UNIVERSITAT POLITÈCNIA DE CATALUNYA - ESCOLA SUPERIOR D'ENGINYERIES INDUSTRIAL, AEROESPACIAL I AUDIOVISUAL DE TERRASSA

Escola Superior d'Enginyeries Industrials, Aeroespacial i Audiovisual de Terrassa
UNIVERSITAT POLITÈCNICA DE CATALUNYA

Project of designing and implementing the orientation control system of a wind turbine model

Author:
Carlos Espinós Garcia

Director:
Joseba Quevedo Casin

Bachelor's Degree in Aerospace Technology Engineering

Delivery date: 22/06/2016

Document: **Report**

Contents

1 Introduction **9**
1.1 Aim . . . 9
1.2 Scope . . . 9
1.3 Requirements . . . 11
1.4 Justification of the need . . . 11

2 Development **15**
2.1 State of the art . . . 15
2.1.1 Wind turbine industry . . . 15
2.1.2 Open-source platforms use at university . . . 18
2.2 Possible solutions . . . 19
2.2.1 Plant basic design . . . 19
2.2.2 Modelling methodologies . . . 22
2.2.3 Control strategy . . . 24
2.3 Plant building . . . 26
2.3.1 Dimensions . . . 26
2.3.2 Components selection . . . 27
2.4 Plant modelling . . . 35
2.4.1 Study of theoretical models . . . 35
2.4.2 Experimental modelling . . . 36
2.5 Control algorithm . . . 43
2.5.1 Angular speed . . . 43
2.5.2 Angular position . . . 48
2.5.3 Behaviour under windy conditions . . . 50

3 Results summary **55**
3.1 Budget and economic feasibility . . . 55
3.2 Environmental impact . . . 58
3.3 Security study . . . 59
3.4 Planning of the next step . . . 60
3.5 Conclusions . . . 61

CONTENTS

List of Figures

1.1 The project's Work Breakdown Structure enables to determine and understand the scope easily. 10
1.2 The maximum power is obtained when wind turbine's rotor is aligned with wind direction. Note that, for a deviation of 20°, nearly 20% of the maximum power is lost. 12

2.1 Relationship between power coefficient, C_P, and tip speed ratio, λ. 16
2.2 Layout of different control systems, sensors and actuators in a large wind turbine. Extracted from [1]. 17
2.3 Plant view of the Pitot probes option. By measuring dynamic pressure in two perpendicular axis, wind direction could be obtained. See Equation 2.1. 20
2.4 Block diagram of plant's main components. Note that the power supply is connected to the electric network. It is used to convert AC current into DC current at the motor's operating voltage. 22
2.5 Side view of the "wind turbine" tower. The component located at the bottom is the rotary encoder while the one above is the gear motor. 23
2.6 Block diagram of the cascade control strategy used for the project. Note that SP stands for set point, E for error and MV for manipulated variable. . 24
2.7 Block diagram of the I+P controller used in the angular speed closed loop. . 25
2.8 View of the bottom part of the "fan" tower. The structure used to counteract the torque generated at the bottom by the motor's thrust can be easily seen. It is used to avoid the tower to rotate back. 29
2.9 Latest version of Arduino boards. Note that the Genuino is the name given to the product outside of the United States. 30
2.10 It is interesting to stress that these two motors present different characteristics. The fan's one can rotate very fast while the wind turbine's motor can deliver a very high torque. 31
2.11 The motors driver is used to deliver the amount of power decided by the Arduino to the motors. 32
2.12 The rotary encoder used sends 1024 pulses per revolution. This precision can be increased four times by having a look at the rising and falling of the two channels as they are equally spaced. 32
2.13 View of the upper part of the "fan" tower. The built-in motor and the coupler can be observed. 33
2.14 Back view of the "wind turbine" tower. The motion transmission can be easily appreciated. See the 4 pinion gears: one attached to the motor shaft, two of them linked to the auxiliary shaft and the last one assembled to the rotary encoder. 34
2.15 Electric circuit driving a motor. 35

LIST OF FIGURES

2.16 Motor's response to a pwm input of 128 bit which correspond to 50% of duty cycle. Note that angular speed at steady state is quite high, more than one revolution per second. The red curve corresponds to a first order system with a time constant of 0,05 s. 38
2.17 Linearity study of the motor's response to a PWM input. 39
2.18 Motor's response to a PWM step input with 16% duty cycle. 40
2.19 Motor's time response for different PWM step inputs. The red curve still represents a first order system with a time constant of 0,20 s. 41
2.20 Comparison of different filters for a PWM step input of 40. 42
2.21 Procedure to determine the transfer function of a 1st order system with and I+P controller. 44
2.22 Controlled system response to $SP = 1024\ ticks/s = 90^o/s$. 46
2.23 Controlled system response to $SP = 2048\ ticks/s = 180^o/s$. 47
2.24 Controlled system response to $SP = 3072\ ticks/s = 270^o/s$. 47
2.25 Controlled system response to $SP = 4096\ ticks/s = 360^o/s$. 48
2.26 System's response to a set point of 90º (1024 ticks). Proportional gain is 4,7. 49
2.27 System's response to a set point of 90º (1024 ticks). Proportional gain is 4,8. 50
2.28 System's response to a position input of 256 ticks (22,5 º). Note that the static error corresponds to 5º. 51
2.29 System's response to a position input of 512 ticks (45 º). Note that the static error is 1º. 51
2.30 System's response to a position input of 768 ticks (67,5 º). Note that the static error corresponds to 0,1º. 52
2.31 System's response to a position input of 1024 ticks (90 º). Note that the static error corresponds to 1º. 52
2.32 System's behaviour under windy conditions. 53

3.1 Gantt chart for the possible plant's improvements. 61

List of Tables

2.1 Ziegler-Nichols tuning table based on system's critical response. 49

3.1 Project's budget. If further detail is desired, it is suggested to consult the attached document Budget. 56
3.2 Feasibility study intermediate results for years 0 to 2. 56
3.3 Feasibility study intermediate results for years 3 to 5. 57

LIST OF TABLES

Chapter 1

Introduction

The purpose of this chapter is to determine as precisely as possible the whole project. First, its aim will be explained in Section 1.1. Then, the different tasks that compose all the work required are ordered in Section 1.2. After that, in Section 1.3, all the specifications that the model should fulfil are listed. Finally, Section 1.4 explains the interest of this project.

1.1 Aim

The goal of the project is to build a model of a wind turbine and its orientation control system. The control system should be able to align the rotor's axis with the wind direction. In this way, the plant that will be built consists of a fan simulating the wind and a wind turbine model that will be orientated thanks to a motor, a micro-controller and some sensors.

1.2 Scope

1. Designing and implementing the orientation control system of a wind turbine model.
   1.1. Plant building.
      1.1.1. Plant design. **Deliverable documents:** REPORT, BUDGET AND DRAWINGS.
         - Definition of the specifications of the plant.
         - Selection of the components.

      1.1.2. Components testing. **Deliverable documents:** TECHNICAL SHEETS.
         - Testing of the electric power system.
         - Testing of the motors.
         - Testing of the pressure sensors.
         - Testing of the orientation sensor (encoder).

      1.1.3. Assembly. **Deliverable documents:** REPORT.

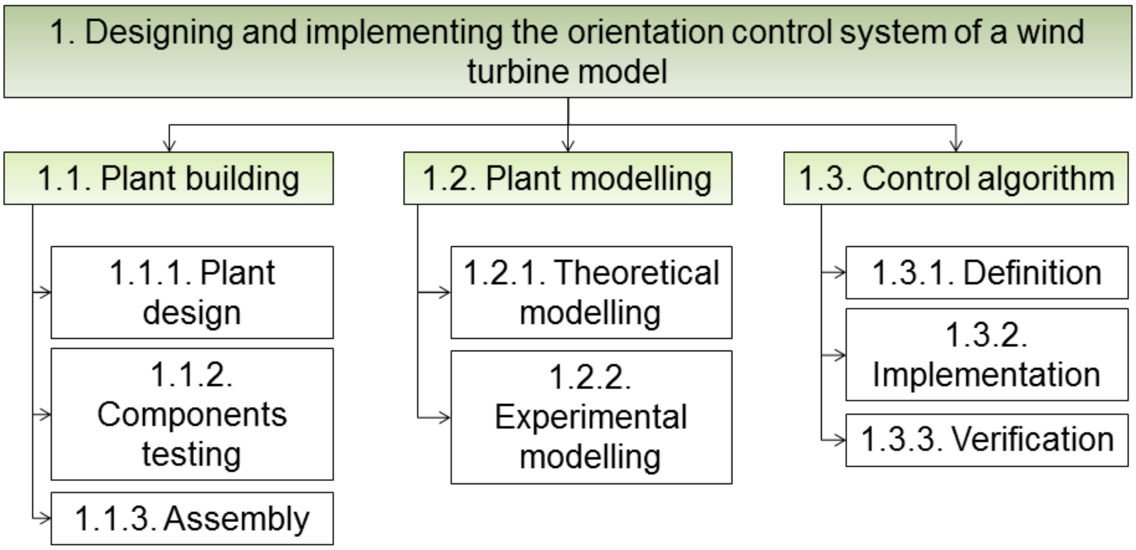


Figure 1.1: The project's Work Breakdown Structure enables to determine and understand the scope easily.

- Electrical connections.
- Mechanical unions.

1.2. Plant modelling.

1.2.1. Study of theoretical models. **Deliverable documents:** REPORT.

- Differential equations of the components.
- Linearisation of the equations.
- Block diagram.

1.2.2. Experimental modelling. **Deliverable documents:** REPORT.

- Study of similar plants.
- Experimental tests: impulse, step and ramp inputs.
- Analysis of the system's response.

1.3. Control algorithm.

1.3.1. Definition. **Deliverable documents:** REPORT AND TECHNICAL SHEETS.

- Study of different strategies.
- Definition of the stability and time requirements.
- Selection of the control strategy.

1.3.2. Implementation. **Deliverable documents:** REPORT AND TECHNICAL SHEETS.

- Determination of the control parameters.
- Programming the control algorithm on the Arduino.

1.3.3. Verification. **Deliverable documents:** REPORT AND TECHNICAL SHEETS.

- Experimental tests: impulse, step and ramp inputs.
- Analysis of the system's response to improve the control algorithm.

1.3 Requirements

- The total economic cost should be less than 300 euros.
- The control system should need less than 1 minute to stabilise.
- The greatest change of wind direction with respect to its initial value should be, at maximum, 90º.

1.4 Justification of the need

Since the 21st century's beginning, human society is starting to realise the importance of climate change. Scientists expect that the current pace of CO2 production will lead to an increase of Earth's temperature so high that could make human life harder due to an increase of famines and catastrophes [2].

For this reason, meetings between governments of the entire world to talk about solutions to this problem are becoming usual. Conference of the Parties (COP) is an example of these meetings. The size of this event, whose repercussion is to the entire world, shows how big the problem is. As the Paris agreement made during the last COP session sates [3], one of the tools that will help us reduce the climate change is wind energy. This way of generating power does not emit carbon dioxide and other harmful gases to the atmosphere so it can substitute actual methods of energy production that do contaminate.

To integrate more deeply this and other renewable sources of energy, it is expected that the electric network will evolve to a more interconnected structure, the so called "Smart Grid". By connecting different regions of a country, the difficulties to match renewable energy production with its demand (due to their unpredictability) can be reduced [4]. This structure will enable electricity consumers to also produce energy [4]. This is why, wind turbines are starting to be manufactured in a more little shape so they can be used by everyone in his own house. To obtain most of the energy from this smaller wind turbine, an orientation control system can be useful due to the unpredictable directions that can take the wind. The statistical studies carried out in big wind farms are no longer a possibility as they are very costly for a particular.

On the other hand, during the late 40 years, control theory has shown how powerful and beneficial it can be for, nearly, every industry. The wind turbine sector is not an exception where control theory has been integrated to improve the productivity. A wide variety of process are controlled in a wind turbine nowadays. Like for example [5]: blade pitch to obtain the desired power, blade pitch to follow the wanted evolution of angular speed when starting, generator torque to regulate rotational speed and yaw attitude to keep the rotor pointing in the wind direction.

The need of a wind direction tracking system can easily be understood with Equation 1.1. The power output from the wind turbine is proportional to the cube of wind speed in rotor shaft direction. This means that the power generated depends with the cube of cosine of the yaw attitude (angle between rotor axis and wind direction). Figure 1.2 shows this relationship. It can be seen that an error until 10º in yaw attitude can be acceptable as no much power is lost. However, the loss of power increases rapidly with this angle.

$$P = \frac{1}{2}\rho(v\cos(\theta))^3 C_P S \tag{1.1}$$

Note that ρ stands for density, v for airspeed, θ for yaw attitude, C_P for power coefficient and S for frontal area.

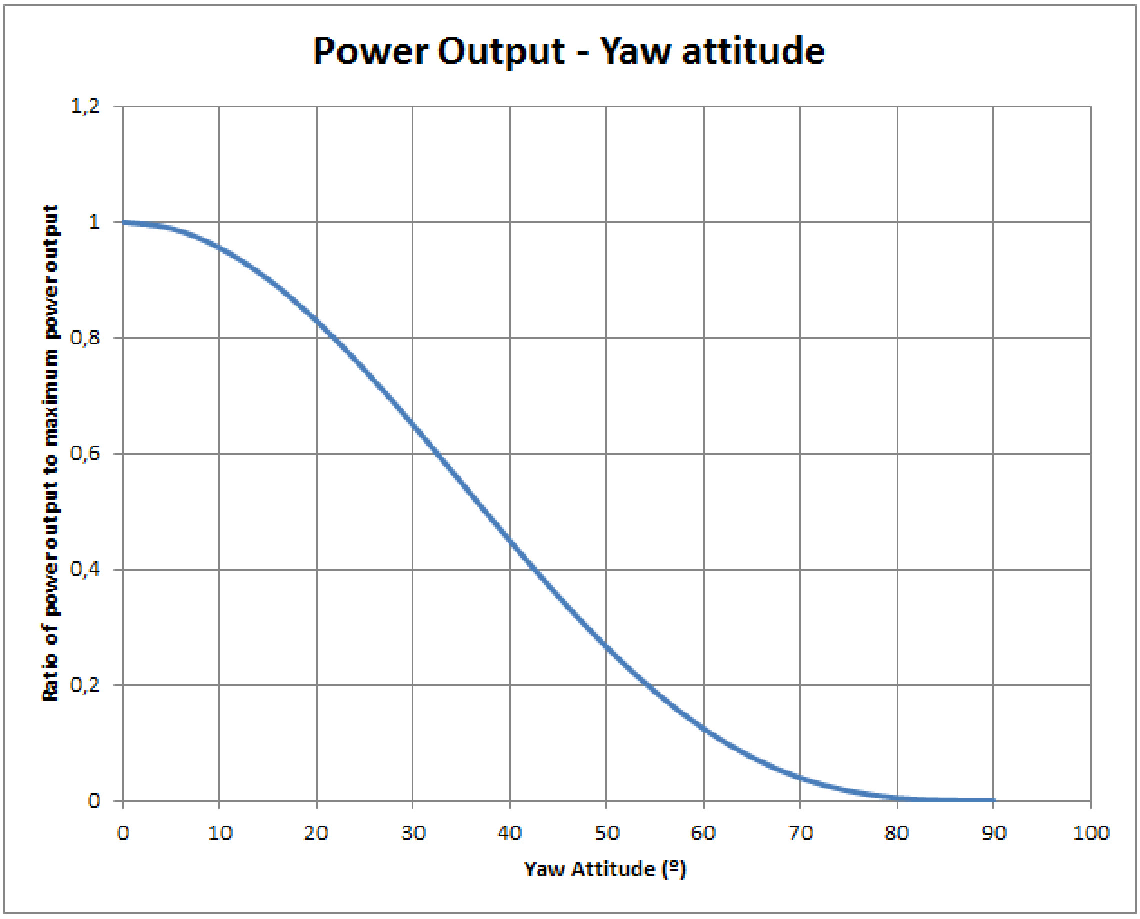


Figure 1.2: The maximum power is obtained when wind turbine's rotor is aligned with wind direction. Note that, for a deviation of 20º, nearly 20% of the maximum power is lost.

From an educational point of view, this project can be appealing as it aims to apply control theory in a real case. Then, students could be able to understand the importance of what they are studying by seeing how it is used in solving a problem they can observe. Moreover, the model should help them to learn better the basic concepts of control theory such as modelling a plant or tuning a controller. In fact, they should be able to change anything of the plant (from the micro-controller code to any electronic or mechanic component). It has been decided to use an Arduino micro-controller due to its ease of use. Additionally, it is becoming very popular in new control laboratories [6], so some experience or knowledge can be exchanged with other universities.

To conclude, it can be said that this thesis is meant to cover two necessities: an environmental one and an educational one. From the environmental point of view, it is tried to highlight the need of wind turbines as they do not emit harmful gases to the atmosphere when energy is produced. On the other hand, from the educational point of view, this project should offer an up to date, and easy to modify, model that can be used as support material to teach any course covering the basics of control theory.

Chapter 2

Development

After understanding the need covered by the project, it has to be developed. First, in Section 2.1, the State of the Art of wind turbine industry, putting special attention on control systems, is revised as well as experiences teaching control theory with open-source hardware like Arduino. Next, in Section 2.2, the main alternatives to achieve the project's goals are studied and the final decision is justified. Then, in Section 2.3, the detailed design of the plant and how it has been assembled are explained. After that, in Section 2.4, the procedure followed to obtain a model for the plant is described. Finally, the method used to tune the orientation controller is shown in Section 2.5.

2.1 State of the art

The state of the art of this thesis is divided into two categories covering the following topics.

- Wind turbine industry: historical evolution, current use of control system and expected changes in wind turbines (specially focusing on its control systems).
- Open source platforms usage at universities: how professors have taken advantage of these devices to teach their students the basics of control theory?

2.1.1 Wind turbine industry

Wind turbines have been used by humans for a long time. Its first purpose was not to produce electrical energy. They were used mainly for agricultural purposes to grind grain and pump water. When the knowledge of electromagnetism increased, the first wind turbines generating electricity appeared. However, no special interest was put on this topic as other means of energy production were available and simpler. Then, the rise of oil price during the 1970's made wind turbines appear as an appealing method to produce electricity as its costs do not depend directly on oil price evolution [5].

With this increase of interest, all the possible wind turbines solutions were tested like the giromill, windmill, vertical axis and horizontal axis. After all of this years, it seems that the best option for energy production is the horizontal axis wind turbine as it is the most common solution. [7] explains that the vertical axis presents some problems of fatigue and vibration that are difficult to solve. Moreover [8] suggests that two and three blades horizontal wind turbines offer the best ability to extract power from air. They can operate in a wide range of wind speeds and extract an acceptable amount of power.

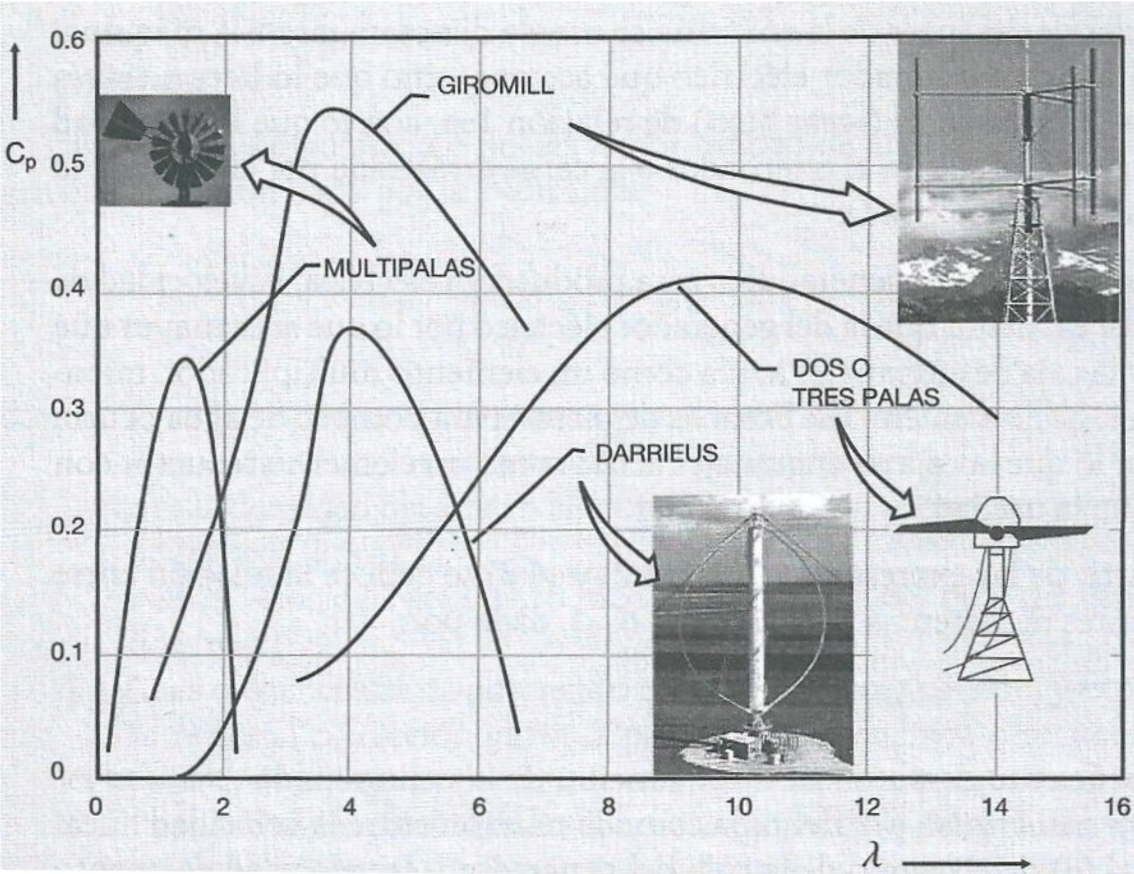


Figure 2.1: Relationship between power coefficient, C_P, and tip speed ratio, λ.

As it can be seen in Figure 2.1, the type of wind turbine that can generate energy in the most diverse situations are two and three blades horizontal axis. Note that the tip speed ratio, $\lambda = \frac{\Omega R}{v_\infty}$, relates the angular speed, Ω, blade length, R with the wind free stream speed, v_∞. Even though, they do not deliver the maximum power, they are able to generate quite a lot of energy in many situations. This is why they have become the most widespread kind of wind turbine nowadays.

The challenges of the wind turbine industry that will drive its evolution can be classified into 4 categories [9]:

- improve prediction of wind conditions.
- enhance wind turbines' technology.
- integrate at a higher level the wind energy to the electric grid.
- develop the offshore wind turbines.

From the technological point of view, it is expected to gain more knowledge into aerodynamics rotor behaviour and structural loads determination as, usually, the materials' strength is the limitation to extract more power from the aerodynamic forces. With the increasing size of wind turbines, their materials have become exposed to higher loads [10].

For large wind turbines, it is also expected to develop its control systems as they are essential to obtain a good performance and enlarge the wind turbine's life cycle [9]. By measuring generator's rotational speed, tower's longitudinal accelerations and blade loads, the wind turbine control system acts on generator's torque, collective and individual pitch to reduce as much as possible the fatigue and stress of wind turbine's materials [1]. Drive train and tower damping systems are examples of such control systems. The other use of control systems is to obtain the maximum power from the wind by tracking its direction and ensuring that generator's frequency matches network's frequency.

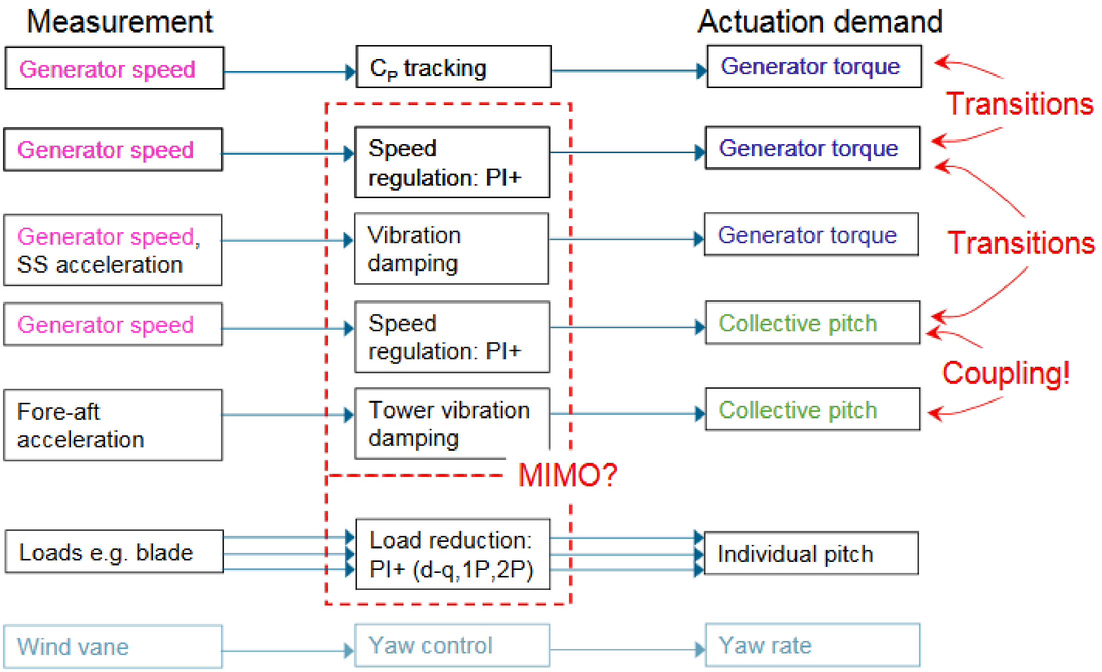


Figure 2.2: Layout of different control systems, sensors and actuators in a large wind turbine. Extracted from [1].

In the future, it is expected that control systems of large wind turbines will include the following improvements. However, yaw tracking control is not meant to change as it offers very good performance nowadays [1].

- *Laser-Doppler anemometers:* These sensors can predict the wind field in the short term. In this way, blade pitch can be set to the less demanding position from the structural point of view.
- *Local actuators:* that only act on a part of the blade, like flaps in an airplane will be added to the nowadays global actuators (that act on the whole blade).
- *Rotor state control:* by changing its natural frequencies, vibrations on the structure could be avoided.
- *Wind turbine farm control:* rotor wake from different wind turbines can interact, so obtaining maximum power and minimum fatigue for the whole farm requires to account for all its wind turbines.

When it comes to smaller wind turbines, control systems are not so implemented as they would make the product too expensive for its aimed market. As they are smaller, the loads are not so severe. This makes it not indispensable to include a control system damping the structures' vibrations and modifying blade attitude to reduce the loads. From performance point of view, wind direction tracking is usually made thanks to a vertical tail acting like a wind vane. This passive control has proven quite good. However, it presents some drawbacks. For example, it cannot decouple tail small oscillations with wind direction from wind turbine yaw attitude. This is why active control of yaw attitude is being studied even for small and medium wind turbines [11].

2.1.2 Open-source platforms use at university

Many references ([6], [12]) coincide stating that until 10 years ago, laboratory platforms were very closed and difficult to interconnect. This made it complicated to share knowledge between different universities as they platforms could be using a diverse range of programming languages. Moreover, these platforms were usually difficult to modify impeding any possible improvement. However, with open-source platforms, like Arduino (borne in 2005), this situation has changed completely. Now, it is possible to exchange knowledge with different working groups as most of these open-source platforms use a C/C++ based language [12]. Moreover, as they are open-source, they are easy to modify. All of this leads to a circumstance where building laboratory work stations is easier.

Reference [12] has used Arduino to control through Bluetooth a robotic arm that can move over the floor. They choosed Arduino as it has an extension making it easier to connect to electric motors and receive Bluetooth orders. Moreover, a Matlab library has been developed for Arduino. In this way, the block diagrams constructed through Simulink can be used to program the Arduino without its own Integrated Development Environment (IDE). This makes it even easier to program and more appealing to beginners.

Reference [6] has taken profit from Arduino to build 4 different laboratory experiments.

- Temperature control of a 3D printer hot end using a PID.
- Automation of a Cartesian robot. It can move on a plane surface of $4\ m^2$.
- Humanoid robot programming.
- Follower robot programming.

Their experience, backed on a student survey, suggests that Arduino is an easy to use platform that motivates students and helps them to enhance their knowledges on programming and electronic hardware. Moreover, Arduino enables to reuse old laboratory material thanks to its versatility.

Reference [13] reflects that teaching students through projects increase their motivation and interest on the subject. It is often perceived as a reward among students to develop a project where they can put into practice the knowledge they have gained through their studies. It is also noted that letting students set the project's scope, makes it more appealing to them. Moreover, it is remarked that adding Arduino to the project has given more options to the students when setting project's scope. However, in this context, Arduino was only used for communication purposes.

Reference [14] takes advantage of Raspberry Pi characteristics to implement a remote virtual laboratory. It is an interesting experience as students do not need for an extra computer. The Raspberry Pi can perform all the required tasks on his own. If it is equipped with Raspbian, Python, SciPy, Matplotlib and NumPy, the data can be analysed to model and control a system. This is the main advantage it offers with respect to Arduino. However, it is interesting to remark that Arduino seems simpler, thus easier, in a first approach.

2.2 Possible solutions

After gaining a broad view of the wind turbine control systems and the use of open-source platforms at university, it is time to consider the different possibilities that could be followed to achieve this project's goals. The Section is structured as follows.

- Section 2.2.1: the different alternatives considered when designing the wind turbine model are explained and the chosen solution is justified.
- Section 2.2.2: the diverse options to obtain a model for the plant are described as well as which ones where chosen to be used in the real wind turbine model.
- Section 2.2.3: the main strategies to control the wind turbine orientation are shown and the final decision of which one to use is described.

2.2.1 Plant basic design

The main issues to address when designing the plant at a basic level were the following.

- Should the wind be produced by a fan or atmospheric wind could be used?
- How should the wind direction be measured?
- How are rotor's blades going to be represented?
- How rotor's axis orientation should be measured?
- What micro-controller is going to be used?
- What materials will be used for the structures?
- How all these components are going to be assembled?

Wind production

First, it was thought that atmospheric wind would be sufficient to move rotor's blades. However, this would make the model be very dependant on weather conditions. So, it would not be able to be used whenever the user desired to. This is why, it was decided to include a "fan" in the model. In this manner, the user would only need to feed the fan's motor to produce the wind that will move the rotor's blades. Moreover, the extra components that are necessary to accomplish this task (rotor blades, an electric motor, a shaft, a coupler and the structure) do not increase model's cost significantly.

Wind direction measuring

This was a debated issue. Two options were considered and it was difficult to choose one of them. It was thought that Pitot probes could be used to measure airspeed in two perpendicular directions. Then, by comparing these two values (through the tangent relationship, see Equation 2.1), wind direction could be obtained. On the other hand, a wind vane was considered. This is a simpler option, thus easier to implement.

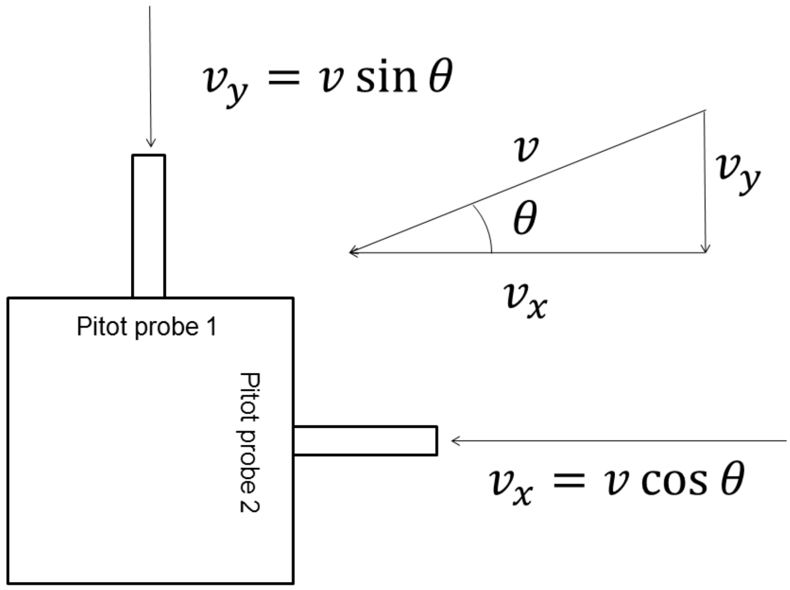


Figure 2.3: Plant view of the Pitot probes option. By measuring dynamic pressure in two perpendicular axis, wind direction could be obtained. See Equation 2.1.

$$\theta = \arctan\left(\frac{v_y}{v_x}\right) \tag{2.1}$$

After considering advantages and drawbacks of both options, it was decided to use the wind vane with an encoder to measure angular position. A threaded shaft and a coupler will be used to attach the wind vane to the encoder. The main reason for discarding the Pitot probes is that they add a lot of complexity to the system. With respect to the wind vane, they require more computational effort from the micro-controller as they are two sensors instead of one. Moreover, the Pitot probes option required to compute a trigonometrical operation which is always difficult to implement in digital controllers.

Rotor blades selection

Different alternatives were possible to simulate the blades of a real wind turbine. They could be built from wood obtaining a plane airfoil with some incidence angle. Another option was to manufacture them using 3D printing. This method enabled a more realistic representation, as a cambered airfoil could be obtained. However, the chosen option was to buy multi-rotor propellers.

This seemed the best option as it was the simplest and cheapest, from the economical and time saving points of view. Certainly, it is less expensive than 3D printing and it offers similar results in terms of fidelity to actual rotor blades. Maybe, wood blades would be cheaper but they would require a lot more time to build and assemble.

Orientation measuring

To determine the angular position of the wind turbine's rotor, an encoder or a taco-dynamo could be used. The encoder outputs directly the orientation whereas the taco-dynamo measures the angular speed. If the taco-dynamo was used, it should be necessary to integrate angular speed to obtain angular position. Even though, this operation can be performed with a micro-controller, it seemed to be less precise than measuring directly the angular position.

However, it is necessary to measure angular speed as it is desired to model it before the angular position. The tacodynamo is the best option without doubts for this task. Nevertheless, using two electronic components would add complexity and cost to the plant. In this way, the assembly of all components would have been more difficult. This increase of cost and difficulty seem to overcome the increase in accuracy when using a tacodynamo instead of a rotary encoder to measure angular speed.

Micro-controller

As it can be derived from the State of the art (Section 2.1), the main options for controlling the model were an Arduino and a Raspberry Pi. Both of them have shown to be very good to implement laboratory work stations and projects at university. Raspberry Pi seems a more complete option than Arduino as an operative system can be installed on it. However, this increase of versatility means also an increase of complexity. As project's requirements can be met with Arduino, it was decided to use it instead of Raspberry to keep simplicity (thus, ease of implementation) in the project.

Materials

This decision was taken rapidly as no other options seemed possible. It was decided to use wood to build the main structure of the model. Then, threaded shafts and nuts were used to attach the different components to the structure.

Assembly

The model's functionalities can be divided into three categories: wind generation, mesuring wind direction and wind turbine's orientation control system. One tower will be used for each functionality. Next, it is explained how each tower's components are attached between them and to the main tower. See Figure 2.4 to understand the relationships between plant's different components.

The "fan" tower is composed by a motor and a propeller. The propeller is attached to a 6mm diameter plain shaft which is joined to the motor thanks to a coupler. Then, the motor (attached by pressure to the tower) is connected to a motor driver that can control the amount of power transmitted to the motor depending on a signal received from the micro-controller. So, the motor driver is connected to the power supply, the micro-controller and the motors.

The "wind direction" tower's components are: a rotary encoder and a wind vane. The wind vane is made from paper board. Then, with a pair of nuts it is attached to a threaded bar that is also connected to the rotary encoder. Then, the rotary encoder is joint to the tower thanks to a wood plate, four threaded shafts and some nuts.

The "wind turbine" tower is made from a small cube representing wind turbine's hub, a motor that can orientate it and a rotary encoder measuring the rotor's angular direction. The encoder and the motor are attached to the main structure in the same way than the encoder in the "wind direction" tower (through a wood plate, four threaded bars and some nuts). Then the cube is joined to the motor's axis thanks to a hub. The angular movement of the cube is transferred to encoder axis by means of pinion gears and an auxiliary shaft connecting motor axis with the encoder's one.

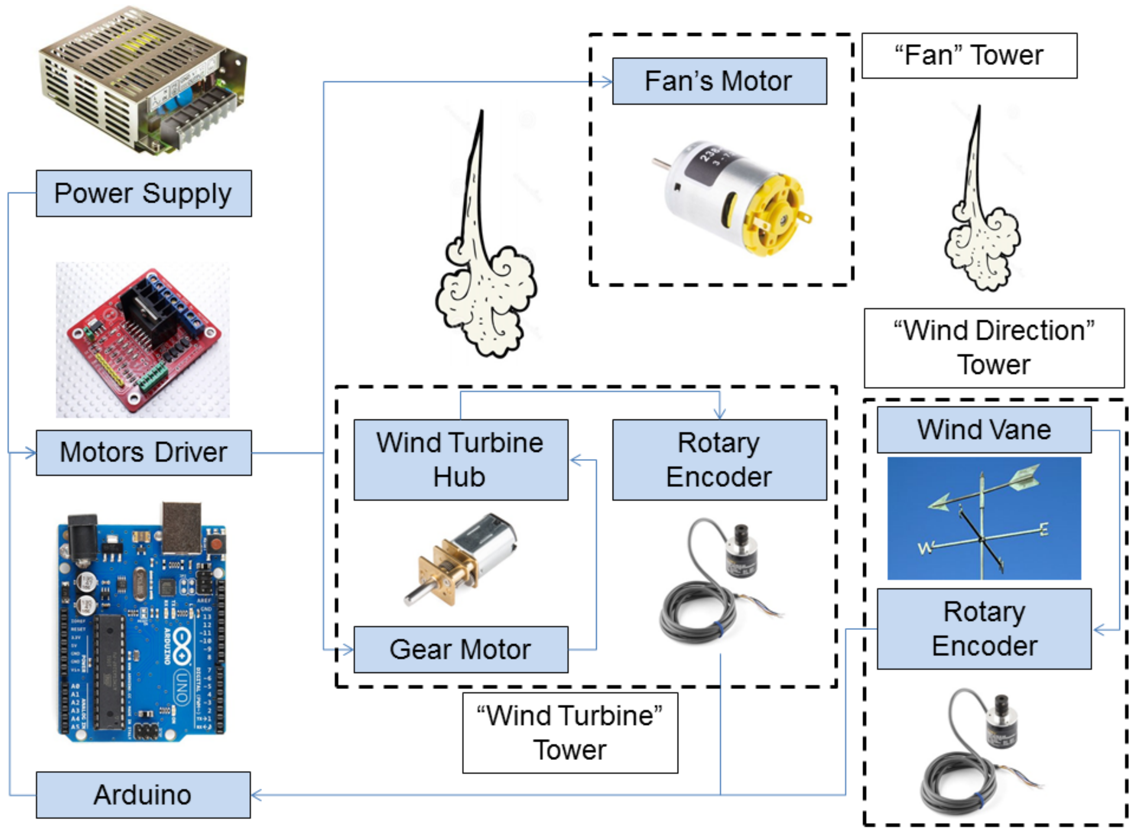


Figure 2.4: Block diagram of plant's main components. Note that the power supply is connected to the electric network. It is used to convert AC current into DC current at the motor's operating voltage.

2.2.2 Modelling methodologies

From control theory, it is clear that modelling the process that is wanted to be controlled is necessary to tune appropriately the controller's parameters. Two options appeared as a possibility to carry out this task: a theoretical modelling or an experimental modelling.

Theoretical modelling

A model could be obtained from the differential equation describing each component behaviour. This was considered as a very large task as it implied studying the physical

Figure 2.5: Side view of the "wind turbine" tower. The component located at the bottom is the rotary encoder while the one above is the gear motor.

process of each component. Moreover, it did not ensure precise results. For example, the aerodynamics theory for wind turbines can be applied to this plant, but, in order to obtain accurate results, it is necessary to execute some Computational Fluid Dynamics (CFD) simulations. As this is a so large topic that a single thesis could be dedicated to it, it was decided to not deepen the theoretical modelling. It was decided to simply review the literature in order to have an idea of what to expect from component's behaviour.

Experimental modelling

By applying a step input to the plant and studying its time response, a model can be obtained. It is also possible to generate a model for the plant by studying its frequency response. However, this did not seem the best option as the plant is not expected to work in situations where the input is oscillating. As the signal that can be used to drive the motor is related to its velocity, it was decided to obtain the transfer function between the pwm signal (emitted from Arduino to the motor driver) and the motor's angular speed.

2.2.3 Control strategy

A cascade control algorithm will be used for the system (see Figure 2.6). A closed loop will check that angular position is correct and, then, emit a velocity set point to the second loop in charge of controlling angular speed. In the velocity loop, the pwm value for the motor driver is decided from the error between set point speed and measured speed.

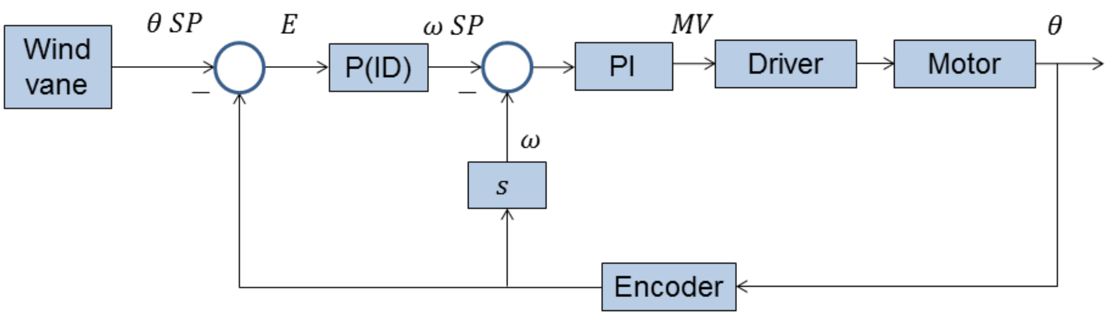


Figure 2.6: Block diagram of the cascade control strategy used for the project. Note that SP stands for set point, E for error and MV for manipulated variable.

This option has been chosen as the best one because with the velocity loop, system can respond faster to perturbations than in a single loop where position error would be directly converted into a Pulse Width Modulation (PWM) order to motor driver. Moreover, by dividing the control system, it is easier to implement and to locate problems when using it.

For the angular speed control of a usual motor, a I+P controller seems sufficient. This is mainly because, if the poles placement technique is applied, as a first order system, two parameters (K_P and K_I) are enough to match the system's response with the desired one. If a first order system it is controlled through an I+P structure, the resulting system is of second order (see Figure 2.21). See that the two coefficients of a critically damped second order system, $2\omega_n$ and ω_n^2, can be matched with the two controller's parameters, K_P and K_I.

Additionally, it is more difficult to saturate an I+P than a PI because the proportional term is only referred to the measured output value. In this manner, if the set point changes suddenly (like in a step manner), the proportional term will not do so, as it only depends on the output (not on the error). If necessary, a derivative term could be added in order to gain speed in system's response. In this case, the derivative term would be parallel to the proportional term to avoid saturating motor's driver.

Moreover, the I+P form allows to apply one of the simplest analytical techniques to tune a controller: the poles placement methodology. Using such a simple technique decreases the probabilities of making errors when applying it. Thus, a more robust implementation is obtained. In a traditional PI structure, poles placement can lead to undesired results as a zero could cancel the action of one pole. However, the I+P structure does not let any zero appear in the transfer function's numerator (see Figure 2.21).

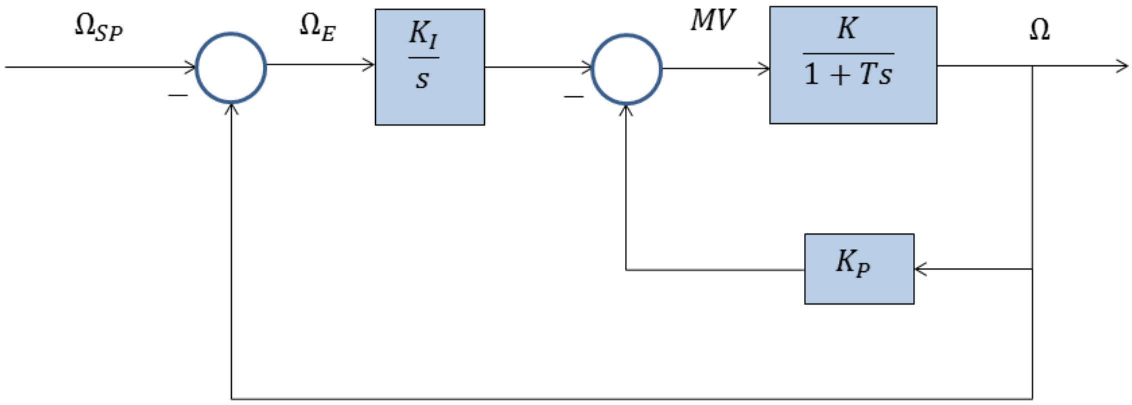


Figure 2.7: Block diagram of the I+P controller used in the angular speed closed loop.

To tune the angular position control system, Ziegler-Nichols rules will be used. This is a common practice in the industry and, usually, a fast way to obtain good results. It was decided to left analytical tuning for a second step as it can be difficult to obtain satisfactory results. It has to be stressed that the experimental model obtained, on which the analytical tuning would be based, is only approximated. Then, controller's parameters when using the poles placement method could be not accurate enough due to the experimental modelling precision.

If the position controller seems to work fine when it is only proportional, the integral and derivative terms will not be included. However, if experience shows that more accuracy, speed or stability are needed, the I and D phases could be added. Then, it is likely that the controller's structure will be that of an I+P or I+PD for the same reason than the velocity controller.

2.3 Plant building

To build the plant, it is necessary to detail as much as possible its design. Section 2.3.1 shows how the main dimensions of the plant were determined. Then, in Section 2.3.2, the considerations taken into account to select every component are explained.

2.3.1 Dimensions

The plant's main dimensions are listed below.

- Propeller's diameter.
- Motor required power.
- Tower's height.
- Tower's separation.

Propeller's diameter

Reference [5] states that a typical rotor diameter for a real wind turbine is 17 m. After that a scale factor of 1:50 is applied. It was though to be an appropriate one as it is used for some aircrafts models [15]. This leads to a diameter of 34 cm for the plant's rotor. By having a look to the available propellers (see Reference [16]), the 14 inch (35,56 cm) diameter propeller was selected as it was the closest one to 34 cm.

Motor required power

This parameter was determined by imposing a similar value of plant's and real wind turbine's Reynolds number. This condition lead to a value of airspeed. Then, using Momentum Theory, the power required for the fan's motor can be determined.

From Reference [5], a typical value of airspeed for large wind turbines can be extracted: 13 m/s. Then, the Reynolds number of a real wind turbine is determined: $Re = \rho v_\infty D/\mu = 15 \cdot 10^6$. After that, the plant's Reynolds number is equalised to this value. Note that plant's values are designed with an accent.

$$Re' = Re \rightarrow \frac{\rho v'_\infty D'}{\mu} = \frac{\rho v_\infty D}{\mu} \rightarrow v'_\infty = \frac{v_\infty D}{D'} = 50 v_\infty = 650 \; m/s \tag{2.2}$$

This result is not possible to obtain. However, this does not imply that it is not possible to build a realistic model. Even though, the Reynolds number can not be the same, it can be high enough to consider that the aerodynamics forces do not change much.

Indeed, the main characteristics of the wind turbine aerodynamics are that of a potential flux. This means that the fluid is meant to be incompressible and inviscid. These assumptions are accepted when the Reynolds number is higher than 10^6. However, sometimes, these assumptions can also be acceptable for $Re = 10^5$. The free stream airspeed is calculated imposing this condition.

$$Re' = 10^5 \rightarrow v'_\infty = \frac{\mu Re'}{\rho D'} = \frac{1,8 \cdot 10^-5 \cdot 10^5}{1,225 \cdot 0,3556} = 4,13 m/s \tag{2.3}$$

After that, Momentum Theory in the hovering case is applied to the fan. This enables to calculate the power needed to obtain this free stream velocity.

$$P = \dot{m} v_\infty = \rho \left(\frac{v_\infty}{2}\right)^2 \pi \left(\frac{D'}{2}\right)^2 v_\infty = \frac{\rho v_\infty^3 \pi D'^2}{16} \tag{2.4}$$

$$P = \frac{1,225 \cdot 4,13^3 \cdot \pi \cdot 0,34^2}{16} = 2\ W \tag{2.5}$$

As the conditions to apply Momentum Theory in this situation (Reynolds number is close to the lower limit), it is decided to use a motor with a little bit more than two times this calculated power. This ensures that the fan will have enough power to move rotor's propeller. This decision does not add cost to the project as motors delivering this range of power are not expensive.

Tower's height

By having a look at typical values of wind turbines sizes [17], it can be seen that for the smaller ones, the ratio of height to rotor's diameter is about 1,5. This can be attributed to the ground effect. Rotor needs to be far enough from the ground so that it does not interfere with its airflow. Moreover, the ratio of height to diameter corresponding to a rotor of similar diameter to the one used to determine plant's rotor dimensions is about this same value (1,5). So, tower's height was determined using this factor.

$$\frac{H}{D} = 1,5 \rightarrow H = 1,5 \cdot 0,36 = 0,54\ m \approx 0,5\ m \tag{2.6}$$

Tower's separation

In order to avoid loss of influence from the fan to the wind vane and the rotor, it is important to keep them not too separated. A separation of about 1 m from the fan to the wind vane seems to represent good enough the free stream conditions of the wind. On the other hand, the distance from the vane to the rotor should be as low as possible to ensure that both towers "feel" the same wind. However, a minimum separation is required to avoid any collision between components of the different towers. For this reason, the distance from wind vane to the rotor has been set as about 30 cm.

2.3.2 Components selection

The components that had to be chosen were the following.

- Propeller.

- Mechanic joints.
- Motion transmission elements.
- Micro-controller.
- DC motor for the fan and the wind turbine hub.
- Driver for the two motors.
- Rotary encoders to measure the wind direction and the wind turbine's hub orientation.

Propeller

The propeller's dimensioning has already been explained in Section 2.3.1. So, the remaining task to do was to select one from the manufacturer's catalogue. The GWS Style Slowfly Propeller 14x4.7 (see Reference [18] for more details) was selected as it seemed the best option in terms of cost and performance. It was not expensive and its shape was appropriate (as it is used to build multi-rotor drones).

Mechanic joints

Different attachments were used. A set of washers, nuts and threaded bars were selected to attach wood plates to the main structure. Then, the electronic components (encoders and motors) were joined to this wood plates through some screws.

However, the fan motor was built-in to the tower as it seemed the easiest option. Nevertheless, experience has proven that the first option (threaded bars, nuts, washer and wood plates) would give more versatility to the model. In this way, components could be changed easier.

After that, motor's shafts needed to be coupled to wider diameter shafts because they were too thin to allocate the components that were required. To do so, a coupler was used [19].

Finally, to join the wind turbine's hub to the motor, a hub was employed [20].

Motion transmission elements

It was necessary to translate wind turbine's hub rotation to the encoder in order to determine its angular position. This was done thanks to a motion transmission mechanism based on pinion gears [21] and an auxiliary plane shaft. It was decided to use the same type of pinion gears to keep simplicity. If gears of different size would have been used, it would have been necessary to do some calculations to determine wind turbine's hub orientation from encoder's measurements.

Additionally, it was necessary to isolate rotor's spinning motion from the wind turbine's hub. This was achieved thanks to a ball-bearing [22]. As no special requirement was set for this component, the cheapest one matching the plant's dimensions was chosen.

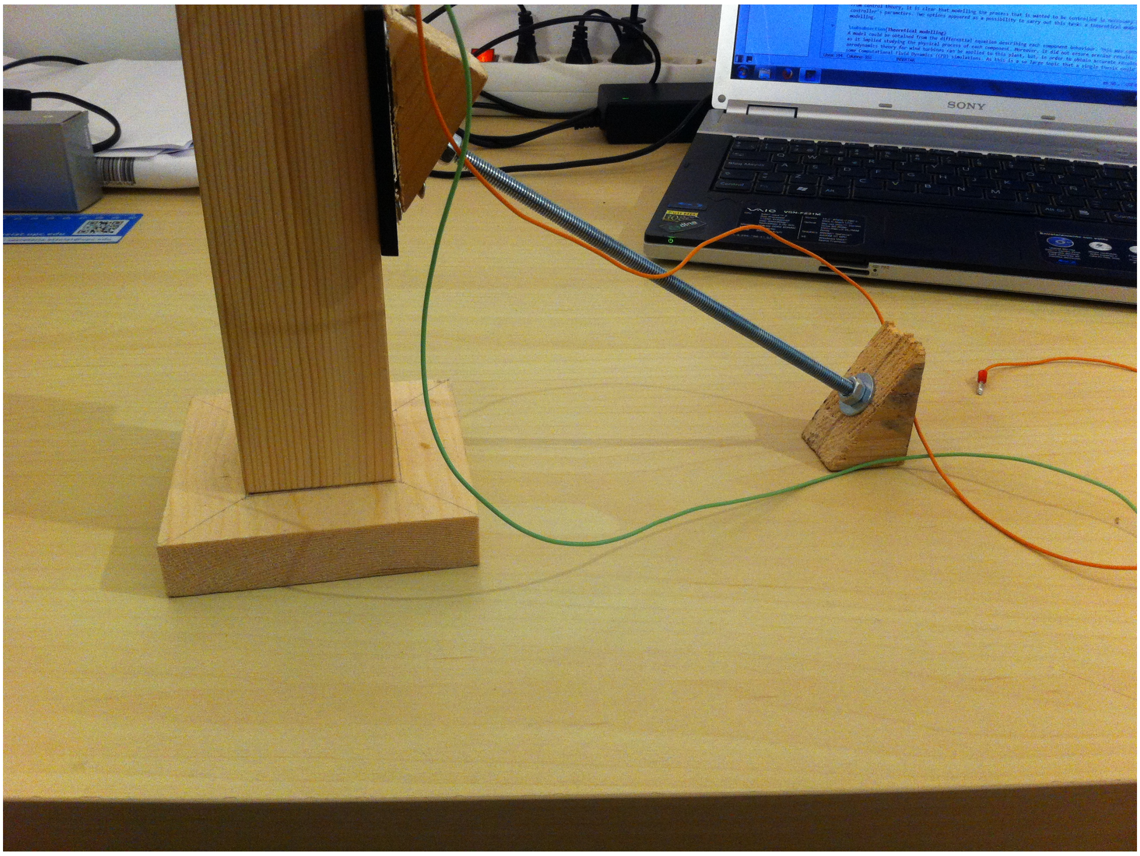


Figure 2.8: View of the bottom part of the "fan" tower. The structure used to counteract the torque generated at the bottom by the motor's thrust can be easily seen. It is used to avoid the tower to rotate back.

Micro-controller

The selection had already been restricted to Arduino micro-controllers in Section 2.2.1. Now, a board had to be selected from all the possibilities offered by Arduino. It was decided to use the UNO rev3 as it is the most available one. Moreover, it is quite cheap and it fulfilled all the requirements.

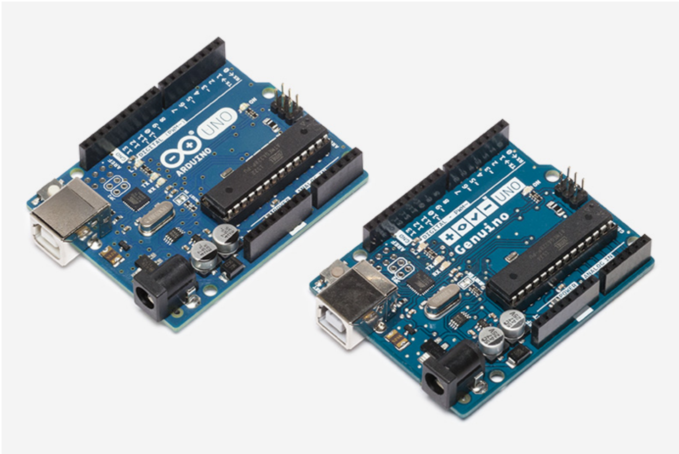


Figure 2.9: Latest version of Arduino boards. Note that the Genuino is the name given to the product outside of the United States.

DC motors

The fan DC motor had to supply around 5 W of power as it was determined in Section 2.3.1. The cheapest motor from the supplier delivering a little bit more than this power was selected [23] as it had not to fit any other specifications.

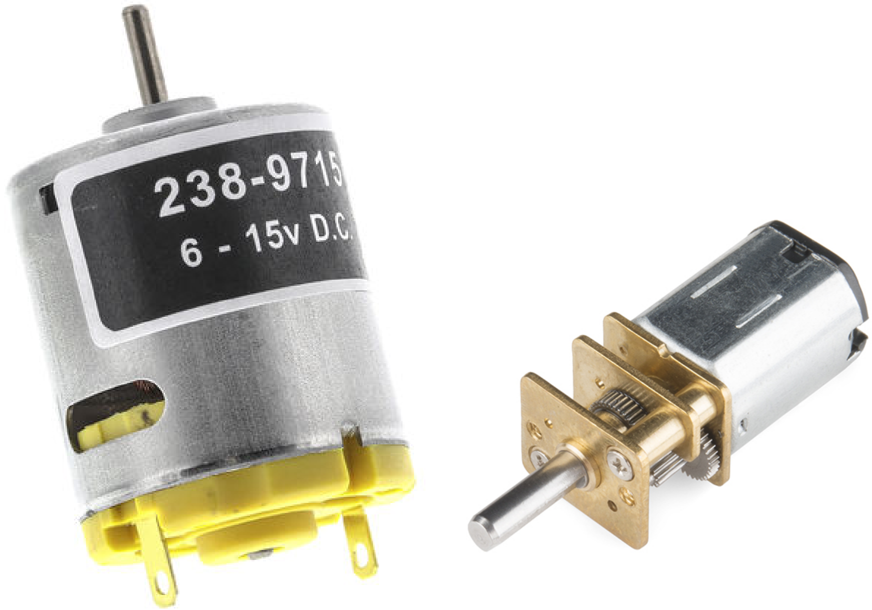
Then, the wind turbine's hub motor had to be chosen. This component is the one in charge of orientating the rotor's axis. Thus, speed is not a requirement but high torque is the main one. For this reason a gear motor with a high gear ratio (converting angular speed into torque) was chosen [24].

Motors driver

The micro-controller can deliver power to other components. Nevertheless, the power that can be supplied is not high enough to feed an electric motor. This is why a driver is used. It is connected to a power supply that delivers the voltage required by the motors and, at the same time, it is connected to the micro-controller that says how much power should be transferred to the motors.

The easiest to use option was selected [25]. In this way, it was avoided to have some problems when soldering or making any other operation required to connect the driver to the motors, power supply and micro-controller.

This driver is based on a H-bridge of transistors to drive the motor. The micro-controller sends a Pulse Width Modulation (PWM) signal to the driver that determines the time the motor should be one, the so called duty cycle. In this way, the voltage applied to the



(a) Fan's motor. (b) Yaw tracker motor.

Figure 2.10: It is interesting to stress that these two motors present different characteristics. The fan's one can rotate very fast while the wind turbine's motor can deliver a very high torque.

motor is regulated. In the linear region, a duty cycle of 50% is equivalent to constantly feeding the motor at 50% of operating voltage.

Rotary encoders

The sensor in charged of measuring the wind direction and angular position of the wind turbine's hub has to be very precise as this is the main goal of the platform. Then, it is not expected to save money with this component.

The next question to answer was whether to choose an absolute or incremental encoder. As the goal of the project is to control angular position, absolute encoder could seem a better option at first sight. However, as the wind turbine's hub has not to rotate more than one revolution, the incremental encoder seems enough.

For these reasons, an incremental encoder with 1024 pulses per revolution and two channels enabling to determine the spinning speed was chosen [26].

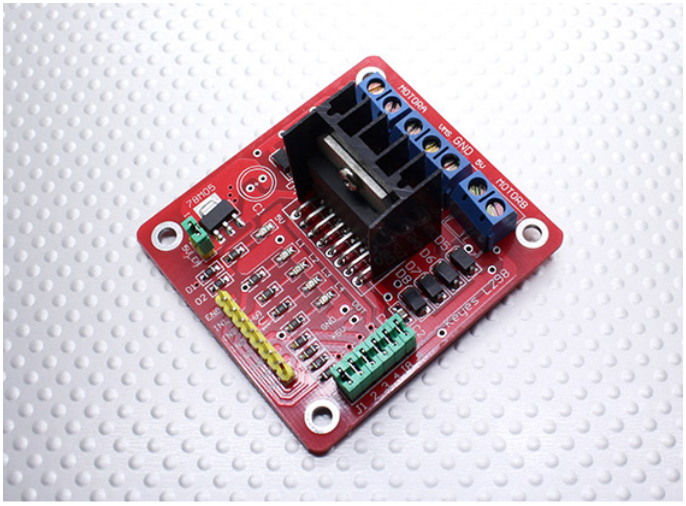


Figure 2.11: The motors driver is used to deliver the amount of power decided by the Arduino to the motors.

Figure 2.12: The rotary encoder used sends 1024 pulses per revolution. This precision can be increased four times by having a look at the rising and falling of the two channels as they are equally spaced.

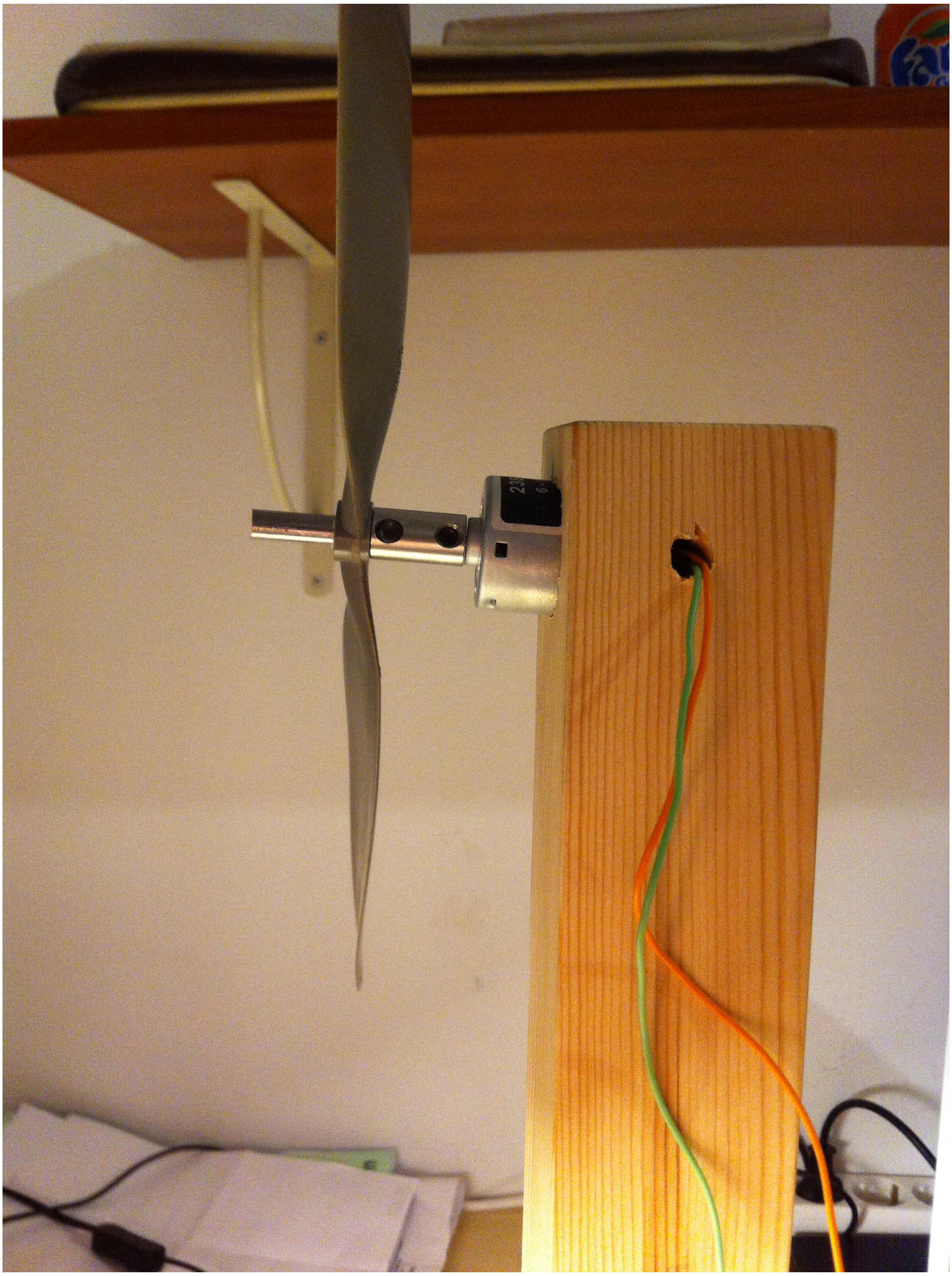


Figure 2.13: View of the upper part of the "fan" tower. The built-in motor and the coupler can be observed.

Figure 2.14: Back view of the "wind turbine" tower. The motion transmission can be easily appreciated. See the 4 pinion gears: one attached to the motor shaft, two of them linked to the auxiliary shaft and the last one assembled to the rotary encoder.

2.4 Plant modelling

As it was explained in Section 2.2.2, it was decided to review the theory modelling plant's components. This is done in Section 2.4.1. After that, a model is obtained for the angular speed of wind turbine's hub thanks to experimental measurements in Section 2.4.2.

2.4.1 Study of theoretical models

To have a first idea of what to expect from experimental modelling results, the physics behind direct current electric motors is revised. After that, the literature is consulted to know the effect of the wind on the rotor angular position.

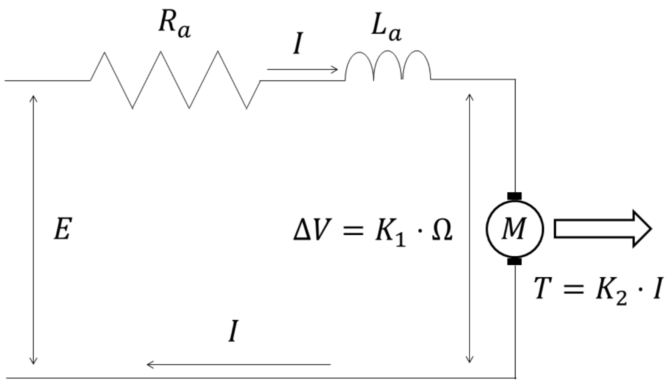


Figure 2.15: Electric circuit driving a motor.

Consider the scheme shown in Figure 2.15. Let's apply Kirchhoff's Law to the electric circuit. E is the voltage applied to the circuit terminals, R_a is the armature resistance, L_a is the armature inductance and K_1 is the constant relating counter-electromotive force with the motor's angular speed Ω. This last relationship is based on Faraday-Lenz law.

$$E = R_a I + L_a \frac{dI}{dt} + K_1 \Omega \tag{2.7}$$

On the other hand, let's apply Newton's second law to the rotational motion of the motor. T stands for the electric torque applied to the motor shaft, J is the motor's inertia, b is the friction factor and K_2 is the constant relating torque with the circuit's current intensity I. This relationship is due to Lorenz forces appearing in electric charges moving inside a magnetic field.

$$T = K_2 I = J \frac{d\Omega}{dt} + b\Omega \tag{2.8}$$

Now, the Laplace transform of both equations is done. Then, the intensity is isolated from the two equations. After that, by equating the two expressions of intensity, the transfer function relating motor's angular speed with the voltage applied to it is obtained.

$$E = R_a I + L_a s I + K_1 \Omega \rightarrow I = \frac{E - K_1 \Omega}{R_a + L_a s} \tag{2.9}$$

$$K_2 I = (Js + b)\Omega \rightarrow I = \frac{(Js + b)\Omega}{K_2} \tag{2.10}$$

$$\frac{E - K_1\Omega}{R_a + L_a s} = \frac{(Js + b)\Omega}{K_2} \rightarrow \frac{\Omega}{E} = \frac{K_2}{(R_a + L_a s)(b + Js) + K_1 K_2} \tag{2.11}$$

If the motor was ideal, the armature inductance would be zero, $L_a = 0$. However this is not the case of project's plant, it can be considered in a first approximation. This is why, an electric motor can be considered as a first order system.

Reference [5] states that aerodynamics forces tend to align the rotor with the wind direction when it is yawed. So, it seems a good approach to first model the motor's speed without wind. Then, control its angular position and observe its response in front of wind. Maybe some corrections should be necessary after experimental measurements, but it is still a good first approximation to obtain the controller's parameters.

2.4.2 Experimental modelling

The angular speed of the motor is modelled thanks to the measurements taken from the encoder. The aim is to relate the input from Arduino to motor driver with the output obtained as an angular speed. First, the experimental results will be presented and then a model for the motor's angular speed will be determined. But before, lets take into account some considerations regarding the flux of information and how it is treated.

Information treatment

It should be noted that Arduino sends a Pulse Width Modulation (PWM) signal to the motor driver with 8 bits of resolution. Then, the value sent by the micro-controller to the driver varies from 0 (for a pwm with no duty cycle) to 255 (for a pwm with 100% of duty cycle).

Moreover, angular speed is not directly measured. It is estimated from encoder readings which indicate angular position. In order to estimate velocity, these values have to be differentiated with respect to time in a numerical method. At this stage, this calculations were performed in a worksheet in order to save time to the micro-controller. Derivatives were implemented in a first-order approximation as the one shown in Equation 2.12.

$$\frac{d\theta}{dt} \approx \frac{\theta(t_k) - \theta(t_{k-1})}{T_s} \tag{2.12}$$

After that, it is interesting to remark the main characteristics of the Arduino code. If the reader is interested in more information about the source code, it can be found in the project's technical sheets.

- Time is measured with micros() instead of millis() as it is a thousand times more precise.
- Use of floats is avoided because of the computational effort they require from the platform [27]. Moreover, integers are meant to yield more accurate results than floats. For this reason, it is decided to work with ticks to measure angular position.

A tick corresponds to a change in the encoder output. They can be related to degrees as 4096 ticks correspond to one revolution (360º).

- Some graphics are shown with º for the angular position, º/s for the angular speed and % for the duty cycle. However, it should be kept in mind that the units that matter to the micro-controller are ticks and ticks/s for the angular position and speed respectively. Moreover, the Arduino expresses the PWM's duty cycle with an 8 bit value not in %. Thus, for the micro-controller, 0 corresponds to 0% duty cycle and 255 to 100%.
- The baud rate is set to the maximum value in order to gain speed when executing the code.
- An input from the user is required to start the experiment. It is introduced by the monitor and communicated to the Arduino thanks to its serial port. Moreover, it ends after a set time.
- In order to ensure that the sampling time is the expected one, an internal loop is created where the minimum operations, apart from measuring and printing their values, are executed.
- The rotary encoder measurements are performed thanks to interrupt functions. It is necessary to keep track of all changes in encoder output as missing them leads to a measuring error. The interrupt functions are attached to any change produced in the two encoder's outputs.

Results

The first step when modelling the plant was to choose a correct sampling time, T_s. For a first order system, it is suggested that the sampling time should be small enough to obtain four measurements before the response arrives to 95% of its steady state value at time $T_{95\%}$. This is $T_s < T_{95\%}/4$. However, it is recommended to chose a value greater than $T_{95\%}/15$ as no more precision is gained and computation effort can be increased.

By having a look at motor's response to an input of 128, 50% of duty cycle, (see Figure 2.16), it can be seen that the smallest value that the sample time can have is 0,01 s as the time constant is of 0,05 s. For a first order system, when the time is 3 times the time constant, the system has reached 95% of its steady state response ($1 - e^{-3T/T} = 1 - e^{-3} = 0,95$).

Now that the sampling time has been set to 0,01 s, it is time to find the linear zone of the motor and its time constant and static gain. To do so, the system's response in front of different duty cycles will be studied. Then a linear regression will be made to determine the static gain and linear zone. The time constant will be the one associated to motor's response in this linear zone.

Figure 2.17 shows clearly that the linear region is for PWM between 25 and 64. This range covers the majority of necessary angular speeds (from one turn per second to nearly 1/4 of turn per second) to properly control motor's angular position. The R^2 parameter indicates that this region can be considered as linear because it is slightly higher than 0,95. Then, system's static gain can be set to $K = 64,189$.

To determine motor's time constant, it is necessary to study its time response to the PWM step input. By having a look at Figure 2.18, it can be seen that time constant can be set

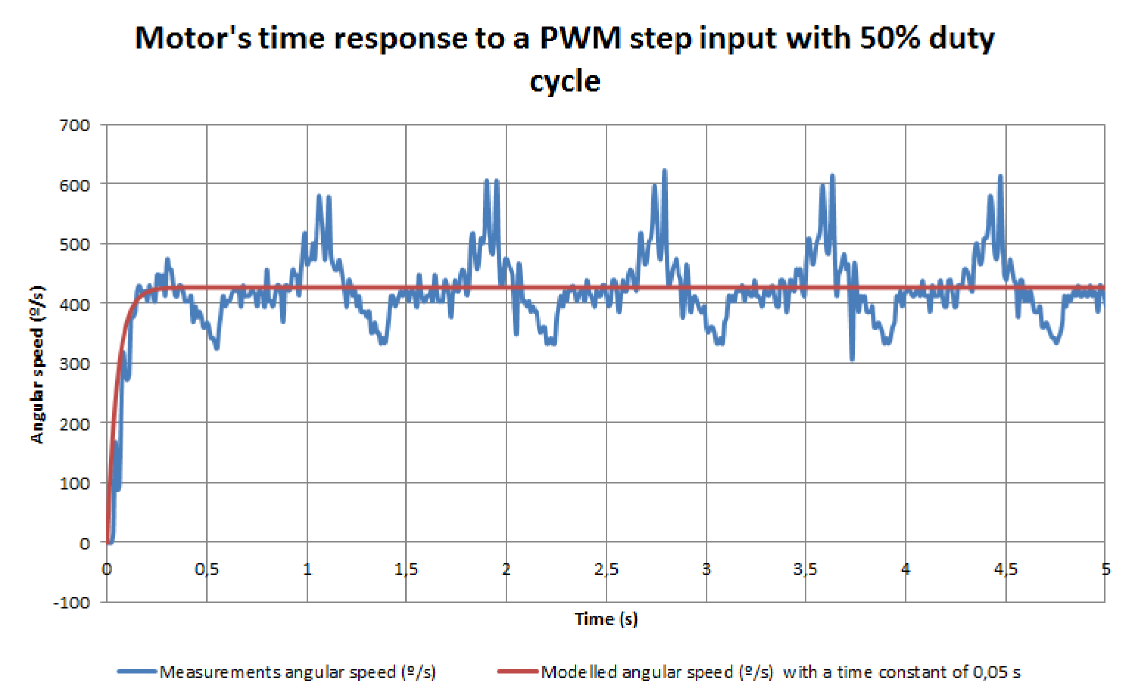


Figure 2.16: Motor's response to a pwm input of 128 bit which correspond to 50% of duty cycle. Note that angular speed at steady state is quite high, more than one revolution per second. The red curve corresponds to a first order system with a time constant of 0,05 s.

to 0,20 s. Even though, for this input, it matches very accurately the measured response, for other inputs, it is not so precise. However, this value seems the best trade-off option as it is a little bit faster for some inputs and sometimes a slightly slower for other inputs (see Figure 2.19).

By observing the measured time response, a lot of noise can be seen. In order to cut off this unwanted measurements, a filter can be implemented in the Arduino. This will be very helpful to avoid false error between set point and actual value of desired speed or position.

To implement this filter, its transfer function is converted to the time domain in the form of a differential equation (see Equation 2.13) thanks to the Laplace anti-transform. Note that Ω_f corresponds to the filtered angular speed while Ω_m is the measured angular speed.

$$\frac{\Omega_f}{\Omega_m} = \frac{1}{1+T_c s} \rightarrow (1+T_c s)\Omega_f = \Omega_m \rightarrow \Omega_f + T_c \frac{d\Omega_f}{dt} = \Omega_m \qquad (2.13)$$

Then, all derivatives are approximated by a first order polynomial considering the current and previous samples (see Equation 2.14). In this way, the filter can be coded to the Arduino.

$$\Omega_f(t_k) + T_c \frac{\Omega_f(t_k) - \Omega_f(t_{k-1})}{T_s} = \Omega_m(t_k) \rightarrow \Omega_f(t_k) = \frac{\frac{T_c}{T_s}\Omega_f(t_{k-1}) + \Omega_m(t_k)}{1+\frac{T_c}{T_s}} \qquad (2.14)$$

The characteristic time of this filter is set to 0,05 s which is a quarter of the time constant. This option has seemed the best trade-off between lowering noise and keeping a repre-

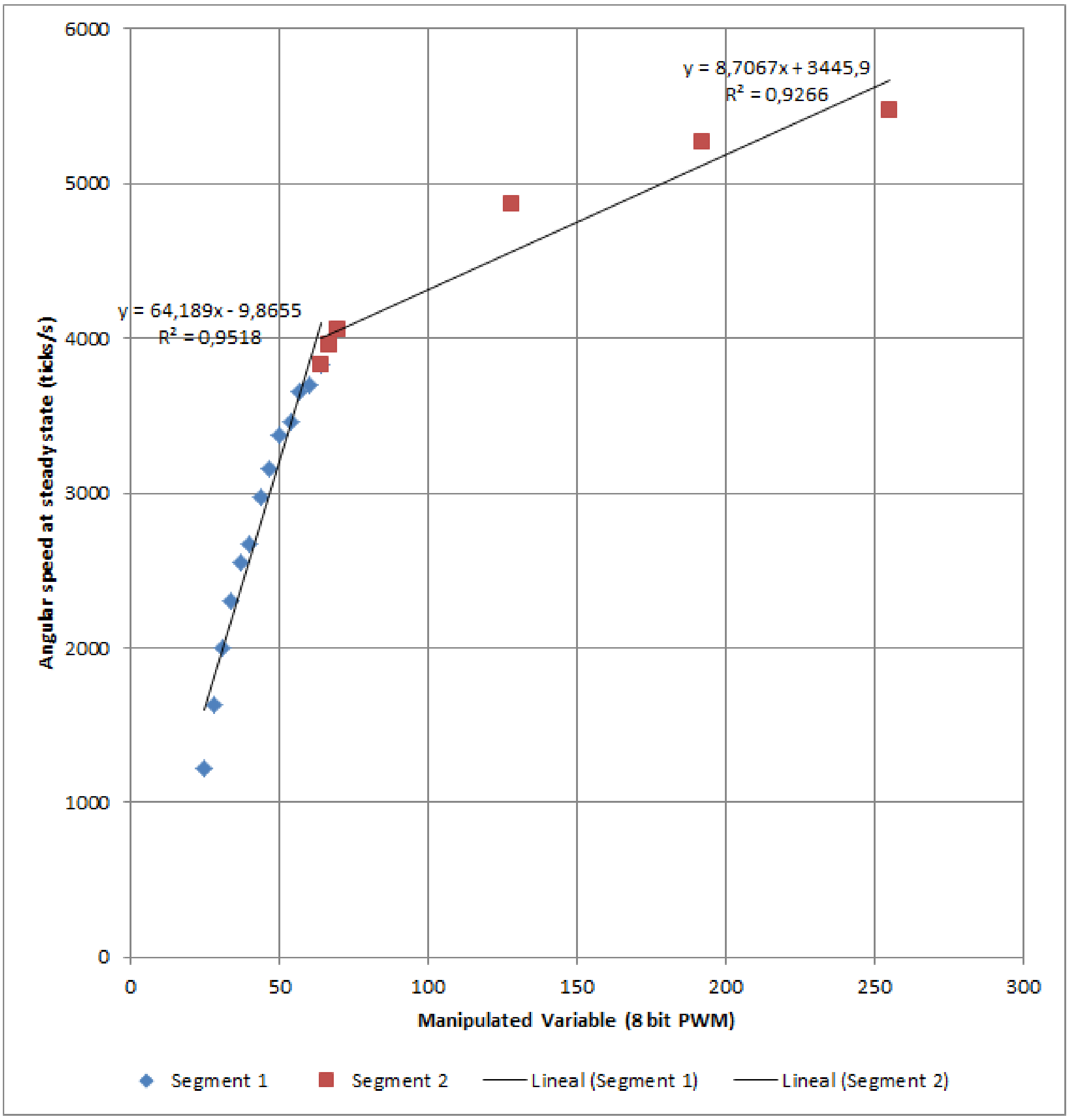


Figure 2.17: Linearity study of the motor's response to a PWM input.

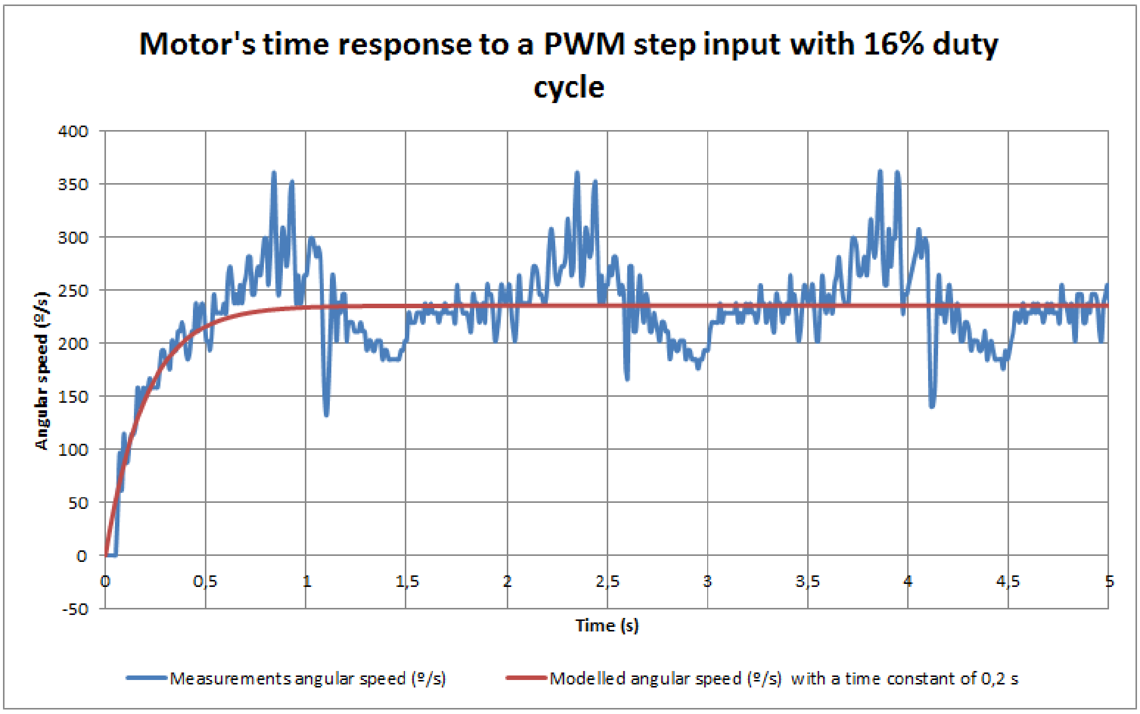
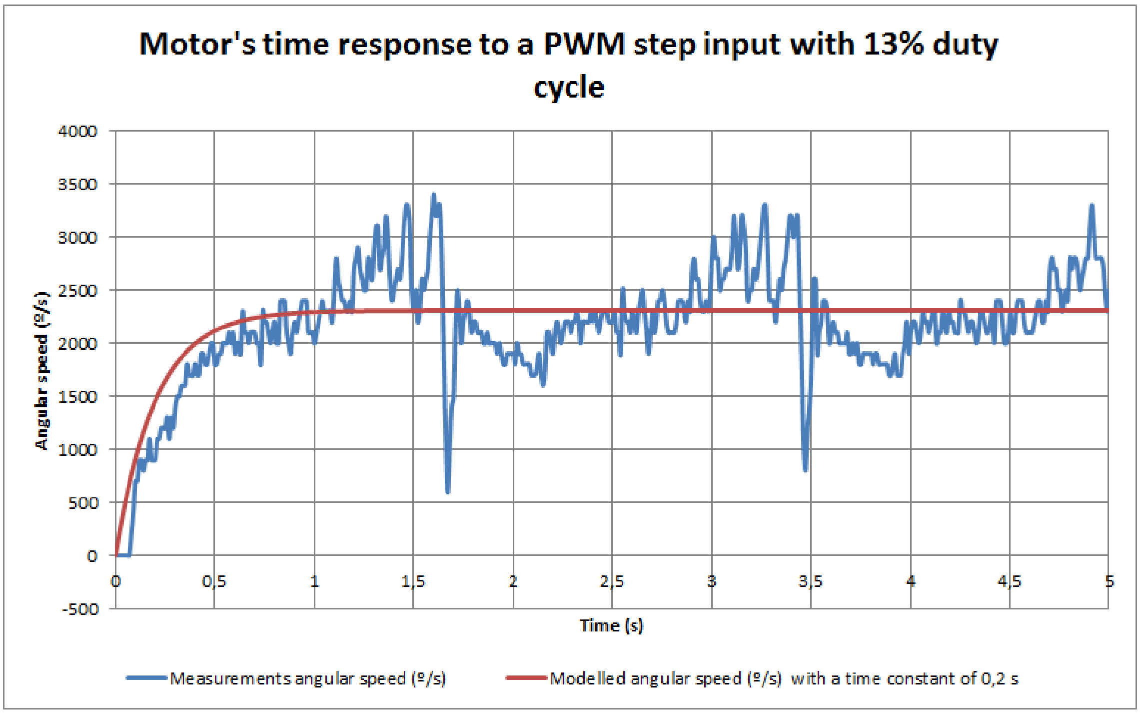
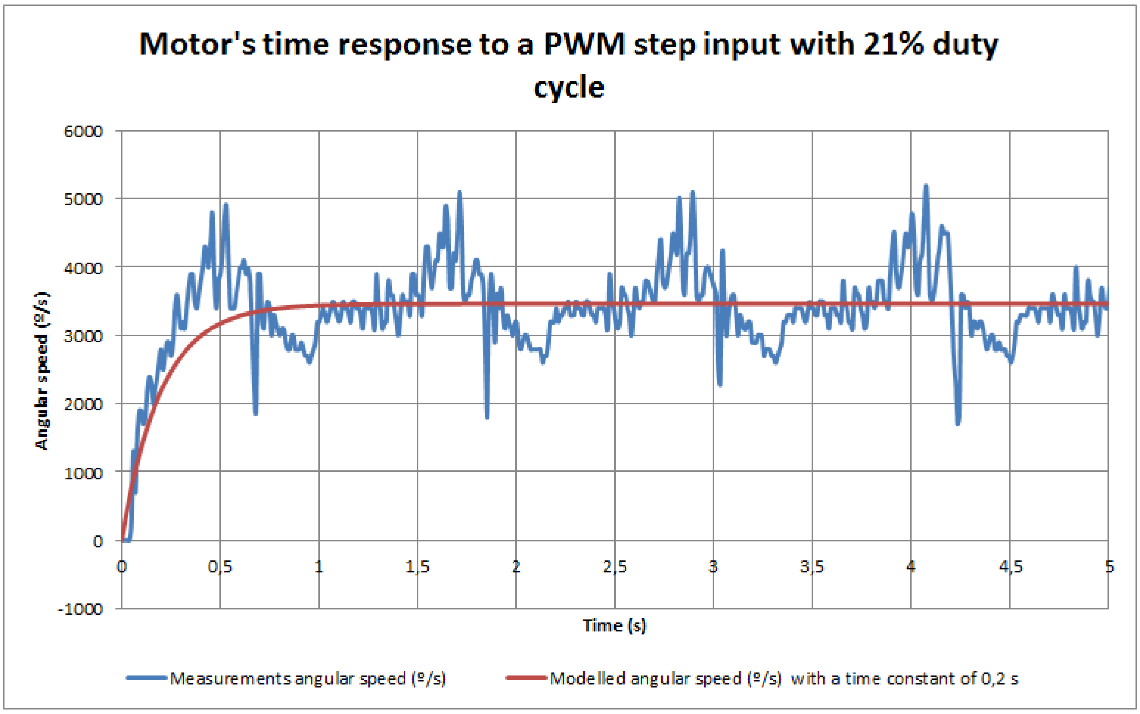


Figure 2.18: Motor's response to a PWM step input with 16% duty cycle.

sentative signal. As it can be seen in Figure 2.20, filters with higher time constant, T_c, (0,10 s, 0,15 s and 0,20 s) seem to modify too much the filtered signal with respect to the measured one. On the other hand, filter with $T_c = 0,05\ s$ represents quite well the evolution of measured signal while noise is lowered at the same time.



(a) System's response for a PWM step input with 13% duty cycle.



(b) System's response for a PWM step input with 21% duty cycle.

Figure 2.19: Motor's time response for different PWM step inputs. The red curve still represents a first order system with a time constant of 0,20 s.

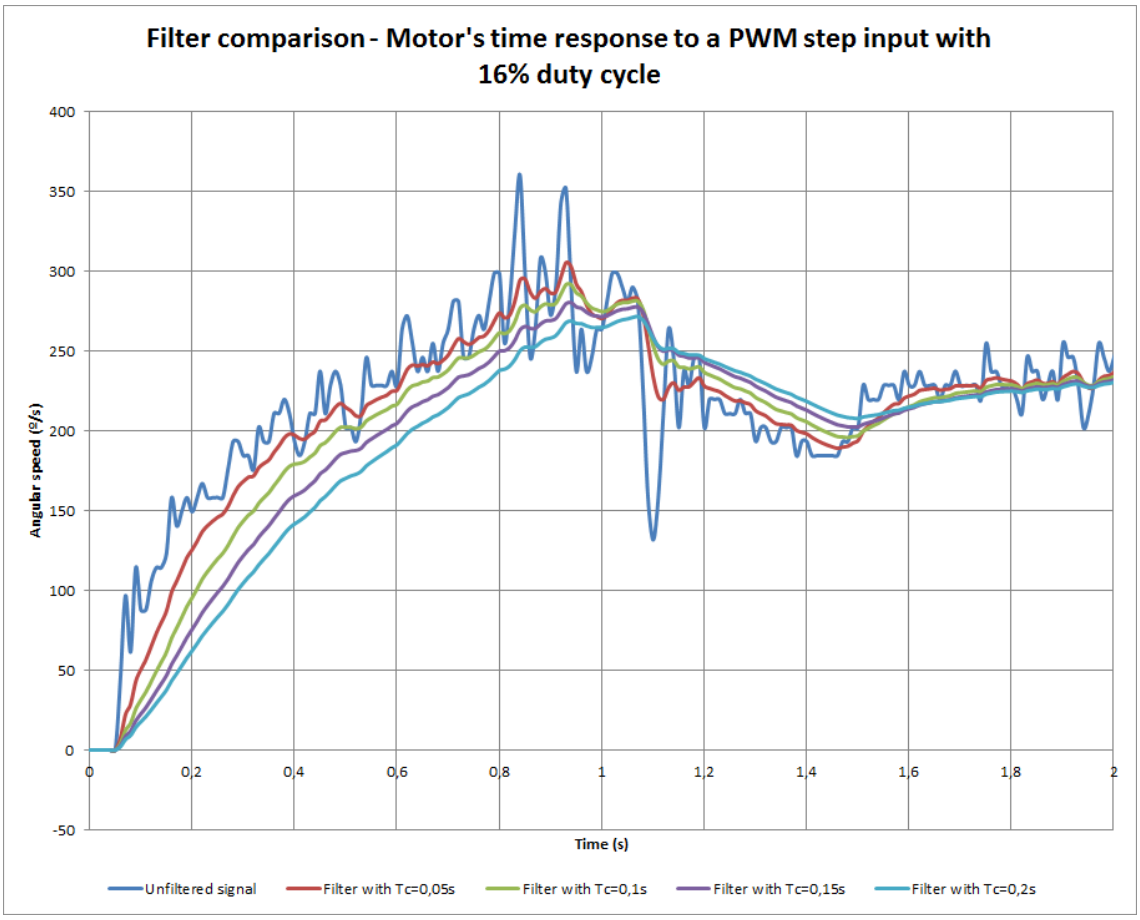


Figure 2.20: Comparison of different filters for a PWM step input of 40.

2.5 Control algorithm

After obtaining an expression that describes the time response of the wind turbine hub's angular speed, it is necessary to determine the controller's parameters. As it was explained in Section 2.2.3, a cascade control strategy has been selected. This means that two controllers have to be tuned: one for the angular speed, Section 2.5.1, and one for the angular position, Section 2.5.2. However, all of this is made in the absence of wind. The modifications made to include the wind and the results obtained are shown in Section 2.5.3.

2.5.1 Angular speed

As a model has been obtained to describe system's behaviour, poles placement technique will be tested to set controller's parameters. Then, it will be explained how the controller has been implemented on the Arduino platform. Finally, the results will be discussed.

I+P tuning through poles placement

To apply the poles placement methodology, the desired denominator and the system's denominator should be made equal. From Figure 2.21, it is clear that the resulting system of an I+P controller acting on a first order system is a second order one. Then, a second order system has to be selected as the desired behaviour of the controlled system. From all the possibilities, the critically damped system is the chosen option as it is a good combination between fast response and stability. Then, by comparing desired system transfer function, $F_d(s)$, with controlled system transfer function, $F_c(s)$, the I+P controller can be tuned.

$$F_d(s) = \frac{\omega_n^2}{s^2 + 2\omega_n s + \omega_n^2} \ \& \ F_c(s) = \frac{KK_I/T}{s^2 + (1 + KK_P)/Ts + KK_I/T} \tag{2.15}$$

From Equation 2.15, the parameters of I+P controller are determined. From Equation 2.16, it is clear that both parameters, K_P and K_I, depend on the natural frequency, ω_n, of the desired system. It is not straightforward what value of ω_n is the best for the system. A first guess can be set as choosing the natural frequency for which the time required to reach the 98% of the steady value, $T_{98\%}$, is four times the system's time constant without any controller, T. Nevertheless, this selection of ω_n will be checked through the experience and changed if it does not prove enough satisfactory.

$$K_P = \frac{2\omega_n T - 1}{K} \ \& \ K_I = \frac{\omega_n^2 T}{K} \tag{2.16}$$

From the time response of a critically damped second order system, it can be seen that $T_{98\%} = 6/\omega_n$ (see reference [28]). If our system's time constant, $T = 0,20\ s$, and, $T_{98\%} = 4T$, it follows that $\omega_n = 4/(6T) = 7,5\ rad/s$. Then, this result, along with Equation 2.16, can be used to determine the controller's parameters: $K_P = 0,03116$ and $K_I = 0,1753$.

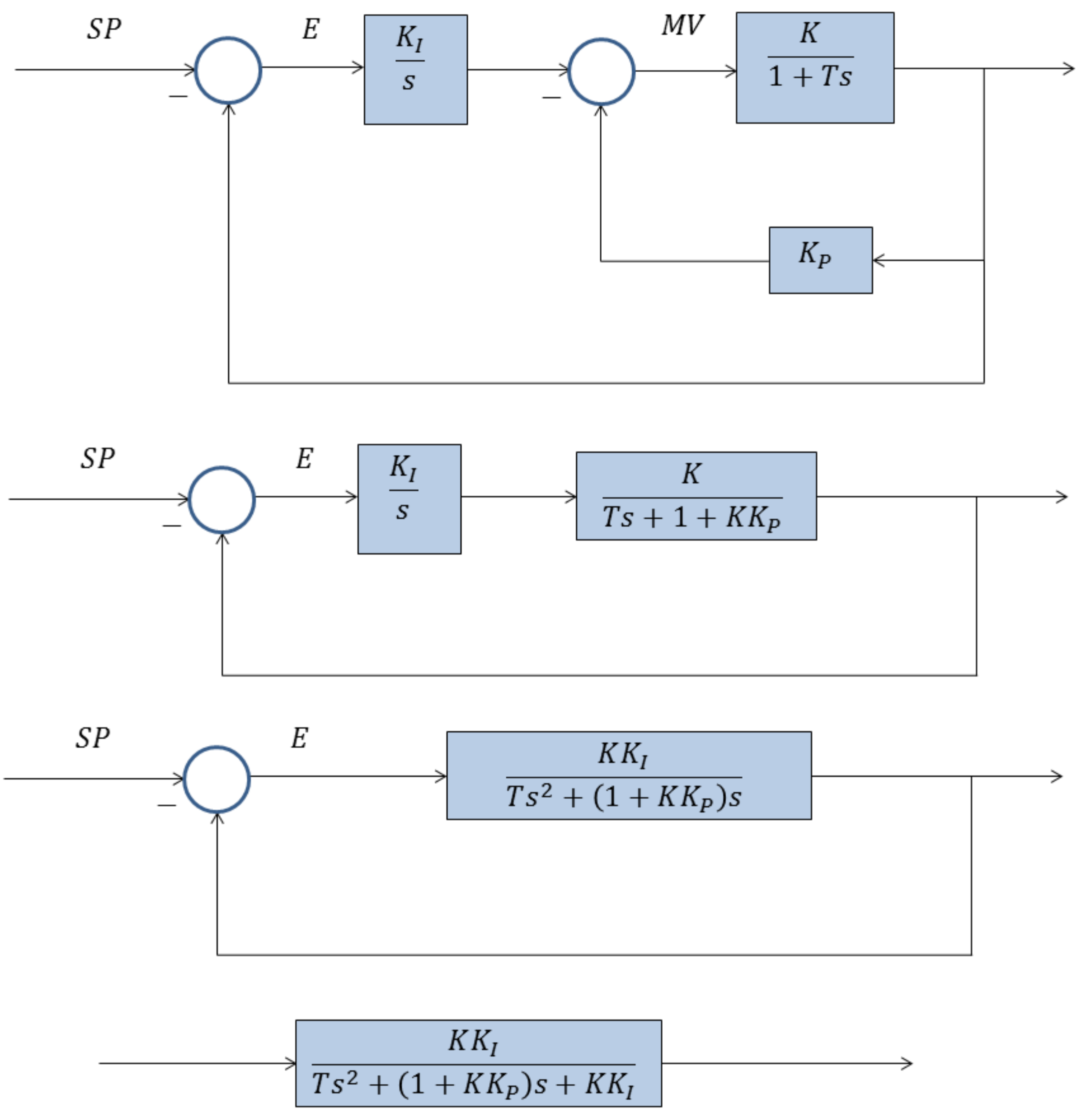


Figure 2.21: Procedure to determine the transfer function of a 1st order system with and I+P controller.

I+P implementation on Arduino

After that, the control algorithm has to be implemented on the Arduino platform. As it is a digital micro-controller, the expressions have to be converted to their discrete form. First, continuous time expression for an I+P controller is written in Equation 2.17 thanks to Figure 2.7. Note that it has been taken into account that $e(t) = sp(t) - \omega(t)$.

$$mv(t) = K_I \int_{t_0}^{t} e(t)dt - K_P\Omega(t) = K_I \int_{t_0}^{t} sp(t)dt - K_I \int_{t_0}^{t} \Omega(t)dt - K_P\Omega(t) \tag{2.17}$$

Then Equation is converted to the discrete form. To do so, the integrals have been approximated as the sum of the variables' values at each sample time multiplied by the sampling time, T_s. The final result is written in Equation 2.18.

$$mv(k) = K_I \sum_{i=0}^{k} sp(i)T_s - K_I \sum_{i=0}^{k} \Omega(i)T_s - K_P\Omega(k) \tag{2.18}$$

To make this algorithm easier to program, the previous manipulated variable value is used, $mv(k-1)$. Resulting in the final form written in Equation 2.21.

First, consider discrete expression for the manipulated variable at the previous time step.

$$mv(k-1) = K_I \sum_{i=0}^{k-1} sp(i)T_s - K_I \sum_{i=0}^{k-1} \Omega(i)T_s - K_P\Omega(k-1) \tag{2.19}$$

Then, split all the sums at the previous time step, like in this manner: $\sum_{i=1}^{k} \Omega(i) = \sum_{i=1}^{k-1} \Omega(i) + \Omega(k)$.

$$mv(k) = K_I \sum_{i=0}^{k-1} sp(i)T_s + K_I sp(k)T_s - K_I \sum_{i=0}^{k-1} \Omega(i)T_s - K_I\Omega(k)T_s - K_P\Omega(k) \tag{2.20}$$

$$mv(k) = mv(k-1) + K_P(\Omega(k-1) - \Omega(k)) + K_I T_s(sp(k) - \Omega(k)) \tag{2.21}$$

To implement this equation in the Arduino platform, it is necessary to have into account that it is better to work with integers instead of floats. As the values of K_P and K_I are decimals, they have to be converted into a fraction expression with integers. To have a high precision, up to 3 numbers are used to for the fraction's numerator and denominator.

Then, instead of using $K_P = 0,03116$ and $K_I T_s = 0,001753$, $K_P = 21/674$ and $K_I T_s = 1/571$ are used. It is important to note that the numerator must multiply the respective variable ($\Omega(k)$, $\Omega(k-1)$ or $sp(k)$) and, then, it must be divided by the denominator. If not, the fraction result is smaller than zero, and as integers are being used, it is truncated to zero. Again, the whole code can be found in the project's technical sheets if more detailed information is desired.

Moreover, the Arduino code includes an anti wind-up instruction to prevent an excessive increase in the manipulated variable due to the integral control action. This phenomenon can slow down the plant's response to a change in the set point if there is some static error during a long period of time [29]. To do so, the manipulated variable value is limited from -255 to 255 which correspond to a PWM with 100% of duty cycle.

Results discussion

As many assumptions have been made, it was necessary to check whether the controller would work properly or not. To do so, it was tested in front of four different speed set points (1024 ticks/s, 2048 ticks/s, 3072 ticks/s, 4096 ticks/s). It was attempted to cover all the linear zone.

For all the time responses, note that the oscillations in the steady state are due to the model. They are not caused by the controller as they were also observed when modelling.

For the two smaller speeds, the controller seemed to not work well (See Figures 2.22 and 2.23). However, the controller worked quite fine for larger speed set points (see Figures 2.24 and 2.25).

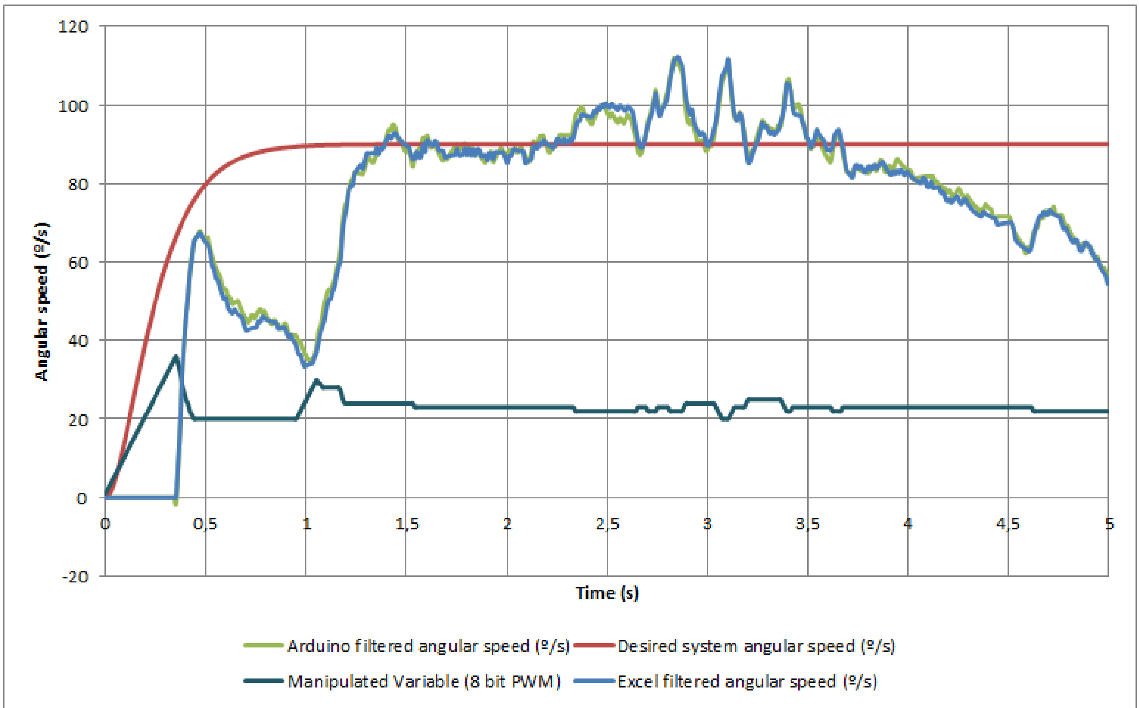


Figure 2.22: Controlled system response to $SP = 1024\ ticks/s = 90^o/s$.

The so non-ideal response for a set point of 1024 ticks/s (Figure 2.22) can be justified as the plant is operating close to the lower limit of the linear region. If it behaves better for other set points where linearity is more clear, the controller can still be accepted.

The system's response for a set point of 2048 ticks/s (Figure 2.23) is quite better than for 1024 ticks/s. However, it still shows some undesired results as a quite high error in the steady state.

When a set point of 3072 ticks/s is set, the system responds a lot better than before (Figure 2.24). This result is acceptable as it follows quite precisely the theoretical time response and it has a low error in the steady state.

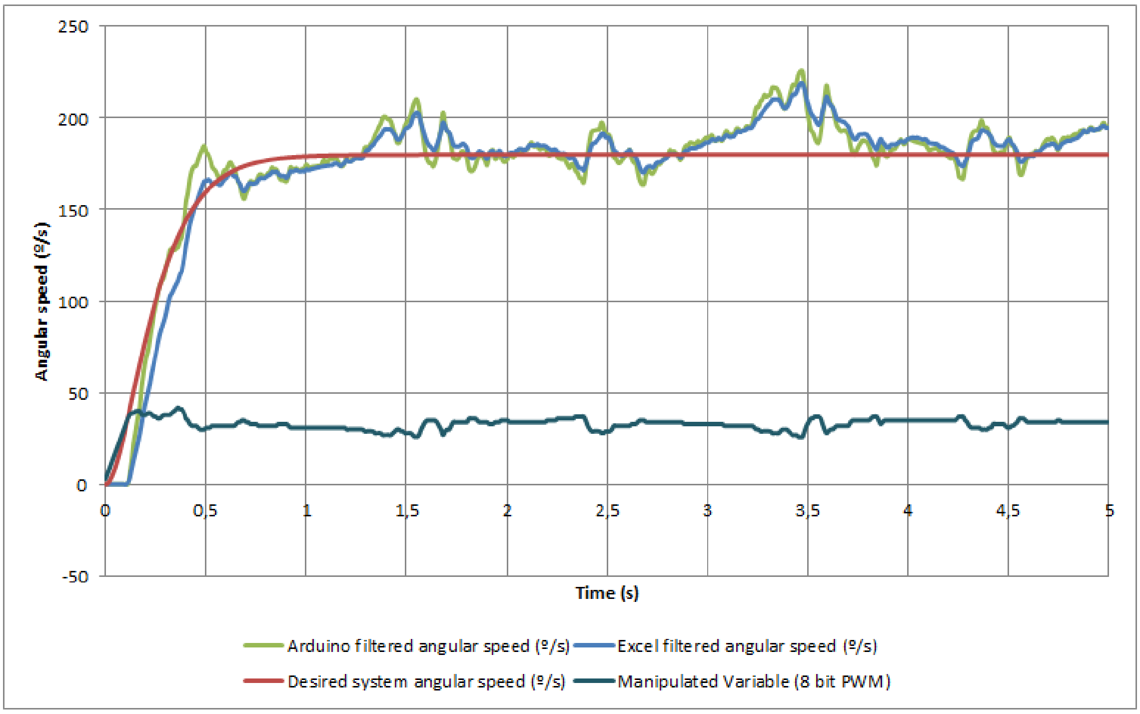


Figure 2.23: Controlled system response to $SP = 2048\ ticks/s = 180^{o}/s$.

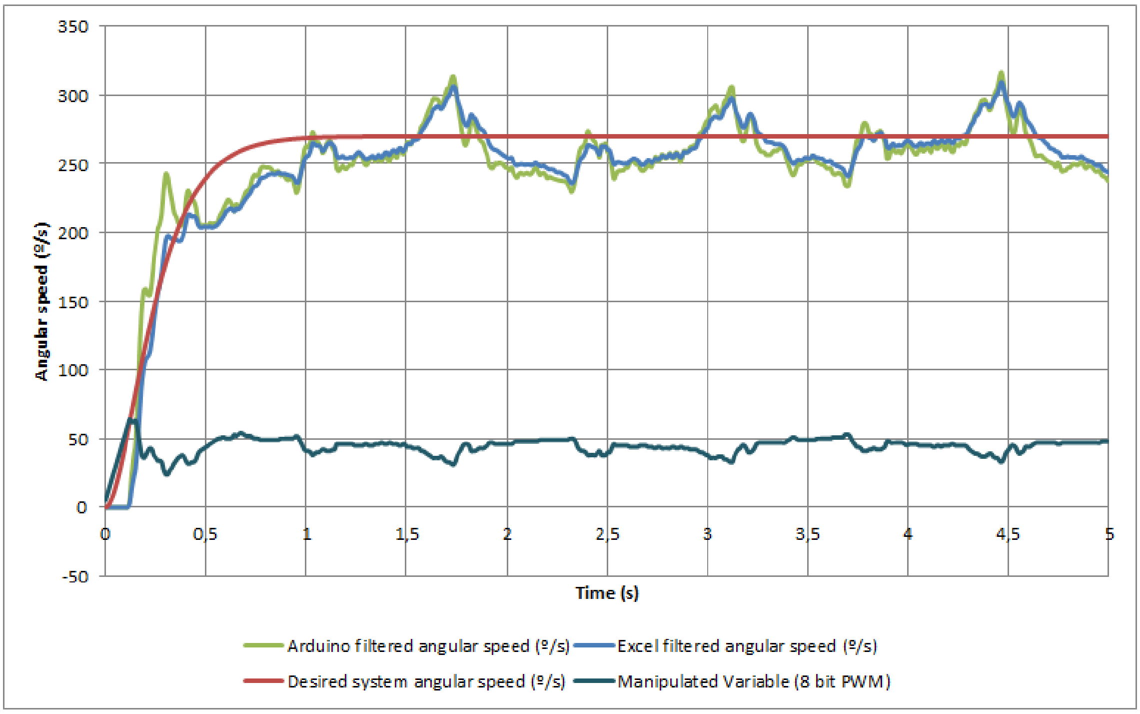


Figure 2.24: Controlled system response to $SP = 3072\ ticks/s = 270^{o}/s$.

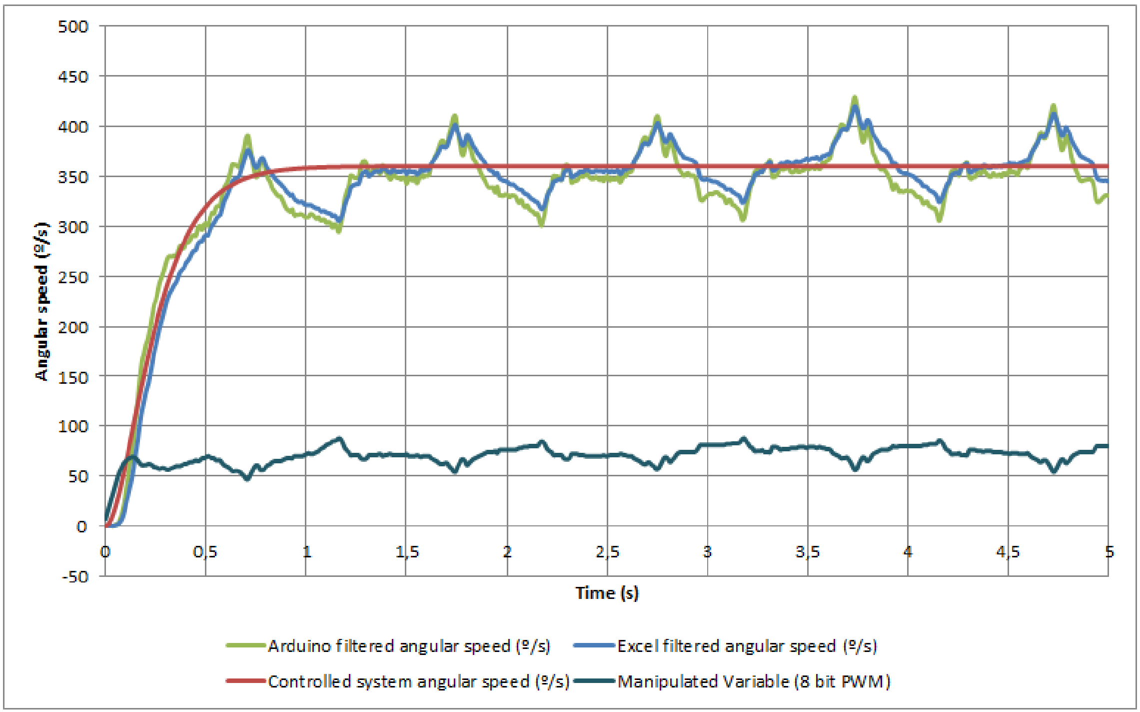


Figure 2.25: Controlled system response to $SP = 4096\ ticks/s = 360^{o}/s$.

In front of a set point of 4096 ticks/s, the plant response (Figure 2.25) is very satisfactory. The results are even a little bit better than for a set point of 3072 ticks/s.

All this analysis, lead to conclude that, for low speed set points, angular speed is difficult to control. To overcome this problem, an adaptive controller could be used. However, this speed controller has to be tested under a position controller. If it works fine, there is no need to use an adaptive control for the angular speed.

2.5.2 Angular position

After obtaining a controller for the angular speed, it was time to determine the orientation controller parameters. As the plant's model has been obtained experimentally, some assumptions have been made to cope with some uncertainties (such as noise). Moreover, some times, it has been necessary to choose a trade-off solution not satisfying all the ideal requirements.

For these reasons, it would have been difficult to obtain a controller from an analytical method. It seemed better to tune a controller thanks to some experimental technique, such as Ziegler-Nichols on closed and proportional loop. Then, if time allowed it, an analytically tuned controller could be tested.

First, the procedure followed to tune empirically the controller is explained. Then, the results obtained thanks to this controller are studied.

Controller experimental tuning

Ziegler-Nichols seems to be one of the most techniques used to determine controller's parameters empirically. Indeed, it is a common standard in the industry [28]. For this project, the closed-loop Ziegler-Nichols method was used.

This technique consists in using a P controller over the process that it is desired to control. The proportional gain starts from a low value. Then, it is slowly increased until the system's response in the steady state is oscillating. This gain is named as the critical gain, K_C, and the oscillation's period is also named as the critical period, T_C. These two values allow to tune a P, PI or PID controller using a table developed by Ziegler and Nichols (see Table 2.1).

Form	P	PI	PID
K_P	$0,5K_C$	$0,45K_C$	$0,6K_C$
K_I	0	$1,2K_P/T_C$	$2K_P/T_C$
K_D	0	0	$K_PT_C/8$

Table 2.1: Ziegler-Nichols tuning table based on system's critical response.

This plant started to oscillate for a proportional gain, $K_P = 4,8$. The critical period was measured to be $T_C = 1,16\ s$. The system's response that led to this results are shown in Figures 2.26 and 2.27.

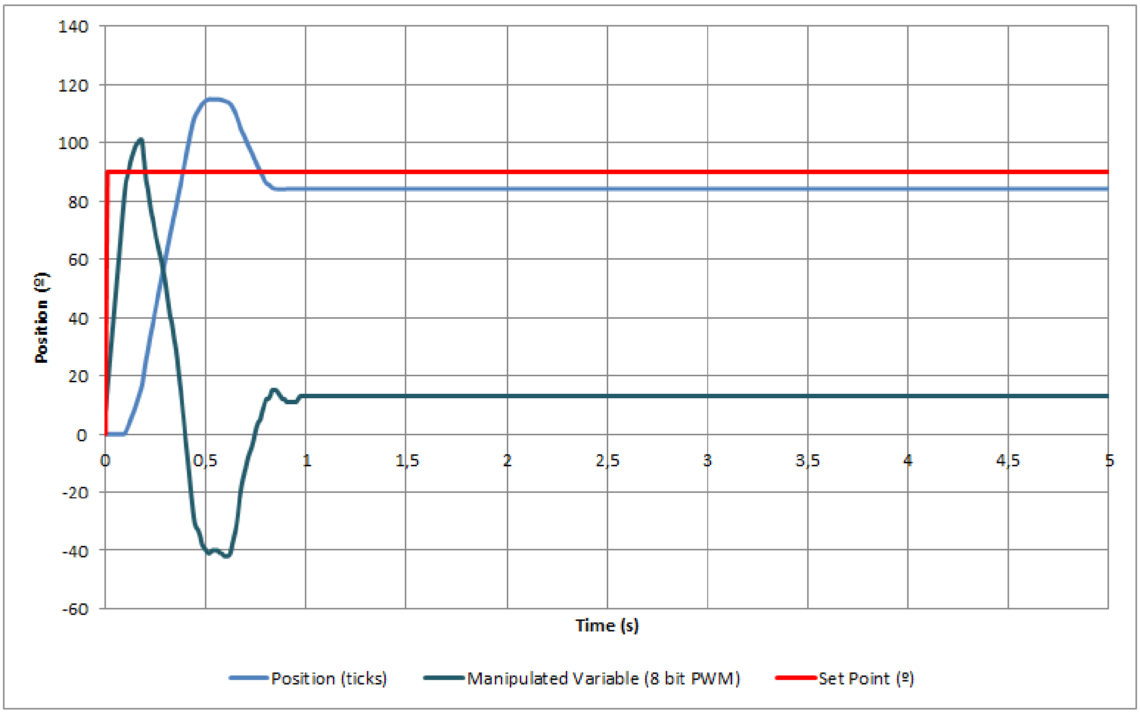


Figure 2.26: System's response to a set point of 90º (1024 ticks). Proportional gain is 4,7.

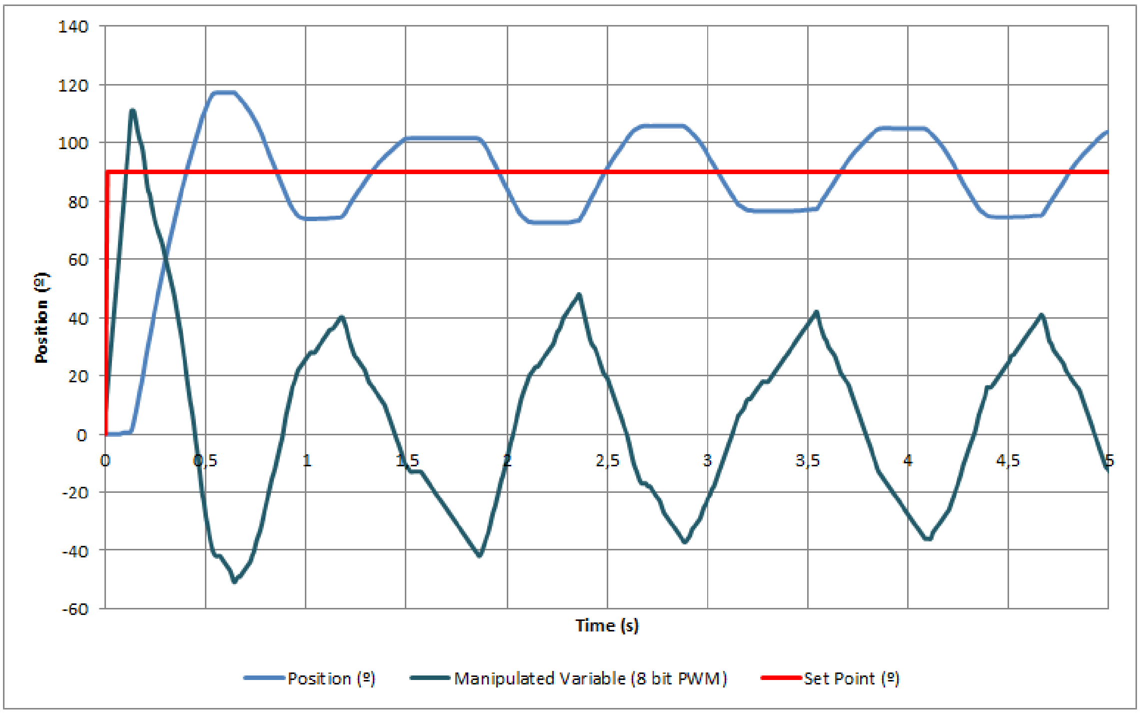


Figure 2.27: System's response to a set point of 90º (1024 ticks). Proportional gain is 4,8.

Controller's test

To gain simplicity, a proportional controller was tested first. If it did not provide satisfactory results (large static error or slow response), the PI and PID structures would have been tried.

Using Ziegler-Nichols table (Table 2.1), the proportional gain for the P controller is determined to be $K_P = 2,4$. This was the first controller tested. Its results are shown in Figures 2.28, 2.29, 2.30 and 2.31.

The results are quite satisfactory. Even though, a considerable error is made for the set point of 22,5º (about 20% of the set point) the controller, in general, performs a good task. Time response is fine as the plant does not need more than one second to reach the set point. Moreover, the rest of static errors are very low (under 5%). All of this results prove that the P controller for the orientation loop is enough.

2.5.3 Behaviour under windy conditions

Until this point, the plant was only tested in an ambience lacking of wind. It was time to check that the wind vane would measure the direction properly and, that the yaw tracker (the controller designed previously) would orientate the wind turbine's hub appropriately under windy conditions.

It is important to note that some changes have been made to the code in order to include the two rotary encoders (one for the wind vane and one for the wind turbine's hub). The encoders are read thanks to interrupt service routines that are executed at any line of the code when a certain circumstance happen (rise, fall or change of the voltage at a certain

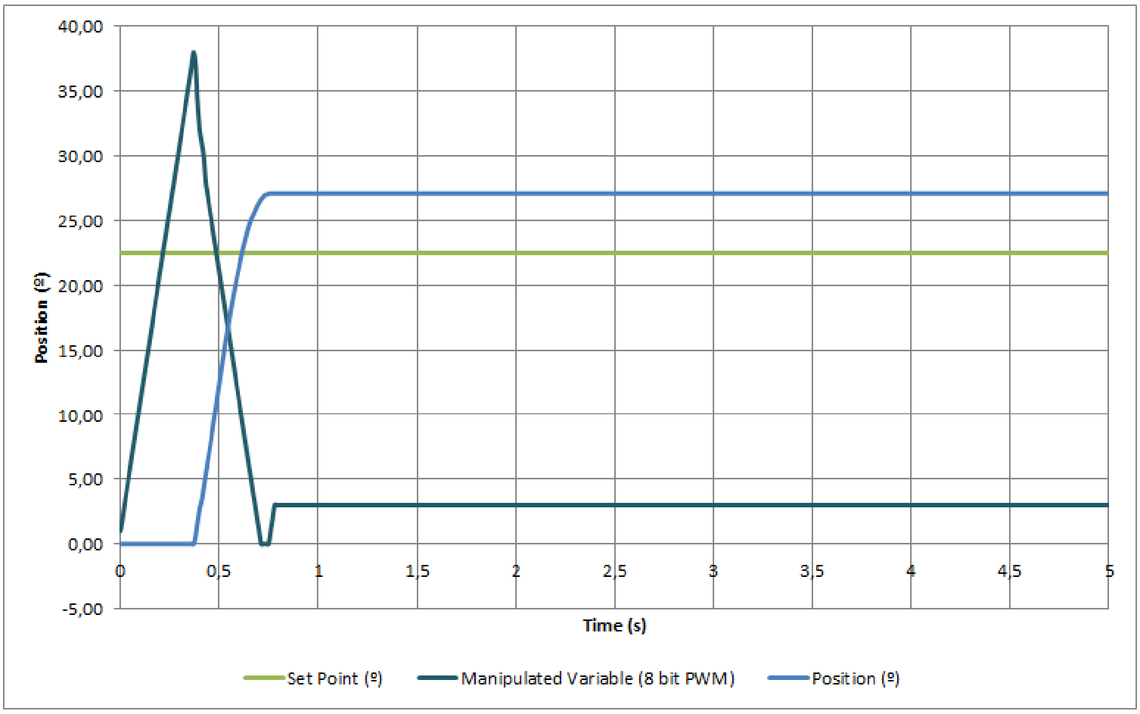


Figure 2.28: System's response to a position input of 256 ticks (22,5 º). Note that the static error corresponds to 5º.

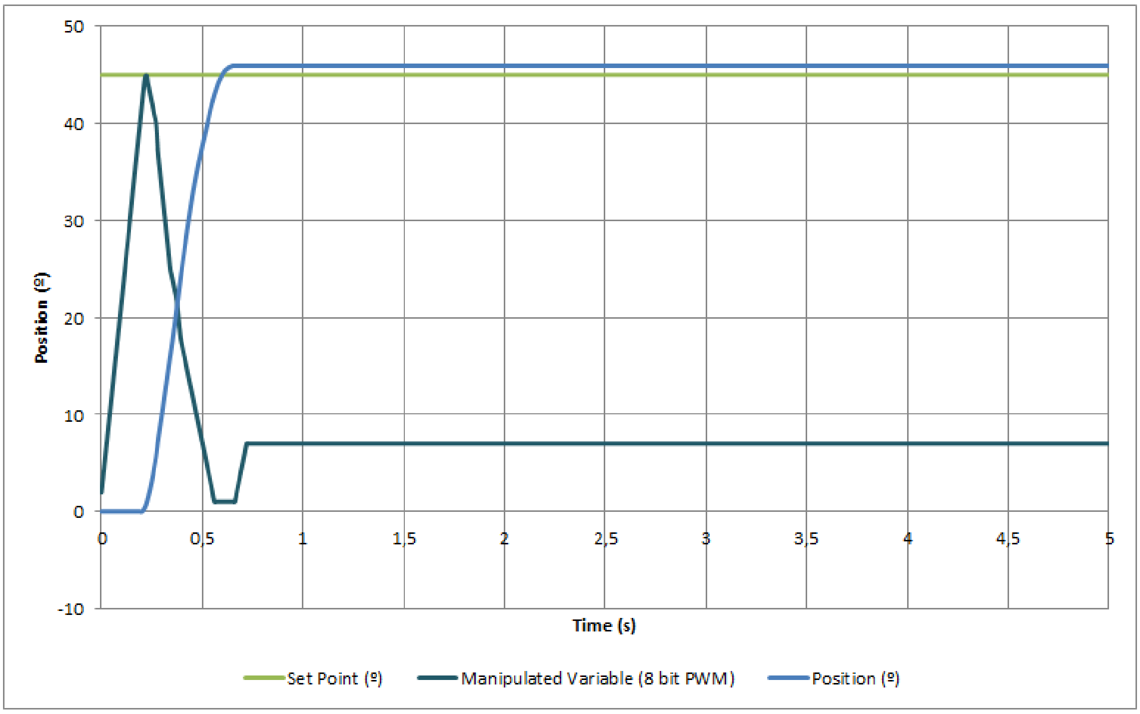


Figure 2.29: System's response to a position input of 512 ticks (45 º). Note that the static error is 1º.

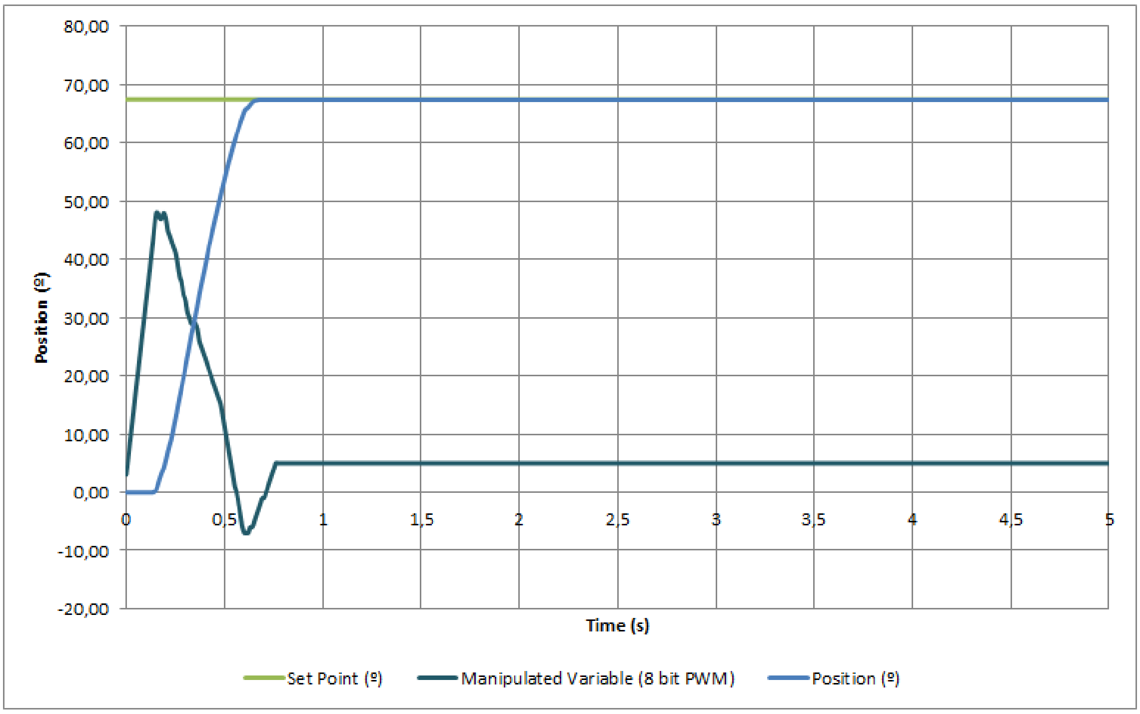


Figure 2.30: System's response to a position input of 768 ticks (67,5 º). Note that the static error corresponds to 0,1º.

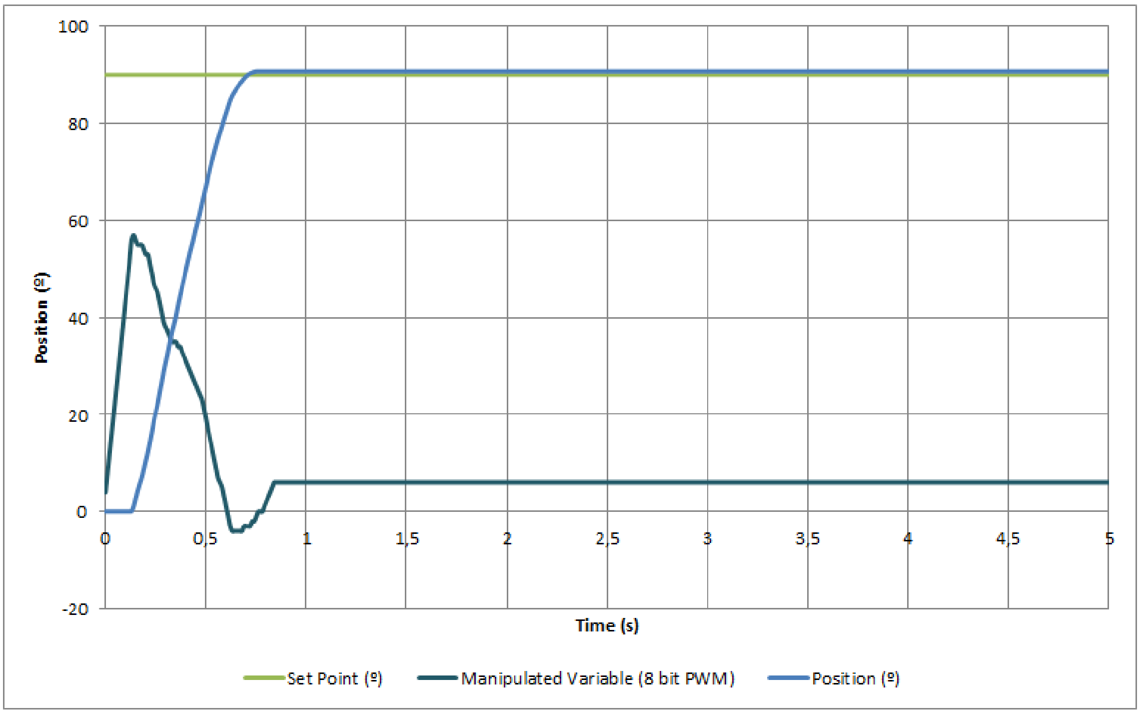


Figure 2.31: System's response to a position input of 1024 ticks (90 º). Note that the static error corresponds to 1º.

pin). The micro-controller used for the project (Arduino Uno rev3) only has two pins to where interrupts can be attached.

On the other hand, every encoder has two channels were information of the rotated angle is transmitted. In fact, a single channel would be enough to transmit this information. However, two channels are necessary if it is desired to determine the direction of rotation. Moreover, more precision can be gained if both channels are read.

For the previous measurements, only one encoder was used (the wind turbine's header one). So, the two Arduino Uno rev3 pins to where an interrupt routine can be attached were used to read the encoder. Nevertheless, at this moment, it was necessary to read two different encoders. For this reason, one encoder was connected to one of the two pins to where an interrupt can be attached.

This did not impede to determine the direction of rotation, but the precision was halved. It did not seem as a big problem because 2048 ticks or pulses per revolution still offer a good resolution (0,18°). This means that from this point, 2048 ticks mean a whole turn, whereas, before, 2048 ticks only meant a half rotation. The project's technical sheets can be consulted for more detail.

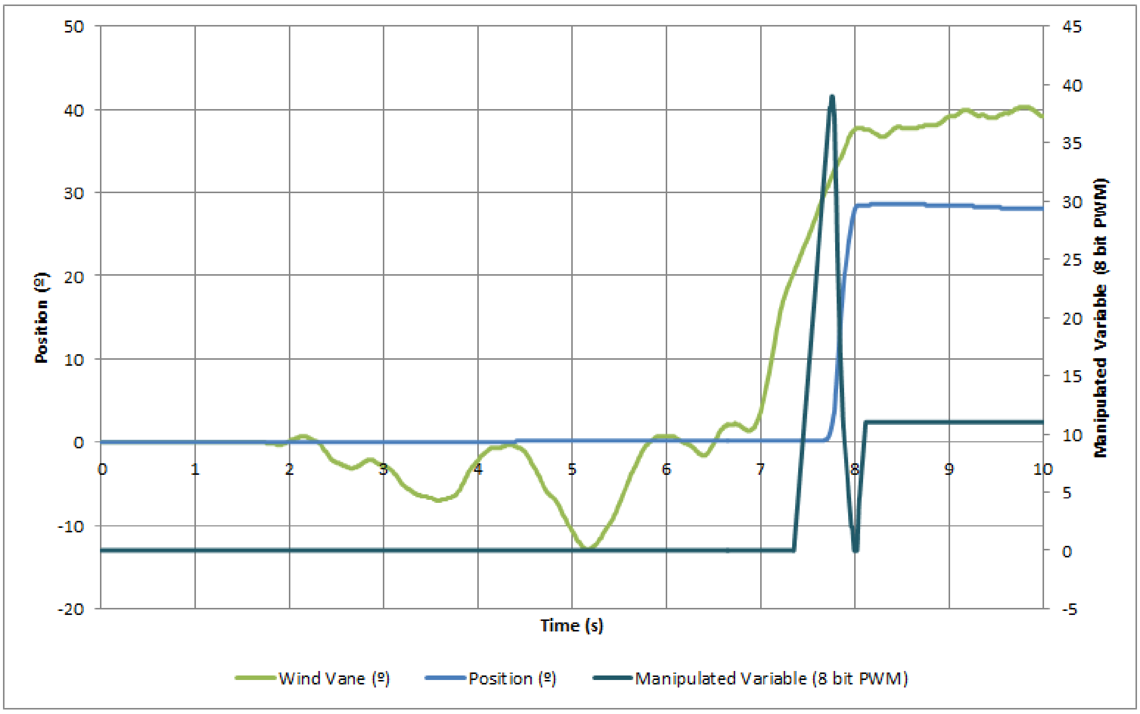


Figure 2.32: System's behaviour under windy conditions.

From system's response under windy conditions (Figure 2.32), it can be observed that the wind vane measurements can vary for a given wind direction. It is thought that turbulence should be the responsible for this.

To cope with this uncertainty, the angular position set point transmitted to the yaw tracker only varies if the change measured of wind direction is greater than 5°. This explains why the wind turbine's hub did not move during seconds 2 to 5.

The plant seemed to respond well to the set point when it was transmitted through the wind vane (see Figure 2.32). It took about 1 s to align with the wind direction. This is acceptable as very few energy is lost due to the non-alignment in one second.

On the other hand, a static error can be appreciated. It is about 60 ticks which correspond to 11º ($60\ ticks \cdot 360^{\circ}/2048\ ticks = 10,55^{\circ}$). However the error can not be neglected, it is acceptable because the loss of power for low yaw deviations (below 10º approximately) are very low (see Figure 1.2).

These results led to conclude that the controller designed was acceptable. Even though, it could have been enhanced to reduce the static errors and time to respond.

Chapter 3

Results summary

After obtaining the final plant layout and controller algorithm, the project's impact and possible enhancements have to be assessed. Finally, the project's conclusions are explained. To accomplish these tasks, this chapter has been structured as follows.

- Section 3.1: the project's cost is determined and it is studied how cheap or expensive it is with respect to other laboratory models.
- Section 3.2: the plant's life-cycle environmental impact is estimated. Special attention is put on the manufacture of each component as it is the main process from this point of view.
- Section 3.3: some security considerations are taken into account to check plant's safe operation.
- Section 3.4: some enhancements for the model are proposed along with their planning and some advices.
- Section 3.5: the results are summarised as well as project's possible evolutions.

3.1 Budget and economic feasibility

The first step when studying the project's economy is to determine its cost. Then, its feasibility can be estimated by comparing model's expenses with that of similar models in the market.

The project's budget is summarised in Table 3.1. It is breakdown in the different suppliers. Note that the manhour cost has not been incurred while realising the project. However, it has to be accounted if it is desired to keep developing the project. How each concept has been calculated is shown in the attached document Budget.

After that, project's feasibility was assessed by determining the savings it could generate when compared to an actual laboratory's model. To do so, the MV-541 model from Alecop [30] has been selected as it is the current work station used in university's laboratory to introduce students to the contol loop of a dc motor.

The acquisition cost of such model has been established from a university bill of the last time it was purchased (2010). To update that value to today's cost of money, the

BUDGET	2.777,15 €
HobbyKing	13,71 €
RS Amidata	47,70 €
Sparkfun	153,53 €
PC Componentes	28,00 €
Fablab	20,00 €
Ferreteria Jacquard	14,20 €
Manhours	2.500,00 €

Table 3.1: Project's budget. If further detail is desired, it is suggested to consult the attached document Budget.

consumer price index has been used. Reference [31] has determined that the increase of price from 2010 to 2016 is 5,3%.

Then, the costs due to the usage of both models are estimated as a 3% of their price because no more information was available. This 3% aims to account for the maintenance costs that are necessary to keep the model in a good state. No other expenses have been considered during the plant's usage. However, it has to be said that the maintenance cost is meant to include both materials and manpower necessary to accomplish this task.

With this information, the yearly savings produced thanks to the project's model with respect to the Alecop model are determined. The money saved thanks to the project's model are shown in Tables 3.2 and 3.3.

Year	0	1	2
Alecop MV-541			
Expenses	5.251,96 €	157,56 €	157,56 €
Project's model			
Expenses	2.777,15 €	83,31 €	83,31 €
Yearly saving	2.474,82 €	74,24 €	74,24 €
Accumulated saving	2.474,82 €	2.549,06 €	2.623,30 €
Discount rate	6%	6%	6%
Updated yearly saving	2.474,82 €	70,04 €	66,08 €
Acc. & updated saving	2.474,82 €	2.544,86 €	2.610,93 €

Table 3.2: Feasibility study intermediate results for years 0 to 2.

The feasibility parameter used to characterise the project is the payback time. This is the time required to save the same amount of money than the project's model costs. It was determined thanks to the following procedure. The variables used are listed below.

- $C = 2.777,15$ €: acquisition cost.
- $S_0 = 2.777,15$ €: the initial saving thanks to the difference in purchase price of the two models.
- $S = 83,31$ €/year: yearly saving thanks to the difference in the maintenance costs of the two models.

Year	3	4	5
Alecop MV-541			
Expenses	157,56 €	157,56 €	157,56 €
Project's model			
Expenses	83,31 €	83,31 €	83,31 €
Yearly saving	74,24 €	74,24 €	74,24 €
Accumulated saving	2.697,55 €	2.771,79 €	2.846,04 €
Discount rate	6%	6%	6%
Updated yearly saving	62,34 €	58,81 €	55,48 €
Acc. & updated saving	2.673,27 €	2.732,08 €	2.787,56 €

Table 3.3: Feasibility study intermediate results for years 3 to 5.

- n: number of years.
- $r = 1/(1+d) = 0,94$: yearly money value reason of decrease. Note that d is the discount rate which has been set to 6%. This is a moderated value as it is not intended to make profit from the investment, just save money.

It is desired to determine the number of years that make the accumulated and updated savings equal to the initial cost of the project's model. This is expressed in Equation 3.1.

$$S_0 + \sum_{i=1}^{n} \frac{S}{(1+d)^i} = C \rightarrow S_0 + \sum_{i=1}^{n} S r^i = C \tag{3.1}$$

Note that the accumulated and updated savings is a geometric series. Thus, the summation can be converted into a product by virtue of Equation 3.2 (for a proof of this result, Reference [32] can be consulted).

$$\sum_{i=1}^{n} S r^i = S \left(\frac{1 - r^{n+1}}{1 - r} - 1 \right) \tag{3.2}$$

Including Equation 3.2 result into Equation 3.1, after some manipulations, lead the final form shown in Equation 3.5.

$$S_0 + S \left(\frac{1 - r^{n+1}}{1 - r} - 1 \right) = C \rightarrow \frac{1 - r^{n+1}}{1 - r} = \frac{C - S_0}{S} + 1 \tag{3.3}$$

$$r^{n+1} = 1 - \left(\frac{C - S_0}{S} + 1 \right) (1 - r) \tag{3.4}$$

$$n = 1 - \frac{\ln \left[1 - \left(\frac{C - S_0}{S} + 1 \right) (1 - r) \right]}{\ln (r)} \tag{3.5}$$

For the project's values, Equation 3.5 lead to determine a payback time of 4,81 years. This is a very good time considering that Arduino is a raising technology. It is expected that its usage will be even more widespread in the future. Considering technology evolution, it seems that in 5 years, the model will keep updated, so the investment can be recovered.

3.2 Environmental impact

To determine the project's environmental impact, its whole life-cycle should be studied. This includes extraction of the raw materials, their processing, manufacturing, distribution, use and disposal. However, all this information requires a lot of time and effort to be gathered in order to have a precise estimation. For this reason, a more qualitative rather than quantitative study has been carried out.

The model's components suppliers were analysed to have an idea of the environmental impact during the life-cycle stages before usage. Then, its usage environmental impact was compared with competitors. Finally, some actions are proposed for the an eco-friendly disposal of the model.

The main suppliers of the project's components are described below from an ecological perspective. In this manner, it is intended to estimate the environmental impact of the materials extraction and manufacturing of project's components.

- TRACO Electronic AG [33]. This company provided the power supply used to feed the motors. It is based in Switzerland and no information appears at first sight on their web page about their environmental concern. However, the company is certified to ISO 14000, which deals with the environmental impact of company's activities [34].
- MFA/Como Drills [35]. This is England based company manufactures the motor used to move the propeller for the "fan" tower. It does not seem to be certified to any norm ensuring environmental manage. However, as it is located in the United Kingdom, its legislation ensures that a minimum respect to the environment is kept.
- ServoCity [36]. This was the provider of mechanic components (pinion gears, shaft couplers and hub). It also supplied the gear motor used to orientate wind turbine's hub. No information appears in its web page about environmental manage. As it is located in the United States, the country's regulation implies a minimum aware of ecological implications. Even though, it can be said that US law is lighter than other European countries.
- Yueqing Ying's YUMO ELECTRIC Co., Ltd. This company manufactured the rotary encoders used to measure the wind direction and the wind turbine's hub orientation. It is based in the Chinese town of Liushi. No information appears in its web page about environmental concern. It only cites sustainable development. China is one of the most pollutant countries in the world partly due to lack of regulation. However, some improvements are being made on their laws to achieve more eco-friendly practices [37].
- HobbyKing [38]. This was the provider of the propellers used to generated wind with the fan and to represent a wind turbine. They also supplied the motor driver. No information is available on their environmental concern at their web page.
- Arduino [39]. This company manufactures the micro-controller used in the project. The controller used for the plant (Arduino UNO rev3) was made in Italy. As no infor-

mation is provided on complying to environmental certifications, the only standard ensuring ecological management is Italian law.

After this overview of the main suppliers, it can be said that project's environmental impact from the materials extraction and manufacturing point of view is close to the average in the electronics industry. Some components are supplied by providers that are very concerned of their ecological management. However, some other companies are located in countries where it seems that sometimes the environment is left in a secondary plane to increase wealth. Moreover, some companies do not prove explicitly to be concerned about their ecological implications. This variety of suppliers results in an environmental impact similar to most of the average electronics equipment.

When using the model, the only footprint left on the environment is due to the energy consumed by the motors, sensors, controller and computer. As it is expected that the energy produced will reduce its carbon emissions [40], the environmental impact of model usage will also decrease. Moreover, in a possible evolution of the project, the wind turbine could generate electrical energy. In this manner, the energy consumed from the network and the environmental impact would be reduced.

When it comes to the model's disposal, it has to be said that the majority of its components could be reused. As it is a very transparent implementation where all components can be seen, they can be easily removed in order to be used in a different place. Rotary encoders, motor driver, dc motors and a micro-controller can found rapidly a different application apart from the plant if it has to be dismantled. These applications could be a robotic arm or a position control system for a 3D printer header. The rest of components that are mainly mechanic (threaded bars, nuts, washers, would beams, bearings and pinion gears) can be reused for any application as they are very common, thus usually needed.

3.3 Security study

To check projects secure operation, all its components will be revised putting special attention on the possible failures.

- Power supply [41]. If too much current is demanded, some internal components could burn it. However, the power supply can deliver up to 4,2 A which is close to the double that motors can require to operate (0,77 A for the fan's motor and 1,6 A for the gear motor in charge of tracking wind direction).
- Motor driver [42]. Two possible failures are identified for this device. Too much heat can accumulated because of chopping the current from power supply to motors' pins in order to administrate the power as required by the micro-controller. To prevent this situation, the driver is equipped with a large heat sink that can significantly reduce the temperature. The other possible failure is that counter electromotive force is generated at the motors due to an input of rotational energy at their shafts. In this case, the driver is protected with diodes. If not, current flowing in the opposite than expected direction could damage the power supply, the motor driver or the micro-controller.
- Micro-controller. The main possible failure for this component is that some short-circuit could happen when reading a digital input without a pull-up or pull-down resistor. However, Arduino has some pull-up and pull-down resitors built in to ensure

that this does not happen. It is only required to include some lines in the code instead of making up the whole circuit.

- Rotary encoder. If the positive and ground wires were not connected properly, it could be damaged. To avoid this situation, the wirings are detailed in the project's technical sheets. Moreover, they are also reminded in the Arduino code.
- DC motors. If an obstacle opposites motor's spin, it could request for more power to the power supply. If it receives too much power, its inner components could burn. This situation has not been considered in the project's development. A current limiting device could be included to avoid this failure mode.
- Propeller. The fan's propeller can damage someone as it spins very fast. Indeed, so much thrust is generated that an auxiliary bar has been needed to counteract the torque that is generated at the tower's base. This avoids the tower to back flip. Moreover, if the propeller starts to spin in an unexpected situation, it could cut someone's hand. For this reason, the Arduino code always asks the user for a signal when he wants to start.

3.4 Planning of the next step

The project can evolve in many different ways to improve the model. Even though the next enhancements have been identified, some more could appear to be interesting.

- Include electricity generation by connecting the rotor shaft to a dc motor. Reference [43] can be consulted to know how to determine the expected output of power from the motor.
- Add a current limiting device to prevent burning the motors if too much current is demanded in case of encountering an obstacle. A hall-effect current sensor, like the one from Reference [44], could be used. It would be read through the micro-controller. If too much current flow is detected, the micro-controller would disconnect the motor from power supply through the driver. Other alternatives for protecting the motor can be found in Reference [45].
- Design the position control loop through an analytical method. The poles placement or zeros cancellations techniques could be tried to obtain a faster and more precise control loop for the position.

Whatever improvement is decided to add, the procedure followed to correctly implement should be the same. It is described below.

1. Study of possible solutions. It is important to know what possibilities are offered thanks to nowadays technology. Then, all the options should be studied and, thanks to some criteria, a final decision should be made. This avoids to redefine the solution when implementing it.
2. Design of the chosen solution. This phase aims to determine all the components and their assembly. It is very important to be very detailed and try to account for all considerations in order to make the implementation quite straightforward. The final result of this phase should be the implementation instructions to make the next step easier and avoid some unexpected problems (like sensors not being enough precise or a motor not having sufficient power to drive the load it is required to).

3. Arrival of the components. At this moment the project is paused. The components order should be done once the detailed design is closed. If not, some components could be ordered that afterwards would never be used if an error was incurred at the design phase. For this reason, it is important to revise more than once the design results.

4. Implementation of the chosen solutions. When the components start to arrive, they should be tested. In this way, a possible manufacturing defect can easily be detected. Then, different components should be assembled little by little. It is important no to couple all components at once. If some error happens, it could be difficult to find out what element failed.

5. Verification of the requirements. It should be checked whether the project improvements are fulfilling their task. If not, it should be tried to explain why the solution is not working well. If time allows it, some changes could be made in order to obtain the desired result. If not, some guidance should be left for a new alternative achieving the searched project's improvement.

6. Results summary. It is a good practice to summarise the project's experience in a report. In this way, some incurred errors could be avoided and some advices could be given.

For the seen model's improvements, a Gantt chart has been made assigning a time duration to each of the aforementioned project stages (see Figure 3.1). Note that the two first ones (possible solutions and detailed design) require more time because they are very important in the success of the next phases.

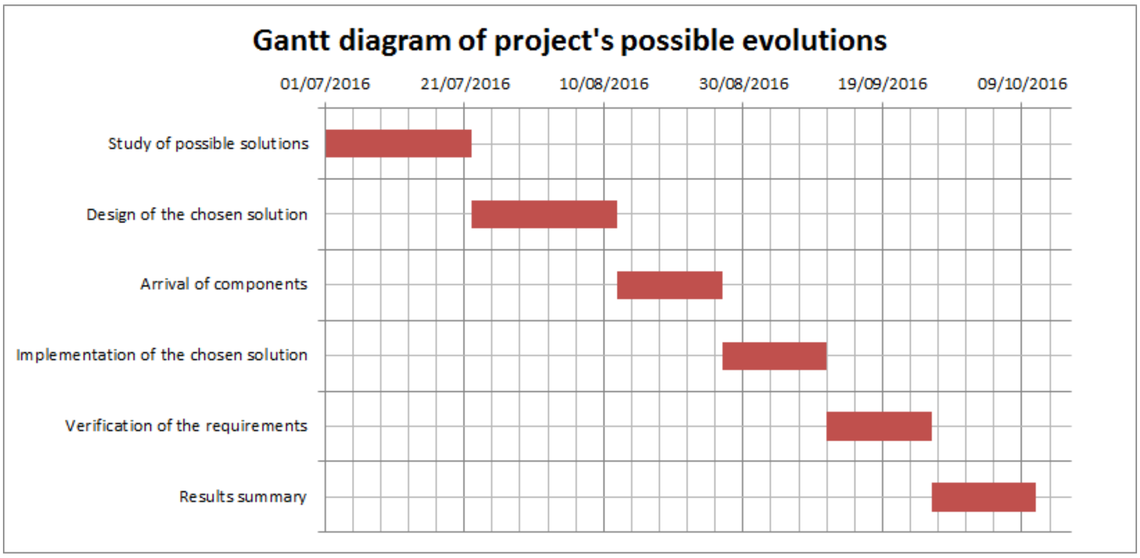


Figure 3.1: Gantt chart for the possible plant's improvements.

3.5 Conclusions

It is time to revise project's requirements and scope to check whether they have been achieved or not. Even though some work packages have not been completed (no ramp test have been carried out neither at experimental modelling nor at controller verification), it can be said that the project's scope has been completed to nearly its 80%.

On the other hand, the requirements have been fulfilled except from project's cost. In terms of control, the project has achieved the desired results. The wind turbine's hub orientates itself in a range even larger than 90º and it requires no more than one second. However, the project's cost has been largely surpassed. This is mainly because manpower costs were not accounted when writting down the requirements. If only the component's acquisition cost is considered, the project's cost is only exceed by 50 €, which is a quite acceptable error.

From the economical point of view, if the model is compared to other work stations currently used at university, it seems to provide a good alternative. It can fulfil more or less the same needs and it is cheaper.

When it comes about its environmental impact, it can be concluded that the previous stages to usage of the plant's are neither very harmful nor very friendly with the environment. It is on the average of the electronics and robotics industry. On the other hand, the plant's usage as well as its disposal can have a low impact on the ecology. Not much energy is consumed when using the model and its components can be easily reused.

Moreover, the plant security study results show that, in general, its operation is quite safe. However, a failure mode has been detected for the motors that was not taken into account when designing it. It does not seem a probable failure. Nevertheless, it can be fixed by using a current limiter device or current sensors.

Finally, some possible evolutions to improve the model have been identified: adding a protection to the motor from a current over their maximum value, tuning the position control loop analytically and generating electricity at the wind turbine's hub thanks to a DC motor. The planning of this improvements has been suggested and some advices have been given.

Bibliography

[1] E. Bossanyi, “Current perspective on wind turbine control,” in *Workshop on Sustainable Control of Offshore Wind Turbines*, University of Hull, September 2012.

[2] UNFCC, “Beginner’s Guide to the Convention.” `http://unfccc.int/essential_background/background_publications_htmlpdf/items/1661.php`.

[3] C. of the Parties, “Adoption of the paris agreement,” 2015.

[4] SmartGrid.gov, “What is the smart grid?.” `https://www.smartgrid.gov/the_smart_grid/smart_grid.html`.

[5] T. Burton, *Wind Energy Handbook*. Wiley and Sons, 2001.

[6] F. Candelas, G. García, S. Puente, J. Pomares, C. Jara, J. Pérez, D. Mira, and F. Torres, “Experiences on using arduino for laboratory experiments of automatic control and robotics,” *IFAC-PapersOnLine*, vol. 48, no. 29, pp. 105–110, 2015.

[7] C. I. Castro, M, *Energía Eólica*. Progensa, 2001.

[8] M. Villarrubia, *Energía Eólica*. CEAC, 2001.

[9] E. W. E. Association, *Wind Energy - The Facts*. Earthscan, 2009.

[10] H. Li, C. Yang, Y. Hu, X. Liao, Z. Zeng, and C. Zhe, “An improved reduced-order model of an electric pitch drive system for wind turbine control system design and simulation,” *Renewable Energy*, vol. 93, pp. 188–200, 2016.

[11] Z. Wu and H. Wang, “Research on active yaw mechanism of small wind turbines,” *Energy Procedia*, vol. 16, no. Part A, pp. 53–57, 2012.

[12] A. Soriano, L. Marín, M. Vallés, A. Valera, and P. Albertos, “Low cost platform for automatic control education based on open hardware,” *IFAC Proceedings Volumes*, vol. 47, no. 23, pp. 9044–9050, 2014.

[13] E. Zouganeli, V. Tyssø, B. Feng, K. Arnesen, and N. Kapetanovic, “Project-based learning in programming classes – the effect of open project scope on student motivation and learning outcome,” *IFAC Proceedings Volumes*, vol. 47, no. 3, p. 12232–12236, 2014.

[14] n. Hoyo, J. L. Guzmán, J. C. Moreno, and M. Berenguel, “Teaching control engineering concepts using open source tools on a raspberry pi board,” *IFAC-PapersOnLine*, vol. 48, no. 29, pp. 99–104, 2015.

[15] Wikipedia, “List of scale model sizes.” `https://en.wikipedia.org/wiki/List_of_scale_model_sizes`.

BIBLIOGRAPHY

[16] HobbyKing, "Multi-Rotors, Drones & Parts > Propellers." http://www.hobbyking.com/hobbyking/store/__1233__501__Multi_Rotors_Drones_Parts-Propellers.html.

[17] T. Magnetica, "Why Are Wind Turbines Getting Bigger?." http://www.terramagnetica.com/2009/08/01/why-are-wind-turbines-getting-bigger/.

[18] HobbyKing, "GWS Style Slowfly Propeller 14x4.7 Grey (CW) (2pcs)." http://www.hobbyking.com/hobbyking/store/__28289__GWS_Style_Slowfly_Propeller_14x4_7_Grey_CW_2pcs_.html.

[19] SparkFun, "Shaft Coupler - 1/4" to 3 mm." https://www.sparkfun.com/products/12412.

[20] SparkFun, "Set Screw Hub - 1/4" Bore." https://www.sparkfun.com/products/12488.

[21] SparkFun, "Gear - Pinion Gear (20T; 0.25" Bore)." https://www.sparkfun.com/products/12529.

[22] R. Components, "Rodamiento de bolas RS Pro, Miniatura, 626-2Z." http://es.rs-online.com/web/p/rodamientos-de-bola/6190014/.

[23] R. Components, "Motor de dc Como Drills 719RE360, 6 - 15 V dc, 10668 rpm, 5,75 W, 0,77 A." http://es.rs-online.com/web/p/motores-dc/2389715/.

[24] SparkFun, "Micro Gearmotor - 90 RPM (6-12V)." https://www.sparkfun.com/products/12285.

[25] HobbyKing, "Kingduino Compatible H-Bridge Motor Driver." http://www.hobbyking.com/hobbyking/store/__37421__Kingduino_Compatible_H_Bridge_Motor_Driver.html.

[26] SparkFun, "Rotary Encoder - 1024 P/R (Quadrature)." https://www.sparkfun.com/products/11102.

[27] Arduino, "Arduino - Float." https://www.arduino.cc/en/Reference/Float.

[28] K. Ogata, *Ingeniería de control moderna*. Pearson, 2010.

[29] A. Massip and J. Quevedo, *Apunts Bàsics de Control Automàtic*. ESAII - UPC, 2016.

[30] Alecop, "Sistemas de regulación y control: Estudio de procesos de control de velocidad y posición Serie 540." http://www.alecop.com/equipment/index.php/es/equipamiento-didactico/sistemas-de-regulacion-y-control-es/estudio-de-procesos-de-control-de-velocidad-y-posicion-serie-540-detail.

[31] I. N. de Estadística, "Cálculo de variaciones del Indice de Precios de Consumo (sistema IPC base 2011)." http://www.ine.es/varipc/verVariaciones.do?idmesini=10&anyoini=2010&idmesfin=5&anyofin=2016&ntipo=1&enviar=Calcular.

[32] Wikipedia, "Serie Geométrica." https://es.wikipedia.org/wiki/Serie_geom%C3%A9trica.

[33] TracoPower, "About|tracopower.com." http://www.tracopower.com/about/.

[34] ISO, "ISO 14000 - Environmental management." http://www.iso.org/iso/home/standards/management-standards/iso14000.htm.

BIBLIOGRAPHY

[35] MFA/ComoDrills, "MFACOMODRILLS.COM." http://www.mfacomodrills.com/profile.html.

[36] ServoCity, "About Us." https://www.servocity.com/html/about_us.html.

[37] C. Gracie, "China struggles to contain the environmental damage of its rapid growth." http://www.bbc.com/news/world-asia-china-33546231.

[38] HobbyKing, "AboutUs." http://www.hobbyking.com/hobbyking/store/uh_aboutUs.asp.

[39] Arduino, "Arduino - AboutUs." https://www.arduino.cc/en/Main/AboutUs.

[40] I. E. Association, "World Energy Outlook 2015." http://www.iea.org/publications/freepublications/publication/WEO2015ES_SPANISH.pdf.

[41] R. Components, "Fuente de alimentación conmutada integrada (SMPS), 50W, 1 salida, Tensión 12V dc, Corriente 4.2A." http://es.rs-online.com/web/p/fuentes-de-alimentacion-de-modo-conmutado-smps-integradas/7703298/.

[42] HobbyKing, "Dual H-Bridge Motor Driver." http://www.hobbyking.com/hobbyking/store/uploads/399423573X19082X47.pdf.

[43] O. Lathrop, "DC Motor as a Generator Voltage and Current." http://electronics.stackexchange.com/questions/152171/dc-motor-as-a-generator-voltage-and-current.

[44] SparkFun, "SparkFun Hall-Effect Current Sensor Breakout - ACS712." https://www.sparkfun.com/products/8882.

[45] D. W. Jones, "Current Limiting for Stepping Motors." http://homepage.cs.uiowa.edu/~jones/step/current.html.