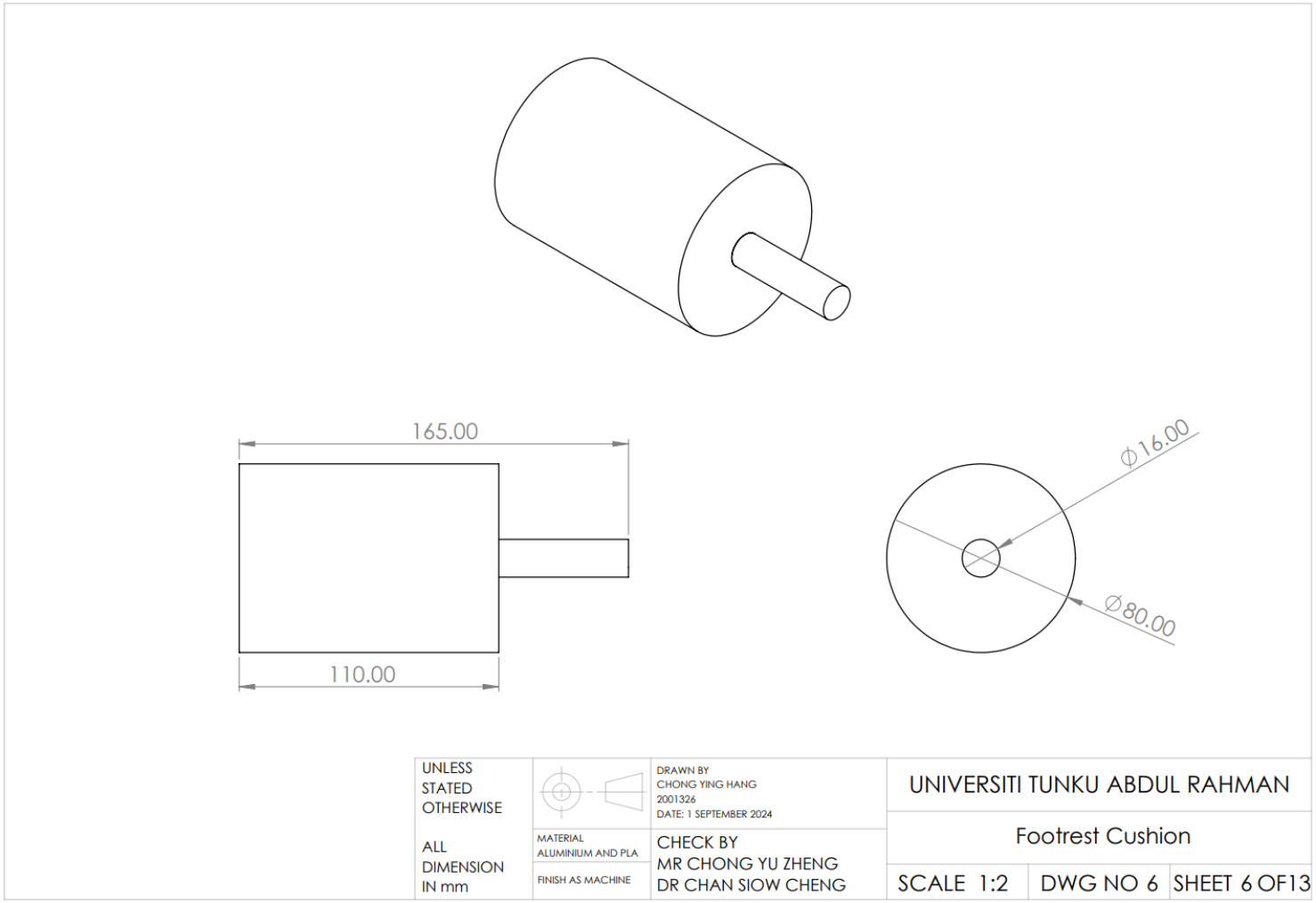


UNLESS STATED OTHERWISE ALL DIMENSION IN mm		DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024	UNIVERSITI TUNKU ABDUL RAHMAN
	MATERIAL ALUMINIUM AND PLA FINISH AS MACHINE	CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	Forearm Support Structure
			SCALE 1:5 | DWG NO 3 | SHEET 3 OF13

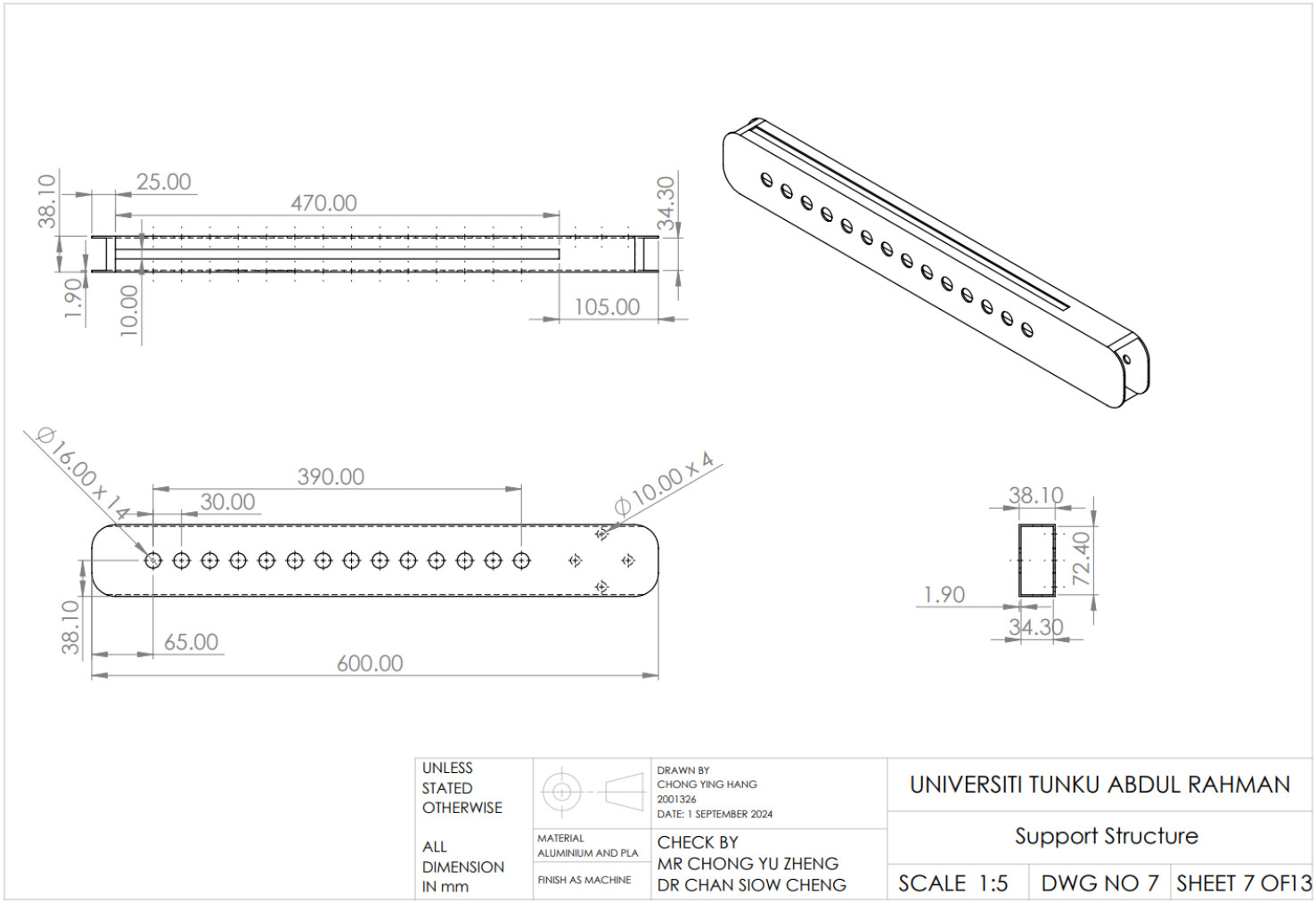

470.00
15.00
29.00
38.10
10.00
163.10
∅80.00
∅16.00 x 14
∅80.00
114.20
76.20
R25.00 x 4
600.00
80.00
38.10
114.20
177.10
UNLESS STATED OTHERWISE
ALL DIMENSION IN mm
MATERIAL ALUMINIUM AND PLA
FINISH AS MACHINE
DRAWN BY
CHONG YING HANG
2001326
DATE: 1 SEPTEMBER 2024
CHECK BY
MR CHONG YU ZHENG
DR CHAN SIOW CHENG
UNIVERSITI TUNKU ABDUL RAHMAN
Shank Support Structure
SCALE 1:5
DWG NO 4
SHEET 4 OF13

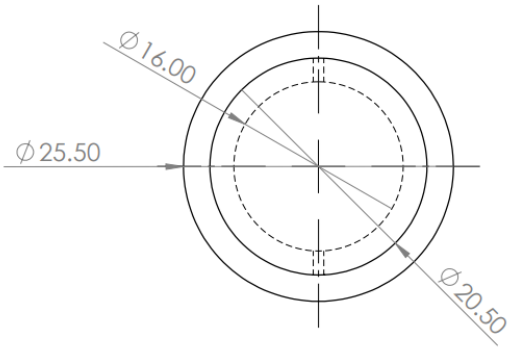


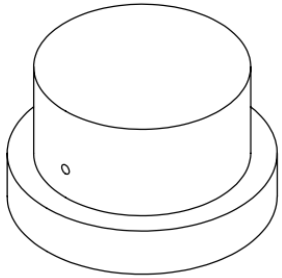
UNLESS STATED OTHERWISE		DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024	UNIVERSITI TUNKU ABDUL RAHMAN
ALL DIMENSION IN mm	MATERIAL ALUMINIUM AND PLA FINISH AS MACHINE	CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	Hand Grip
			SCALE 1:2 | DWG NO 5 | SHEET 5 OF 13

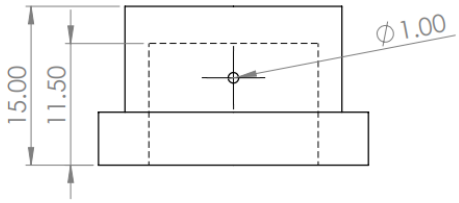


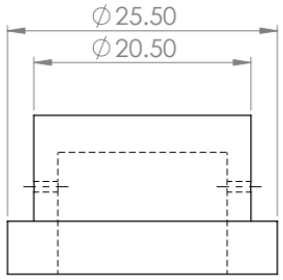
UNLESS STATED OTHERWISE ALL DIMENSION IN mm		DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024	UNIVERSITI TUNKU ABDUL RAHMAN
	MATERIAL ALUMINIUM AND PLA	CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	Footrest Cushion
	FINISH AS MACHINE		SCALE 1:2 | DWG NO 6 | SHEET 6 OF13


38.10
25.00
470.00
34.30
1.90
10.00
105.00
Ø16.00 x 14
390.00
30.00
Ø10.00 x 4
38.10
65.00
600.00
38.10
72.40
1.90
34.30
UNLESS STATED OTHERWISE
ALL DIMENSION IN mm
MATERIAL ALUMINIUM AND PLA
FINISH AS MACHINE
DRAWN BY
CHONG YING HANG
2001326
DATE: 1 SEPTEMBER 2024
CHECK BY
MR CHONG YU ZHENG
DR CHAN SIOW CHENG
UNIVERSITI TUNKU ABDUL RAHMAN
Support Structure
SCALE 1:5
DWG NO 7
SHEET 7 OF13

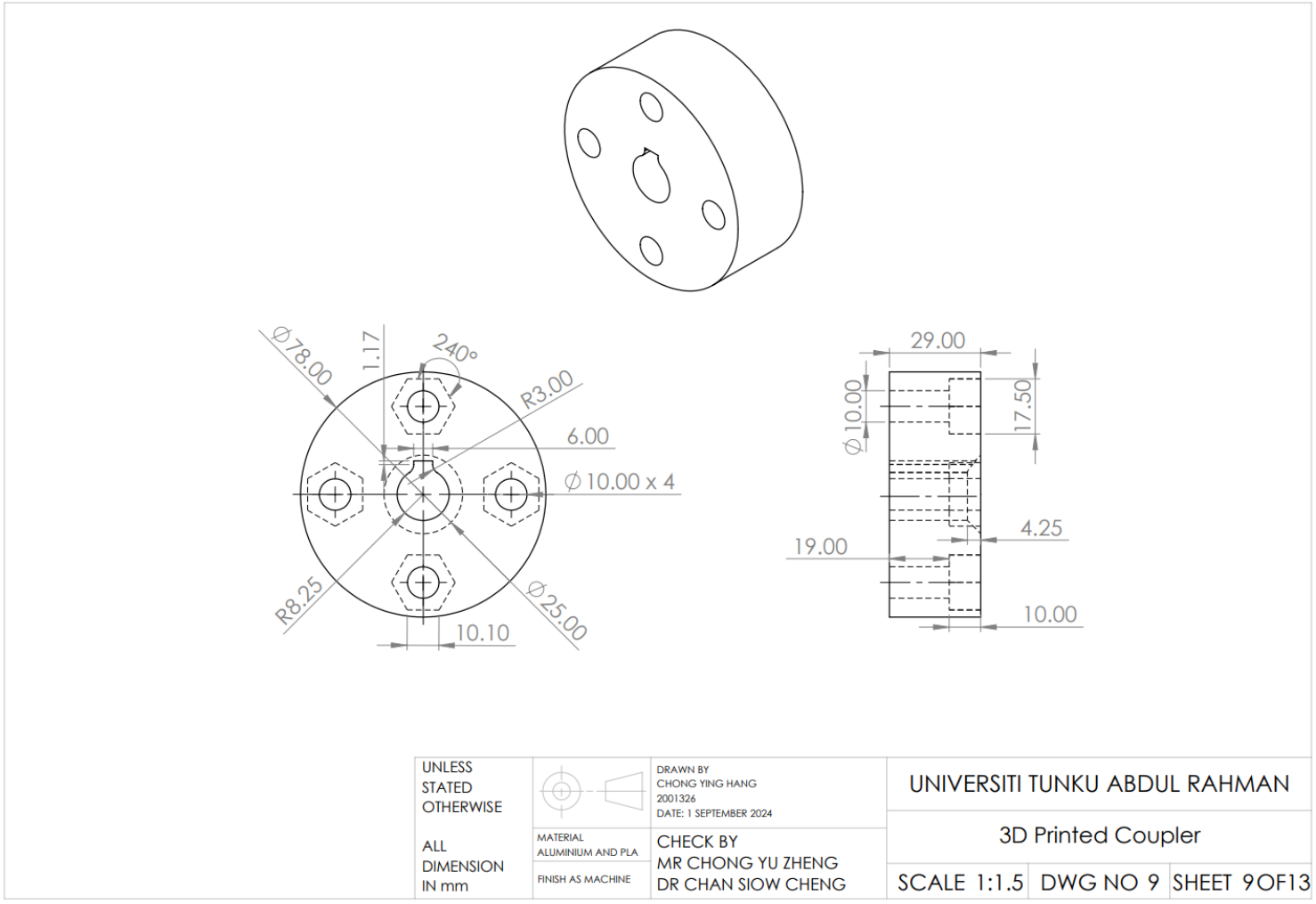




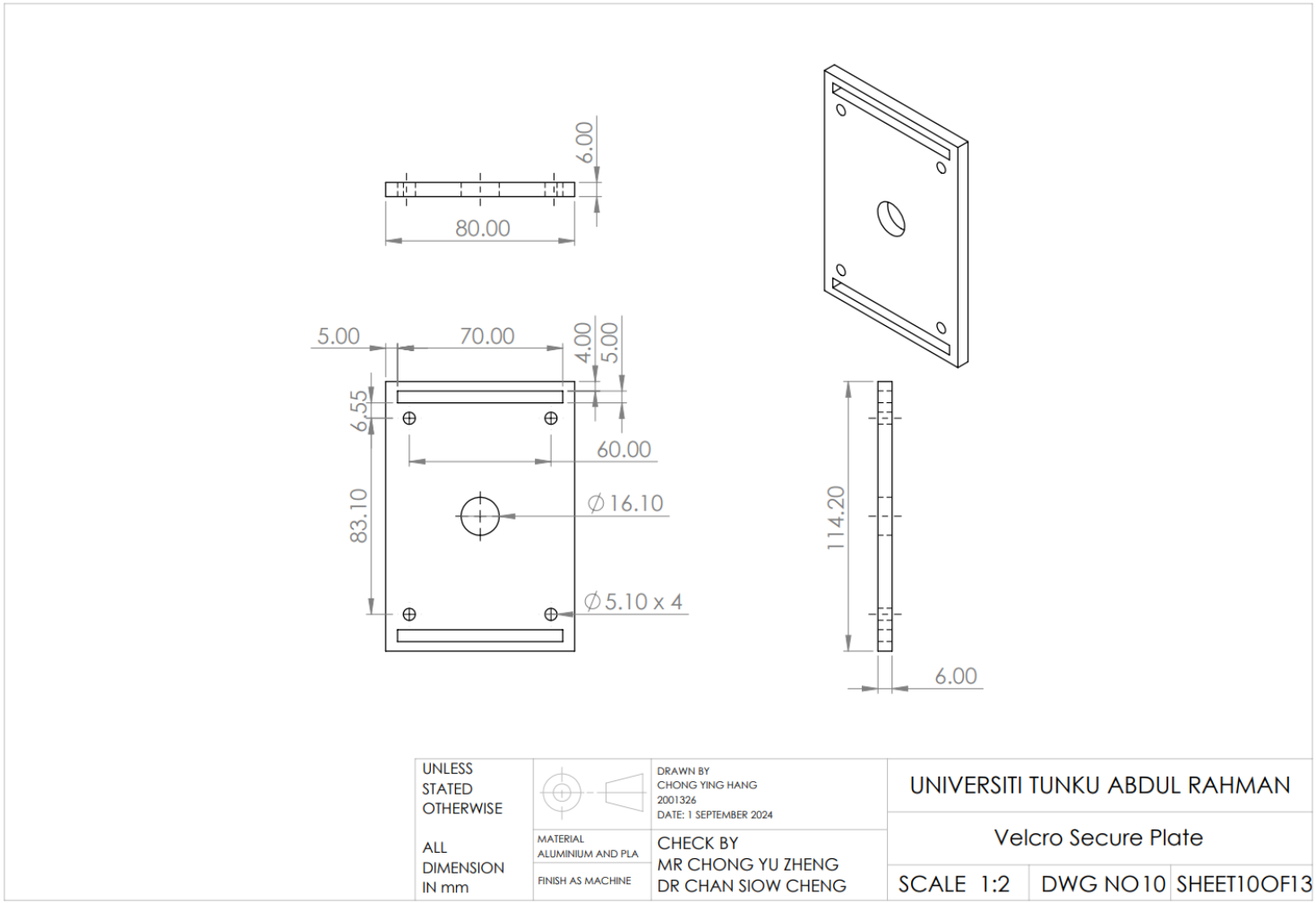


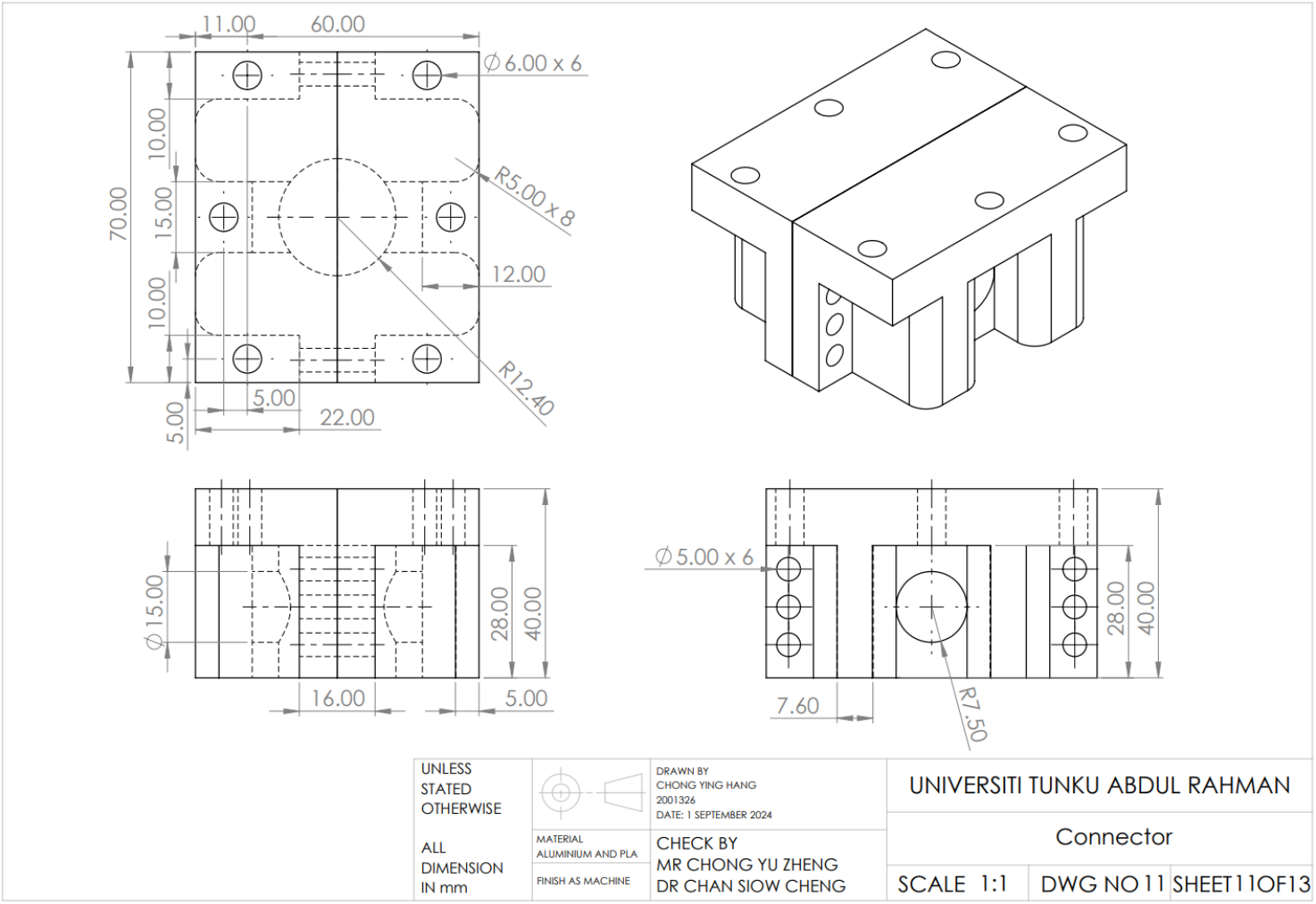


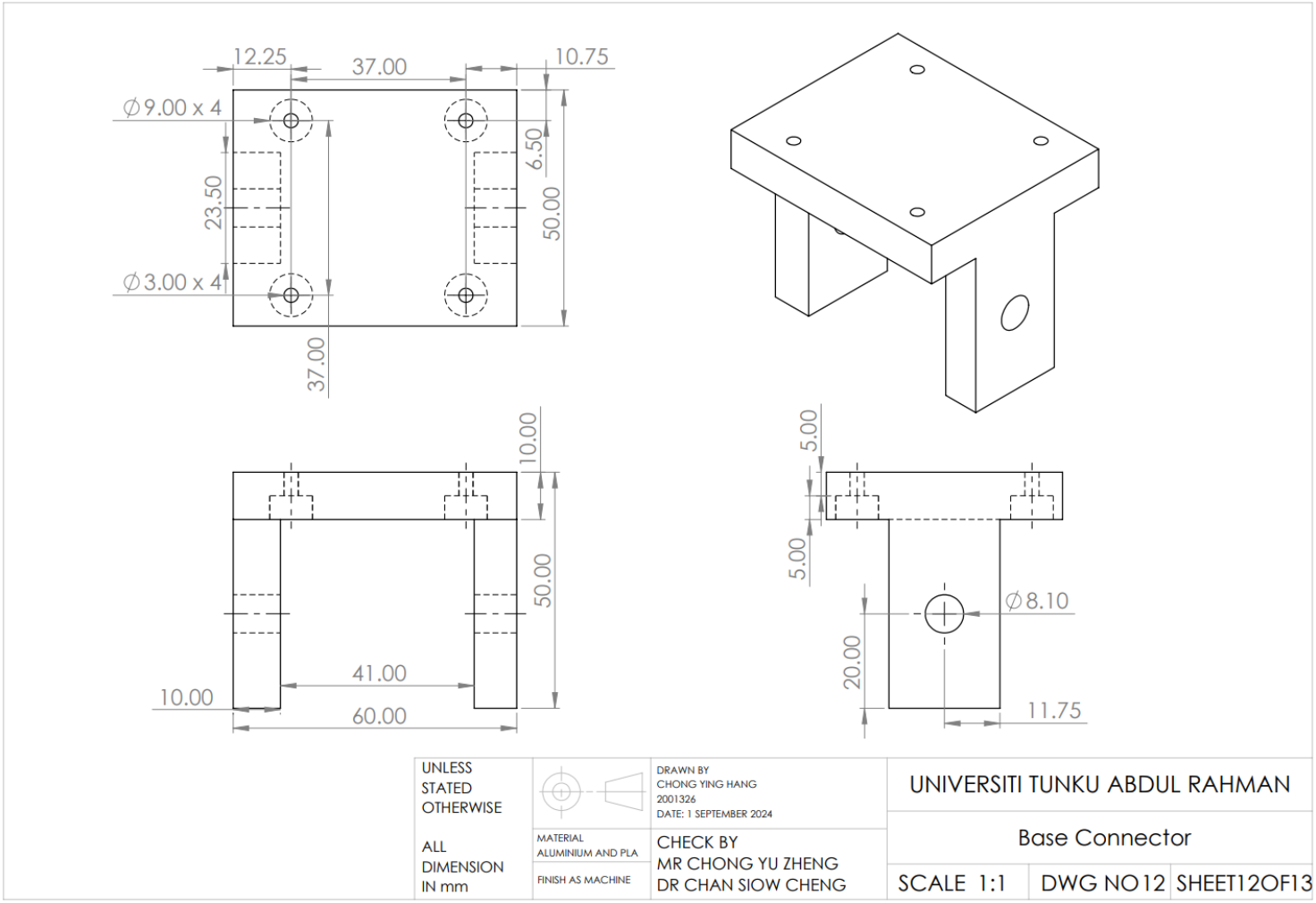
UNLESS STATED OTHERWISE ALL DIMENSION IN mm		DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024	UNIVERSITI TUNKU ABDUL RAHMAN
	MATERIAL ALUMINIUM AND PLA	CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	Stopper
	FINISH AS MACHINE		SCALE 2:1 | DWG NO 8 | SHEET 8 OF13



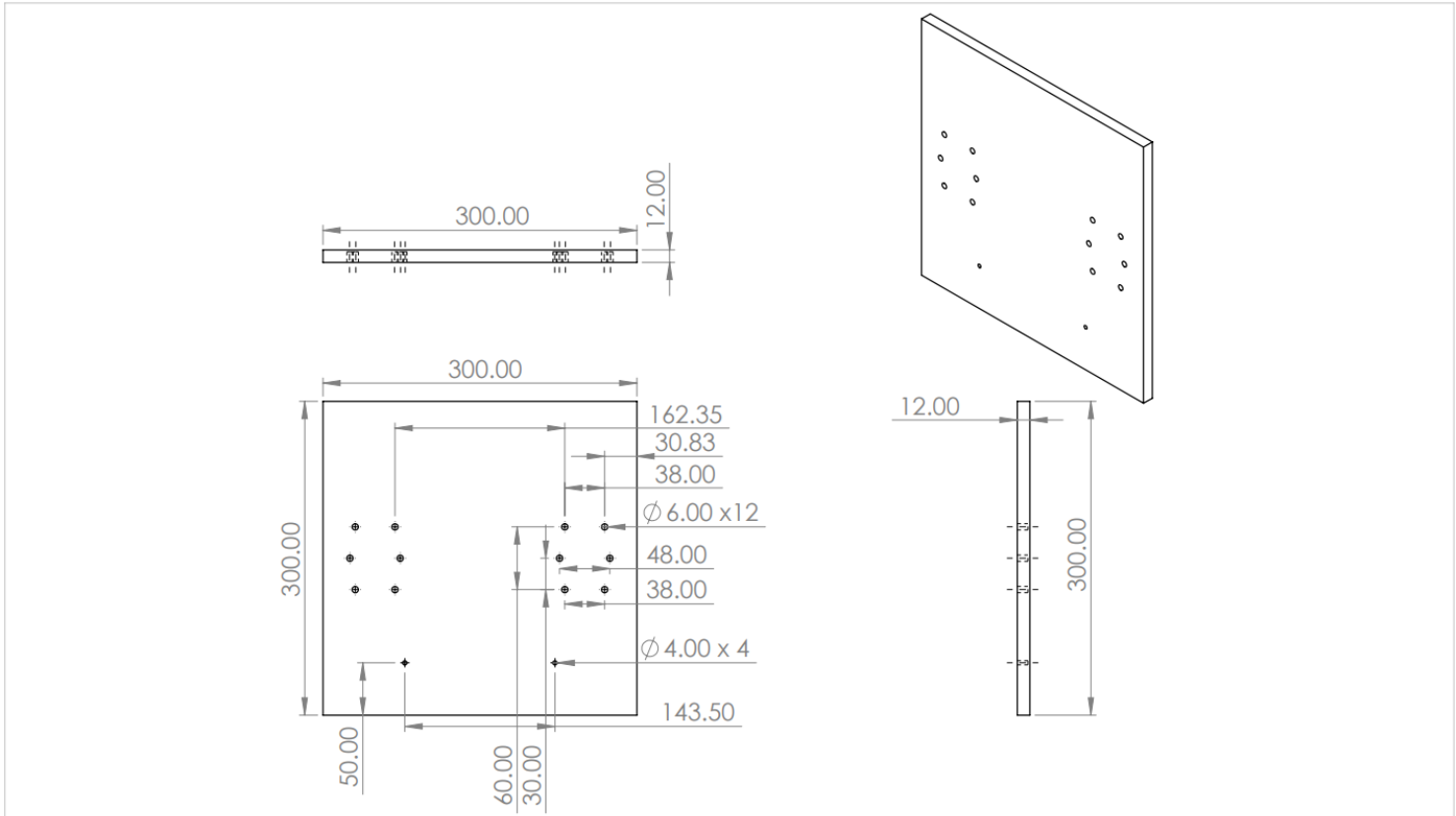
UNLESS STATED OTHERWISE ALL DIMENSION IN mm	MATERIAL ALUMINIUM AND PLA FINISH AS MACHINE	DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024 CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	UNIVERSITI TUNKU ABDUL RAHMAN 3D Printed Coupler SCALE 1:1.5 | DWG NO 9 | SHEET 9OF13


6.00
80.00
5.00
70.00
4.00
5.00
6.55
60.00
83.10
∅16.10
∅5.10 x 4
114.20
6.00
UNLESS STATED OTHERWISE
ALL DIMENSION IN mm
MATERIAL ALUMINIUM AND PLA
FINISH AS MACHINE
DRAWN BY
CHONG YING HANG
2001326
DATE: 1 SEPTEMBER 2024
CHECK BY
MR CHONG YU ZHENG
DR CHAN SIOW CHENG
UNIVERSITI TUNKU ABDUL RAHMAN
Velcro Secure Plate
SCALE 1:2
DWG NO10
SHEET10OF13


11.00
60.00
∅6.00 x 6
10.00
15.00
10.00
70.00
R5.00 x 8
12.00
R12.40
5.00
5.00
22.00
∅15.00
28.00
40.00
16.00
5.00
∅5.00 x 6
28.00
40.00
7.60
R7.50
UNLESS STATED OTHERWISE
ALL DIMENSION IN mm
DRAWN BY
CHONG YING HANG
2001326
DATE: 1 SEPTEMBER 2024
MATERIAL
ALUMINIUM AND PLA
FINISH AS MACHINE
CHECK BY
MR CHONG YU ZHENG
DR CHAN SIOW CHENG
UNIVERSITI TUNKU ABDUL RAHMAN
Connector
SCALE 1:1
DWG NO 11
SHEET11OF13



UNLESS STATED OTHERWISE ALL DIMENSION IN mm		DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024	UNIVERSITI TUNKU ABDUL RAHMAN
	MATERIAL ALUMINIUM AND PLA	CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	Base Connector
	FINISH AS MACHINE		SCALE 1:1 | DWG NO12 | SHEET12OF13



UNLESS STATED OTHERWISE ALL DIMENSION IN mm		DRAWN BY CHONG YING HANG 2001326 DATE: 1 SEPTEMBER 2024	UNIVERSITI TUNKU ABDUL RAHMAN
	MATERIAL ALUMINIUM AND PLA	CHECK BY MR CHONG YU ZHENG DR CHAN SIOW CHENG	Mild Steel Platform
	FINISH AS MACHINE		SCALE 1:5 | DWG NO13 | SHEET13OF13

Appendix B: Arduino Code for Sensor Data Collection and Firebase Data Storage.

```
////////////////////////////////////////////////////////////
// Libraries
////////////////////////////////////////////////////////////
#include <Wire.h>
#include <I2Cdev.h>
#include <MPU6050_light.h>
#include <WiFi.h>
#include <Firebase_ESP_Client.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include "addons/TokenHelper.h"
#include "addons/RTDBHelper.h"

////////////////////////////////////////////////////////////
// Wi-Fi & Firebase Credentials
////////////////////////////////////////////////////////////
#define WIFI_SSID "YH iPhone"
#define WIFI_PASSWORD "chong020318"
#define API_KEY "AIzaSyBDPIMOboyU-wWr1w9WBnkVoq3uVJw3ky8"
#define DATABASE_URL "https://medical-74468-default-rtdb.asia-southeast1.firebasedatabase.app/"
#define PROJECT_ID "medical-74468"
#define USER_EMAIL "yinghang020318@gmail.com"
#define USER_PASSWORD "chong020318"

////////////////////////////////////////////////////////////
// Global Variables
////////////////////////////////////////////////////////////
FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", 8 * 3600, 60000);  // GMT+8

MPU6050 mpu(Wire);

String userid = "";
bool dataRetrieved = false;
bool newSession = true;
unsigned long lastSendMillis = 0;
unsigned long lastTimestamp = 0;
String currentSession = "";
////////////////////////////////////////////////////////////
// Setup
////////////////////////////////////////////////////////////
void setup() {
  Serial.begin(115200);
  delay(500);

  // Connect to Wi-Fi
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(300);
  }
  Serial.println("\nConnected with IP: " + WiFi.localIP().toString());

  // Firebase config
  config.api_key = API_KEY;
  config.database_url = DATABASE_URL;
  auth.user.email = USER_EMAIL;
  auth.user.password = USER_PASSWORD;
  config.token_status_callback = tokenStatusCallback;

  Firebase.begin(&config, &auth);
  Firebase.reconnectWiFi(true);

  // Time Client
  timeClient.begin();
  timeClient.update();
```

```
  // MPU6050
  Wire.begin();
  byte status = mpu.begin();
  Serial.print(F("MPU6050 status: "));
  Serial.println(status);
  while (status != 0) { }
  Serial.println(F("Calculating offsets, do not move MPU6050"));
  mpu.calcOffsets();
  Serial.println("Offset Done!\n");

  // Get user ID from Firestore
  getUserID();
}
////////////////////////////////////////////////////////////
// Get User ID from Firestore
////////////////////////////////////////////////////////////
void getUserID() {
  if (!Firebase.ready()) return;

  String path = "DocumentID/updatecurrentID";
  if (Firebase.Firestore.getDocument(&fbdo, PROJECT_ID, "", path.c_str())) {
    FirebaseJson json;
    json.setJsonData(fbdo.payload());
    FirebaseJsonData jsonData;

    if (json.get(jsonData, "fields/userid/stringValue")) {
      userid = jsonData.stringValue;
      Serial.println("User ID: " + userid);
      dataRetrieved = true;
    } else {
      Serial.println("User ID field not found!");
    }
  } else {
    Serial.println("Error retrieving document: " + fbdo.errorReason());
  }
}
////////////////////////////////////////////////////////////
// Main Loop
////////////////////////////////////////////////////////////
void loop() {
  mpu.update();
  timeClient.update();

  // Monitor heap and stack
  Serial.print("Free Heap: ");
  Serial.println(ESP.getFreeHeap());

  if (!dataRetrieved || !Firebase.ready() || userid.length() == 0) return;

  // Start new session if flagged
  if (newSession) {
    time_t rawTime = timeClient.getEpochTime();
    struct tm *timeinfo = localtime(&rawTime);

    char dateStr[9];
    strftime(dateStr, sizeof(dateStr), "%Y%m%d", timeinfo);

    char timeStr[7];
    strftime(timeStr, sizeof(timeStr), "%H%M%S", timeinfo);

    currentSession = "users/" + userid + "/Sensor/" + String(dateStr) + "/Time" + String(timeStr);
    newSession = false;

    Serial.println("New session: " + currentSession);
  }
```

```
  // Send data every 2 seconds
  if (millis() - lastSendMillis > 2000) {
    lastSendMillis = millis();
    float rom = mpu.getAccAngleX();
    float accY = mpu.getAccY();
    float gyroX = mpu.getGyroX();

    if (!isnan(rom)) {
      unsigned long currentTimestamp = millis();
      if (currentTimestamp <= lastTimestamp)
        currentTimestamp = lastTimestamp + 1;
      lastTimestamp = currentTimestamp;

      String docPath = currentSession + "/" + String(currentTimestamp);
      FirebaseJson content;
      content.set("fields/ROM/doubleValue", round(rom * 100.0) / 100.0);
      content.set("fields/accY/doubleValue", round(rom * 100.0) / 100.0);
      content.set("fields/gyroX/doubleValue", round(rom * 100.0) / 100.0);
      content.set("fields/Time/integerValue", currentTimestamp);


      if (Firebase.Firestore.patchDocument(&fbdo, PROJECT_ID, "(default)", docPath.c_str(), content.raw(), "ROM,accY,gyroX,Time")) {
        Serial.println("Data uploaded");
      } else {
        Serial.println("Upload failed: " + fbdo.errorReason());
      }
    }
  }
}
```

Appendix C: Arduino Code for Main System.

```
#include <IRremote.hpp>
#include <AccelStepper.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define IR_RECEIVER_PIN 32  // IR receiver pin
#define STEP_PIN  14        // Step pin for DM860H
#define DIR_PIN   13        // Direction pin
#define ENABLE_PIN 27       // Enable pin (optional)
#define LIN_1 4
#define LIN_2 5
#define ESTOP_PIN 15
#define MOTOR_PIN 25

// Define HEX codes for IR remote (Replace with actual values)
#define IR_CODE_MODE1 0xBA45FF00 //
#define IR_CODE_MODE2 0xB946FF00 //
#define IR_CODE_MODE3 0xB847FF00 //
#define IR_CODE_MODE4 0xBB44FF00 //
#define IR_CODE_MODE5 0xBF40FF00
#define IR_CODE_MODE6 0xBC43FF00
#define IR_CODE_MODE7 0xF807FF00
#define IR_CODE_MODE8 0xEA15FF00
#define IR_CODE_MODE9 0xF609FF00
#define IR_CODE_START 0xE619FF00 //
#define IR_CODE_STOP 0xF20DFF00 //
#define IR_CODE_UP  0xE718FF00  //
#define IR_CODE_DOWN  0xAD52FF00  //
#define IR_CODE_CW  0xA55AFF00  //
#define IR_CODE_CCW 0xF708FF00  //
#define IR CODE COMPLETE 0xE916FF00  //
#define IR_CODE_COMPLETE 0xE916FF00  //
#define IR_REPEAT   0x0         // IR repeat signal

#define STEPS_PER_REV 1600  // Steps per revolution (based on driver setting)
#define GEAR_RATIO 50
#define OUTPUT_STEPS_PER_REV (STEPS_PER_REV * GEAR_RATIO)
#define STEPS_PER_DEGREE (OUTPUT_STEPS_PER_REV / 360.0)
//Hand
long steps90 = 90 * STEPS_PER_DEGREE;
long steps85 = 85 * STEPS_PER_DEGREE;

long steps70 = 70 * STEPS_PER_DEGREE;
long steps30 = 30 * STEPS_PER_DEGREE;
long steps40 = 40 * STEPS_PER_DEGREE;
long steps50 = 50 * STEPS_PER_DEGREE;
long steps100 = 100 * STEPS_PER_DEGREE;
long steps140 = 140 * STEPS_PER_DEGREE;
//leg


long steps45 = 45 * STEPS_PER_DEGREE;


IRData receivedData;
AccelStepper stepper(AccelStepper::DRIVER, STEP_PIN, DIR_PIN);
LiquidCrystal_I2C lcd(0x27, 16, 2); // Set the LCD address to 0x27 for a 16x2 display
```

```
// Stepper Control Variables

bool isMoving = false;
bool isClockwise = true;
bool mode1Active = false;
bool mode1Ready = false;
bool mode2Active = false;
bool mode2Ready = false;
bool mode3Active = false;
bool mode3Ready = false;
bool mode4Active = false;
bool mode4Ready = false;
bool mode5Active = false;
bool mode5Ready = false;
unsigned long lastSignalTime = 0;
const unsigned long timeout = 300;
uint32_t lastValidCode = 0;
long initialPosition = 0;  // Stores initial position
static bool movingTo45 = true;
static bool movingTo70 = true;
static bool movingTo100 = true;
static bool movingTo140 = true;
bool systemLocked = false;
bool isLinMoving = false;
uint32_t linLastCode = 0;
void setup() {
    Serial.begin(115200);
    IrReceiver.begin(IR_RECEIVER_PIN, ENABLE_LED_FEEDBACK);
    stepper.setMaxSpeed(6500);
    stepper.setAcceleration(16500);

    pinMode(ENABLE_PIN, OUTPUT);
    digitalWrite(ENABLE_PIN, LOW);
    pinMode(LIN_1, OUTPUT);
    pinMode(LIN_2, OUTPUT);
    digitalWrite(LIN_1, LOW);
    digitalWrite(LIN_2, LOW);

    lcd.init();
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print("IR Stepper Ready");

    Serial.println("IR Stepper Control Ready...");
    stepper.setCurrentPosition(0);
    initialPosition = stepper.currentPosition();


    pinMode(ESTOP_PIN, INPUT_PULLUP); // NO button – reads HIGH normally

}
```

```
void loop() {

    if (digitalRead(ESTOP_PIN) == LOW) {

      systemLocked = true;
      lcd.setCursor(0, 1);
      lcd.print("Emergency Stop Pressed");

      if (systemLocked) {
        ESP.restart();
        return;
      }

    }
    if (IrReceiver.decode()) {
        receivedData = IrReceiver.decodedIRData;
        uint32_t currentCode = IrReceiver.decodedIRData.decodedRawData;

        //Ignore noise signals but allow long-press signal (0x0)
        if (currentCode == 0xFFFFFFFF) {
            IrReceiver.resume();
            return;
        }

        // Ignore unrecognized codes
        if (currentCode != IR_CODE_MODE1 && currentCode != IR_CODE_MODE2 &&
            currentCode != IR_CODE_MODE3 && currentCode != IR_CODE_MODE4 &&
            currentCode != IR_CODE_MODE5 && currentCode != IR_CODE_START &&
            currentCode != IR_CODE_STOP && currentCode != IR_CODE_UP && currentCode != IR_CODE_DOWN &&
            currentCode != IR_CODE_CW && currentCode != IR_CODE_CCW &&
            currentCode != IR_CODE_COMPLETE && currentCode != IR_REPEAT) {
            IrReceiver.resume();
            return;
        }

        lcd.clear();
        lcd.setCursor(0, 0);
        if (currentCode == IR_CODE_CW || (currentCode == IR_REPEAT && lastValidCode == IR_CODE_CW)) {
            lcd.setCursor(0, 1);
            lcd.print("motor CW");
            mode1Active = false;
            mode2Active = false;
            mode3Active = false;
            mode4Active = false;
            mode5Active = false;
            isClockwise = true;
            isMoving = true;
            stepper.setSpeed(500);
            lastValidCode = IR_CODE_CW;
        }
        else if (currentCode == IR_CODE_CCW || (currentCode == IR_REPEAT && lastValidCode == IR_CODE_CCW)) {
            lcd.setCursor(0, 1);
            lcd.print("motor CCW");
            mode1Active = false;
            mode2Active = false;
            mode3Active = false;
            mode4Active = false;
            mode5Active = false;
            isClockwise = false;
            isMoving = true;
            stepper.setSpeed(-500);
            lastValidCode = IR_CODE_CCW;
        }
```

```
        else if (currentCode == IR_CODE_UP || (currentCode == IR_REPEAT && linLastCode == IR_CODE_UP)) {
            digitalWrite(LIN_1, HIGH);
            digitalWrite(LIN_2, LOW);
            lcd.setCursor(0, 1);
            lcd.print("Moving Up");
            isLinMoving = true;
            linLastCode = IR_CODE_UP;
        }
        else if (currentCode == IR_CODE_DOWN || (currentCode == IR_REPEAT && linLastCode == IR_CODE_DOWN)) {
            digitalWrite(LIN_2, HIGH);
            digitalWrite(LIN_1, LOW);
            lcd.setCursor(0, 1);
            lcd.print("Moving Down");
            isLinMoving = true;
            linLastCode = IR_CODE_DOWN;
        }

        else if (currentCode == IR_CODE_MODE1) {
            mode1Ready = true;
            mode1Active = false;
            movingTo70 = true;
            Serial.println("Mode 1: Moving 90° CCW");
            lcd.print("Mode 1: 90 CCW");
            stepper.move(-steps90);
            while (stepper.distanceToGo() != 0) stepper.run();
            Serial.println("Mode 1: Ready");
            lcd.setCursor(0, 1);
            lcd.print("Mode 1 Ready");
        }
        else if (currentCode == IR_CODE_MODE2) {
            mode2Ready = true;
            mode2Active = false;
            movingTo100 = true;
            Serial.println("Mode 2: Moving 90° CCW");
            lcd.print("Mode 2: 90 CCW");
            stepper.move(-steps90);
            while (stepper.distanceToGo() != 0) stepper.run();
            Serial.println("Mode 2: Ready");
            lcd.setCursor(0, 1);
            lcd.print("Mode 2 Ready");
        }
        else if (currentCode == IR_CODE_MODE3) {
            mode3Ready = true;
            mode3Active = false;
            movingTo140 = true;
            Serial.println("Mode 3: Moving 90° CCW");
            lcd.print("Mode 3: 90 CCW");
            stepper.move(-steps90);
            while (stepper.distanceToGo() != 0) stepper.run();
            Serial.println("Mode 3: Ready");
            lcd.setCursor(0, 1);
            lcd.print("Mode 3 Ready");
        }
        else if (currentCode == IR_CODE_MODE4) {
            mode4Ready = true;
            mode4Active = false;
            movingTo45 = true;
            Serial.println("Mode 4: Moving 45° CCW");
            lcd.print("Mode 4: 45 CCW");
            stepper.move(-steps45);
            while (stepper.distanceToGo() != 0) stepper.run();
            Serial.println("Mode 4: Ready");
            lcd.setCursor(0, 1);
            lcd.print("Mode 4 Ready");
        }
        else if (currentCode == IR_CODE_MODE5) {
            mode5Ready = true;
            mode5Active = false;
            movingTo45 = true;
            Serial.println("Mode 5: Moving 45° CCW");
            lcd.print("Mode 5: 45 CCW");
            stepper.move(-steps45);
            while (stepper.distanceToGo() != 0) stepper.run();
            Serial.println("Mode 5: Ready");
            lcd.setCursor(0, 1);
            lcd.print("Mode 5 Ready");
        }
```

```
    else if (currentCode == IR_CODE_START && (mode1Ready || mode2Ready||mode3Ready || mode4Ready|| mode5Ready)) {
        if (mode1Ready) {
            Serial.println("Start: Limited Functional");
            lcd.print("Limited");
            mode1Active = true;
        }
        if (mode2Ready) {
            Serial.println("Start: Moderate Functional");
            lcd.print("Moderate");
            stepper.move(-steps30);
            mode2Active = true;
        }
        if (mode3Ready) {
            Serial.println("Start: Full Functional°");
            lcd.print("Full");
            stepper.move(-steps50);
            mode3Active = true;
        }
        if (mode4Ready) {
            Serial.println("Start: Oscillate 45°");
            lcd.print("Oscillate 45");
            //stepper.move(steps5);
            mode4Active = true;
        }
        if (mode5Ready) {
            Serial.println("Start: Oscillate 45°");
            lcd.print("Oscillate 45");
            stepper.move(-steps40);
            mode5Active = true;
        }
    }
    else if (currentCode == IR_CODE_STOP) {
        mode1Active = false;
        mode2Active = false;
        mode3Active = false;
        mode4Active = false;
        mode5Active = false;
        movingTo100 = true;
        movingTo140 = true;
        movingTo70 = true;
        movingTo45 = true;
        stepper.stop();
        Serial.println("Motor Stopped");
        lcd.print("Motor Stopped");
    }
    else if (currentCode == IR_CODE_COMPLETE) {
        Serial.println("Returning to Start");
        lcd.print("Returning Start");
        returnToInitialPosition();
    }
    else if(receivedData.decodedRawData == IR_REPEAT){
      isMoving = (lastValidCode == IR_CODE_CW || lastValidCode == IR_CODE_CCW);
    }


    lastSignalTime = millis();
    IrReceiver.resume();
}
```

```
    if (isMoving) {
        stepper.runSpeed();
    }
    if (mode1Active) {
        loopMode1();
    }
    if (mode2Active) {
        loopMode2();
    }
    if (mode3Active) {
        loopMode3();
    }
    if (mode4Active) {
        loopMode4();
    }
    if (mode5Active) {
        loopMode5();
    }
    if (millis() - lastSignalTime > timeout) {
        isMoving = false;
        stepper.setSpeed(0);
    }
    if (millis() - lastSignalTime > timeout && isLinMoving) {
        digitalWrite(LIN_1, LOW);
        digitalWrite(LIN_2, LOW);
        isLinMoving = false;
        linLastCode = 0;
    }
}
void loopMode1() {
    if (!mode1Active) return;

    Serial.println("Oscillating...");
    stopMotor();
    lcd.setCursor(0, 1);
    lcd.print("Limited");

    if (stepper.distanceToGo() == 0) {
        movingTo70 = !movingTo70;
        stepper.move(movingTo70 ? -steps70 : steps70);

        stopMotor();
    }
    stopMotor();
}
void loopMode2() {
    if (!mode2Active) return;

    Serial.println("Oscillating...");
    stopMotor();
    lcd.setCursor(0, 1);
    lcd.print("Moderate");

    if (stepper.distanceToGo() == 0) {
        movingTo100 = !movingTo100;
        stepper.move(movingTo100 ? -steps100 : steps100);
        stopMotor();
    }
    stopMotor();
}
void loopMode3() {
    if (!mode3Active) return;

    Serial.println("Oscillating...");
    stopMotor();
    lcd.setCursor(0, 1);
    lcd.print("Full");

    if (stepper.distanceToGo() == 0) {
        movingTo140 = !movingTo140;
        stepper.move(movingTo140 ? -steps140 : steps140);

    }
    stopMotor();
}
void loopMode4() {
    if (!mode4Active) return;

    Serial.println("Oscillating...");
    stopMotor();
    lcd.setCursor(0, 1);
    lcd.print("Moderate");

    if (stepper.distanceToGo() == 0) {
        movingTo45 = !movingTo45;
        stepper.move(movingTo45 ? -steps45 : steps45);

    }
    stopMotor();
}
```

```
void loopMode5() {
    if (!mode5Active) return;

    Serial.println("Oscillating...");
    stopMotor();
    lcd.setCursor(0, 1);
    lcd.print("Full");

    if (stepper.distanceToGo() == 0) {
        movingTo45 = !movingTo45;
        stepper.move(movingTo45 ? -steps85 : steps85);

    }

    stopMotor();
}
void stopMotor(){
    while (stepper.distanceToGo() != 0) {
      if (IrReceiver.decode()) {
          receivedData = IrReceiver.decodedIRData;

          if (receivedData.decodedRawData == IR_CODE_STOP) {
              mode1Active = false;
              mode1Ready = false;
              mode2Active = false;
              mode2Ready = false;
              mode3Active = false;
              mode3Ready = false;
              mode4Active = false;
              mode4Ready = false;
              mode5Active = false;
              mode5Ready = false;

              stepper.stop();
              Serial.println("Stopped Oscillation");
              IrReceiver.resume();
              return;
          }
          IrReceiver.resume();
      }
      stepper.run();
    }
}
void returnToInitialPosition() {
    mode1Active = false;
    mode2Active = false;
    mode3Active = false;
    mode4Active = false;
    mode5Active = false;
    mode1Ready = false;
    mode2Ready= false;
    mode3Ready = false;
    mode4Ready= false;
    mode5Active = false;
    movingTo70 = true;
    movingTo100 = true;
    movingTo140 = true;
    movingTo45 = true;
    stepper.moveTo(initialPosition);
    while (stepper.distanceToGo() != 0) {
        stepper.run();
    }
    Serial.println("Returned to Initial Position");
    lcd.setCursor(0, 1);
    lcd.print("At Start Pos");
}
```

Appendix D: Torque Derivation for Forearm and Shank.

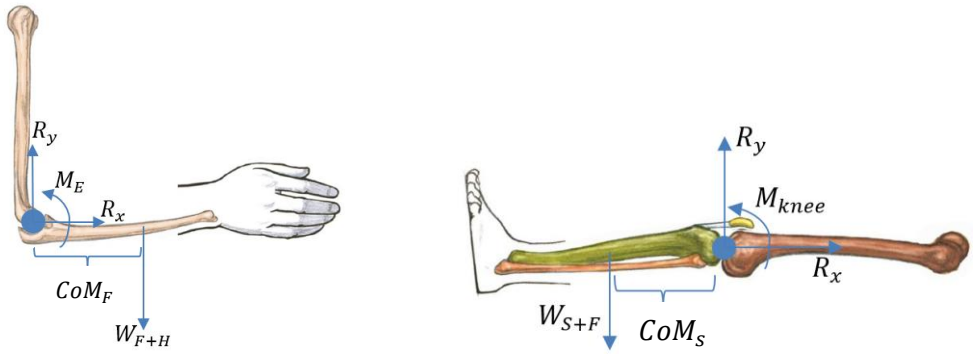


Figure D-1: Reaction Force Acting on Body Segments: (a) Forearm and Elbow Joint (b) Shank and Knee Joint.

Table D-1: Derivation of Minimum Torque for Body Segment.

Forearm and Elbow Joint	**Shank and Knee Joint**
$\overset{+}{\rightarrow}\boldsymbol{\Sigma F_x = 0}$	$\overset{+}{\rightarrow}\Sigma F_x = 0$
$\boldsymbol{R_x = 0}$	$R_x = 0$
$\overset{+}{\uparrow}\boldsymbol{\Sigma F_y = 0}$	$\overset{+}{\uparrow}\Sigma F_y = 0$
$\boldsymbol{R_y - W_{F+H} = 0}$	$R_y - W_{S+F} = 0$
$\boldsymbol{R_y - m_{F+H}g = 0}$	$R_y - m_{S+F}g = 0$
$\overset{+}{\circlearrowleft}\boldsymbol{\Sigma M = 0}$	$\overset{+}{\circlearrowleft}\Sigma M = 0$
$\boldsymbol{M_E - W_{F+H}CoM_F = 0}$	$M_{knee} - W_{S+F}CoM_S = 0$
$\boldsymbol{M_E - (m_{F+H}g)CoM_F = 0}$	$M_{knee} - (m_{S+F}g)CoM_S = 0$
$\boldsymbol{M_E = (m_{F+H}g)CoM_F}$	$M_{knee} = (m_{S+F}g)CoM_S$
$\therefore \boldsymbol{\tau_{BS\,max} = M_E = (m_{F+H}g)CoM_F}$	$\therefore \tau_{BS\,max} = M_{knee} = (m_{S+F}g)CoM_S$

where

F_x = horizontal force, N

R_x = horizontal reaction force, N

F_y = vertical force, N

R_y =vertical reaction force, N

W_{F+H} = the forearm and hand weight, N

W_{S+F} = the shank and foot weight, N

CoM_F = the center of mass location of forearm, N

CoM_S = the center of mass location for shank, N

M_E = the moment at elbow joint, N

M_{knee} = the moment at knee joint, N

m_{F+H} = forearm and hand mass, N

m_{S+F} = shank and foot mass, N

Appendix E: Speed Calculation of Stepper Motor.

Parameters:

1. Motor Specifications:
   - Motor Step Angle: 1.8° per full step
   - Micro stepping: 8 micro steps per full step
   - Steps per Revolution (Motor Shaft): 200 steps/rev

2. Gear Reduction:
   - Gear Ratio: 50:1

Calculations:

$$Steps\ per\ rev = \frac{200steps}{rev} \times \frac{8microsteps}{step}$$
$$= \frac{1600steps}{rev}$$

$$Output\ steps\ per\ rev = \frac{1600steps}{rev} \times 50$$
$$= \frac{80{,}000steps}{rev}$$

$$Degrees\ per\ step\ at\ output = 80{,}000steps360°$$
$$= 0.0045°per\ step$$

$$Steps\ per\ second\ (High\ Speed) = \frac{29.25°/s}{0.0045°}$$
$$\approx \frac{6500steps}{sec}$$

$$Steps\ per\ second\ (Low\ Speed) = \frac{13.5°/s}{0.0045°}$$
$$\approx \frac{3000steps}{sec}$$