YILDIZ TECHNICAL UNIVERSITY

FACULTY OF MECHANICAL ENGINEERING

DESIGN AND CONTROL OF AN ASSISTIVE EXOSKELETON KNEE MECHANISM

Bahadır Sami Ekmen	1706A021
Yücel Can Aksu	1706A017
Mustafa Şimşek	1606A022

MECHATRONIC SYSTEM DESIGN REPORT

Project Consultant: Assoc. Prof. Cüneyt Yılmaz

YILDIZ TECHNICAL UNIVERSITY

FACULTY OF MECHANICAL ENGINEERING

DESIGN AND CONTROL OF AN ASSISTIVE EXOSKELETON KNEE MECHANISM

Bahadır Sami Ekmen	1706A021
Yücel Can Aksu	1706A017
Mustafa Şimşek	1606A022

MECHATRONIC SYSTEM DESIGN REPORT

Project Consultant: Assoc. Prof. Cüneyt Yılmaz

ABSTRACT

Design and Control of an Assistive Exoskeleton Knee Mechanism

Prepared by **Mustafa Şimşek, Yücel Can Aksu, Bahadır Sami Ekmen**

June, 2021

In this senior design project, design and implementation of an assistive lower-extremity exoskeleton knee mechanism have been done. The knee mechanism is primarily used to support people with walking difficulties due to orthopedic disorders, orthopedic trauma, or the difficulty of movements because of old age. The thesis consists of a mechanical design of a knee mechanism, mathematical derivation of equations of motion, dynamic modelling, MATLAB modelling.

Within the scope of the thesis work, the mechanical design of the exoskeleton is performed using Solidworks software. The stress analysis of the critical positions where the mechanism is subjected to maximum loading is also will carry out using Solidworks software. Equations of motions for a rigid single pendulum model that represents the knee mechanism has been derived and a MATLAB Simulink model will be build based on the governing equations. Also, another MATLAB model will be built by transferring a solid model created on Solidworks software to the MATLAB Simulink environment. A PID computed torque controller is designed to control both mathematical and Simscape models. Both MATLAB models will be fed by the desired walking trajectory and design verifications and validations will be carry out by different kind of outputs such as tracking error and required torque values. A real-time MATLAB graphical user interface is created to acquire data from system's sensors. Later, an artificial neural network algorithm will be used order to classify the sensor signals taken from the users underfoot.

Keywords: Lower Extremity Exoskeleton, Wearable Exoskeleton, Walking Assistive Exoskeleton, Computed Torque Control, PID Control, Force-Signal Based Control.

ACKNOWLEDGEMENTS

Firstly, we would like to express our sincere gratitude to our advisor Assoc. Prof. Cüneyt Yılmaz for the continuous support of our project, for his patience, motivation, and immense knowledge. His guidance helped us in all the time of research and writing of this thesis.

TABLE OF CONTENTS

ABSTRACT i
ACKNOWLEDGEMENTS ii
TABLE OF CONTENTS iii
TABLE LIST vi
FIGURE LIST vii
APPENCICES viii
SYMBOL LIST ix
1. INTRODUCTION 1
1.1. Goal of the Project 2
1.2. Motivation 3
1.3. Literature Research 4
2. REQUIREMENT SPECIFICATIONS 12
2.1. Market Requirements 12
2.2. Technical Requirements 13
2.2.1. Actuators 13
2.2.2. Gearbox 14
2.2.3. Encoder 14
2.2.4. ESC 15
2.2.5. Sensors 15
2.2.6. Controller 16
2.2.7. Battery 16
2.2.8. Spring 17
2.2.9. Frame of the Mechanism 17
3. THEORETICAL BACKGROUND 18
3.1. Intention Determination Method 18
3.2. Control Method 20
4. DESIGN 21
4.1. Mechanical Design 21
4.2. Electronic Schematic 24
4.3. Mathematical Model 25
4.4. Controller Design 27
5. DESIGN VERIFICATIONS 29
5.1. Mechanical Simulation 29

5.2. Controller Simulation Output ... 30
5.3. Entire System Simulation ... 33
6. DESIGN IMPLEMENTATION ... 34
6.1. Mechanical Implementation ... 34
6.2. Electronical Implementation ... 35
6.2.1. Force Sensor ... 35
6.2.2. Arduino as ADC ... 36
6.2.3. Entire Electronic System ... 36
6.3. Control Implementation ... 37
6.3.1. Controller ... 37
6.3.2. Motor ... 37
6.3.3. Motor Driver ... 38
6.3.4. PID Control Method ... 39
7. IMPLEMENTATION ANALYSIS ... 40
8. ASSUMPTIONS AND ERROR ANALYSIS ... 41
8.1. Assumptions ... 41
8.2. Error Analysis ... 41
9. RISK ANALYSIS ... 42
9.1. Risk Management Plan ... 42
9.2. Taken Measures for Predictable Risks ... 44
10. WORKING PLAN ... 45
10.1. Schedule of the Project ... 45
10.2. Task Distribution ... 46
11. BUDGET ... 47
11.1. Cost of the Project ... 47
11.2. Financial Support Applications ... 48
12. RESULTS, SUGGESTIONS AND FUTURE WORKS ... 49
12.1. Results ... 49
12.2. Summary, Suggestions and Future Works ... 49
REFERENCES ... 52
APPENDICES ... 54
Appendices A ... 54
Appendices B ... 57
CURRICULUM VITAE ... 69
• Mustafa Şimşek ... 69

- Yücel Can Aksu 69
- Bahadır Sami Ekmen.. 69

TABLE LIST

Table 1: TÜİK Statistics about Disabled Individuals [1] 3
Table 2: Technical Requirements.. 13
Table 3: Risk Register.. 43
Table 4: Schedule of the Project 45
Table 5: Task Distribution 46
Table 6: Cost of the Project.. 47

FIGURE LIST

Figure 1: Exoskeleton History[2] 4
Figure 2: Lokomat by Hocoma AG [3] 6
Figure 3: Hardyman[4] 6
Figure 4: A Pneumatic Exoskeleton[5] 7
Figure 5: BLEEX[6] 7
Figure 6: HULC[7] 8
Figure 7: LTU-LEE[8] 8
Figure 8: EKSO[9] 9
Figure 9: HAL[11] 9
Figure 10: C-Brace[12] 10
Figure 11: C-Brace Mechanism[12] 10
Figure 12: C-Leg[13] 11
Figure 13: Motor Reducer 14
Figure 14: Force Sensor[18] 15
Figure 15: Regulator Circuit 16
Figure 16: Force Acting on the Foot Sole Parts[19] 18
Figure 17: Encoder and its Output[20] 19
Figure 18: PID Control System 20
Figure 19: Entire Exoskeleton 21
Figure 20: U - Joints on the Hip and Wrist 21
Figure 21: Knee Part of The Exoskeleton 22
Figure 22: Motor and Reducer on the Exoskeleton 22
Figure 23: Spring and Encoder on the Exoskeleton 23
Figure 24: Electronic Box 23
Figure 25: Electronic Schematic 24
Figure 26: Mathematical Model of the System 25
Figure 27: Controller Diagram 27
Figure 28: Free Body Diagram of the System 27
Figure 29: Main Frame Mechanical Simulation 29
Figure 30: Bevel Gear Mechanical Simulation 30
Figure 31: Position of the Leg 30
Figure 32: Torque Produced by the Motor 31
Figure 33: Torque Produced by the Spring 31
Figure 34: Output Torque 32
Figure 35: Torque Showcase 32
Figure 36: Simulink Model of the Entire System 33
Figure 37: Motor Controller 33
Figure 38: Implemented Design 34

APPENCICES

Appendix A 1: Encoder Specifications 54
Appendix A 2: Motor Driver's Specifications 55
Appendix A 3: Force Sensor Specifications 56

Appendix B 1: MATLAB-Simulink Controller of the System 57
Appendix B 2: Transfer Function Constants 57
Appendix B 3: PID Constants 58
Appendix B 4: Desired Torque Function 58
Appendix B 5: Position Function 59
Appendix B 6: Spring Torque Function 60
Appendix B 7: Simscape Exoskeleton System 60
Appendix B 8: Arduino ADC Code 61
Appendix B 9: Raspberry Python PID Code 62

SYMBOL LIST

μ_R	Planetary Gear Efficiency
μ_m	Motor Efficiency
μ	Efficiency
τ_{em}	Motors Nominal Torque
τ_L	Load Torque
CPT	Counts Per Turn
$U(s)$	Control Signal
E(s)	Error Signal
K_p	Proportional Coefficient
K_i	Integral Coefficient
K_d	Derivative Coefficient
τ_m	Motor Torque
θ	Motor Angle/Position
$\dot{\theta}$	Angular Velocity
$\ddot{\theta}$	Angular Acceleration
J	Moment of Inertia
J_M	Moment of Inertia of Motor
J_L	Moment of Inertia of Load
b	Friction Coefficient
b_M	Friction Coefficient of Motor
b_L	Friction Coefficient of Load
α	Load Angle/Position
M	Gear Reduction Ratio
G(s)	Transfer Function

1. INTRODUCTION

Mechatronics is an appropriate combination of the words mechanical and electronics, and in 1969 an engineer from the Japanese Yasukawa Electric Company coined the word "mechatronics" to mean the merger of the fields of mechanical and electronic engineering. Mechatronics is a multi-controlled engineering branch based on mechanical, electronic, software and control engineering.

Biomechatronic is a multidisciplinary field of science that adapts software, mechanical and electronic design principles to biological systems. Typical applications include developing mechanical designs in meso, micro and nano scales for the purpose of carrying or transmitting a force of biological systems, supporting living things with smart control systems, viewing, and tracking biological parameters of organisms. Biomechatronic; biology, mechanical engineering, electronic engineering, computer engineering, physiology, and medicine, is a field of science where joint studies will be carried out involving many disciplines.

The purpose of biomechatronic systems is to support limb movements, strengthen limbs, make problematic body functions functional, or perform functions by replacing the units that perform these functions themselves.

Exoskeleton robots are a unique form of professional service robots, deployed in a wide range of applications, intended to mimic, augment, or enhance the body's own movements. These robots provide essential support for human motion, with potential uses ranging from consumer products to military deployment.

- Exoskeletons are wearable devices that work in tandem with the user.
- Exoskeletons are placed on the user’s body and act as amplifiers that augment, reinforce or restore human performance.
- Exoskeletons can be made from rigid materials such as metal or carbon fibre, or they can be made entirely out of soft and elastic parts.

- Exoskeletons can be powered and equipped with sensors and actuators, or they can be entirely passive.
- Exoskeletons can be mobile or fixed/suspended (usually for rehabilitation or teleoperation).
- Exoskeletons can cover the entire body, just the upper or lower extremities, or even a specific body segment such as the ankle or the hip.

1.1. Goal of the Project

The main and ideal goal of this project is to develop an exoskeleton knee mechanism that will help the elderly and/or disabled people to continue their daily lives with ease and help them perform basic movements such as walking, climbing stairs, sitting down-getting up.

In this project to achieve project's goals it is aimed to run tests and simulations of the application of artificial intelligence and control algorithm by using the dynamic equations which collected from the exoskeleton. To do these tasks, followings must be performed:

- Solidworks modelling of the exoskeleton mechanism
- Transferring this Solidworks model to MATLAB-Simulink and running simulation to obtain ideal parameters for actuators.
- Using these parameters, selecting materials, and producing the parts.
- Using computed torque control as a control algorithm
- Selection and simulation of the sensors and actuators.
- Using neural network and machine learning in artificial intelligence.
- Creating a user-friendly interface.

By following these steps, can present a stable system which can interpret with sensors of selection autonomously, which will be taken from user's ankle/knee with the machine learning method and understand the intention of the user which will activate actuators of the exoskeleton following this intention.

1.2. Motivation

According to research by TUİK[1] in 2010, which can be seen in Table 1, 8.8% of disabled people suffers from orthopedic disability. In context there is 1 billion disabled individuals on Earth and approximately 1.5 million on Turkey. 8.8% of that number rounds up to 125.000. The data which gathered from TURKSTAT 2017 says that the number of people over the age of 65 in Turkey is close to 7 million. These 2 groups of individuals have the problem of interruption of daily physical activities due to muscle or cartilage problems. Also in the same research it is stated that 56% of these individuals are male and 39% of these individuals' age are between 25-44.

Table 1: TÜİK Statistics about Disabled Individuals [1]

(%)

	Toplam Total	Görme engelli Visual disability	İşitme engelli Hearing disability	Dil ve konuşma engelli Language and speech disability	Ortopedik engelli Orthopedic disability	Zihinsel engelli Intellectual disability	Ruhsal ve duygusal engelli Mental and emotional disability	Süreğen hastalık Chronic illness	Çoklu engellilik Multiple disability
Toplam-Total	**100.0**	**8.4**	**5.9**	**0.2**	**8.8**	**29.2**	**3.9**	**25.6**	**18.0**
Cinsiyet-Sex									
Toplam-Total	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
Erkek-Male	58.6	67.0	57.5	67.0	56.2	61.1	67.9	56.2	53.5
Kadın-Female	41.4	33.0	42.5	33.0	43.8	38.9	32.1	43.8	46.5
Yaş grubu - Age group									
Toplam-Total	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
0 - 6	4.9	1.4	9.6	25.1	3.7	7.4	2.0	3.6	3.7
7 - 14	16.2	5.1	17.4	37.1	5.1	36.1	10.5	4.6	11.5
15 - 24	17.2	16.1	20.9	14.9	13.1	27.5	9.3	9.2	14.9
25 - 44	27.7	36.2	32.4	11.7	39.2	23.3	49.5	23.6	25.0
45 - 64	18.9	25.5	12.0	7.5	22.1	4.9	22.1	33.1	18.4
65 +	15.2	15.8	7.7	3.7	16.7	0.8	6.6	25.9	26.4

Main motivation in this project is to increase the living standards of the elderly and/or lower limb related disabled people. Also, the exoskeleton mechanism in this project can be used to improve the working conditions of certain works; lower the chance of deformations and diseases in the legs.

Furthermore, the numbers in the market of exoskeletons in Turkey are astronomical. People who do not have a fortune cannot pay 75.000 USD for a lower limb related exoskeleton.

1.3. Literature Research

The exoskeleton itself is a term that centred by human. It suits human body and function like its corresponding part. This can be seen as human's intelligence with the power of machine.

Exoskeleton robots can be classified by the purpose of muscle strength support, power assistance and power augmentation systems. They can also be categorized according to muscle strength supporting parts: upper limb systems, lower limb systems, upper and lower limbs integrated systems, and specific joint muscle strength support system. You can see the history of the exoskeleton in Figure 1.

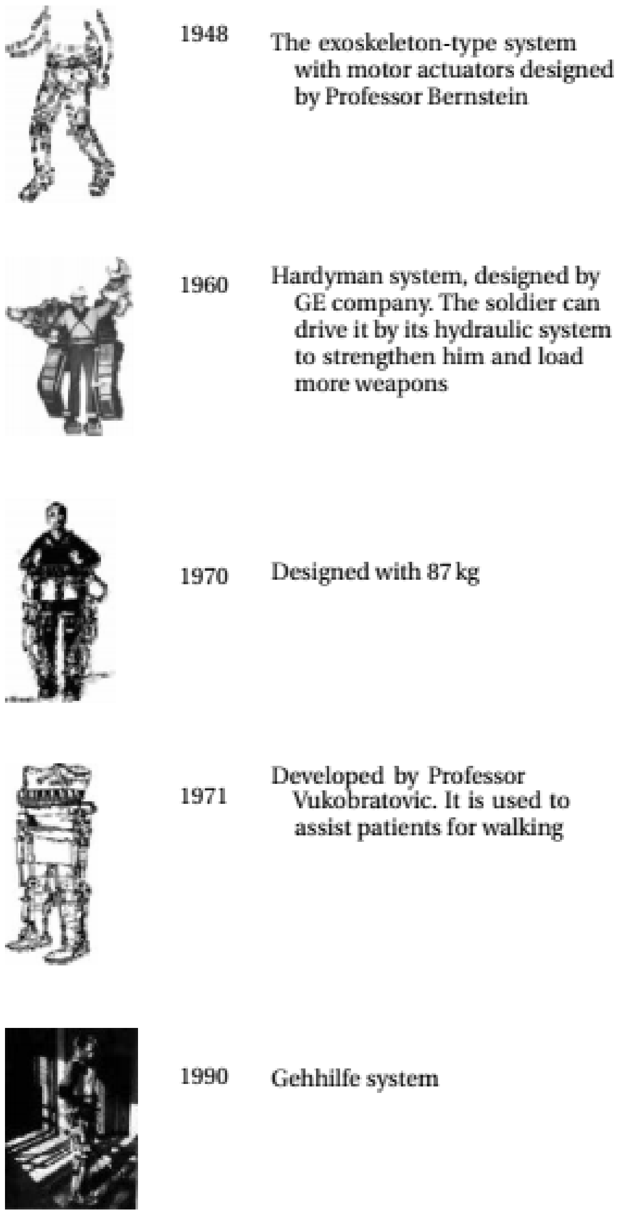


Figure 1: Exoskeleton History[2]

Nicholai A. Bernstein thought up plans for an above the-knee, electric-motor driven prosthesis to provide movement to casualties of war, but it stayed as a thought. And then the first functional exoskeleton, Hardyman, was built by General Electric in 1968. The suit was powered by hydraulics. The drawback of the Hardyman that the required pumps and bladders occupied almost a room. M. Vukobratovic and his team developed an exoskeleton that assists human movement in 1972. A similar product named Gehhilfe by the German prosthetics company Otto Bock was also developed for the same reason in 1990.

Recently, the hybrid assistive limb (HAL) developed by Professor Yoshiyuki from Tsukuba University to support and strengthen the physical capabilities of user. To control the HAL sensors such as angle sensors, myoelectrical sensors, floor sensors, etc. is used. And it had hybrid control systems, which consist of the autonomous controller such as posture control and the comfortable power assist controller based on biological feedback and predictive feed forward.

The Defense Advanced Research Projects Agency (DARPA) proposed a project, Exoskeleton for Human Performance Augmentation (EHPA) to strengthen the human user. the system contained two hydraulic driven metal legs, an in-built power supply source and a backpack-like frame for load. More than 40 different sensors are used to gather information. The computer in the backpack-like frame is like a brain and controls the whole system. A hybrid power source supplied the energy for hydraulic actuators and control systems.

Additionally, the labs in Nanyang Technological University and Kanagawa University have also reached some achievements in these years.

Physical disability and paralyzed rehabilitation are one of the other areas of the application of exoskeletons. Since MIT created the first rehabilitation exoskeleton arm for muscle dystrophy patients, Golden Arm and MIT-MAUS in 1970s and 1990s, respectively, Stanford University developed the Arm Guide and MIME prototype systems in 2000 and Rutgers University succeeded the research on the lower limb exoskeleton rehabilitation system.

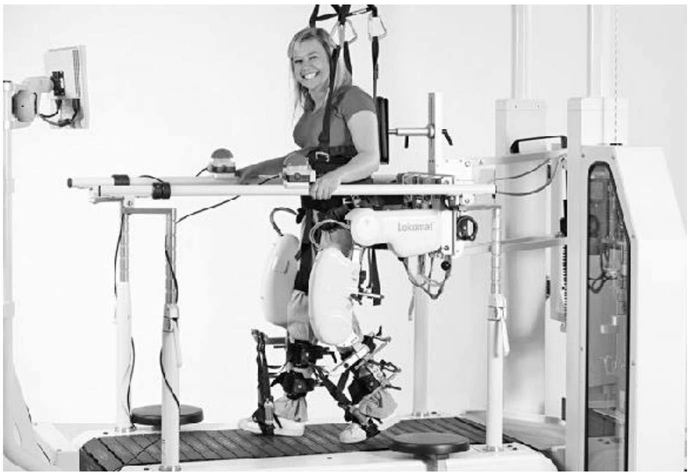


Figure 2: Lokomat by Hocoma AG [3]

Hocoma AG from Switzerland designed the Lokomat, in Figure 2, which was a fixed movement rehabilitation exoskeleton. It can adjust the training gait style according to the biofeedback of the patient, Also Haptic Walker was designed based on the principle of programmable footplates so that the patient's feet were attached to two footplates, which moved his feet on arbitrary foot trajectories.

Figure 3: Hardyman[4]

Hardyman, can be seen in Figure 3, is a project started in 1965 to create a reinforced exoskeleton that can increase manpower by 25 times. Hardiman exoskeleton experiments resulted in uncontrollable movements that could harm the user, so this exoskeleton was never run with a human inside. The project was not successful overall.

In 1991, a pneumatic power support suit, which can be seen in Figure 4, was designed, a wearable utility suit that significantly reduced the physical burden of the wearer.

The exoskeleton suit consists of shoulders, arms, a spine, a waist, and legs. Each joint has an angle sensor (potentiometer), and elbow, waist and knee joints have pneumatic actuators. These actuators are operated by solenoid valves and air pumps. These air pumps are driven by DC motors. In addition, Muscle sensors were used to control this mechanism.

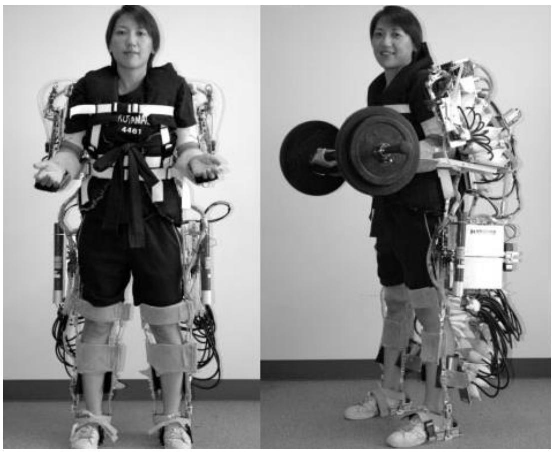


Figure 4: A Pneumatic Exoskeleton[5]

The Defense Advanced Research Project Agency (DARPA) financed the BLEEX, in Figure 5, project in 2000. U.C. Berkeley's Human Engineering and Robotics Laboratory has successfully demonstrated the first experimental Exoskeleton in which the user can carry a heavy load while feeling only a few kg. The Berkeley exoskeleton system can be used by soldiers, disaster relief workers, wildfire fighters. The device helps to move necessary equipment and materials over all types of terrain with minimal effort.

Figure 5: BLEEX[6]

The hip, knee and foot joints of the device were moved with electro-hydraulic pistons. More than 40 sensors were used to control the device.

Figure 6: HULC[7]

HULC, in Figure 6, and LTU-LEE, in Figure 7, as the example of the power augmentation exoskeleton robots. These exoskeleton robots mostly used in military while can be used other areas too. HULC can nullify the weight up to 90kg and reduce oxygen consumption 5%-12% at a 3.2 km/h speed without a payload while LTU-LEE can nullify the weight up to 30kg and can achieve 4 km/h walking speed, 13.5 km/h running speed which making it one of the fastest reported powered exoskeletons in the world.

Figure 7: LTU-LEE[8]

Figure 8: EKSO[9]

EKSO, in Figure 8, and HAL, in Figure 9, as the example of the power assistance exoskeleton robots. EKSO supports the human during standing and walking while HAL can enable the user whose muscles are not capable of supplying the power required to walk [10].

Figure 9: HAL[11]

One of the closest products in market to project exoskeleton this thesis about is C-Brace, can be seen in Figure 10, from Ottobock. But Ottobock only manufactures this product for EU, US, UK. Only Ottobock products that can find are for amputated people.

Figure 10: C-Brace[12]

In Figure 11, you can see C-Brace and its circuitry. In shared videos and data, it is obtained that they are using spring in ankle which is made from carbon and fiberglass in addition to this spring there is an ankle moment sensor in there as well. The black box at the side of the knee has Li-Po battery, a knee angle sensor which has precision of 0,02 seconds, a microprocessor which process the signals coming from the multiple sensors, at last a mechanical part which consists of motors and “piston”.

Figure 11: C-Brace Mechanism[12]

C-Brace mostly used in physiotherapy which allow the users to do basic things and get the muscle memory going such as sitting up and down, climbing stairs, slow walking with hand supports. Ottobock says: “A bionic leg brace for people living with partial paralysis, spinal injury, post-stroke and post-polio syndrome.

The C-Brace® computer-controlled orthosis is an entirely new approach to walking when back injuries or leg muscle weakness limit you.” For C-Brace

Last price tag tracked was around 75,000 USD. Including tax, inflation and many other things Turkish price tag for C-Brace probably be around 100,000-120,000 USD.

What appears to be the closest to C-Brace is this C-Leg model for amputated people which you can see in Figure 12. Which has similar movement principals but far more complex intention identification.

Figure 12: C-Leg[13]

Ottobock Turkey talks about C-Leg as; for over 15 years, more than 40 thousand worldwide uses C-Leg. Product is for patients with the expectation of high stability and independency. Also, mobility and freedom are. Knee joint is checked by a complex sensor. With this sensor, C-Leg can adapt to various walking speeds.

2. REQUIREMENT SPECIFICATIONS

2.1. Market Requirements

In this project market requirements taken on 3 main courses:

a) Safety:

- Must be reliable
- Must be robust
- Must not disturb other functions of body

b) Function:

- Must be precise
- Must adapt to speed
- Must be compatible for both legs

c) Competition:

- Must be durable
- Must be innovative
- Personalization
- Must has competitive price
- Ease of use
- Must be light

2.2. Technical Requirements

Technical requirements with the explanations and research of it can be seen in Table 2.

Table 2: Technical Requirements

Technical Requirements	Explanation
Must be used by up to an 80 kg user.	46% of the people with an orthopedic disability is 70-80 kg[1].
Force sensor must be able to measure up to 200N depending on the different points of foot sole.	Maximum point force that an 80 kg person can produce at a point[14].
Speed must be 25mm/sec.	An average person’s knee compression and rebound speed[15].
Weight of the project must be less than 1.5 kg.	Most of the similar products' knee part with a motor weighs 1.5 kg.
120° operational range	Human knees average extension and flexion angles.[16]
Assistment rate: 50%	Since project just covers the knee, it is overcostly to assist 100%
Total cost must be less than 5000TL.	Budget is 5000TL

2.2.1. Actuators

Required torque and velocity values are important for the movement of the knee exoskeleton. To select the actuators that meets the required torque and speed, as a first step in accordance with the anthropometric value’s average human knee speed (linear and radial) and average necessary torque has been calculated. After the calculations and all the required values are gathered. As a result, an average human’s knee can use peak of 7.5 Nm at his gait cycle. Since the project aims to assist at a 50% rate, this value drops to 3.75 Nm. With the factor of safety of 1.5 the exoskeleton needs 5.6 Nm torque. After a literature and market search, considering these required values (speed, torque, weight of the motor, price, size), has been selected as: 5840-31ZY with integrated reducer.

2.2.2. Gearbox

You can see the inside of the integrated gearbox of the motor in Figure 13. Worm gearbox is used to satisfy our torque need.

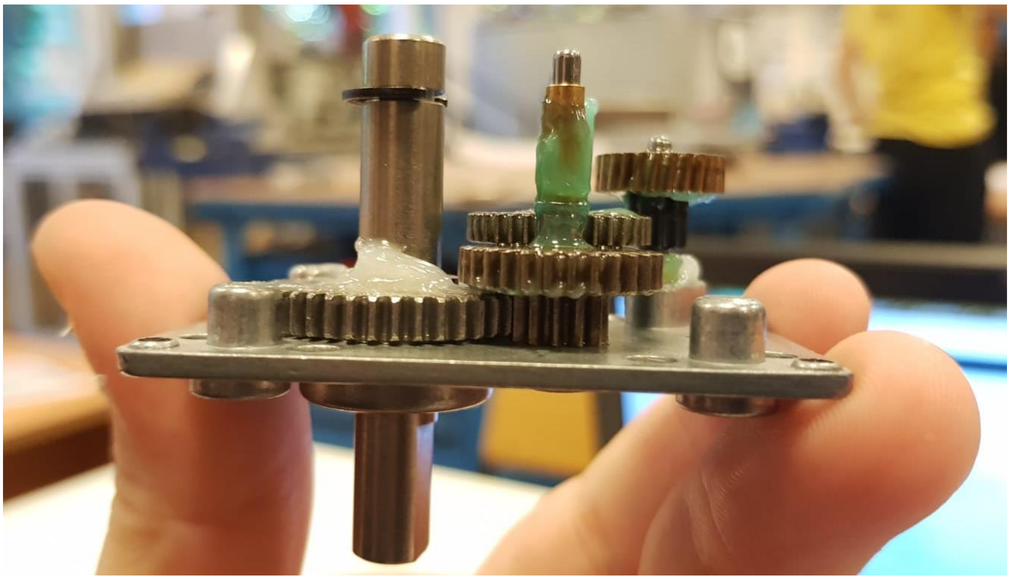


Figure 13: Motor Reducer

2.2.3. Encoder

Incremental encoder is decided to use for this project. The average angular velocity of the human knee joint is 350 deg/s [17].

$$350 \, {}^{deg}/_{s} = 6.1 \, {}^{rad}/_{s} = 58.25 \, rpm = 0.97 \, rps \qquad \text{Eq.1}$$

Resolution for the encoder:

$$\frac{360^o}{CPT} = \frac{360^o}{1024} = 0.351 \qquad \text{Eq.2}$$

It means that out encoder can show 0.3 degree for each count. To obtain working frequency of the encoder, multiply angular velocity of knee joint and CPT value of the encoder,

$$CPT \times RPS = 1024 \times 0.97 = 993.28 \, Hz \qquad \text{Eq.3}$$

The chosen encoder's specifications can be found in Appendix A1.

2.2.4. ESC

- Should be able to give 12V to motor.
- Should be able to drive 16W DC motor.
- Can put out the 3A continuous current, 5A peak current for the motor to be driven.

Motor driver should be able to provide enough current to the motor to generate sufficient torque. Pololu MC33926, its specifications can be seen in Appendix A2, provides 3A continuous, 5A peak current because the chosen motor, 5840-31ZY, has 10Nm peak torque and 3A continuous current would provide 6Nm which is more than enough in our case.

2.2.5. Sensors

The intention of the user will be determined by the pressure sensor it will placed on the sole of the foot. It is considered to place more than one pressure sensor on the sole of the foot so that understanding which part of the sole of the foot is pressurized at which stage of the walking period.

Chosen sensor is Keyestudio FlexiForce Pressure Sensor, its specifications can be seen in Appendix A3, can be seen in Figure 14. This sensor is an FSR (Force Sensing Resistor) type of sensor. The output resistance of the sensor decreases when pressure is applied. This sensor has an adjustable measuring range. According to datasheet, the measuring range can be increased by changing the input voltage and the reference resistance.

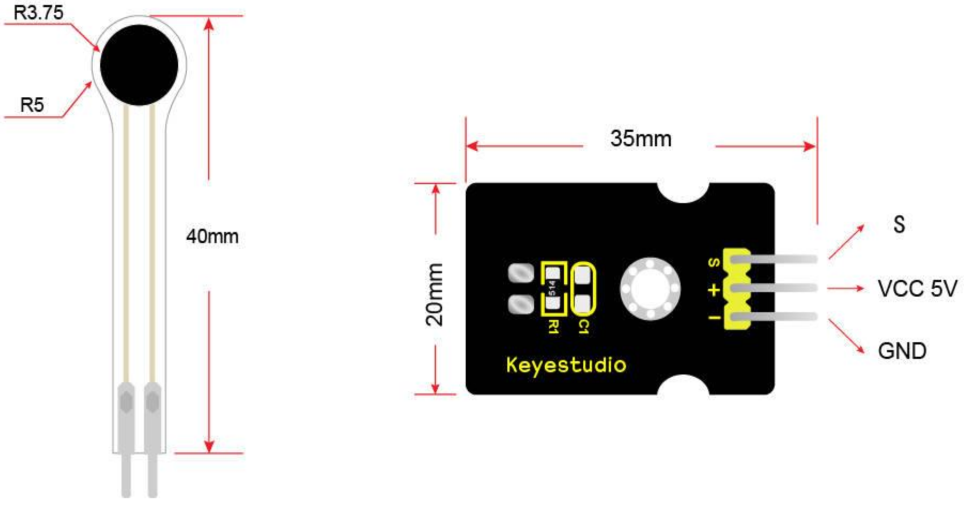


Figure 14: Force Sensor[18]

2.2.6. Controller

As for the controller, personal computers used for the simulation and testing. After everything is settled and certain, it is planned to use Raspberry Pi 4. It has Quad-Core Arm Cortex-A72 CPU, 2GB LPDDR4-3200 SDRAM, Bluetooth 5.0, 40 pin GPIO header, 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless and microSD storage port on it. Raspberry Pi 4 being quite enough for the project made us choose the Raspberry Pi 4.

2.2.7. Battery

In this project it is decided to work up to 2 hours without charging the system. So, calculations for 2 hours have been done for 11Amps (including all components such as, Raspberry Pi, ESC, etc.) and for this 2-hour system needs a battery which is 12V and 19Ah.

This entire system uses a regulator, designed by us, feeding from a battery, providing five and three point five voltages can be seen in Figure 15.

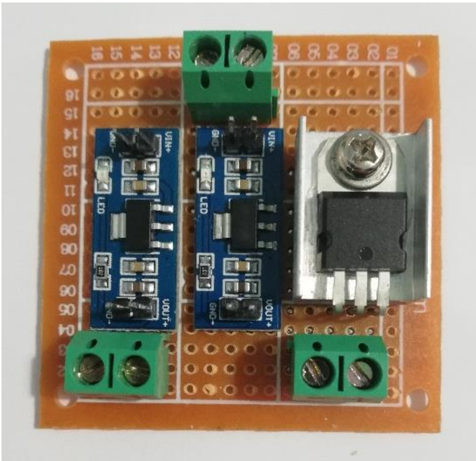


Figure 15: Regulator Circuit

2.2.8. Spring

For saving up on the energy and easy, smooth flexion movement it is planned to use halazone spring on the system. Implementation of any other springs such as torsional or compression was either blocks up too much space or does not have level of compression that system need. Calculations of spiral springs are pretty much like the compression springs so; MATLAB modelling of the spring was easy as well.

2.2.9. Frame of the Mechanism

To select the material that will be used in the frame of the mechanism SolidWorks simulation has been used. Stress, displacement, and strain tests has been applied with 150% nominal stress. As a result, the material that will be used in the frame has been selected as 3D print material Pet-C.

3. THEORETICAL BACKGROUND

3.1. Intention Determination Method

In this project, the intention determined by the angle information of knee obtained from the encoder and the force on the sole of the foot through the force sensors on the sole of the foot.

Electromyography technique can also be used in this project, but Electromyography was not preferred due to the difficulty of receiving signals from muscles and the plans regarding the exoskeleton mechanism that individuals with different muscle lengths and volumes can use.

Thanks to the foot sole force sensor, the position of the user's leg can be reached. The information from the pressure sensor under foot and the angle sensor on the knee joint is transmitted to the controller (Raspberry Pi 4) and the necessary electrical signals are transmitted to the DC motor placed on the knee joint. In this way, the leg muscles of the user are supported in the required amount.

To use the foot sole force sensor, it is useful to first examine the walking period and how much force is in which part of the foot sole. The force acting on the sole of the foot is given experimentally[19] in Figure 16.

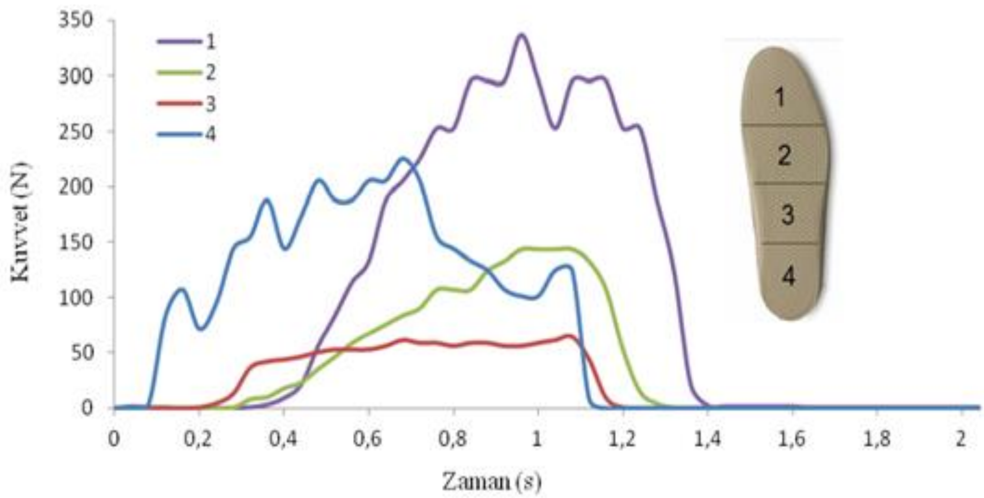


Figure 16: Force Acting on the Foot Sole Parts[19]

When the experimental data analysed, it is observed that the sole of the foot is examined in 4 sections. According to cell locations and cell number of the sensor it will used in this project, the foot sole can be evaluated in 2 or 3 parts.

Since the motor to be used on the knee joint in this project has an encoder, there is no need to use an extra angle sensor. The angle information of the knee joint will be obtained from the motor encoder.

Encoder: It is the name given to an electromechanical device that produces a digital electrical signal while a shaft is rotating. All movements and position information of precisely controlled machines are provided by encoders.

Two slit lines are used in encoders, one can be seen in Figure 17, for the direction of rotation. The signal coming from the upper slit is called A channel and the signal coming from below is called the B channel. As the shaft rotates, pulses are generated in these channels at a frequency proportional to the shaft speed. The phase difference between these two signals gives information about the direction of rotation.

The number of slits on the encoder is the resolution of the encoder. The more slits there are, the more precise angle information encoder outputs.

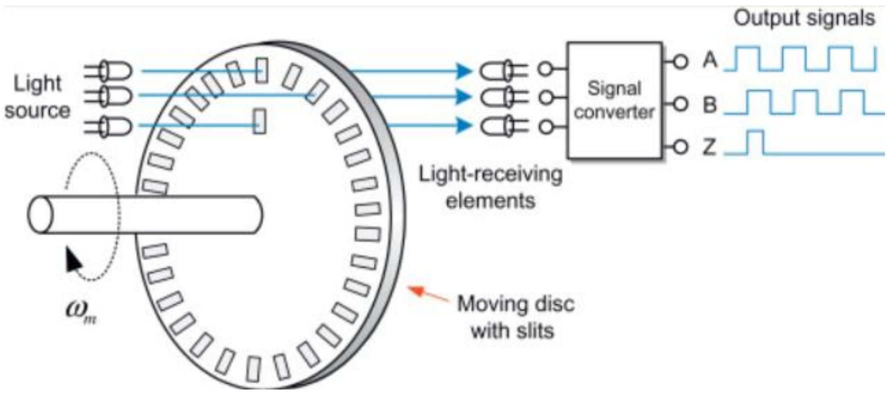


Figure 17: Encoder and its Output[20]

Encoders usually have a third channel. This channel, called the Z or index channel. This channel generates a pulse each turn and gives information about the number of rotations of the shaft. In this project, this channel is not needed since the engine will never turn full tour.

3.2. Control Method

Although the advance of many complicated control theories and techniques, most of the industrial processes currently are still controlled by Proportional-Integral-Derivative (PID) controllers[21]. This shows that this very simple control technique has great potential for fulfilling the needs for many industrial processes.

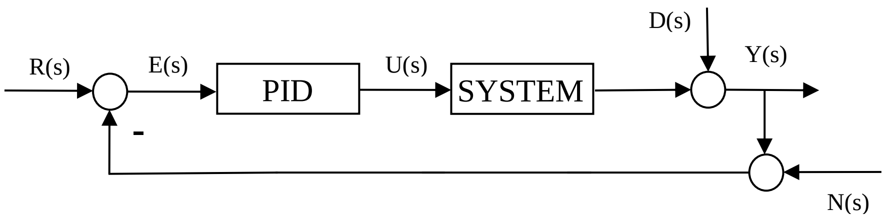


Figure 18: PID Control System

Where R(s) is reference signal, E(s) is error signal, U(s) is controlled signal, D(s) is disturbance signal, N(s) is noise signal and Y(s) is output signal.

The controller model can be written as:

$$U(s) = \left[K_p + \frac{K_i}{s} + K_d s\right] E(s) \quad \text{Eq.4}$$

PID controller must establish closed-loop stability, shape the dynamic and the static qualities of the output response [Y/R] and the disturbance response [Y/D], attenuate the effect of the measurement noise N(s)[21].

4. DESIGN

4.1. Mechanical Design

System has to be as light as possible for to control and the user’s ergonomy.

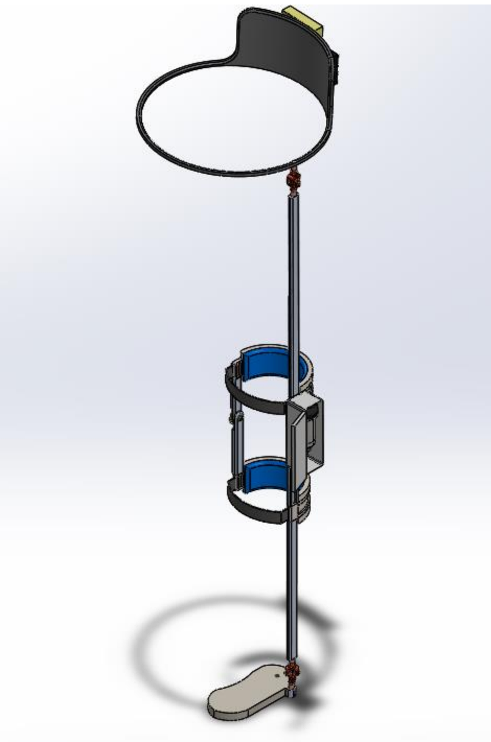


Figure 19: Entire Exoskeleton

Electronic system which includes; Raspberry Pi 4, Arduino and MC3992(ESC) controller on its belt can be seen in Figure 19.

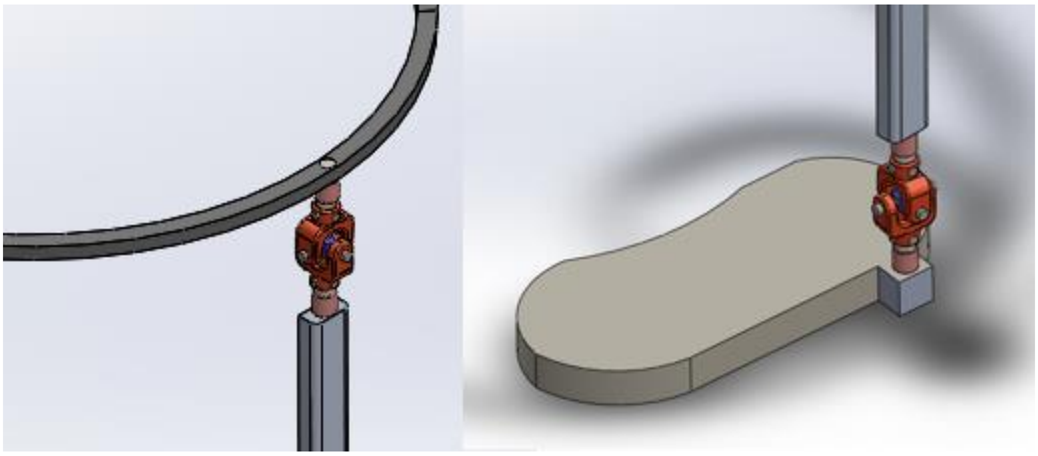


Figure 20: U - Joints on the Hip and Wrist

Hip and Writs has passive joints, can be seen in Figure 20, for additional mobility on the hip and wrist.

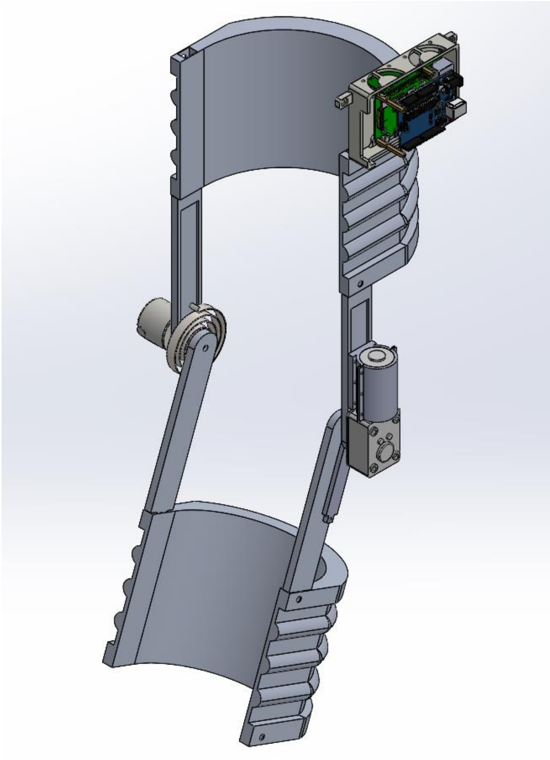


Figure 21: Knee Part of The Exoskeleton

Increasing surface area and addition of comfortable material to the touching areas of the knee part of the exoskeleton, which can be seen in Figure 21, have been done for improving ergonomics. Using C shaped die also increased inertia and lowered the weight of the system.

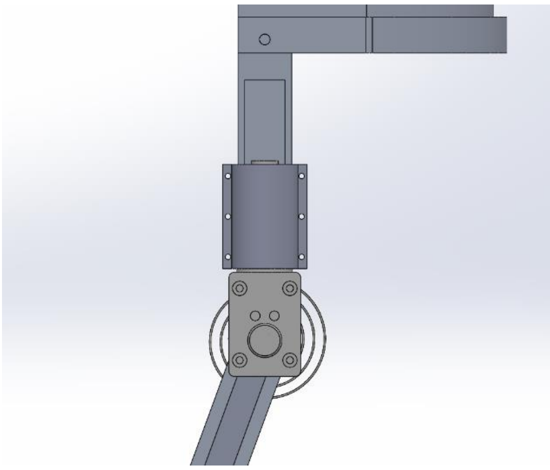


Figure 22: Motor and Reducer on the Exoskeleton

In Figure 22, you can see the 5840-31ZY, L type worm gearbox. Having L type gearbox helped with efficiency and space because if the gearbox was I type changing the axis would require another bevel gear.

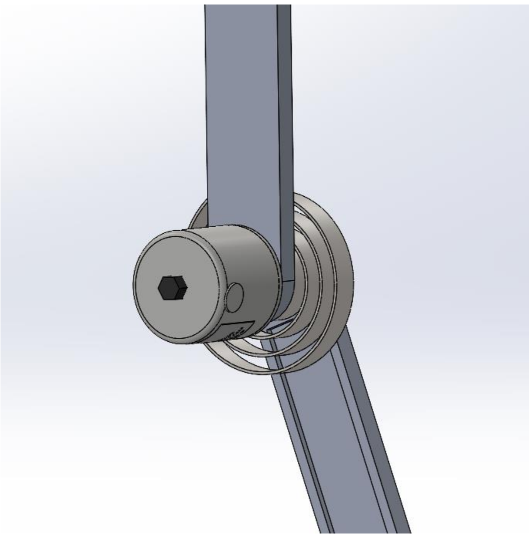


Figure 23: Spring and Encoder on the Exoskeleton

Selected motor does not include encoder so as a solution we used motor at the opposite site of the encoder. Since it is an optic encoder it does not affect the precision of the data output. Encoder's shaft also helps with the attachment of the spring, can be seen in Figure 23, to the system.

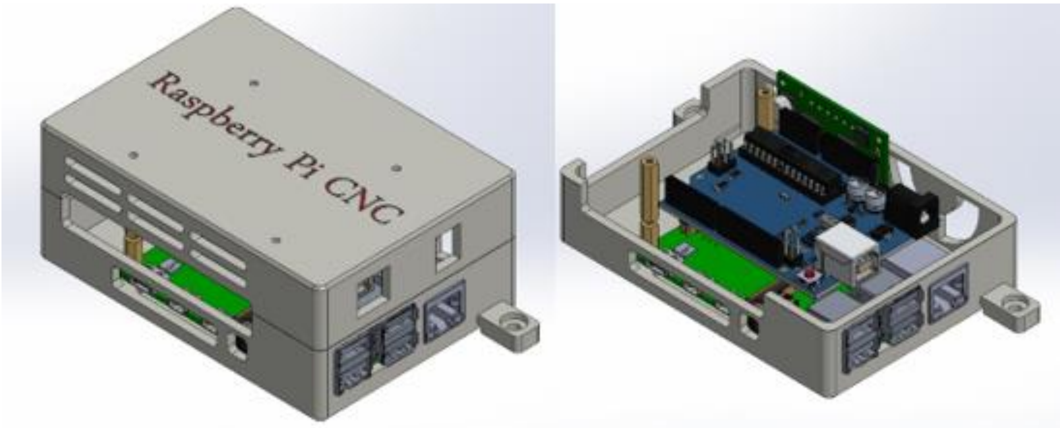


Figure 24: Electronic Box

Raspberry Pi 4, Arduino and MC33926 will be in electronic box, can be seen in Figure 24.

4.2. Electronic Schematic

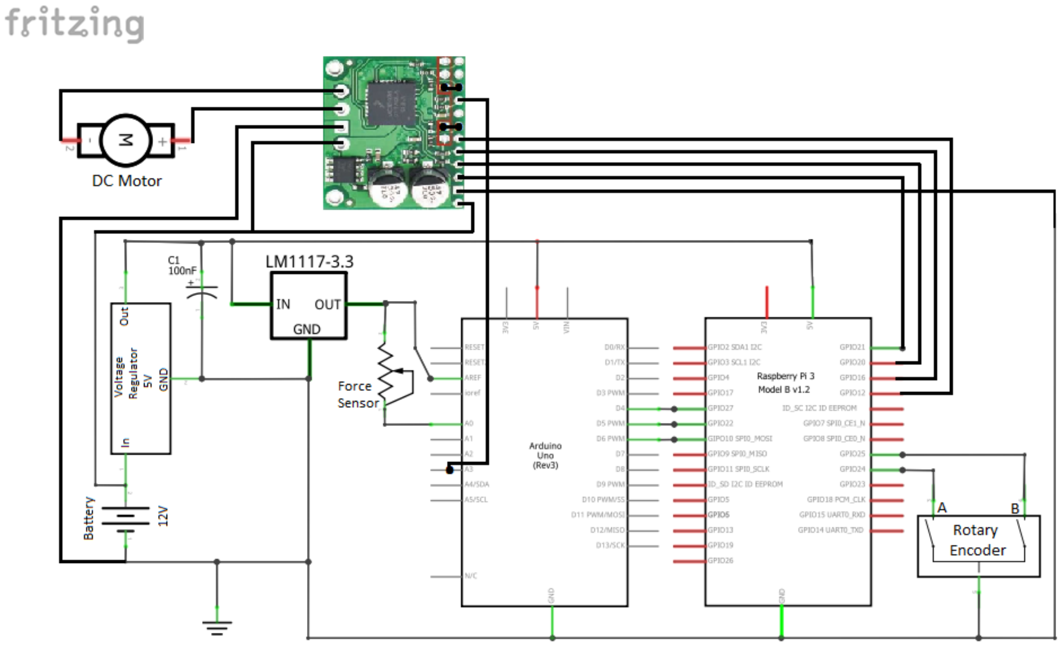


Figure 25: Electronic Schematic

Electronic schematic of the system can be seen in Figure 25. System has two sensors and one actuator. Force sensor will be placed under user's foot. Force sensor is for determination of intention and the rotary encoder for checking if the motor has been driving in a desired position and speed. As for controller Raspberry Pi 4 is used and Arduino is used for analog to digital signal conversion. 12V Battery is used to feed the system. 5V Voltage regulator is required for the Arduino, Raspberry Pi and BEC circuit of the speed controller. 3.3 Voltage regulator is needed for force sensor.

4.3. Mathematical Model

For deriving mathematical model, Simplify the system as just DC motor and add external torque produced by spring to the system as in Figure 26. [22][23][24].

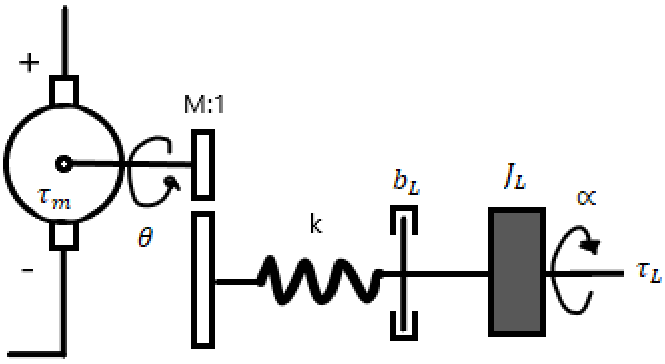


Figure 26: Mathematical Model of the System

$$J\ddot{\theta} + b\dot{\theta} = \tau_m \quad \text{Eq.5}$$

$$J = J_m + M^2 J_L \quad \text{Eq.6}$$

$$b = b_m + M^2 b_L \quad \text{Eq.7}$$

$$\propto = M\theta \quad \text{Eq.8}$$

Work done by the load shaft equals to work done by the motor shaft as shown in the Eq. 9;

$$\tau_L \propto = \tau_m \theta \quad \text{Eq.9}$$

Where, J, can be seen in Eq. 6, is the equivalent moment of inertia of the system, J_L is the loads moment of inertia, J_m is the motors moment of inertia, b, can be seen in Eq. 7, is equivalent friction coefficient, b_L is the loads friction coefficient, b_m is the motors friction coefficient, ∝ is the angle of the load shaft, θ is the angle of the motor shaft, M is the gear reduction ratio: $M = \frac{\propto}{\theta}$.

If the Laplace transformation of Eq. 5 is taken, it is obtained

$$(Js^2 + bs)\theta = \tau_m \quad \text{Eq.10}$$

If τ_m(s), can be seen in Eq. 10, taken as input and θ(s), can be seen in Equation 8, as output, the transfer function is obtained as in Eq. 11.

$$G(s) = \frac{\theta}{\tau_m} = \frac{1}{Js^2 + bs} \qquad \text{Eq.11}$$

Since system will be getting load shaft angle information, it is necessary to express transfer function as load torque and load angle form. To develop this function, Eq. 8, Eq. 9 and Eq. 10 is taken and combined in one equation, which is Eq. 12.

$$G(s) = \frac{\propto}{\tau_L} = \frac{M^2}{Js^2 + bs} \qquad \text{Eq.12}$$

Thus, the transfer function relating the desired torque input τ_L(s) to the output angle $\propto$(s) is a second-order system.

4.4. Controller Design

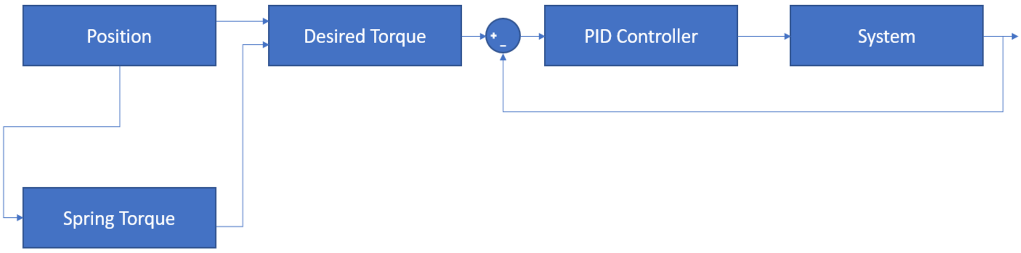


Figure 27: Controller Diagram

Figure 27 shows controller diagram of the system based on the transfer function. MATLAB-Simulink model can be found in Appendix B1. For transfer function constants and PID constants script's check Appendix B2 and Appendix B3.

System's desired input torque function is depending on where the lower leg is. So, in Figure 28 you can see the free body diagram of the system.

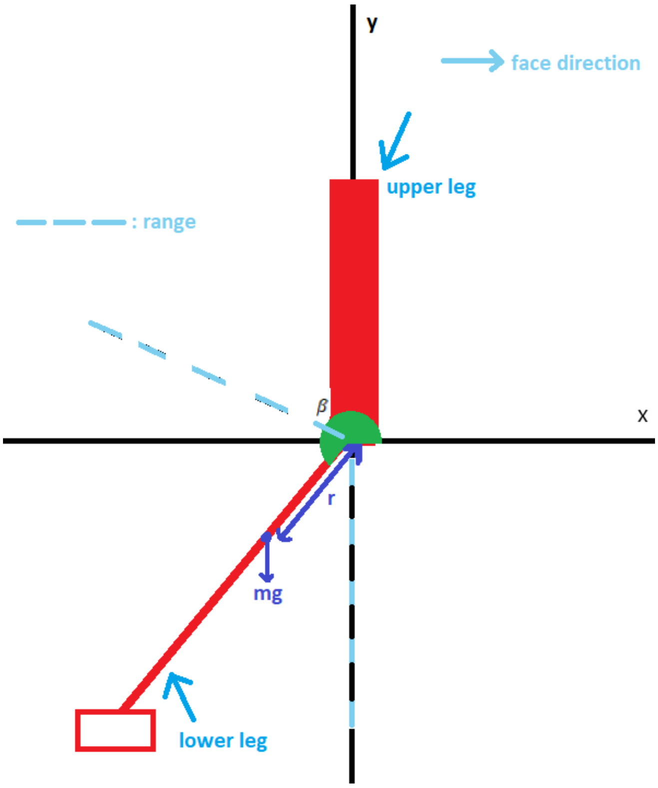


Figure 28: Free Body Diagram of the System

Where, ß is the position of the lower leg, angle, r is the distance between knee and the centre of lower leg mass and mg is the force of gravity. So, the maximum torque needed can be found where ß is -180°, can be seen as in Eq. 13 and Eq. 14.

$$\tau_{max} = mgr\cos\beta \qquad \text{Eq.13}$$

$$\tau_{max} = 4.8 \times g \times 0.15 \times \cos(-180) = -7.06\ Nm \qquad \text{Eq.14}$$

With the 50% assisment rate and 50% tolerance, needed torque can be calculated as in Eq. 15.

$$\tau_{needed} = \tau_{max} \times 50\% \times (1 + 50\%) = -5.3\ Nm \qquad \text{Eq.15}$$

With spring torque added to system the desired torque function can be found at Appendix B4.

There is another function called alfa that gives current angle and angle an instant before as output used as to determine the direction of the lower leg. And for the simulations sake a function that raises and lowers leg in a second periodically has created. Function can be seen in Appendix B5.

For the spring torque it is calculated as in Eq. 16.

$$\tau = \frac{\pi E b t^3 \theta}{6L} \qquad \text{Eq.16}$$

Where, E is the modulus elasticity of the material, θ is angular deflection in revolutions, L is the length of the active material, b is the material width, t is the material thickness and τ is the torque.

Spring Torque Function can be found in Appendix B6.

5. DESIGN VERIFICATIONS

5.1. Mechanical Simulation

Simulations has been done on Solidworks. The stress applied to the pieces are %50 more than normal to make it more stable. Choice of material Aluminium LM24 prove that the material is robust and light enough for the system.

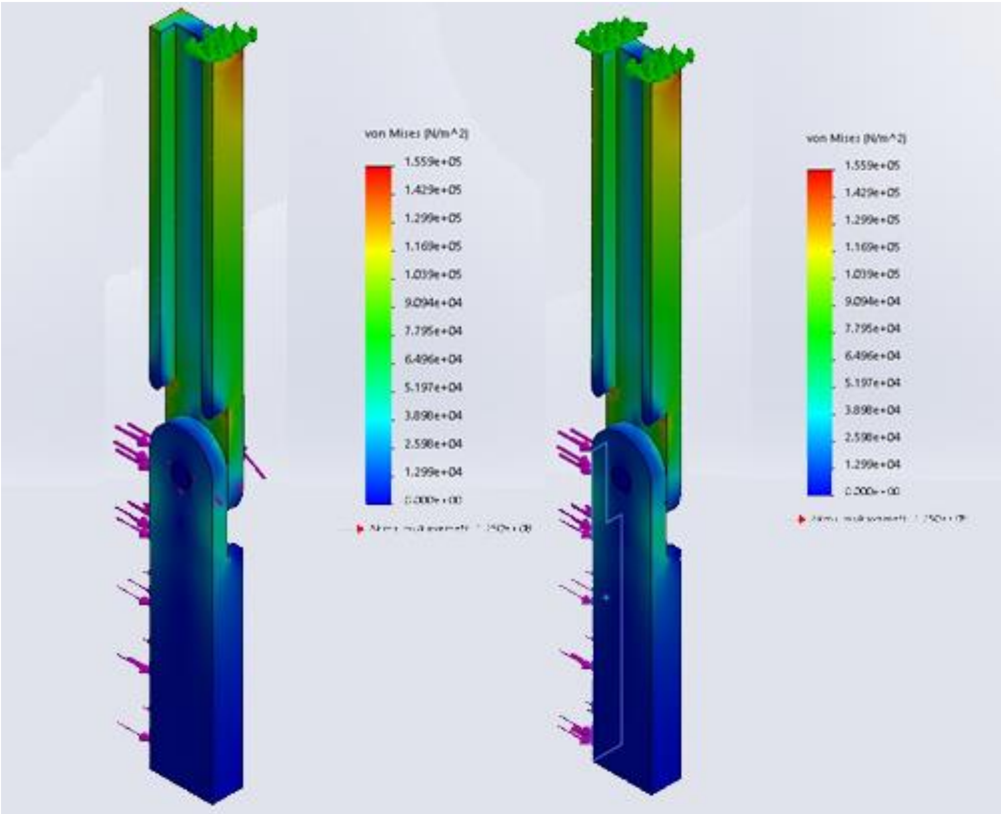


Figure 29: Main Frame Mechanical Simulation

As you can see in the Figure 29, stress on the main frame stayed under the critical level. In Figure 30, you can see the stress on the bevel gears shaft which is likewise under the critical level as wanted.

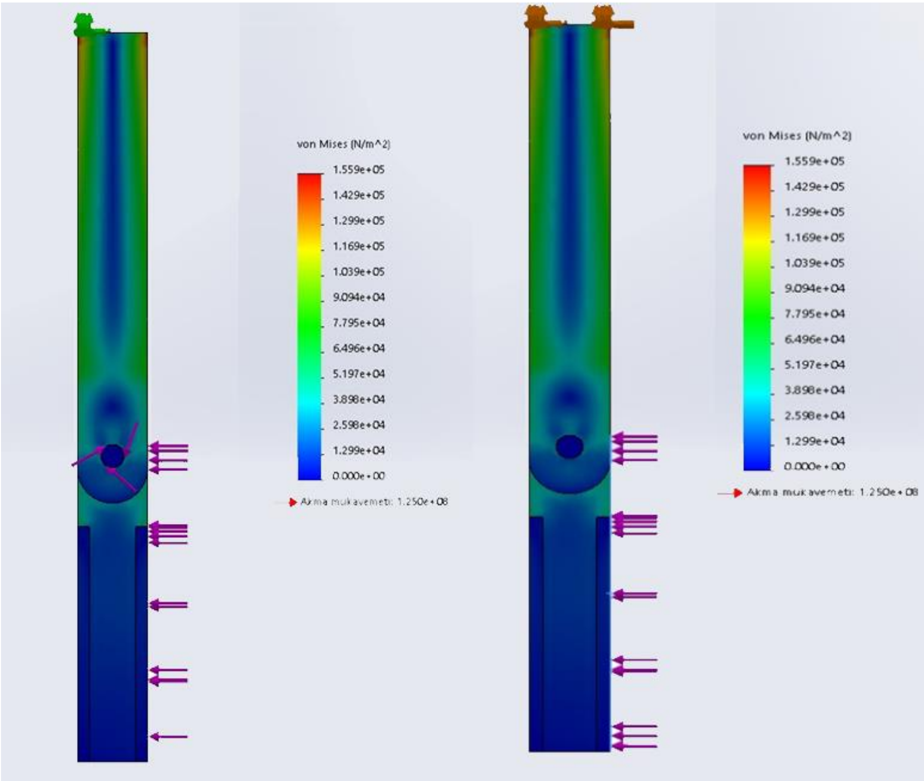


Figure 30: Bevel Gear Mechanical Simulation

5.2. Controller Simulation Output

For a 2 second periodic gait cycle which means 1 second extension, 1 second flexion, position of the leg can be seen in Figure 31.

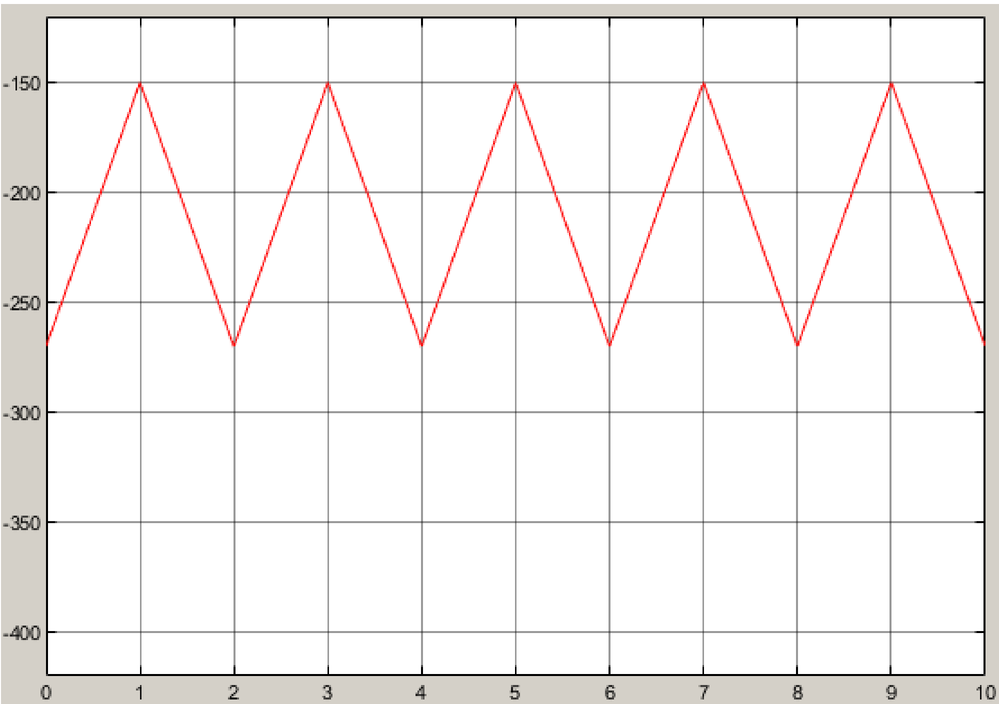


Figure 31: Position of the Leg

After the calculations via obtained from the position and direction information, controller transmits motor the desired torque. In Figure 32, you can see the torque produced by motor.

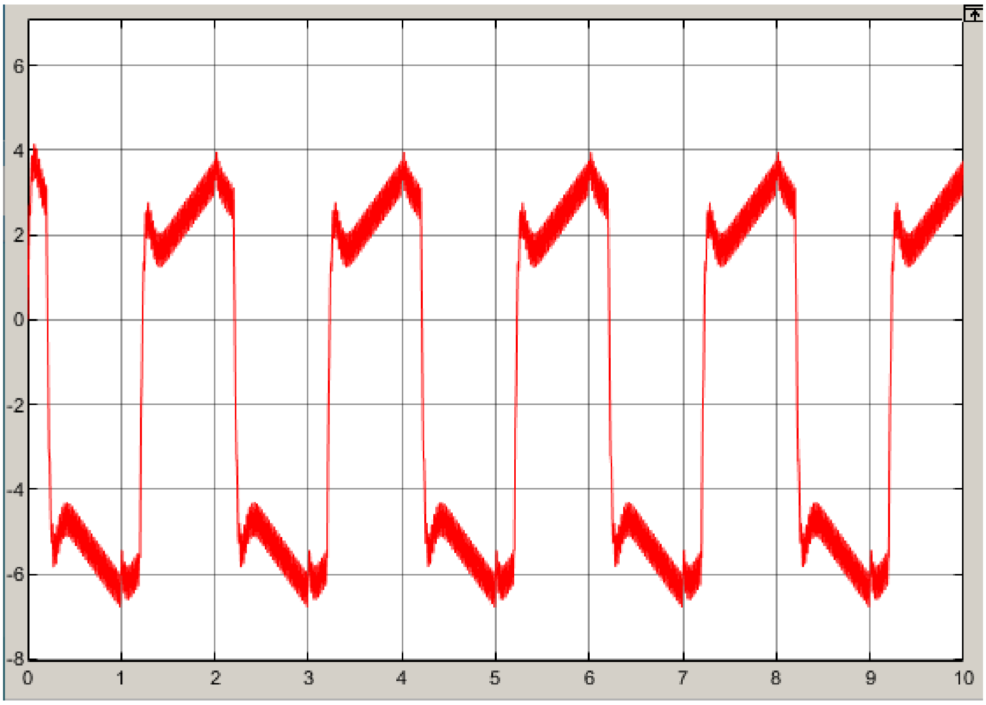


Figure 32: Torque Produced by the Motor

Depending on the position of the leg spring produces torque. In Figure 33 you can see the torque produced by spring.

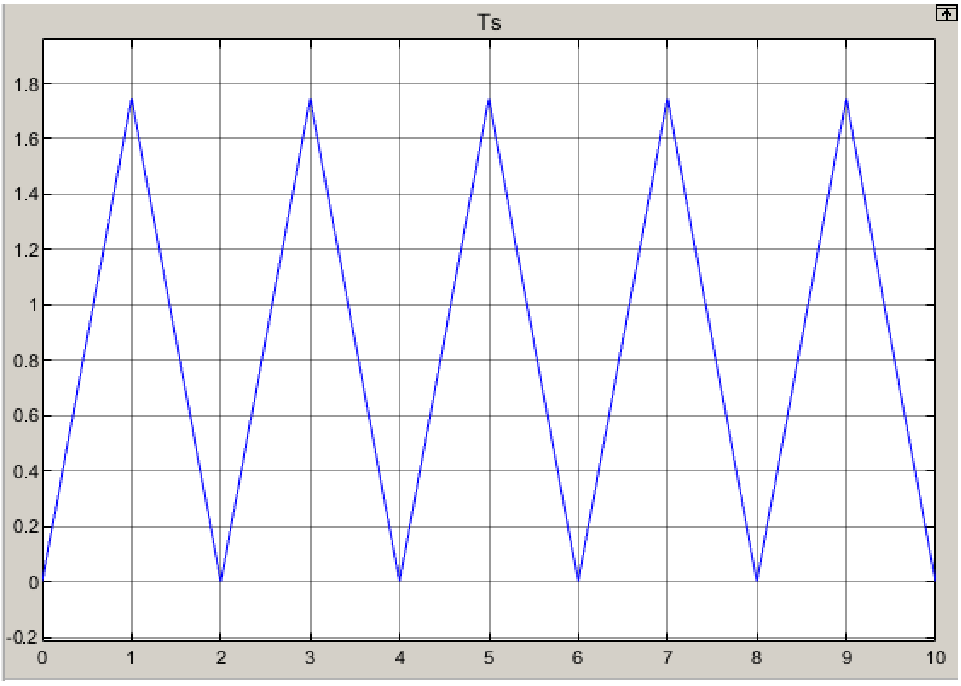


Figure 33: Torque Produced by the Spring

In Figure 34, you can see output torque which is the sum of motor and spring torque.

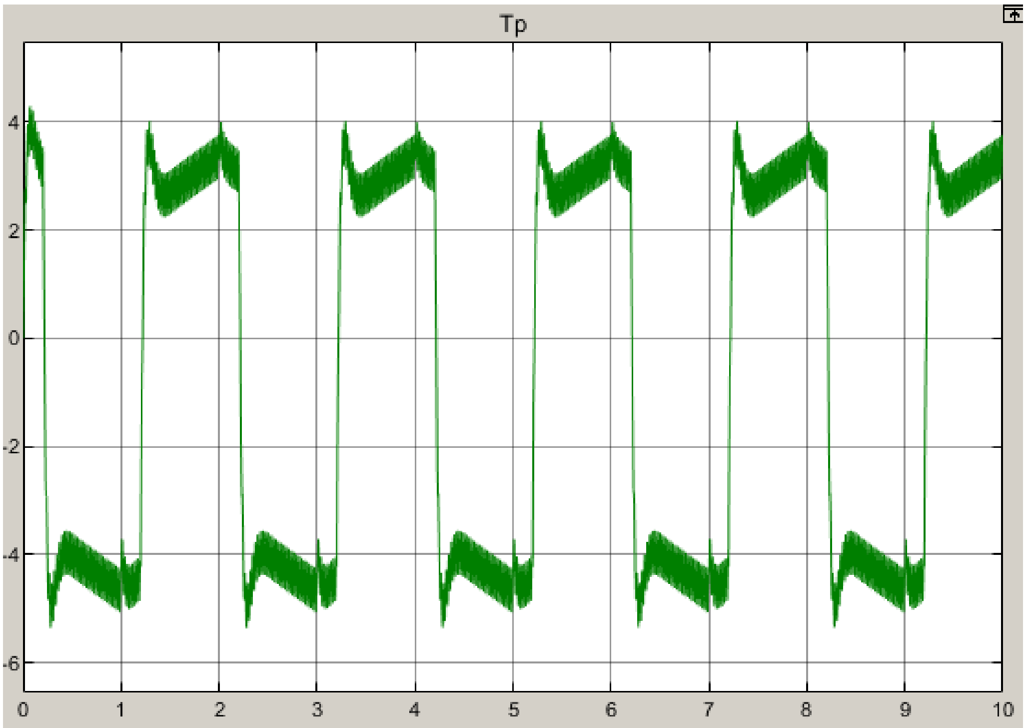


Figure 34: Output Torque

In figure 35, you can see the output torque and the spring torque in the same figure.

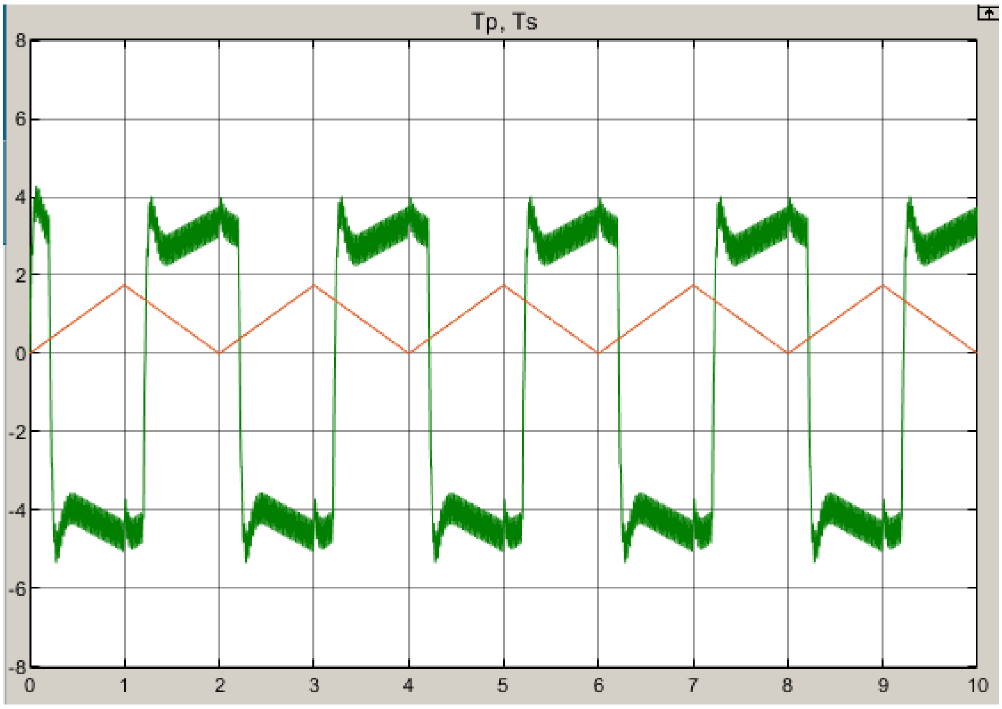


Figure 35: Torque Showcase

5.3. Entire System Simulation

You can see the project's Simulink model in the Figure 36. This model is created for the Simscape simulation. Models consist of the exoskeleton and Motor-Controller, Spring blocks.

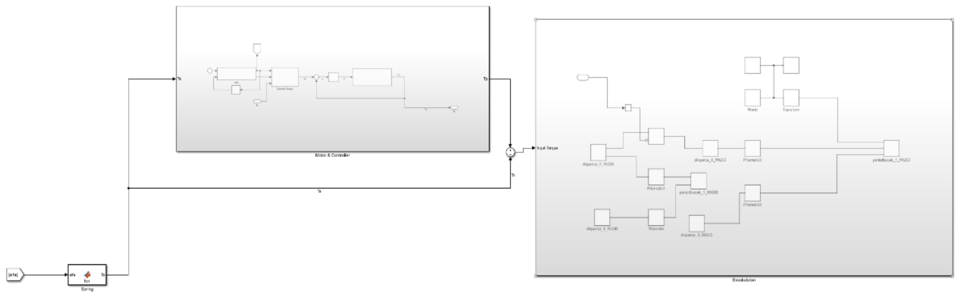


Figure 36: Simulink Model of the Entire System

Breakdown of the Exoskeleton block and Motor-Controller block can be seen on Appendix B7 and Figure 37. With generated input from the Motor-Controller and Spring blocks it is simulated for exoskeleton's trajectory and speed. Further details and codes can be seen in Appendix B2.

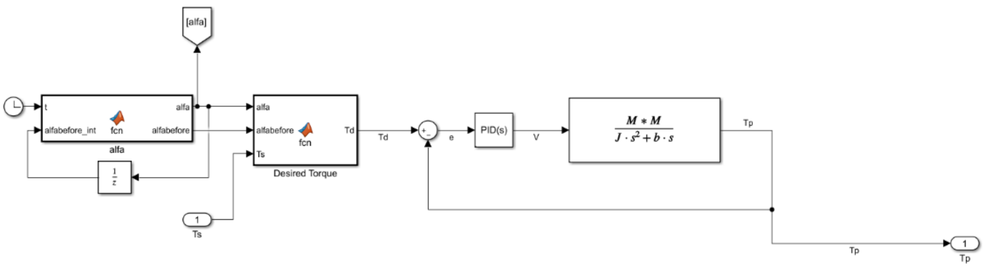


Figure 37: Motor Controller

6. DESIGN IMPLEMENTATION

6.1. Mechanical Implementation

The body material as mentioned chosen as Pet-C. Choosing this material instead of aluminum lightened the project and for further alleviation C-profile has been used all throughout the frame. The total mechanical system weight is 0.15 kg. The entire system is 0.55kg. Mechanical system shown in figure 38 and figure 39. Initial design of the project has changed to lighten the exoskeleton and ease up the control can be seen in figure 37.

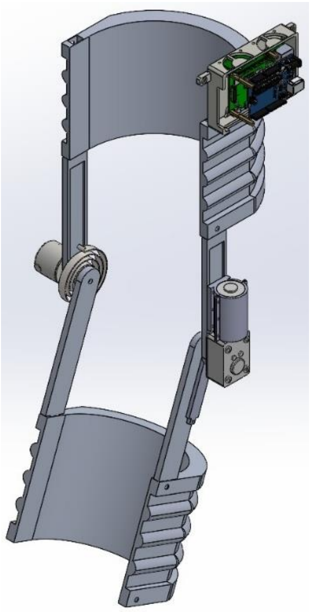


Figure 38: Implemented Design

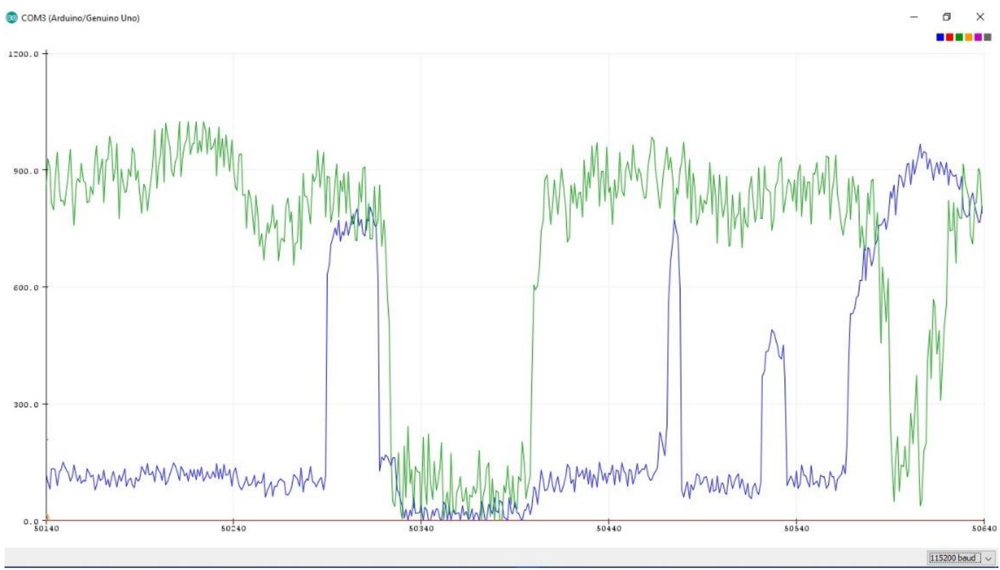
Jk

6.2. Electronical Implementation

6.2.1. Force Sensor

The sensor board was removed from the sensor and a cable is added between the sensor and the sensor card. A soft material was is under the sensor to absorb some of the leg pressure.

The sensor board was mounted on the side of the slipper as seen in Figure X. Headers were placed on the sensor ends for easy assembly and disassembly. Socket-end cable is connected to sensor boards. The data acquired from sensor as test subject walking can be seen in Figure X.



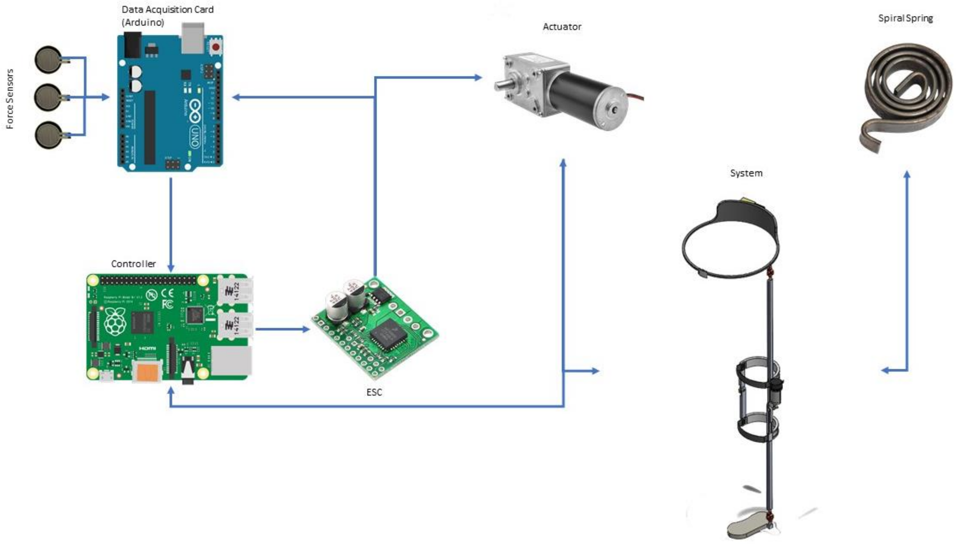
6.2.2. Arduino as ADC

The Arduino was used just for converting the signal from analog to digital in this project. Four of analog inputs were used. A0 and A1 inputs are for force sensors. These inputs were converted to digital as 3 bits. A2 input is for potentiometer which is converted to digital as 5 bits. This input was used to test the mechanism as an encoder. A3 input is used for current feedback of the motor driver to be used in PID torque control of the motor.

The Arduino code that is used can be seen in Appendix B8.

6.2.3. Entire Electronic System

The force sensors are placed on the slipper to perceive the force applied from user's foot. The signals from the sensors are converted to digital signal on Arduino Uno. This digital signal is transferred to Raspberry Pi 4 which evaluates the signals from the force sensors and gives the necessary commands to the motor driver. The motor driver drives the motor to the desired position. The spiral spring is mounted on the knee joint to increase the efficiency of the system. It supports the motor and reduces the energy usage. The entire electronical system can be seen in Figure X.



Entire system is tested as you can see in Figure X. Users wear the slippers and pressure of the foot sole is determined via force sensors on the slippers. These analog sensor signals are sent to Arduino to convert to digital signals. And then these digital signals are used on Raspberry Pi 4 to determine the user’s intention.

Moreover, a potentiometer was used as an encoder to determine the position of the motor on this test mechanism. When the joint of mechanism turns, this potentiometer also turns. To get the position information, a simple code is written. The motor driver gets the desired PWM signals and drives the motor with this PWM signals as user’s desire.

This entire system uses a regulator, designed by us, feeding from a battery, providing five and three point five voltages.

6.3. Control Implementation

6.3.1. Controller

Raspberry Pi 4 is used to control the system with the software language Python 3. Since the Raspberry is a flexible and compact microcontroller, it contains systems’ main functions. Motor Control Algorithm and Intention Determination are done in Raspberry Pi 4.

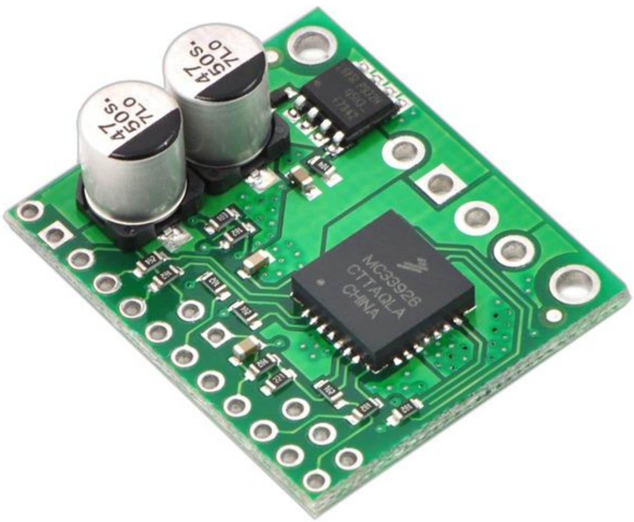
6.3.2. Motor

The motor which is 5408-31ZY can be seen in Figure X. It provides up to 10 Nm at 5 A continuous and 15 RPM speed. Also, it has 0.36 kg weight. It has L-type gearbox integrated.

And you can see the breakdown structure of the motor in Figure X. Its L-type gearbox with the high torque output was the key factor to be chosen.

6.3.3. Motor Driver

For motor driver, MC33926 is used, can be seen in Figure X. It has an operating range of 5V to 28V and can deliver 3A continuously, 5A peak current to a DC motor. The MC33926 supports ultrasonic (up to 20kHz) PWM and features current feedback. IN1, IN2, D2, FB, EN pins are used for motor direction control, PWM, current-feedback and enabling respectively.



6.3.4. PID Control Method

Raspberry Pi Python PID code can be seen in Appendix B9.

7. IMPLEMENTATION ANALYSIS

8. ASSUMPTIONS AND ERROR ANALYSIS

8.1. Assumptions

In this design following assumptions are made:

- In contrast of the other exoskeleton knee mechanism designs for helping full disabled individuals, this mechanism is design to assist partially disabled individuals.
- The exoskeleton is designed and manufactured in the consideration of its adult users.
- From exoskeleton motions, the abduction and adduction movements are disabled for the demo version to avoid its effect on flexion and extension movements which is the main task of the exoskeleton system.
- The system is assumed to be working on obstacle-free, plain road.
- Maximum weight of the user is determined as 80 kilograms and height limits are 175 to 195 centimetres. Upper limit for assist rate is 50%.
- Foot size of the user assumed to be 43.

8.2. Error Analysis

The weight of the system can be reduced by using different dies. Thus, position control can get easy and the load on the motor can be lighten. Low load on the motor means more lifetime of the gearbox and the motor. Force sensors might not fit every user's foot in same way so different models for different foot sizes can be made. Using the exoskeleton on the slightly sloped road might harm the motor or overload the motor.

9. RISK ANALYSIS

It is deeply analysed the project with the help of the books A Guide to the Project Management Body of Knowledge [25] and The PMBOK Guide - A Project Manager's Book of Forms [26]. Risk Register is prepared, can be seen in the Table 3.

9.1. Risk Management Plan

- Strategy:
- Methodology:
- Risk Categories:
- Risk Management Funding:
- Contingency Protocols:
- Frequency and Timing:
- Stakeholder Risk Tolerances:
- Risk Tracking and Audit:

Table 3: Risk Register

ID	Risk Statement	Probability	Impact				Score (1-5)	Response
			Scope	Quality	Schedule	Cost		
1	Covid-19 shutdowns	HIGH	LOW	MEDIUM	HIGH	LOW	4	Doing work packages that can be done instead
2	Increase in exchange rate	HIGH	MEDIUM	MEDIUM	LOW	HIGH	4	Searching for new sponsors
3	Lack of Knowledge	MEDIUM	LOW	MEDIUM	MEDIUM	MEDIUM	4	Taking some courses in free time
4	Inability to come together	MEDIUM	LOW	MEDIUM	MEDIUM	LOW	3	Doing work packages that is not needed to come together
5	Delay in supply	MEDIUM	LOW	LOW	MEDIUM	LOW	2	Go to the market to get physically if possible
6	Technical Malfunctions	LOW	LOW	LOW	MEDIUM	HIGH	4	Try to fix or refund and buy a new one
7	Health Problems	LOW	LOW	MEDIUM	HIGH	LOW	4	Tolerance in duration estimate
8	Insufficient Budget	LOW	MEDIUM	MEDIUM	MEDIUM	LOW	4	Tolerance in budget estimate

9.2. Taken Measures for Predictable Risks

- Tolerance is given when budget and duration planning
- Semester holiday is used to work on the project
- Jobs that needs physical contact have been rushed
- Gotten in contact with an alternative supplier
- Some courses is taken to finalize the project successfully
- In case of budget deficit, new sponsors is searched

10. WORKING PLAN

10.1. Schedule of the Project

You can see the predicted working plan in Table 4. There can be unexpected situations, so team members try to go ahead of the plan. Every member is reporting their work at every weekend. So, there will be no wasting time on reporting.

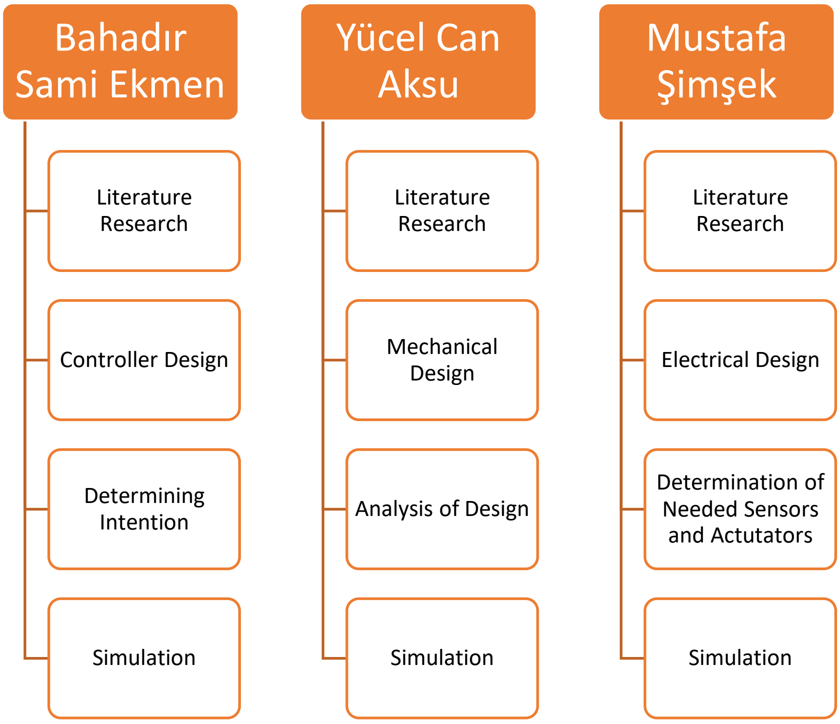
Table 4: Schedule of the Project

Activity	Duration									
	October	November	December	January	February	March	April	May	June	July
Literature Research and Conceptional Design										
Modelling and Designing of Knee Mechanism										
Designing Electronical System										
Designing and Experimenting of Controller										
Simulation										
Prototype Calculations										
Manufacturing and Getting Necessary Components										
Assemblying										
Circuit Board and Sensor Implementations										
Determining, Fixing Problems										
Finishing Documenting										

10.2. Task Distribution

Basically, main areas of the project are determined as mechanic, electronic and control. Members consult each other most of the time. Because it is believed that other team members possibly know something that do not known by another member or can give different idea about that matter. Beyond of believing, some of team members has experience on certain subjects. But of course, members want to work on certain specialities, and it is agreed on such a task distribution that can be seen in Table 5.

Table 5: Task Distribution



11. BUDGET

11.1. Cost of the Project

The cost of the project can be seen in Table 6.

Table 6: Cost of the Project

Product	Explanation	Unit Price	Quantity	Total Cost
5840-31ZY	Motor	450 ₺	1	450 ₺
Keyeye Studio FlexiForce	Force Sensor	75 ₺	4	300 ₺
OVw6-06-2HC	Encoder	165 ₺	1	165 ₺
Springs	-	35 ₺	2	70 ₺
Raspberry Pi 4	Controller	500 ₺	1	550 ₺
Arduino Uno	Data Acquisition	50 ₺	1	50 ₺
MC33926	ESC	200 ₺	1	200 ₺
Li-Po Battery	-	1000 ₺	1	1000 ₺
Pet-C	Body	1500 ₺	7	1500 ₺
Total				**4285 ₺**

11.2. Financial Support Applications

Project has applied TÜBİTAK, its logo can be seen in Figure X, 2209-B and got accepted.

And applied to TeknoFest, its logo can be seen in Figure X, and passed the preliminary test.

12. RESULTS, SUGGESTIONS AND FUTURE WORKS

12.1. Results

The result of this project is a demo version of an assistive exoskeleton. In this demo version there are couple of things to be set. Starting with human tests to analyse the data. Assembly of the motor and the main exoskeleton body and implementation of an Li-Po battery so that product do not have to be plugged to a wall. The errors and values that gathered from simulations on Simscape, MATLAB or Solidworks will be different than the real-life values and errors. It can be seen on the force sensor very clearly. The reason is that non-linearities such as friction, which is included in simulations, moist, temperature and the list goes on. Therefore, at the beginning of the project it is decided to use at least 20% of tolerance on any component. This tolerance almost worked for every aspect of the project but the spring as weight wise over the tolerance this still did not affect the movement but lower the efficiency.

12.2. Summary, Suggestions and Future Works

In this thesis, an assistive exoskeleton knee mechanism is designed and implemented. The main purpose of the leg mechanism is to support individuals with walking difficulties. Which can be caused by trauma, orthopaedic or neurological disorder or old age. For control of the leg mechanism "Computed Torque Control" method has been used.

The thesis consists of the mechanical design of the exoskeleton mechanism, mathematical model equations of body and motion, 3D modelling, MATLAB modelling, controller design, simulations of every aspect and implementations.

Mechanical design of the exoskeleton was performed in Solidworks. The stress analysis of exoskeleton's critical parts also done in Solidworks with the assumption of 10 N.m torque is being applied to pins and clamp ends. Equation of motions are represented in singular pendulum model that represents the knee and the leg. This equation has been driven and MATLAB Simulink model has created. Also, Solidworks 3D model of the exoskeleton has been transferred to Simulink Simscape multibody environment. PID computed torque controller has been designed to control both models. Both models are fed with desired torque and walking trajectory. Design and coding errors found by these simulations. The force sensors have used for intention determination. Dependent force levels on the different parts of the foot sole are the main idea for the intention determination.

Designed exoskeleton can be used by individuals with different body types and body sizes. From 175 cm to 195 cm height and up to 100 kg weight. Based on this material selection has been made. Aluminium AL24 was thought at first but the weight and the cost of the materials was over the limits. So, considering the Solidworks verifications with tolerance chosen material is Pet-C for the body. Material is light, easy and fast to produce.

The actuator selection has been made according to torque values obtained by the mathematical model calculations. At first, it has chosen as Maxon EC-90 Flat as a main actuator but in real-life distributor of the product has given us a price doubling the entire budget. So, project needed a cheaper option. Worm gearbox motors have love rpms but high torques. Low RPMs was not a problem for the project. So, choice was 5840-31ZY motor which can give up to 10 Nm of torque and 15 RPM at max.

The thesis consists of all calculations, simulations, errors, and implementations of the project.

In conclusion, a stable control and electronic mechanism that can follow the given trajectory, durable exoskeleton mechanisms have been designed. Unfortunately, at the time project is not fully assembled. The control system can support 3 movements. Sitting down, getting up and walking. The force sensor system can measure with little noise which is filtered and perfected for intention determination. The resultant product to be made of this project can be considered as a demo version of an exoskeleton knee mechanism.

The real-life test and simulation results for both models verify the design and give confidence for the future works. Planned future works for the project as follows: Firstly, assembly of the control system and the mechanical system must be done. In addition, for better and compact implementations of the electronical systems a mechanical system is going to be designed. All electronic equipment is going be placed to the waist mechanism designed. Li-Po batteries are going to be used instead of a power adapter. Finally, mechanical improvements such as reducing the mechanism weight or buying better motor might increase the system performance and reduce the errors.

REFERENCES

1. TÜİK. (2010). Engellilerin Sorun ve Beklentileri Araştırması.
2. C-J Yang, J-F Zhang, Y Chen, Y-M Dong and Y Zhang. (2008). A review of exoskeleton-type systems and their key technologies. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science. 222: 1599.
3. Duschau-Wicke, Alexander & Brunsch, Tom & Lünenburger, Lars & Riener, Robert. (2008). Adaptive Support for Patient-Cooperative Gait Rehabilitation with the Lokomat. 2357 - 2361. 10.1109/IROS.2008.4650578.
4. Robertson A. (2013). A look back at GE's decade-spanning search for a man-powered robot suit. Retrieved from https://www.theverge.com/2013/8/16/4627702/a-look-back-at-ge-robotic-exoskeleton-program
5. Kanagawa Institute Technology. (2005). Retrieved from https://www.kait.jp/english/index.html
6. Berkeley Robotics & Human Engineering Laboratory. (2010). HULC. Retrieved from https://bleex.me.berkeley.edu/research/exoskeleton/hulc/
7. H.Kazerooni. (2004). "Berkeley Lower Extremity Exoskeleton (BLEEX)". UCBerkeleyNews.
8. Japan Ministry of Defense. (2019). LTU-LEE.
9. ekso BIONICS. (2011). EKSO.
10. Heedon Lee, Wansoo Kim, Jungsoo Han, and Changsoo Han. (2012). The Technical Trend of the Exoskeleton Robot System for Human Power Assistance. International Journal of Precision Engineering and Manufacturing. Vol. 13, No. 8, 1491-1497.
11. CYBERDYNE. (2004). HAL.
12. Otto Bock. (2020). C-Brace. Retrieved from https://shop.ottobock.us/Orthotics/Custom-Orthotics/KAFO-KO---Knee-Ankle-Foot-Orthosis-Knee-Orthosis/C-Brace/c/4036
13. Otto Bock. (2020). C-Leg. Retrieved from https://www.ottobock.com.tr/tr/protezler/lower-limb/solution-overview/knee-joint-c-leg/

14. Sensor Products Inc.. (2006). Foot Mapping Sensor System. Retrieved from https://www.sensorprod.com/foot-plate-pressure-sensor.php
15. OGR34. (2015). Knee Speed/Velocity Split for damper. Retrieved from https://www.eng-tips.com/viewthread.cfm?qid=386103
16. NASA. (2020). Human Performance Capabilities. Retrieved from https://msis.jsc.nasa.gov/sections/section04.htm
17. İ. Mürüvvet. (2011). Bir Alt Ekstremite Ortezinin Kinetik ve Kinematik Analizi. Dokuz Eylül University.
18. Keyestudio Thin Film Pressure Sensor Datasheet. Retrieved from https://wiki.keyestudio.com
19. Toygar E. M. , Özkurt A. , Kıral Z. , Çakmakçı M. (2012). İnsan Bacak Hareketleri İçin Prototip Dış İskelet Robotik Sisteminin Mekanik Tasarımı ve Hareket Verilerinin Yapay Sinir Ağları ile Elde Edilmesi. Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi. p. 234-248.
20. Ağustoslu Şafak. (2020). Arduino, PIC ve diğer gömülü sistemlerle Enkoder Kullanımı. Retrieved from https://www.mikrobotik.com/wp2/2020/09/24/arduino-pic-ve-diger-gomulu-sistemlerle-enkoder-kullanimi/
21. Johnson A. Michael, Moradi H. Mohammad. PID Control: New Identification and Design Methods. London. Ian Kingston Publishing Services. 2005.
22. Schilling J. Robert. Fundamentals of Robotics: Analysis and Control. 2003, p. 265-268.
23. Juanjuan Zhang, Chien Chern Cheah, Steven H. Collins. Experimental comparison of torque control methods on an ankle exoskeleton during human walking. 2015 IEEE International Conference on Robotics and Automation (ICRA). 2015, p. 5584-5589.
24. Lynch, K., Park, F., “Modern Robotics: Mechanics, Planning, And Control”, 2017, p. 403.
25. PMI. (2017). A Guide to the Project Management Body of Knowledge – Sixth Edition.
26. Cynthia Snyder Dionisio. (2017). A Companion to the PMBOK Guide – A Project Manager’s Book of Forms – Third Edition.

APPENDICES

Appendices A

Appendix A 1: Encoder Specifications

GENERAL INFORMATION	
Counts per turn	1024
Number of channels	2
Line Driver	Yes
Max. electrical speed	56000 rpm
Max. mechanical speed	5000 rpm

TECHNICAL DATA	
Supply voltage V_{CC}	5.0V ± 10.0%
Output signal	Incremental
Driver used logic	Differential, CMOS
Output current per channel	-4...4 mA
Signal rise time	100 ns
Measurement condition for signal rise time	CL=25pF, RL=1kOhm
Signal fall time	100 ns
Measurement condition for signal fall time	CL=25pF, RL=1kOhm
Min. state duration	125 ns
Direction of rotation	A before B CW
Typical current draw at standstill	15 mA
Max. moment of inertia of code wheel	65 gcm²
Operating temperature	-40...+100 °C

Appendix A 2: Motor Driver's Specifications

Dimensions

Size:	1.00" x 1.20"[1]
Weight:	0.14 oz[1]

General specifications

Motor driver:	MC33926
Motor channels:	1
Minimum operating voltage:	5 V[2]
Maximum operating voltage:	28 V[3]
Continuous output current per channel:	2.5 A[4]
Peak output current per channel:	5 A
Current sense:	0.525 V/A
Maximum PWM frequency:	20 kHz[5]
Minimum logic voltage:	2.5 V[6]
Maximum logic voltage:	5.5 V
Reverse voltage protection?:	Y[7]

Notes:

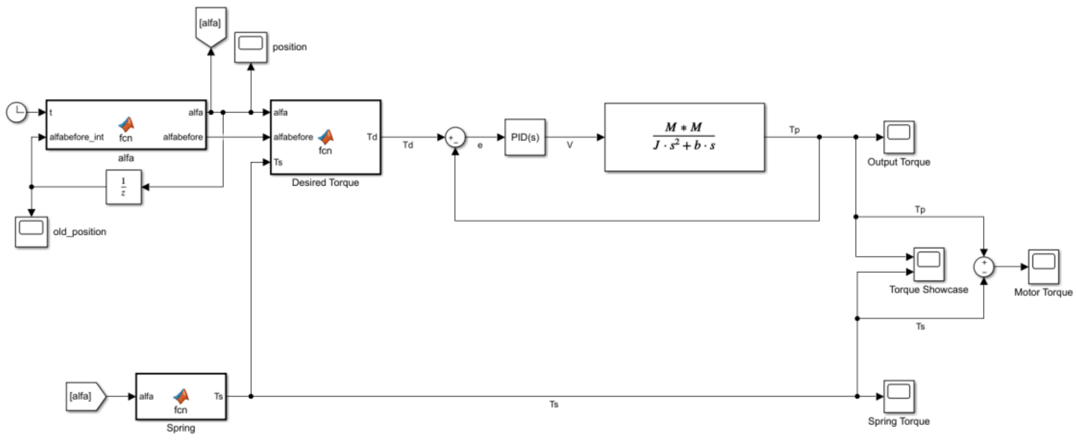
1 Without included hardware.
2 Operation from 5 V to 8 V reduces maximum current output.
3 The device is protected for transients up to 40 V.
4 Can be improved by addition of heat sink or forced air flow.
5 SLEW pin should be HIGH for frequencies above 10 kHz.
6 Input HIGH threshold can be as high as 2.0 V.
7 On motor voltage only; logic voltage does not have reverse protection.

Appendix A 3: Force Sensor Specifications

Thickness	0.203 mm (0.008 in.)
Length	56.9 mm (2.24 in.)
Width	31.8 mm (1.25 in.)
Sensing Area	25.4 mm (1 in.) diameter
Connector	2-pin Male Square Pin
Substrate	Polyester
Pin Spacing	2.54 mm (0.1 in.)
Linearity (Error)	< ±3% of full-scale Line drawn from 0 to 50% load
Repeatability	< ±2.5% Conditioned sensor, 80% of full force applied
Hysteresis	< 4.5% of full-scale Conditioned sensor, 80% of full force applied
Response Time	< 5µsec Impact load, output recorded on oscilloscope
Drift	< 5% per logarithmic time scale Constant load of 111 N (25 lb.)
Operating Temperature	- 40°C - 60°C (-40°F - 140°F) Convection and conduction heat sources
Durability	≥ 3 million actuations Perpendicular load, room temperature, 22 N

Appendices B

Appendix B 1: MATLAB-Simulink Controller of the System



Appendix B 2: Transfer Function Constants

```
%% Constants

L   = 0.091*10^-3;  % Inductance of motor
R   = 0.108;        % Resistance of motor
k_t = 0.03;         % Torque constant
k_e = 0.03;         % Speed constant
k   = 0.1;          % Spring constant
J_m = 0.0835;       % Rotor inertia
m_l = 4.8;          % Load mass
r_l = 0.16;         % Load radius
b_m = 0.01;         % ?Friction coefficient of motor
b_l = 0.01;         % ?Friction coefficient of load
M   = 1/43;         % Reduction Ratio

%% System Constants

J_l = (m_l*r_l^2)/2;
J   = J_m + M^2*J_l;
b   = b_m + M^2*b_l;
```

Appendix B 3: PID Constants

```
syms s Kp Kd
max_overshoot = 0.01;
rise_time     = 0.1;
zeta          = 0.826; % calculated by hand
w_n           = (1-0.4167*zeta+2.917*zeta^2)/rise_time;

%G    = 0.92/(s^2+1187*s+2.63);                       % Calculated Form
%G    = (M*k_t)/((J*L)*s^2+(L*b+J*R)*s+(k_t*k_e+R*b));
% Speed Control
G    = (M*M)/(J*s^2+b*s);                              % Torque Control

G_c  = Kp + Kd*s;
T_s = G_c*G/(1+G_c*G);

pretty(simplify((G)))
%pretty(simplify((T_s)))

[n,d] = numden(T_s);
C = fliplr(coeffs(d,s));

eqn1 = C(2)/C(1) == 2*zeta*w_n;
Kd = solve(eqn1,Kd);

eqn2 = C(3)/C(1) == w_n^2;
Kp = solve(eqn2,Kp);
```

Appendix B 4: Desired Torque Function

```
function Td = fcn(alfa, alfabefore, Ts)

m = 4.8;     % kg
g = 9.81;    % m/s^2
r = 0.15;    % m

%% CCW positive

if alfa-alfabefore<0
```

```
Td = (m*g*r*abs(sin(alfa))-Ts)*0.5*1.5;

else

Td = -((m*g*r*abs(sin(alfa)))+Ts)*0.5*1.5;

end
```

Appendix B 5: Position Function

```
function [alfa, alfabefore] = fcn(t, alfabefore_int)

if mod(t,2)==0

    alfa= -270;

elseif (mod(t,2)>0 && mod(t,2)<1)

    alfa = 120*mod(t,2)+-270;

else

    alfa = -150-120*(mod(t,2)-1);

end

if t==0

    alfabefore = -270;

else

    alfabefore = alfabefore_int;

end
end
```

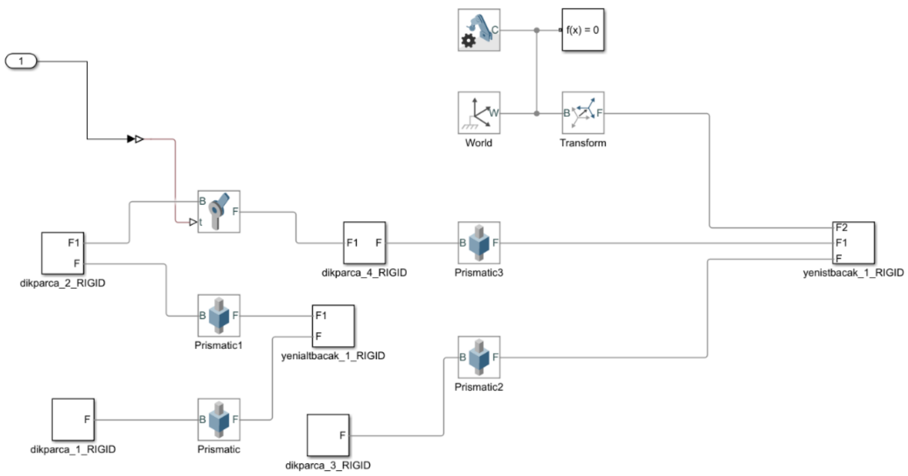
Appendix B 6: Spring Torque Function

```
function Ts = fcn(alfa)


alfa_spring = -270;
E = 100*10^9;              % Pascal
width = 200*10^-3;       % meter
thickness = 1*10^-3;      % meter
length = 500*10^-3;        % meter


Ts = (pi*E*width*(thickness^3)*((alfa-alfa_spring)/360))/6*length;
```

Appendix B 7: Simscape Exoskeleton System



Appendix B 8: Arduino ADC Code

B8 – Yeni Arduino kodunu koyacaksın unutma

Appendix B 9: Raspberry Python PID Code

```
import RPi.GPIO as GPIO
import numpy as np
import PID
from time import sleep

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)  # Disabling annoying warnings
IN1, IN2, D2 = 27, 17, 12  # IN1, IN2, PWM pin allocations
en1, en2, en3, en4, en5 = 14, 15, 18, 23, 24
FB1, FB2, FB3, FB4, FB5 = 10, 9, 11, 0, 5
s_front, s_back = 19, 26

GPIO.setup(IN1, GPIO.OUT)
GPIO.setup(IN2, GPIO.OUT)
GPIO.setup(D2, GPIO.OUT)
GPIO.setup(en1, GPIO.IN)
GPIO.setup(en2, GPIO.IN)
GPIO.setup(en3, GPIO.IN)
GPIO.setup(en4, GPIO.IN)
GPIO.setup(en5, GPIO.IN)
GPIO.setup(FB1, GPIO.IN)
GPIO.setup(FB2, GPIO.IN)
GPIO.setup(FB3, GPIO.IN)
GPIO.setup(FB4, GPIO.IN)
GPIO.setup(FB5, GPIO.IN)
GPIO.setup(s_front, GPIO.IN)
GPIO.setup(s_back, GPIO.IN)

m = 4.8
g = 9.81
r = 0.15
c_torque = 2  # Nm/A
```

```
alfa_spring = -270
E = 10 * 10 ** 9
width = 100 * 10 ** -3
thickness = 1 * 10 ** -3
length = 500 * 10 ** -3
Kp = 108140
Ki = 79823
Kd = 6733


def decimal(x1, x2, x3, x4, x5):
dac = x1 * 2 ** 0 + x2 * 2 ** 1 + x3 * 2 ** 2 + x4 * 2 ** 3 + x5 * 2 ** 4
return dac


def aci(encoder):
pos = 90 * encoder / 32
return pos


def niyet(front_sensor, back_sensor):
if front_sensor == 1 and back_sensor == 0:
intent = "alt bacak geri"
elif front_sensor == 0 and back_sensor == 0:
intent = "alt bacak ileri"
elif front_sensor == 1 and back_sensor == 1:
intent = "ayakta"
elif front_sensor == 0 and back_sensor == 1:
intent = "oturma"
else:
intent = "tanımlanamadı"
return intent
```

```
pwm = GPIO.PWM(D2, 100)  # GPIO.PWM(pwm pini, frekans(Hz))
pwm.start(0)  # maksimum %60 olmalı çünkü sürücü 3A verebiliyor maksimum
pid = PID.PID(Kp, Ki, Kd)
pid.setSampleTime(0.001)
while True:
sensor_front = GPIO.input(s_front)
sensor_back = GPIO.input(s_back)
print(sensor_front, sensor_back)
intention = niyet(sensor_front, sensor_back)

dig_en1 = GPIO.input(en1)
dig_en2 = GPIO.input(en2)
dig_en3 = GPIO.input(en3)
dig_en4 = GPIO.input(en4)
dig_en5 = GPIO.input(en5)
dec_en = decimal(dig_en1, dig_en2, dig_en3, dig_en4, dig_en5)
alfa = aci(dec_en)

dig_FB1 = GPIO.input(FB1)
dig_FB2 = GPIO.input(FB2)
dig_FB3 = GPIO.input(FB3)
dig_FB4 = GPIO.input(FB4)
dig_FB5 = GPIO.input(FB5)
current_current = round(decimal(dig_FB1, dig_FB2, dig_FB3, dig_FB4, dig_FB5),
3)  # Sürücüden gelen akım okunuyor
#     print(dec_cur, dig_FB1, dig_FB2, dig_FB3, dig_FB4, dig_FB5)

# Spring Torque
Ts = (np.pi * E * width * (thickness ** 3) * ((alfa - alfa_spring) / 360)) / (6 * length)
# Reference Signal (torque/torque constant = current) # ref = (m*g*r*np.sin(alfa)-
Ts)*0.5*1.5/c_torque

if intention == "alt bacak geri":
ref_x = (m * g * r * np.sin(np.deg2rad(45)) + Ts) * 0.5 * 1.5 / c_torque
```

```
elif intention == "alt bacak ileri":
ref_x = (m * g * r * np.sin(np.deg2rad(0)) + Ts) * 0.5 * 1.5 / c_torque
elif intention == "ayakta":
ref_x = (m * g * r * np.sin(np.deg2rad(0)) + Ts) * 0.5 * 1.5 / c_torque
# elif intent == "oturma":
#     ref_x = (m * g * r * np.sin(np.deg2rad(90)) + Ts) * 0.5 * 1.5 / c_torque
# elif intent == "kalkma":
#     ref_x = (m * g * r * np.sin(np.deg2rad(0)) + Ts) * 0.5 * 1.5 / c_torque
else:
ref_x = (m * g * r * np.sin(np.deg2rad(0)) + Ts) * 0.5 * 1.5 / c_torque
print("napıyon anlamıyorum")
# ref_x = (m * g * r * np.sin(np.deg2rad(alfa)) + Ts) * 0.5 * 1.5 / c_torque
ref = round(ref_x, 3)
# AÇI=0  REF = 0.40
# AÇI=90 REF = 3.04
# PID
pid.SetPoint = ref  # Referans
error = round(ref - current_current, 3)  # Referanstan şu anki değer çıkarılıp hata
bulunuyor
pid.update(error)  # PID'ye error Feedbacki veriyor
target = round(pid.output, 0)

maks = 21695535
minimum = -18517492
target_max = 60
target_min = 0
target_pwm = round((target / (maks - minimum) * (target_max - target_min)) +
target_min, 0)
print(target_pwm)
# Clipper, Clamper eklenecek

# MOTOR PID CONTROL
if intention == "alt bacak geri":
if 85 >= alfa > 0:
```

```
if target_pwm < 0:
target_pwm_abs = abs(target_pwm)
if target_pwm_abs <= 0:
target_pwm_abs = 0
if target_pwm_abs >= 60:
target_pwm_abs = 60
pwm.ChangeDutyCycle(target_pwm_abs)
GPIO.output(IN1, GPIO.LOW)
GPIO.output(IN2, GPIO.HIGH)
else:
if target_pwm <= 0:
target_pwm = 0
if target_pwm >= 60:
target_pwm = 60
pwm.ChangeDutyCycle(target_pwm)
GPIO.output(IN1, GPIO.HIGH)
GPIO.output(IN2, GPIO.LOW)
print("Kullanıcı yürümekte")
elif alfa == 0:
pwm.ChangeDutyCycle(10)
GPIO.output(IN1, GPIO.HIGH)
GPIO.output(IN2, GPIO.LOW)
print("0 dereceyi geçemezsin")
elif alfa > 85:
pwm.ChangeDutyCycle(10)
GPIO.output(IN1, GPIO.LOW)
GPIO.output(IN2, GPIO.HIGH)
print("90 dereceyi geçemezsin")
else:
pwm.ChangeDutyCycle(0)
print("arıza")
elif intention == "alt bacak ileri":
if 85 >= alfa > 0:
if target_pwm < 0:
```

```
target_pwm_abs = abs(target_pwm)
if target_pwm_abs <= 0:
target_pwm_abs = 0
if target_pwm_abs >= 60:
target_pwm_abs = 60
pwm.ChangeDutyCycle(target_pwm_abs)
GPIO.output(IN1, GPIO.LOW)
GPIO.output(IN2, GPIO.HIGH)
else:
if target_pwm <= 0:
target_pwm = 0
if target_pwm >= 60:
target_pwm = 60
pwm.ChangeDutyCycle(target_pwm)
GPIO.output(IN1, GPIO.HIGH)
GPIO.output(IN2, GPIO.LOW)
print("Kullanıcı yürümekte")
elif alfa == 0:
pwm.ChangeDutyCycle(10)
GPIO.output(IN1, GPIO.HIGH)
GPIO.output(IN2, GPIO.LOW)
print("0 dereceyi geçemezsin")
elif alfa > 85:
pwm.ChangeDutyCycle(10)
GPIO.output(IN1, GPIO.LOW)
GPIO.output(IN2, GPIO.HIGH)
print("90 dereceyi geçemezsin")
else:
pwm.ChangeDutyCycle(0)
print("arıza")
elif intention == "ayakta":
pwm.ChangeDutyCycle(0)
print("Kullanıcı ayakta")
else:
```

```
pwm.ChangeDutyCycle(0)
print("Beklenmedik durum")
```

CURRICULUM VITAE

- **Mustafa Şimşek**
- **Yücel Can Aksu**
- **Bahadır Sami Ekmen**