UNIVERSIDADE
BEIRA INTERIOR

# Development of an Engine Test Bench for Hybrid Electric Propulsion Systems

**(versão final após defesa)**

**Manuel António Caetano Azevedo**

Dissertação para obtenção do Grau de Mestre em
**Engenharia Aeronáutica**
(mestrado integrado)

Orientador: Prof. Doutor Francisco Miguel Ribeiro Proença Brójo

**Covilhã, 15 de julho de 2025.**

## Declaração de Integridade

Eu, Manuel António Caetano Azevedo, que abaixo assino, estudante com o número de inscrição 44357 do Mestrado Integrado em Engenharia Aeronáutica da Faculdade de Engenharia, declaro ter desenvolvido o presente trabalho e elaborado o presente texto em total consonância com o **Código de Integridades da Universidade da Beira Interior**.

Mais concretamente afirmo não ter incorrido em qualquer das variedades de Fraude Académica, e que aqui declaro conhecer, que em particular atendi à exigida referenciação de frases, extratos, imagens e outras formas de trabalho intelectual, e assumindo assim na íntegra as responsabilidades da autoria.

Universidade da Beira Interior, Covilhã 15/07/2025

Assinado por: **MANUEL ANTÓNIO CAETANO AZEVEDO**
Num. de Identificação: 14958086
Data: 2025.07.15 12:24:06+01'00'

Para a minha mãe, a minha irmã, os meus avós,
a Leonor e a Dinda.

# Acknowledgements

As with any achievement, there were people along the way that helped me and without them this work would not have been possible. To everyone I crossed paths with during my school and university journey, I want to express my gratitude — without you, I would not be where I stand today. There are, however, some persons to which I want to express particular acknowledgements.

To my mother, for all her hard work and for teaching me to never back down in the face of difficulties and setbacks. To my father, for sparking my deep passion for aviation, which led me to this course and shaped my academic path. To my sister for always being present with a word of advice or a smile when those were most needed.

My sincere appreciation also goes to my supervisor, professor Francisco Brójo, for always being present and all the encouragement and guidance provided. I could not quantify in words the gratitude I have for the amount of patience, feedback, availability and advice demonstrated during this project.

To my girlfriend, Leonor, for her constant support, companionship, and love throughout the entire course, especially when facing major challenges such as this dissertation.

To all the janitors, canteen, bar, and faculty workers my kindest gratitude for ensuring that all students had the best conditions to conduct their studies, living their best years in UBI, and taking this university to its highest level.

This work would not have been possible without two of my dearest colleagues, João Antunes and Pedro Oliveira, which, with all the patience, helped me navigate all the issues encountered, and helping me achieve the good results of this dissertation. My acknowledgements extend to all of my colleagues and friends for the help that you have given me and all the good times that we enjoyed throughout life.

Last but not least, I want to thank all my family members, specially my grandparents and my godparents for all the lessons thought, stories shared, love given, and support showed during my whole life but particularly during this course.

Thank you.

# Resumo

Nos últimos anos, o setor aeroespacial tem dado uma grande ênfase à redução do seu impacto ambiental. Embora as emissões dos motores de combustão tradicionais tenham sido progressivamente reduzidas, também foram desenvolvidos novos tipos de unidades de propulsão. Apesar de os sistemas de propulsão elétrica serem adequados para pequenos UAVs, esta solução não é viável para aeronaves de maior dimensão devido às limitações de peso. Com o objetivo de reduzir as emissões sem aumentar o peso do sistema, surgiram sistemas de propulsão híbridos. Estes sistemas utilizam duas fontes de energia distintas, como gasolina e eletricidade. Ao testar a sua eficiência, é necessário que o banco de ensaio consiga medir o consumo de energia de ambas as fontes. Esta dissertação foca-se na construção de um banco de ensaio para sistemas de propulsão até 2 kW, com capacidade para medir o consumo tanto de combustível como de eletricidade. Para conceber o banco de ensaio, foi realizada uma investigação aprofundada sobre os vários tipos de dinamómetros, analisando as suas vantagens e desvantagens de forma a fundamentar a escolha. Foi então construído um travão de correntes de Eddy, tendo sido desenvolvido um algoritmo em MATLAB para determinar os seus parâmetros principais. Foi ainda programado e implementado um sistema completo de controlo e aquisição de dados, com o objetivo de recolher e analisar os dados obtidos. No entanto, devido a falhas no equipamento e a limitações de tempo, não foi possível utilizar o sistema inicialmente previsto, sendo necessárias algumas adaptações. Para validar o banco de ensaio, foram realizados dois testes distintos: um com um motor de combustão e outro com um motor elétrico. Durante os testes, verificou-se que, embora o banco de ensaio funcionasse conforme o esperado, uma das células de carga é demasiado grande para os motores a testar, introduzindo incertezas nas medições. Além disso, a inércia do sistema poderá ser excessiva para motores elétricos de menor dimensão, o que poderá causar sobreaquecimento nos seus componentes.

# Palavras-chave

Bancada de Teste de Motores, Travão de Correntes de Eddy, Motor, Potência, Binário, Velocidade, Dinamómetro, Combsutível, Bateria

# Abstract

In recent years, a big focus of the aerospace sector has been to reduce it environmental impact. While the emissions of traditional combustion engines have been continuously reduced, new types of power units have been developed. Whilst electric power units can be used on small Unmanned Aerial Vehicle (UAV), this solutions is not viable for bigger ones due to its weight. In an effort to reduce emissions and not increase the weight of the power system, recently a series of Hybrid propulsion systems have been developed. These systems use 2 different types of energy source such as gasoline and electricity. When testing its efficiency it is necessary for the test bench to be capable of measuring the energy consumption of both sources. This dissertation focuses on building a test bench for propulsion systems with up to 2 kW of power and to have the ability of measuring the power consumption of both fuel and electricity. In order to build the test bench a thorough investigation of the various types of dynamometers was conducted, presenting their advantages and disadvantages in order to have a pondered choice. In the end, an Eddy current brake was built and for that a MATLAB algorithm was developed to decide the key parameters of it. In order to collect and analyse data a whole control and data collection system was programmed and built. Lastly, due to some faults on the equipment and time constraints, it was not possible to use the initially thought system and some modifications had to be applied. In order to validate the test bench two separate test were conducted. One using a combustion engine and other using an electric motor. During these tests it was found that while the test bench operated as intended, one of the load cells is too big for the engines that will be tested, causing uncertainty on the system and that the system may have to much inertia for smaller electric motors causing some overheating on its systems.

# Keywords

Engine Test Bench (ETB), Eddy Current Brake (ECB), Engine, Motor, Power, Torque, Speed, Dynamometer, Fuel, Battery

# Contents

# List of Figures

# List of Tables

# Acronyms and Abreviations

**AC** Alternating Current

**CAD** Computer Aided Design

**DCA** Departamento de Ciências Aeroespaciais

**DC** Direct Current

**DP** Digital Potentiometer

**ECB** Eddy Current Brake

**EM** Electric Motor

**ESC** Electronic Speed Control

**ECU** Engine Control Unit

**ETB** Engine Test Bench

**ETBCDS** ETB Control and Data System

**HEPS** Hybrid Electric Propulsion System

**I2C** Inter-Integrated Circuit

**ICE** Internal Combustion Engine

**LCD** Liquid Crystal Display

**PS** Power-supply

**PCB** Printed Circuit Board

**SFC** Specific Fuel Consumption

**SPI** Serial Peripheral Interface

**TPS** Throttle Position Sensor

**UAV** Unmanned Aerial Vehicle

# Nomenclature

| | | |
|---|---|---|
| $A$ | Area | $m^2$ |
| $a$ | Distance | m |
| $\alpha$ | Temperature coefficient of resistance | $^\circ C^{-1}$ |
| $B$ | Magnetic flux density | T |
| $d$ | Thickness | m |
| $\varepsilon$ | Relative error | - |
| $\varepsilon_{creep}$ | Creep error | - |
| $\varepsilon_{hysteresis}$ | Hysteresis error | - |
| $\varepsilon_{non-repeatability}$ | Non-repeatability error | - |
| $\varepsilon_{total}$ | Total error | - |
| $\mathfrak{F}$ | Sum of the magnetic and anti-magnetic forces | A |
| $\mathfrak{F}_{Eddy}$ | Anti-magnetic force generated by the Eddy currents | A |
| $\mathfrak{F}_i$ | Magnetic force generated byt the current on the coil | A |
| $H$ | Magnetic field intensity | $A\,m^{-1}$ |
| I | Current | A |
| $I_net$ | Current around the magnetic circle | A |
| $J_e$ | Eddy currents density | $A\,m^{-2}$ |
| $K$ | Constant of the dynamometer system | - |
| $l$ | Equivalent length of the magnetic circle | m |
| $l_c$ | Length of the equivalent magnetic circle in the core | m |
| $l_g$ | Length of the gap between the eletromagnet pole shoes | m |
| $\dot{m}_f$ | Mass fuel flow | $g\,h^{-1}$ |
| $N$ | Engine speed | rpm |
| $N_{coil}$ | Number of turns in the coil | - |
| $n_{coil}$ | Number of coils | - |
| $\eta$ | Efficiency | - |
| $\phi$ | Magnetic flux | Wb |
| $P_{consumed}$ | Power consumed by the engine/motor | W |
| $P_{electric}$ | Dissipated power | W |
| $P_{electric}$ | Electrical power | W |
| $P_{fuel}$ | Thermal power of the fuel | W |
| $P_{shaft}$ | Power in the shaft of the engine | W |
| $Q_{HV}$ | Heating value of the fuel | $MJ\,kg^{-1}$ |
| $R$ | Resistance | $\Omega$ |
| $r$ | Equivalent radius of the pole shoe | m |
| $\mathfrak{R}$ | Total reluctance | H |
| $\mathfrak{R}_a$ | Air reluctance | H |
| $\mathfrak{R}_c$ | Core and coil reluctance | H |
| $\mathfrak{R}_d$ | Disk reluctance | H |

| | | |
|---|---|---|
| $\mathfrak{R}_d$ | Reluctance of the air gap and disk | H |
| $\mathfrak{R}^*_{d_{ax}}$ | Reluctance in the axial direction | H |
| $\mathfrak{R}^*_{d_{ra}}$ | Reluctance in the radial direction | H |
| $\rho_0$ | Electric resistivity at $20\,^\circ$°C | |
| SFC | Specific Fuel Consumption | $\mathrm{g\,kW^{-1}\,h^{-1}}$ |
| $\sigma$ | Electric conductivity | $\mathrm{S\,m^{-1}}$ |
| $T$ | Temperature | °C |
| $\tau$ | Torque | N m |
| $\mu$ | Magnetic permeability | $\mathrm{H\,m^{-1}}$ |
| $\mu_0$ | Magnetic permeability of the vacuum | $\mathrm{H\,m^{-1}}$ |
| $\mu_c$ | Relative magnetic permeability of the core | - |
| $\mu_r$ | Relative magnetic permeability of the disk | - |
| U | Voltage | V |
| $V$ | Volume | $\mathrm{m^3}$ |
| $\omega$ | Angular speed | $\mathrm{rad\,s^{-1}}$ |

# Chapter 1

# Introduction

## 1.1 Motivation

In recent years, great efforts have been made by organisations worldwide to decrease the emission of greenhouse gases into the atmosphere which are known to have a major contribution to the climate changes that are occurring throughout the world. Even though transportation as a whole isn't the most significant sector in terms of the amount of greenhouse gases produced and aviation is not one of the greater greenhouse gas emitters inside of the transportation sector [1], one of the biggest challenges it faces right now is to achieve carbon neutrality by 2050 [2].

One solution that would decrease the emissions of greenhouse gases is an all-electric propulsion system. However, with the battery technology currently available, an all-electric medium/large-size aircraft or UAV is not feasible due to the low energy density that batteries have [3], which would increase the aircraft's empty weight and in turn decrease its useful load.

There are, however, various solutions that could provide milder transitions, at least for small to medium-sized aircraft/UAVs, such as the Hybrid Electric Propulsion System (HEPS). This type of propulsion system combines the traditional Internal Combustion Engine (ICE) with the newer, more efficient Electric Motor (EM) and power management strategies that allow the increase of overall efficiency of these systems [4]. Nonetheless, for this solution to work, there is a need for a full characterization of the HEPS, which allows the development and optimization of the Engine Control Unit (ECU) maps and results in more efficient engines, with longer operational lives and less impact on the environment. To obtain all the data possible from the HEPS, it is needed an instrument, capable of measuring the speed of the engine, the torque produced, the power consumed, etc - an Engine Test Bench (ETB).

An ETB is an invaluable tool for the designing process of a power plant, particularly when the components are sourced from different manufacturers. Although engine and motor suppliers typically provide torque, speed, and efficiency curves, these often lack comprehensive characterization of the engine or motor [5].

While a typical bench-top dynamometer can cost upwards of €10,000, the electronics and materials used in this system's construction cost significantly less than this value [6]. Given these savings, it is often more practical for budget-constrained laboratories to build

their test benches rather than purchase commercially available models. The dynamometer system discussed here was specifically built to test the power and torque capabilities of a HEPS developed at the Propulsion Laboratory of Departamento de Ciências Aeroespaciais (DCA) while also obtaining other parameters like the fuel flow of the ICE, the electrical current drawn by the EM and the speed of the engine. Another characteristic of an ETB is its versatility, meaning that even though the ETB was designed with a HEPS as the object of testing, it can be also used for testing ICE or EM separately.

## 1.2 Objectives

The main objective of this dissertation is to design and build an ETB capable of test propulsion systems with up to 2 kW of power, record various types of data such as speed of the engine, torque produced and power consumed and deal with electric and fuel energy sources.

To clarify the scope and relevance of the dissertation, the path towards achieving the main goal is divided into several steps, which contribute to the learning process. These steps include:

1. Understanding the operation of an ETB;
2. Enumerating the various types of ETBs while analyzing their benefits and incapacities to justify the chosen type of ETB;
3. Implementing a MATLAB code able to obtain core parameters for the design of the ETB;
4. Improving the building method from an ETB built in the laboratory;
5. Programming a microcontroller to be capable of controlling the ETB and collecting data;
6. Characterizing the performance of a HEPS.

## 1.3 Dissertation Structure and Layout

The present dissertation has five chapters and 4 appendixes written to provide a clear and concise depiction of the work made during the build of the ETB. The first chapter outlines the motivation for this work as well as the objectives to be achieved during the work development.

The second chapter, ”Bibliographic Review”, begins with a description of the most common types of ETBs while highlighting their advantages and disadvantages to facilitate the

selection of the appropriate ETB design. This chapter also covers key data collection concepts used during the operation of the ETB.

In the third chapter, ”Design and Development of the Engine Test Bench”, the selection of the ETB design type is justified, using the advantages and disadvantages described previously. It also explains the MATLAB model developed to obtain essential parameters for the ETB construction. Additionally, the chapter details the development of the data collection procedure together with the integration of various sensors used to guarantee the normal operation of the ETB.

The fourth chapter, ”Assembly and testing of the Engine Test Bench”, begins by describing the assembling process and the issues found during it and how they were solved. It then details the procedure used for testing of an ICE and an EM.

Finally, the fifth chapter offers a reflection on the work carried out, discussing the goals that were accomplished, the challenges that were encountered, and suggesting ways to enhance and extend the study in the future.

# Chapter 2

# Bibliographic Review

Since the invention of the steam-powered engine, there has been a need for testing, initially as a mean for comparing different machines. This can be evidenced by the introduction of a unit of power by James Watt, later known as the horsepower. The Imperial horsepower is the amount of power needed to move 550 lb up to one foot in one second. Watt created this unit due to his need to compare different machines in a more relatable framework [7].

Later, Gaspard de Prony invented what is regarded as the first engine-testing device, the Prony Brake which is still used in some scenarios. This device consisted of a flywheel attached to the shaft of the engine and wooden shoes that created friction on the flywheel when pressed [8].

In the wake of its invention, different types of ETBs were created that mainly differ on the dynamometer used, such as the Hysteresis, the Eddy Current, the Hydraulic and the EM ETBs which will be discussed further on this chapter.

## 2.1 Operating Principles of the ETB

In order to obtain the torque and power curves of an engine, as illustrated in figure 2.1, it is required to obtain the torque, power, and speed data from the engine. While the torque and speed of the engine can be directly measured, the power in the shaft of the engine has to be calculated using the equation below.

$$P_{shaft} = \tau\omega \tag{2.1}$$

Although different types of ETBs serve different purposes, and because of that their characteristics are different, there are some operating principles that are common to all. A good ETB has to be capable of:

1. measure the torque produced by the engine;
2. measure the speed of the engine;
3. control the torque absorbed;
4. dissipate the power produced by the engine [8].

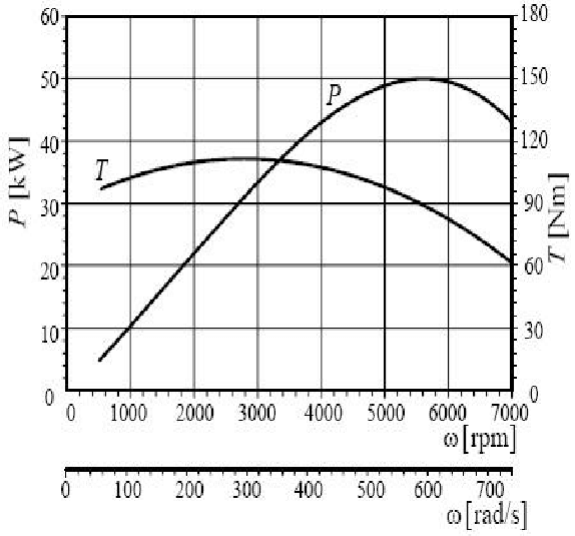



Figure 2.1: Power and torque curves of an engine [9].

Another requirement that must be fulfilled in this ETB is that it has to be capable of determining the efficiency of the engine tested. For that, the next equation must be used.

$$\eta\,[\%] = \frac{P_{shaft}}{P_{consumed}} \times 100 \tag{2.2}$$

Where $P_{consumed}$ is the power consumed by the engine. While $P_{shaft}$ is calculated using the same equation for both ICE and EM, $P_{consumed}$ has to be calculated using different equations, depending if an ICE or an EM is being tested. In the case of the EM the electric power used by the engine can be calculated using the equation below if the EM uses Direct Current (DC) [10].

$$P_{consumed} = P_{electric} = UI \tag{2.3}$$

Where $U$ is the voltage of the circuit and $I$ is the current. To determine the $P_{consumed}$ of an ICE it is necessary to apply the following equation.

$$P_{consumed} = P_{fuel} = \dot{m}_f Q_{HV} \tag{2.4}$$

Where $P_{fuel}$ is the thermal power of the fuel, $\dot{m}_f$ is the fuel flow, and $Q_{HV}$ is the heating value of the fuel.

### 2.1.1 Measurement of torque

Currently, there are two primary ETB architectures, each with its approach to measuring engine torque: the trunnion-mounted design and the in-line shaft design [11].

The trunnion-mounted setup, which has been in use since the era of the Prony Brake, positions the power-absorbing element on bearings aligned with the engine's shaft. Torque measurement in this configuration is managed by a restraining mechanism that applies force tangentially at a set distance from the central axis of the machine [8].

Initially, torque in trunnion-mounted systems was measured using a spring balance combined with weights to counteract the engine's torque, as shown in Fig. 2.2. To ensure the balance consistently measured the tangential force as torque fluctuated, the weights are adjusted to prevent a direction change. This method remained standard until electronic load cells became prevalent in the early 21st century [12].

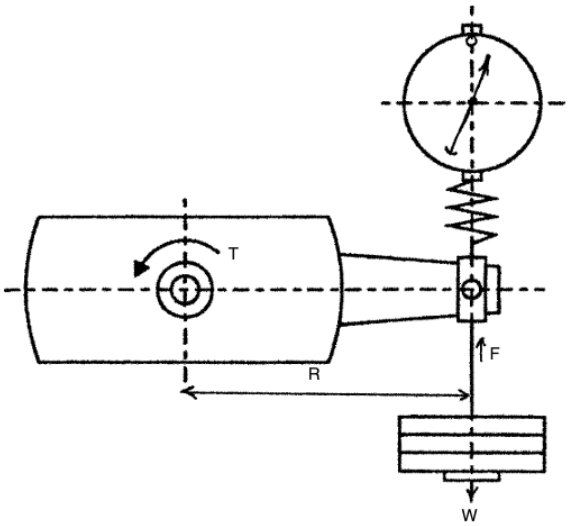



Figure 2.2: Diagram of a trunnion-mounted dynamometer with the use of weights and a spring balance for measurement of torque [11].

There are two main advantages of using a strain gauge together with a bridge circuit and an amplifier, instead of the traditional approach of a balance and dead-weights. First, due to the high stiffness of the load cell, weights are no longer required, as the direction of the force is always tangential to the axis of the machine. However, this also introduces a drawback: the load cell has a finite fatigue life when exposed to high usage cycles [11]. The second advantage is that the load cell enables data to be recorded electronically, which simplifies the data recording and analysis process.

The main disadvantage of using a trunnion-mounted dynamometer is that it requires frequent maintenance and inspection to ensure that the bearings of the structure are still in good condition [8]. These bearings, which are usually either ball, cylinder or hydrostatic bearings are very prone to brinelling, local identation of the races or even fretting due to

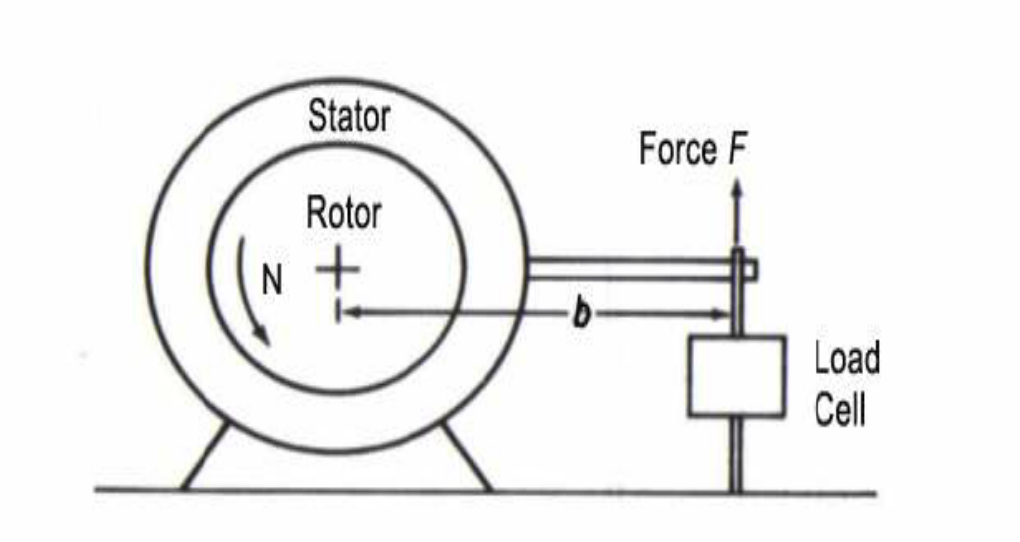



Figure 2.3: Diagram of a trunnion-mounted dynamometer employing a load cell for measurement of torque [13].

the fact that they operate under unfavourable conditions with almost no angular movement [8]. Because of that, it is necessary a periodic inspection and turning of the outer bearing to avoid bad calibration [11].

A Schenk dynamometer or flexural support dynamometer is mounted on flexure support that is fixed to the frames [8] as can be seen in Fig. 2.4. While this mounting reduces the wear and tear of the ETB since there isn't any friction, it has some drawbacks due to the limited torsional stiffness of the flexures and most importantly the ambiguity of the centre of rotation of the axis [11].

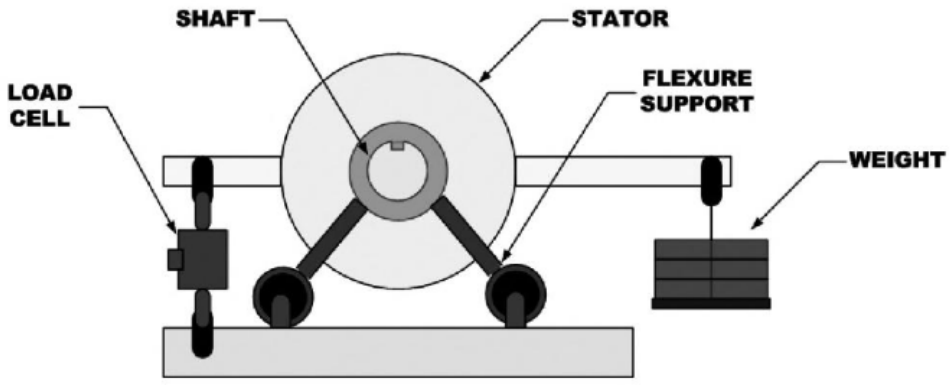



Figure 2.4: Diagram of a Schenk dynamometer [8].

Finally, the fixed mounting or torque shaft dynamometer employs an in-line torque transducer mounted between the brake mechanism and the engine [11] as can be seen in Fig. 2.5. While this system eliminates the oscillation of the cradle assembly and the necessity of torque corrections when studying the transient regime of an engine there are some disadvantages, listed below, that may decrease the accuracy of the dynamometer [11, 14].

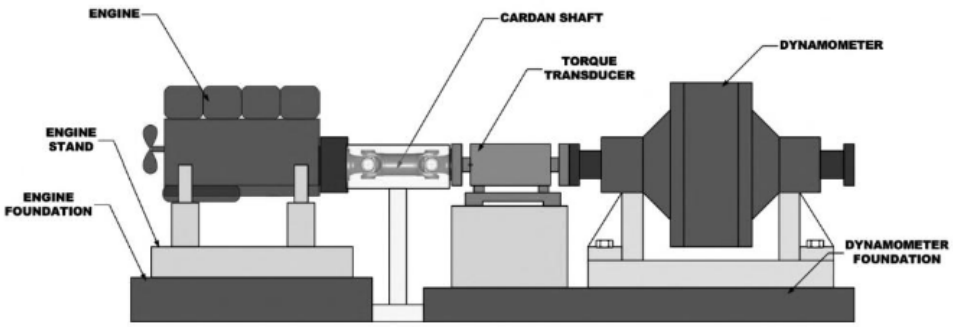



Figure 2.5: Diagram of a fixed mounting dynamometer with a torque transducer [8].

- The torque transducer has to be overrated for the dynamometer because it has to absorb any torque peak of the engine that a trunnion-mounted dynamometer doesn't experience.
- Since the torque transducer is part of the shaft, it is required that the dynamometer is built with high precision. This precision in necessary to avoid any type of bending and/or axial stresses on the torque transducer that could cause any loss of accuracy.
- The calibration of these devices is harder than the the trunnion-bearing ETB since it requires the locking of the shaft and the fixation of an horizontal arm on the shaft.

### 2.1.2 Measurement of rotational speed

There are various methods of measuring the angular speed of an engine in an ETB. One of those uses a toothed wheel and a pulse sensor while others use, for example, an optical encoder system or a Hall-effect sensor [15, 16]. The toothed wheel is a robust and reliable system that can operate well if it is maintained and the gap between the wheel and the sensor is kept as in the design [11]. On the other hand, the mounting and operation of the optical encoder system isn't usually as straight forward as the system described previously. If there is any misalignment between the drive shaft and the encoder it can result in decline of accuracy due to the high number of pulses it uses per second [11]. The Hall-effect sensor consists of a semiconductor in which an electric current is passed through [16]. When a magnetic field passes near the sensor, voltage is induced and a pulse can be detected. When using a shaft with a magnet it is possible to count the revolutions in an interval of time and thus calculate the speed of the engine [16].

With the rise of new bidirectional ETBs capable of operating in all four quadrants (Fig. 2.6), it is important to consider the direction of rotation [11]. While optical encoders can use different sets of tracks to identify the rotational direction, as seen in figure 2.7, with the toothed wheel system it is important to implement a system that can also keep track of the direction of rotation. It is also important for the system designer to clarify which

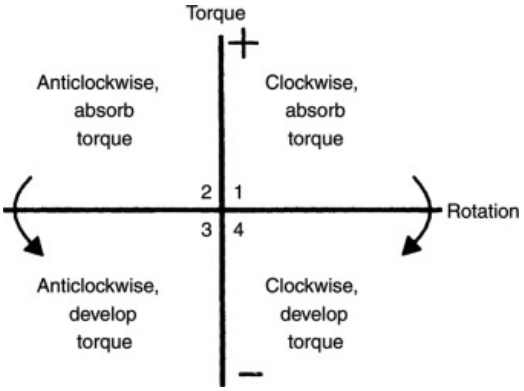



Figure 2.6: Quadrants of operation of a dynamometer [12].

convention was used when designing the ETB so that all operators know it when using the dynamometer.

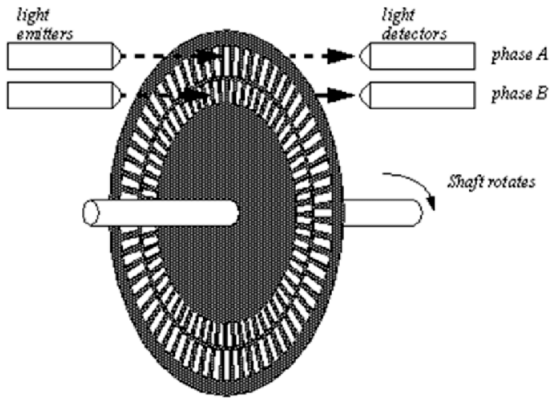



Figure 2.7: Diagram of an optical encoder system with two separate tracks, allowing the sensor to know the direction of rotation [17].

### 2.1.3 Types of dynamometer

The type of dynamometer chosen will influence the way that torque is controlled and absorbed and how the power will be dissipated. There are plenty of types of dynamometers to choose from. Since this is one of the most important choices to be made during this project, this section presents the most used types of dynamometers.

#### 2.1.3.1 Friction-based dynamometers

These machines, as the name suggest, use the friction generated between two parts to brake the engine, control, and absorb the torque of the engine while dissipating the power through heat generated in the process [15]. The term ”Friction-based dynamometers” encompass a lot of different dynamometers, such as the Prony brake, that was mentioned before, or even the rope brake, which uses a rope that creates friction around a drum [8].

Nowadays, the most used friction-based dynamometer consists of series of brakes that are water-cooled to prevent heating (Fig. 2.8). This type of dynamometer is useful due to its simplicity, low-cost, and also have the advantage of generating torque at low rotational speeds [11].

There are however some caveats of using this kind of dynamometers. Due to being based on friction, there is a lot wear on the dynamometers, which brings the necessity of regular replacement of the brakes. Another drawback is that without cooling it is impossible to have long tests due to the large amount of heat generated during the testing process [15].

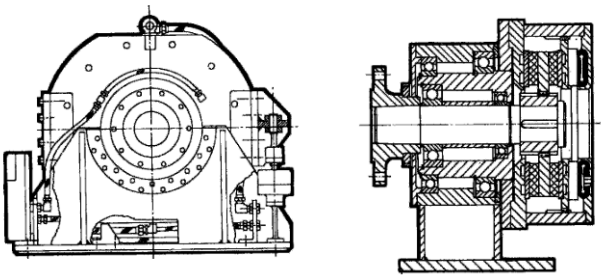


Figure 2.8: Diagram of a water-cooled brake utilized as a dynamometer [11].

#### 2.1.3.2 Hydraulic Dynamometers

Hydraulic dynamometers, also known as water brakes or hydrokinetic dynamometers, consist of a rotor connected to a shaft that is turning at the speed of the engine [8]. The rotor creates, together with the stator inside the water-tight casing, a set of toroidal recesses. These recesses are divided into different pockets due to radial vanes on the rotor, as can be seen in Fig. 2.9. With the rotation of the rotor, a transfer of angular momentum from the rotor to the stator happens and torque is absorbed [18].

During this process, a high amount of turbulent shear in the water is created, and as a result, the water heats up which dissipates the power transferred [11]. This dynamometer is usually designed in a way that the power is absorbed with minimal damage to the rotor and the stator, even with all the cavitation involved in the process. To achieve this, passages in the rotor that vent the centre of the vortex to the atmosphere are employed [18].

Two kinds of hydrokinetic machines use the principles previously explained but differ in the way that the torque is controlled. These are the constant fill and the variable fill machines [8].

##### Constant fill machines

Constant fill machines, or sluice plate design machines, were first designed by William Froude in 1877 [18]. In this type of dynamometer, the torque is controlled by a sluice gate which moves back and forth either manually or controlled by a motorized mechanism [8]. The movement of the gate exposes or hides the vanes of the rotor from the vanes of the

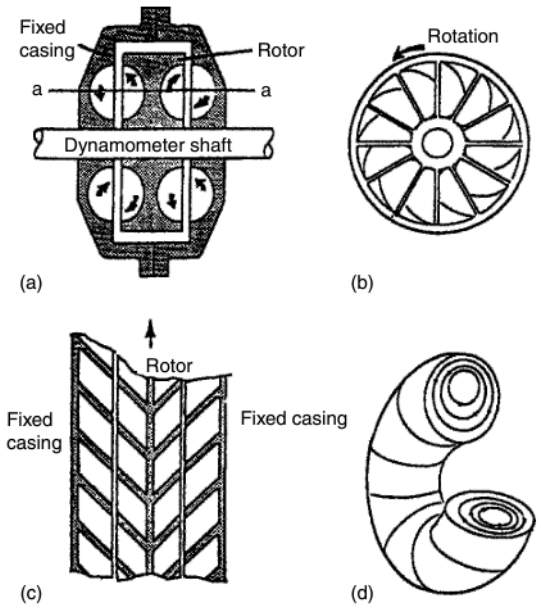



Figure 2.9: Diagrams of the operating principle of a hydraulic dynamometer: (a) dynamometer cross-section; (b) rotor end view; (c) unwrapped rotor & casing section; (d) toroidal vortex formation [11].

stator, which controls the power and torque absorbed [12].

### Variable fill machines

In this type of machine there are no sluice gates but at least two valves, the inlet and outlet valves, which control the mass of water inside the machine and in turn control the torque absorbed by the dynamometer [8]. The main advantage of this design is the fact that the response from this dynamometer is faster than the response from sluice gate machines [11].

In both cases, there is, however, the need for sufficient water flow for the system. It is needed to constantly change the water inside the casing, due to the heating happening during the process, which could result in a loss of efficiency from the dynamometer [18]. Another drawback found in these machines is that good isolation is needed to avoid water leakage from the casing, which increases the complexity of the design and assembling of the dynamometer [8].

There is another type of hydrokinetic dynamometer which is the disk dynamometer. This machine consists of a series of flat discs located in between flat stators with a very small clearance [11]. In this design, torque is created by the movement of water and the consequent shearing between the fluid and the discs. In these machines as in variable fill machines torque is controlled by the variation of the mass of water in the casing. However, these machines work poorly at low speeds, being more suitable for high-speed appliances, such as gas turbine loading [11].

#### 2.1.3.3 EM powered dynamometers

This type of dynamometers makes use of an EM for absorbing power and torque by converting the mechanical power in the shaft into electricity [11]. The electricity is then transmitted into a load circuit, which can be controlled manually or automatically [16]. During the test, the motor will inevitably heat, making it necessary for the designer to think of a cooling process, which is commonly a forced airflow over the motor [16].

The torque control in these machines is achieved by a load circuit. Usually, the circuits consist of variable resistors capable of operating under big currents and high temperatures. By varying the resistance of the circuit, different loads are applied [16]. Some advantages of an automatic load circuit are that it takes some working load from the operator and can be used to test certain types of operations that wouldn't be possible with manual loads.

One of the main advantages of these dynamometers is that they are capable of motoring, and with that, starting the engine that is going to be tested [8]. This characteristic also allows them to operate in all four quadrants of operation [16]. Another advantage is the fast response times the electric motors have, which is becoming more important nowadays with the increased interest in transient testing [11].

There are two types of EM-based dynamometers, the DC and the Alternating Current (AC) machines [16]. The difference between both is that one of them uses a DC-powered motor and the other an AC motor. It is worth mentioning that DC motors typically have higher inertia, lower maximum speeds and lower rates of speed change compared to recent AC motors. DC motors also have a commutator which might increase the amount of maintenance required [11].

#### 2.1.3.4 ECB dynamometers

ECB dynamometers make use of Eddy currents which are circular flows of electromagnetic currents generated when there is relative movement between the conductor and a magnetic field [11]. It usually consists of a thin high-permeable metallic disk connected to the engine shaft that rotates through a magnetic flux created by one or more coils with their associated core's pole shoes [19], as can be seen in Fig. 2.10.

This movement creates a series of Eddy-currents on the disk, which retards the rotational movement it has, creating a counter-acting torque which can then be used to determine the curves of the engine [19]. During the use of the ECB the mechanical power absorbed is dissipated in the form of heat in the disk [21].

ECB machines have various advantages, such as the low noise produced, the lack of wear since there is no contact between the parts, the fast response time, the load uniformity once the disk temperature stabilizes, and lower maintenance costs [19]. However, as with

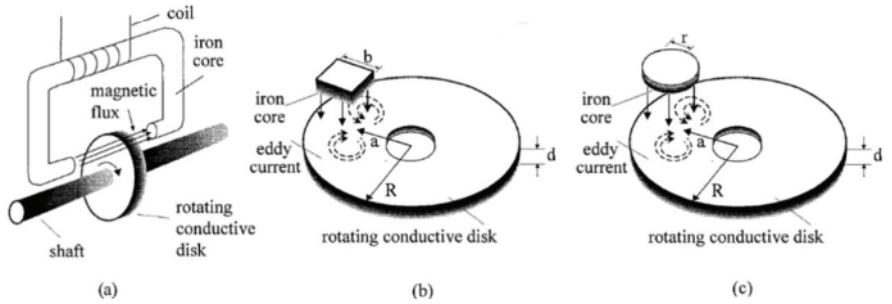



Figure 2.10: Diagrams of the principles of operation of an ETB: (a) Diagram of an ECB; (b) and (c) cross view of the disk and the iron core [20].

all systems it also has some disadvantages, like being dependant on the velocity of motion, the magnitude of the magnetic field, and the high temperatures reached during test [19].

The second disadvantage can be surpassed by varying the current on the coils and, with that, control the amount of torque absorbed [19]. However, speed is directly proportional to the torque available, meaning that when the speed is $0$ there is not any torque [22]. This in turn means that an ECB can not be used for stopping the engine. The last disadvantage can be surpassed by the employment of some kind of cooling system. Because of that there are two main categories of ECB [8, 11]:

- Wet-gap ECB: in this type of ECB, the gap between the core pole shoes and the rotating disk is submerged in a cooling fluid;
- Dry-gap ECB: in this type of ECB, the cooling of the disk is indirect which can be done by forced airflow or cooling passages filled with water or other cooling fluid.

The wet-gap ECB will suffer from corrosion from the cooling fluid, which will increase the need for maintenance while also increasing the complexity of design and assembling.

### 2.1.4 Quadrants of operation of a dynamometer

A key consideration when selecting a dynamometer is its operational quadrant capability. As it was described previously, while some dynamometers, like the electric-motor based dynamometers, are capable of absorbing and creating power by motoring, other are only capable of absorbing power. Other thing to take into account it's the direction of rotation [11]. Most engines, inclunding those used in airplanes or UAV run in a clockwise direction when seen form the shaft [12].

As it is depicted in figure 2.6, dynamometers can be classified by the number of quadrants they can operate in. Most dynamometers can be used in the first and second quadrants. There are, however, some exceptions, like hydraulic brakes, due to the shape of the rotor. By fitting a shaft on both ends of the casing this difficult can be surpassed [11].

Some of the advantages of having a dynamometer capable of operating in all quadrants are the following [11].

- the dynamometer is capable of starting the engine being tested;
- the dynamometer can be used to study mechanical losses;
- dynamometers capable of four quadrant opration are more fit for transient testing due to their ability of fast load changes and sometimes even torque reversals.

### 2.1.5 Dynamometer envelope of operation

From the description given in 2.1.3 of the different dynamometer types, it can be understood that when designing an ETB the type of brake device used has to be chosen based on the type of testing that is going to take place.

For example, it is advisable, for best accuracy, to select the smallest dynamometer capable of handling the engine that will undergo testing. This leads to lower inertia of the brake, which is better specially for transient tests, since the torque corrections will be lower [11].

It is also necessary to analyse the maximum torque and power-speed curves of the dynamometer, to make sure that the dynamometer is capable of testing the engine at all speeds [16]. Figure 2.11 illustrates the envelope of a hydraulic brake, the maximum torque, and power-speed curves of an engine. As it can be seen, the engine's maximum torque and power are outside the dynamometer envelope in the low-speed regime, which means that this dynamometer wouldn't be capable of testing this engine.

The elements that limit the dynamometer envelope are as follows [16].

- Maximum shaft torque allowed;
- Maximum power permitted;
- Minimum torque absorbed;
- Maximum speed allowed.

While some dynamometers are not limited by the minimum torque, like EM-based ones, others are, which will lead, inevitably, to different envelopes that have to be analysed.

## 2.2 Engine testing modes

Defining the intended testing modes is crucial during ETB design to achieve optimal outcomes. The suitability of a dynamometer for various testing modes is a key consideration

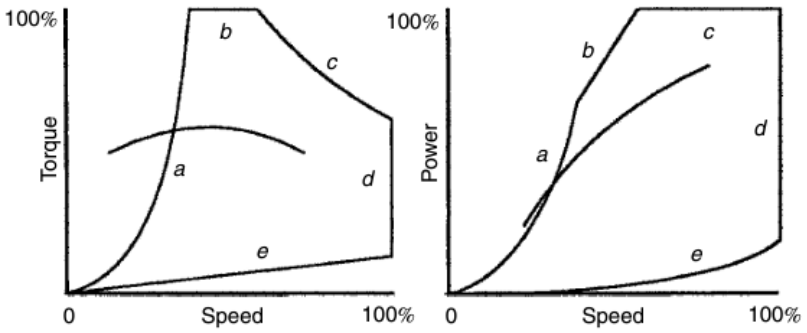



Figure 2.11: Engine maximum torque and power speed curves plotted over hydraulic brake envelope [11].

and if that isn't defined, the ETB could end up not complying with its objectives. Until the beginning of the 21st century most of ETBs were only capable of steady-state testing [11]. However, with the rise of the need for emission testing in the transient regime, machines capable of dynamic testing and steady-state testing were developed.

Despite their differences, both the steady-state and the dynamic test programs are sequences of values of engine torque and speeds [23]. Because of that, the only variables that will define the outcome of the test are the engine's throttle and the position of the torque setting of the dynamometer [11].

It is also worth noting that both the engine and the dynamometer will have different torque-speed curves for each position of the throttle and the torque setting. It is the interaction between the curves that will determine the stability of the engine-dynamometer pair and subsequently dictate if the test of a given engine is viable on a dynamometer [11]. The engine throttle and dynamometer setting can be controlled using various modes, as outlined in the following section [11].

- Engine throttle:
  - position mode - used for maintaining a throttle setting;
  - speed mode - used for maintaining a constant rotational speed;
  - torque mode - used for maintaining a constant torque.
- Dynamometer setting:
  - position mode - used for maintaining the same control setting;
  - speed mode - used for maintaining the same speed;
  - torque mode - used for maintaining the same torque;
  - power law mode - used for reproducing a certain torque-speed characteristic.

Almost all combinations of engine throttle mode/dynamometer setting mode are possible, and will be presented next.

### 2.2.1 Steady-state testing

During steady-state testing, the engine and the dynamometer both must find a stable state where data can be acquired [11]. This can be achieved using various combinations of the modes presented in 2.2.

It is important to ensure that the state is stable during engine testing because, if not, the engine might run away or start hunting, resulting in an unusable test [11]. Fig. 2.12 illustrates two cases of torque-speed curves in different test settings. In (a) the angle at which both curves meet is almost perpendicular ensuring that the test is stable and that there is almost no chance for the engine to leave this state. However, in (b) the angle at which both curves meet is much shallower which is a signal that a test in these settings may not be viable.

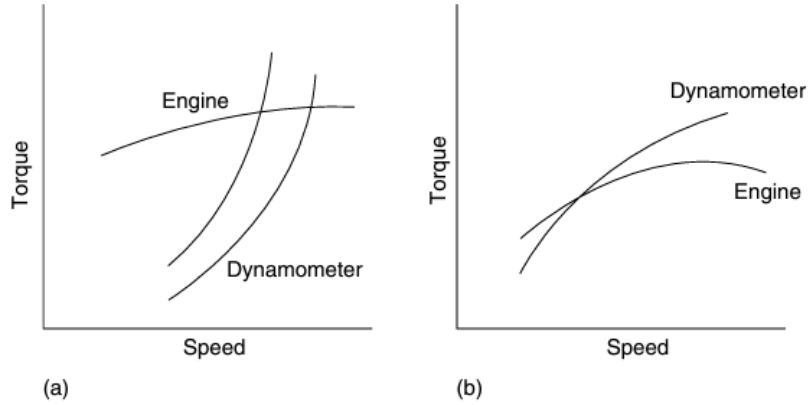



Figure 2.12: (a) Stable hydraulic dynamometer in position mode; (b) Unstable hydraulic dynamometer in position mode [11].

#### 2.2.1.1 Position/position mode

This is the most basic mode of testing with both the throttle of the engine and the torque setting of the dynamometer being set independently as illustrated in Fig. 2.12. This control mode is an open-loop system since there is no feedback between the engine and the dynamometer [11].

Since this mode can be used for finding the stability of the test bench and the engine, it is very useful when there is the need to find faults [11].

#### 2.2.1.2 Position/speed mode

In order to operate to operate on this mode, the dynamometer requires some type of controller to ensure the speed of the engine stays the same despite the throttle position [11]. The graph of the torque-speed curves is displayed in Fig. 2.13. (a) . As it is possible

to conclude, the dynamometer is capable of adapting and changing its torque setting to maintain the same speed. This mode is mostly used for plotting the torque-speed curves of the engine at the same speed, and it is usually stable, since it ensures that the speed of the engine never changes [11].

#### 2.2.1.3 Speed/torque mode

This mode is mostly used when testing a new engine due to its inherent safety. On a new engine it is usual that as the friction decreases, its power starts to rise, which increases speed since the torque is set on the dynamometer, as can be seen in figure 2.13. (d) [11]. To avoid damage to the engine, the speed controller will close the throttle ensuring that the engine's speed doesn't rise, which could result in damages.

#### 2.2.1.4 Position/torque mode

This mode is mostly used on governed engines, due to their "drooping" characteristic. This leads to a drop of RPM as the torque increases, which means that the speed mode of the dynamometer isn't suitable for this kind of testing. Because of that, the dynamometer must be set to a fixed torque as illustrated in Fig. 2.13. (c) and the machine will always absorb this set value [11]. It is important to note that this mode can lead to stalling the engine, which must be avoided, to avert any damages.

#### 2.2.1.5 Position/power law mode

In this mode, the dynamometer is programmed to give a torque-speed curve in the following form [11].

$$\tau = K\omega^n \tag{2.5}$$

Where $K$ is a constant of the system. When $n = 2$ the system behaves like a marine propeller [11]. This means that this mode is particularly useful when testing ship engines. It is also worth noting that this mode is safe, since the dynamometer will always keep the engine in a pre-determined torque-speed curve [11].

### 2.2.2 Dynamic engine testing

Dynamic engine testing is used when the command values of the test are changed, based on time. Contrary to most of the modes presented previously, this type of engine test-

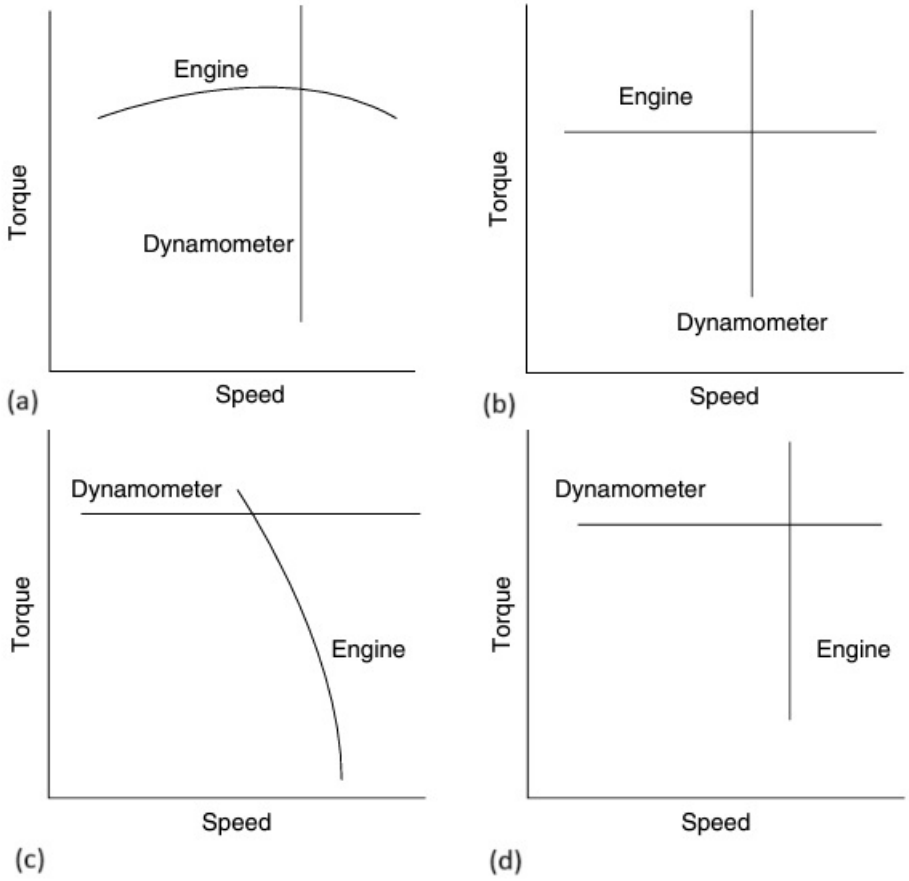



Figure 2.13: (a) Position/speed control mode; (b) Torque/speed control mode; (c) Position/speed control mode; (d) Speed/torque control mode [11].

ing involves a closed-loop, this means that there is feedback between the engine and dynamometer controllers, which will influence the test [23].

ETBs capable of dynamic testing can be classified into two different categories, presented below [23].

- Transient test stand;
- Simulation test stand.

These two types differ in how the test is conducted. While in the transient test stand the values are preset, and the test is only carried out to see the response of the engine, in a simulation test stand the values are determined with the help of a computer model, and can be changed during the test [23].

Not all dynamometer types are suited for dynamic testing. Nowadays, the most used dynamometer type for dynamic testing is the AC motor, due to their lower inertia and fast response times [11].

In dynamic testing, the inertia of the dynamometer system plays a role since it will influ-

ence the torque measured by the load cell. With digital systems this drawback is easy to surpass using the following equation [11].

$$\tau_1 - \tau_2 = \frac{2\pi N I}{60^2} \tag{2.6}$$

In this equation, $\tau_1$ is the torque on the shaft of the engine, $\tau_2$ is the torque registered by the load cell, $N$ is the speed of the engine, and $I$ is the inertia of the brake system.

As with steady-state testing, there are also control modes for dynamic testing [11]. The one presented below is, however, the most used.

#### 2.2.2.1 Torque/speed mode

This control mode, illustrated in Fig. 2.13. (b), is useful when dynamic testing an engine, since it allows the shaft to be continuously at the same speed while varying the throttle.

This situation can be encountered by an engine when, for example, an airplane faces a stronger headwind, or when a car is climbing up a hill and it needs to develop more torque, while keeping the same speed [11].

## 2.3 Data acquisition system

As with all ETBs, it is necessary to develop a system to acquire the data from the engine. In this case, apart from the torque and speed, which will then be used to calculate the power, it will also be necessary to acquire other data such as fuel consumption to characterize the engine. First of all, however, it is necessary to understand how can the data be acquired.

In the past, as explained previously, dynamometers relied on a set of analogue instruments, such as the spring balance to measure all the necessary data [8]. This required manual recording of all measurements by the operator, which could decrease the accuracy of the test and increase the workload of the operator. Nowadays, however, the industry standard is the use of electronic components to record the data in the form of signals. If done right, this method is much more accurate and puts less work on the human operator.

### 2.3.1 Signal

Signals can be defined as functions that carry some type of information [24]. While not all signals have to be electrical, in this study we are going to focus on them, since all signals used in this work will be electrical. For electric signals, most vary some properties, such as current or voltage, as a function of time [16]. Even though it is possible to classify signals

based on a lot of properties, it is usual to classify them into two different categories, based on their discretization. Based on this criteria, signals can either be time-continuous, also known as analogue signals, or time-discrete signals [24].

While, as the name states, analogue signals are defined for all values of the time variable, time-discrete signals are only defined in some instants [24]. The period between those instants may vary depending on the system.

### 2.3.2 Transducers

Transducers are used worldwide because of their ability to transform a form of energy into another.A common example of a transducer is a microphone, which converts sound waves into electrical signals. This transformation, however, doesn't have to be in only one way, another form of transducers are sound speakers, that transform electricity into sound [24].

In the case of the system being developed, most of the transducers used are sensors. Sensors react to a physical phenomenon, by creating an electric signal that can be used to interpret that physical process. On the other hand, if it is needed to transform an electric signal into a physical phenomenon, then the use of an actuator is required [25].

### 2.3.3 Sampling

When analysing phenomena using a computer, it is physically impossible to store all values of a certain measurement, like speed, due to the finite memory that the computer has. Because of that, it is necessary to sample the signal. Sampling is usually done at a constant interval throughout the global time of the study [24].

As it was said in 2.1.2, when designing a system, careful consideration of the sampling interval is crucial. If it is too short, it can be reductant and result in the usage of unnecessary memory. On the other hand, if it is too long, resolution can be lost and important data can be lost. This phenomenon is called aliasing, and is illustrated in figure 2.14. In this case, the top signal was recorded with the right sampling, interval while the bottom one is undersampled resulting in a reconstructed signal with a much lower frequency than the original one [26].

To avoid, this the Nyquist-Shannon theorem can be used. This theorem states that, when sampling a signal, the samples need to have, at least, a larger frequency than twice the maximum frequency in the original signal. Using this theorem it can be assured that there hasn't been any loss of relevant information, and that it is possible to reconstruct the original signal using the samples [26].

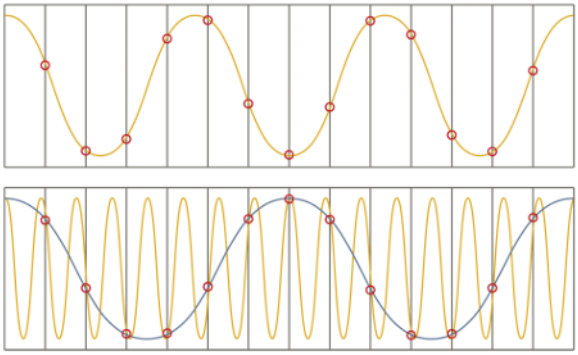


Figure 2.14: Aliasing [26].

### 2.3.4 Quantization

When sampling a signal it is also normal to quantize the values of the samples. The reason behind this is the same as the reason behind sampling, memory is not infinite [24]. Quantization can be defined as the process of segmenting a continuous quantity into a countable, finite number of sub-units, typically of a uniform magnitude [27].

The problem with this is the same as with sampling, if a number is quantized with too much detail, then it is wasting computer memory. Conversely, approximating a number with fewer digits risks losing important details, particularly if subsequent calculations rely on the quantized value, specially if there are calculations using the number quantized [27].

It is also important to note that, when a signal is sampled and quantized, it can be referred to as a digital signal [24]. This processes have to be taken into account when designing a system because they will affect the accuracy of the system. This, in turn, could be the difference between the data being reliable or unreliable.

### 2.3.5 Layout of cables

Notably, a crucial yet often overlooked aspect of system design is the cabling layout, which significantly impacts accuracy and error rates. Sensor cables, which preferably have the shortest possible lengths, are used to transmit signals in ranges of voltage from 0 to 10 V and DC currents from 0 to 5mA usually. These values are almost nothing, when compared to the voltages and currents across power-supply cables [11]. Because of this discrepancy, it is vital to have an organized layout, with the minimal length of cable possible, to avoid loops or coils [12].

Without this concern in mind, it is possible to cause electrical interferences caused by the different currents and voltages run on different wires. Electrical interferences can be categorized into three main types, which are discussed below [11].

- Inductive interference, which is caused by the magnetic flux created by electrical

currents in one conductor that induce voltages in a conductor next to it. This can be reduced by not running power cables near signal cables, using different cable trays for different types of cable and/or twisting cables of devices, that use dual connection, which can reduce up to 25 times the inductivity and using shielded cables.

- Capacitive interference can occur when cables with different voltages are next to each other. It can be reduced by the use of shielding on cables, not running power and signal cables next to each other or by separating cables with different voltages.

- Electromagnetic interference is caused by many noise-emitting sources, such as spark plugs. This can induce both voltage and current fluctuations across conductors, potentially disrupting system performance. To avoid this, the layout of the cables in trays is the most important countermeasure and should be done as illustrated in Fig. 2.15. Another countermeasure is the employment of a separate tray for signal cables, since the currents and voltages on the conductors are much less than the voltage on power cables.

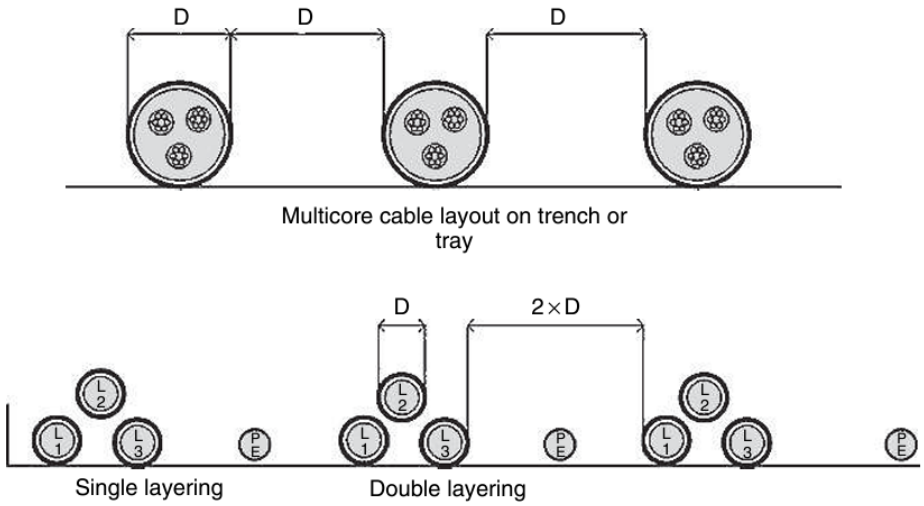



Figure 2.15: Suggested layout of cables on a metal tray [11].

### 2.3.6 Data Filtering

Since this project works will work with a few analogue sensors, it will have to deal with some degree of noise on the signal. Excessive noise can render data unusable, making filtering a crucial step on data analysis [11]. This can be done with hardware or software filters [28].

In the case of software filtering, various filters can be applied, among those the low-pass, the median and the moving-average filters [28].

#### 2.3.6.1 Median filter

A median filter is one of the simplest filters to implement [28]. In this filter, noise is assumed to be a pulse. The filter applies the following algorithm:

1. an array is created. The size of this array is defined by the user and it is called a window array;
2. the window array is filled with values from the sensor;
3. the median value of the array is selected [28].

One of the strengths of this filter is that it doesn't have a great delay when big changes of values in the sensor happen, specially if the window array doesn't have many elements [28]. The size of the window greatly influences the result. If the window array is too big, a lot of data is lost, while if the value is too small the filtered values can become too similar to the raw data, which can be an issue, if there are a lot of spikes on the unfiltered data [28].

#### 2.3.6.2 Moving Average Filter

The moving average filter uses a logic similar to that of the median filter. However, instead of the filtered value being the median of the window array, the filtered value is the average value of the window array [29]. This results in a faster process of filtering, since there is no need to sort the window array into an ascending or descending order [28].

In this filter, the same happens, as with the median filter, if the window size is too big, a lot of data loss will happen, so it is important to keep this in mind when choosing the window size [29].

#### 2.3.6.3 Low-Pass Filter

This filter, based on an integrator, removes irregularities and some oscillations to the signal [28]. Its biggest advantage is the fact that it is one of the fastest filters being the fastest of the discussed here. The simplest form of this filter is the one below [28].

$$y^f_{(i)} = (1 - K)x^{nf}_{(i)} + Ky^f_{(i-1)} \tag{2.7}$$

Where $K$ is the coefficient that determines the smoothing, $y^f_{(i-1)}$ is the previous filtered value, $x^{nf}_{(i)}$ is the current non-filtered value and $y^f_{(i)}$ is the current filtered value [28]. It is important to note that with a higher $K$ value, more data will be lost and the filtered signal becomes smoother. The main weakness of this filter is the fact that it creates a "shift".

This means that, when there is a big difference in the un-filtered values, the filter takes a long time to adapt, which can be unacceptable for some types of sensors.

# Chapter 3

# Design and Development of the ETB

## 3.1 Development of the braking system

### 3.1.1 Choice of the braking device

In order to choose the braking device, a decision table was used. For that, it was decided to evaluate all types of dynamometers presented in 2.1.3. The evaluation was based on the most important criteria for this project, which will be presented below. Then each type of machine was classified from 1 to 4 in that criteria, with 1 being the worst classification and 4 the best. At the end, all the classifications from the different criteria were summed up and the chosen dynamometer was the one with the higher classification.

Even though some criteria such as the necessity of cooling or the quadrants of operations apply to other projects, some of the criteria were evaluated according to the necessities of this project.

#### 3.1.1.1 Number of quadrants of operation

This criterion was used to classify the different dynamometer types into the number of quadrants in which they are capable of operating. As mentioned in 2.1.4 this number is important since it can reduce the number of outside equipment needed to test an engine such as a starter motor. It also helps when transient testing and studying losses of the system. Because of that, it was considered a necessity to consider this when deciding the type of dynamometer to use.

#### 3.1.1.2 Cooling needs

The necessity of cooling is a decisive factor when building an ETB because it can add a lot more complexity to the assembly and design. While in some cases there is only the need to change the water in use such as in hydraulic brakes, in other cases it can imply forced airflow such as with the electric motor applications or even more complex systems as the ones used in some ECB. In the case of the Prony brake there is a need for cooling due to the brake fade that happens after operating it for a long time [30]. Since this can be solved by the use of forced airflow it was decided to give the same classification to the Prony brake and the electric motors in this topic.

#### 3.1.1.3 Environmental cost

The environmental cost is based on both the amount of material that is wasted and the energy dissipation method by the dynamometer. In this field, both the hydraulic brakes and the Prony brakes waste a lot of material with the Prony brake needing a lot of component replacement due to wear. The hydraulic brake also wastes a lot of water since there is always the need to exchange the water to avoid overheating.

Only the AC and DC electric motors are capable of dissipating its energy through a method that allows its usage afterwards leading to a lower environmental cost. In these cases, the generated electricity can be redirected to the grid or into a battery which allows its usage afterwards. All the other types of dynamometers dissipate their energy through heat which results in energy wasting, raising their environmental costs.

It is important to note that in certain criteria like this, the classification might not be straightforward. In this case, a 4 doesn't mean having the higher environmental cost but the best scenario which is a lower environmental cost.

#### 3.1.1.4 Empirical knowledge

This criterion was taken into account due to previous works done in the lab that could help with the design and assembly of the ETB. As referenced in this work, *Alexandre Nunes* and *Carlos Novo* [16, 19] both worked on two separate ETBs and acquired knowledge during their work. It was also taken into account the work of *João Farinha* [31] on the development of an ETB using an AC electric motor.

These works help by avoiding certain known issues already found when they were done and because of that, this knowledge was considered.

#### 3.1.1.5 Financial Cost

The final financial cost of this project is an important matter as stated before. The ETB is being developed in-house to avoid paying thousands of euros for an already-built ETB. Because of that, it is necessary to weigh in the cost of the project.

While the Prony brake is cheap to assemble due to the low prices of its components, the electric motor dynamometers are more expensive due to their high prices. In the case of the hydraulic brakes, the high price comes from the high precision needed to build the rotor and stator. The ECB however doesn't have a high price due to a lot of in-stock materials such as the iron used for the core.

#### 3.1.1.6 Simplicity

The simplicity of a project is of high importance since it can work to eliminate possible failure points thus working towards a more efficient and reliable outcome. Due to that, it was deemed important to evaluate the simplicity of all the dynamometers and consider it when making the final choice of dynamometer.

In this case, the Prony brake was considered the simplest type due to the low number of parts involved. The hydraulic brakes were both considered the most complex types because of their high number of parts and the involvement of fluids which could result in leakage. Both EMs were considered simpler than the ECB because they basically consist of the motor and its case while the ECB involves the coil, the core and the distance between the pole shoes and the disk. These things have to be done with some precision which increases the complexity of the machine.

#### 3.1.1.7 Availability of materials

As stated previously, if the materials needed are already in stock, the cost of the project will be lower and the assembly will be faster. Due to that, the ECB has the higher classification in this topic because most of the materials needed were available at the laboratory. For the other types, some of the materials were available in the laboratory while others weren't and because of that all of the other types have the same classification.

#### 3.1.1.8 Maintenance required

Maintenance can be a big part of the ETB and because of that it is mandatory to at least evaluate it when assembling one. Most of the maintenance necessary for a machine like an ETB is changing worn-out parts, inspecting moving parts such as the rotors and bearings and in the case of DC motors inspecting the built-in commutator.

Considering this it was deemed that the ECB and the AC EM brake had the least amount of necessary maintenance since the only maintenance needed is the regular inspection of bearings. The other types, however, need more maintenance with the Prony brake being the one with more wear on the brake pads needing a lot of replacement of these parts. The DC EMs also have a higher degree of maintenance due to the commutator while the hydraulic brakes need more inspection of the joints, valves, rotor and stator due to the use of a liquid.

#### 3.1.1.9 Cost of maintenance

It is also important to consider the cost of the maintenance necessary for which type. While most of the maintenance required for the Prony Brake will be the change of the

pads or the disks, thus being relatively cheap, for the hydraulic brakes or the EMs the maintenance can more expensive, increasing their operational cost. The ECB was considered to have a middle cost of maintenance since most of it will be lubricating bearing but in the case there is the need for changing the power source or the coil, the cost will go up.

With all criteria evaluated, Table 3.1 was created.

Table 3.1: Braking device decision table.

| Criteria | Prony Brake | Constant fill machine | Variable fill Machine | AC electric motor | DC electric motor | ECB |
|---|---|---|---|---|---|---|
| Number of quadrants of operation | 2 | 2 | 2 | 4 | 4 | 2 |
| Necessity of cooling | 3 | 3 | 3 | 2 | 2 | 2 |
| Environmental cost | 1 | 1 | 1 | 4 | 4 | 3 |
| Empirical knowledge | 3 | 1 | 1 | 3 | 2 | 4 |
| Financial Cost | 4 | 2 | 2 | 1 | 1 | 4 |
| Simplicity | 4 | 1 | 2 | 3 | 3 | 2 |
| Availability of materials | 2 | 2 | 2 | 2 | 2 | 4 |
| Maintenance required | 1 | 1 | 1 | 3 | 2 | 4 |
| Cost of maintenance | 3 | 2 | 2 | 2 | 3 | 3 |
| **Total** | **23** | **15** | **16** | **24** | **23** | **28** |

With this table in mind, it was decided to opt for an ECB dynamometer. As stated in 2.1.3.4, ECBs work by passing a magnetic field on a moving high-permeability disk. This induces Eddy currents on the disk which counteracts the movement and creates torque. For the design of the brake, it was deemed necessary to develop a MATLAB model which could help obtain certain parameters necessary.

### 3.1.2 MATLAB model

To model a phenomenon it is necessary to first understand the theory that goes behind it. In this section the theory used to create the model will be presented first and then the algorithm to implement it into a MATLAB code.

ECBs have two different domains which are separated by a peak region, as illustrated in Fig. 3.1, explained below [21, 32–34].

- The low-speed domain in which the breaking torque is directly proportional to the rotational speed, even though this relation decreases as the speed increases. In this region, the magnetic field induced by the Eddy currents is very small when compared with the original magnetic field.
- In the critical speed region the induction caused by the eddy currents, $B$ can not be ignored anymore when compared with the magnetic induction at zero speed $B_0$. In this region, the disk has the most breaking torque and the abscissa of the peak point is called the demagnetization speed.

- When the brake is at its high-speed domain, the braking torque is inversely proportional to the speed. In this region, the mean of induction decreases even further and at infinite speed the magnetic field will be cancelled by the induction caused by Eddy currents.

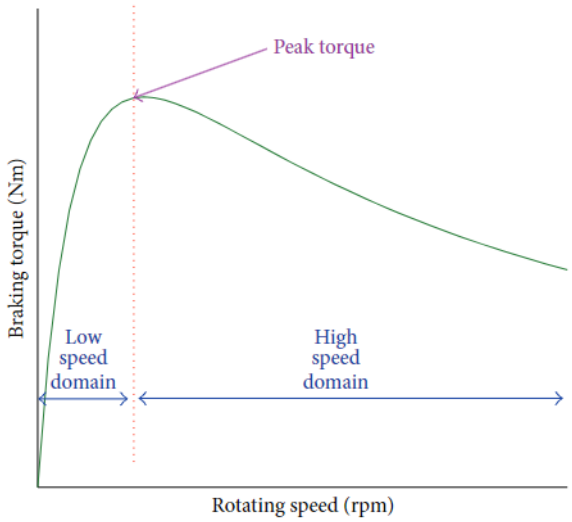



Figure 3.1: ECB torque-speed curve [32].

#### 3.1.2.1 Requirements for the ECB

Before understanding the mathematical model behind the Eddy current brake it is necessary to define some parameters of the ECB.

1. It was decided to use a pulley and belt system to guarantee that the shaft has always enough torque to avoid stalling of the engine. For this case, every time the shaft of the engine does 4 revolutions, the brake shaft does 1 revolution.
2. It was also decided that since most small engines have a high idle speed (around 3000 rpm [35]), it was decided that the core should not be saturated over 375 rpm of the brake, corresponding to 1500 rpm of the engine.
3. To avoid any kind of torsion stresses, the material of disk was decided to be nonferromagnetic.
4. The material of the core should be Iron which has a saturation limit of around 2 T [19]. As a safety factor, the saturation limit counted was of 1.5 T.
5. The ECB must be able to dissipate 2 kW of power at 2000 rpm. This value as obtained due to the engine that will be tested on the dynamometer. When modelling it was taken into account a safety factor of at least 1.25 to prevent any kind of hysteresis losses or higher harmonics [19].

6. The iron core will have a square shape. Comparing to a round core, the square pole shoe guarantees a higher breaking torque while keeping the same critical speed [19].

#### 3.1.2.2 Mathematical model

Although many models have been created to described the physics behind an ECB, this work will use the model developed by *Zhou et all* [32] since it takes into account the demagnetization as well as the temperature of the disk. While initially it was intended to have 2 separate coil-core systems it was later decided to use only one since it was determined that a single system could dissipate the power intended. This option can bring some balancing issues but it also lowers the number of parts and increases the simplicity of the system. Because of that, it was decided that the dynamometer would consist of a single rotating disk with only one C-shaped electromagnet. Before delving into the mathematical model it is necessary to make some assumptions.

1. The cross-section area of the magnetic flux in the disk is equal to the cross-section area of the pole shoes;
2. The temperature of the disk doesn't vary with time;
3. The disk thickness is minimal concerning the disk radius, meaning that the reluctance in the radial direction is much more significant than the reluctance in the axial direction ($\mathfrak{R}^*_{d_{ra}} >> \mathfrak{R}^*_{d_{ax}}$). This means that all magnetic flux can cross the disc in the axial direction.
4. The magnetic flux density doesn't increase further after reaching saturation.

With these assumptions in mind, the equivalent magnetic circle consists of a magnetic circle with magnetic and anti-magnetic forces together with the air-gap, coil and disk reluctances in series, as illustrated in Fig. 3.2.

The relation between the magnetic field intensity and the applied current is given by the Ampere-Maxwell equation, as follows.

$$I_{net} = \oint_l H\, dl \tag{3.1}$$

$I_{net}$ is the current around the magnetic circle and is calculated using $I_{net} = N_{coil} I$ with $N$ being the number of turns of the coil and $I$ the electrical current applied in the coil. $H$ is the magnetic field intensity and $l$ is the equivalent length of the magnetic circle. When faced with only one electromagnet composed of a single core and coil the integral of equation 3.1 can calculated as $\oint_l H\, dl = Hl$ since the path of integration is a closed circle. $\Phi$, the magnetic flux, can be calculated using the equation below.

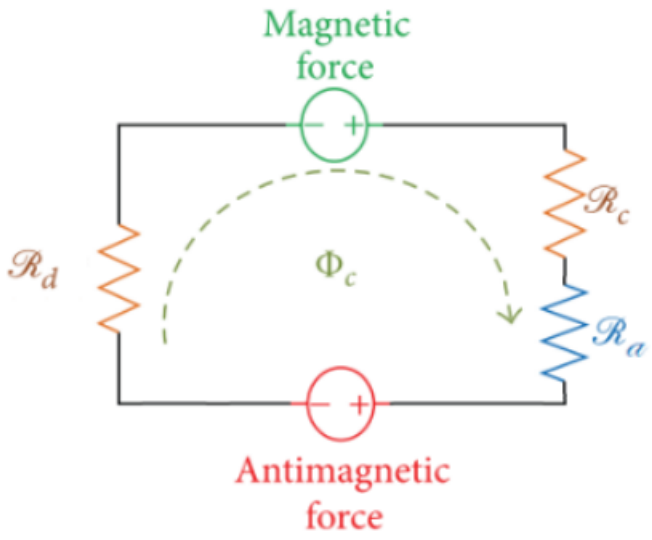



Figure 3.2: Equivalent magnetic circle of the ECB used in the dynamometer [19].

$$\Phi = BA = \frac{\mu N_{coil} I A}{l} \tag{3.2}$$

In this equation, $B$, the magnetic flux density, can be calculated by $B = \mu H$. In this equation, $\mu$ is the magnetic permeability and $A$ is the area of the pole shoe. The magnetic flux can also be defined as the result of the division of the combined magnetic force which is the sum of the magnetic and anti-magnetic forces, $\mathfrak{F}$, by the total reluctance, $\mathfrak{R}$, which will be calculated as the overall resistance in the electric circles.

$$\Phi = \frac{\mathfrak{F}}{\mathfrak{R}} \tag{3.3}$$

Using the equivalent magnetic circle illustrated in Fig. 3.2, the magnetic flux of the coil-core system is,

$$\Phi_c = \frac{\mathfrak{F}_i - \mathfrak{F}_{Eddy}}{\mathfrak{R}_c + \mathfrak{R}_d + \mathfrak{R}_a} \tag{3.4}$$

In this equation $\mathfrak{F}_i$ is the magnetic force generated by the current applied to the coil, $\mathfrak{F}_{Eddy}$ is the anti-magnetic force generated by the Eddy currents on the disk and $\mathfrak{R}_c$, $\mathfrak{R}_d$ and $\mathfrak{R}_a$ are the reluctances of the coil and core, the disk and the air, respectively. These reluctances can be calculated as follows.

$$\mathfrak{R}_c = \frac{l_c}{\mu_c \mu_0 A} \tag{3.5}$$

$$\mathfrak{R}_d = \frac{d}{\mu_r \mu_0 A} \tag{3.6}$$

$$\mathfrak{R}_a = \frac{l_g - d}{\mu_0 A} \tag{3.7}$$

In these equations $l_c$ is the length of the equivalent magnetic circle in the core, $\mu_c$ is the relative permeability of the core, $\mu_0$ is the permeability of the vacuum, $d$ is the thickness of the disk, $\mu_r$ is the relative permeability of the disk and $l_g$ is the gap between the electromagnet pole shoes. Because $\mu_c$ is much higher than $\mu_0$, with iron having a relative permeability of around 5000, the reluctance of the core will be minimal and can be disregarded. Because of that and knowing that the gap reluctance, $\mathfrak{R}_g$, is the sum of the air-gap and disk reluctances, equation 3.4 can be rewritten as follows.

$$\Phi_c = \frac{\mathfrak{F}_i - \mathfrak{F}_{Eddy}}{\mathfrak{R}_g} \tag{3.8}$$

The magnetic force in this equation, $\mathfrak{F}_i$ can be calculated as $\mathfrak{F}_i = N_{coil} I$ and the defined antimagnetic force, $\mathfrak{F}_{Eddy}$, is generated by the Eddy currents on the disk and can be calculated as

$$\mathfrak{F}_{Eddy} = \oint_s J_e dS = J_e dr. \tag{3.9}$$

Where $r$ is the equivalent radius of the pole shoe of the core and $J_e$ is the density of the Eddy currents at the disk:

$$J_e = \sigma \cdot a \cdot (\omega \times B_c). \tag{3.10}$$

Where $a$ is the distance from the disk centre to the pole shoe centre, $\omega$ is the angular speed, $\sigma$ is the electric conductivity of the disk and $B_c$ is the magnetic flux density. The electric conductivity is given by:

$$\sigma = \frac{1}{\rho_0(1+\alpha T)}. \tag{3.11}$$

In the above equation $\rho_0$ is the electric resistivity at 20° C, $\alpha$ is the temperature coefficient of resistance and $T$ is the temperature of the disk. The magnetic flux density is given by:

$$B_c = \frac{\Phi_c}{A}. \tag{3.12}$$

Given that $A = \pi r^2$ and considering equations 3.9, 3.10 and 3.12, $\mathfrak{F}_{Eddy}$ can be calculated by

$$\mathfrak{F}_{Eddy} = \sigma.a.d.r\left(\omega \times \frac{\Phi_c}{\pi r^2}\right). \tag{3.13}$$

From 3.8, $\mathfrak{F}_{Eddy}$ can also be calculated using the following equation.

$$\mathfrak{F}_{Eddy} = N_{coil}I - \Phi_c.\mathfrak{R}_g \tag{3.14}$$

Using 3.13 and 3.14 it is possible to calculate the magnetic flux as,

$$\Phi_c = \frac{N_{coil}I}{\Re_g + \sigma.a.d.r.\omega/A}. \tag{3.15}$$

Since one of the assumptions was that when the core became saturated the magnetic flux does not increase, then when reached the saturation point, equation 3.10 can be rewritten as,

$$J_e = \sigma \cdot a \cdot (\omega \times B_{sat}). \tag{3.16}$$

The dissipated power by the Eddy currents crossing the disk can be obtained by integrating $\rho J_e^2$ over the cylindrical volume $V = Ad$. In the end, the dissipated power is given by the following equation.

$$P_d = \rho . J_e^2 . V = \rho . J_e^2 . A . d \tag{3.17}$$

The torque can then be calculated using,

$$\tau_b = \frac{P_d}{\omega} \tag{3.18}$$

To obtain the critical speeds of the brake, the following equation can be used.

$$\omega_c = \frac{l_g - d}{\mu_0 . \sigma . a . d . r} + \frac{d}{\mu_r . \mu_0 . \sigma . a . d . r} \tag{3.19}$$

If the disk used is nonferromagnetic, equation 3.19 can be simplified to the following form.

$$\omega_c = \frac{l_g}{\mu_0 \omega a d r} \tag{3.20}$$

#### 3.1.2.3 Algortihm development

The above mathematical model was implemented in MATLAB. For that it was used the flowchart illustrated in Fig. 3.3. The code, which can consulted in annex A is divided in two sections, one where different materials can be compared and the second in which a single material can be seen in more detail.

The whole process was iterative and used to determined the final parameters of the brake, which are presented on Table 3.2. The value for the maximum angular speed takes into account that some electric motors are capable of operating at really high speeds, around 15000 rpm. It is also important to note that the angular speed on the model represents the rotational speed of the brake shaft and not the rotational speed of the engine.

The temperature of the disk was assumed to be of 200 °C because a forced-air mechanism is intended to be used for cooling. The properties of the materials studied can be consulted on Table 3.3 [36–39].

#### 3.1.2.4 Results

As it is presented in A, the critical speed and breaking torque at critical speed were calculated and are presented in Table 3.4.

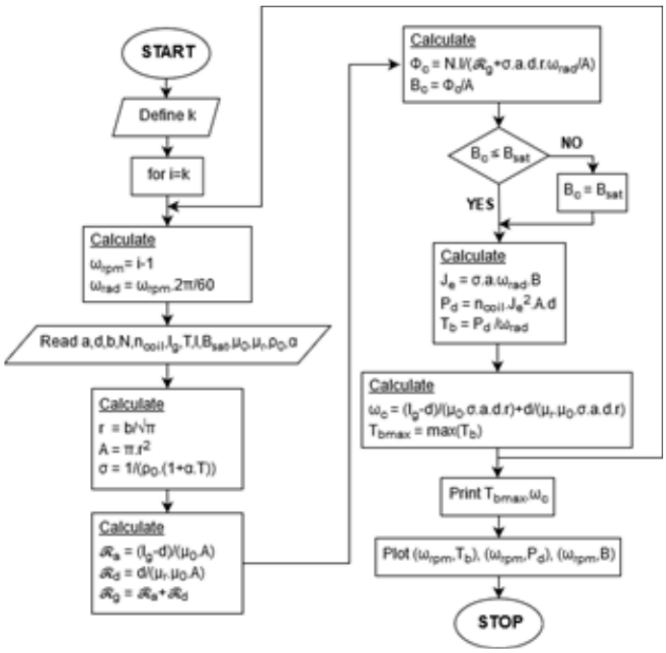



Figure 3.3: Flowchart used to implement the MATLAB code [19].

Table 3.2: Parameters of the ECB.

| Parameter | Description | Value | Unit |
|---|---|---|---|
| a | Distance from the centre of the disk to the centre of the pole shoe | 0.08 | m |
| d | Thickness of the disk | 0.004 | m |
| b | Width of the pole shoe | 0.040 | m |
| $_{max}$ | Maximum rpm | 4000 | rpm |
| $N_{coil}$ | Number of turns on the coil | 315 | - |
| $n_{coil}$ | Number of coils | 1 | - |
| $l_g$ | Distance between pole shoes | 0.007 | m |
| $\mu_0$ | Permeability of the vacuum | $4 \times \pi \times 10^{-7}$ | $Hm^{-1}$ |
| T | Temperature of the disk | 200 | ˚C |
| I | Current | 30 | A |
| $B_{sat}$ | Limit of magnetic saturation of the core | 1.5 | T |

Table 3.3: Rotational disk materials' properties [19].

| Parameter | Copper | Aluminuim | Austenitic Stainless Steel | Iron | Mild Steel | Electrical Steel | Unit |
|---|---|---|---|---|---|---|---|
| $\rho_0$ | $1.69 \times 10^{-8}$ | $2.65 \times 10^{-8}$ | $73 \times 10^{-8}$ | $9.70 \times 10^{-8}$ | $14 \times 10^{-8}$ | $59 \times 10^{-8}$ | $\Omega m^{-1}$ |
| $\alpha$ | $4.30 \times 10^{-3}$ | $4.5 \times 10^{-3}$ | $0.94 \times 10^{-3}$ | $5 \times 10^{-3}$ | $3 \times 10^{-3}$ | $2 \times 10^{-3}$ | $˚C^{-1}$ |
| $\mu_r$ | 1 | 1 | 1.02 | 5000 | 2000 | 7000 | - |

As can be seen in table 3.4 and taking into account the requirements presented in 3.1.2.1, it is not possible to use the Austenitic Stainless Steel, given that its critical speed is greater than 5000 rpm. It is also possible to verify that Copper can not assure a 2 kW of dissipated power at 2000 rpm as was required. In Fig. 3.4 the behaviour of the 3 different materials

Table 3.4: Critical speed, breaking torque at critical speed, breaking torque at 2000 rpm and power dissipated at 2000 rpm of the studied materials.

| Material | $\omega_c$ [rpm] | $\tau_c$ [Nm] | $\tau_{2000\,\text{rpm}}$ [Nm] | $P_{2000\,\text{rpm}}$ [kW] |
|---|---|---|---|---|
| Copper | 215 | 24.437 | 8.58 | 1.8 |
| Aluminium | 345 | 24.437 | 12.27 | 2.57 |
| Austenitic Stainless Steel | 5876 | 24.714 | 18.73 | 3.92 |

when a 30 A current is applied can be observed.

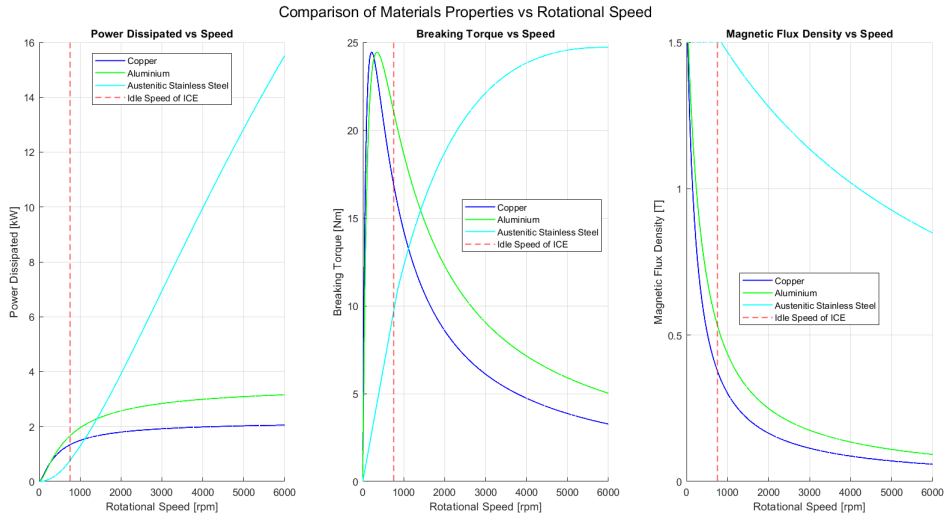



Figure 3.4: Representation of the Power Dissipated, Breaking Torque and Magnetic Flux Density vs Speed of the different materials.

With everything that was presented in mind, Aluminium was the chosen material to build the disk of the ECB. To better visualize the behaviour of this material at different currents and speeds it was decided to create 3D maps of the dissipated power and breaking torque that illustrate its behaviour. This maps are presented in Fig. 3.5 and Fig. 3.6, respectively.

## 3.2 Components of the ETB Setup

### 3.2.1 Torque measuring device

As mentioned earlier in chapter 1 there are some ETBs developed in the C-MAST Propulsion Laboratory such as the ones presented in [16,19]. These works will impact the choices made in this project since some of the empirical knowledge acquired during the assembly of those ETBs will be used. Another thing to keep in mind is the fact that since one of the objectives for this project was to reduce costs, it was expected that most of the materials used on the ETB were materials available in the lab. Because of that, some of the choices

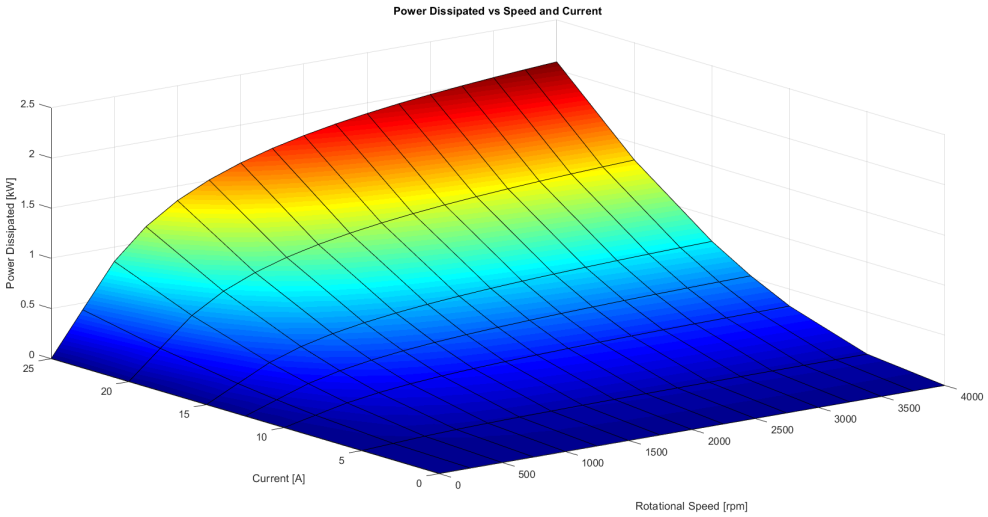



Figure 3.5: Representation of the dissipated power by the brake at different currents and speeds.

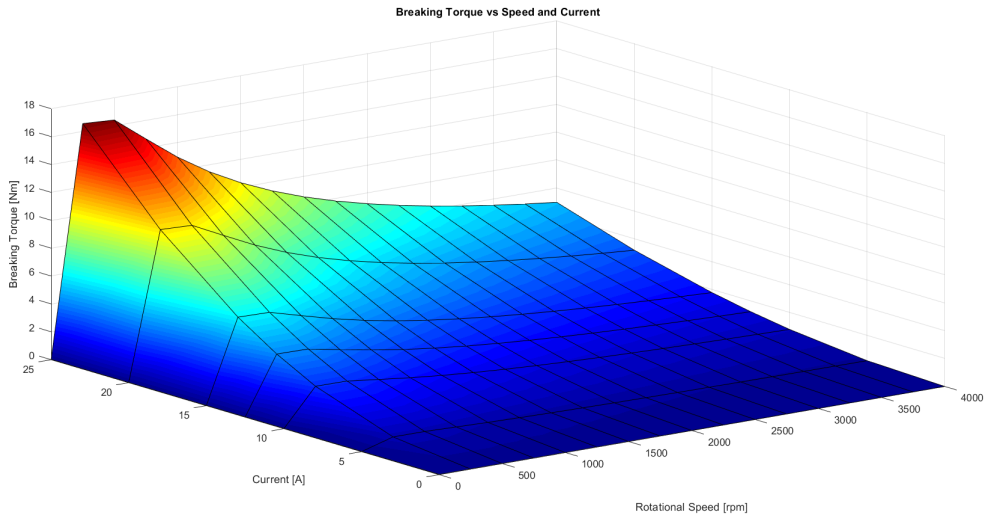



Figure 3.6: Representation of the breaking torque of the brake at different currents and speeds.

explained in this dissertation might not be the most efficient or the most straightforward.

With the reasons given above in mind and since there were some load cells of the DYLY-103 model in the lab, this was the transducer used. This type of load cell avoids the problems that would arise when using a spring balance such as the use of weights. It also avoids the calibration problems and the high-precision assembling process needed to use an in-shaft transducer. This load cell, illustrated in Fig. 3.7 is an S-shaped load cell which can measure while being compressed or tractioned. This is an advantage in comparison with other types of load cells because it can measure the torque even if the dynamometer is not working in the first quadrant. The load cell selected has a range of 20 kg. This value

was decided after the full weight of the ECB that the load cell would have to support was known and found to be over 10 kg. This meant that the cell with the minimum range that could support the weight was a 20 kg one.


Figure 3.7: Example of a S-shaped DYLY-103 load cell [40].

There are however some disadvantages with the use of this load cell. First of all, it requires the use of a HX711 amplifier as illustrated in figure 3.8. Without it, the acquisition system would have some problems reading the signals of the load cell. This amplifier can then be integrated on the code of the ETB using the `HX711.h` library. Secondly, it requires some level of inspection to avoid failure from fatigue as stated previously in 2.1.1.

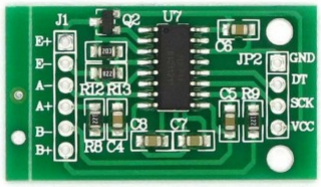



Figure 3.8: HX711 amplifier used on the load cells [41].

### 3.2.2 Mounting system of the ECB

With the use of a load cell, it is impossible to build a fixed mounting for the dynamometer. Because of that and also due to the empirical knowledge from the previous works and cost it was decided to build a trunnion bearing mounting.

This mounting avoids the problems of the rotational centre moving when the dynamome-

ter is loaded which would happen using flexure supports. It is, however, necessary to periodically inspect the bearings to avoid any kind of brinelling or fretting that can happen and with that decalibrate the dynamometer. The lubricant or grease depending on the type of bearing must be also inspected to avoid any kind of friction and wear.

#### 3.2.2.1 Coil and Core

With all the key parameters of the ECB obtained, the design of the dynamometer was started. The first step of this process was to design the core and coil set. For the core, it was decided to use small Iron plates with 3.1 mm of thickness. The decision to choose small thickness plates instead of a solid core has to do with the fact that the Eddy current losses of the core are reduced when the core is laminated and the laminations are isolated from each other [42]. The big plate used for the core was already in stock in the laboratory. To cut the plate into the intended format it was used a water cutting machine from FabLab. After cutting, each small part was painted to ensure that each lamination was isolated from the other next to it. Then, they were mounted to form two parts that would lock each other when in place. To ensure that all plates were fixed 4 screws were used in each half. The drawing of the plates and core can be consulted on B.1.

For the winding process of the coil some suggestions from *Nunes* were accepted. On his ETB, the coils were winded directly onto the iron core. Due to the C-shape of the core, this process was slow and difficult and it was deemed necessary to make it faster and easier. To avoid that it was decided to 3D print a support for the coil in PLA to facilitate the winding process and then to place it on the core. Firstly, it was necessary to know the size that the coil would take in order to draw the support and for that it was necessary to select the wire thickness. In this ECB since the current will be high it was decided to use a 3 mm thick copper wire. With this thickness it is possible to estimate the size of the coil taking into account the values from table 3.2 and the dimensions of the core.

Table 3.5: Accumulated number of turns for each layer and radius of the coil.

| Number of layers | Number of turns in the coil | Radius of the coil |
|---|---|---|
| 1 | 46.6667 | 46 |
| 2 | 93.3333 | 49 |
| 3 | 140 | 52 |
| 4 | 186.6667 | 55 |
| 5 | 233.3333 | 58 |
| 6 | 280 | 61 |
| 7 | 326.6667 | 64 |

With table 3.5 done it was found that the coil would have a diameter of 64 mm and thus the support would have to be able to accommodate that size. The drawing of the coil support can be consulted on B.2.

The winding process of the coil was outsourced since there wasn't equipment available

on the laboratory capable of doing this work. However during this process the number of turns were lost resulting in a coil that may or may have more or less turns than the intended number. Because of that this issue will have to be studied later on to check if this inaccuracy has some effect in the work of the ETB.

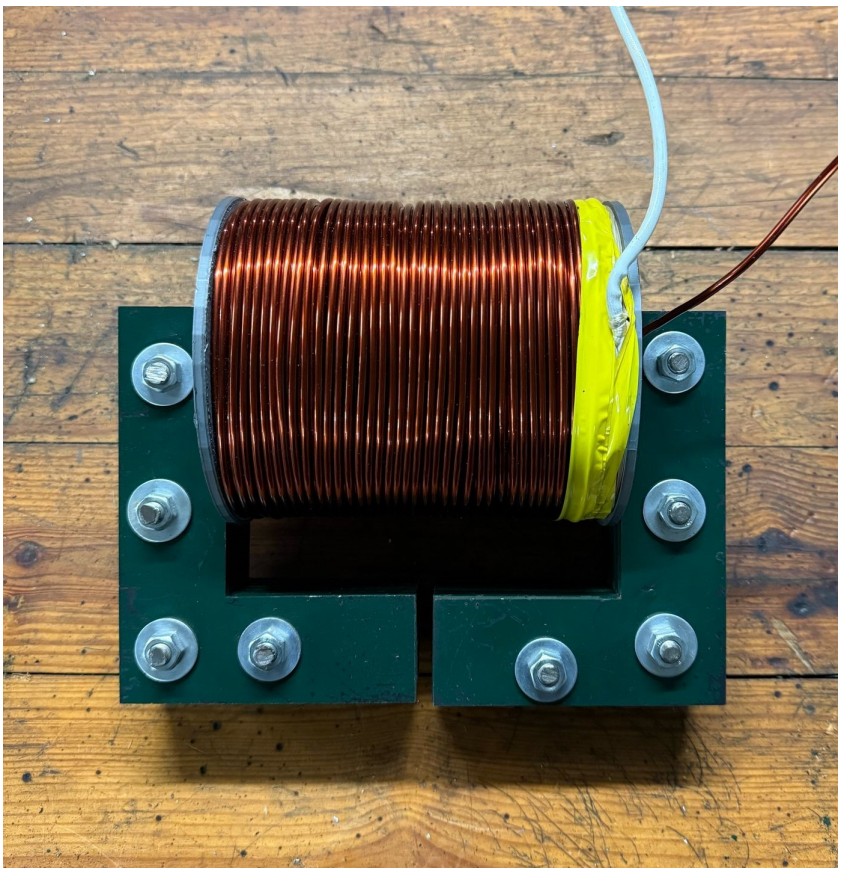


Figure 3.9: Core and coil set assembled.

The result of this whole process is illustrated in Fig. 3.9.

#### 3.2.2.2 Disk and shaft

As studied in 3.1.2, it was decided to use aluminium for the rotating disk. After deciding the materials, a disk was designed in a Computer Aided Design (CAD) software in order to comply with the parameters of table 3.2. The technical drawing of the disk can be found in B.3.

The shaft used on the build was a 20 mm diameter non-ferromagnetic aluminium shaft. This choice was made to avoid any disturbances in the magnetic field that the shaft could create if it was magnetic.

To connect the disk to the shaft, an axis rack support was used. It was needed to make some holes on the disk and use screws to fix the disk onto the rack.

#### 3.2.2.3 Pulley and belt

On one end of the shaft there is a toothed pulley with 60 teeth, connected to another pulley with 15 teeth trough a toothed belt with 2 cm of width. This system creates a demultiplication in terms of rotational speed in 4 times which in turns increases the torque in four times. This system was needed due to the fact that small engines might not have a high enough torque for the brake.

However, this system can not be forgotten when acquiring and analysing the speed and torque data since the value that will be acquired by both sensors will not correspond to the true speed and torque of the engine.

#### 3.2.2.4 Central Structure

To secure the coil and core set, connect it to the load cell and provide support to the shaft attached to the rotating disk, a central structure was designed and assembled. To do that it was firstly necessary to understand the requirements for the structure, which are presented below.

1. It has to have an adequate size in order to ensure that the distance between the centre of the pole shoe and the shaft is as stated in 3.2.
2. During testing the structure must not deform in order to ensure that the reaction force of the load cell is always perpendicular to the arm between the load cell and the shaft.
3. The material of the structure should not be magnetic to ensure that the magnetic field created in the coil isn't dissipated to the structure.

#### 3.2.2.5 Design of the central structure

The design of the structure was inspired on the ETB built by *Nunes*. In this case, however, the load cell will be connected below the structure instead of above as it was done in that work. This decision was made due to the fact that this ETB will be assembled on a table, which was not the case for that dynamometer.

To ensure that the requirements presented above were met, the structure, which can be viewed in figure 3.10, consists of 2 longitudinal plates connected by 2 rectangular plates and reinforced by a series of L-beams. The transversal plates give rigidity to the assembly in order to avoid any torsion that might occur during testing.

During the design process it was decided to build the structure in a configuration which allowed for another set of coil and core to be installed, if needed. Because of that, the aluminium structure is much bigger than the required for just a single coil. The chosen

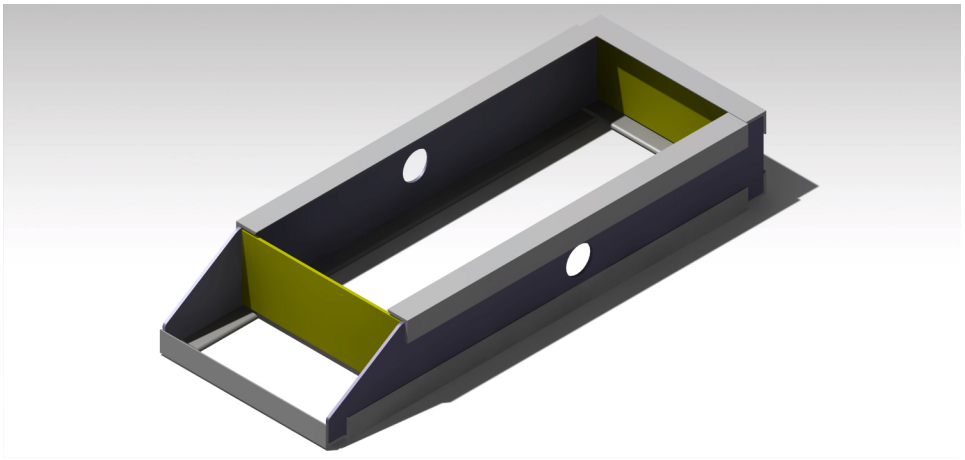


Figure 3.10: CAD model of the central structure of the ETB.

thickness for the plates and beam was of 4 mm which ensures that the second requirement is also fulfilled.

#### 3.2.2.6 Assembly of the Central Structure and the ECB

On the longitudinal plates a hole with a radius of 25 mm was made for the passage of the shaft. The load cell will be positioned with an arm of 385 mm which has to be taken into account when calculating the torque. The technical drawing of the structure can be consulted on B.4.

To connect the shaft of the disk to the structure, 2 different bearings were attached to the longitudinal plates. As seen in the technical drawing of the structure, another hole was made to connect the structure to the load cell that will be used to measure torque. As mentioned earlier it was of the upmost importance that the the structure of the ECB was parallel to the table in order to guarantee that the load cell would be perpendicular to it. As can be seen on figure 3.11, this was achieved, which ensures that the force generated by the torque of the engine will only have a vertical component, which will be detected by the load cell.

The final assembled structure is illustrated in figure 3.12 with the key components of the ECB numbered.

## 3.3 ETB Electronic System

During the operation of every ETB it is necessary to control it and also to acquire the data required. Although these systems could have been divided into two separate platforms, it was decided to join both systems into the same platform, the ETB Control and Data System (ETBCDS). This provides a more compact solution as well as more versatility


Figure 3.11: Level used to ensure that the structure was perpendicular to the load cell.

since if, for example, the need for transient testing ever comes up it can be more easily integrated.

Firstly, however, it is needed to analyse the requirements for each sub-system. For that, table 3.6 was developed.

Table 3.6: Requirements for the control and data acquisition sub-systems.

| Control system | Data acquisition system |
|---|---|
| Ability to provide power to the coil. | Acquire force readings from the load cell. |
| Ability to control the current on the coil. | Acquire speed readings from the hall sensor. |
| Ability to do semi-automatic testing to an engine. | Determine the power consumed by the engine/motor. |

Even though they may seem a bit generic, the requirements above, when met, guarantee that the ETB will operate as intended. To join the two sub-systems in the same platform, it was decided to use an Arduino Mega 2560 R3.

In the next sections the logic, functions, the equipment, and finally the calibration of the

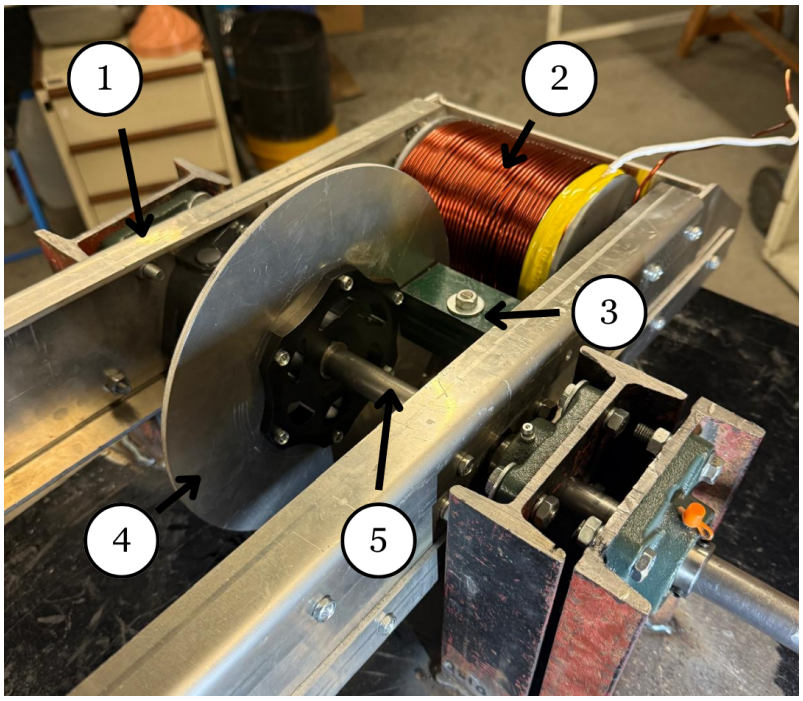



Figure 3.12: ECB setup used with the key components numbered: **1**-Aluminium central structure; **2**-ECB coil; **3**-Iron core; **4**-Rotating disk; **5**-Shaft.

ETBCDS will be presented.

### 3.3.1 System logic

While some of the requirements presented in table 3.6 might be straight-foward, such as the readings of speed and torque, others require some more complex systems and logic such as the automatic testing of the engine.

This requirement was presented since in future iterations the ETB one of the objectives might the test of programmed missions. For that, it is fundamental that the ETB might be controlled by the Arduino, if needed. In this version, however, it was decided to program the ETB for static and manual testing only. Both of these test modes basically work using the position/position mode presented in 2.2.1.1. Although there are safer modes, this was the selected one due to its simplicity of implementation.

The selection between these 2 modes of operation is made in a ”main menu” where the user might select each of the modes. It was also decided to give the user the chance of *taring* the load cells while on the main menu. The flowchart used to implement this logic is illustrated in figure 3.13.

In the case of the manual mode, it was decided that depending on the inputs from the user, the load on the brake should increase or decrease. At the same time the ETBCDS must be able to receive all the data necessary for testing.

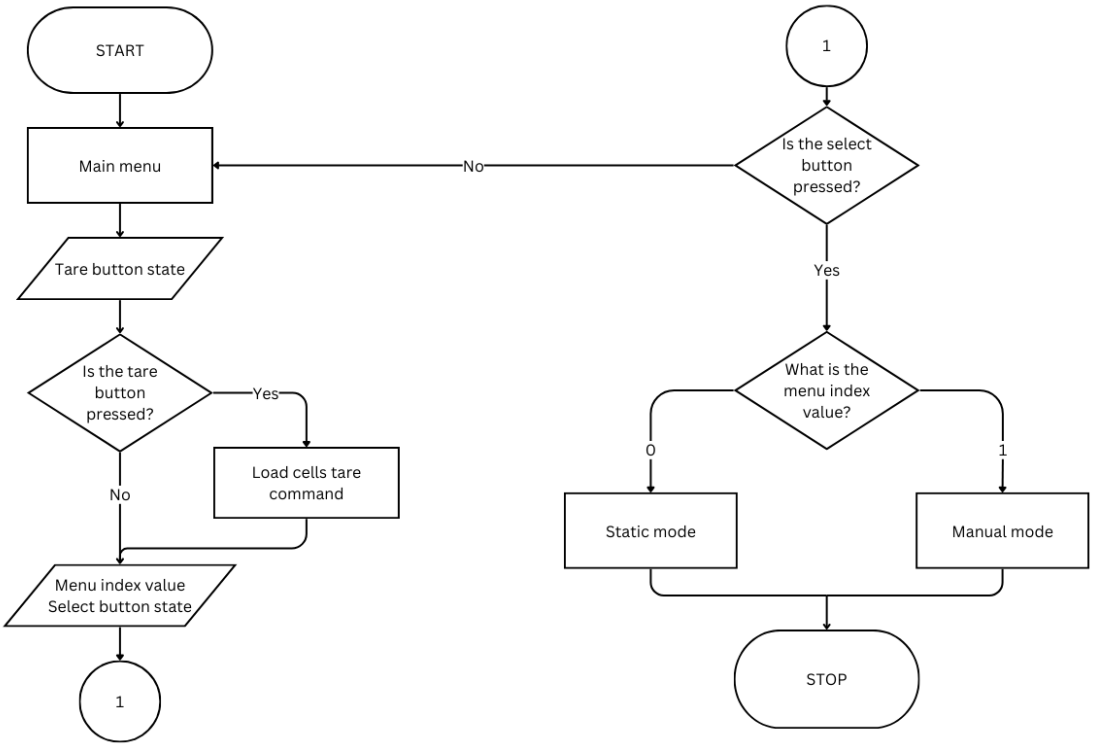



Figure 3.13: Flowchart of the main menu logic implemented.

1. Time;
2. Load current;
3. Force on the load cell;
4. Speed of the engine;
5. EM current;
6. ICE fuel mass.

Which can then be used to calculate the $P_{shaft}$, $P_{consumed}$ and $\eta$ using equations 2.1, 2.4, 2.3 and 2.2. The flow chart on figure 3.14 represents the logic behind the manual mode.

In the case of the static mode it was decided to implement an algorithm that would allow for the tests to be automatic. In this case it was decided to test the engine during 60 seconds and after that increase the load on the shaft. To avoid any damage to the engine, the operator has to input the minimum speed allowed before the test has to be stopped. This test is done with the engine in the same throttle condition and can then be repeated for other conditions. The data collected in this mode is the same as in manual mode. Figure 3.15 represents the logic behind this mode of testing.

While the manual mode of testing could be done with a simple potentiometer that controls the coil Power-supply (PS), as it is evident by figure 3.15, the static mode requires the use

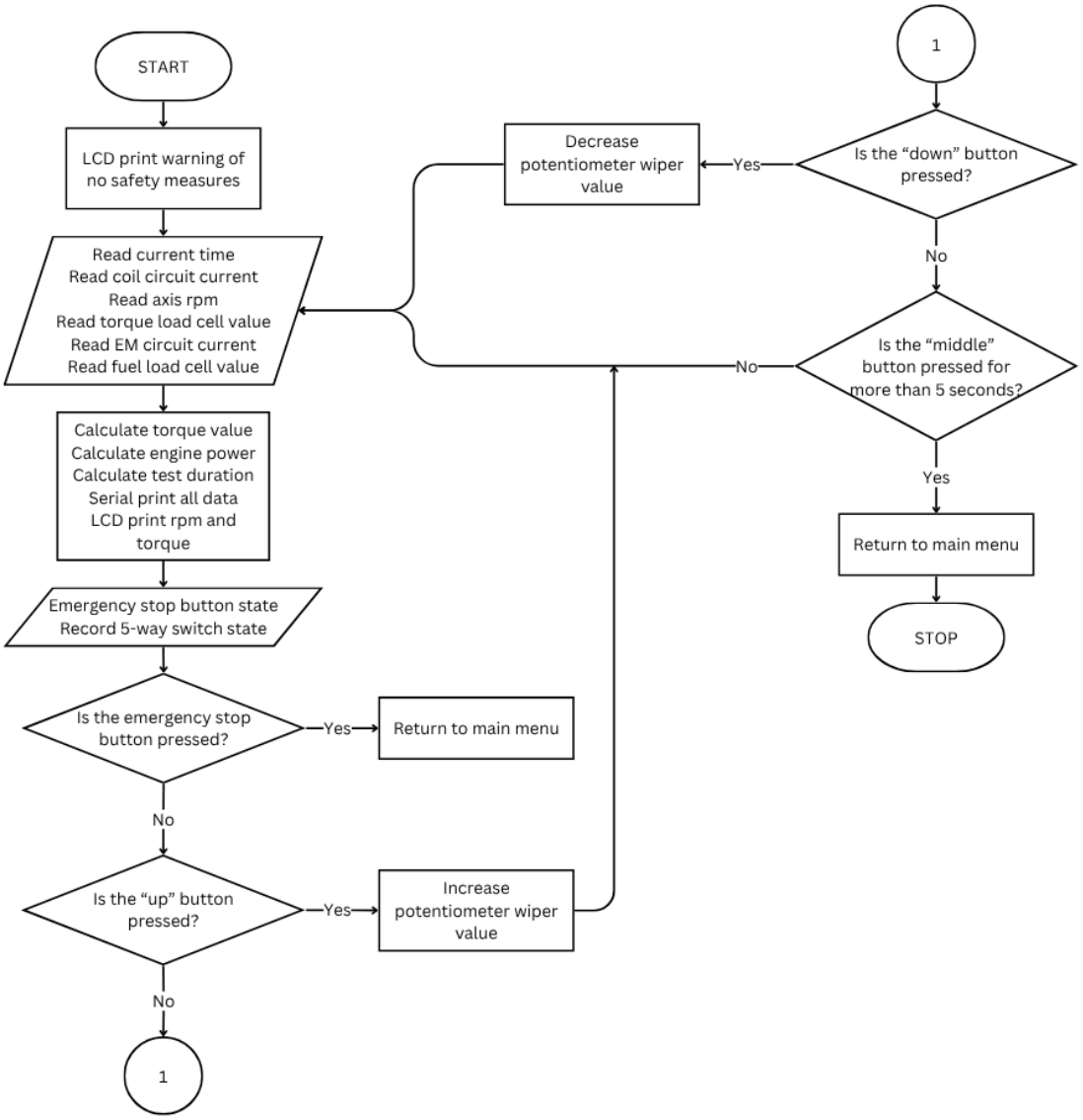



Figure 3.14: Logic implemented for manual mode of testing.

of a Digital Potentiometer (DP) that can be controlled by the Arduino. It is also important to note that, even though the action is not presented in the flow charts, the load cells may be tared at any time during testing.

After the logic was developed, the Arduino code was written which can be consulted on C.1. In order to read the data that is printed out through the serial port the *PuTTY* software was used. This allows to record all data that is printed on the serial monitor in a file. On this case, all files were recorded with the `.txt` format. This allows the use of different software for post-processing the data such as *Excel* or *MATLAB*.

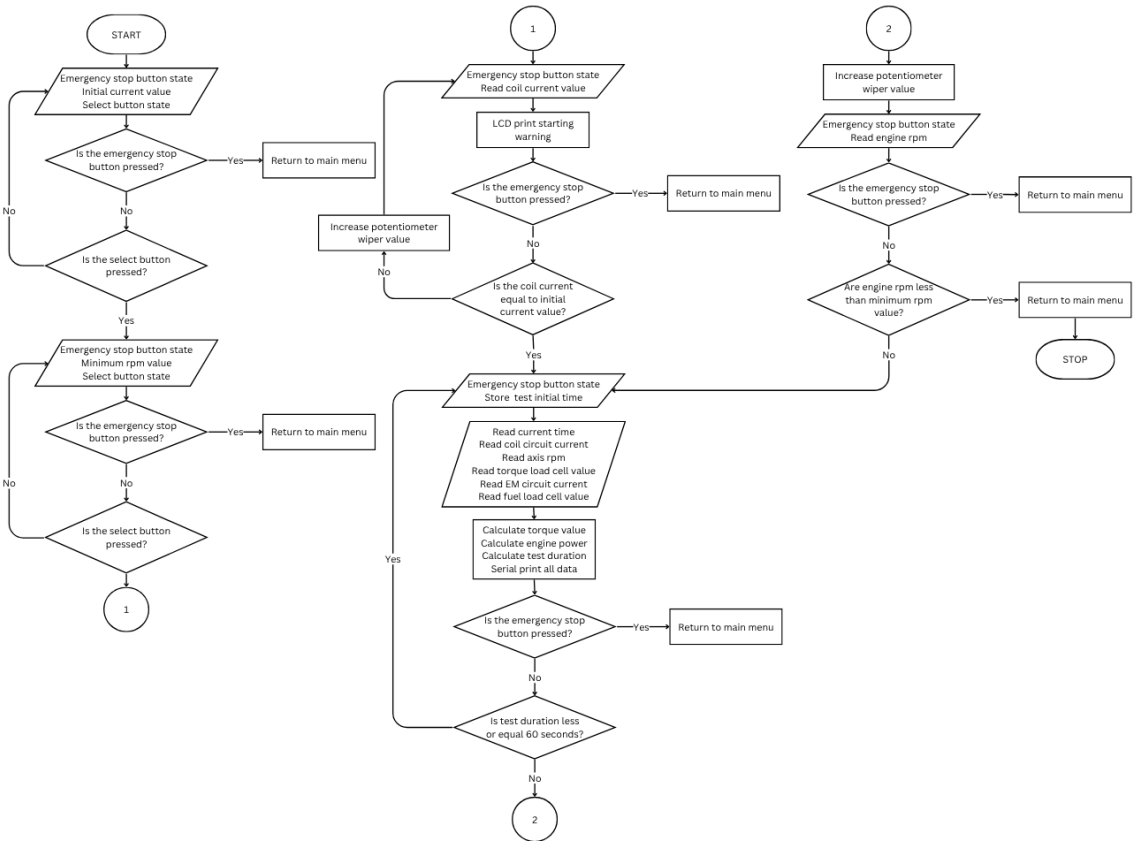



Figure 3.15: Logic implemented for static mode of testing.

### 3.3.2 Electronic equipment

#### 3.3.2.1 Coil PS

When choosing the PS for the coil it was necessary to select one which could guarantee the parameters obtained previously (presented in table 3.2). With that in mind, the PS chosen to power the coil of the ECB is the MS-1500-12. Table 3.7 presents its characteristics found on the product datasheet.

Table 3.7: MS-1500-12 PS information [43].

| | |
|---|---|
| Rated Power | 1500 W |
| Input Voltage | 100 - 240 V ∼ |
| Output Voltage | 0 - 12 V DC |
| Output Current | 0 - 125 A |
| Dimensions | 234 × 124 × 64 mm |

As it can be found in table 3.7, the output current rating of the PS is a lot higher (416%) than the condition for the operation of the ECB. Even though this is not ideal, since the accuracy of the PS decreases, this model was the only one available that fulfilled the requirements since it was the only model found which assured that the current supplied could be changed from 0 to 30 A.

#### 3.3.2.2 Arduino Mega 2560 R3

To control all the electronic equipments and also read the data from the sensors, an Arduino Mega 2560 R3 microcontroller was selected. This board allows Serial Peripheral Interface (SPI) and Inter-Integrated Circuit (I2C) connections that will be needed for this project. Figure 3.16 illustrates the pinout diagram of the Arduino Mega 2560 R3 board.

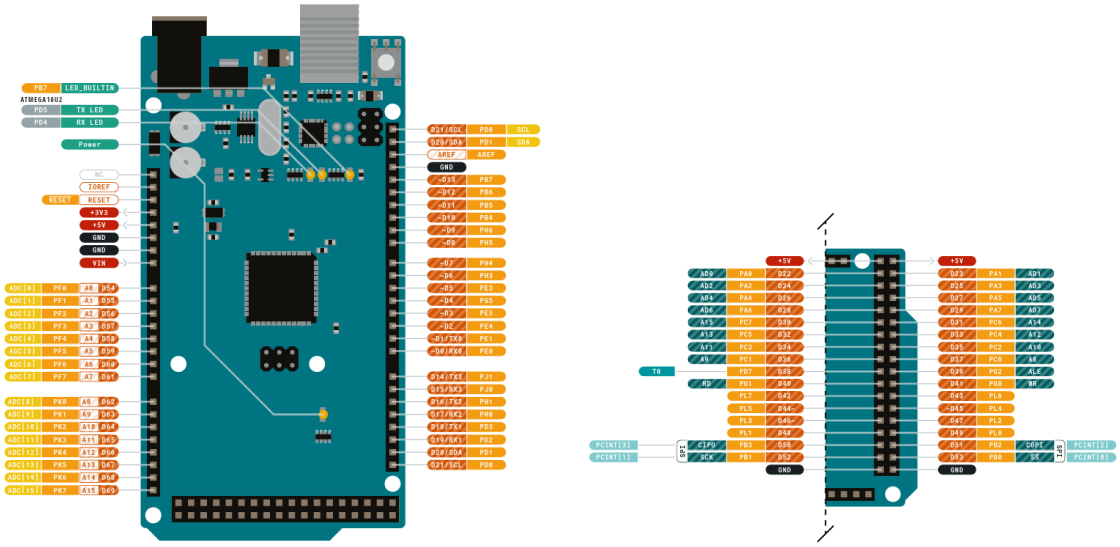


Figure 3.16: Arduino Mega 2560 R3 pinout [44].

Furthermore, when compared to other boards such as an Arduino Nano or Arduino Uno, this board allows for more sensors and actuators to be connected due to having more pins.The choice of using an Arduino platform and compatible sensors was taken due to three main points.

1. The relative low cost of the microcontroller, the sensors, actuators and other hardware;

2. The high level of customization that can be achieved when using this platform;

3. Highly accessible documentation which allows a more straight-forward design and assembly process. This also facilitates any kind of problem solving if that need ever arises.

To avoid that the wires disconnect from the board due to vibrations it was decided to use an available shield that connects the wires to the board with sockets and plugs securing these connections.

#### 3.3.2.3 Digital Potentiometer

As mentioned in 3.3.1, it was necessary to use a DP to control the ECB. To decide on which model to use it was firstly necessary to check the resistance value of the built-in potentiometers of the coil PS.

The coil PS has two different potentiometers, one that controls the output voltage and other that controls the output current. Both of them have the same resistance value of $5\,\mathrm{k}\Omega$. Using this value, the MCP4231 DP was selected to be employed on the ETB.

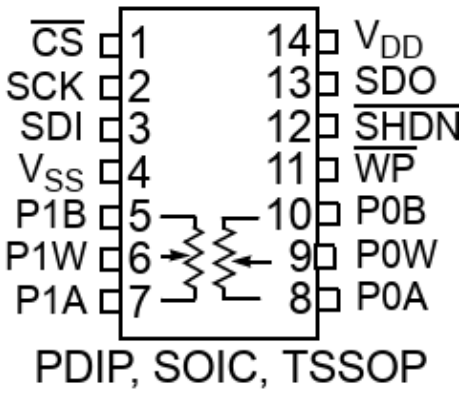



Figure 3.17: Pinout of the MCP42X1 DP family [45].

The MCP4231 is a 7-bit, which translates to 128 wiper steps, DP capable of using the SPI communication protocol. The specific model used, MCP4231, has already integrated 2 different DP on the same chip which means that only one chip has to be used to replace both built-in potentiometers [45]. Figure 3.17 illustrates the pinout of the MCP4231 DP.

The SPI is a very common communication protocol used to facilitate the communication between various devices such as micro-controllers, sensors, display devices or memory chips. This protocol is capable of transferring the data without any interruption, which doesn't happen on other protocols such as I2C [46]. In the case of the ETB the Arduino takes the role of the master while the MCP4231 takes the role of slave. In order to use this protocol the `<SPI.h>` library had to be used.

#### 3.3.2.4 Hall sensor

In order to measure the speed of the engine, a VMA313 HallSensor was employed. As explained previously in 2.1.2 this sensor is activated when a magnetic field passes near it. Because of that, a small magnet had to be attached to the shaft of the ETB. Figure 3.18 illustrates this assembly.

In order to use this sensor an `interrupt()` has to be used. This means that every time a certain condition is met, in this case the signal from the sensor, a certain function is called.

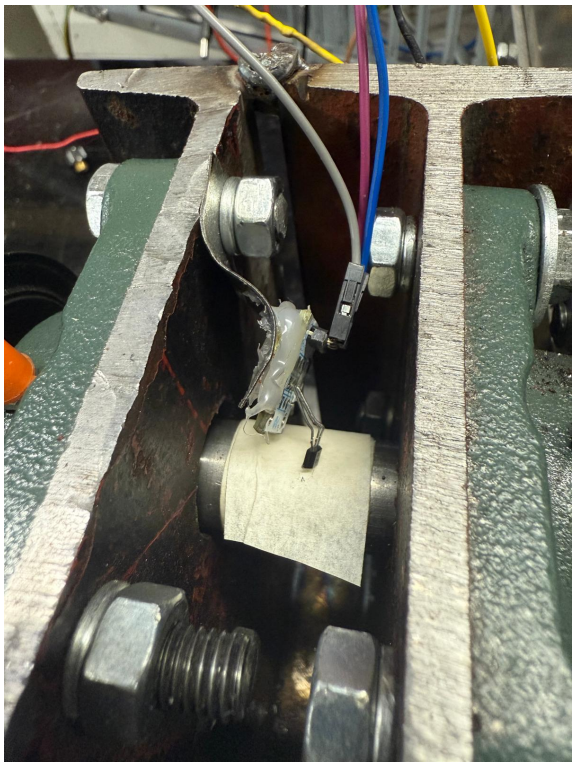


Figure 3.18: Hall Sensor and magnet assembled on the ETB.

In this case, this function, `countRpm()`, obtains the interval between each signal from the sensor which is then used to obtain the speed of the engine.

#### 3.3.2.5 Current shunt and module

This ETB needs to measure current in two different circuits, the first one is the coil circuit in order to know the value of the current going trough the coil. The second instance where current needs to be measured is in the EM circuit in order to apply equation 2.3 and calculate the power consumed by the motor.

There are two main types of current sensors available, the first, a linear Hall sensor circuit that is near a copper conductor. When current flows through the conductor, the Hall sensor circuit detects it and transforms it into a corresponding voltage. The second type consists of a shunt resistor. This type of resistor is a small, but very precise resistor that is put in series in the circuit and has two probe wires that are used to measure the voltage drop across the shunt. Using the known resistance of the shunt and the measured voltage drop it is possible to calculate the current in the circuit using Ohm's law [47].

$$I = \frac{U_{shunt}}{R_{shun}} \tag{3.21}$$

Figure 3.19: Current shunt used on the ETB.

Since Hall current sensors such as the ACS712 generate a lot of noise and the measurements done in ETB require a more precise measurement it was decided to opt for a shunt resistor. The shunts selected, that are illustrated in fig. 3.19 are rated for $50\,\mathrm{A}$ and have an output voltage of $75\,\mathrm{mV}$. Since the Arduino used has a 10-bit resolution for $5\,\mathrm{V}$ in the analogue pins and the shunt used has an output voltage of only $75\,\mathrm{mV}$ results in a resolution of about $3.255\,\mathrm{A\,bit^{-1}}$ which isn't enough for the accuracy required.

Because of this it was decided to use the shunt together with an altered INA219 module as illustrated in fig. 3.20. This module is equipped with a $100\,\mathrm{m\Omega}$ shunt which was, in this case, taken out and the probe wires from the $50\,\mathrm{A}$ shunt were soldered into the module. Since this module uses an Arduino library it was necessary to change the `setCalibration_32V_2A()` function to account for the different resistor. This altered function can be consulted in annex C.2.

#### 3.3.2.6 Fuel load cell and HX711 amplifiers

To measure the fuel consumption of the ICE it is possible to use a flow meter or a weight scale. The big advantage of the flow meter is that it requires less steps to measure the fuel flow since it can measure that value directly. However, it is more susceptible to vibration errors thus annulling the advantages of this type of sensors.

Because of this a DYLY-103 $5\,\mathrm{kg}$ load cell was used to measure the fuel tank. With the data from the load cell it is then possible to apply numerical methods to calculate the fuel flow of the ICE.

This load cell also requires the use of a HX711 amplifier due to the reasons presented in 3.2.1.

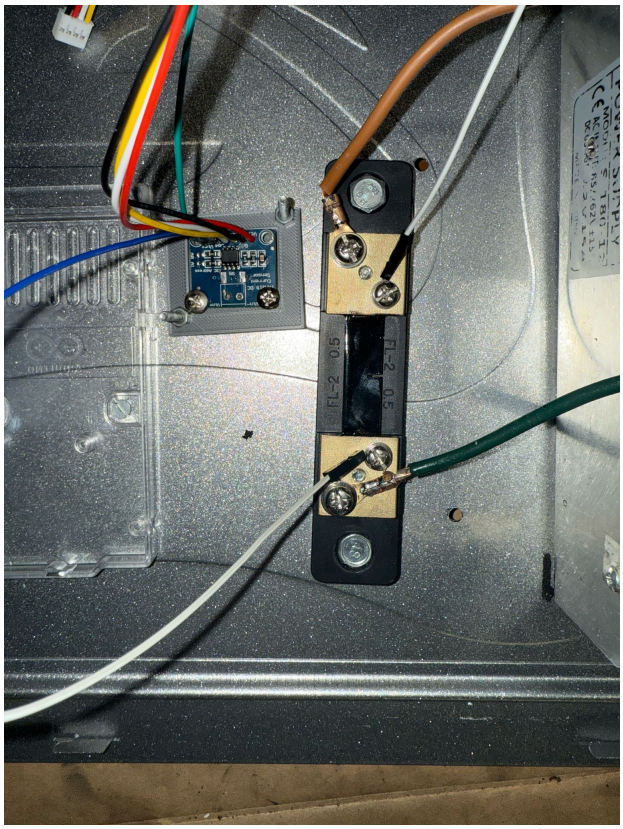


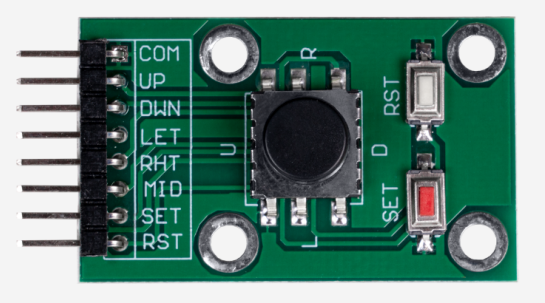
Figure 3.20: Current Shunt and INA219 module.

#### 3.3.2.7 Liquid Crystal Display (LCD) and naviagtion swith

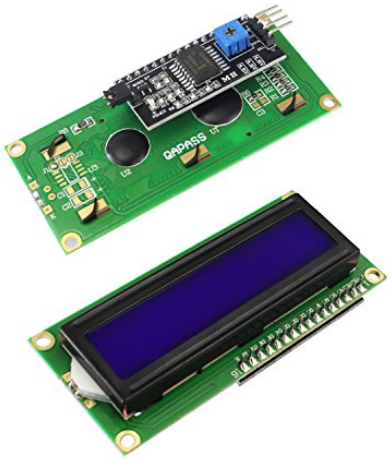
In order to have a semi-independent system it was decided to have some components that receive the inputs from the operator of the ETB and also outputs what is happening to the operator.

In order to receive the input from the user it was decided to use a 5-way navigation switch from JOY-IT. With this switch the operator is able to configure the ETB and manage its operation. The main reason for the selection of this component was the fact that it has more buttons than, for example, a potentiometer integrated with a push button. This results in less points of failure since more functions can be integrated into the same component.

On the other hand, in order to communicate with the operator it was decided to use a 16x2 LCD display integrated with an I2C module. This module facilitates the connection between the display and the Arduino and at the same time shortens the code necessary to work with the screen when using the `<LiquidCrystal_I2C.h>` and `<Wire.h>` libraries. Figure 3.21 illustrates these 2 components.




(a)



(b)

Figure 3.21: (a) 5-way navigation switch used [48]; (b) 16x2 LCD display with I2C module [49].

#### 3.3.2.8 Cooling fans

Since the majority of the equipment described before will generate some heat it is necessary to create some system capable of dissipating it. For the electronic equipment it was decided to use 2 different cooling fans sourced from an old computer that were available. These fans both operate at a voltage of $12\,\mathrm{V}$ and draw $0.14\,\mathrm{A}$ of current.

It is also important to cool off the disk and the coil since their operation depends on the temperature not increasing a lot. For that it was decided to use a bigger fan that was available in the laboratory that operates at $12\,\mathrm{V}$ and draws $6.67\,\mathrm{A}$ of current when operating at maximum power. Figure 3.22 shows the cooling fan used to force airflow through the disk and the coil.

#### 3.3.2.9 Secondary PS and protection equipment

In order to power the Arduino and the cooling fans, a secondary PS was needed. For this, it was necessary to have a DC PS capable of outputting $12\,\mathrm{V}$ since all the equipment works at that voltage and at least $7\,\mathrm{A}$ since that is the current consumed by the Arduino, the small cooling fans and the big cooling fan. Since a S-180-12 DC PS with the characteristics represented in table 3.8 was available in the laboratory it was decided to use it. Although this PS will output much more current than the necessary it was the only model accessible at the time capable of outputting more than $7\,\mathrm{A}$.

To protect the electronic equipment of the ETB a series of fuses were installed according to the current rating of each component. To have an easier access to the fuses a small plate was 3D printed. This plate also houses the switches for each system and can be seen in

Figure 3.22: Coil and disk cooling fan installed on the ETB table.

Table 3.8: Secondary PS characteristics.

| S-180-12 PS | |
|---|---|
| Input voltage | AC 100 ~240 V |
| Input frequency | 50 ~60 Hz |
| Output voltage | 12 V |
| Output power | 180 W |
| Output current | 15 A |
| Dimensions | 200x40x58 mm |

figure 3.23.

### 3.3.3 System set-up and calibration

While most of the devices used on the ETBCDS can be used without changes or calibration, such as the LCD, the joystick and the Hall sensor, some sensors require some changes and/or calibration in order to be incorporated into the code.

#### 3.3.3.1 INA219 address change

To be used with the I2C protocol together with the other sensors, one of the INA219 modules requires some changes in its address. Since the default address, as can be seen on figure 3.24 (a), is 0x40, if no changes are made in the board, the 2 devices used would have the same address. To avoid this one of the pins (A0 and A1) on the module must be soldered. Since only two modules will be used on this ETB pin A0 was soldered which changed the address of the module used for measuring EM current. The new address, 0x41, was then used in the code.


Figure 3.23: 3D printed plate that houses the fuses and switches of the electric system of the ETB.

#### 3.3.3.2 Load cell calibration and error testing

In order to integrate the HX711 module into the code, the calibration factor of each load cell had to be obtained. To find it, a known weight must be used and with the raw reading from the load cell, the calibration factor can be calculated using equation 3.22.

$$Calibration\ factor = \frac{Raw\ reading}{Real\ mass} \tag{3.22}$$

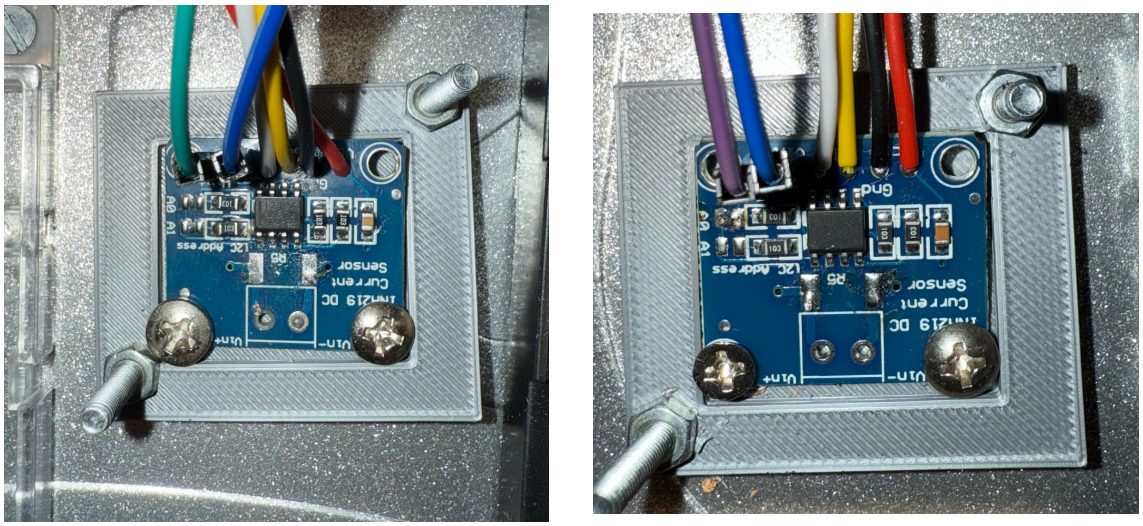
To reduce the time taken by this process a code was developed and used, which can be consulted in C.3. The obtained values calibration factor values are presented in table 3.9. The set up used of these processes is illustrated in figure 3.25.

Table 3.9: Calibration factors used for each load cell.

| Load cell | Calibration factor |
|---|---|
| ETB load cell - 20 kg | 213.49105 |
| Fuel load cell - 5 kg | 860.35888 |

With the calibration factors obtained, it was time to obtain the error values of the load cells. On this type of equipment, the main types of error are as follows:

1. Hysteresis error;
2. Non-repeatability error;



(a) (b)

Figure 3.24: (a) INA219 with 0x40 address; (b) INA219 with A0 pin soldered and 0x41 address.

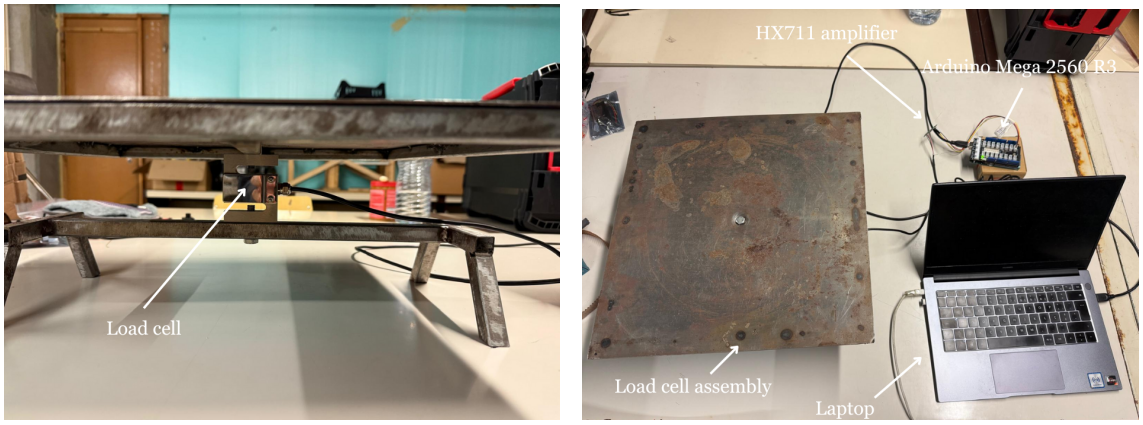



Figure 3.25: Set-up used for the calibration process of the load cells.

3. Creep error;

Using the same test set-up, each of these types was studied and the value for the full-scale error of each load cell was determined. The total error was obtained by adding all the error values of each type. Figure 3.26 illustrates the testing done to obtain the hysteresis and non-repeability errors of the 20 kg load cell.

### Hysteresis error

This phenomenon can be defined as the inaccuracy that a system has when measuring the same input, in this case force, in the loading or the unloading phases. To measure this, as can be seen in figure 3.26, different loads were incremented and after that decremented for 3 different trials. The load cell is always measuring the mass of the different weights which can then be compared and used to calculate the hysteresis effect of each load cell. Tables 3.10 and 3.11 present the mass readings during this process for both load cells.

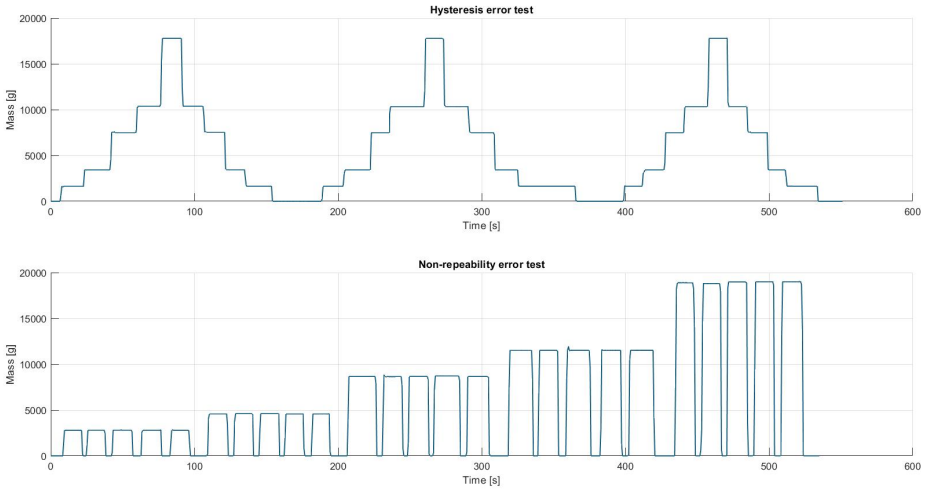



Figure 3.26: Hysteresis and non-repeability error testing.

Table 3.10: Hysteresis error test readings for the 20 kg load cell.

| | Average measured mass [g] | | |
|---|---|---|---|
| Real mass [g] | 1st try | 2nd try | 3rd try |
| 0 | -0.3588 | -0.09243 | 0.271875 |
| 1624.63 | 1624.557 | 1629.259 | 1628.28 |
| 3420.26 | 3433.521 | 3430.135 | 3431.337 |
| 7472.01 | 7498.053 | 7491.732 | 7490.055 |
| 10282.49 | 10354.49 | 10340.09 | 10341.35 |
| 17667.49 | 17778.05 | 17767.56 | 17769.77 |
| 10282.49 | 10360.93 | 10344.46 | 10345.43 |
| 7472.01 | 7502.875 | 7495.425 | 7495.544 |
| 3420.26 | 3440.012 | 3434.201 | 3434.65 |
| 1624.63 | 1632.02 | 1635.305 | 1632.195 |
| 0 | 3.810357 | 1.04231 | 1.102393 |

Table 3.11: Hysteresis error test readings for the 5 kg load cell.

| | Average measured mass [g] | | |
|---|---|---|---|
| Real mass [g] | 1st try | 2nd try | 3rd try |
| 0 | 0.08275 | 0.521865 | 0.64575 |
| 225.57 | 225.3450189 | 226.235 | 226.3415 |
| 983.44 | 979.5138269 | 984.911 | 983.6207 |
| 2050.06 | 2046.250529 | 2052.712 | 2048.976 |
| 3104.5 | 3100.029239 | 3108.537 | 3103.038 |
| 4110.76 | 4106.737421 | 4117.659 | 4109.314 |
| 3104.5 | 3101.356934 | 3109.887 | 3104.626 |
| 2050.06 | 2047.067792 | 2053.807 | 2050.44 |
| 983.44 | 980.5869048 | 985.5866 | 984.3496 |
| 225.57 | 225.8130426 | 226.4493 | 226.9646 |
| 0 | 0.576918919 | 0.752719 | 0.841733 |

With this data, it is possible to obtain the Hysteresis error. For that, equation 3.23 was used. Tables 3.12 and 3.13 present the hysteresis error for all measurements on all tries

for each load cell.

$$\varepsilon_{hysteresis}[\%] = \frac{|Increasing\ Load - \ Decreasing\ Load|}{Full - scale\ value} \times 100 \tag{3.23}$$

Table 3.12: Hysteresis error values for the 20 kg load cell.

| $\varepsilon_{hysteresis}$ [%] - 20 kg load cell | | | |
|---|---|---|---|
| Mass [g] | 1st try | 2nd try | 3rd try |
| 0 | 0.0208% | 0.0057% | 0.0042% |
| 1624.63 | 0.0373% | 0.0302% | 0.0196% |
| 3420.26 | 0.0325% | 0.0203% | 0.0166% |
| 7472.01 | 0.0241% | 0.0185% | 0.0274% |
| 10282.49 | 0.0322% | 0.0218% | 0.0204% |

Table 3.13: Hysteresis error values for the 5 kg load cell.

| $\varepsilon_{hysteresis}$ [%] - 5 kg load cell | | | |
|---|---|---|---|
| Mass [g] | 1st try | 2nd try | 3rd try |
| 0 | 0.010% | 0.005% | 0.004% |
| 225.57 | 0.009% | 0.004% | 0.012% |
| 983.44 | 0.021% | 0.014% | 0.015% |
| 2050.06 | 0.016% | 0.022% | 0.029% |
| 3104.5 | 0.027% | 0.027% | 0.032% |

In order to not underestimate the error of the load cells it was decided to choose the biggest value recorded. This means that the hysteresis error for the 20 kg load cell is $\varepsilon_{hysteresis} = 0.037\%$ while for the 5 kg load cell is $\varepsilon_{hysteresis} = 0.032\%$.

**Non-repeatability error**

Non-repeatability refers to the difference of the measurements when the load cell is exposed to the same load repeatedly. This differences, in the case of the ETB can happen mainly due to vibrations caused by the engine on the work bench for the case of the 20 kg cell. In the case of the 5 kg one this can happen due to sloshing of the fuel inside the tank or even the position of the tank in relation to the axis of the load cell.

To obtain the value of the error, the test consists of 5 different trials with 5 different loads. The weights were placed on the load cell for 10 seconds and after that were removed. It was then checked if the reading returned to 0 g. This test is illustrated on figure 3.26.

Tables 3.14 and 3.15 present the data gathered during these tests for each load cell. After the data was gathered, the difference between each trial was calculated for each load. Table 3.16 presents this process for the biggest mass on the 20 kg cell. The other tables involved in this process are presented in C.4 Just as it was done with the hysteresis error, the biggest value was used to calculate the non-repeatability error of each load cell using equation 3.24.

Table 3.14: Non-repeatability error test readings for the 20 kg load cell.

| | Average measured mass [g] | | | | |
|---|---|---|---|---|---|
| Mass [g] | 1st try | 2nd try | 3rd try | 4th try | 5th try |
| 2783.85 | 2790.972 | 2807.124 | 2810.024 | 2801.132 | 2795.391 |
| 4579.48 | 4586.266 | 4599.279 | 4595.136 | 4580.598 | 4584.65 |
| 8631.23 | 8654.686 | 8640.248 | 8687.421 | 8713.053 | 8655.69 |
| 11441.71 | 11490.86 | 11511.77 | 11509.81 | 11506.19 | 11517.09 |
| 18826.71 | 18890.48 | 18809.92 | 18978.98 | 19003.38 | 19006.15 |

Table 3.15: Non-repeatability error test readings for the 5 kg load cell.

| | Average measured mass [g] | | | | |
|---|---|---|---|---|---|
| Mass [g] | 1st try | 2nd try | 3rd try | 4th try | 5th try |
| 225.57 | 225.305 | 225.4754 | 226.9172 | 225.9835 | 224.8324 |
| 983.44 | 985.4554 | 985.4622 | 984.7312 | 988.4372 | 986.4511 |
| 2050.06 | 2050.665 | 2058.14 | 2052.121 | 2055.729 | 2060.854 |
| 3104.5 | 3112.576 | 3109.758 | 3112.033 | 3111.802 | 3122.509 |
| 4110.76 | 4108.176 | 4116.216 | 4114.769 | 4115.551 | 4100.693 |

Table 3.16: Differences between each trials for the biggest mass used during testing on the 20 kg load cell.

| Difference - 18 826.71 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 80.55706 | 88.50699 | 112.9068 | 115.6792 |
| 2nd try | | 169.064 | 193.4638 | 196.2362 |
| 3rd try | | | 24.39979 | 27.17217 |
| 4th try | | | | 2.772386 |

$$\varepsilon_{non-repeatability}[\%] = \frac{|Reading\ A - Reading\ B|}{Full - scale\ value} \tag{3.24}$$

The non-repeatability error value was then obtained to be $\varepsilon_{non-repeatability} = 0.981\%$ for the 20 kg load cell and $\varepsilon_{non-repeatability} = 0.310\%$ for the 5 kg one.

**Creep error test**

The creep test, illustrated in figure 3.27, was made to ensure that the load cell can output the same value, when exposed to the same load for prolonged periods of time. This helps understand the characteristics of the transducers and also validate their performance under continuous loading.

The test was conducted by firstly exposing the load cell to a mass of 18 826.71 g for a long period of time for the 20 kg while the 5 kg cell was exposed to 4110.76 g and, after taring the cell, the same test was conducted but without any mass on the cell. When the test was finished the error could be obtained using equation 3.25.

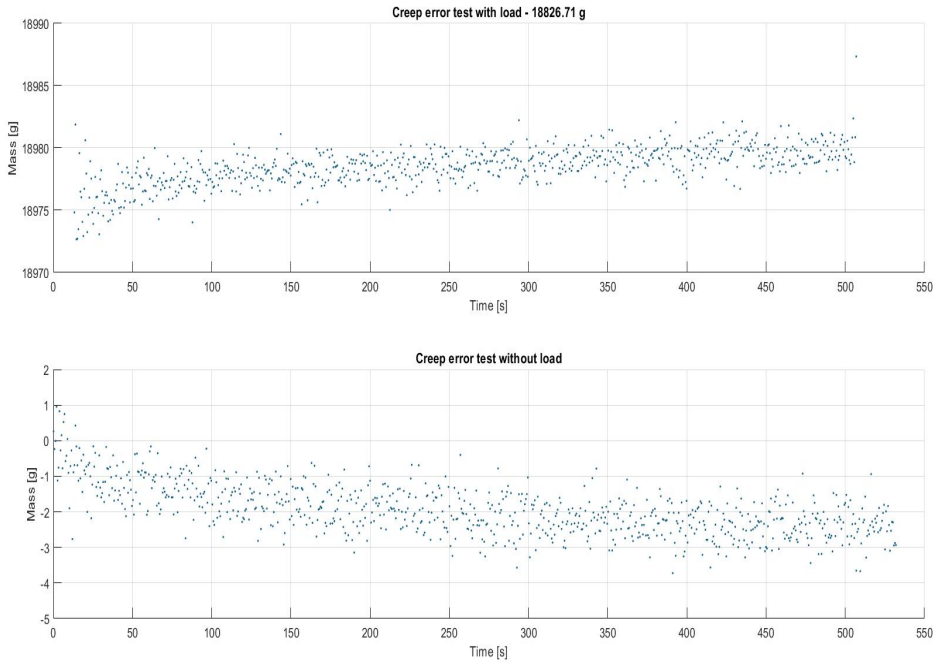



Figure 3.27: Creep test conducted on the 10 kg load cell.

$$\varepsilon_{creep}[\%] = \frac{|Maximum\ difference\ point - Initial\ point|}{Full - scale\ value} \times 100 \tag{3.25}$$

With this, it was found that for the 20 kg load cell, $\varepsilon_{creep} = 0.041\%$ while $\varepsilon_{creep} = 0.114\%$ for the 5 kg.

Table 3.17: Load cell full-scale errors.

| | 20 kg load cell | 5 kg load cell |
|---|---|---|
| $\varepsilon_{hysteresis}$ | 0.037% | 0.032% |
| $\varepsilon_{non-repeatability}$ | 0.981% | 0.310% |
| $\varepsilon_{creep}$ | 0.041% | 0.114% |
| $\varepsilon_{total}$ | 1.059% | 0.457% |

Table 3.17 presents the values of each type of error and their sum which corresponds to the total full-scale value of error in each load cell. These values were gathered in order to avoid underestimating the error of the load cells.

### 3.3.4 ETBCDS assembly

After the code development and calibration processes were finished, a circuit diagram was made in order to visualize the final result of the ETBCDS. This diagram is presented in figure 3.28.

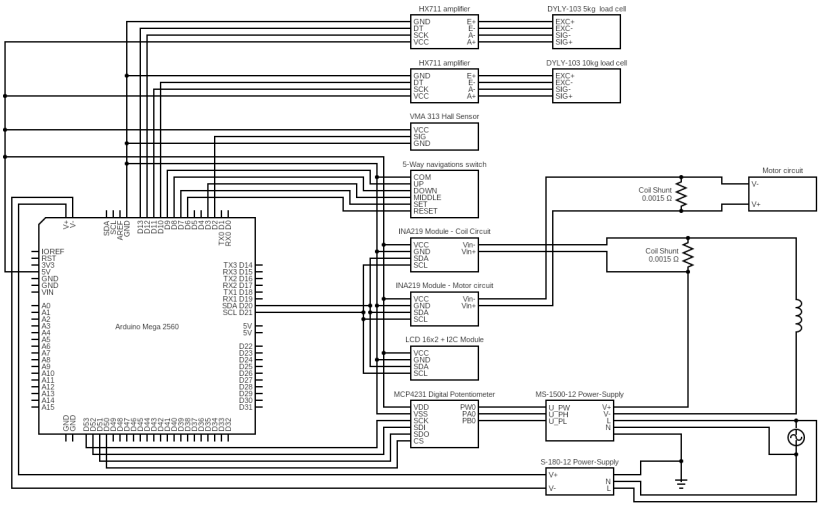



Figure 3.28: ETBCDS circuit diagram.

The assembly of the ETBCDS used an old desktop casing that was able to fit almost all the materials presented in 3.3.2. The assembly was done in order to avoid having the ETBCDS exposed to vibrations inherent to the operation of the ETB. These vibrations can cause inaccuracies on some sensors or even the loosening of contacts between wires which could jeopardize the operation of the ETB.

Figure 3.29 shows the layout of the ETBCDS. Although the box may seem too big for the amount of sensors it houses, it was selected in case the ETBCDS got any upgrades, which avoids the need of changing the housing entirely. The connections were made in order to avoid any kind of interferences as it was explained on 2.3.5, this meant having one exit for the power cables of the coil, EM and fan cables and other exit for the signal cables.

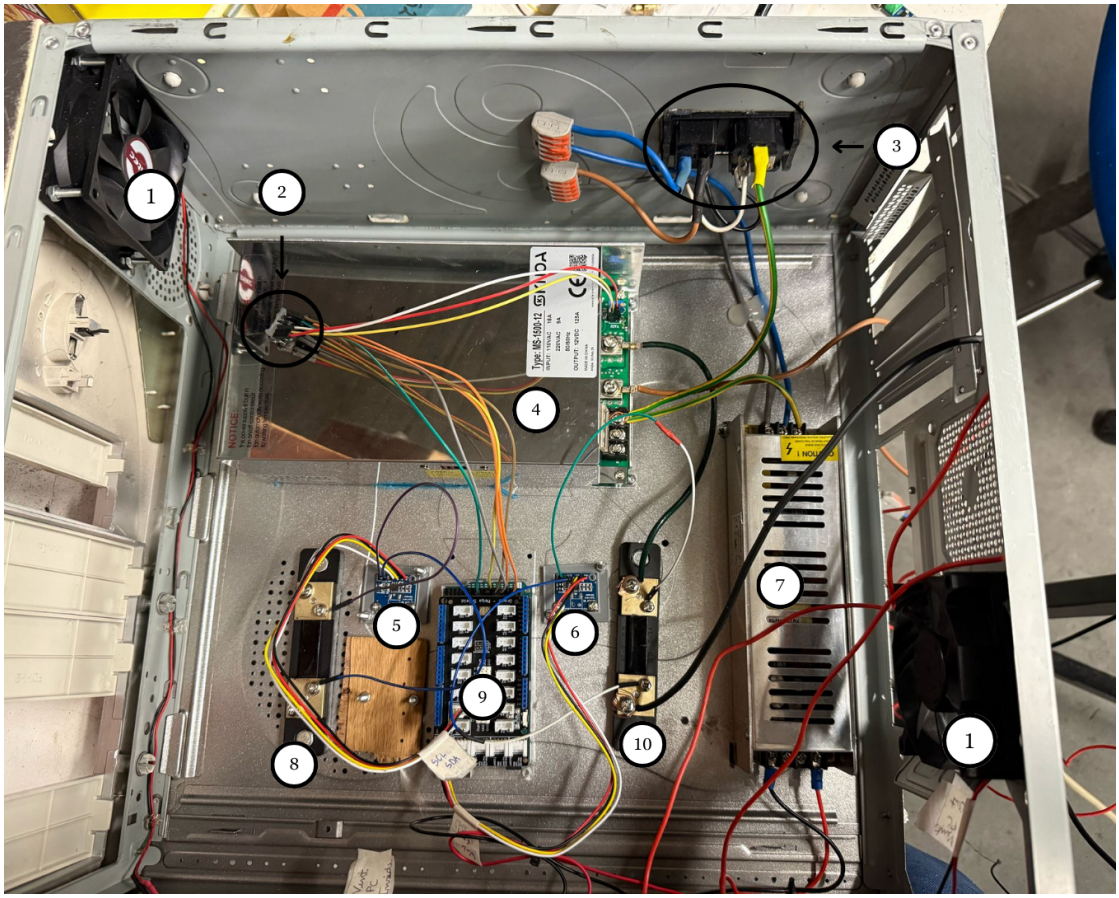



Figure 3.29: ETBCDS casing setup with the relevant components labelled: **1**-ETBCDS hardware cooling fans; **2**-MCP4231 DP; **3**-AC power inlet socket with integrated fuse and switch; **4**-MS-1500-12 PS; **5**-INA219 amplifier module used for measuring the current drawn by the EM; **6**-INA219 amplifier module used for measuring the current drawn by the coil; **7**-S-180-12 PS used for powering secondary systems; **8**-Shunt resistor used for measuring the current drawn by the EM; **9**-Arduino Mega 2560 R3; **10**-Shunt resistor used for measuring the current drawn by the coil.

# Chapter 4

# Assembly and testing of the Engine Test Bench

During the assembly of the ETB some problems were encountered and had to be solved. The first of them was the table top used to support the ECB and the engine that was deemed too weak to support the vibrations present during operation. This had to be taken into account since the other ETB developed in the laboratory by *Nunes* suffered from this phenomenon which caused bending on the structure thus not maintaining the perpendicularity between the ETB structure and the load cell.

To avoid this, it was decided to weld a steel beam from one end to the other that would pass beneath the location of the engines tested. This beam can be seen in place on figure 4.1.

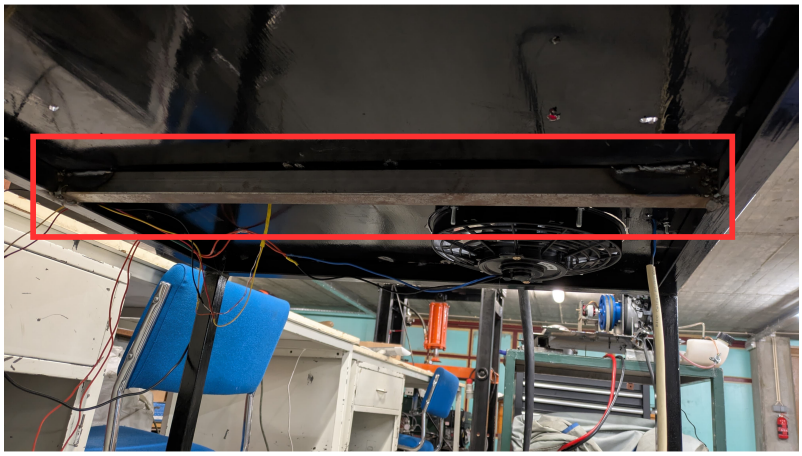


Figure 4.1: Steel beam used to reinforce the table top of the ECB.

This solution was found to be enough to avoid bending, since during testing this phenomena did not occur even in operation conditions in which vibrations were highest i.e. during the biggest load condition of the brake at lower speeds of the engine.

The second, and main problem faced was the fact that the coil PS stopped working after all components were assembled and tested independently. To trace the origin of this problem all connectors and wires were individually checked and found to be in good condition. After that it was found that the fuse installed in the switches' plate (3.3.2.9) had melted. After checking that the problem persisted after replacing it by a different one with higher current capacity, it was decided to open the MS-1500-12 PS in an attempt to trace the issue.

To do this it was necessary to desolder components from the PS Printed Circuit Board

(PCB). Not all components could be tested since the laboratory does not have that capacity. After this whole process the origin of the short-circuit could not be found and the MS-1500-12 PS had to be replaced.

Unfortunately, due to time constraints it wasn't feasible to order a new MS-1500-12 PS and a different model available in the laboratory had to be used. The characteristics of this PS can be found on table 4.1.

Table 4.1: OWP8010H PS information [50].

| | |
|---|---|
| Rated Power | 8000 W |
| Input Voltage | $3\phi$380 VAC $\pm$10% |
| Output Voltage | 0 - 100 V DC |
| Output Current | 0 - 100 A |
| Dimensions | $425 \times 132 \times 551$ mm |

Due to its bigger size and a more complex communication protocol and interface, the logic prepared for the MS-1500-12 model could not be used. It was decided to opt for a manual approach, in which the operator will have to set the load by changing the current on the coil circuit. Even though this is not what was intended since the beginning, due to time constraints, as said before, it was the only approach that was possible.

## 4.1 Testing of an Internal Combustion Engine

In order to validate the ETB it was decided to test an ICE and also an EM. Both of these engines were selected based on the availability in the laboratory and also taking into account its rated power and size.

In the case of the ICE, which will be reported first, it was decided to test a Honda GX35 engine which is represented on figure 4.2. Its characteristics can be consulted on table D.1.

In order to adapt the engine to the testing rig, it was necessary to do some modifications. First of all, since its main purpose is to power brush cutters its support points had to be changed in order to fix the engine into the table top. There were also some plastic covers that were removed since they were not necessary to test the engine and only made it more difficult to access some components such as the throttle and the carburettor. The final setup of the engine on the ETB can be seen on figure 4.3 (a). The fuel tank had to be also moved in order to acquire the fuel consumption data. For that it was necessary to use longer fuel lines and, at the same time ensure that these lines were tight to avoid any leakage and air getting into the system. The fuel tank and the load cell can be seen on figure 4.3 (b).


Figure 4.2: Honda GX35 engine [51].

(a) (b)

Figure 4.3: (a) GX35 engine integrated into the ETB; (b) GX35 fuel tank on top of the 5 kg load cell.

### 4.1.1 Methodology of testing

Initially, as presented at 3.3.1, it was decided to use a position-position mode of testing were both the throttle and the load would be set solely based on their position. However, since the system operation became more manual, thus allowing more time versatility, and the engine is not equipped with a Throttle Position Sensor (TPS) that could reliably register the throttle position, ensuring precise data it was decided to change the engine operation to the speed mode also presented in 3.3.1. Taking this into account the speeds at which the engine was tested were at 4000, 5000, 6000 and 7000 rpm. Even though, according to the manufacturer the engine maximum is of 8500 rpm, speeds over 7000 rpm were found to result in a quick heating up of the engine block and thus were considered to unsafe for longer periods of testing.

However, before testing, it was necessary to test the engine in a variety of loads in order to find the biggest load it could support. To find this value, the engine was working at

different values of throttle and the load was incrementally increased until the engine could not achieve the speeds presented previously. At the end of this process it was found that the engine could handle a maximum load of 8 A of current on the coil.

With this value decided, the following testing protocol was elaborated.

1. Start the engine;
2. Increase the speed until reaching 7000 rpm;
3. Tare the load cell (this step was taken to ensure that any small uncertainty was ruled out before loading the engine since with no load, the engine shouldn't create torque);
4. Increase the current on the coil circuit gradually to avoid major tensions on the pulley system until reaching $8\,\text{A}$;
5. Adjust the throttle to ensure the engine is at 7000 rpm;
6. Record the serial data from the Arduino during 1 minute;
7. Reduce the throttle until the engine reaches 6000 rpm;
8. Repeat the last 2 steps for the other 2 engine speeds.

### 4.1.2 Problems encountered

Even though most of the testing period of the ICE went smoothly without any issue, there were some factors that impaired it. First of all, the load cell used is capable of measuring $20\,\text{kg}$ which translates to a torque 12 times greater of what the engine tested will be generating at maximum. This resulted in a non-negligible degree of uncertainty that was surpassed by considering the best scenario in which the load cell detected the highest torque. This problem can also be solved by incorporating, on the empty side of the central structure another coil and core of the same weight that will balance the structure. This in turn makes the use of a load cell with a lower range possible which will ensure a smaller uncertainty and error. This approach was not taken also due to time constraints.

The second problem encountered during testing was ensuring that the fuel line was completely tight. While at bigger speeds the amount of vibrations was small, at lower speeds this was not the case. Even though the tank was not on the structure as the ETB, the vibrations were passed by the fuel lines and at the junction between the tank and the fuel pipes small bubbles of air entered the fuel hose. While in most times this didn't cause any significant changes on the engine operation, sometimes, if the air reached the carburettor, it could reduce engine performance or even shutting it down. When this happened the test was repeated to ensure that only significant data was recorded.

The third and last problem faced during the ICE testing was the amount of noise that the load cell recording the mass of the fuel tank faced. Since the engine also has a return line

this would cause some high variation on the fuel mass inside the tank. While initially the tests were repeated to avoid this phenomena, this approach was later abandoned since it is not possible to avoid it.

### 4.1.3 Results

Before presenting the results of the testing done to the GX35 engine it is necessary to present the power and torque curves provided by the manufacturer.

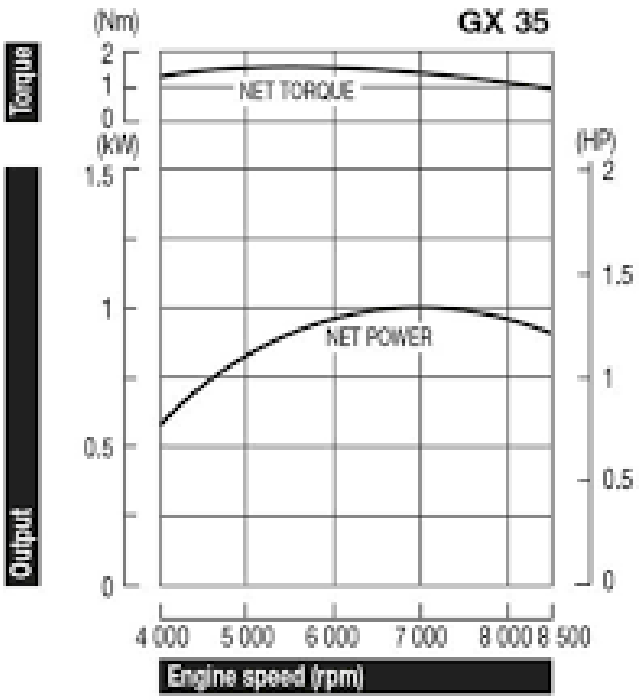



Figure 4.4: Honda GX35 power and torque curves provided by the manufacturer [51].

After testing, the serial data recorder was analysed using *Excel*. The speed, torque and power data were first checked to ensure that all significant noise was taken out. After this, the data was averaged out and the following table was elaborated.

Table 4.2: Speed, torque and power values recorded for the Honda GX35.

| N [rpm] | $\tau$ [N m] | $P_{shaft}$ [W] |
|---|---|---|
| 4176.833 | 0.505591 | 221.3431 |
| 5218.4 | 0.661622 | 361.5539 |
| 5981.217 | 0.65055 | 407.5072 |
| 7029.094 | 0.582658 | 428.8858 |

In order to visualize this, both the power and the torque curves were plotted as a scatter

plot and then trend lines that best fitted the data were calculated. Both scatter plots and trend lines are presented on figure together with the trend lines equations and the $R^2$ values.

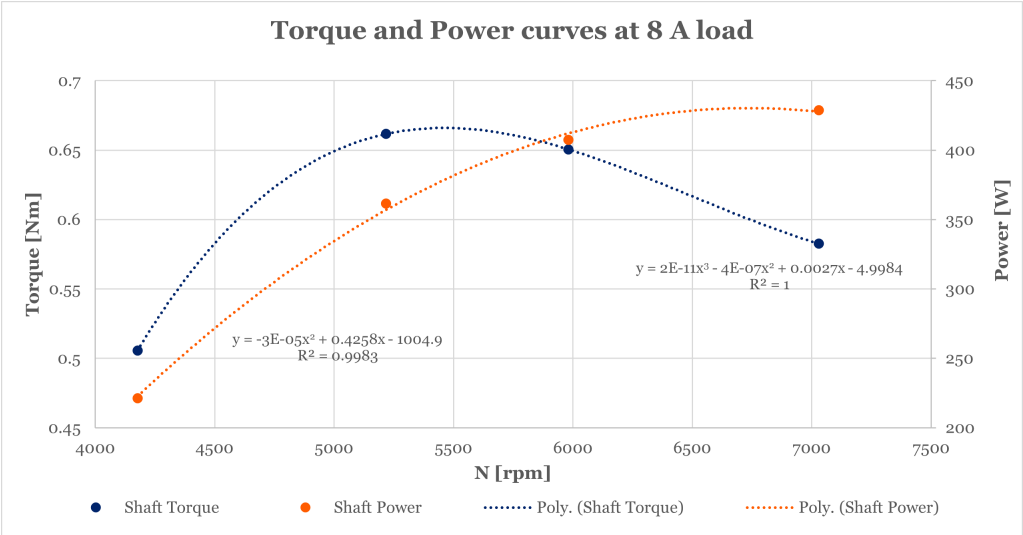



Figure 4.5: Honda GX35 power and torque curves obtained during test.

When the compared to the original curves presented in 4.4, the torque and power curves obtained during testing both have the same form as the original ones, with the torque curve peaking at around 5500 rpm and the power curve near 7000 rpm, as expected. Unfortunately, since it was not possible to test the engine at above 7000 rpm it was not possible to confirm that the engine power declines further at speeds above 7000 rpm. However, this is the most expected scenario since at higher speeds, as torque decreases even further, power will start to decrease as well because the engine is not capable of ingest the same amount air, thus becoming more inefficient [52].

However, when compared to the max values of power and torque provided by the manufacturer to the maximum values obtained through the trend lines it is possible to observe that during testing it was only possible to obtain 41.6% and 42.8% of maximum torque and power, respectively as can be seen on table 4.3.

Table 4.3: Maximum values of Torque and Power obtained in testing compared to the values provided by Honda.

| N [rpm] | $\tau_{manufacturer}$ [N m] | $\tau_{obtained}$ [N m ] | $\varepsilon$ [%] | $P_{manufacturer}$ [W] | $P_{obtained}$ [W] | $\varepsilon$ [%] |
|---|---|---|---|---|---|---|
| 5500 | 1.6 | 0.6660 | -58.3752 | | | |
| 7000 | | | | 1000 | 428.1819 | -57.1818 |

This high degree of error can be attributed to various things.

- The previously presented uncertainty related to the load cell.
- The pulley and belt system will cause some inefficiencies on the system through fric-

tion.

- The testing conditions present on the laboratory were not the exact ones used by the manufacturer when testing the engine which will inherently cause some differences on the results obtained.

As previously stated, the ETB was designed to also measure the amount of power consumed. In this case, the amount of fuel consumed by the engine. After plotting the data gathered by the load cell that can be consulted at D.2. The slope of each linear trend line was the used to calculate the fuel mass flow per minute and also the Specific Fuel Consumption (SFC) for each speed using the following equation.

$$\mathrm{SFC} = \frac{\dot{m}_f}{\mathrm{P}_{\mathrm{shaft}}} \tag{4.1}$$

Where $\dot{m}_f$ is the mass fuel flow per hour and $\mathrm{P}_{\mathrm{shaft}}$ is the power produced by the engine in kW. The mass fuel flow values were also used to calculate the thermal power consumed and also the thermal efficiency of the engine using equations 2.4 and 2.2. The heating value of gasoline used to calculate the thermal power consumed was $Q_{HV} = 47.3\,\mathrm{MJ\,kg^{-1}}$ [53]. The values obtained can be seen on table 4.4.

Table 4.4: Fuel flow, SFC, thermal power consumed and efficiency obtained for the GX35 engine.

| N [rpm] | $\dot{m}_f$ [$\mathrm{g\,h^{-1}}$] | SFC [$\mathrm{g\,kW^{-1}\,h^{-1}}$] | $\mathrm{P_{consumed}}$ [W] | $\eta$ [%] |
|---|---|---|---|---|
| 4176.833 | 341.1042 | 1541.066 | 4481.7300 | 4.9388 |
| 5218.4 | 334.6627 | 925.6231 | 4397.0959 | 8.2226 |
| 5981.217 | 328.1086 | 805.1603 | 4310.9826 | 9.4528 |
| 7029.094 | 455.4517 | 1061.942 | 5984.1296 | 7.1671 |

When plotted, the SFC values fit a parabolic curve that has the expected form for the SFC curve of an ICE, as can be seen on figure 4.6.

The efficiency and power consumed data were also plotted against the speed of the engine and their parabolic curves were also calculated and can be observed on in figure 4.7. As expected the power consumed of the engine initially drops but it starts to increase after a peak at around 5500 rpm as the fuel consumption also increases. On the other hand, the efficiency of the engine increases initially until it reaches a peak at around 6000 rpm, at that point it starts to decrease.

These results validate the effectiveness of the developed ETB for characterizing the performance of ICEs. Despite some deviation from manufacturer specifications, which can be attributed to measurement uncertainties and test conditions, the obtained torque, power, SFC, and thermal efficiency curves follow the expected theoretical trends. The ETB successfully captured the performance behaviour of the Honda GX35 engine across a wide

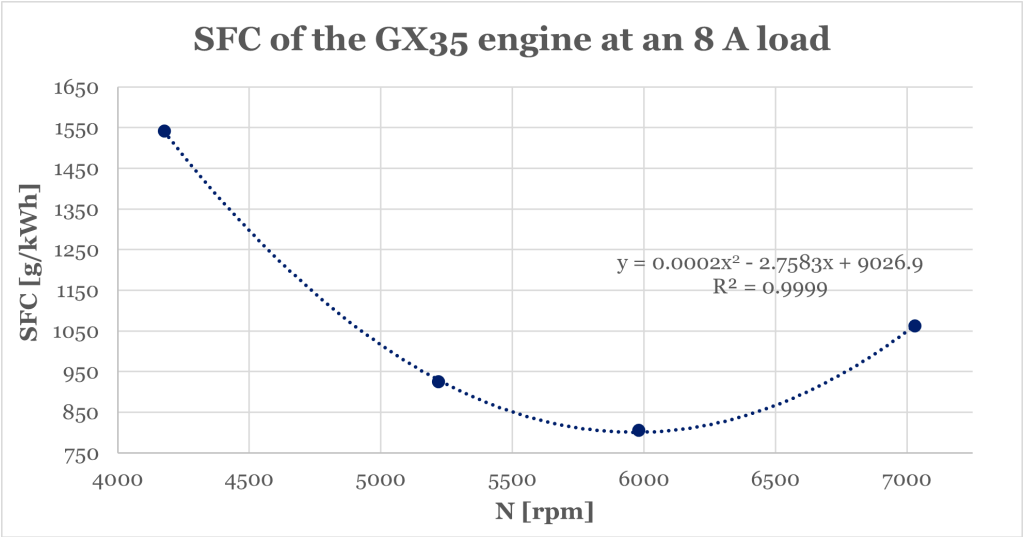



Figure 4.6: Honda GX35 SFC values and parabolic curve with an 8 A load.

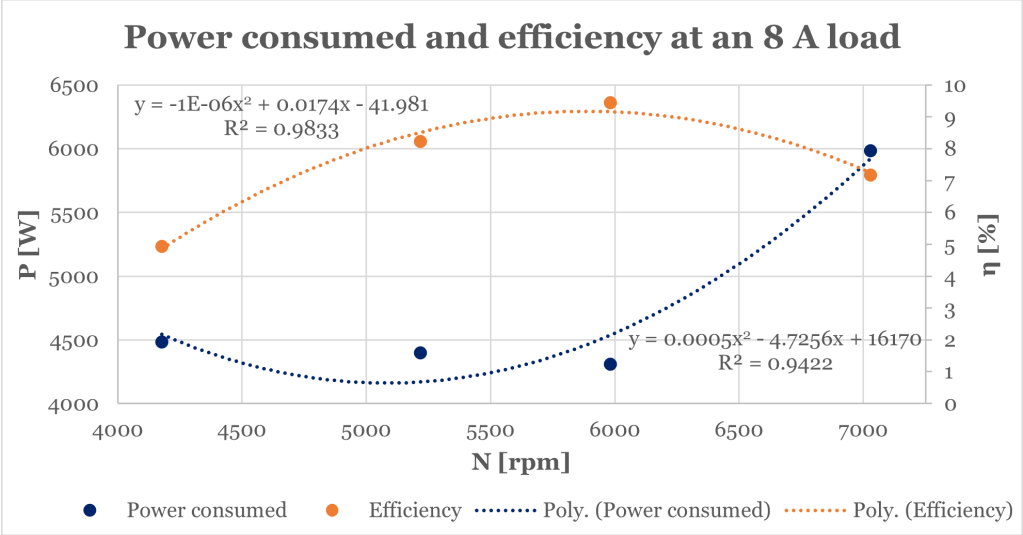



Figure 4.7: Power consumed and efficiency of the Honda GX35 engine with an 8A load.

range of speeds, confirming its suitability for future experimental studies and optimization of small-scale gasoline engines.

## 4.2 Testing of an EM

Even though the use of the ECB as a braking device has been validated with the testing of the GX35 engine, there was still the need of testing an EM to ensure that the ECB could reliably measure the electrical power consumed by a motor.

As with the GX35 engine, the motor tested had to be chosen from the ones available in the laboratory. Because of this, the model that best suited the ETB constructed was the

C4250 brushless motor. This EM has to be used together with a BuyWeek 40 A Electronic Speed Control (ESC). Both of these products characteristics' are presented on table 4.5.

Table 4.5: C4250 motor and ESC technical specs [54, 55].

| C4250 Outrunner Brushless motor specs | |
|---|---|
| Maximum power | 1760 W |
| Voltage range | 11.1 - 36 V |
| Motor speed | 6000 - 9000 rpm |
| Resistance | 0.068 Ω |
| No load current | 1.5 - 1.6 A |
| Torque | 1 - 2 N m |
| **BuyWeek 40 A ESC specs** | |
| Maximum stable current | 40 A |
| Connector type | XT60 |
| Maximum instantaneous current | 55 A |
| Battery capacity | 2 - 4 S |

Unfortunately, contrary to the GX35 engine it was not possible to gather data about the torque and power curves of this EM.

### 4.2.1 Methodology of testing

As with the ICE the control modes for the tests were the Speed-Position mode. This also has to do with the fact that it is not possible to register the position of the throttle of the motor. This fact, as with the Honda engine, would translate into a less reliable test if the position-position modes were used.

As will be explained later, during testing the ESC heated up considerably and because of that it was decided to use a small load. This small load of 2 A ensured a safer test even though the motor was not tested to its fullest capacity. Since the current shunt and INA219 were found to have small errors and were not subjected to a lot of noise, it was possible to reduce the amount of testing for each speed of the engine to 10 s in an effort to reduce the time that the engine was subjected to load.

With this in mind, the following protocol was elaborated.

1. Start the EM;
2. Increase the speed until reaching 9000 rpm;
3. Tare the load cell;
4. Set the current on the coil to 2 A;
5. Adjust the throttle to ensure the engine is at 9000 rpm;
6. Record the serial data from the Arduino during 10 s;

7. Reduce the throttle until the engine reaches 8000 rpm;
8. Repeat the last 2 steps for the other 2 engine speeds.

### 4.2.2 Problems encountered

As said previously in 4.2.1, the biggest difficulty encountered during testing was the fact that the ESC heated up considerably while the motor was under load. When the load was higher the temperature rose at a rate of $12\,^{\circ}\mathrm{C}\,\mathrm{min}^{-1}$.

This phenomena could not be ignored since there was an instance in which the temperature got so high that one of the ESC exploded. This can be seen on figure 4.8.

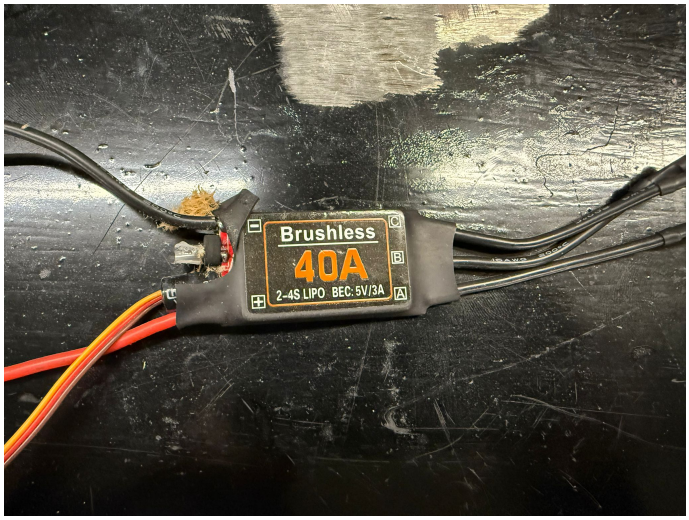



Figure 4.8: Exploded ESC during a test of the EM.

In order to avoid this, two different solutions were adopted. The first one was to reduce the amount of time in which the motor was subjected to the load. The second one was to secure the ESC to an ice block during testing in an effort to enhance the cooling of it as can be seen on figure 4.9. These 2 solutions proved to be adequate since the rate of heating reduced and this situation never repeated again.

Another issue faced during testing was the fact that the engine was not able to work properly, specially during start due to the fact that the ECB may have to much inertia for this type of motors. This motor, projected to propel an aircraft using a propeller, is not made to deliver high torque outside of its working speeds. Because of that even with 2 batteries on series delivering a 25.19 V voltage, the motor was only capable of achieving its minimum operating speed of 6000 rpm.

This will inevitably affect the results obtained, however, since the working capacities of the ECB have been validated through the test of the of the GX35 ICE, this test of the EM is essentially to validate the system that registers the current consumed by the EM and the ESC.

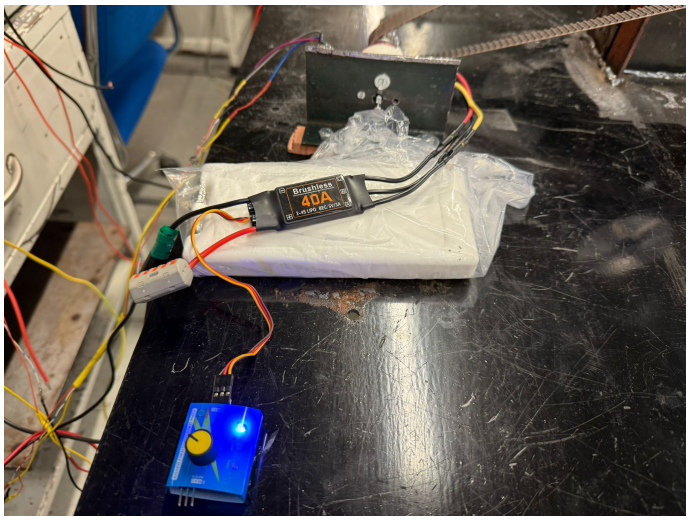



Figure 4.9: ESC set up used during testing to ensure good cooling of the system.

Due to this problem, the protocol of testing had to be altered since the motor did not achieve speeds greater than 6000 rpm. The new protocol can be consulted below.

1. Start the EM;
2. Increase the speed until maximum (in this case 6000 rpm);
3. Tare the load cell;
4. Increase the current on the coil to 2 A.
5. Record the serial data from the Arduino during 10 s;
6. Reduce the throttle until the speed of the motor reduces 1000 rpm;
7. repeat the last 2 steps until the speed of the motor is 2000 rpm.

### 4.2.3 Results

As talked previously in 4.2 it was not possible to have the manufacturer data about the power and torque curves of the motor. However, since we know that this motor is a brushless electric motor, the expected torque and power curve will have the following form.

As with the ICE, the data recorded was analysed using *Excel*. After checking the data for cleaning out the noise present, the data presented on table 4.6 was recorded.

Even without plotting the torque and power data it is possible to understand that the motor did not behave as the presented theoretical curves presented on figure 4.10.

As can be seen on figure 4.11, contrary to the theoretical curve, the torque of the motor increases linearly until the final speed of 6000 rpm. In the case of the power curve, it also

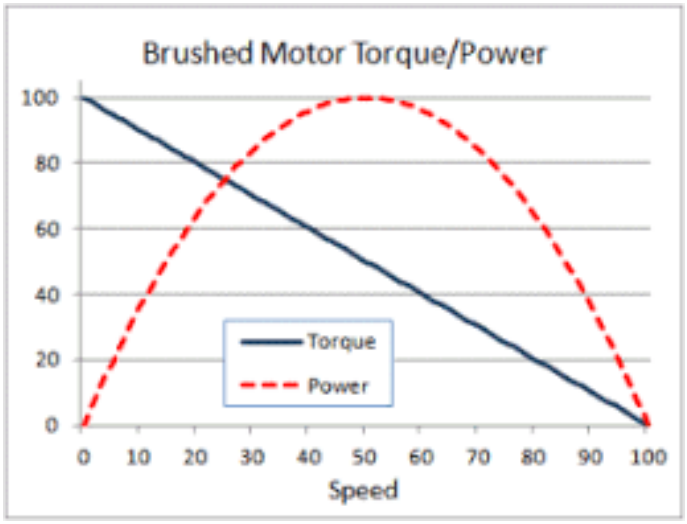



Figure 4.10: Theoretical power and torque curves of a brushless motor [56].

Table 4.6: Speed, torque and power values recorded for the EM.

| N [rpm] | $\tau$ [N m] | $P_{shaft}$ [W] |
|---|---|---|
| 2076.056 | 0.0790 | 17.1684 |
| 3128.683 | 0.0961 | 31.4813 |
| 4122.333 | 0.1167 | 50.3598 |
| 5016.929 | 0.1293 | 67.9083 |
| 6079.136 | 0.1571 | 99.9822 |

increases linearly, however this may mean that the expected peak of the parabolic curve may have not been reached yet.

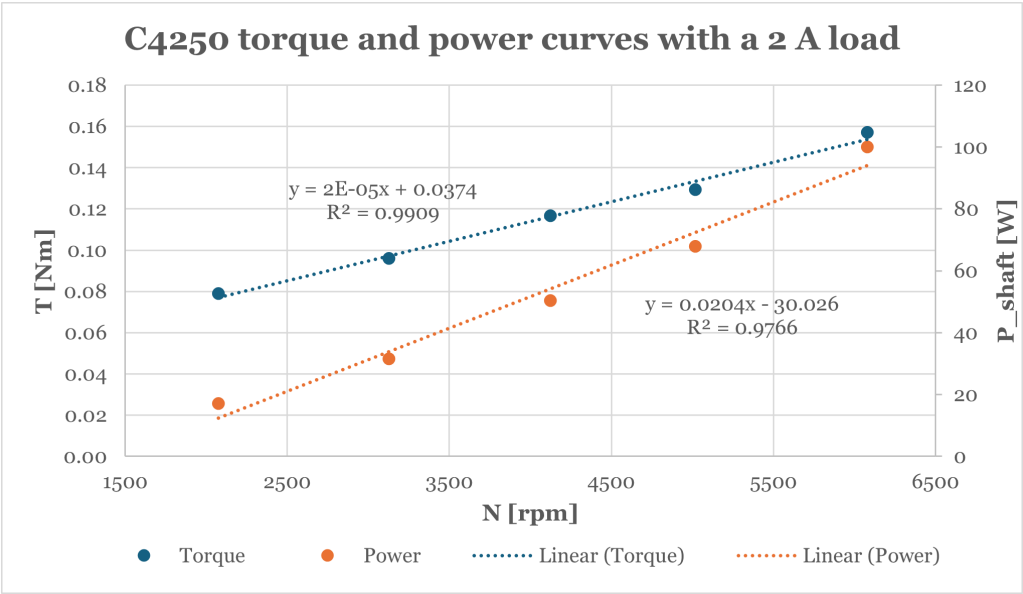



Figure 4.11: C4250 torque and power curves obtained during testing.

This phenomenon can be explained due to the motor not being able to reach its operational

speed range. To fix this, the engine may be subjected to a greater voltage which could help achieving higher speeds. This possibility was not applied due to a lack of more batteries or a PS capable of doing it.

Nonetheless, it was possible to register the current drawn out by the motor and consequently gather data about the power consumed and efficiency of the EM. This data is presented on table 4.7.

Table 4.7: Current, electrical power and efficiency values recorded for the C4250 EM.

| N [rpm] | I [A] | $P_{consumed}$ [W] | $\eta$ [%] |
|---|---|---|---|
| 2076.056 | 3.3967 | 85.5620 | 20.0654 |
| 3128.683 | 5.3883 | 135.7311 | 23.1938 |
| 4122.333 | 7.8767 | 198.4132 | 25.3813 |
| 5016.929 | 10.1239 | 255.0218 | 26.6284 |
| 6079.136 | 12.6309 | 318.1726 | 31.4239 |

When plotted, as in figure 4.12, it is possible to observe that both the electrical power consumed and the efficiency of the motor increase linearly. While this behaviour was expected in terms of power consumption, since for electric motors it is expected that motor consumes more energy, it was not expected that the efficiency of the engine would also increase. This could also be explained by the inertia of the system not allowing the motor to speed up to its operational range.

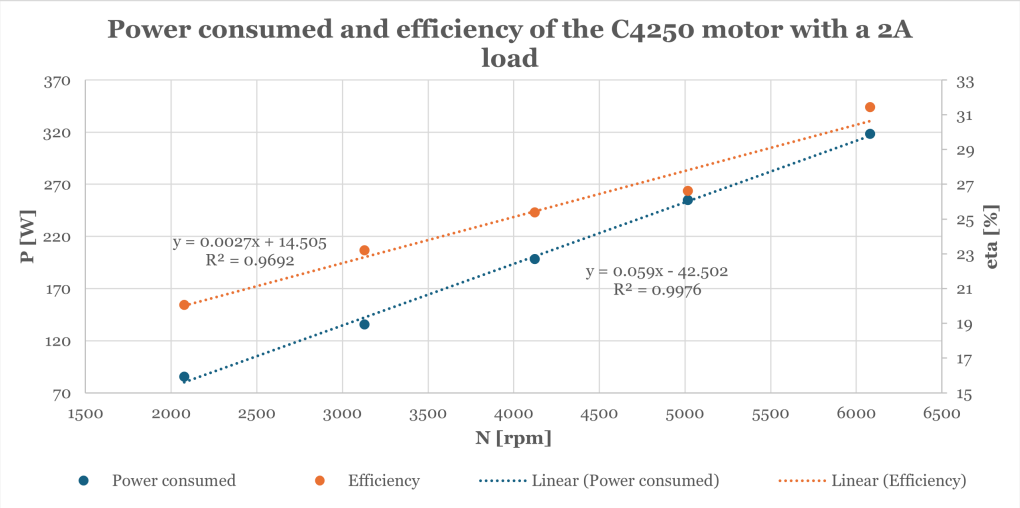



Figure 4.12: C4250 electrical power consumed and efficiency when subjected to a 2 A load.

Despite the challenges encountered during testing—such as excessive heating of the ESC and the motor's inability to reach its full operational range—the results obtained effectively validate the use of the ETB system. The data collected demonstrates the capability of the ETB to reliably measure the electrical power consumed and to function as a dynamometer for evaluating the performance of electric motors under load. However, it is important that further studies of the performance of EM using this ETB must have in mind the characteristics of this system to ensure good and viable results.

## 4.3 ECB and ETBCDS performance

As said previously, the number of turns of coil is not known for sure due to a error on the winding process. Because of this it was expected that there would be some differences on the braking power of the ECB. However, it was found that these differences were negligible and that the number of turns on the coil is approximately the projected.

Other factor that was assumed when studying the ECB was the temperature reached by the disk. As stated previously in 3.1.2, the assumed temperature was of $200\,^{\circ}\text{C}$. However, during testing the maximum measured temperature of the disk was of $47\,^{\circ}\text{C}$. This discrepancy might have to do with the fact that $2\,\text{kW}$ were expected to be dissipated while only 1 quarter of that value was indeed tested. The lower temperature also avoided the use of the big cooling fan which was found to induce a bit more of vibrations, that at lower power and torque might have influenced the results.

When compared to the 3D maps showed in figures 3.5 and 3.6 it is possible to state that the ECB behaved as expected, thus validating the algorithm developed.

During the data analysis process, it was noticed that there was no significant data lost by aliasing or quantization. The data collected showe all phenomena that were expected to be observed, and at the same time contained some spikes due to noise. Since the values were stored on `.txt` files, the amount of memory required for this process was low (at maximum 1.5 MB), which meant that no alteration had to be done to the sampling rate or the quantization standard that were set, by default, to the maximum.

# Chapter 5

# Conclusion

Looking in retrospective, it is safe to say that all but one of the goals proposed in 1.2 were achieved.

Initially a thorough study of the operation of an ETB was conducted in which the various types of dynamometers were presented and analysed in order to justified the choice made later on during the design fase.

After the choice in favour of the ECB was made, a MATLAB code, taking into consideration the magnet's core saturation and the critical speed of it was done. This allowed a better understanding of the operation of a ECB while, at the same time, also allowed a more confident determination of its key parameters such as the distance from the center of disk, the number of turns on the coil, the material of the disk and the current on the coil.

The outsourcing of the winding process of the coil and the use of a 3D printed mould allowed a faster and more streamlined process and improved the overall quality of construction when compared to previous ECBs assembled in the laboratory. A good cable management also ensured that no interferences between power and signal cables was felt during the operation of the ETB.

Although in the end the MC4231 DP was not used due to the failure of the MS-1500-12 PS, this system was proved to be functioning, and in the future, when the faulty PS is replaced, can be put to work immediately without major work needing to be done.

Lastly, it was not possible to test a HEPS. This happened due to the fact that at the time of the conclusion of the practical work of this dissertation there was no model of HEPS available for testing in the laboratory. In order to surpass this problem two separate tests were conducted, one using an ICE which validated the use of the ETB and its capabilities of registering fuel consumption data. The other used an EM which also validated the the ECB and the capability of the ETB of measuring and registering the current consumed by the EM.

## 5.1 Challenges and limitations

During the course of this work some major challenges and limitations were faced. The major challenge faced was time. Since this project was intended to be concluded during one semester, there were some key features that were not considered to be developed and could improve the quality of the project. These include for example the capacity of

transient testing, the replacing of the faulty MS-1500-12 PS or even a better integration of the ETBCDS and the new OWP8010H PS. The time that took for the arrival of certain parts such as the aluminium plates which delayed the conclusion of the dissertation, also challenged the work conducted.

Another big challenge faced was the fault that occurred on the main PS of the ETB which meant that the original design and project had to be greatly altered in order to finish the work. However, since no great modification in the physical assembly were done and the original code is still available, it is possible to re-integrate the MS-1500-12 model into the ETB.

Lastly, the two biggest limitations faced during testing were the uncertainty of the 20 kg load cell and the inertia of the system. The first one caused multiple repetitions of the test to ensure that the results obtained were usable, considering the manufacturer data of the engine. The second one caused a high rate of heating of the ESC used and prevented the motor from reaching its operational speed which caused the results of torque and power to be contrary to what was expected.

## 5.2 Future Work

As with any work, there is always room for improvements and further development. For this work, the next points were identified as being the improvements that would bring more value into the ETB.

- Replace the MS-1500-12 PS with a new one or integrate the OWP8010H PS which has better communication protocols into the ETBCDS.

  This action would result in a semi-automatic ETB that would work as initially intended and alleviate the workload of the testing operator.

- Assemble a new core and coil and replace the 20 kg load cell.

  As said previously, this action would better balance the current structure which would then eliminate the need for such a big load cell, thus creating an opportunity to use a smaller one such as a 10 kg. Apart from this, with another core+coil set, the ETB would also benefit from a bigger braking power and allow it to test bigger engines/motors. Another benefit is the fact that since the disk is being subjected to forces opposed to each other it is expected that the vibrations would decrease overall.

- Add transient testing capabilities.

  As referred previously, it was decided to only develop the static testing capabilities of the ETB. However, adding the transient capabilities to this device would greatly improve its value.

- Integrate the ETBCDS into a broader platform such as LabView.

  By using a more capable and complex platform such as LabView it would be possible to control at the same time the throttle and ECB in the same place. This would unlock other control modes that are unfeasable at this time. Other benefit is the fact that it would be possible to plot at real time the various data collected.

# Bibliography


[1] H. Ritchie, "Sector by sector: where do global greenhouse gas emissions come from?," *Our World in Data*, 2020. https://ourworldindata.org/ghg-emissions-by-sector. 1

[2] "Net-Zero Carbon Emissions by 2050 — iata.org." `https://www.iata.org/en/pressroom/pressroom-archive/2021-releases/2021-10-04-03/`. Accessed 09-08-2024. 1

[3] C. Ronneau, *Énergie, pollution de l'air et développement durable*. Presses univ. de Louvain, 2013. 1

[4] J. M. T. Matlock, *Evaluation of hybrid-electric propulsion systems for unmanned aerial vehicles*. PhD thesis, 2019. 1

[5] R. E. Batzer, *Design and Construction of a Dynamometer*. PhD thesis, Massachusetts Institute of Technology, 2011. 1

[6] S. Kusumba, *Dynamometer proportional load control*. PhD thesis, Cleveland State University, 2004. 1

[7] "Horsepower | Definition, Unit, and Facts | Britannica — britannica.com." `https://www.britannica.com/science/horsepower`. [Accessed 29-05-2025]. 5

[8] J. S. Killedar, *Dynamometer: theory and application to engine testing*. Xlibris Corporation, 2012. 5, 7, 8, 9, 10, 11, 12, 13, 14, 20

[9] W. Yehia and M. Mostafa, "Practical considerations for marine propeller sizing," pp. 717–724, 06 2014. 6

[10] D. Halliday *et al.*, "Fundamentos de física," 2001. 6

[11] A. J. Martyr and M. A. Plint, *Engine testing: theory and practice*. Elsevier, 2011. 7, 8, 9, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 22, 23

[12] A. J. Martyr and D. R. Rogers, *Engine testing: electrical, hybrid, IC engine and power storage testing and test facilities*. Butterworth-Heinemann, 2020. 7, 10, 12, 14, 22

[13] M. Kaisan and G. Pam, "Determination of engine performance parameters of a stationary single cylinder compression ignition engine run on biodiesel from wild grape seeds/diesel blends," *Journal of Energy, Environment & Carbon Credits*, vol. 3, no. 3, pp. 15–21, 2013. 8

[14] N. R. Mate and D. Dhande, "Design and development of two wheeler retarder type dynamometer portable test platform," *International Journal of Engineering Research & Technology*, vol. 3, no. 2, 2014. 8

[15] R. Schaley, J. Seibert, B. Wolden, T. Bon, and G. Bora, "Small engine test stand proposal," 2011. 9, 10, 11

[16] C. A. Novo, "A low power engine test stand," 2016. 9, 13, 15, 20, 28, 38

[17] E. Soriano, F. Blaya, J. M. Del Burgo, M. Islán, and R. D'Amato, "Filament advance detection sensor for fused deposition modelling 3d printers," 04 2018. 10

[18] J. Raine and P. Hodgson, "Computer simulation of a variable fill hydraulic dynamometer; part 1: torque absorption theory and the influence of working compartment geometry on performance," *Proceedings of the Institution of Mechanical Engineers, Part C: Mechanical Engineering Science*, vol. 205, no. 3, pp. 155–163, 1991. 11, 12

[19] A. J. R. Nunes and F. M. R. P. Brojo, "Designing an eddy current brake for engine testing," *KnE Engineering*, pp. 743–756, 2020. 13, 14, 28, 31, 32, 33, 37, 38

[20] K. Lee and K. Park, "Optimal robust control of a contactless brake system using an eddy current," *Mechatronics*, vol. 9, no. 6, pp. 615–631, 1999. 14

[21] J. Wouterse, "Critical torque and speed of eddy current brake with widely separated soft iron poles," in *IEE Proceedings B (Electric Power Applications)*, vol. 138, pp. 153–158, IET, 1991. 13, 30

[22] D. J. T. Cruz, "Design of an innovative car braking system using eddy currents," 2005. 14

[23] W. Voos, "Dynamic engine testing," tech. rep., SAE Technical Paper, 1992. 16, 19

[24] R. Priemer, *Introductory signal processing*, vol. 6. World scientific, 1991. 20, 21, 22

[25] B. G. Liptak, *Instrument Engineers' Handbook, Volume One: Process Measurement and Analysis*. CRC press, 2003. 21

[26] E. Por, M. van Kooten, and V. Sarkovic, "Nyquist–shannon sampling theorem," *Leiden University*, vol. 1, no. 1, pp. 1–2, 2019. 21, 22

[27] R. Gray and D. Neuhoff, "Quantization," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2325–2383, 1998. 22

[28] D. G. Chirkov and A. D. Stotckaia, "Methods of software data filtering for working with sensors in the field of robotics," in *2020 XXIII International Conference on Soft Computing and Measurements (SCM)*, pp. 121–124, 2020. 23, 24

[29] S. W. Smith, *The scientist and engineer's guide to digital signal processing*. California Technical Pub., 2002. 24

[30] J. Grochowicz, K.-H. Wollenweber, C. Agudelo, and H. Abendroth, "Brake dynamometer test variability-analysis of root causes," tech. rep., SAE Technical Paper, 2010. 27

[31] J. A. N. Farinha, "On the design and study of a parallel hybrid electric propulsion test apparatus for uavs," 2022. Available at https://fenix.tecnico.ulisboa.pt/downloadFile/1970719973969198/Tese.pdf. 28

[32] Q. Zhou, X. Guo, G. Tan, X. Shen, Y. Ye, and Z. Wang, "Parameter analysis on torque stabilization for the eddy current brake: a developed model, simulation, and sensitive analysis," *Mathematical Problems in Engineering*, vol. 2015, no. 1, p. 436721, 2015. 30, 31, 32

[33] G. Sokolov, "Analysis of electrodynamic brake for utilization in systems with rotating shafts," 2016. 30

[34] E. Simeu and D. Georges, "Modeling and control of eddy current brake," *IFAC Proceedings Volumes*, vol. 28, no. 8, pp. 109–114, 1995. 30

[35] Honda, "Manual de instrues - gx25, gx35, gx 50." https://www.honda-engines-eu.com/files/files/owners-manual-gx25-35-50-portuguese-3pz3v6000.pdf. Accessed 14-11-2024. 31

[36] J. D. Kraus, K. R. Carver, and S. H. Burns, "Electromagnetics," *American Journal of Physics*, vol. 45, no. 1, pp. 113–114, 1977. 36

[37] "Electrical Resistivity Data - Ness Engineering Inc. — nessengr.com." https://www.nessengr.com/technical-data/resistivity/. Accessed 15-11-2024. 36

[38] J. R. Barnes, *Robust Electronic Design Reference Book: no special title*, vol. 2. Springer Science & Business Media, 2004. 36

[39] C. Moosbrugger, *ASM ready reference: electrical and magnetic properties of metals*. Asm International, 2000. 36

[40] "S type load cells dyly-103 - calt sensor — caltsensor.com." https://caltsensor.com/product/s-type-load-cells-dyly-103/. Accessed 11-11-2024. 40

[41] "Hx711 - amplificador p/ células de carga/sensores de peso duplo canal." https://www.botnroll.com/pt/forca-pressao-vibracao/2980-amplificador-p-c-lulas-de-carga-sensores-de-peso-hx711.html. Accessed 19-02-2025. 40

[42] M. K. Kazimierczuk, *High-frequency magnetic components*. John Wiley & Sons, 2009. 41

[43] "Open-frame power supply unit module 12VDC, 400W, IP20 — orno.pl." https://www.orno.pl/en/open-frame-power-supplies/

403-open-frame-power-supply-unit-module-12vdc-400w-ip20-5908254803352.html#download. Accessed 20-11-2024. 49

[44] Arduino.cc, "Arduino Mega 2560 Rev3 pinout datasheet." https://docs.arduino.cc/resources/pinouts/A000067-full-pinout.pdf, 2010. Accessed 25-11-2024. 50

[45] M. T. Incorporated, "MCP4XXX 7/8-Bit Single/Dual SPI Digital POT with Volatile Memory." https://www.microchip.com/en-us/product/mcp4231#Overview. Accessed 27-02-2025. 51

[46] S. Campbell, "Basics of the SPI Communication Protocol — circuitbasics.com." https://www.circuitbasics.com/basics-of-the-spi-communication-protocol/. Accessed 27-02-2025. 51

[47] "Power in Electric Circuits | Ohm's Law | Electronics Textbook — allaboutcircuits.com." https://www.allaboutcircuits.com/textbook/direct-current/chpt-2/power-electric-circuits/. Accessed 25-11-2024. 52

[48] "5-Way Navigation Switch | Joy-IT — joy-it.net." https://www.joy-it.net/en/products/COM-5WS. Accessed 21-02-2025. 55

[49] "LCD 16x2 I2C - Robotparts — robotparts.pt." https://robotparts.pt/produto/lcd-16x2-i2c/. Accessed 21-02-2025. 55

[50] "Owon OWP8010H power supply — eleshop.eu." https://eleshop.eu/owon-owp8010h-power-supply.html. Accessed 22-05-2025. 66

[51] "Honda engines - GX35 — honda-engines-eu.com." https://www.honda-engines-eu.com/en/products/power-units/gx35. Accessed 23-05-2025. 67, 69, 113

[52] K. Cheenkachorn, C. Poompipatpong, and C. Ho, "Performance and emissions of a heavy-duty diesel engine fuelled with diesel and lng (liquid natural gas)," *Energy*, vol. 53, p. 52–57, 05 2013. 70

[53] M. Noor, A. Wandel, and T. Yusaf, "The development of mild combustion open burner experimental setup," 07 2013. 71

[54] "Buyweek 40a controlador de velocidad esc sin escobillas." https://www.amazon.es/dp/B0BGC1QQ6T?ref_=cm_sw_r_cso_wa_apan_dp_K1WS3F9XBE39G6511BGM&previewDoh=1. Accessed 26-05-2025. 73

[55] "C4250 outrunner brushless motor." https://www.amazon.es/dp/B07QC32PP8?starsLeft=1&ref_=cm_sw_r_cso_wa_apan_dp_AVBC2SAXT0A954PFZMVR. Accessed 26-05-2025. 73

[56] “FingerTech Robotics — fingertechrobotics.com.” https://www.fingertechrobotics.com/brushless-torque.php. Accessed 27-05-2025. 76

# Appendix A

# MATLAB Code

```
clc;
clear all;

% Data
eddy = [0.08;... %a
0.004;... % thickness of the disk
0.040;... % width of the pole shoe
4000;... % maximum rpm
1100;... % Number of turns on the coil
1;... % number of coils
0.007;... % distance between the pole shoes
4 * pi() * 10^(-7);... % permeability of vacuum
200;... % temperature of the disk
20;... % current
1.5]; % limit of magnetic saturation of the core

% Materials matrix with characteristics (resistivity, temp. coeff.,...
%rel. permeability)
materials = [1.69 * 10^(-8) 4.30 * 10^(-3) 1;... % Copper
2.65 * 10^(-8) 4.50 * 10^(-3) 1;... % Aluminium
73.00 * 10^(-8) 0.94 * 10^(-3) 1.02];


materials_name = ["Copper"; "Aluminium"; "Austenitic Stainless Steel";];

% Additional parameters added to eddy array
eddy(12) = eddy(3) / sqrt(pi()); % Equivalent radius of pole shoe
eddy(13) = pi() * eddy(12)^2; % Cross-sectional area of pole shoe

%% Compute Results for Each Material and Rotational Speed

w_rpm = 0:eddy(4);
w_rad = w_rpm * 2 * pi() / 60; % Angular speed in rad/s

Copper = zeros(eddy(4)+1, 3);
Aluminium = zeros(eddy(4)+1, 3);
ASS = zeros(eddy(4)+1, 3);
Iron = zeros(eddy(4)+1, 3);
MS = zeros(eddy(4)+1, 3);
ES = zeros(eddy(4)+1, 3);

% Calculate power dissipated, breaking torque, and flux density for...
%each material
for n = 1:eddy(4) + 1
Copper(n,:) = results(w_rad(n), eddy, materials(1,:));
Aluminium(n,:) = results(w_rad(n), eddy, materials(2,:));
ASS(n,:) = results(w_rad(n), eddy, materials(3,:));

end

% Plot Power Dissipated
figure;
hold on;
plot(w_rpm, Copper(:, 1), 'b', 'LineWidth', 1, 'DisplayName', 'Copper');
plot(w_rpm, Aluminium(:, 1), 'g', 'LineWidth', 1, 'DisplayName',...
'Aluminium');
plot(w_rpm, ASS(:, 1), 'c', 'LineWidth', 1, 'DisplayName',...
'Austenitic Stainless Steel');
xline(750, '--r', 'LineWidth', 1, 'DisplayName', 'Idle Speed of ICE');
xlabel('Rotational Speed [rpm]');
ylabel('Power Dissipated [kW]');
title('Power Dissipated vs Speed');
legend('Location', 'best');
grid on;
hold off;

% Plot Breaking Torque
```

```
figure;
hold on;
plot(w_rpm, Copper(:, 2), 'b', 'LineWidth', 1, 'DisplayName', 'Copper');
plot(w_rpm, Aluminium(:, 2), 'g', 'LineWidth', 1, 'DisplayName',...
'Aluminium');
plot(w_rpm, ASS(:, 2), 'c', 'LineWidth', 1, 'DisplayName',...
'Austenitic Stainless Steel');
xline(750, '--r', 'LineWidth', 1, 'DisplayName', 'Idle Speed of ICE');
xlabel('Rotational Speed [rpm]');
ylabel('Breaking Torque [Nm]');
title('Breaking Torque vs Speed');
legend('Location', 'best');
grid on;
hold off;

% Plot Magnetic Flux Density
figure;
hold on;
plot(w_rpm, Copper(:, 3), 'b', 'LineWidth', 1, 'DisplayName',...
'Copper');
plot(w_rpm, Aluminium(:, 3), 'g', 'LineWidth', 1, 'DisplayName',...
'Aluminium');
plot(w_rpm, ASS(:, 3), 'c', 'LineWidth', 1, 'DisplayName',...
'Austenitic Stainless Steel');
xline(750, '--r', 'LineWidth', 1, 'DisplayName', 'Idle Speed of ICE');
xlabel('Rotational Speed [rpm]');
ylabel('Magnetic Flux Density [T]');
title('Magnetic Flux Density vs Speed');
legend('Location', 'best');
grid on;
hold off;

%% Critical Rotation Speed and Maximum Breaking Torque Calculation
% Calculations for each material
n_mat = size(materials, 1);
for i = 1:n_mat
rho_0 = materials(i, 1);
alfa = materials(i, 2);
miu_r = materials(i, 3);

sigma = 1 / (rho_0 * (1 + alfa * eddy(9)));
w_c(i, 1) = round(((eddy(7) - eddy(2)) / (eddy(8) * sigma * eddy(1)...
* eddy(2) * eddy(12)) + eddy(2) / (miu_r * eddy(8) * sigma...
* eddy(1) * eddy(2) * eddy(12))) * 60 / (2 * pi()));

% Ensure w_c does not exceed bounds
if w_c(i, 1) > size(Copper, 1)
w_c(i, 1) = size(Copper, 1);
end
end

% Output critical speed and breaking torque
fprintf('\n%s:\n - w = %.0f rpm, %.3f Nm of breaking torque;\n',...
materials_name(1), w_c(1, 1), Copper(w_c(1, 1), 2));
fprintf('%s:\n - w = %.0f rpm, %.3f Nm of breaking torque;\n',...
materials_name(2), w_c(2, 1), Aluminium(w_c(2, 1), 2));
fprintf('%s:\n - w = %.0f rpm, %.3f Nm of breaking torque;\n',...
materials_name(3), w_c(3, 1), ASS(w_c(3, 1), 2));

%% 3D Plots for Different Currents

% Define current values and RPMs
current_values = [20, 16, 12, 8, 4, 2, 0];
eddy_v2 = eddy;

w_rpm2 = [4000, 3750, 3500, 3250, 3000, 2750, 2500, 2250, 2000, ...
1750, 1500, 1250, 1000, 750, 500, 250, 0];
w_rad2 = w_rpm2 * 2 * pi() / 60;

% Preallocate result matrices based on w_rpm2 and current_values sizes
num_speeds = length(w_rpm2);
num_currents = length(current_values);

data_power = zeros(num_speeds, num_currents);
data_torque = zeros(num_speeds, num_currents);
data_flux_density = zeros(num_speeds, num_currents);

% Calculate results for different current values
for j = 1:num_currents
eddy_v2(10) = current_values(j);
for i = 1:num_speeds
```

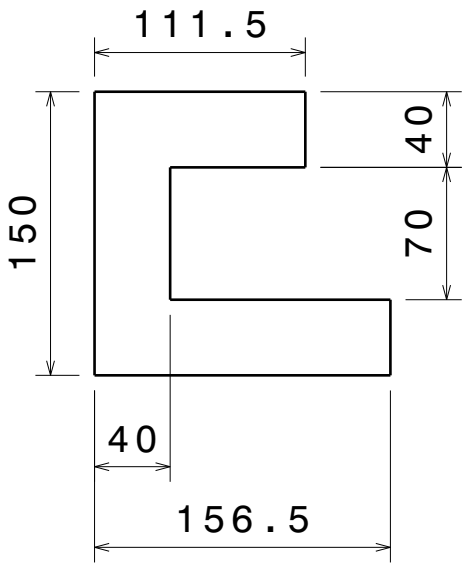
```
149                 if w_rpm2(i) == 0
150                 data_power(i, j) = 0;  % Set power to 0
151                 data_torque(i, j) = 0;  % Set torque to zero
152                 data_flux_density(i, j) = 1.5;  % Set flux density to 1.5
153                 else
154                 ASS = results(w_rad2(i), eddy_v2, materials(2, :));
155                 data_power(i, j) = ASS(1);           % Power dissipated
156                 data_torque(i, j) = ASS(2);          % Breaking torque
157                 data_flux_density(i, j) = ASS(3);    % Magnetic flux density
158                 end
159                 end
160                 end
161
162
163                 % Prepare meshgrid for 3D plot
164                 [w_rpm_grid, current_grid] = meshgrid(w_rpm2, current_values);
165
166                 % 3D plot for Power Dissipated
167                 figure;
168                 surf(w_rpm_grid, current_grid, data_power', 'FaceColor', 'interp',...
169                 'EdgeColor', 'k');
170                 xlabel('Rotational Speed [rpm]');
171                 ylabel('Current [A]');
172                 zlabel('Power Dissipated [kW]');
173                 title('Power Dissipated vs Speed and Current');
174                 colormap(jet); % Apply color map
175
176                 grid on;
177
178                 % 3D plot for Breaking Torque
179                 figure;
180                 surf(w_rpm_grid, current_grid, data_torque', 'FaceColor', 'interp',...
181                 'EdgeColor', 'k');
182                 xlabel('Rotational Speed [rpm]');
183                 ylabel('Current [A]');
184                 zlabel('Breaking Torque [Nm]');
185                 title('Breaking Torque vs Speed and Current');
186                 colormap(jet); % Color map
187
188                 grid on;
189
190                 % 3D plot for Magnetic Flux Density
191                 figure;
192                 surf(w_rpm_grid, current_grid, data_flux_density', 'FaceColor', 'interp'...
193                 , 'EdgeColor', 'k');
194                 xlabel('Rotational Speed [rpm]');
195                 ylabel('Current [A]');
196                 zlabel('Magnetic Flux Density [T]');
197                 title('Magnetic Flux Density vs Speed and Current');
198                 colormap(jet);
199
200                 grid on;
201
202
203
204                 %%
205
206                 function y = results(w_rad,eddy, materialdata)
207                 %this part divides the array materialdata into different variables for
208                 %easier comprehension
209                 rho_0 = materialdata(1);
210                 alfa = materialdata(2);
211                 miu_r = materialdata(3);
212
213                 sigma = 1/(rho_0*(1+alfa*eddy(9)));%electric conductivity
214                 R_a = (eddy(7)-eddy(2))/(eddy(8)*eddy(13));%reluctance of the air
215                 R_d = eddy(2)/(miu_r*eddy(8)*eddy(13));%reluctance of the disk
216                 R_g = R_a + R_d;%global reluctance
217
218                 phi_c = (eddy(5)*eddy(10))/(R_g + sigma*eddy(1)*eddy(2)*eddy(12)*w_rad...
219                 /eddy(13));%magnetic flux at this specific angular speed
220                 B_c = phi_c/eddy(13);%magnetic flux density at this specific angular
221                 %speed
222
223                 if B_c >= eddy(11)
224                 B_c = eddy(11); %the value of the magnetic flux density is equal to
225                 %the value of the magnetic flux density of saturation if the
226                 %calculated value is greater than the magnetic flux density of
227                 %saturation of the core.
228                 end
229
```

```
230                 rho = rho_0*(1+alfa*eddy(9));
231
232                 J_e = sigma * eddy(1) * w_rad * B_c;%calculates the induced eddy
233                 %currents density
234                 P_d = eddy(6)* J_e^2 * eddy(13) * eddy(2) * rho * 10^(-3);%calculates
235                 %the dissipated power
236                 T_b = P_d/w_rad*10^3;%calculates the breaking torque of the ECB
237                 y = [P_d T_b B_c];
238                 end
239
```

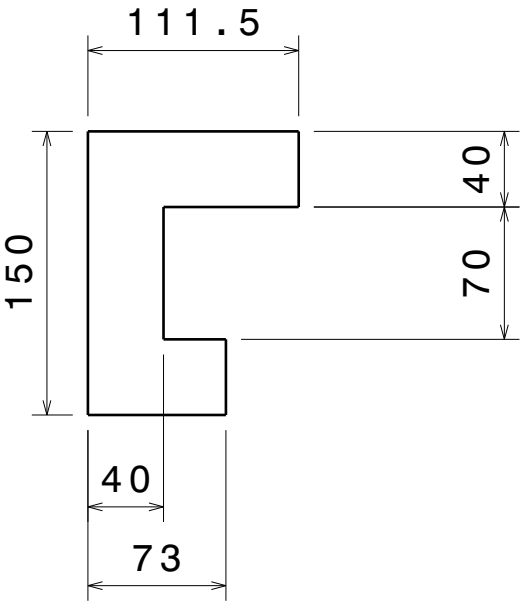
# Appendix B

# Technical Drawings

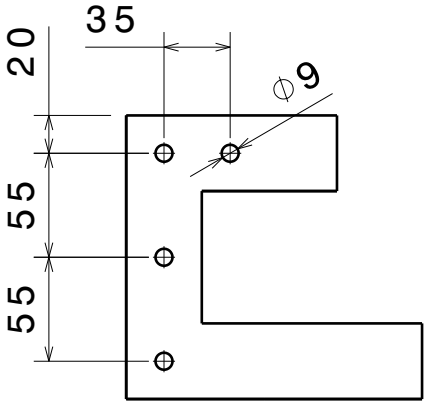
## B.1 Iron Core



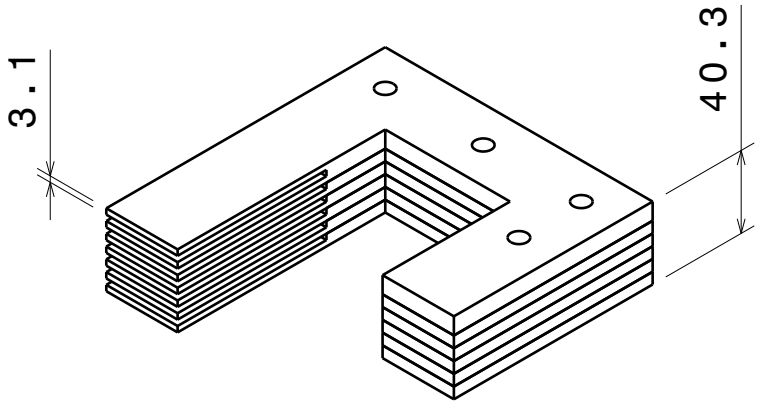

Front view - Single plate
Big




Front view - Single plate
Small

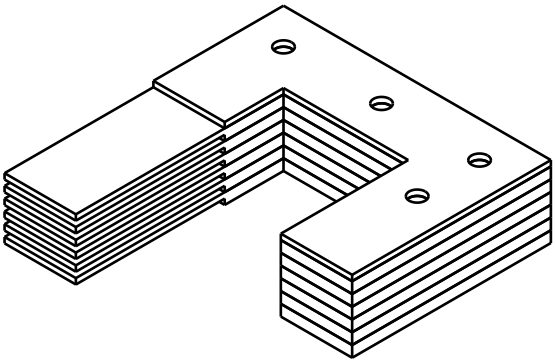



Front view - Half Core

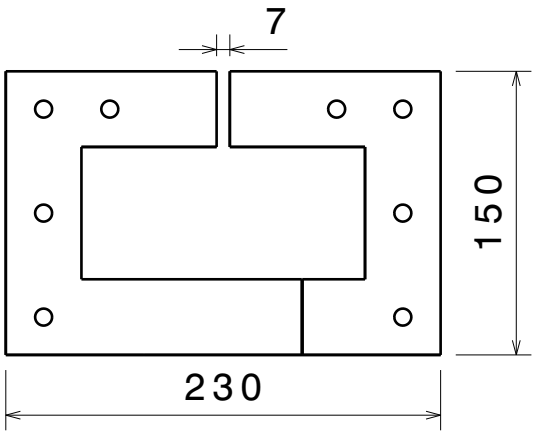



Isometric view - Left Half of the Core
Scale: 1:3

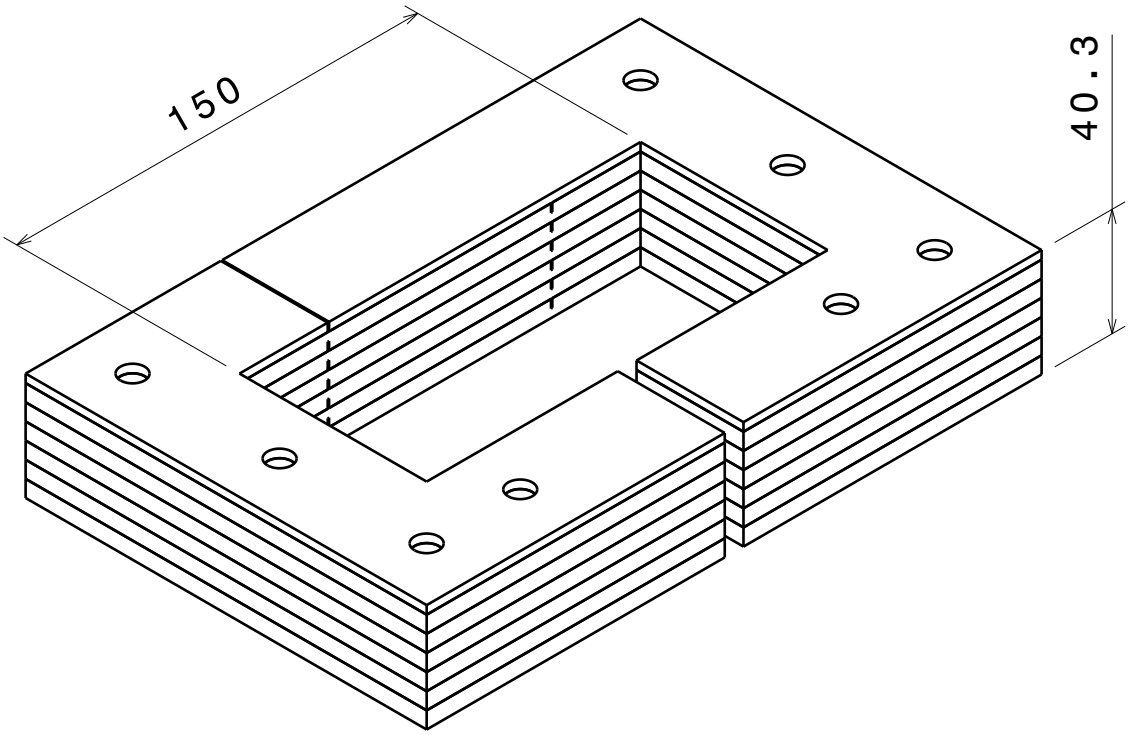
| | | | |
|---|---|---|---|
| DESIGNED BY: Manuel Azevedo<br>DATE: 23/06/2024 | # Iron Core | | I _ |
| | | | H _ |
| | | | G _ |
| MASS (kg) 7.354 | | | F _ |
| SIZE A4 | CMAST - Propulsion Lab | | E _ |
| | | | D _ |
| SCALE 1:4 | NOTES: All dimensions in mm. | SHEET 1/1 | C _ |
| | | | B _ |
| | | | A _ |



Isometric view - Right half of the Core
Scale: 1:3




Top view - Full Core

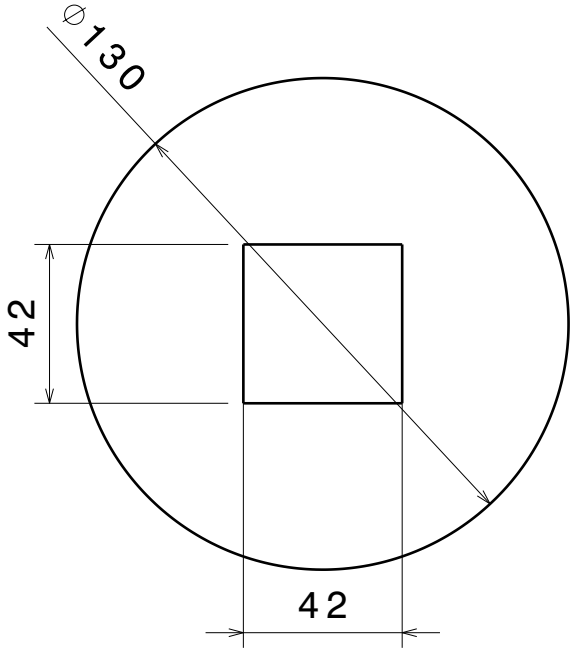


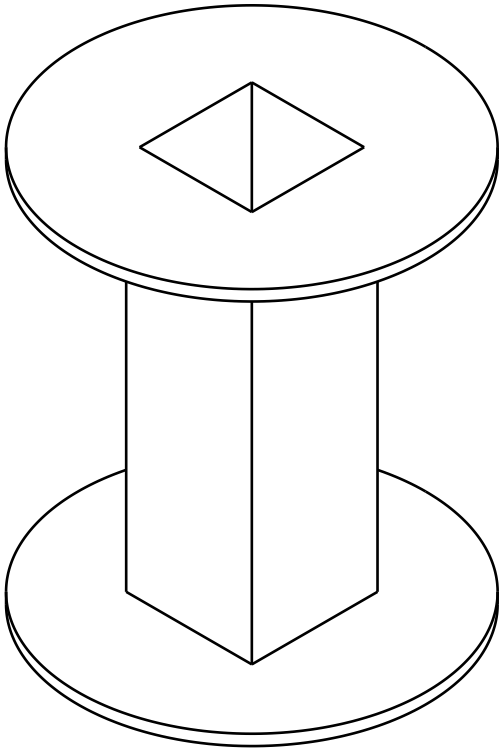

Isometric view - Full Core
Scale: 1:2

| | | | |
|---|---|---|---|
| DESIGNED BY: Manuel Azevedo<br>DATE: 23/06/2024 | # Iron Core | | I _ |
| MASS (kg) 7.354 | | | H _ |
| | | | G _ |
| | | | F _ |
| SIZE A4 | CMAST - Propulsion Lab | | E _ |
| | | | D _ |
| SCALE 1:4 | NOTES All dimensions in mm. | SHEET 1/2 | C _ |
| | | | B _ |
| | | | A _ |

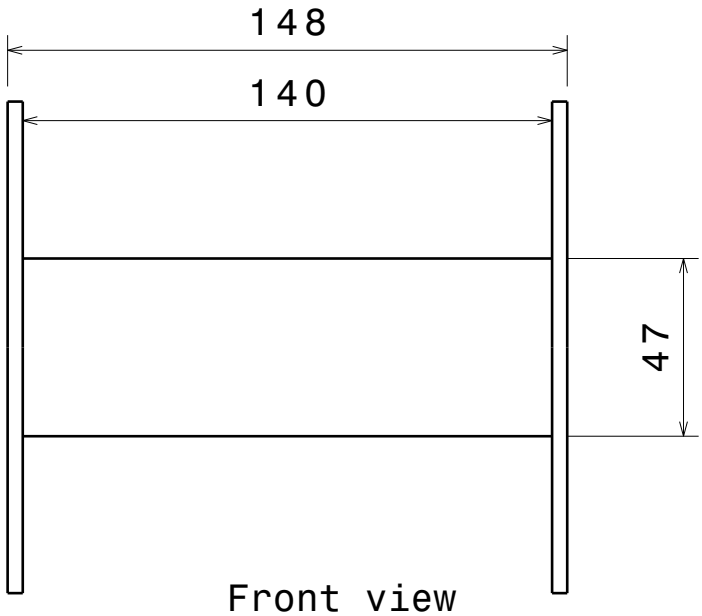
## B.2 Coil Support

Front view



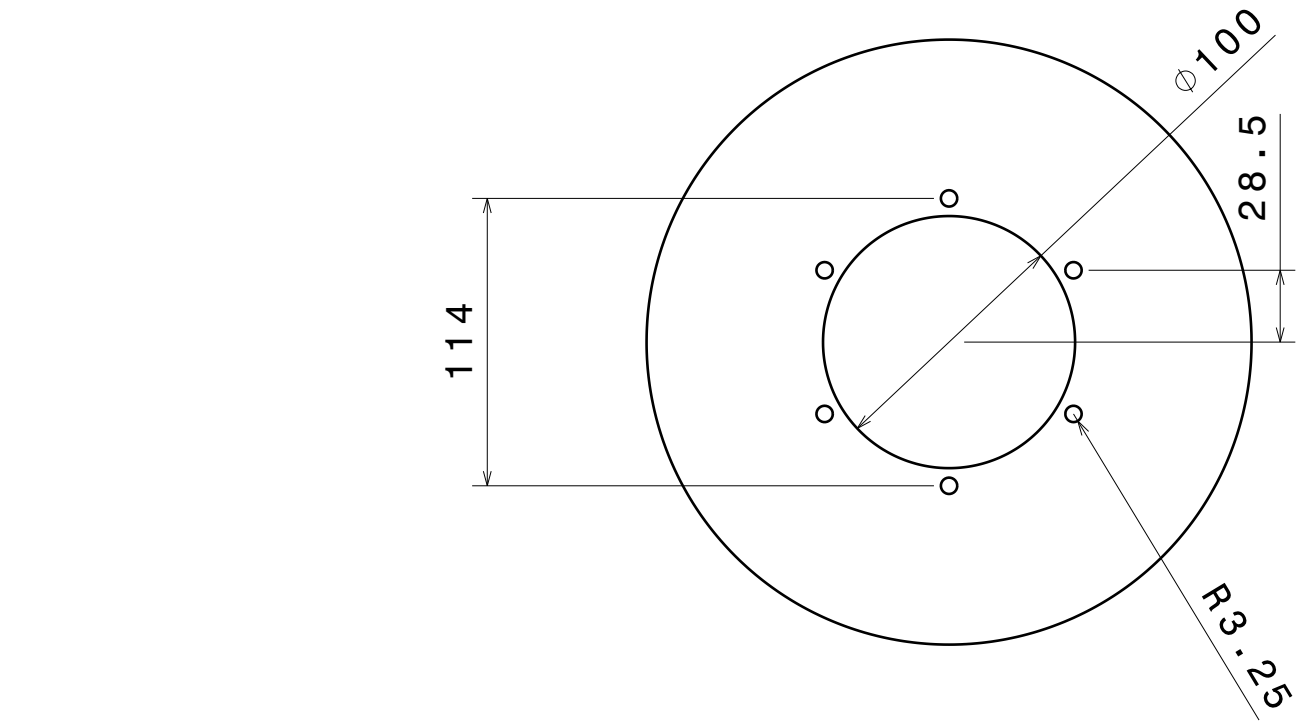
Isometric view



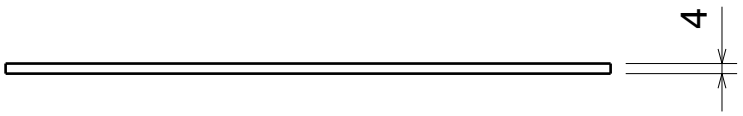

Front view

| | | | |
|---|---|---|---|
| DESIGNED BY: Manuel Azevedo<br>DATE: 15/06/2024 | | Coil Central Support | I _<br>H _<br>G _<br>F _<br>E _<br>D _<br>C _<br>B _<br>A _ |
| WEIGHT (g): 69.61 | | | |
| SIZE: A4 | | CMAST - Propulsion Lab | |
| SCALE: 1:2 | MATERIAL: PLA | NOTES: All dimensions in mm | SHEET: 1/1 |

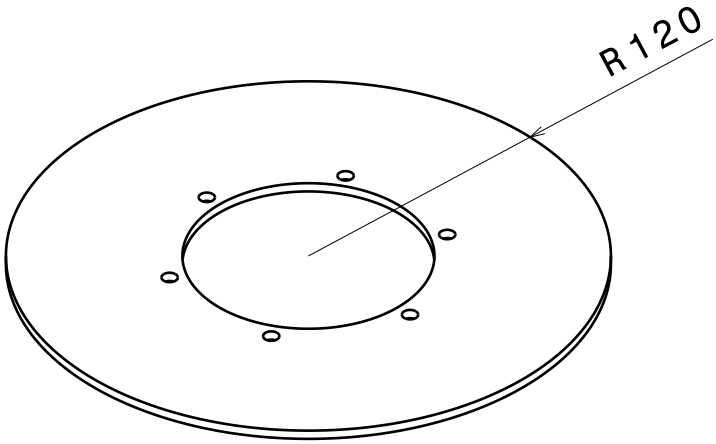
## B.3 Rotating Disk




Front view



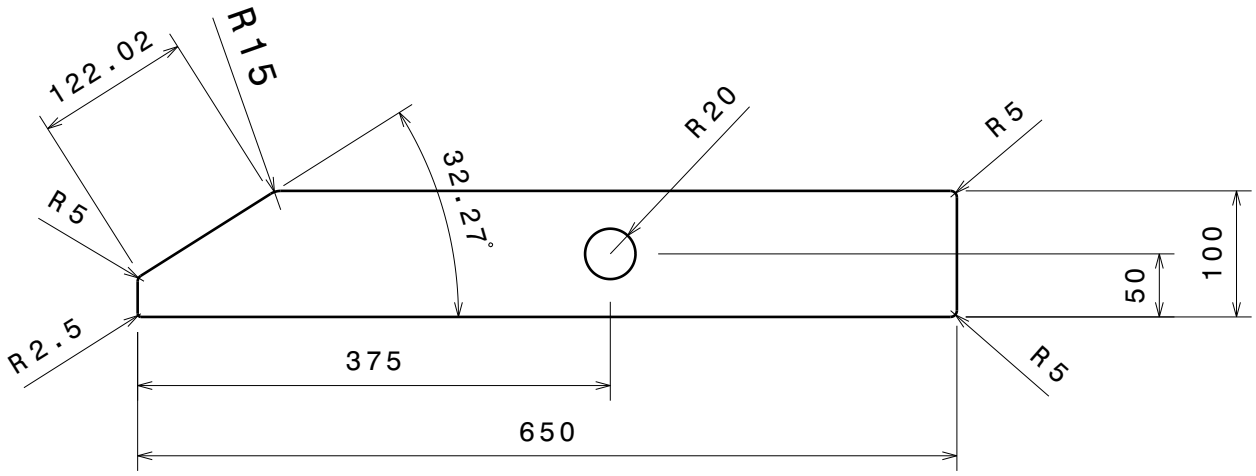

Side view



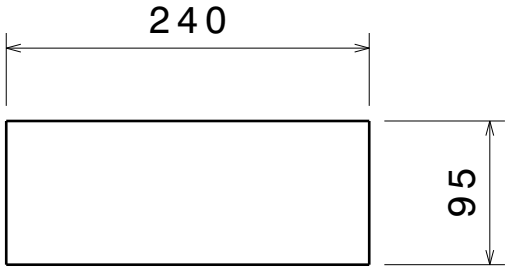

Isometric view

| | | | |
|---|---|---|---|
| DESIGNED BY: Manuel Azevedo<br>DATE: 15/10/2024 | # Rotating Disk | | I _ |
| | | | H _ |
| | | | G _ |
| WEIGHT (kg) 1.621 | | | F _ |
| SIZE A4 | CMAST Propulsion Lab | | E _ |
| | | | D _ |
| SCALE 1:3 | Notes: All dimensions in mm. | SHEET 1/1 | C _ |
| | | | B _ |
| | | | A _ |

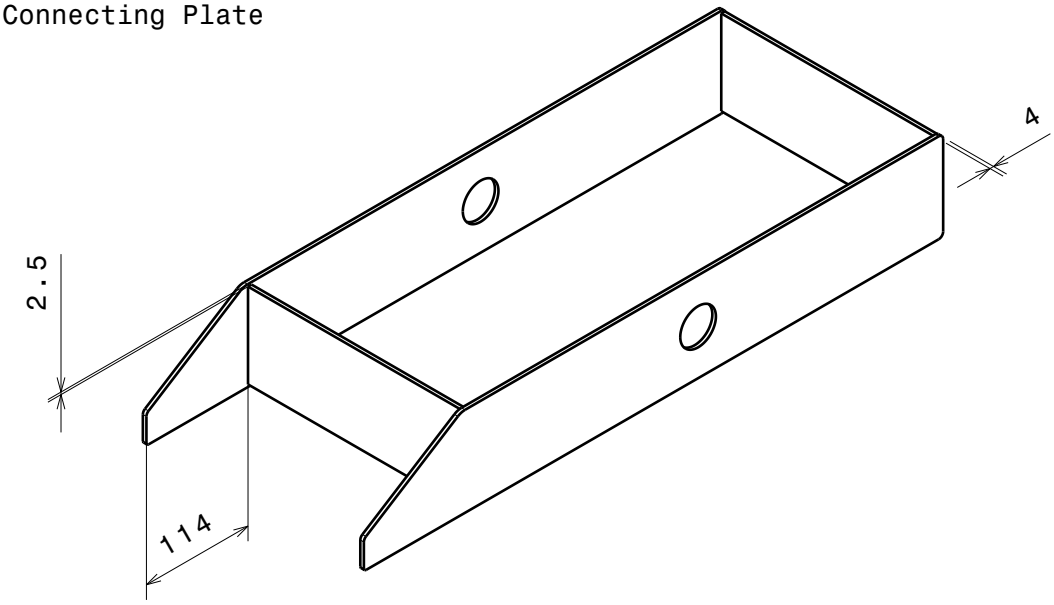
## B.4 Central Structure




Front view - Single Plate

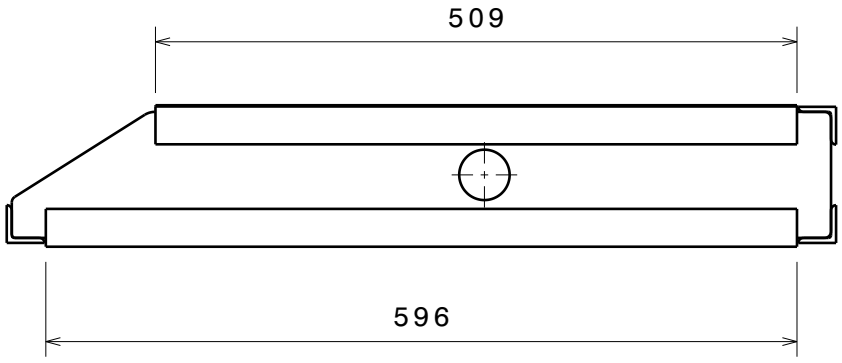



Front view - Connecting Plate
Scale: 1:5

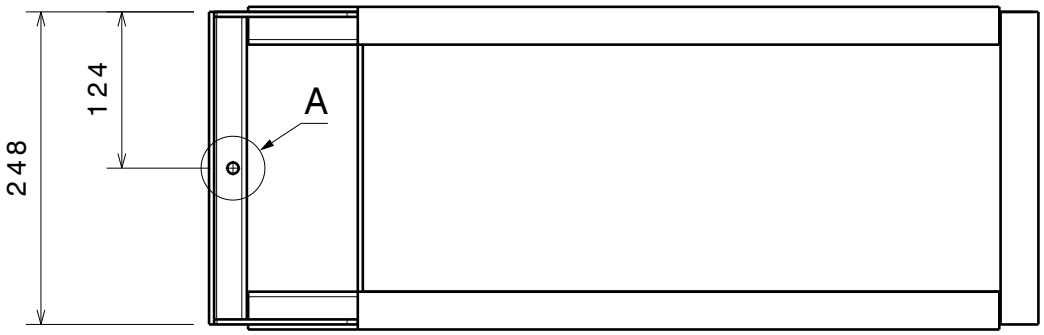



Isometric view - Longitudinal plates
and connecting plates

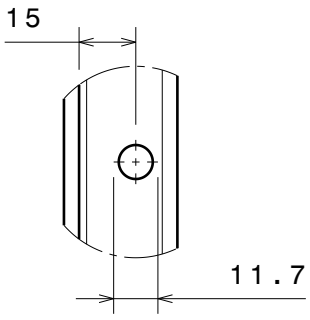
| | | | |
|---|---|---|---|
| DESIGNED BY: Manuel Azevedo<br>DATE: 29/9/2024<br>WEIGHT (kg): 3.53 | | Central Structure | I _<br>H _<br>G _<br>F _ |
| SIZE A4 | | CMAST Propulsion Lab | E _<br>D _ |
| SCALE 1:6 | | NOTES: All dimensions in mm. | SHEET 1/2<br>C _<br>B _<br>A _ |



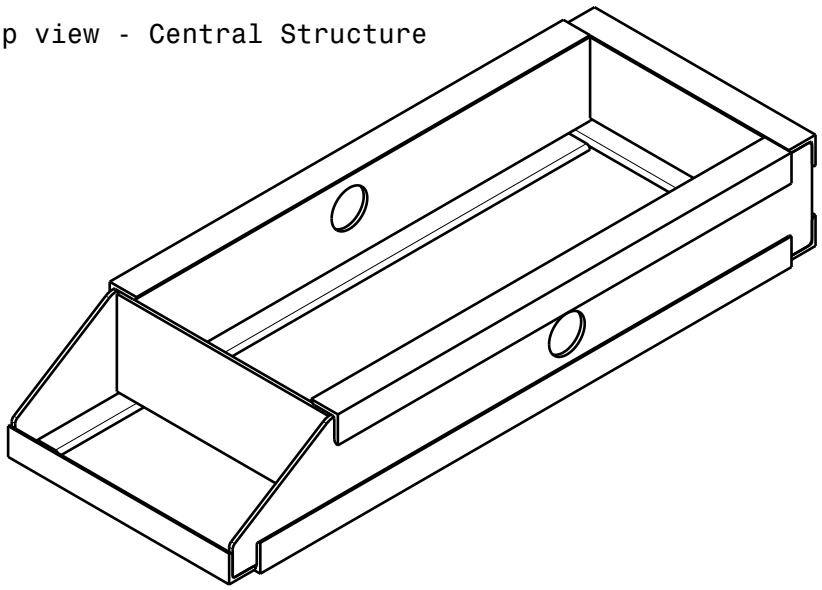

Front view - Central Structure




Top view - Central Structure




Detail A
Scale: 1:2



Isometric view - Central Structure

| | | | |
|---|---|---|---|
| DESIGNED BY: Manuel Azevedo | | # Central Structure | I _ |
| DATE: 29/9/2024 | | | H _ |
| WEIGHT (kg): 3.53 | | | G _ |
| | | | F _ |
| SIZE A4 | | CMAST Propulsion Lab | E _ |
| | | | D _ |
| SCALE 1:6 | | NOTES: All dimensions in mm. | SHEET 1/2 |
| | | | C _ |
| | | | B _ |
| | | | A _ |

# Appendix C

# ETBCDS

## C.1 ETBCDS Arduino code

```
//FINAL MASTER THESIS: Master's degree in Aeronautical Engineering
// TITLE:
// CONTACT:             Manuel António Caetano Azevedo - manuelcaetano40@gmail.com - https://www.linkedin.com/in/im-manuel-azevedo/
// UNIVERSITY:          UBI, Faculdade de Engenharia

//-------------------------------------------- Code structure and information ----------------------------------------------------
// This code was made to control and collect data from an Engine Test Bench (ETB). The ETB was developed as a master's dissertation
// at University of Beira Interior. The ETB will have, in this version, 2 different modes of operation, "Static" and "Manual". The
// static mode is automatic with just the selection of the initial load and minimum rpm needed. On the "manual" mode the test is done
// manually with the user having to select the load.
// The code is structured in levels on loop() and steps on the different modes functions. Every time a task is completed the step or
// level value is incremented until it is needed to return to a previous level/step.


#include "HX711.h"                                                  // Library used to get data from the HX711 amplifier
#include <Wire.h>
#include <LiquidCrystal_I2C.h>                                      // Library used for I2C module and LCD
#include <string.h>                                                 // Library used to better manipulate strings
#include <DigiPotX9Cxxx.h>                                          // Library for easier use of X9c102 potentiometer
#include <Adafruit_INA219.h>                                        // Library for use of the INA219 current sensor module
#define seconds() (millis()/1000.0)

// Load Cell for measurement of engine torque with HX711 module
const int LOADCELL_DOUT_PIN = 10;                                    // Pin connected to DOUT of HX711
const int LOADCELL_SCK_PIN = 11;                                     // Pin connected to SCK of HX711
HX711 scale_engine;                                                        // Start link with HX711
float Calibration = 431.31;                                         // Calibration for load cell
float Distance = 0.40;                                              // Distance from the load cell to the axle
float Mass;                                                         // Mass measured in grams
float Weight;                                                       // Weight measured in Newtons
float Torque;                                                       // Engine torque in N.m

// Load Cell for fuel consumption measurement with HX711 module
const int fuel_cell_DT_pin = 13;                                    // Pin connected to DOUT
const int fuel_cell_SCK_pin = 12;                                   // Pin connected to SCK
HX711 scale_fuel;                                                   // Start link with HX711
float fuel_calibration = 431.31;                                    // Calibration for the load cell
float fuel_mass;                                                    // fuel mass measured in grams


// RPM measurements
const int rpmPin = 2;                                               // Pin 2 connected to hall sensor signal
int rpm;                                                            // RPM value to be displayed
float PrevTime = 0;                                                 // Time of previous magnet detection in microseconds
float Duration = 0;                                                 // Time elapsed between magnet detection in microseconds
double time = 0;                                                    // Used to count 60 seconds from the start of the test
float time_abs = 0;                                                 // Variable used to record the absolute time in the "Collect_Data()"
float real_rpm = 0;                                                                                                    //function

//Power Measurements
float Total_Power = 0;                                              // Total power delivered to the axle
float Bat_Power = 0;                                                // Electric power delivered by the battery to the Esc

// Buttons - these variables are used to avoid that noise has any type of influence on the push buttons
int lastButtonState = LOW;
int buttonState = LOW;
unsigned long lastDebounceTime = 0;
const unsigned long debounceDelay = 200;
unsigned long buttonPressStartTime = 0;                             // Time when button press started
const unsigned long pressDuration = 5000;                           // 5 seconds in milliseconds
bool buttonPressed = false;                                         // To keep track of the button state
bool buttonPressedFor5Seconds = false;                              // To keep track if the button has been pressed for 5 seconds

```

```
64  //Selectors - variables used for selecting the mode of operation, the initial load and the minimum rpm
65  int menuIndex = 0;
66  const int numMenuItems = 3;
67  int initial_current = 0;
68  int rpm_min = 0;
69  int new_rpm_min = 0;
70  int rpm_adjusted = 0;
71  int level = 1;
72  int step = 0;
73  int minInput = 1023;
74  int maxInput = 0;
75  int minOutput = 0;
76  int maxOutput = 4000;
77  int increment = 100;
78
79  //Current Sensor variables
80  Adafruit_INA219 ina219_coil(0x40);
81  Adafruit_INA219 ina219_motor(0x41);
82  float current_load = 0;
83  float current_load2 = 0;
84  float average_current = 0;
85  float sum = 0;
86
87  // Data collection
88  float time_now = 0;                                          // Time at the present reading
89  char type[7];                                                // String to know which type of test it was done
90  int reps = 0;                                                // Variable to know if the header is printed or not
91
92  //Joystick variables
93  const int upPin = 9;                                        // Pin connected to the "up" button
94  const int downPin = 8;                                      // Pin connected to the "down" button
95  const int selectPin = 3;                                    // Pin connected to the "select" button
96  const int setPin = 7;                                       // Pin connected to the "set" button
97  int lastTareButtonState = HIGH; // Assuming active-low configuration
98  const int rstPin = 6;                                        // Pin connected to the "Reset" button
99
100 //Digital potentiometer initialization
101 DigiPot Pot(33,31,29);                                          // Pins:(INC, U/D, CS)
102
103
104 LiquidCrystal_I2C lcd(0x27, 16, 2);
105
106 int mapToIncrements(int value, int inMin, int inMax, int outMin, int outMax, int increment) {
107         int range = (outMax - outMin) / increment;
108         int mappedValue = map(value, inMin, inMax, 0, range);
109         return outMin + (mappedValue * increment);
110 }
111
112 void setup() {
113         lcd.init();                                         // Start I2C link with LCD
114         lcd.backlight();                                    // Turn on LCD backlight
115
116         lcd.setCursor(0,0);
117         lcd.print("Starting checks");
118
119         // Definition of pin modes and activation of internal resistors
120         pinMode(upPin, INPUT_PULLUP);                                   // "Up" button on the joystick
121         pinMode(downPin, INPUT_PULLUP);                                 // "Down" button on the joystick
122         pinMode(selectPin, INPUT_PULLUP);                               // Button used to select and confirm modes and values
123         pinMode(setPin, INPUT_PULLUP);                                  // Button used to tare the scale
124         pinMode(rstPin, INPUT_PULLUP);                                  // Emergency stop button
125         pinMode(rpmPin, INPUT_PULLUP);                                  // pin used for the Hall sensor signal
126
127         attachInterrupt(digitalPinToInterrupt(rpmPin), countRpm, RISING);     // Interrupt used to call function countRPM() everytime there is
128         // signal from the Hall Sensor
129
130         scale_engine.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);        // Start link with HX711
131         scale_engine.tare();                                            // Tare scale to zero
132         scale_engine.set_scale(Calibration);                            // Adjust scale to calibration factor
133
134         scale_fuel.begin(fuel_cell_DT_pin, fuel_cell_SCK_pin);          // Start link with HX711
135         scale_fuel.tare();                                              // Tare fuel scale to 0
136         scale_fuel.set_scale(fuel_calibration);                         // Adjust scale to calibration factor
137
138         ina219_coil.setCalibration_32V_2A();                            //Calibrate the ina219 modules for the shunts used (50A, 75mV)
139         ina219_motor.setCalibration_32V_2A();                           // for this custom calibration some changes had to be made to the library
140         // (see Adafruit_INA219.cpp)
141
142         Serial.begin(9600);
143         delay(1000);
144
```

```
145                 lcd.clear();
146                 lcd.setCursor(0, 0);
147                 lcd.print(" The Dyno Bench ");                                  // Delay routine to stabilize scale.tare
148                 lcd.setCursor(0, 1);
149                 lcd.print("   C-MAST UBI   ");
150                 delay(1500);
151                 updateMenu();
152
153                 Pot.set(0);                                                      // Set potentiometer to 0 in the beginning
154         }
155
156         void loop() {
157                 // In the loop() only the main things are done. These includes selecting the mode of operation, Tare the load cell
158                 // and emergency stoping the ETB. In "level" 1 the joystick is used to walk trough the different modes and when
159                 // the middle button is pressed the mode is selected and "level" is incremented to 2. When this happen the select-
160                 // ed mode function is called.
161
162
163                 if (level == 1) {
164                         // Check the "up" button
165                         if (digitalRead(upPin) == LOW && millis() - lastDebounceTime > debounceDelay) {
166                                 lastDebounceTime = millis();
167                                 menuIndex--;
168                                 if (menuIndex < 0) menuIndex = numMenuItems - 1;  // Wrap around to last item
169                                 updateMenu();
170                         }
171
172                         // Check the "down" button
173                         if (digitalRead(downPin) == LOW && millis() - lastDebounceTime > debounceDelay) {
174                                 lastDebounceTime = millis();
175                                 menuIndex++;
176                                 if (menuIndex >= numMenuItems) menuIndex = 0;  // Wrap around to first item
177                                 updateMenu();
178                         }
179
180                         if (digitalRead(selectPin) == LOW && millis() - lastDebounceTime > debounceDelay) {
181                                 lastDebounceTime = millis();
182                                 // Action for the selected menu option
183                                 selectMenuItem();
184                                 level = 2;
185                         }
186                 }
187                 else if (level == 2) {
188                         if (menuIndex == 0) {
189                                 estatico();
190                         } else if (menuIndex == 1) {
191                                 lcd.clear();
192                                 lcd.setCursor(0,0);
193                                 lcd.print("Still not done");
194                                 delay(3000);
195                                 level = 1;
196                                 updateMenu();
197                         } else if (menuIndex == 2) {
198                                 manual();
199                         }
200                 }
201
202
203                 // In each loop, the tare and emergency buttons are checked. If any of them is pressed,
204                 // according action is taken.
205
206                 int tareButtonState = digitalRead(setPin);  // Read the button on pin 43
207
208                 // Check for button press (high to low transition)
209                 if (lastTareButtonState == HIGH && tareButtonState == LOW) {
210                         scale_engine.tare();  // Tare the load cell
211                         scale_fuel.tare();
212                         lcd.clear();
213                         lcd.setCursor(0, 0);
214                         lcd.print("Load Cells Tared");
215                         delay(1000);  // Short delay to show feedback on the LCD
216                         updateMenu(); // Refresh the menu after taring
217                 }
218
219                 lastTareButtonState = tareButtonState; // Update the last state
220
221                 // Check if Emergency Stop button (pin 41) is pressed
222                 if (digitalRead(41) == LOW) {
223                         delay(50); // Debounce delay
224                         if (digitalRead(41) == LOW) { // Confirm press
225                                 // Stop all processes, reset states, and return to main menu
```

```
                                level = 1;
                                step = 0;
                                Pot.set(0); // Reset potentiometer to 0
                                lcd.clear();
                                lcd.setCursor(0, 0);
                                lcd.print("Emergency stop");
                                delay(2000); // Show message for 2 seconds
                                updateMenu(); // Go back to the main menu
                        }
                }
        }


        void updateMenu() {
                // This function prints to the LCD the Current Menu option. Each time the joystick is used
                // "menuIndex" value changes and the different option appear on the bottom line of the LCD

                lcd.clear();
                lcd.setCursor(0, 0);
                lcd.print("Menu: ");

                switch (menuIndex) {
                        case 0:
                        lcd.setCursor(0, 1);
                        lcd.print("1 - Estatico");
                        break;
                        case 1:
                        lcd.setCursor(0, 1);
                        lcd.print("2 - Transiente");
                        break;
                        case 2:
                        lcd.setCursor(0, 1);
                        lcd.print("3 - Manual");
                        break;
                }
        }

        void selectMenuItem() {
                // Actions for when the "middle" button of the joystick is pressed, selecting the mode of
                // operation

                lcd.clear();
                lcd.setCursor(0, 0);
                lcd.print("Selecionado!");
                lcd.setCursor(0,1);
                delay(1000);
                lcd.clear();
        }

        void estatico()
        {
                // This function corresponds to the static mode of testing. In this mode, the first var-
                // iables, Current and RPM are selected. To select these values, a joystick is used.
                // After each value is selected the value "step" is incremented and the function conti-
                // nues.
                if(step == 0){
                        current();
                        step = 1;
                }
                else if (step == 1)
                {
                        int new_initial_current = initial_current;  // Initialize to avoid jumps
                        int upState, downState, selectState;

                        // Read the joystick directions and button state
                        upState = digitalRead(upPin);
                        downState = digitalRead(downPin);
                        selectState = digitalRead(selectPin);

                        // Adjust initial_current based on up/down movement
                        if (upState == LOW && initial_current < 30) {  // Joystick up pressed
                                new_initial_current++;
                        } else if (downState == LOW && initial_current > 0) {  // Joystick down pressed
                                new_initial_current--;
                        }

                        // Update the display only if initial_current changed
                        if (new_initial_current != initial_current) {
                                initial_current = new_initial_current;
                                current();  // Call to update the display or handle current
                        }
```

```
307
308                     // Debouncing for select button to confirm selection
309                     if (selectState != lastButtonState) {
310                             lastDebounceTime = millis();  // Reset debounce timer
311                     }
312
313                     if ((millis() - lastDebounceTime) > debounceDelay) {
314                             if (selectState != buttonState) {
315                                     buttonState = selectState;
316                                     if (buttonState == LOW) {  // Joystick select button pressed
317                                             step = 2;
318                                             //show "Selecionado!" on LCD
319                                             lcd.clear();
320                                             lcd.setCursor(0,0);
321                                             lcd.print("Selecionado!");
322                                     }
323                             }
324                     }
325                     lastButtonState = selectState;  // Update last button state
326                     delay(250);  // Small delay for processing
327                     // Initialize new_rpm_min at the beginning
328             }
329             else if (step == 2)
330             {
331                     rpm_selecter();
332                     // Check joystick states
333                     int upState = digitalRead(upPin);      // Joystick up pin
334                     int downState = digitalRead(downPin);  // Joystick down pin
335                     int selectState = digitalRead(selectPin); // Joystick select pin
336
337                     // Adjust new_rpm_min based on joystick movement
338                     if (upState == LOW) {  // Joystick up pressed
339                             new_rpm_min += 50;  // Increment by 50
340                     } else if (downState == LOW) {  // Joystick down pressed
341                             if (new_rpm_min >= 50) {   // Prevent going below 0
342                                     new_rpm_min -= 50;  // Decrement by 50
343                             }
344                     }
345
346                     // Update the display only if new_rpm_min changed
347                     if (new_rpm_min != rpm_min) {  // Check if there's a change
348                             rpm_min = new_rpm_min;
349                             rpm_selecter();  // Update the display or handle the new rpm_min
350                     }
351
352                     // Debouncing for select button to confirm selection
353                     if (selectState != lastButtonState) {
354                             lastDebounceTime = millis();  // Reset debounce timer
355                     }
356
357                     if ((millis() - lastDebounceTime) > debounceDelay) {
358                             if (selectState != buttonState) {
359                                     buttonState = selectState;
360                                     if (buttonState == LOW) {  // Joystick select button pressed
361                                             step = 3;
362                                             lcd.clear();
363                                             lcd.setCursor(0, 0);
364                                             lcd.print("Selecionado!");
365                                             delay(1000);  // Delay to show confirmation message
366                                     }
367                             }
368                     }
369                     lastButtonState = selectState;  // Update last button state
370                     delay(250);  // Small delay for processing
371             }
372             else if(step == 3)
373             {
374                     lcd.clear();
375                     lcd.setCursor(0,0);
376                     lcd.print("Teste a comecar!");
377                     lcd.setCursor(0,1);
378                     lcd.print("Set throttle!");
379                     //Serial.println("O teste vai ser começar!");
380                     //Serial.println("Garante que o acelerador está na posição a testar");
381                     step = 4;
382             }
383             else if(step == 4)
384             {
385                     // POWER SUPPLY START-UP
386                     // The start-up of the Power Supply is done through the potentiometer.
387                     // With the current value selected, an average of 10 current values is conducted
```

```
388                     // to ensure that the current on the coil is the selected.
389
390                     sum = 0;   // Reset sum to zero before starting
391
392                     for (int i = 1; i <= 10; i++) {
393
394                             current_load = ina219_coil.getShuntVoltage_mV()*50/75;              // Calculate the current
395
396                             sum += current_load;                                            // Accumulate current values
397                             delay(50);                                                      // Short delay between readings to smooth out noise
398                     }
399
400                     average_current = sum / 10;                                       // Calculate the average current
401
402                     // Adjust potentiometer based on the average current
403                     if (average_current - initial_current < -0.5) {                 // If average current is less than target (below threshold)
404                             Pot.increase(1);                                                // Increment potentiometer
405                     } else if (average_current - initial_current > 0.5) {           // If average current is greater than target (above threshold)
406                             Pot.decrease(1);                                                // Decrement potentiometer
407                     } else {
408                             // when the average is equal to the value selected, "step" is increment and the sequence continues
409                             step = 5;
410                             time = millis();
411                             time_abs = millis();
412                     }
413
414             }
415             else if(step == 5)
416             {
417                     // During one minute the function "Collect_Data()" is called and data is collected
418
419                     if (millis() - time < 60000)
420                     {
421                             strcpy(type, "static");
422
423                             Collect_Data();
424                     }
425                     else
426                     {
427                             step = 6;
428                     }
429             }
430             else if(step == 6)
431             {
432                     // the load is incremented by one step and data continues to be collected
433                     // in order to see the response of the engine to the change of load
434
435                     int load_increment = 1;
436                     Pot.increase(load_increment);
437
438                     step = 7;
439                     Collect_Data();
440                     time = millis();
441             }
442             else if(step == 7)
443             {
444                     // for 2 seconds data is collected which gives time for the engine to stabilize
445                     // on the new load
446
447                     if(millis() - time < 2000){
448                             lcd.clear();
449                             lcd.setCursor(0,0);
450                             lcd.print("A testar limite.");
451                             Collect_Data();
452                     }
453
454                     RPM();                  // the rpm of the engine under the new load is calculated
455                     int rpm_test = rpm*4;
456
457                     Collect_Data();
458
459                     if (rpm_min < rpm_test)   // if the rpm of the engine is greater than the minimum selected
460                     {                       // the test sequence may continue, returning to step 5
461                             step = 5;
462                             time = millis();
463                     }
464                     else                    // if the rpm of the engine is lower than the value selected, the
465                     {                       // the test has to be stopped to avoid stalling/damaging the engine.
466                             level = 1;              // "level" and "step" must return to 1 and 0 respectively and the di
467                             step = 0;               // gital potentiometer must return to 0.
468                             lcd.clear();
```

```
                                lcd.setCursor(0,0);
                                lcd.print("Limite atingido!");
                                lcd.setCursor(0,1);
                                lcd.print("Teste parado!");
                                delay(1000);
                                updateMenu();
                                Pot.set(0);
                                reps = 0;
                                return;
                        }

                }

        }

        void manual()
        {
                // In this function the inputs in the joystick translate directly into current at the coil.
                // Because of that it was deemed necessary to remind the operator to not exceed the limits
                // of the engine in step 1.
                // The function works by collectin data using the "Collect_Data()" function and checking in
                // each loop if the up or down button of the joystick has been pressed. If that is true,
                // then the current is incremented or decremented, respectively. If the middle button is
                // pressed the function is terminated. For that "level" and "step" must be set to 1 and 0 and
                // the digital potentiometer must be set to 0.
                if (step == 0)
                {
                        lcd.clear();
                        lcd.setCursor(4,0);
                        lcd.print("Cuidado!");
                        lcd.setCursor(0,1);
                        lcd.print("Nao ha limite!");

                        delay(2000);
                        step = 1;
                }
                else if (step == 1)
                {
                        strcpy(type,"manual");
                        Collect_Data();

                        int upState = digitalRead(upPin);
                        int downState = digitalRead(downPin);

                        // Increment potentiometer value if the up button is pressed
                        if (upState == LOW) {
                                Pot.increase(1);  // Increment
                                delay(50);  // Small delay for debouncing
                        }

                        // Decrement potentiometer value if the down button is pressed
                        if (downState == LOW) {
                                Pot.decrease(1);  // Decrement
                                delay(50);  // Small delay for debouncing
                        }

                        buttonState = digitalRead(selectPin); // Read the state of the button

                        if (buttonState == LOW) { // Button is pressed (assuming active-low configuration)
                                if (!buttonPressed) { // Button was not previously pressed
                                        buttonPressStartTime = millis(); // Record the time when the button was first pressed
                                        buttonPressed = true; // Update button state to pressed
                                        buttonPressedFor5Seconds = false; // Reset flag for press duration
                                } else {
                                        // Check if the button has been pressed for 5 seconds
                                        if ((millis() - buttonPressStartTime) >= pressDuration) {
                                                buttonPressedFor5Seconds = true; // Set flag indicating button has been pressed for 5 seconds
                                        }
                                }
                        } else { // Button is released
                                if (buttonPressed) { // Button was previously pressed
                                        buttonPressed = false; // Update button state to released
                                        if (buttonPressedFor5Seconds) {
                                                lcd.clear();
                                                lcd.setCursor(0,0);
                                                lcd.print("Voltar ao menu!");
                                                //Serial.println("A voltar ao menu principal");
                                                level = 1;
                                                step = 0;
                                                buttonPressedFor5Seconds = false; // Reset flag
                                                Pot.set(0);
```

```
                                        reps = 0;
                                        delay(1500);
                                        updateMenu();
                                }
                        }
                }
        }

}

void current()
{
        // function to display the value of the current during the selection process

        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Corrente inicial");
        lcd.setCursor(0,1);
        lcd.print(initial_current);
        lcd.print(" A");
}

void rpm_selecter() {
        // function to display the value of the minimum rpm during the selection process

        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("RPM minimas");
        lcd.setCursor(0,1);
        rpm_adjusted = round(rpm_min/50)*50;        // this step is done to round the value of rpm to the nearest 50
        lcd.print(rpm_adjusted);
}

void countRpm() {
        Duration = micros() - PrevTime;                      // Calculates time difference between revs in microsecond
        PrevTime = micros();
}

void RPM()
{
        rpm = 60000000 / Duration;                                  // rpm = (1/ time millis)*1000*1000*60;
        if (micros() - PrevTime  > 2*1000000)                       // Check if motor stopped - unchanged after 2s
        {
                rpm = 0;
        }
}

void Collect_Data()
{
        time_now = seconds() - time_abs/1000.0;                         // time since the start of the first test
        // Load Cell Code
        // Read the weight
        Mass = scale_engine.get_units(3)/4;                             // Average of 3 readings from the test bench load cell
        // it also takes into account the 4:1 pulley system implemented
        Weight = Mass * 1e-3 * 9.80665;                                 // Conversion of mass into weight
        Torque = Weight * Distance;

        fuel_mass = scale_fuel.get_units(3);                            // Average of 3 readings from the fuel load cell

        current_load = ina219_coil.getShuntVoltage_mV()*50/75;           // Read the voltage output from the coil
        // shunt and calculate the current on the circuit

        current_load2 = ina219_motor.getShuntVoltage_mV()*50/75;         // Read the voltage output from the motor shunt
        // and calculate the current on the circuit

        RPM();
        real_rpm = rpm*4;                                              // Calculate the engine shaft RPM

        Total_Power = Torque * real_rpm;

        if (reps == 0)
        {
                // Prints a header if it is the first rep of testing with is column data and unit.
                Serial.println("type,Load current,Duration,RPM,Mass [g],Weight [N],Torque[Nm],Total Power [W],Current Consumed by motor[A],Fuel mass [g]");
                reps = 1;
        }


        // Print all data into the serial monitor. To record this any recording software can be used.
        // Data is printed in order to be recorded as a .csv for easier handling.
        Serial.print(type);
```

```
                Serial.print(",");
                Serial.print(current_load);
                Serial.print(",");
                Serial.print(time_now,4);
                Serial.print(",");
                Serial.print(rpm);
                Serial.print(",");
                Serial.print(Mass);
                Serial.print(",");
                Serial.print(Weight);
                Serial.print(",");
                Serial.print(Torque);
                Serial.print(",");
                Serial.print(Total_Power);
                Serial.print(",");
                Serial.print(current_load2);
                Serial.print(",");
                Serial.println(fuel_mass);

                if (step == 5 || step == 1){
                        // To ensure that all important information is displayed in the LCD during steps 6 and 7 this
                        // this conditional statement is implemented.

                        lcd.clear();
                        lcd.setCursor(0,0);
                        lcd.print("RPM:");
                        lcd.print(rpm);
                        lcd.setCursor(0,1);
                        lcd.print("Torque:");
                        lcd.print(Torque);
                }
        }
```

## C.2 Adafruit INA219 library code changes

```
                void Adafruit_INA219::setCalibration_32V_2A() {
                        // VBUS_MAX = 32V
                        // VSHUNT_MAX = 0.075V (75mV) -> shunt voltage range
                        // RSHUNT = 0.0015 (1.5 mΩ)

                        // 1. Determine max possible current
                        // MaxPossible_I = VSHUNT_MAX / RSHUNT
                        // MaxPossible_I = 50A

                        // 2. Determine max expected current
                        // MaxExpected_I = 50A

                        // 3. Calculate possible range of LSBs (Min = 15-bit, Max = 12-bit)
                        // MinimumLSB = MaxExpected_I / 32767
                        // MinimumLSB = 50 / 32767   0.001526A (1.526 mA per bit)
                        // MaximumLSB = MaxExpected_I / 4096
                        // MaximumLSB = 50 / 4096   0.0122A (12.2 mA per bit)

                        // 4. Choose an LSB between the min and max values
                        // CurrentLSB = 0.002 (2 mA per bit)
                        float currentLSB = 0.002; // 2 mA per bit

                        // 5. Compute the calibration register
                        // Cal = trunc(0.04096 / (CurrentLSB * RSHUNT))
                        // Cal = trunc(0.04096 / (0.002 * 0.0015))   13653
                        uint16_t calibration_value = 13653;

                        ina219_calValue = calibration_value;

                        // 6. Calculate the power LSB
                        // PowerLSB = 20 * CurrentLSB
                        // PowerLSB = 20 * 0.002 = 0.04 (40 mW per bit)
                        ina219_powerMultiplier_mW = 0.04; // Power LSB = 40 mW per bit

                        // Set multipliers to convert raw current/power values
                        ina219_currentDivider_mA = 0.002; // Current LSB = 2 mA per bit

                        // Set Calibration register to 'Cal' calculated above
                        Adafruit_BusIO_Register calibration_reg =
                        Adafruit_BusIO_Register(i2c_dev, INA219_REG_CALIBRATION, 2, MSBFIRST);
                        calibration_reg.write(ina219_calValue, 2);
```

```
42
43                      // Set Config register to take into account the settings above
44                      uint16_t config = INA219_CONFIG_BVOLTAGERANGE_32V |
45                      INA219_CONFIG_GAIN_8_320MV | INA219_CONFIG_BADCRES_12BIT |
46                      INA219_CONFIG_SADCRES_12BIT_1S_532US |
47                      INA219_CONFIG_MODE_SANDBVOLT_CONTINUOUS;
48                      Adafruit_BusIO_Register config_reg =
49                      Adafruit_BusIO_Register(i2c_dev, INA219_REG_CONFIG, 2, MSBFIRST);
50                      _success = config_reg.write(config, 2);
51              }
52
53
```

## C.3 Calibration factor code

```
1               #include "HX711.h"
2
3               #define DT 2  // HX711 Data pin
4               #define SCK 3 // HX711 Clock pin
5
6               HX711 scale;
7               float calibration_factor = 1.0; // Initially 1, will be recalculated
8               bool calibrated = false;
9
10              void setup() {
11                      Serial.begin(9600);  // Set baud rate to 9600
12                      scale.begin(DT, SCK);
13
14                      Serial.println("Remove all weight. Waiting for automatic tare...");
15                      delay(2000);
16                      scale.tare();
17                      Serial.println("Tare completed!");
18                      Serial.println("Place a known weight and enter the real value in the Serial Monitor.");
19              }
20
21              void loop() {
22                      if (scale.is_ready()) {
23                              float reading = scale.get_units(10);  // Average of 10 readings
24
25                              if (calibrated) {
26                                      float weight = reading / calibration_factor;
27                                      Serial.print("Mass: ");
28                                      Serial.print(weight, 3);
29                                      Serial.println(" g");
30                              } else {
31                                      Serial.print("Raw reading: ");
32                                      Serial.println(reading, 3);
33                              }
34                      } else {
35                              Serial.println("HX711 not found.");
36                      }
37
38                      if (Serial.available() > 0) {
39                              String input = Serial.readStringUntil('\n');
40                              input.trim();
41                              float known_weight = input.toFloat();
42
43                              if (known_weight > 0) {
44                                      calibration_factor = scale.get_units(10) / known_weight;
45                                      Serial.print("Calibration factor calculated: ");
46                                      Serial.println(calibration_factor, 5);
47                                      Serial.println("Now the weight will be measured correctly.");
48                                      calibrated = true;
49                              } else {
50                                      Serial.println("Invalid input. Please enter a valid number.");
51                              }
52                      }
53
54                      delay(1000);
55              }
```

## C.4 Non-repeatability error test tables

Table C.1: Differences between each trials for 2783.85 g on the 20 kg load cell.

| Differences [g] - 2783.85 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 16.1522 | 19.0523 | 10.1600 | 4.4196 |
| 2nd try | | 2.9001 | 5.9922 | 11.7326 |
| 3rd try | | | 8.8923 | 14.6327 |
| 4th try | | | | 5.7404 |

Table C.2: Differences between each trials for 4579.48 g on the 20 kg load cell.

| Differences [g] - 4579.48 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 13.0129 | 8.8700 | 5.6676 | 1.6157 |
| 2nd try | | 4.1428 | 18.6804 | 14.6286 |
| 3rd try | | | 14.5376 | 10.4857 |
| 4th try | | | | 4.0519 |

Table C.3: Differences between each trials for 8631.23 g on the 20 kg load cell.

| Differences [g] - 8631.23 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 14.4374 | 32.7355 | 58.3672 | 1.0045 |
| 2nd try | | 47.1729 | 72.8046 | 15.4420 |
| 3rd try | | | 25.6317 | 31.7310 |
| 4th try | | | | 57.3627 |

Table C.4: Differences between each trials for 11 441.71 g on the 20 kg load cell.

| Differences [g] - 11 441.71 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 20.9056 | 18.9480 | 15.3229 | 26.2250 |
| 2nd try | | 1.9576 | 5.5827 | 5.3194 |
| 3rd try | | | 3.6251 | 7.2770 |
| 4th try | | | | 10.9021 |

Table C.5: Differences between each trials for 225.57 g on the 5 kg load cell.

| Differences [g] - 225.57 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 0.17044 | 1.61223 | 0.67855 | 0.47262 |
| 2nd try | | 1.44179 | 0.50811 | 0.64306 |
| 3rd try | | | 0.93368 | 2.08485 |
| 4th try | | | | 1.15117 |

Table C.6: Differences between each trials for 983.44 g on the 5 kg load cell.

| Differences [g] - 983.44 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 0.00682 | 0.72424 | 2.98180 | 0.99566 |
| 2nd try | | 0.73107 | 2.97498 | 0.98883 |
| 3rd try | | | 3.70605 | 1.71990 |
| 4th try | | | | 1.98615 |

Table C.7: Differences between each trials for 2050.06 g on the 5 kg load cell.

| Differences [g] - 2050.06 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 7.47531 | 1.45670 | 5.06459 | 10.18899 |
| 2nd try | | 6.01861 | 2.41071 | 2.71368 |
| 3rd try | | | 3.60789 | 8.73229 |
| 4th try | | | | 5.12440 |

Table C.8: Differences between each trials for 3104.5 g on the 5 kg load cell.

| Differences [g] - 3104.5 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 2.81776 | 0.54310 | 0.77338 | 9.93289 |
| 2nd try | | 2.27467 | 2.04439 | 12.75065 |
| 3rd try | | | 0.23028 | 10.47598 |
| 4th try | | | | 10.70626 |

Table C.9: Differences between each trials for 4110.76 g on the 5 kg load cell.

| Differences [g] - 4110.76 g | | | | |
|---|---|---|---|---|
| | 2nd try | 3rd try | 4th try | 5th try |
| 1st try | 8.04060 | 6.59381 | 7.37575 | 7.48227 |
| 2nd try | | 1.44679 | 0.66484 | 15.52287 |
| 3rd try | | | 0.78194 | 14.07608 |
| 4th try | | | | 14.85803 |

# Appendix D

# ICE test data

## D.1 Honda GX35 specs

Table D.1: Honda GX35 engine technical specs [51].

| Honda GX35 engine | |
|---|---|
| Engine type | 4-stroke single cylinder OHC petrol engine |
| Bore & stroke | 39 x 30 cm |
| Displacement | 35.8 $cm^3$ |
| Compression ratio | 8.0:1 |
| Net Power | 1.0 kW (1.3 HP) / 7000 rpm |
| Max. net torque | 1.6 N m (0.16 kgfm) / 5500 rpm |
| Ignition system | Transistorised |
| Starter | Recoil |
| Fuel tank capacity | 0.63 L |
| Fuel cons. at net power | 0.71 L $h^{-1}$ at 7000 rpm |
| Lubrication | Oil mist |
| Engine oil capacity | 0.1 L |
| Dimensions (L x W x H) | 198 x 234 x 240 cm |
| Dry weight | 3.33 kg |

## D.2 Honda GX35 fuel mass graphs

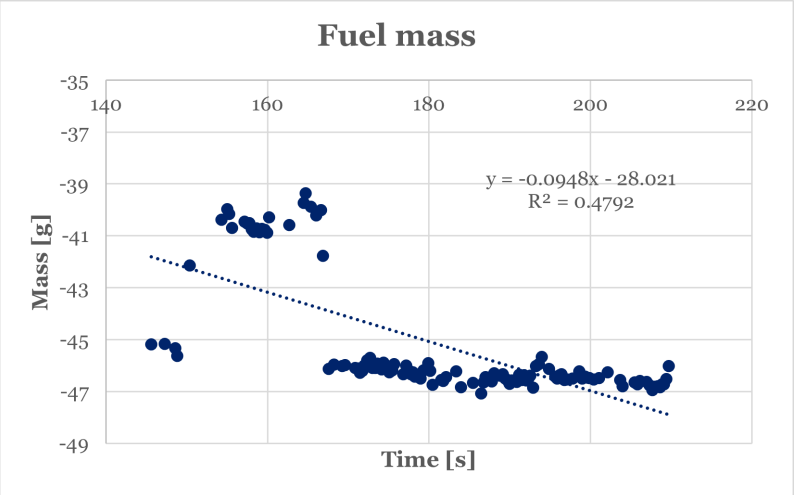



Figure D.1: Fuel mass plot at an engine speed of 4000 rpm.

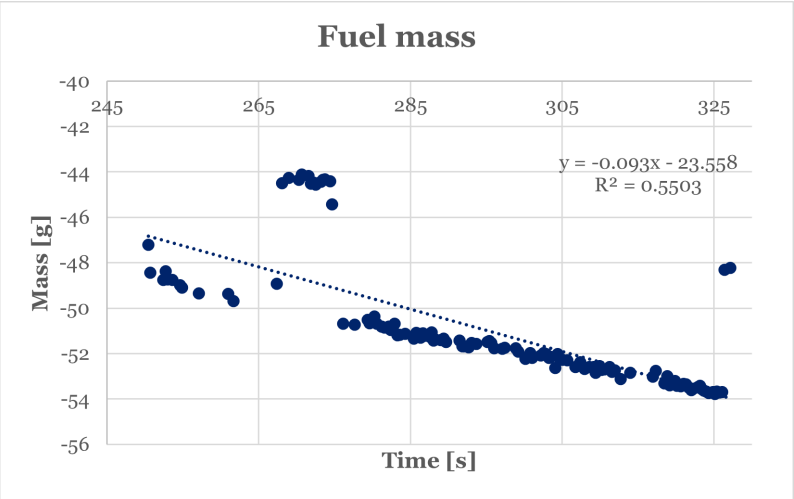



Figure D.2: Fuel mass plot at an engine speed of 5000 rpm.

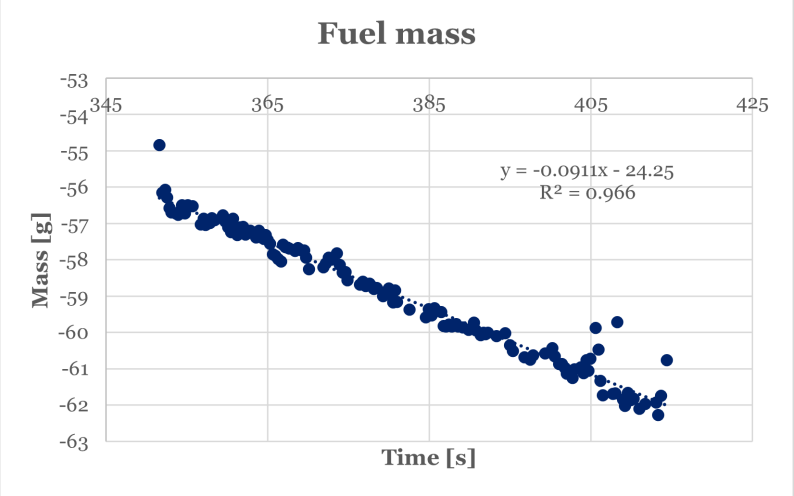



Figure D.3: Fuel mass plot at an engine speed of 6000 rpm.

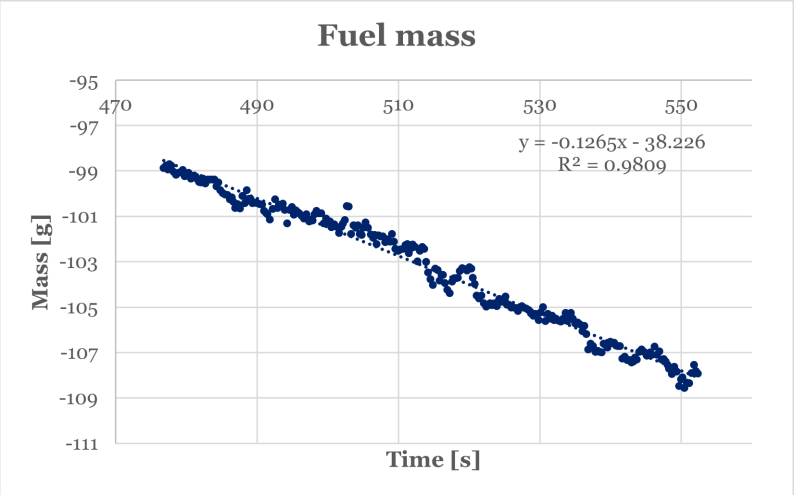



Figure D.4: Fuel mass plot at an engine speed of 7000 rpm.