Development of an Affordable Zwift Enabled Smart Bicycle Trainer

by

Dylan Charl Eksteen
22623906

Mechatronic Project 478

Draft Report

Study leader: Dr. G. Venter

September 2022

forward together • saam vorentoe • masiye phambili

Department of Mechanical and Mechatronic Engineering
Departement Meganiese en Megatroniese Ingenieurswese
Privaat Sak X1, Private Bag X1, Matieland, 7602
Tel: +27 21 808 4204 | www.eng.sun.ac.za

Declaration

By submitting this report electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Signature: ....

Date: September 19, 2022

Student Name: D. C. Eksteen

Student Number: 22623906

MECHATRONIC PROJECT 478:
EXECUTIVE SUMMARY

Student: D. C. Eksteen

Title of Project
Development of an Affordable Zwift Enabled Smart Bicycle Trainer
Objectives
• Design and build an inexpensive bicycle trainer with controllable resistance • Design and implement software for connecting and interfacing with Zwift • Test and calibrate the resistance and connectivity of the trainer
What is current practice and what are its limitations?
• Commercial smart trainers that are compatible with Zwift are very expensive and thus inaccessible to many consumers • Accessories such as training tyres and additional cassettes also add to the costs of many available trainers
What is new in this project?
The project will develop a focused product, that will be inexpensive, simple and intuitive to use and easy to maintain
If the project is successful, how will it make a difference?
The developed trainer will make the Zwift training experience more accessible and affordable to a wider range of consumers. The trainer will also not need many accessories to use, and will be usable by many different types of bicycles
What are the risks to the project being a success? Why is it expected to be successful?
• Connectivity with Zwift: There is a risk that Zwift will require proprietary software in order to connect hardware to the platform • Supply and manufacturing of components. There is a global shortage of components and manufacturing may be expensive and time consuming
What contributions have other students made?
• There are various different examples of non-smart trainers on the market, which will serve as examples of what technologies work with similar products • There are some examples of developed BLE connectivity, as well as some libraries to enable the use of python in the development process

Which aspects of the project will carry on after completion and why?
NA
What arrangements have been made to expedite continuation?
• The developed software and electronics schematics will be available on a GitHub repository under an open license • The hardware design and concept will be documented and presented in this report, and in the final project poster

September 19, 2022

Student	Date	Lecturer

MECHATRONIC PROJECT 478:
ECSA EXIT LEVEL OUTCOME EVALUATION

ELO 1: Problem Solving	
Demonstrate competence to identify, assess, formulate and solve convergent and divergent engineering problems creatively and innovatively.	• Chapters 1,2,3 and 4 • Poster

- Identify the need for an inexpensive smart trainer that will be able to interface with Zwift.
- Indicate the criteria needed to connect with Zwift, as well as have a usable trainer done in Introcuction and Literature Review. (Chapter 1 and 2).
- Best sollution for general problem presented, then split into seperate sections to solve further ito. Software and Hardware requirements (Chapter 3 and 4)

ELO 2: Application of Scientific and Engineering Knowledge	
Demonstrate competence to apply knowledge of mathematics, basic science and engineering sciences from first principles to solve engineering problems.	• Section 4.1

- Model of Forces involved in trainer for use in basic hardware design and component selection.
- Mathematical model and analysis of Eddy Current Brake. (Section 4.1)

ELO 3: Engineering Design	
Demonstrate competence to perform creative, procedural and non-procedural design and synthesis of components, systems, engineering works, products or processes.	• Chapter 2 and 3

- Design the software approach, communicating with Zwift through BLE protocol.
- Design a hardware approach for both the resistance brake and the general trainer.
- Design the electro-mechanical control of the braking unit and the controlling software. (Chapter 2 and 3)

ELO 5: Engineering Methods, Skills and Tools, Including Information Technology	
Demonstrate competence to use appropriate engineering methods, skills and tools, including those based on information technology.	• Chapter 3 and 4

- Engineering method applied to solving the communication and control of the trainer with an external Zwift host.
- Engineering method applied to selecting, designing and implementing the hardware components needed for the trainer.
- Adequate consideration for the goal of the product applied to the planning and design phases.

ELO 6: Professional and Technical Communication	
Demonstrate competence to communicate effectively, both orally and in writing, with engineering audiences and the community at large.	• Report • Poster • Presentation

- The Report and the oral presentation will aim to prove the ability to achieve the communication.
- The Report will focus on presenting the project to a technical and engineering audience where the poster will be aimed at presenting to the wider community.

ELO 8: Individual, Team and Multidisciplinary Working	
Demonstrate competence to work effectively as an individual, in teams and in multi-disciplinary environments.	• Section X

- Project focuses on Individual work, with guidance from supervisor and input from other students working under same supervisor.
- The project proposal, progress report, progress presentation, draft report and final report will be handed in to achieve this.

ELO 9: Independent Learning Ability	
Demonstrate competence to engage in independent learning through well-developed learning skills.	• Chapter 2 & 3

- The learning of Bluetooth technology and application of Eddy Current Brake models prove independent learning.

Contents

List of Tables

List of Figures

List of Abbreviations

API	Application Program Interface
BLE	Bluetooth Low Energy
BR/EDR	Basic Rate/Enhanced Data Rate
BSIG	Bluetooth Special Interest Group
Crr	Coefficient of Rolling Resistance
Cw	Wind Resistance Coefficient
DIS	Device Information Service
FTMS	Fittness Machine Data
GATT	Generic ATTribute Profile
IDE	Integrated Development Environment
IPC	Inter-Process Communicationt
MTB	Mountain Bike
RPC	Remote Procedure Calls
SIM	Indoor Bike Simulation
UHF	Ultra-High Frequency

List of Symbols

B_r	Magnet Remanence Field	T
B	Magnetic Flux Density	T

Chapter 1

Introduction

1.1 Background

Cycling is a very old sport that has been practised and followed around the world for decades. It is great for building fitness and for keeping active, and often also serves as a means of transportation for commuting. Cycling as a sport is often very social and competitive in nature as people would often cycle in large groups where they would compete between each other and with other groups. It is this social and competitive nature that often attracts people to taking up cycling as a sport.

In recent years, it has become very popular for cyclists to train indoors as this is often safer than cycling on busy city streets, it makes cycling less reliant on the weather and is more convenient. The problem is that indoor cycling can often be very boring and lonely.

That is where Zwift comes in. Zwift is an online based virtual reality platform that allows indoor cyclists to virtually ride courses in the real world whilst communicating and competing with other people in real time through the platform. It makes indoor training more social and less boring.

In order to use the Zwift platform, you need a compatible trainer or training device in order to send riding information to Zwift and to receive feedback that will adjust the riding experience based on the virtual circumstances. These trainers are very expensive and thus not very accessible to many consumers.

This project will aim to convert a non-compatible cheap trainer to be Zwift compatible through hardware and software development and implementation. The project, conducted by Mr DC.Eksteen as part of Mechatronic Project 475, stems from a proposal from Prof G.Venter.

1.2 Aim and Objectives

The aim of the project is to modify and add to a cheap trainer to enable Zwift compatibility. The objectives will be to:

- Develop Hardware to:
    1. Monitor and measure the rider inputs. These include, for example, cycling speed, cadence, power, heart rate etc.
    2. Change the riding conditions based on feedback from Zwift platform. These include, for example, rolling resistance, incline etc.
- Develop Software to:
    1. Collect and process rider inputs and to communicate this to the Zwift platform.
    2. Receive feedback information from Zwift and enable hardware to change the riding conditions as mentioned above.
- Combine developed Hardware and Software to produce a smart trainer that is comparable to smart trainers that are available on the market, but at a lower cost point. (Needs refinement)

1.3 Motivation

Motivation will focus on:

- What:
  - Zwift = popular indoor training and competitive platform.
  - Bicycle trainer = Convenient way to train cycling without needing to cycle outside.
- Why:
  - Indoor training increasing in popularity.
  - Making access to Zwift more accessible and thus promoting cycling and exercise.
- How:
  - Convert existing "dumb" trainer into smart trainer.
  - Enable trainer to connect with Zwift platform.
  - Using and leveraging existing industry standards to produce a robust solution.

Chapter 2

Literature Review

The literature review is performed to define the requirements for the connection with Zwift, as well as discovering and understanding the various desired and required features and products that exist on the market today.

2.1 Overview

2.2 Zwift: Technology and Requirements

As mentioned above, Zwift uses real-time training data to drive the avatar in the virtual world. In order to achieve this, Zwift needs to run on a device that has :

- An internet connection to connect to the Zwift servers via the Application Program Interface (API).
- Bluetooth Low Energy (BLE) or ANT+ functionality to receive the data from the training session.

This device is usually a laptop, mobile phone or tablet, and will also display the avatar in the virtual world to the player. Moving forward, this device will be referred to as the "host" device as it is the central device controlling the user experience. (von Bromley, 2022)

2.2.1 Basic Requirements

As mentioned above, Zwift requires real-time tracking data from the player in order to control the avatar. The interactiveness and accuracy of the experience greatly depends on the amount and nature of the data that is sent to the host device (also sometimes referred to as the 'main unit').

Requirement	Impact
Speed data	Most basic data requirement and is essential to the use of the Zwift platform.
Cadence data	Not essential for platform, but a feature that many players prefer to have available.
Power data	Not essential for platform, although it is the preferred metric that Zwift will use to measure the effort of the player.

2.2.2 BLE vs ANT+

Both BLE and ANT+ devices communicate using Ultra-High Frequency (UHF) electromagnetic waves with frequencies around 2.4 GHz. The communication also takes part over a small distance.

BLE for Fitness Devices

BLE is one of two standards for Bluetooth communication that has been developed and maintained by Bluetooth Special Interest Group (BSIG).[1]The specific details of the technology and standard is discussed in Section 2.4. For the sake of Zwift requirements, this section will look at the specific protocol of the BLE specification that Zwift supports.

In the past, when Zwift was just launched, all of the Bluetooth communication was performed using proprietary protocols provided by the manufacturers of trainers and training equipment as there has not yet been a standard protocol defined to handle Bluetooth communication of controllable sports equipment.

On 14 February 2017, BSIG adopted the Fittness Machine Data (FTMS) protocol to the BLE Generic ATTribute Profile (GATT). Then, in late 2021 Zwift announced that they will be supporting FTMS in their latest update, and thus any new trainer that would like to interact with the Zwift platform would be required to follow the FTMS protocol as is discussed in Section 2.4. (Zwift Forums)

Table 2.1 below shows the different commands within the FTMS protocol that Zwift uses to control connected trainers. This can be deducted from the control modes that are available on the platform, and have been confirmed by a Zwift engineer on the official support forums.

[1]The other being Bluetooth Basic Rate/Enhanced Data Rate (BR/EDR)

Table 2.1: Zwift Supported FTMS Commands

Command	Description
Set Target Resistance Level	Sends the desired resistance level to the trainer.
Set Target Power	Sends the targeted power (Watts) that the trainer should aim to maintain.
Start or Resume	Starts or resumes a training session on the trainer
Stop or Pause	Stop or pause a training session on the trainer.
Set Indoor Bike Simulation Parameter	Set the simulation parameters on the trainer

Depending on the type of workout that is being performed, Zwift will use the different control point commands listed in Table 2.1 above. When the user is normally riding around within the Zwift world, it will send Indoor Bike Simulation (SIM) parameter commands. This is explained in more detail in Section 2.4. When the user is using the "Workout Mode", Zwift has the option to engage "ERG" mode, where it will send a desired power target that the trainer should aim to maintain regardless of the speed. Thus the user can decide to go faster, with the trainer lowering the resistance, or go slower with the trainer having an increased resistance. Lastly, the user can adjust the resistance on their trainer from within the Zwift software, requiring Zwift to then send that specific command to the trainer.

2.3 Existing Trainer Technology

2.4 BLE Communication

BLE works by forming pico - and scatter-net networks between various master and slave devices. Each device in the network is either a master or a slave, but never both. The devices then form a piconet when one or more slaves connect and synchronize with a single master device. The role of the master device is then to manage the slave devices through the protocol. This forms a network as can be seen in Figure 2.1 below.

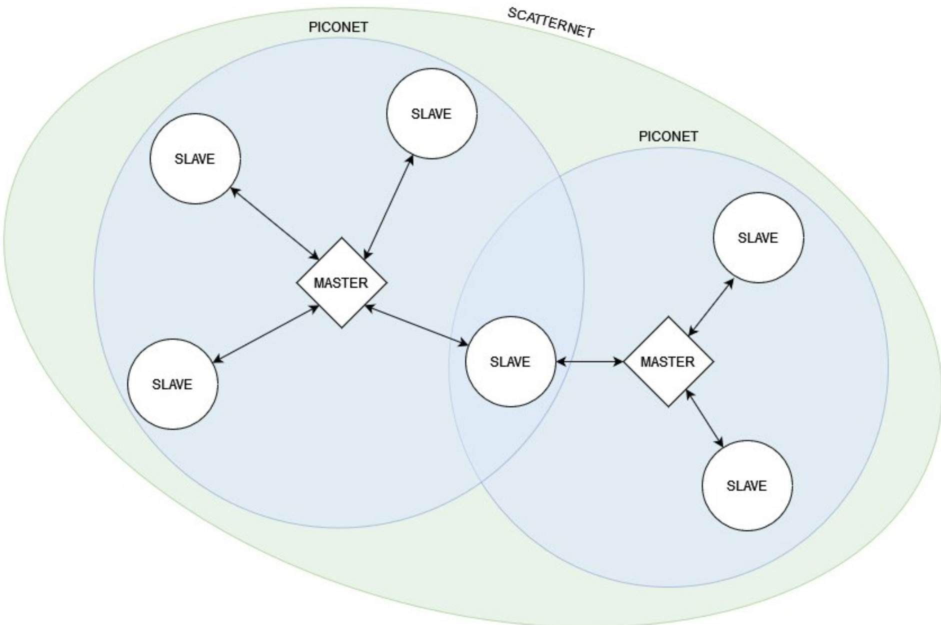


Figure 2.1: BLE Star Network Topology

The master thus determines how often a slave is allowed to communicate, and slaves only communicate when requested to do so by a master. This is with one exception in the case of "advertising". Advertising is a feature that allows a slave device to announce that it has something to transmit to the master. This is either an event message or a measurement value.

Some other features worth considering are listed in Table 2.2 below:

Table 2.2: Technical Specifications of the BLE Standard

Description	Specification	Detail
Data Package Size	64 bit (min) to 216 bit (max)	The size of any single data package that can be sent between a master and slave(s).
Data Transfer Rate	1 Mbps	The rate at which the data is typically transferred between devices.
Security	128-bit AES	Security and encryption varies in levels, but generally data is encrypted between devices with a "shared secret" key that is generated during pairing.
Latency	6 ms	Time between device starting transmission and data getting transferred. (Aka network lag).
Power Consumption	0.01 W to 0.5 W	The power required by a device to join and operate on the piconet.

FTMS Protocol

Within the FTMS protocol, there is a sub-protocol dedicated to Indoor Bike Data that focuses specifically on the application of the protocol for bike trainers. (Bluetooth®Service Specification, section 4.9).

The specification outlines and defines services that are supported by the protocol. Table 2.3 and 2.4 below lists the available features that are of interest to the project. Machine Features are Parameters that can be requested from the trainer, and Target Features are training parameters that can be set from an external device. (Bluetooth®Service Specification)

Table 2.3: Relevant Optional Fitness Machine Features of FTMS Protocol

Number	Feature	Description
NA[a]	Instantaneous Speed	Trainer can determine the instantaneous speed of the user.
0	Average Speed Supported	Trainer can determine the average speed of the user for the duration of the session.
1	Cadence Supported	Trainer can determine pedalling rate of the user (rpm).
7	Resistance Level Supported	Trainer can determine the resistance the user experiences.
10	Heart Rate Measurement Supported	Trainer is able to determine the heart rate of the user.
14	Power Measurement Supported	Trainer is able to determine the power generated by the user.

[a]This feature is compulsory for protocol.

The parameters that would be relevant to the control of a bicycle trainer connected to Zwift are listed in Table 2.4 below:

Table 2.4: Relevant Optional Target Setting Features of FTMS Protocol

Number	Feature	Description
1	Inclination Target Setting	Trainer can adjust difficulty to simulate inclination.
2	Resistance Target Setting	Trainer can adjust resistance level.
3	Power Target Setting	Trainer can adjust difficulty to maintain power target.
13	SIM Parameters	Trainer supports SIM mode.

SIM mode

This mode is defined in the FTMS documentation, and needs Control Permission to be activated. Once the mode is activated, the trainer will receive parameter arrays with the data shown in Table 2.5 below to determine the required resistance level.

Table 2.5: SIM Mode Parameter Data

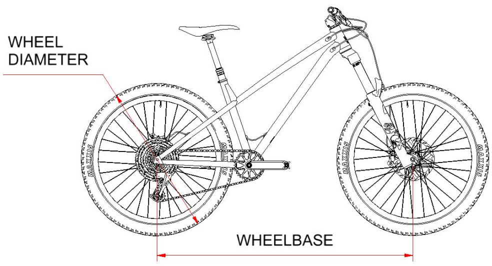
Parameter	Unit	Size	Resolution
Wind Speed	$\mathrm{m\,s^{-1}}$	16 bit	0.001
Grade	%	16 bit	0.01
Coefficient of Rolling Resistance (Crr)		8 bits	0.0001
Wind Resistance Coefficient (Cw)	$\mathrm{kg\,m^{-1}}$	8 bits	0.01

The *Grade* indicates the gradient of the road that the avatar is travelling on. The Coefficient of Rolling Resistance (Crr) is determined by the road surface and wheel type of the avatar in the game, where the *Wind Resistance Coefficient (Cw)* is determined by the player's weight and height, and can then be reduced by in-game events such as equipment selection and drafting behind other avatars when this feature is active. Currently Zwift does not utilize the *Wind Speed*

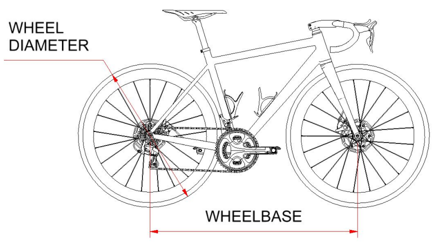
parameter.
These parameters are then used by the trainer to determine the resistance required to simulate the conditions in the game. This is what creates the realistic training experience.

2.5 Bicycle Specifications

The two most common types of bicycles that are expected to be used on the trainer are Mountain Bikes (MTBs) and road bicycles. The dimensions that are relevant to the implementation of the trainer are the wheelbase, wheel diameter and weight, as shown in Figure 2.2 below.



(a) MTB (model by: Pratama, 2021)



(b) Road Bicycle (model by: Morozev, 2017)

Figure 2.2: Bicycle Dimensions

Although these dimensions vary between brands, models and sizes, a general overview of common dimensions can be found by comparing the largest and smallest size dimensions from the best selling models in each category, as well as the average size that can be expected. The best selling bike models for each category are shown with their dimensions in Table 2.6 below: (Lin, 2021)

Table 2.6: Best selling road bicycle specifications (Geometry Geeks)

Road Bike	Wheelbase		MTB	Wheelbase	
	Smallest	Largest		Smallest	Largest
Specialized Tarmac	969 mm	1012 mm	Specialized Epic	1116 mm	1211 mm
Specialized Roubaix	981 mm	1024 mm	Trek Fuel EX	1144 mm	1323 mm
Cervelo R3	971 mm	1024 mm	Specialized Stumpjumper	1152 mm	1302 mm

2.6 Eddy Current Brake

2.7 Sensor Technology

2.8 Conclusion

Chapter 3

Software Development

The software that was developed for this project was written in Python 3, using Visual Studio Code Integrated Development Environment (IDE). The software was then deployed on a Raspberry Pi 3 B+ computer.

3.1 Structure

The software for the project has been written to form a very modular structure with separate "handlers" being written/developed for different functions of the greater application. There are some exceptions where existing libraries and servers were used, but these are referenced in the relevant sections later in this chapter.

Figure 3.1 below shows the modular structure that was implemented for the control of the Raspberry Pi 3B+ single-board computer

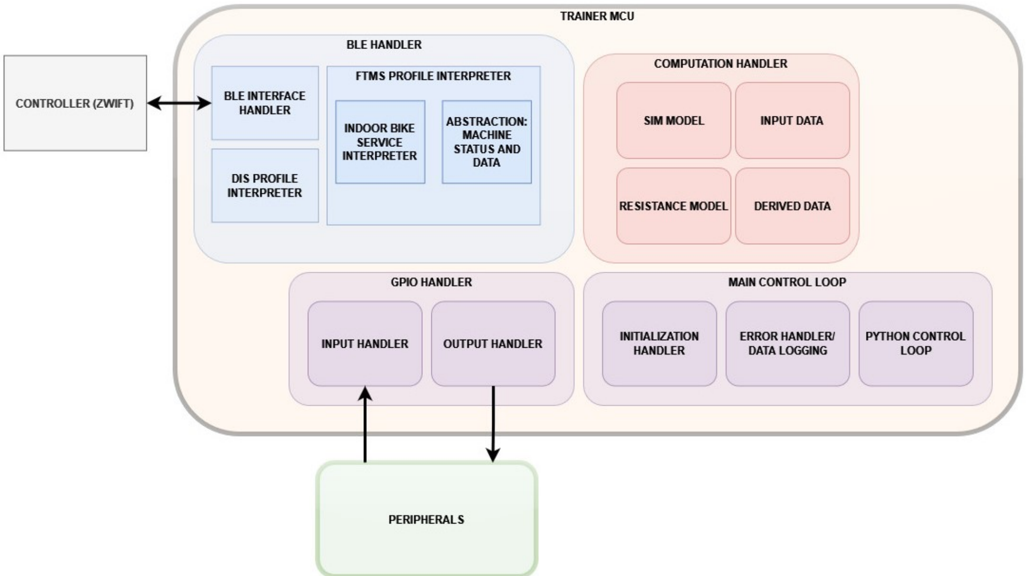


Figure 3.1: Overview of General Software Structure

3.2 BLE Handler

The BLE handler is responsible for all of the Bluetooth communication protocols, and has to strictly conform to the BLE specifications in order to ensure compatibility with common devices. The handler consists of a few different services that in tern manage some service or function within the greater protocol.

The BLE protocol was designed to be used with various layers of abstraction between the low level operations of the system. This results in the structure for the abstraction layers as seen in Figure 3.2 below:

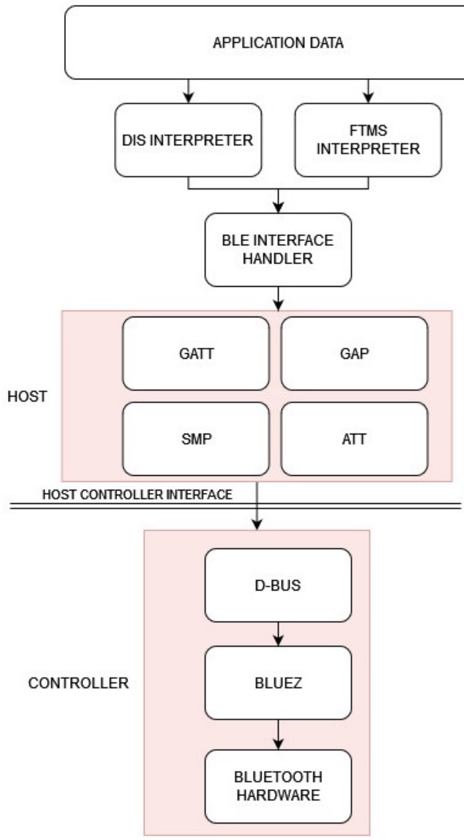


Figure 3.2: BLE Protocol Stack

The abstraction all the way up to the host already exists and is implemented on the Raspberry Pi. The D-Bus is an Inter-Process Communicationt (IPC) method that allows for Remote Procedure Calls (RPC) between the various processes. The role of the BLE Interface Handler is thus to handle communication between the main process and Bluez. Bluez is the official Linux Bluetooth protocol stack, and is what controls the Bluetooth operations of the system. (Lee, 2020)

The FTMS and Device Information Service (DIS) Interpreters receive and send various BLE commands, requests and replies between the BLE Interface Handler and the running application. It is here where messages are split up into their various components before being acted apron, and replies get created in the required format before being sent to the Bluez stack.

Chapter 4

Hardware Development

For the project, hardware was developed with the aim of achieving the objectives shown in 4.1 below:

Table 4.1: Objectives of Hardware Development

Hardware Design Outcome	Description	Required Outcome
HDO 1	Eddy Current Brake	Design for adaptable length
HDO 2	Speed Sensors	Adequate Braking Force
HDO 3	Frame and Shafts	Description
HDO 4	Outcome 4	Description

4.1 Expected Operating Conditions

Roller trainers require the cyclist to be travelling at some speed for stability. Thus, the expected operating range of a cyclist on the trainer will typically range between 10 $\mathrm{km\,h^{-1}}$ and 50 $\mathrm{km\,h^{-1}}$. For the design, a maximum high speed of 60 $\mathrm{km\,h^{-1}}$ was considered. The corresponding drum speeds are shown in Figure 4.1 below.

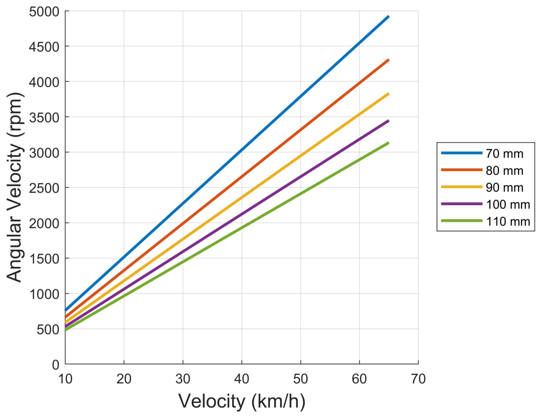


Figure 4.1: Rotational Speed of Roller Size Comparison

For the Eddy Current brake, as discussed in Section 4.2, higher speeds will result in higher braking force and thus a larger available resistance range. On the other hand, higher speeds will also increase the free-rolling resistance of the trainer, and allow for less range at higher speeds. Thus, 90 mm drums were selected as a good compromise, and were selected for the model.

Typical cyclists average between 75 and 100 W, and pro cyclists can reach up to 400 W, during a 1 hour workout.

4.2 Eddy Current Brake Design

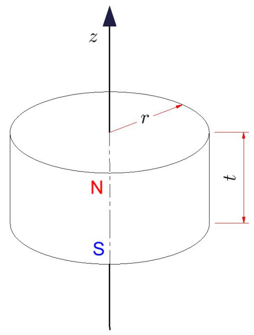
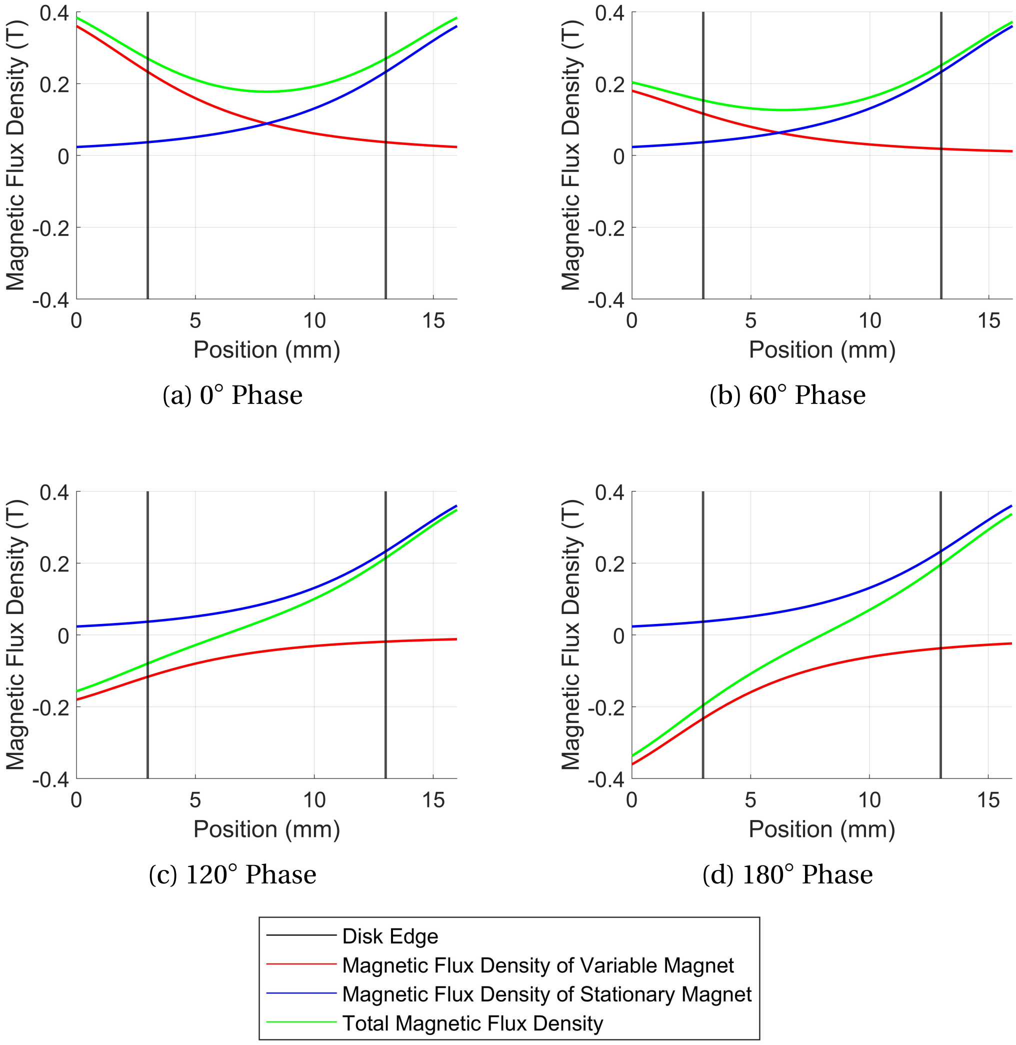


Figure 4.2: Magnetic Flux Density of Disc Magnet

(Addapted from Supermagnete)

$$B = \frac{B_r}{2}\left(\frac{t+z}{\sqrt{r^2+t+z^2}} - \frac{z}{r^2+z^2}\right) \quad (4.1)$$



(a) 0° Phase

(b) 60° Phase

(c) 120° Phase

(d) 180° Phase

Figure 4.3: Flux Distribution

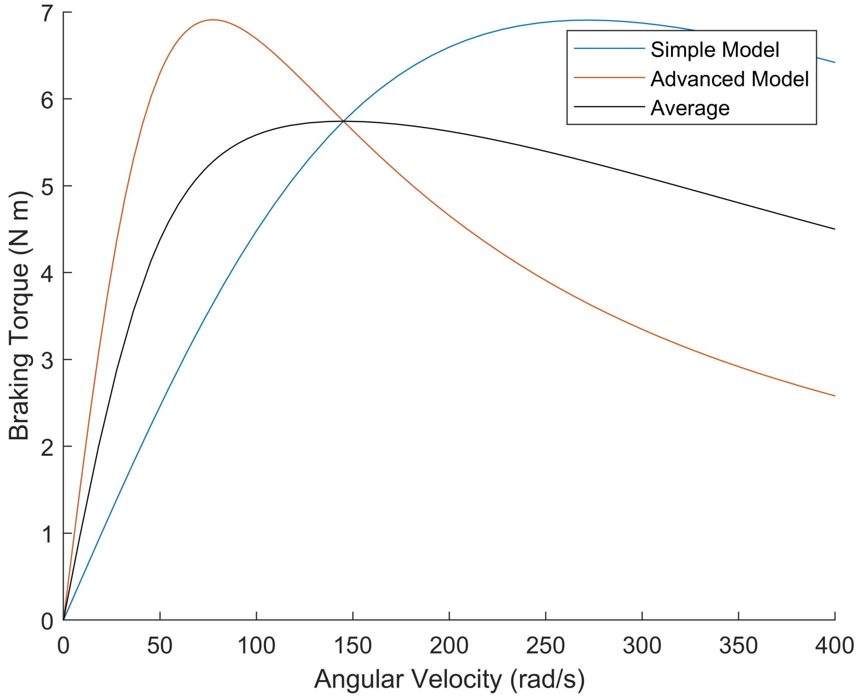


Figure 4.4: Figure 7

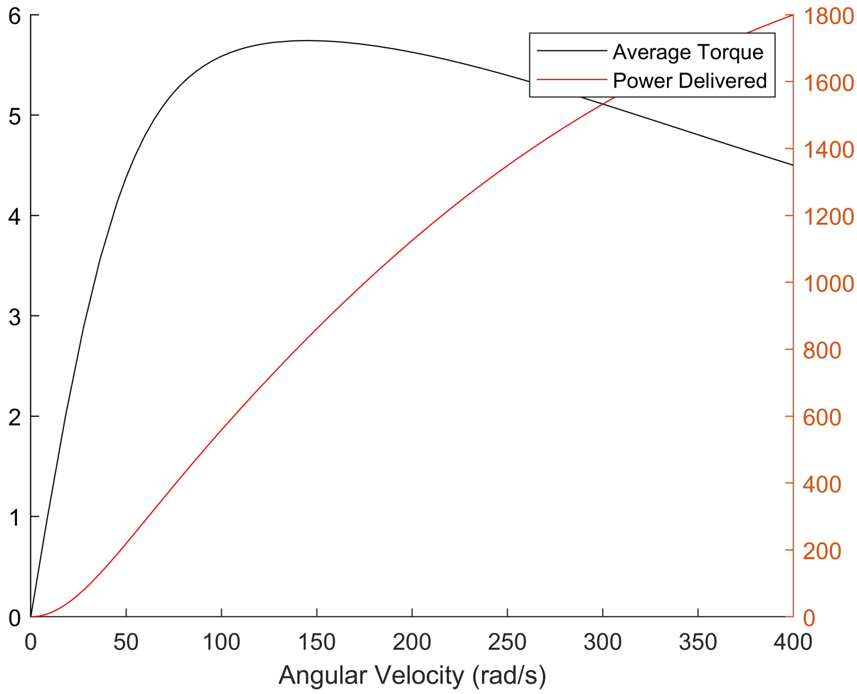


Figure 4.5: Figure 8

4.3 Sensors

Speed sensors are required to measure the speed of the cyclist ...

4.3.1 Concept Evaluation

Sensors are required to measure the angular velocity of both the rear cylinders. It was decided that using a infrared transmitter-receiver device with a reflective strip on the cylinder would be sufficient. The expected maximum sampling rate, as determined in section

List of References

Bluetooth® Service Specification (2017 February 14). Fitness machine service v1.0 [online].
Availible at: https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx?doc_id=423422.
[Accessed 1 January 2022].

Geometry Geeks (2022). Find and compare bike geometry. [online].
Availible at: https://geometrygeeks.bike/.
[Accessed 18 September 2022].

Lee, A. (2020 June 29). Creating a ble peripheral with bluez [online].
Availible at: https://punchthrough.com/creating-a-ble-peripheral-with-bluez.
[Accessed 1 January 2022].

Lin, B. (2021 December 27). The pro's closet. [online].
Availible at: https://www.theproscloset.com/blogs/news/best-selling-bikes-2021.
[Accessed 1 January 2022].

Morozev, M. (2017 December 30). Grabcad [online].
Availible at: https://grabcad.com/library/road-bike-12.
[Accessed 1 January 2022].

Pratama, M.R.R. (2021 November 20). Grabcad [online].
Availible at: https://grabcad.com/library/am-ht-1.
[Accessed 1 January 2022].

Supermagnete (2010 January 1). Faq: How do you calculate the magnetic flux density? [online].
Availible at: https://www.supermagnete.de/eng/faq/How-do-you-calculate-the-magnetic-flux-density.
[Accessed 1 January 2022].

von Bromley, S. (2022). Zwift: your complete guide: Everything you need to know [online]. Availible at: `https://www.bikeradar.com/features/zwift-your-complete-guide`. [Accessed 1 January 2022].

Zwift Forums (2022 January 2). Improvements to ftms controllable trainers [online]. Availible at: `https://forums.zwift.com/t/bluetooth-improvements-for-controllable-trainers-treadmills-january-2022/569781/38?page=4`. [Accessed 1 January 2022].

Appendix A

Eddy Current Brake Calculations

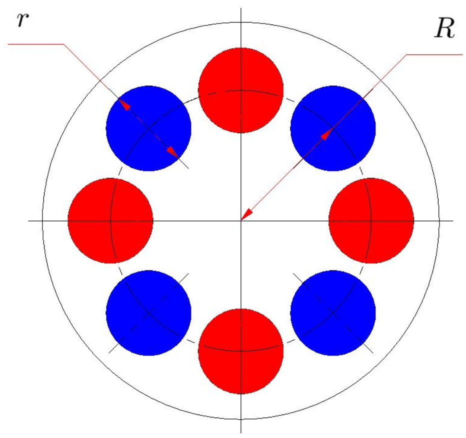


Figure A.1: Eddy Current Brake Dimensions

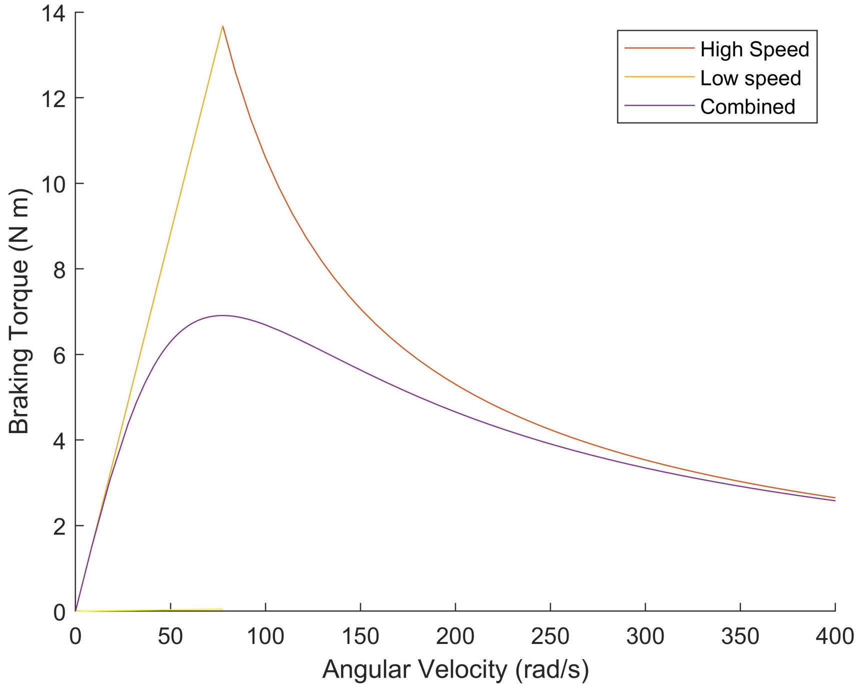


Figure A.2: Figure 5

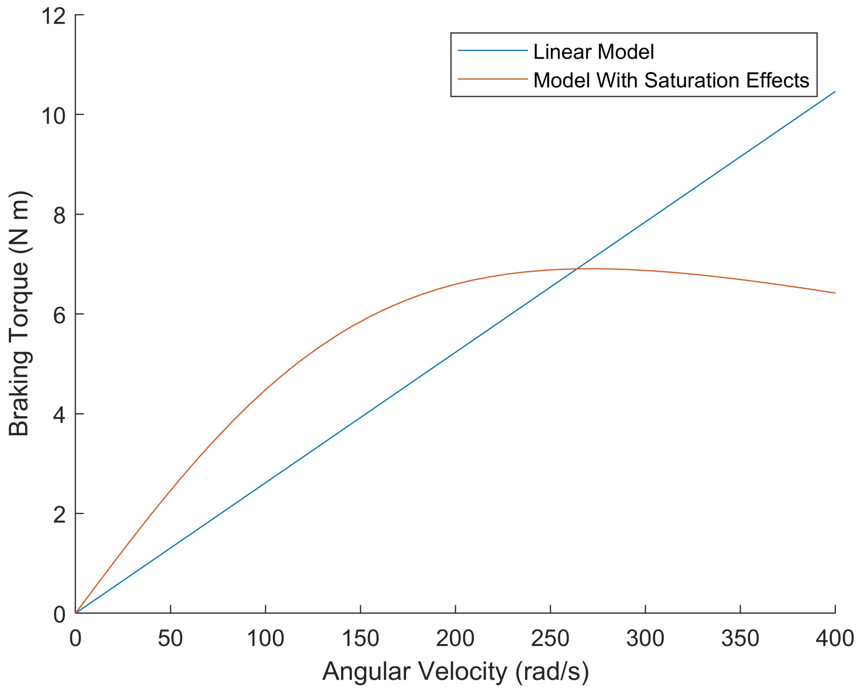


Figure A.3: Figure 6