



وزارة البحث العلمي والتعليم العالي.
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique



جامعة عبد الحميد بن باديس مستغانم
Université Abdelhamid Ibn Badis Mostaganem
كلية العلوم والتكنولوجيا
Faculté des Sciences et de la Technologie
Département de Génie Électrique

N° d'ordre : M...../ GE /2026

MEMOIRE
Pour obtenir le diplôme de
MASTER EN ELECTROTHERMIQUE
Spécialité : Electrotechnique Industrielle

Présenté par :

Moussaoui iness

Thème

Conception et construction d'un banc d'essai didactique
pour les régulateurs P, PI et PID

Soutenu le 28/ 06/ 2026, devant le jury composé de :

Président	ADDA Ahmed	MCB	Université de Mostaganem
Examineur	BENYAMINA Mansour	MCA	Université de Mostaganem
Encadrant	BENZIDANE Mohammed Ridha	MCB	Université de Mostaganem
Co-Encadrant	MESKINE Said	Pr	Université de Mostaganem

Année Universitaire 2025-2026



Dedication

We often think that success is measured by money or fame, I believe that success is; the glow forming in the eyes of those who believed in me, even when I was feeling low and left behind.

Hanse, we should be grateful to all of those who stood by our side and gave us strength to continue in our path.

Just like any living creature on this earth, none of us would bloom without god's will; so first of all, I thank God for brightening my view and helping me.

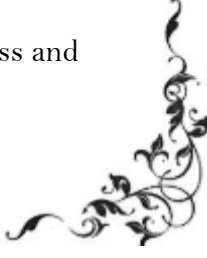
In this happy day I would've loved to have you sitting there, next to my mother and smiling at me with pride, I would've loved for you to clap by the end of my presentation. This is for you father; I hope my success is reaching your evergreen soul.

To the woman who never stopped having faith in me, to the woman who never stopped fighting for me, to the woman who gave me all the love and support I needed to pursue my dream, to my queen and dear mother Hadjira, thank you for all your deeds, I only wish to return back half of what you gave me.

To my dearest brother thamer , my two sisters Amira and Insaf, and my little Minou . Thank you for being the pillars holding me together when times were tough; thank you for your unconditional love and care.

I would like to express my sincere gratitude to Dr. Benzidane and Prof. Meskine for their guidance, support, and mentorship. Their encouragement and dedication made this work possible, and without them, this achievement would not have become a reality.

Last but never the least, big appreciations to my best friends iness and haroun . You are the truest.



Declaration

The idea, concept, design, development, fabrication, and implementation of the prototype presented in this project were conceived and carried out within the ECP3M Laboratory. The laboratory played a fundamental role throughout all stages of the project, from the initial idea and technical planning to the development, assembly, testing, and validation of the prototype.

Furthermore, all components, materials, equipment, tools, and technical resources required for the realization of this work were provided by the ECP3M Laboratory. The laboratory's scientific guidance, technical expertise, and continuous support were essential to the successful completion of this project.

This work stands as a result of the resources, facilities, and collaborative environment offered by the ECP3M Laboratory, without which the development and realization of this prototype would not have been possible.

ABSTRACT:

This project was focused on designing and developing a didactic platform for students to learn about PID control systems, using the ESP32 microcontroller. The system consists of an HC-SR04 ultrasonic distance sensor, a brushless direct current (BLDC) motor with an electronic speed controller (ESC), and a web-based user interface for real-time monitoring and controlling the operation of the motor.

The experimental evaluation of three different modes of control (P, PI, and PID) provided an analysis of how each of these types of control affects the performance of the system. The results of this work demonstrated that the PI controller yielded the best compromise of accuracy, stability, and response time.

The project also included an entrepreneurial component in the form of a Business Model Canvas (BMC) and Business Plan (BP), demonstrating the potential for this product to be commercially available as an educational solution for colleges and trade schools.

Keywords: PID controller; ESP32 microcontroller; automated control; didactic platform; Embedded Systems; Real Time Monitoring.

ملخص

لتعلم أنظمة (PID) ركز هذا المشروع على تصميم وتطوير منصة تعليمية لطلاب الهندسة الكهربائية والإلكترونية يتكون النظام من مستشعر مسافة ESP32 ، باستخدام وحدة التحكم الدقيقة (PID) التحكم التناسبي التكامل التفاضلي مزود بوحدة تحكم إلكترونية في السرعة (BLDC) ، ومحرك تيار مستمر بدون فرش HC-SR04 فوق صوتي ، وواجهة مستخدم عبر الإنترنت لمراقبة تشغيل المحرك والتحكم فيه في الوقت الفعلي (ESC) ، تحليلًا لكيفية تأثير كل نمط منها على أداء النظام. (PID ، PI ، P) أتاح التقييم التجريبي لثلاثة أنماط تحكم مختلفة. حققت أفضل توازن بين الدقة والاستقرار وسرعة الاستجابة PI وأظهرت نتائج هذا العمل أن وحدة التحكم ، مما يبرز إمكانية طرح (BP) وخطة عمل (BMC) كما تضمن المشروع جانبًا رياديًا في شكل نموذج عمل تجاري. هذا المنتج تجاريًا كحل تعليمي للكليات والمعاهد المهنية ؛ تحكم آلي ؛ منصة تعليمية ؛ أنظمة مضمنة المراقبة ESP32 ؛ وحدة تحكم دقيقة PID الكلمات المفتاحية: وحدة تحكم الأنظمة.

RÉSUMÉ

Ce projet visait à concevoir et développer une plateforme didactique permettant aux étudiants d'apprendre les systèmes de contrôle PID à l'aide du microcontrôleur ESP32. Le système comprend un capteur de distance ultrasonique HC-SR04, un moteur à courant continu sans balais (BLDC) avec un contrôleur de vitesse électronique (ESC) et une interface utilisateur web pour la surveillance et le contrôle en temps réel du fonctionnement du moteur.

L'évaluation expérimentale de trois modes de contrôle différents (P, PI et PID) a permis d'analyser l'influence de chacun sur les performances du système. Les résultats ont démontré que le contrôleur PI offrait le meilleur compromis entre précision, stabilité et temps de réponse.

Le projet comportait également un volet entrepreneurial, sous la forme d'un Business Model Canvas (BMC) et d'un Business Plan (BP), démontrant le potentiel de commercialisation de ce produit en tant que solution pédagogique pour les établissements d'enseignement supérieur et les écoles professionnelles.

Mots-clés : Contrôleur PID ; Microcontrôleur ESP32 ; Contrôle automatisé ; Plateforme didactique ; Systèmes embarqués ; Surveillance en temps réel.

Summary:

general Introduction	13
----------------------------	----

Chapter I: Fundamentals of Automatic Control and PID Regulation

I-1 introduction.....	16
I-2 Servo-control and Regulation Reminders.....	16
I-2-1 Tracking Servo-control.....	16
I-2-2 Regulation.....	17
I-2-2-1 objectives of regulation.....	18
I-2-2-2 Regulation Typology.....	19
a. Open-loop regulation.....	19
b. Closed-loop control.....	20
c. Suitability of Closed-Loop Regulation for the Proposed System.....	21
I-2-2-3 General Regulation Principle.....	21
I-2-2-4 On-Off Regulation (Tout Our Rein, or TOR)	22
I-3 Study of Industrial PID Controllers.....	23
I-3-1 Analog Control.....	24
I-3-1-1 PID Actions.....	24
a-Proportional Action.....	24
b-integral action.....	25
c-Derivative Action.....	25
d-Integral and Proportional Actions (PI).....	26
e- Derivative and Proportional Action (PD).....	26
f- Proportional, Integral, and Derivative Actions (PID).....	26
g- Comparative Analysis of P, PI and PID Controllers.....	27
I-3-1-2 Different PID Controller Structures.....	27

I-4 Performance Criteria of Control Systems.....	28
I-4-1 rise time.....	28
I-4-2 overshoot.....	29
I-4-3 Settling time, t_s	29
I-4-4 steady-state error.....	30
I-4-5 stability.....	31
I-5 Digital Control in Embedded Systems.....	31
I-6 conclusion.....	33

Chapter II: Design and Development of an ESP32-Based PID Experimental Bench

II-1 Introduction.....	35
II-2 System Specifications and Requirements.....	36
II-2-1 Project Objectives.....	36
II-2-2 Functional Requirements.....	36
II-2-3 Technical constraints.....	37
II-3 System Architecture	37
II-3-1 Simplified Block Diagram.....	38
II-3-2 Description of the Integrated Feedback Loop.....	38
II-4 Hardware Selection and Specifications.....	39
II-4-1 Microcontroller Unit: ESP32.....	39
II-4-2 Ultrasonic Sensor: HC-SR04.....	40
II-4-3 Actuation System: (ESC and Brushless Motor)	41
II-4-4 Battery Power Supply.....	43
II-5 Software Design and PID Implementation.....	44
II-5-1 Development Environment.....	45

II-5-2 PID Control Algorithm.....	45
II-5-3 Program Structure.....	46
II-5-4 Control Modes (P, PI, PID)	46
II-6 Mechanical Assembly and Integration.....	46
II-6-1 Electrical Wiring and Connections.....	46
II-6-2 Mechanical Assembly and 3D-Printed Structure.....	48
II-6-3 Final Prototype Assembly.....	49
II-7 User Interface and Real-Time Monitoring.....	50
II-7-1 Web Interface Design.....	50
II-7-2 Real-Time Data Monitoring.....	51
II-7-3 Educational Contribution of the Interface.....	51
II-8 Safety Considerations and System Limitations.....	52
II-9 Conclusion.....	54

Chapter III: Experimental Validation and Performance Analysis

III-1 Introduction.....	56
III-2 Experimental Conditions and Testing Procedure.....	56
III-2-1 Experimental Setup.....	56
III-2-2 Web-Based Monitoring and Control Interface.....	57
III-2-3 Test Procedure.....	59
III-2-4 Performance Evaluation Criteria.....	59
III-3 System Architecture and Software Design.....	60
III-3-1 Understanding the System: A Modular Organigram.....	60
III-3-2 Hardware Pin Configuration.....	61
III-3-3 Wireless Communication and Embedded Web Server.....	62
III-3-4 Trajectory Planning and Waypoint Management.....	63

III-3-5 Automatic Arming Sequence.....	64
III-3-6 Automatic Soft Landing Mechanism.....	65
III-3-7 Automatic Mission Completion and Auto-Abort.....	66
III-3-8 Emergency Ceiling Protection.....	66
III-3-9 LED-Based Visual Status Indication.....	67
III-3-10 Telemetry Tracking: Mathematical Performance Metrics.	67
III-4 Performance Evaluation Criteria.....	69
III-4-1 Rise Time.....	69
III-4-2 Overshoot.....	69
III-4-3 Settling Time.....	69
III-4-4 Steady-State Error.....	69
III-4-5 Stability.....	69
III-5 Experimental Results and Discussion.....	69
III-5-1 Influence of the Proportional Gain.....	69
III-5-2 Influence of the Integral Action.....	70
III-5-3 Influence of the Derivative Action.....	71
III-5-4 Comparative Analysis of P, PI and PID Controllers.....	72
III-6 Educational Contribution of the Developed Platform.....	74
III-7 Conclusion.....	75
The startup part.....	77
General conclusion.....	80
Bibliographies.....	81

figure list:

Figure I-1: response of feedback control system to a step setpoint.....	17
Figure I-2: response of a regulated process to a step disturbance.....	18
Figure I-3: open-loop system.....	19
Figure I-4: closed-loop system.....	20
Figure I-5: schematic diagram of the principle of regulation.....	22
Figure I-6: on-off regulation.....	23
Figure I-7: block diagram of the control loop.....	23
Figure I-8: unit-step response curve showing t_d , t_r , t_p , M_p , t_s	30
Figure II-1: simplified block diagram of the closed-loop control system.....	36
Figure II-2: comprehensive physical-logical architecture of the PID control system....	36
Figure II-3: ESP32 development board used as the main controller.....	40
Figure II-4: ultrasonic sensor"HC-SR04".....	41
Figure II-5: working principle of the HC-SR04 ultrasonic sensor.....	41
Figure II-6: brushless motor.....	42
Figure II-7: electronic speed controller"50A".....	43
Figure II-8: brushless DC motor and an electronic speed controller.....	43
Figure II-9: battery power supply.....	44
Figure II-10: electrical wiring diagram of the system.....	47
Figure II-11:3D printed mechanical components.....	49
Figure II-12: web interface.....	51
Figure III-1: web-based monitoring and PID tuning interface.....	58
Figure III-2: system organigram: code logic and functional data flow.....	61
Figure III-3: ESP32 hardware pin configuration.....	62
Figure III-4: wireless communication and web server architecture.....	63
Figure III-5: trajectory planner web interface.....	64

Figure III-6: automatic arming sequence and timeline.....	65
Figure III-7: automatic soft landing mechanism.....	66
figure III-8: the response of the system using proportional control.....	70
Figure III-9: the response of the system using proportional integral control “kp=2; ki=0.5”	70
Figure III-10: the response of the system using proportional integral control “kp=2; ki=0.5”	71
figure III-11: the response of the system using proportional integral Derivative control “kp=2; ki=1.4; kd=1”	72
Figure III-12: the response of the system using proportional integral Derivative control “kp=2; ki=2; kd=2”	73

List of tables:

Tableau I-1: comparative analysis of P, PI, PID controllers.....	27
Tableau I-2: different PID controller structures.....	28
Tableau II-1: electrical connections of the system.....	48
Table III-1: Comparative Analysis of P, PI and PID Controllers.....	74

List of Abbreviations:

Abbreviation	Full Meaning
Pid	Proportional–Integral–Derivative
Pi	Proportional–Integral
P	Proportional
Pd	Proportional–Derivative
PWM	Pulse Width Modulation
ESP32	Espressif 32-bit Microcontroller

HC-SR04	HC-SR04 Ultrasonic Distance Sensor
ESC	Electronic Speed Controller
BLDC	Brushless Direct Current
DC	Direct Current
KP	Proportional Gain
KI	Integral Gain
KD	Derivative Gain
GPIO	General Purpose Input/output
i/O	Input/output
Wi-fi	Wireless Fidelity
USB	Universal Serial Bus
PVC	Polyvinyl Chloride
TF	Transfer Function
SSE	Steady-State Error
TOR	Tout Our Rien (On-Off Control)
VCC	Voltage at the Common Collector
GND	Ground
TRIG	Trigger
IoT	Internet of Things
IDE	Integrated Development Environment

General Introduction:

Automatic control systems are integral to modern industrial operations where it's imperative to maintain stability, accuracy and efficiency of the system. Automatic control systems are employed in many different applications, including but not limited to robotics, manufacturing systems, transportation, aerospace, and process control.

While the Proportional–Integral–Derivative (PID) controller is the primary control technique used in most applications because of its simplicity and effectiveness, the design/tuning of PID controllers is a key component of engineering education. Simply having a theory about how the PID control techniques work is often insufficient to fully understand these concepts; in addition, hands-on experimentation and visual can help demonstrate how different PID control parameters can impact system performance. This experimentation can assist in relating theoretical concepts to practical, real-world applications.

With this background in mind, the goal of this project is to create and develop a didactic PID control bench using an ESP32 microcontroller. The developed platform includes an ultrasonic sensor for position measurements, a brushless motor and Electronic Speed Controller (ESC) for actuation, and a web-based user interface to allow for real-time monitoring and parameter setting. Users can then experiment with each form of the PID control and observe the associated dynamic response of the system.

This project aims to create an inexpensive, interactive educational System-which will allow students to use a bench to study Automatic Control Theory by monitoring real-time and tuning PID Controllers Gain value(s). The bench will give students the opportunity to evaluate how well the controller has worked, by understanding how changes in the proportional, integral, and derivative gain values affect performance.

To achieve these objectives, this dissertation is divided into three chapters. In Chapter 1, we will describe the theoretical concepts of Automatic Control & PID Control Systems. In Chapter 2, a description will be given on how to build an Experimental Bench with both Hardware and Software Components to accomplish the experimental goals. Finally, in Chapter 3 we will show the results of various tests performed using the

Experimental Bench and discuss the Performance of the Controller based on a Comparison with other Controllers that were tested.

Chapter I:

Fundamentals of Automatic Control and
PID Regulation

I-1 Introduction:

Automatic control is a part of modern industrial systems. This is because it allows machines and processes to operate with accuracy and consistency. They can do this without anyone telling them what to do.

In industrial settings, it's important to maintain proper levels for factors such as temperature, pressure, and speed. We need to take action when there are issues. That's why we have rules and control systems in place. These systems form the foundation of automation engineering.

Looking at regulation systems allows us to see how a process compares a current value to a target value. It can then make the needed changes to fix the error. There are methods to manage and control processes in industrial settings. PID controllers are the most commonly used and they work quite well. They are simple. Make systems work better.

This part covers the basic ideas of control. We'll begin with a few reminders about servo-control and regulation. Next, we will examine loop and closed-loop systems. After that, we will discuss industrial PID controllers. What they can do. Finally, we will talk about how to figure out how good control systems are performing. We will examine factors such as rise time, overshoot, and stability. This will help us see how systems that are controlled work.

I-2 Servo-control and Regulation Reminders:

I-2-1 Tracking Servo-control:

The term "servo-control" describes a system whose principal goal is to swiftly reach its setpoint value (shown as C) and maintain it (where $Y(t)$ is the system's measured value, displayed in blue in Figure I-1), independent of external disturbances.

In order to properly adjust the system, the general idea is to compare the setpoint with its current condition. Another name for this is a closed-loop control system [6].

Figure I-1 illustrates the response of a servo-controlled process to a step change in the setpoint [3].

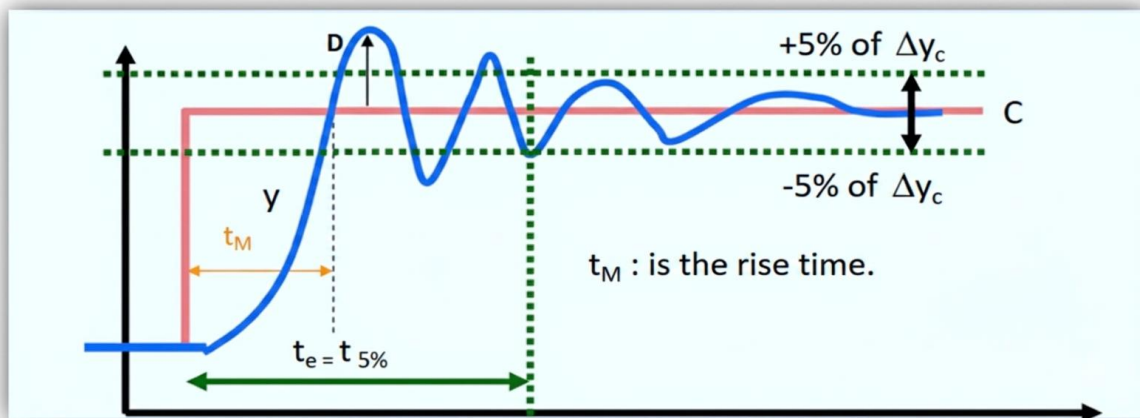


Figure I-1: response of feedback control system to a step setpoint

➤ **Examples: [6]**

- **Temperature Control (Servomechanism):** In a heat treatment furnace, reaching a particular temperature profile over time.
- **Airflow Ratio Control:** Achieving optimal (stoichiometric) combustion by regulating air flow in relation to gas flow.

I-2-2 Regulation:

Automation (or Control Systems Engineering) is a subfield of engineering science that includes regulation. It refers to the collection of methods used to maintain a physical variable at a particular value called the setpoint (shown as Y) without the need for human intervention.

The controlled variable, $y(t)$, is greatly impacted by external disturbances (disturbing variables), yet the setpoint, which stands for the process's intended goal, stays constant [8] [1].

Figure I-2 illustrates how a regulated process reacts to a step disturbance [3].

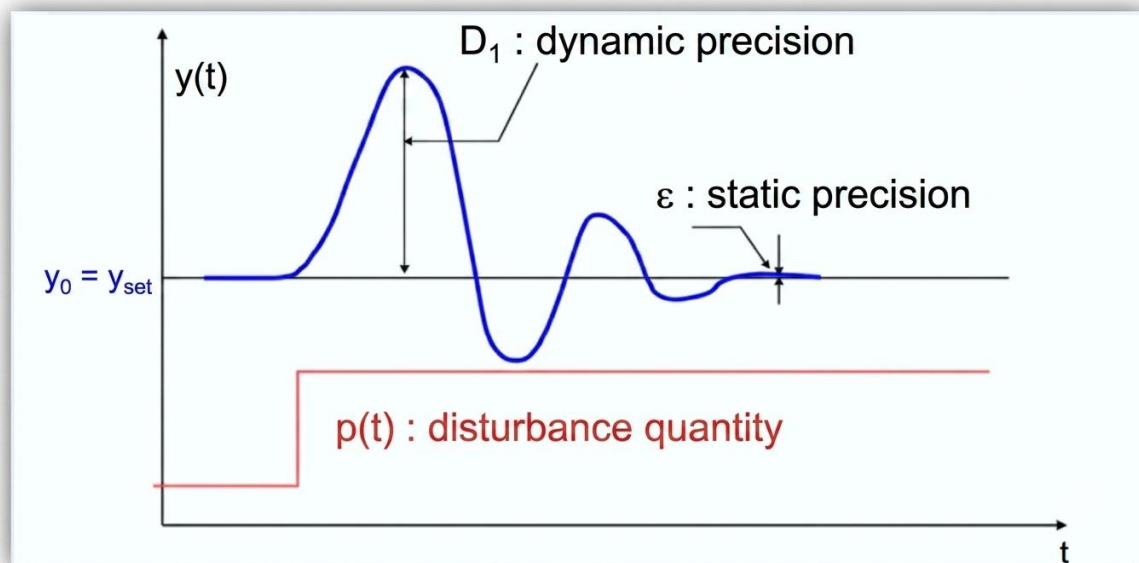


Figure I-2: response of a regulated process to a step disturbance

➤ **Examples: [7]**

- Controlling the temperature in a space that is subject to climatic fluctuations (changes in the outside weather).
- A tank's level is controlled by a number of supply (inflow) and discharge (outflow) rates.
- controlling the pH of wastewater flows meant to be dumped into rivers.

I-2-2-1 objectives of regulation:

The goal of feedback control or regulation is to guarantee that a process operates in accordance with standards established by a specification sheet (cahier des charges). Aspects related to energy efficiency and environmental preservation, as well as personnel and facility safety, must be considered.

The requirements specify the qualitative standards that must be met, which are typically converted into quantitative standards, like:

- stability.
- Precision (accuracy).
- Rapidity (Speed).
- Evolutionary laws (tracking) [1].

➤ **Examples of qualitative objectives include the following: [2]**

- Using a burner to achieve proper air-gas combustion.
- Keeping a product mixture's quality consistent.
- Achieving a steady fluid flow rate in a pipe in accordance with specifications.
- Adjusting the furnace temperature based on a specific set of instructions.

I-2-2-2 Regulation Typology: There are two types of regulation loops: open-loop regulation and closed-loop regulation.

a. Open-loop regulation: An open-loop system is one in which there is no feedback between the input and the output, which means no output information is involved. It is traditionally made up of a process, an actuator, and a controller. As illustrated in the following figure [9].

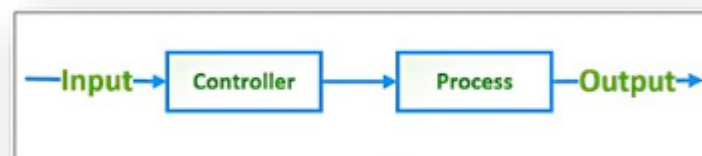


Figure I-3: open-loop system

Benefits and limit of Open-Loop Systems:

There are Benefits and limit to the open-loop approach, such as:

➤ **benefits: [7]**

- It makes it possible to anticipate events and get extremely quick reaction speeds.
- Because the system is dynamically stable, oscillation is not a concern.
- In situations where final control is not achievable, it is the only practical approach.

➤ **Limit: [10]**

- Poor anti-interference ability: open-loop control system has no feedback link, so it is less resistant to external interference.

- Sensitivity to disturbances: A disturbance can impact the output at any moment, even if the output is in line with the setpoint. This cannot be fixed by the operator.

b. Closed-loop control:

In closed-loop regulation, the controlled variable is influenced by the regulating variable (manipulated variable) to keep it within predetermined bounds even in the face of disturbances. A BF (Closed-Loop) system is depicted in the figure below [3].

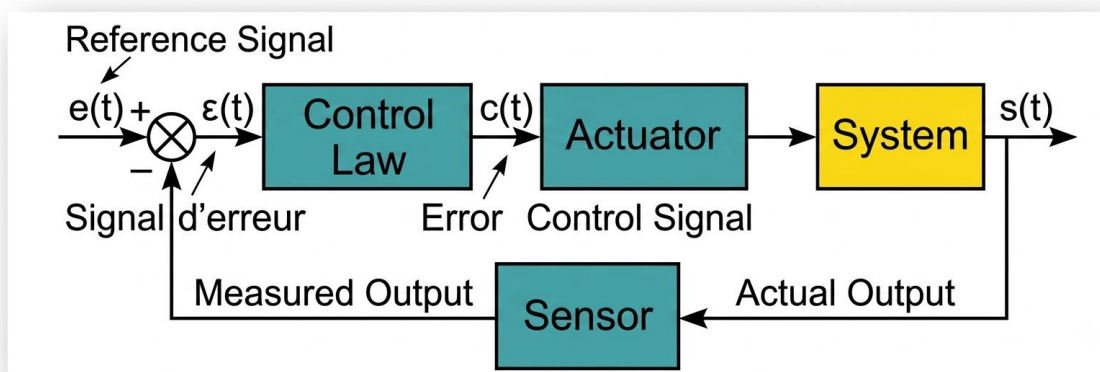


Figure I-4: closed-loop system

➤ Positive Features: [7]

- Throughout the process, feedback automatically compensates for a significant amount of disruptive elements.
- It is not required to fully understand the laws or the behavior of the many loop components, especially the process.
- When choosing components, it is helpful to understand the statistical and dynamic patterns of the many phenomena that are observed.

➤ Negative Features: [7]

- The accuracy and precision of the setpoint and the measured values determine how precise and accurate the regulation is.
- The features of the several loop components, particularly the process, determine the dynamic behavior of the loop.

-
- Disturbances or eventual alterations in the setpoint value must appear at the process output for the regulation to send a command to the control element.

c. Suitability of Closed-Loop Regulation for the Proposed System:

Open-loop systems are straightforward and user-friendly. They are constrained. Open-loop systems are insufficient when there is a process interference or when precise control is required. The issue is that the controller is unable to determine whether it has achieved its objective because it does not receive any information about what is actually happening.

Closed-loop regulation is not the same. The output is continuously checked. evaluates it against the target value. It automatically makes the necessary corrections if something is incorrect. As a result, the system becomes more dependable and can maintain the controlled variable around the intended setting.

Our effort aims to regulate a flying axis's location. Air disturbances, the mechanics' reaction, and the motor's inertia are some of the factors that can influence its motion. Giving an order and hoping for the best is insufficient to maintain stability in a circumstance like this.

Closed-loop regulation is therefore the best option. In order for the controller to identify any issues and immediately modify the motor command, we employ a sensor to provide feedback. This implies that our system can provide us with accuracy, correct errors more quickly, and act more steadily—all of which are crucial for understanding how PID control operates.

The proposed system benefits greatly from the closed-loop regulation method since it employs the sensor as a feedback component. The controller can quickly adjust the motor command by using the ultrasonic sensor to identify any deviations. For the proposed system, this is crucial. The suggested system must be precise and reliable. The Proposed System is more precise and stable because to the closed-loop regulating system. For the proposed system, the closed-loop regulation system is the preferred option [12].

I-2-2-3 General Regulation Principle:

Physical quantities must be kept at predetermined levels in the majority of commercial, industrial, and even residential products and installations, even when these amounts are impacted by internal or external changes. For instance, the temperature of an oven, the flow rate of a gas pipeline, or the level of a water tank—all of which are naturally variable—must be controlled by taking the necessary steps on the relevant process.

A straightforward adjustment, referred to as open-loop control, is adequate to reach and maintain the desired value if the disturbances impacting the quantity to be controlled are slow or insignificant (e.g., manually adjusting a water faucet).

This kind of correction is inadequate most of the time since it is too rough or erratic. As a result, the controlled variable, sometimes referred to as the regulating variable, must be continuously compared to the target value (setpoint) and adjusted accordingly [7].

The concept of regulation is depicted in the figure below

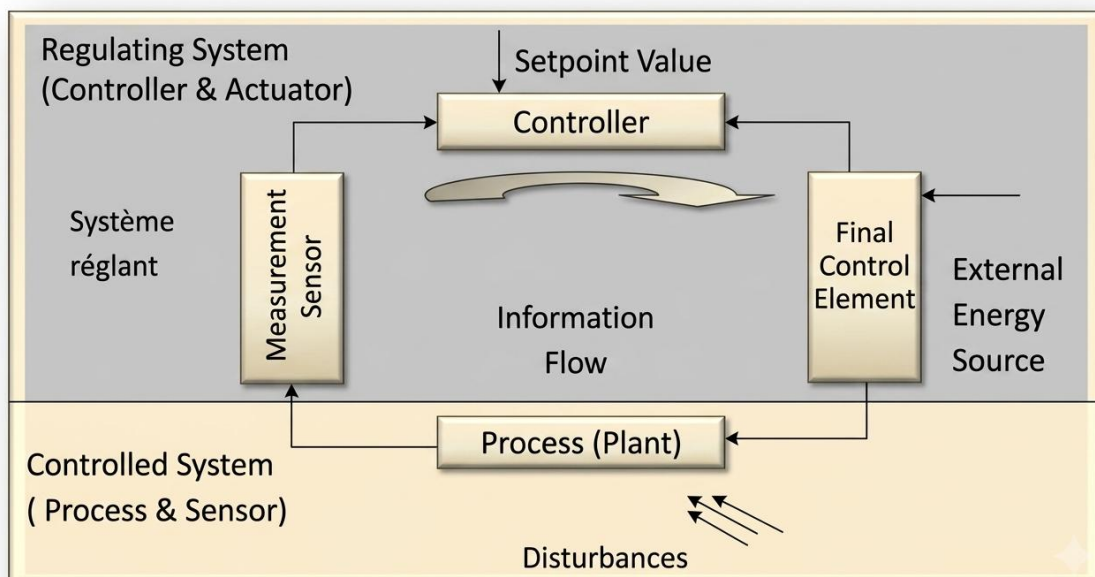


Figure I-5: schematic diagram of the principle of regulation

I-2-2-4 On-Off Regulation (Tout Our Rein, or TOR):

This device is mostly used for processes with high inertia, such as temperature or level variations. The last control element, which can only function at 100% or 0%, is what defines on-off regulation. An open or closed contact can be used to depict the regulator's action [4][1].

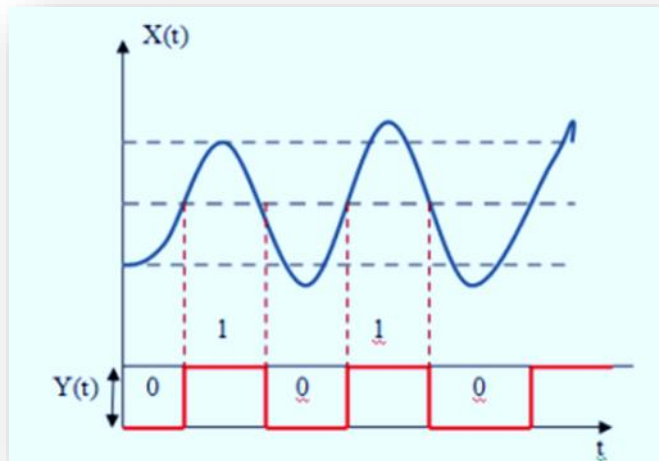


Figure I-6: on-off regulation

I-3 Study of Industrial PID Controllers:

An industrial PID controller that receives data from industrial applications. Feedback is used by a PID controller to manage a process. The controller applies a correction based on proportional, integral, and derivative terms after continually calculating an error value as the difference between a process variable and a desired setpoint [1].

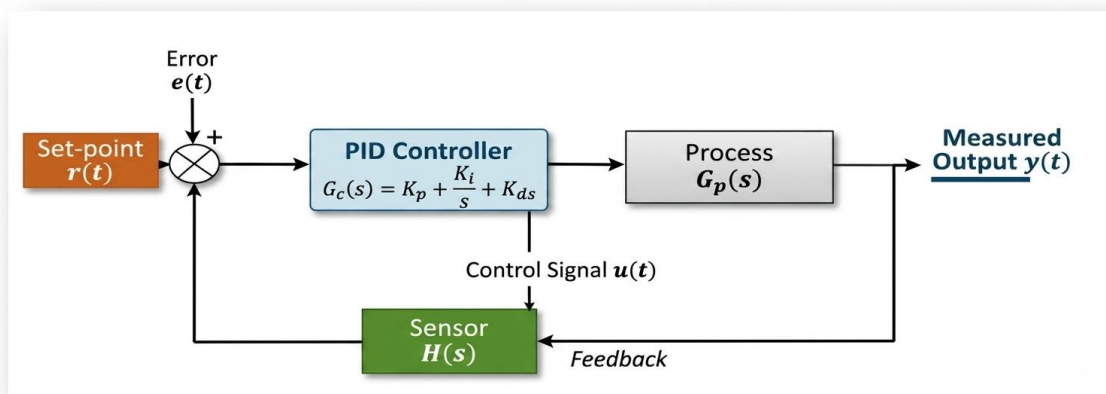


Figure I-7: block diagram of the control loop

Industrial Process Control has two tips:

- ❖ Digital control
- ❖ analog control

I-3-1 Analog Control:

A system that runs on a constant voltage or current signal. Analog signals, which are electrical values that correspond to a continuous physical quantity, are the basis for the operation of analog controllers.

I-3-1-1 PID Actions:

These three steps are combined to create a PID controller, which basically performs the following three tasks: [7]

- The proportional function increases the accuracy and speed of the system.
- Steady-state error (offset) is eliminated by integral function.
- The derivative function speeds up the process of rectification.

a-Proportional Action: The proportional action minimizes the error ε between the measurement and the setpoint, which lowers the response time and rise time. It is found that increasing the controller gain K_c speeds up the closed loop's overall behavior and reduces steady-state error. The stability of the closed loop limits the use of large gain

values, despite the temptation to do so in order to accelerate the process reaction. In fact, an overly high gain value causes oscillations and promotes system instability. [7]

- The controller's output is providing:

$$Y(t)=K_C.\varepsilon(t) \quad (1)$$

- Its transfer function is therefore:

$$C(s)=K_c \quad (2)$$

b-integral action:

As long as the error between the setpoint and the measurement is not zero, the integral action continues to operate in proportion to the area of the error. Since the integral operation removes the steady-state error or offset, it is said to give steady-state precision. The integral time T_i characterizes the integral action. The controller's output is provided by [7]:

$$Y(t)= \frac{K_c}{T_i} \int_0^t \varepsilon(t)dt \quad (3)$$

Similar to proportional action, an excessive amount of integral action causes the control loop to become unstable. Stability and speed must be balanced when tweaking it. Although the integral term increases accuracy, it also introduces a phase lag of $(-\pi/2)$, which could cause the system to become unstable because of the decreased phase margin [11].

The integral action's transfer function generally looks like this:

$$C(s)=\frac{K_c}{(t_i.s)} \quad (4)$$

c-Derivative Action:

This action considers the velocity, or rate of change, of the error between the measured value and the setpoint. It functions as a stabilizing element, in contrast to integral activity. [7]

in actuality, it yields an output that fluctuates in direct proportion to the rate of change of the mistake ε :

$$Y(t) = T_d \frac{d\varepsilon(t)}{dt} \quad (5)$$

The transfer function that follows is implied by this:

$$C(s) = T_d s \quad (6)$$

By acting on the error's variance, the derivative action speeds up the system (reduces response times). By increasing the phase margin through a phase lead ($+\pi/2$), it also enhances system stability. This action should never be utilized on its own since it is zero in steady-state conditions [11].

d-Integral and Proportional Actions (PI):

The dynamic response of a pure integral controller is comparatively slow. Conversely, a proportional controller can't totally eradicate the steady-state error, but it can respond quickly to control deviations [11].

The benefits of the P controller (quick response to deviation) and the I controller (removal of steady-state error) can be combined by combining proportional and integral actions.

The way it transfers is:

$$C(p) = K_c \left(1 + \frac{1}{T_{ip}}\right) \quad (7)$$

e- Derivative and Proportional Action (PD):

This blends derivative and proportional operations. Although noise sensitivity prevents this ideal version of the controller from being physically realized, its transfer function is as follows [11]:

$$C(p) = K_c (1 + T_d P) \quad (8)$$

f- Proportional, Integral, and Derivative Actions (PID):

PID control combines the benefits of each operation to provide optimal regulation: The I component removes the steady-state error, the D component opposes changes in the

controlled variable and stabilizes the control loop, and the P component responds to the emergence of a control deviation. Because of this, this kind of controller is most frequently utilized in industrial settings [7].

The structure of the PID action determines its transfer function (TF); for instance, the TF for the mixed (ideal) structure is as follows [11].

$$C(p) = K_c \left(1 + \frac{1}{T_{ip}} + T_d p \right) \quad (9)$$

g- Comparative Analysis of P, PI and PID Controllers:

After presenting the different corrective actions, it is useful to summarize the main characteristics of the most commonly used controllers. The following table highlights the principal differences between P, PI, and PID control in terms of dynamic performance and regulation quality [14].

Criteria	P controller	Pi controller	Pid controller
Reponse speed	fast	Moderate	Fast and balanced
Steady-state error	present	Eliminated	Eliminated
Overshoot	Can be significant	Moderate	Reduced
Stability	Moderate	Good	Very good
Complexity of tuning	simple	Medium	More complex
Regulation quality	Acceptable	Good	Excellent

Tableau I-1: comparative analysis of P, PI, PID controllers

A general observation from this comparison is that the PID controller offers the most complete and efficient regulation. By combining rapid response, good precision, and improved stability, it is considered the most suitable solution for systems requiring high performance.

I-3-1-2 Different PID Controller Structures:

The P, I, and D parameters can be combined in a PID controller in a number of ways. The PID controller can really have a series, parallel, or mixed (ideal) form, as seen in the following table [13]:

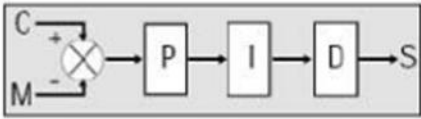
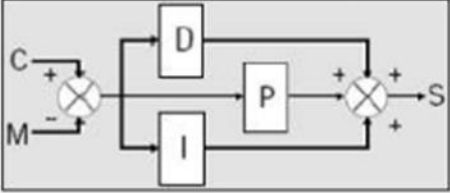
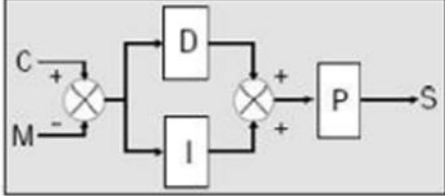
Structure du régulateur PID	Schéma et fonction de transfert
Série	 $K_P \left(\frac{T_i + T_d}{T_i} + \frac{1}{pT_i} + pT_d \right)$
Parallèle	 $K_P + \frac{1}{pT_i} + pT_d$
Mixte	 $K_P \left(1 + \frac{1}{pT_i} + pT_d \right)$

Tableau I-2: different PID controller structures

I-4 Performance Criteria of Control Systems:

I-4-1 rise time:

➤ Definition:

Rise time (T_r) is the time taken for the response to rise from 10% to 90% of its final value after application of a step input. [12]

➤ Physical Meaning:

- The rise time is reflecting the speed of response of system.
- Fast system has a small rise time.

-
- A long rise time means a slow system.

The rise time represents the period required for the motor driven system to reach near the desired setpoint after a change in reference input in developed didactic bench [12].

➤ **Influence of PID Controllers:** [12]

- By increasing the proportional gain, the rise time becomes shorter (in general).
- Rise time is moderately affected by integral action.
- The error derivative act shapers transient capacity and reaction steady.

I-4-2 overshoot:

➤ **Definition:**

The maximum overshoot is the maximum peak value of the response curve measured from unity. If the final steady-state value of the response differs from unity, then it is common to use the maximum percent overshoot. It is defined by [15]

$$Mp\% = \frac{c(tp) - c(\infty)}{c(\infty)} \times 100 \quad (10)$$

- **Physical meaning:** In the experimental bench, it represents the situation where the motor surpasses the target position before returning to the desired value [15].

➤ **influence of PID Controllers:** [15]

- A high proportional gain increases overshoot.
- The integral action may increase overshoot.
- The derivative action reduces overshoot by improving damping.

I-4-3 Settling time, t_s :

➤ **Definition:**

The settling time is the time required for the response curve to reach and stay within a range about the final value of size specified by absolute percentage of the final

value (usually 2% or 5%). The settling time is related to the largest time constant of the control system. Which percentage error criterion to use may be determined from the objectives of the system design in question. [12]

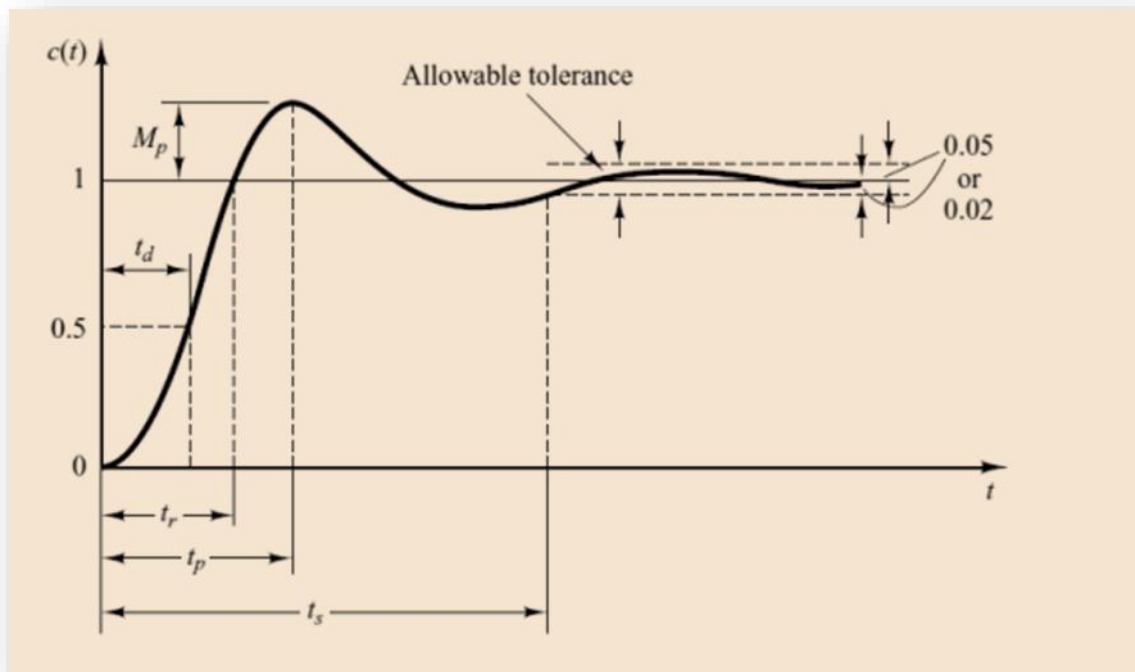


Figure I-8: unit-step response curve showing t_d , t_r , t_p , M_p , t_s

➤ Influence of PID Controllers: [16]

- Increasing proportional gain reduces settling time but may increase oscillations.
- Integral action may increase settling time.
- Derivative action reduces settling time by improving damping.

I-4-4 steady-state error:

• Definition:

The steady-state error is the difference between the desired final output and the actual one. Because a non-zero error is required to drive it, a proportional controller generally operates with a steady-state error. Steady-state error (SSE) is proportional to the process gain and inversely proportional to proportional gain. SSE may be mitigated by adding a compensating bias term to the setpoint AND output or corrected dynamically by adding an integral term.[5]

$$E_{ss} = \lim_{t \rightarrow \infty} [r(t) - y(t)] \quad (11)$$

➤ **Influence of Controllers:**

- P controller → non-zero steady-state error.
- PI controller → eliminates steady-state error.
- PID controller → eliminates steady-state error.

I-4-5 stability:

• **Definition:**

A control system is said to be stable if the output is always finite in value for every finite and bounded input. In other words, for a finite bounded input being applied to the system, the output should remain finite and not diverge and should always come to a finite value.

A stable system will always return to rest after being slightly disturbed, while an unstable system will exhibit larger oscillations and wilder behavior with growing amplitude or without any limits. [12]

I-5 Digital Control in Embedded Systems:

There is always digital technology. For this reason, digital embedded platforms are increasingly being used in control systems. These platforms from the outdated analog circuits are being used. These systems do not use fixed hardware components to create the control law. It is constructed using a real-time math problem-solving technology.

Digital control operates by gathering data about the environment from sensors. After that, it makes use of math to interpret the data. It then transmits a signal to the actuator. All of this repeatedly occurs. This allows the controller to respond quickly when something goes wrong. [18]

Analog control is inferior to digital embedded control. Modifying the control settings is simpler. It is more adaptable. Additionally, you can incorporate features like communication and monitoring within the system. [12] [18]

It is simpler to test new ideas and assess their effectiveness when PID algorithms are used in embedded control. You don't need to replace the hardware in order to modify or improve the controller because it is software. [12] [17]

Digital embedded control is a solution for contemporary automatic systems because of this. It is particularly useful in labs and schools where you need to be able to make changes and observe the results over time. [18] [17]

Because digital embedded control is user-friendly and effective, it is the best option. Because digital embedded control systems are useful and effective, their application is growing. [18]

I-6 conclusion:

In this chapter, the basic concepts of servo-control and regulation were presented in order to understand the operation of control systems. The principles of open-loop and closed-loop systems were explained, followed by the study of PID controllers and their different structures. The effects of proportional, integral, and derivative actions on system performance were analyzed. Finally, the main performance criteria such as rise time, overshoot, settling time, steady-state error, and stability were introduced. These theoretical concepts form the foundation for the practical implementation of the PID control system in the next chapter.

Chapter II:

***Design and Development of an ESP32-
Based PID Experimental Bench***

II Introduction:

Earlier, we looked at core ideas behind automated control, focusing especially on how regulation works, feedback loops operate, then touching on what proportional, integral, or derivative behaviors add to overall function. Those frameworks form a necessary starting point when learning about current methods found across industry settings.

Still, getting comfortable with these ideas means going beyond textbooks into hands-on testing. Real systems tend to act unlike their predicted versions - surprises pop up because of noise, quirks in materials, fuzzy readings, and lag. So building a setup where learners can watch such effects unfold live starts to make sense.

This chapter sets out to show how a teaching-focused control setup was built to demonstrate P, PI, and PID controller behavior. Rather than staying abstract, it gives learners direct experience through interaction with an actual working system. One part watches performance changes when tuning values shift. Another allows live testing across varied scenarios. Built with clarity in mind, the device connects textbook ideas to physical response. Real-time adjustments reveal what formulas alone cannot convey. Learning happens while observing reactions on screen. The method replaces passive study with active seeing.

Out there among tiny computers, the ESP32 stands out - its speed and built-in wireless tools fit perfectly into live feedback setups. Because it talks easily to sensors and moving parts, data keeps flowing without breaks. With everything linked up, decisions happen fast, right after readings come in. This loop never really stops - it just runs. What you get is something that learns, reacts, then adjusts again.

This part walks through each step of building the system. Starting with how everything fits together, it shows how pieces communicate across the setup. Moving forward, the physical parts take center stage - microcontroller choice matters, just like actuators, sensors, and where power comes from. Code gets explored next, especially how the PID logic works under varying conditions.

A screen layout gets built so people can interact with the setup while watching how key values change moment by moment. Seeing adjustments play out live helps learners grasp how different settings shape what the system does.

Only after everything else comes safety - a core part of building learning tools with moving parts or circuits. Built-in fixes work quietly, keeping things running without getting in the way.

This way, the system built becomes a full teaching aid showing how automatic control works in theory while giving hands-on understanding of actual use. Analysis and discussion of what came out of the setup appear in the following chapter.

II-2 System Specifications and Requirements:

II-2-1 Project Objectives:

The goal of this project is to build an affordable system that can be used to study automatic control systems. It will be used as a way for students to experience PID controllers in a hands-on manner.

It really does offer students a chance to see the effects of using integral and derivative gains on a real process. When students use the system they can see how each control action affects things like what happens, how much it overshoots, how long it takes to settle down, and how accurate it is in the end.

The system also has been designed to be extremely interactive so students can change the controller settings when the system is running without having to reprogram it. This makes learning much more fun because students see how changing the controller settings affect the system. The system focuses on making it easy for students to learn about PID controllers and how they work.

II-2-2 Functional Requirements:

To complete the job, the system must do several actions. These are about how it works, how accurate it is, and how it communicates with other devices. First, the system must be able to react quickly. To keep the system steady, it must swiftly receive and

process input from sensors. This implies that the controller can respond fast to system changes while ensuring that everything operates smoothly.

Second, the system needs to be capable of measuring distances. The sensor we use must be capable of providing us with control over the system's location and ensuring that it operates consistently.

Finally, the system must be capable of communicating with external devices without the use of cables. The ESP32 facilitates this by enabling us to use Wi-Fi or Bluetooth. This implies we can communicate data to someone who isn't physically there with the system, allowing them to see and control what's happening in real time.

II-2-3 Technical constraints:

When developing a system, we must consider both the restrictions and the system's requirements. The system includes moving pieces, such as the flying axis, which is critical for safety.

We must ensure that the flying axis is contained within a guide to prevent it from moving in an uncontrolled manner. This is necessary to ensure that the system is safe to use throughout our studies.

We must also consider the other side of the situation. The device uses a brushless motor, which requires a lot of power to operate, especially when it speeds up quickly. As a result, we must select a power source capable of providing the system with the power it requires as the power requirements fluctuate. This manner, the system will work properly. Do what it is designed to do. The system also has to be portable. The experimental bench should be small and lightweight so that it may be easily transported to labs and used. We want the system to be portable and easy to use in areas like laboratory environments, with the system.

II-3 System Architecture: The proposed system uses a closed-loop control structure. This structure brings together sensing, processing and actuation in one framework.

- It allows for monitoring of the system output.
- The system can correct deviations, from the desired reference in time.

- To help understand the system we will first look at a block diagram.
- Then we will dive into a physical and logical representation.

II-3-1 Simplified Block Diagram:

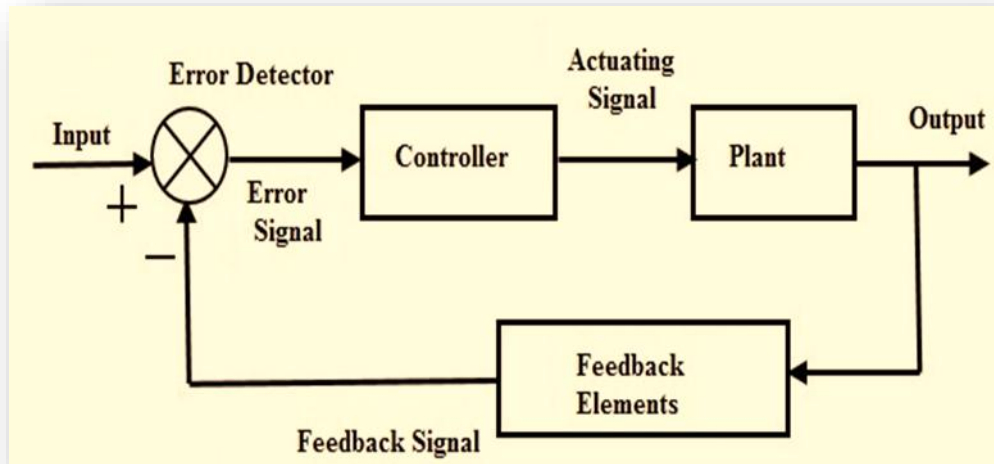


Figure II-1: simplified block diagram of the closed-loop control system

Figure II-1[20] depicts a simplified representation of the system. It depicts the primary functional components: the setpoint, controller, actuator, process, and feedback sensor. Before delving into the particular design, this figure provides a general understanding of how the system operates.

II-3-2 Description of the Integrated Feedback Loop:

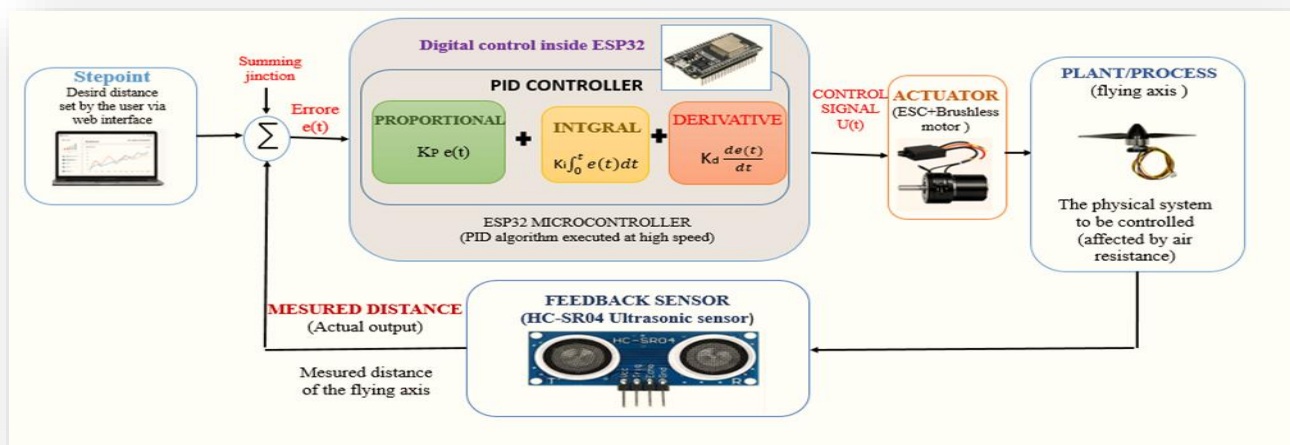


Figure II-2: comprehensive physical-logical architecture of the PID control system

In Figure II-2, you can see how the system uses a closed-loop control setup from the ESP32 microcontroller that links together parts and processing needed for regulation.

The system starts with setting the target distance, which is the desired distance entered by the user via the interface. This target distance is compared to the distance measured by the ultrasonic sensor. The difference between these two values results in an error signal.

The PID controller in the ESP32 looks for this error and performs three actions: Integral and derivative. It calculates the control signal to send to the actuator, which is the ESC and brushless motor. The actuator then creates the motion needed to adjust the system.

The sensor continues to measure the system output and sends it back to the controller. This is a closed-loop system, where the system self-corrects itself in time to keep it stable and accurate even with disruptions.

II-4 Hardware Selection and Specifications:

This section describes the electronic components we selected for the didactic control system. We chose each component because it fits the system's requirements, is readily available, reasonably priced, and simple to use.

II-4-1 Microcontroller Unit: ESP32:

➤ Definition:

The ESP32 is a low-cost microcontroller that can be found in many things such as robots, and IoT devices. This microcontroller has multiple functionalities, such as Pulse Width Modulation control modules, multiple digital I/O ports, and a powerful processing unit. The ESP32 can connect to Wi-Fi. [21]

➤ Principle of Operation:

The main computer in this system is the ESP32. After calculating the distance via the HC-SR04 ultrasonic distance sensor, the ESP32 uses a special way of solving the data called a PID control algorithm. The ESP32 creates a PWM signal and sends that information to the ESC. The second function of the ESP32 is to connect with the web

interface via Wi-Fi, allowing for configuration changes and real-time monitoring via the ESP32 and its Wi-Fi connection to the internet. [21]

➤ Reason for Selection:

We chose the ESP32 microcontroller as a component of the system since it has many amazing capabilities and is suitable for real-time embedded applications.



Figure II-3: ESP32 development board used as the main controller

As shown in Figure II-3 The ESP32 microcontroller combines processing capability, input and output interfaces, and communication tools. One of the features of the ESP32 microcontroller is that it contains two cores, which allows it to perform multiple tasks simultaneously, such as control processing and communication. This means that the control algorithm can function properly while still communicating with users and sending data.

The ESP32 microcontroller also includes a highly precise Pulse Width Modulation module. This is vital for managing the actuator, particularly when it comes to making the motor run at a certain pace. Also, the ESP32 microcontroller includes Wi-Fi, which simplifies system design by eliminating the need to integrate communication components. This implies we can create a web-based interface to monitor events in real time and alter settings.

II-4-2 Ultrasonic Sensor: HC-SR04

- **Definition:** HC-SR04, an ultrasonic sensor is used to measure distance by sending out ultrasound waves. [19]

➤ Working Principle:

The sensor sends out an ultrasound pulse from its transmitter and listens for the pulse's reflection (or echo) to return to its receiver. The travel time of the wave is measured by the ESP32 and is used to determine the distance travelled. The measured distance is used as the feedback in the control loop. [22]



Figure II-4: ultrasonic sensor"HC-SR04"

➤ Reason for Selection:

The ultrasonic sensor is really important for getting the feedback in the control loop. It measures where the system is.

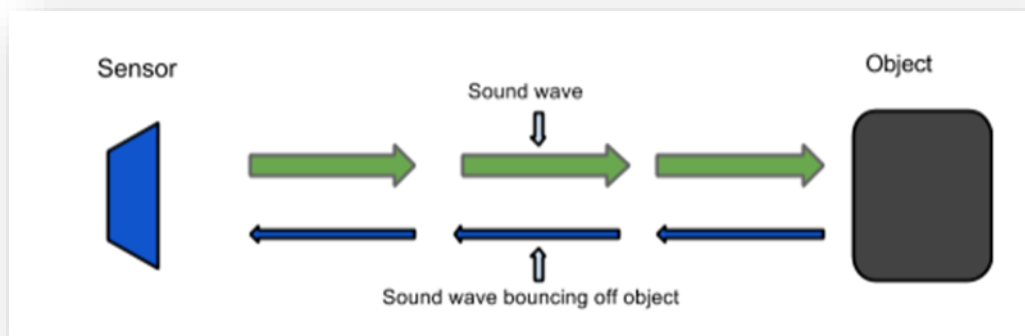


Figure II-5: working principle of the HC-SR04 ultrasonic sensor

As shown in Figure II-5 the HC-SR04 ultrasonic sensor works by sending out a pulse and then measuring how long it takes for the echo to come back. This tells us how away something is.

The ultrasonic sensor is used a lot in schools and in embedded systems because it is simple and does not cost a lot of money. It is also pretty good, at measuring distances that're not too far away. The ultrasonic sensor can detect when the system position changes. It helps to control the system by giving us the feedback we need from the ultrasonic sensor.

II-4-3 Actuation System: (ESC and Brushless Motor)

❖ Brushless Motor:

➤ Definition:

A Brushless DC (BLDC) motor is an electric motor that operates without mechanical brushes. It uses electronic commutation to control the rotation of the motor. [23]

➤ Operating Principle:

The motor converts electrical energy into mechanical motion. The speed of the motor is controlled by the ESC according to the PWM signal generated by the ESP32. The produced motion creates the force required to move the flying axis and maintain the desired position. [24]



Figure II-6: brushless motor

❖ Electronic Speed Controller (ESC):

➤ Definition:

An Electronic Speed Controller, or "ESC" for short, is an electronic controller that controls the speed of a brushless electric motor by providing it with the correct amount of electrical power. [23]

➤ Operating Principle:

The ESC will take a pulse width modulation (PWM) signal from an ESP32 and convert this PWM signal into a sequence of commands that will allow the brushless motor to operate correctly. As you increase the duty cycle of the PWM signal, the ESC can increase the speed of the brushless motor to match the output of the controller. [24]



Figure II-7: electronic speed controller"50A"

➤ **Reason for Selection:**

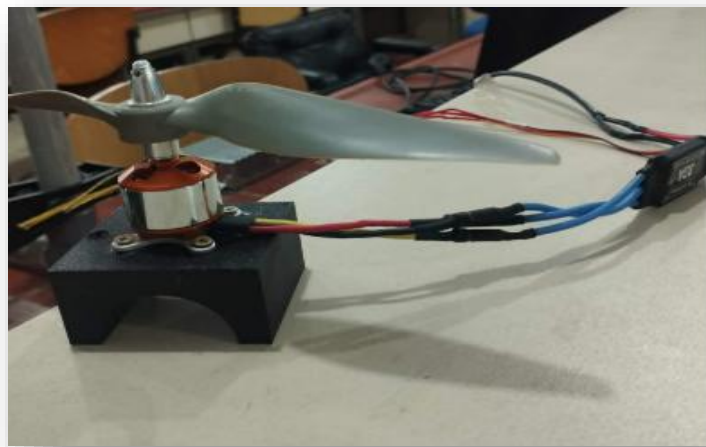


Figure II-8: brushless DC motor and an electronic speed controller

As shown in Figure II-8 the actuation system, which includes a brushless DC motor and an electronic speed controller (ESC):

- The brushless motor is employed because it is efficient, long-lasting, and responsive. It outperforms brushed motors in terms of performance and maintenance, making it suitable for long-term usage in control systems.
- The ESC interfaces the microprocessor and the motor. It receives a signal from the ESP32 and converts it into the appropriate electrical signals to run the motor.

This design allows for exact adjustment of the motor speed based on the control command sent to the brushless DC motor and the ESC.

II-4-4 Battery Power Supply:

- **Definition:**

An electrochemical cell that stores energy chemically and provides electrical energy from it. [26]

➤ **Operating Principle:**

The battery provides electrical power to an ESP32, ESC, and brushless motor. The battery uses the stored chemical energy (the energy source) to generate (produce) the electrical energy for circuit elements as it operates. [26]

➤ **Reason for Selection:**



Figure II-9: battery power supply

Using a battery allows for autonomous operation (no external power supply) makes the experiment bench more portable, and simplistically allows for a simple method of using the experiment bench in a laboratory.

II-5 Software Design and PID Implementation:

We have covered the physical parts of the setup until now. But how the system really works isn't just about the hardware. What actually makes it “smart” is the software inside the ESP32. The software's job sounds easy: it checks the sensor information and compares it to the target value and then figures out how the motor should act. But this needs to be done constantly and quickly so the system stays steady. The control program runs in a repeated loop which helps the system respond nearly instantly to changes.

II-5-1 Development Environment:

The program was created with the Arduino IDE. This tool is widely used for embedded systems and works well with the ESP32. It is easy to manage inputs and outputs and communication without needing complicated setups.

In this project, the ESP32 does many tasks at once. It checks distance using the ultrasonic sensor and runs the control algorithm and sends commands to the actuator and manages user interface communication. While these tasks vary, they all run within a structured program which ensures smooth operation.

II-5-2 PID Control Algorithm:

The PID controller is what makes the software work. It figures out how the system should behave based on the difference between where we want it to be and where it actually is.

Instead of going too deep into theory, it is more useful here to understand how each part behaves in practice:

- The proportional part of the PID controller reacts to the error away. If the system is far from where we want it to be it makes a strong response. This helps the system move quickly to where it needs to be. If the response is too strong it can make the system oscillate.
- The integral part of the PID controller looks at the error that has built up over time. It is really useful when the system gets close to where we want it to be. Just cannot seem to get all the way there. By adding this effect, the PID controller can get rid of errors that just will not go away.
- The derivative part of the PID controller looks at how the error's changing. It helps slow down the system before it goes far and misses where we want it to be. This makes the system more stable. Reduces oscillations.

When we put these three parts of the PID controller together we get a response that's both fast and stable. The PID controller produces a response that is what we need.

II-5-3 Program Structure:

The program is simple. It works really well. First when the program starts it sets up all the parts it needs to work. This means setting up the sensor defining how the PID controller will work and getting the communication going.

Then the system goes into a loop where it does the thing over and over again. Each time through the loop it reads the sensor value calculates the error uses the PID algorithm to figure out what to do and sends a signal to the motor.

This loop keeps going all the time, which lets the system adapt to what's happening right now. How fast the loop goes is really important because if it is too slow the system can become unstable. If it is fast it can control things really well.

II-5-4 Control Modes (P, PI, PID):

We made the system able to work in modes so it can be used to teach people about PID controllers. Of just using the full PID controller the system can also work in simpler modes like P or PI.

This lets users see clearly how each part of the PID controller affects the system. For example, if we just use control the system responds quickly but it also has a noticeable error when it gets to where it is supposed to be. If we add the part the system becomes more accurate and if we add the derivative part it reduces overshoot.

This makes the system really useful not because it works, but also because it can be used to learn about PID controllers. Users can try settings and see what happens which makes it a great tool, for learning.

II-6 Mechanical Assembly and Integration:

II-6-1 Electrical Wiring and Connections:

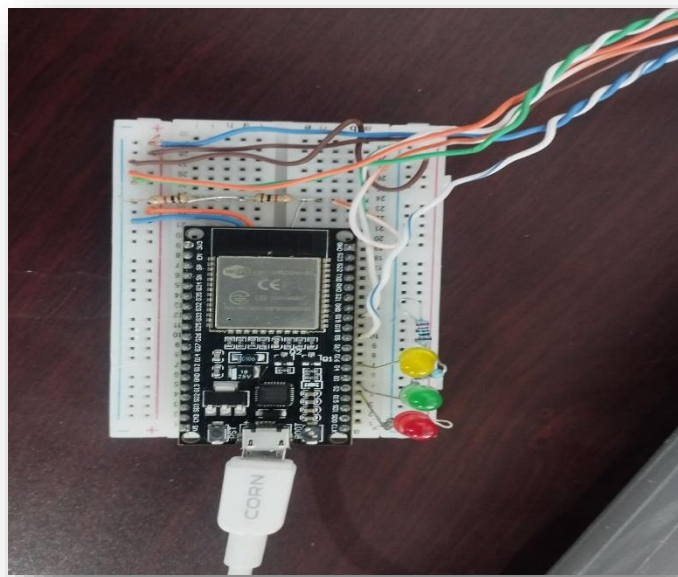
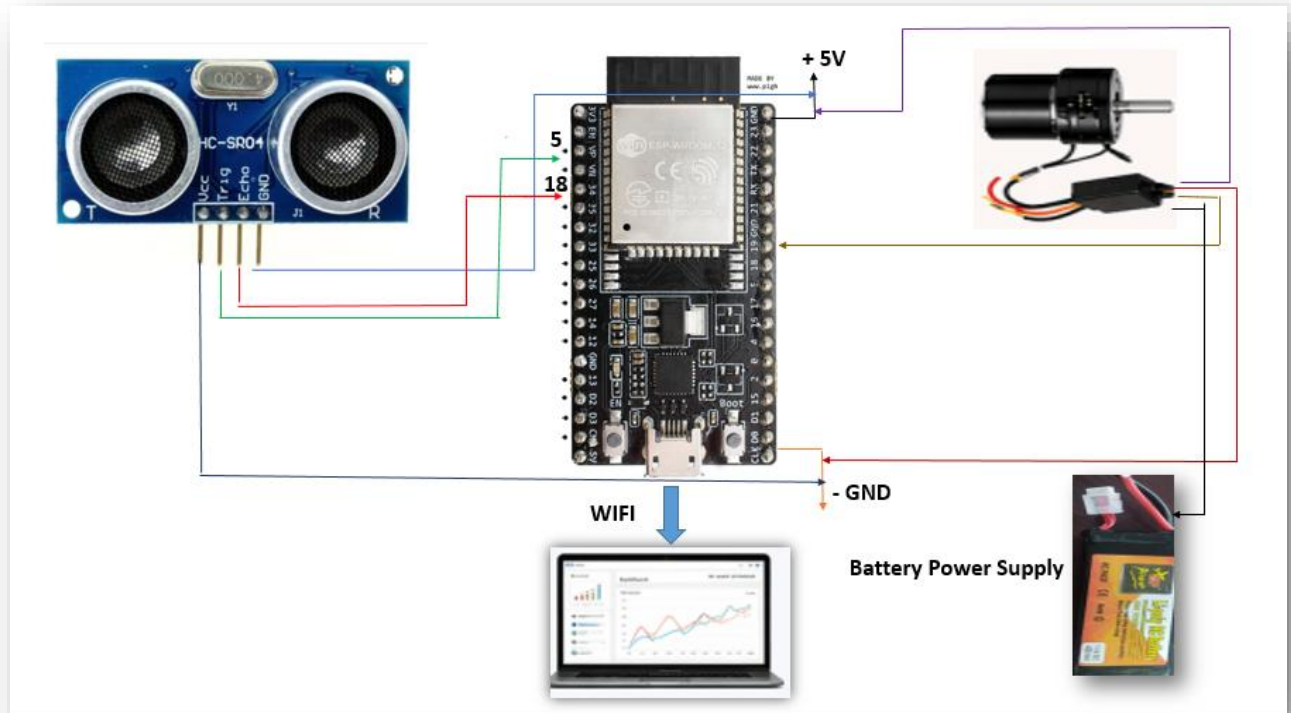


Figure II-10: electrical wiring diagram of the system

Component	ESP32 Pin
HC-SR04 TRIG	GPIO 5
HC-SR04 ECHO	GPIO 18
ESC SIGNAL	GPIO 19
Common ground	CND
HC-SR04 VCC	5V

Tableau I-1:electrical connections of the system

The HC-SR04 ultrasonic sensor, which is shown in Figure II-10 has four terminals that connect to the ESP32. The VCC terminal on the HC-SR04 sensor is connected to a 5V source. The GND terminal on the HC-SR04 sensor is connected to the ground. The GPIO 5 pin on the ESP32 is used for the TRIG signal on the HC-SR04 sensor. The GPIO 18 pin on the ESP32 is used for the ECHO signal on the HC-SR04 sensor. When the HC-SR04 ultrasonic sensor is connected to the ESP32 the ESP32 can send a signal and then measure how long it takes for the sound to come back. This allows the ESP32 to estimate the distance to an object using the HC-SR04 sensor.

The 50A Electronic Speed Controller or ESC controls a brushless motor. The ESC uses three wires to control the brushless motor. The ESP32 sends a control signal to the ESC using the GPIO 19 pin. The ESP32 generates a PWM signal that tells the ESC how voltage to supply to the brushless motor. The amount of voltage is adjusted using a PID algorithm. The PID algorithm is based on the mode of control that is being used.

All of the devices including the HC-SR04 sensor and the ESC share the same ground reference point. This is necessary, for signaling and communication. It is also necessary for the closed loop controls to work. The ESP32 has a Wi-Fi feature that allows it to send data to a monitoring system. This provides real-time monitoring of the parameters of the HC-SR04 ultrasonic sensor

II-6-2 Mechanical Assembly and 3D-Printed Structure:

In order to put the prototype together correctly, many custom parts had to be printed in order to guide the moving axis and align the electronic components properly. 3D printing provided a solution to make the parts easy to use and flexible.



Figure II-11:3D printed mechanical components

As shown in Figure II-11 a slouch carriage with rollers allows the swinging axis to slide easily on the vertical PVC guide rails. The slouch carriage supports the brushless motor, and the swinging axis can glide forward a uniform distance.

In addition, 3D printed supports were used to move the ultrasonic sensor properly relative to the moving part and ensure it reflected the correct direction from the sensor to the moving part. This improved distance measurement accuracy and yielded more quality feedback in the control loop.

Use of 3D printing in this project has several advantages. It enables creation of customized geometries to the dimensions of the prototype, reduces the manufacturing cost and facilitates future design changes.

Additive manufacturing was a powerful process to design a compact, practical, reproducible didactic bench.

II-6-3 Final Prototype Assembly:

Once the electrical and mechanical parts were ready, all components were brought together into a single operational prototype.

The ultrasonic sensor was positioned in such a way that it could continuously measure the displacement of the flying axis. The brushless motor and its propeller were fixed securely on the moving element, while the ESP32 and the ESC were mounted on the main structure to keep the wiring organized and protected.

At this point, the system was no longer considered as separate modules, but as one integrated control bench. Every action generated by the controller could now produce a direct physical reaction, and every movement of the system could immediately be measured and corrected.

This final integration stage confirmed the feasibility of the proposed design and prepared the platform for the experimental tests presented in the following chapter.

II-7 User Interface and Real-Time Monitoring:

One of the goals of this project was to create a control system that is not only functional, but also easy to use and understand from an educational point of view. To do this a web-based user interface was. Added to the ESP32.

The ESP32 has built-in Wi-Fi which means it can host a web server and create an interactive control page that can be accessed from any computer or smartphone on the same network. This makes it easy to monitor the system from a browser without needing to install software.

II-7-1 Web Interface Design:

The web interface was designed to make it easy for the user to interact with the control system. From this page the user can set the desired setpoint. Change the PID gains K_p , K_i and K_d in real time.

This means that the user does not need to stop the system or re-upload the program every time they want to change a parameter. Soon as a new value is entered it is sent to the ESP32, which applies the new control settings right away.

This real-time interaction makes the experimentation process more interactive. Students can see how changing one parameter affects the system, which makes the platform a useful learning tool.

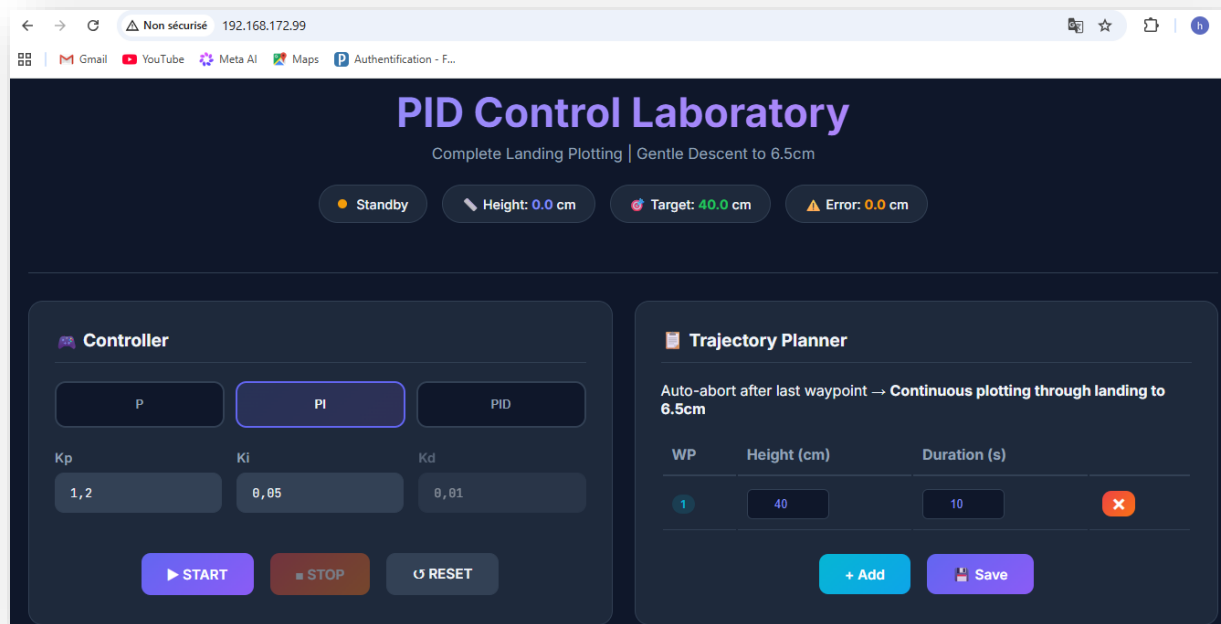


Figure II-12: web interface

II-7-2 Real-Time Data Monitoring:

The interface also allows the user to monitor the system variables in time. While the system is running, important values like the measured distance, setpoint and control signal are. Displayed on the screen.

This feature is very useful because it allows the user to see how the system responds moment by moment. Of just watching the mechanical movement the user can analyze the numerical data and understand how the controller works.

The interface can also display curves to show how the output changes over time. This helps to identify problems like overshoot, oscillations or slow settling.

II-7-3 Educational Contribution of the Interface:

The addition of this interface makes the project more valuable, for learning. In a control system changing controller parameters can be complicated and require technical expertise. With this interface the process is easy and accessible.

Students can do tests in a short time compare different PID configurations and develop a better understanding of automatic control behavior.

In this way the interface is not a monitoring screen but an active part of the learning experience. The user interface and real-time monitoring make it easy for students to learn about control systems. The ESP32 and user interface work together to provide a learning experience.

II-8 Safety Considerations and System Limitations:

The experimental control bench must be strong and safe to use in addition to being functional. The technology we are discussing combines control with a high-speed brushless motor that generates motion. As a result, as we were building it, we had to consider many factors.

The moving part was one of the things we were concerned about. The motor could move abruptly when we were conducting PID trials. not be steady. The reason for this is that the criteria were not yet established. We ensured that the moving component was led so it would remain on the path and not stray in order to prevent this from occurring.

Additionally, we ensured that the propeller and motor were securely fastened. The motor can travel at high speeds, so if it is not fastened firmly, it may vibrate and compromise its stability and construction quality. When we repeated the testing, we need an attachment to maintain everything functioning properly.

We configured the system so that the power circuit and control circuit are independent on the side. The sensing and computing are handled by the ESP32, which is powered by a USB. Through the ESC, a battery provides power to the motor. This prevents the controller from resetting and lessens issues brought on by variations in the current.

While developing the system, we also discovered a few restrictions. For learning, the ultrasonic sensor is adequate. It doesn't always provide reliable readings, particularly when the moving component rotates somewhat. The feedback signal may become quite loud as a result. Additionally, the parameters we select determine how well the system is regulated because we must adjust the PID gains by trial and error.

These restrictions are typical for a prototype and do not diminish the system's educational value. By demonstrating that measurement issues always have an impact on actual control systems, they actually make the experiments more realistic. The experimental control bench is still a useful tool for understanding control systems and how they function in practical situations.

The system we are discussing and the experimental control bench are crucial for understanding control systems. Both the system under discussion and the experimental control bench have certain drawbacks. For learning about control systems, they are still quite helpful. The experimental control bench serves as a tool for understanding the system under discussion and the experimental control bench itself.

II-9 Conclusion:

This chapter deals with the practical implementation of the proposed didactic PID control bench. Starting from the determined project specifications, the analysis chose the major hardware components, following the measurement, actuation, and digital processing requirements. Following that, the software part was elaborated to implement the PID algorithm and guarantee the continuous interaction of the sensor with the controller and the actuator.

Integration, in which the electrical assembly, mechanical structure, and 3D-printed supports are integrated into one complete and functional prototype, is also highlighted. At the same time, the development of a web-based monitoring interface introduces an interactive aspect to the system, making parameter adjustment and response observation more accessible during experimentation.

The many precautions and practical restrictions described in the final part highlighted the bench's actual operating conditions and proved the practicality of the suggested design from both technical and pedagogical aspects.

The system is now fully operational and available for experimentation. The following chapter will thus concentrate on presenting the collected data, tweaking the controller, and analyzing the dynamic performance of the constructed platform.

Chapter III:

Experimental Validation and Performance Analysis

III-1 Introduction:

After completing the design and development of the ESP32-based PID experimental bench, the next step consists in evaluating the behavior of the system under real operating conditions. The developed prototype integrates sensing, control, actuation, and real-time monitoring functions within a single interactive platform.

The purpose of this chapter is to present the experimental tests performed on the system and to analyze the responses obtained under different controller configurations. Through the developed web interface, it becomes possible to monitor the measured distance, visualize the evolution of the system response, observe the ESC output signal, and modify the control parameters directly during operation.

The interface also allows switching between proportional (P), proportional–integral (PI), and proportional–integral–derivative (PID) control modes while adjusting different values of K_p , K_i , k_d . In addition, different setpoint values can be introduced in real time in order to study the dynamic reaction of the flying axis under varying operating conditions.

The comparison between the three control strategies is mainly based on the influence of the stability and the speed and precision of the system. The obtained responses are evaluated according to the classical performance criteria of automatic control, namely rise time, overshoot, settling time and steady state error.

The experimental observations presented in this chapter make possible to evaluate both the technical performance and the didactic value of the developed didactic bench.

III-2 Experimental Conditions and Testing Procedure:

This section describes the experimental setup and technique used to assess the performance of the created PID control bench. The tests were conducted on the whole prototype created in the previous chapter, which included the sensing, control, actuation, and monitoring stages.

III-2-1 Experimental Setup:

The experimental apparatus is comprised of an ESP32, an HC-SR04 ultrasonic distance sensor, an HC-30A Brushless ESC, a Brushless DC Motor and a re-chargeable battery which is the main source of the system. The mechanics of the prototype are built based on a vertical guiding system constructed of 3D printing components.

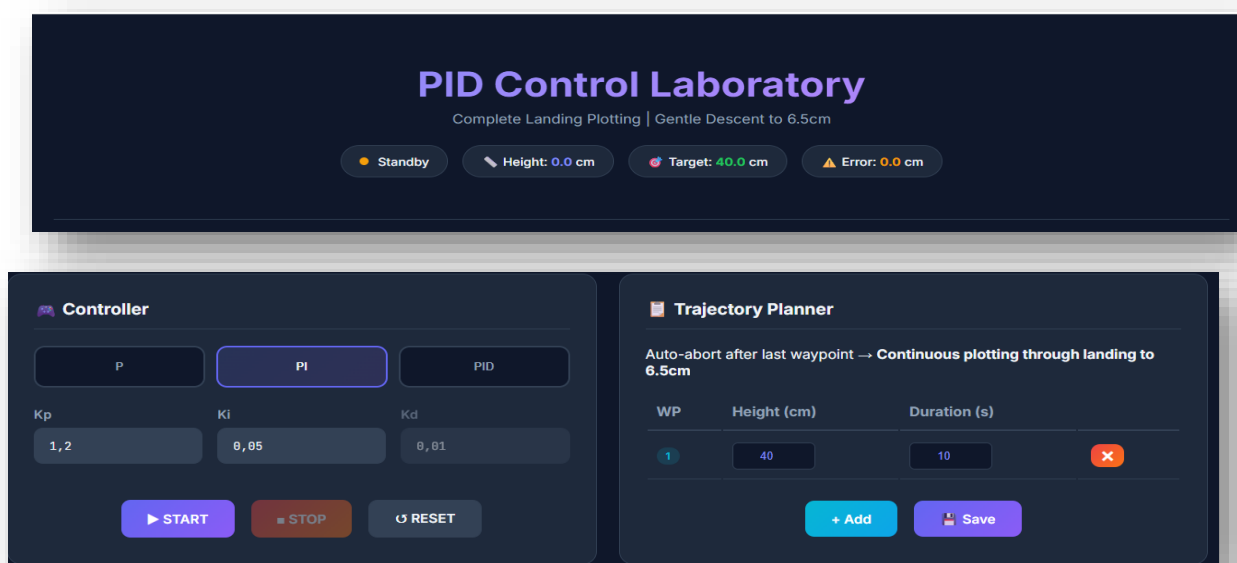
During operation the ultrasonic sensor will measure the location of the moving axis constantly and transfer this reading to the ESP32 control module. Depending on the regulation mode chosen, and based on the difference between the desired and actual reading the ESP32 outputs a PWM signal to control the ESC, and therefore the motor.

The prototype functions as a completely embedded autonomously powered system. It runs off of the battery with no need for any additional adapter, this gives the system mobility, and makes for a simpler experimental apparatus.

The tests were performed under normal laboratory conditions with the apparatus vertical to insure the moving axis can move smoothly.

III-2-2 Web-Based Monitoring and Control Interface:

An interface for real-time observation and control via the web was developed in order to enhance the interactivity of the didactic bench. It allows real-time supervision of the system behavior as well as direct modification of the controller parameters while the system is running.



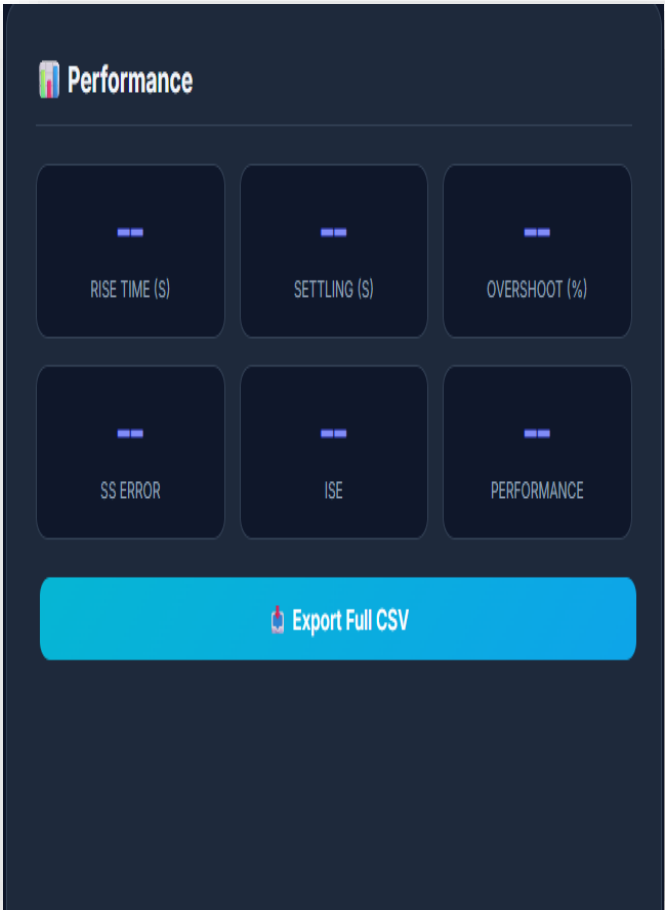
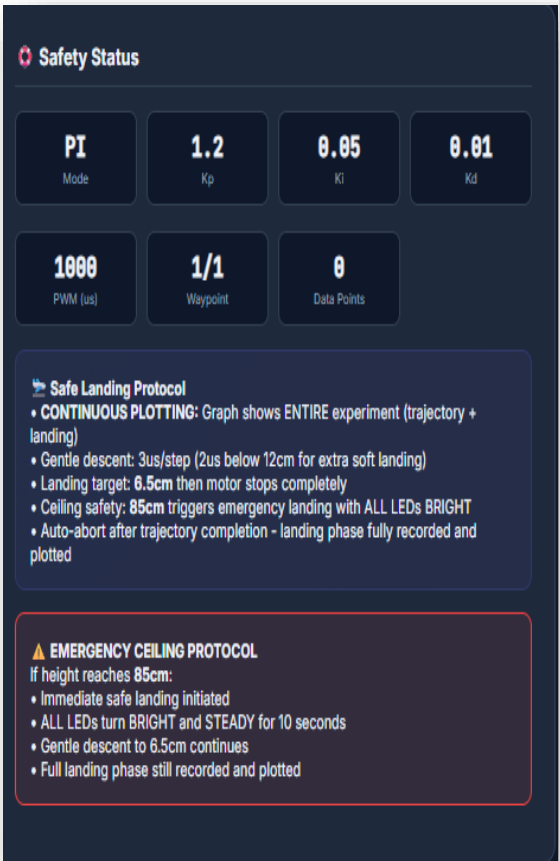
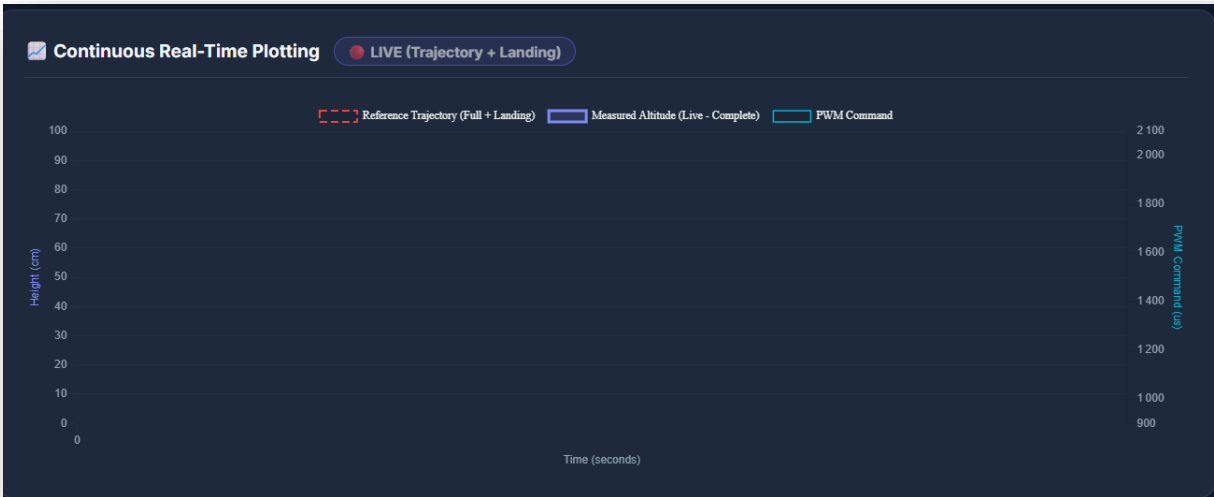


Figure III-1: web-based monitoring and PID tuning interface

In order to enhance operational safety, the implemented maximum allowable moving height of 85 cm would stop the system at the applicable limit while reducing motor speed as the moving axis approaches this maximum height, thus allowing smooth operation with no mechanical impacts.

In addition to the maximum allowable moving height, the platform also includes trajectory generation capabilities that allow for the specification of several different height references correlated to a specific time period. This capability allows the system to create pre-defined motion profiles and provides an advanced environment for evaluating the performance of the controller.

Finally, the interface allows for performance monitoring and data storage. Data from experimental results can be stored for future analysis, and assist in making Benchmark comparisons among controller settings and trajectory settings.

III-2-3 Test Procedure:

The experimental tests were performed according to the following procedure:

- The prototype was powered and connected to the wireless monitoring interface.
- A reference setpoint was introduced through the web dashboard.
- One controller mode (P, PI, or PID) was selected.
- The corresponding controller gains were adjusted progressively.
- The evolution of the measured distance and the ESC output was observed in real time.
- The response curves and numerical values displayed on the interface were recorded for analysis.
- This procedure was repeated for different controller configurations and different setpoint values in order to evaluate the influence of the corrective actions on the system behavior.
- Different trajectory profiles consisting of multiple height references were programmed and executed through the interface.
- The safety function associated with the maximum height limit was verified during operation.

III-2-4 Performance Evaluation Criteria:

Responses to this control system were analyzed using the standard performance criteria of automatic control systems [12]. These parameters were:

- Rise Time: time taken for the system to reach its desired value.
- Overshoot: maximal departure from setpoint.
- Settling Time: time needed for the response to stabilize.
- Steady-State Error: difference between the final measured value and the desired reference.

With these features in mind, we will compare the performance of P, PI and PID controllers as well as the effectiveness of the development of the didactic bench.

III-3 System Architecture and Software Design:

By combining the low-cost power of the ESP32 with sophisticated Proportional-Integral-Derivative (PID) algorithms and integrated safety protocols, this platform delivers unparalleled control and reliability. It features a complete web-based interface for dynamic tuning and historical data visualization, making complex control theory accessible and intuitive.

III-3-1 Understanding the System: A Modular Organigram

The platform is meticulously structured to ensure stability, responsiveness, and safety. The figure III-2 give an illustration through the functional organigram, explaining how the code handles each layer.

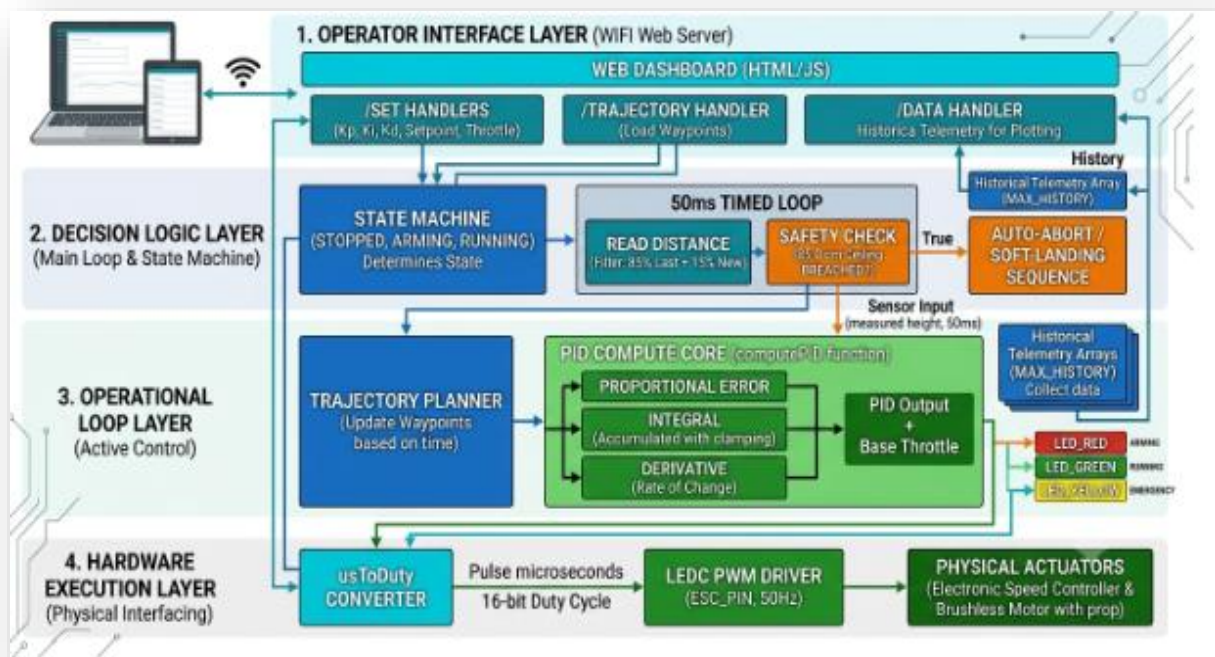


Figure III-2: system organigram: code logic and functional data flow

III-3-2 Hardware Pin Configuration:

The first section of the program defines all hardware connections used throughout the project.

Constants are assigned for the ultrasonic sensor trigger and echo pins, the Electronic Speed Controller (ESC) output pin, and the three status LEDs.

This approach centralizes the hardware configuration, making future modifications straightforward without requiring changes throughout the entire codebase. In addition, fixed constants such as the minimum and maximum ESC pulse widths, safety altitude threshold, landing altitude, and sensor sampling intervals are defined globally to ensure consistent operation across all program modules.

By separating configuration parameters from the control logic, the software becomes easier to maintain and adapt to different hardware configurations.

```
// ===== PINS =====  
#define TRIG_PIN    5  
#define ECHO_PIN    18  
#define ESC_PIN     19  
#define LED_RED     15  
#define LED_GREEN   2  
#define LED_YELLOW  4
```

Figure III-3: ESP32 hardware pin configuration

III-3-3 Wireless Communication and Embedded Web Server:

To eliminate the need for external networking infrastructure, the ESP32 is configured to operate as an independent Wi-Fi Access Point. Upon startup, it creates its own wireless network that allows any nearby computer or mobile device to connect directly.

Once connected, users can access an embedded web server hosted entirely on the ESP32. This web interface serves as the primary Human-Machine Interface (HMI), providing controls for:

- Starting and stopping experiments.
- Resetting the controller.
- Selecting P, PI, or PID operating modes.
- Adjusting controller gains.
- Modifying the base throttle.
- Configuring trajectory waypoints.
- Displaying live telemetry.
- Visualizing system performance through real-time graphs.

Hosting the interface locally on the microcontroller removes dependency on cloud services or external software and enables fully portable operation.

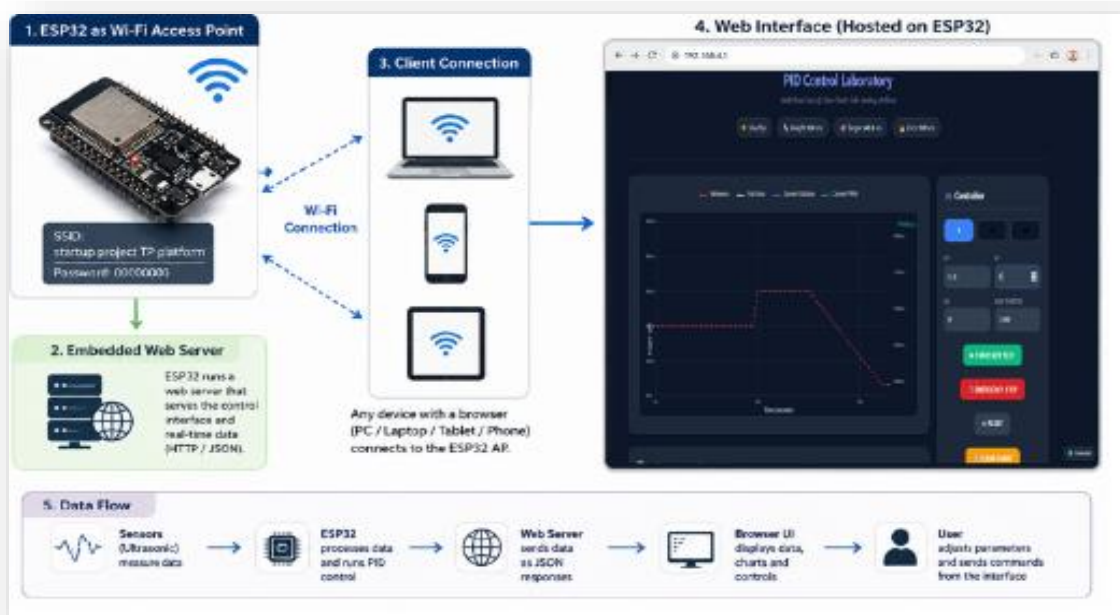


Figure III-4: wireless communication and web server architecture

III-3-4Trajectory Planning and Waypoint Management:

Rather than limiting experiments to a single fixed altitude, the software supports programmable trajectories composed of multiple waypoints.

Each waypoint specifies:

- A target height.
- A duration for which that height should be maintained.

During execution, the controller automatically transitions between waypoints based on elapsed time while continuously updating the active setpoint used by the PID controller.

This functionality enables dynamic experiments involving step responses, multi-level altitude profiles, and sequential positioning tasks without manual intervention.

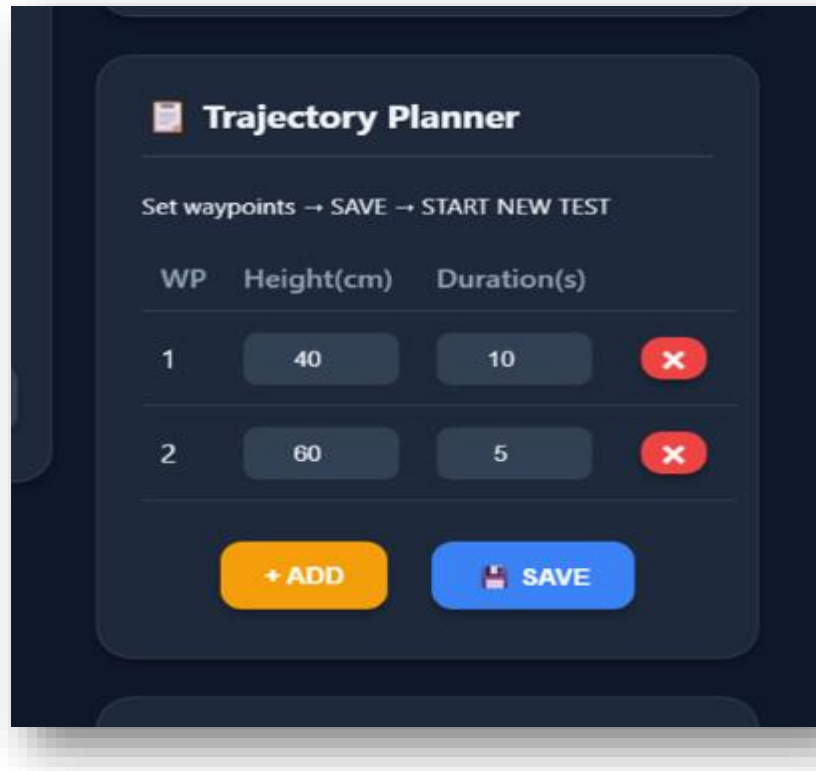


Figure III-5: trajectory planner web interface

III-3-5 Automatic Arming Sequence:

Before motor activation, the software performs a controlled ESC arming procedure.

When the user initiates an experiment, the ESC is first commanded to its minimum throttle value for a predefined delay period. Only after successful completion of this initialization does the system transition into active control mode.

This staged startup prevents unexpected motor behavior and ensures compatibility with standard brushless ESC initialization requirements.

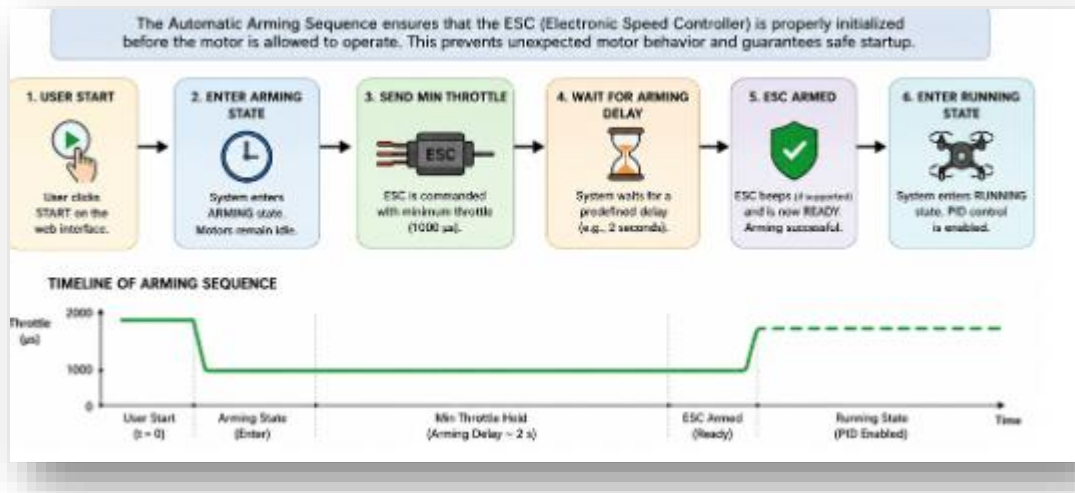


Figure III-6: automatic arming sequence and timeline

III-3-6 Automatic Soft Landing Mechanism:

To ensure safe shutdown, the project incorporates an autonomous soft landing procedure.

Whenever the user stops an experiment or a mission completes successfully, the controller does not immediately disable motor power. Instead, the ESC command is gradually reduced according to the measured altitude, producing a controlled descent.

As the platform approaches the landing zone, the descent rate decreases further to minimize impact before the motor is finally commanded to its minimum operating value.

This strategy protects both mechanical components and experimental equipment from abrupt drops.

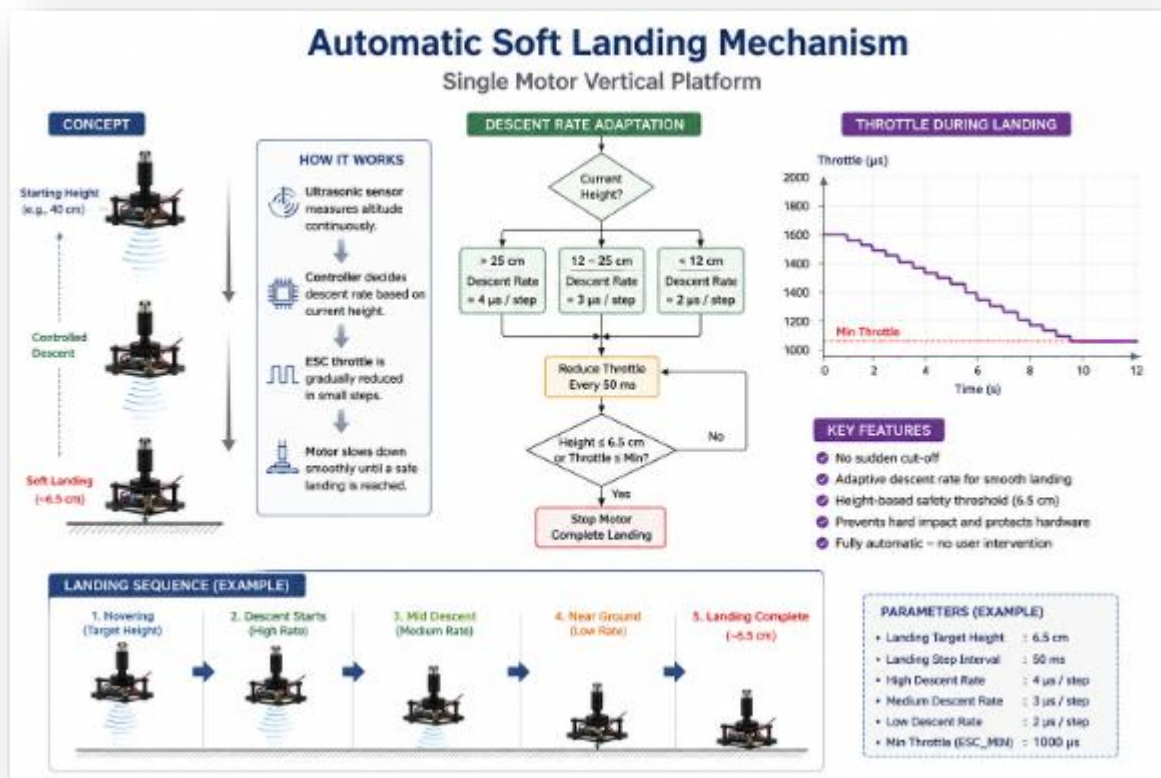


Figure III-7: automatic soft landing mechanism

III-3-7 Automatic Mission Completion and Auto-Abort:

Once all programmed trajectory waypoints have been executed, the software automatically initiates the landing procedure without requiring user intervention.

This auto-abort mechanism ensures that experiments terminate in a predictable and repeatable manner while preventing unnecessary hovering after the trajectory has been completed.

By automating the shutdown sequence, the system improves operational safety and experimental consistency.

III-3-8 Emergency Ceiling Protection:

One of the most important safety features implemented in the software is the altitude ceiling protection mechanism.

A maximum allowable height of 85 cm is continuously monitored. If the measured altitude exceeds this threshold, the controller immediately exits normal operation and initiates an emergency landing sequence.

During this process:

- Normal PID control is suspended.
- Emergency status flags are activated.
- Visual indicators are updated.
- Controlled descent begins automatically until a safe landing is achieved.

This safeguard prevents excessive vertical movement that could damage the apparatus or surrounding equipment.

III-3-9 LED-Based Visual Status Indication:

Three LEDs are incorporated to provide immediate visual feedback regarding system status.

Different illumination and blinking patterns communicate conditions such as:

- Standby mode.
- ESC arming.
- Normal PID operation.
- Soft landing.
- Emergency shutdown.

This simple interface enables rapid assessment of system state even when the web interface is unavailable.

III-3-10 Telemetry Tracking: Mathematical Performance Metrics.

For formal academic validation, the platform calculates real-time analytical performance indices directly inside the firmware loop. Utilizing the discrete time step (dt), the system actively accumulates data for three fundamental error integrals:

- **ISE (Integral of Squared Error):** Penalizes large tracking errors heavily, indicating how aggressively the controller drives towards the target

$$ISE = \int e^2 dt \quad (1)$$

- **IAE (Integral of Absolute Error):** Measures the total cumulative deviation over the course of the test run

$$IAE = \int |e| dt \quad (2)$$

- **ITAE (Integral of Time-weighted Absolute Error):** Multiplies absolute error by time, placing a heavy mathematical penalty on long-term steady-state oscillations

$$ITAE = \int t|e| dt \quad (3)$$

To objectively assess controller performance, the software computes several standard control engineering metrics during each experiment.

These include:

- **Rise Time:** The time required for the response to reach approximately 90% of the target value.
- **Settling Time:** The time required for the response to remain within a specified tolerance band around the reference.
- **Peak Time:** The instant at which the maximum response occurs.
- **Percent Overshoot:** The extent to which the response exceeds the desired setpoint.
- **Steady-State Error:** The remaining error after transient effects have diminished.
- **Integral of Squared Error (ISE):** A cumulative measure that emphasizes large deviations.
- **Integral of Absolute Error (IAE):** A measure of total accumulated tracking error.
- **Integral of Time-weighted Absolute Error (ITAE):** A performance criterion that increasingly penalizes errors that persist over time.

These indicators provide quantitative evidence of controller quality and facilitate systematic comparison between different tuning parameters.

III-4 Performance Evaluation Criteria:

III-4-1 Rise Time:

Rise time measures how quickly a system can respond to changes and if it has a fast enough speed for its required function. The smaller the rise time, the quicker the controller reacts to variations or alterations in the reference.

III-4-2 Overshoot:

Overshoot is defined as the measurement of how far above the desired setpoint the output will deviate during the transient response.

III-4-3 Settling Time:

Settling time is used to evaluate how long it takes for the system to return to a stable operating condition after a disturbance or a change in the setpoint.

III-4-4 Steady-State Error:

Steady-state error is defined as the difference (steady-state value) between the desired reference (set-point) and the output of the system.

III-4-5 Stability:

Stability is determined by the extent to which the system can provide a bounded output response without overshoot) and/or has minimal oscillation during operation.

These criteria were used to compare the performance of the P, PI, and PID controllers implemented on the developed didactic platform. [12]

III-5 Experimental Results and Discussion:

III-5-1 Influence of the Proportional Gain:

K_P=2, K_I=0, K_D=0



figure III-8: the response of the system using proportional control

Results confirm that controlled axis is significantly below desired 40 cm setpoint; even though error increases PWM signal, system can never reach reference value; large steady-state error indicates that proportional action does not provide sufficient compensation for system dynamics to provide accurate position control.

III-5-2 Influence of the Integral Action:

KP=2; KI=0.5



Figure III-9: the response of the system using proportional integral control “kp=2; ki=0.5”

Including an integral controller into the feedback loop of a closed loop control system will result in improving the output around the reference value and reducing the steady state error. However, the system will still oscillate, indicating that the system is not accurately tracking the reference value.

KP=2; ki=1



Figure III-10: the response of the system using proportional integral control “kp=2; ki=0.5”

The steady-state error has been reduced from its original value and there is only a small amount of overshoot, as the system is tracking very closely to the desired set point. The stability of the system has also been maintained while providing an acceptable tradeoff between accuracy and stability based on the smoothness of the system response.

III-5-3 Influence of the Derivative Action:

KP=2; KI=1.4; KD=1



figure III-11: the response of the system using proportional integral Derivative control “ $k_p=2$; $k_i=1.4$; $k_d=1$ ”

Adding a derivatives component to include dampening has stabilized the dynamic response to become less oscillatory than that of prior tests, yielding an enhanced level of stability to the overall dynamic characteristics. Derivative action also enhanced the transitory performance and offered smoother method of tracking the reference signal.

$$K_p = 2, K_i = 2, K_d = 2$$



Figure III-12: the response of the system using proportional integral Derivative control
“ $k_p=2$; $k_i=2$; $k_d=2$ ”

Testing was done with increased derivative gain in conjunction with increased proportional and integral gains. Although the reference trajectory was still being followed by the system, both the altitude and PWM signal exhibited significant fluctuations. This type of behavior represents that high gain settings for a set of controllers can lead to increased measurement noise and decreased overall performance.

III-5-4 Comparative Analysis of P, PI and PID Controllers:

Through experimental tests, the effects of each of the control actions were recognized on the behavior of the developed platform.

With the proportional controller, a quick response was produced, but the steady-state error could not be fully removed. Increasing the proportional gain resulted in quicker responses, although this also produced more oscillations.

The addition of the integral control action reduced the steady-state error while improving the accuracy of tracking. Excessive integral gain would, however, create greater overshoot and oscillatory behavior.

The derivative control action improved the system damping and helped to correct oscillations. Thus, the PID controller provided a more stable and accurate response than either the P controller or the PI controller.

The PID controller was the best compromise between speed of response, stability and accuracy, and was therefore the most appropriate control strategy for the newly developed educational platform.

Criterion	P	PI	PID
Rise Time	fast	Moderate	Fast
Overshoot	Medium	High	Low
Settling Time	Medium	Long	Short
Steady-State Error	Present	Very Low	Very Low
Stability	Moderate	Moderate	Good

Table III-1: Comparative Analysis of P, PI and PID Controllers

III-6 Educational Contribution of the Developed Platform:

Besides its control performance, the developed platform has a great educational value. The system allows students to visualize the practical behavior of P, PI and PID controllers and to directly compare their effects on the dynamic response of a real process.

Together with a web-based monitoring interface, it allows for a real-time visualization of the measured altitude, the reference trajectory and the control signal. Also, controller parameters can be tuned directly through the interface without the need for ESP32 reprogramming, thus making the experiments faster and more interactive.

The trajectory management and safety functions also add to the educational interest of the platform, enabling users to carry out advanced experiments in safe operating conditions. Therefore, the developed system is an effective didactic tool for teaching automatic control, embedded systems and real time monitoring techniques.

III-7 Conclusion:

In this chapter, the developed didactic PID control platform was experimentally evaluated under different controller configurations. The obtained results showed the influence of the proportional, integral, and derivative actions on the system response. Among the tested configurations, the PI controller provided the best overall performance in terms of accuracy and stability. The experiments also confirmed the effectiveness of the web-based monitoring interface for real-time observation and controller tuning. Overall, the developed platform successfully achieved its educational and technical objectives and proved to be a useful tool for studying automatic control systems.

The startup part

1-Project Presentation:

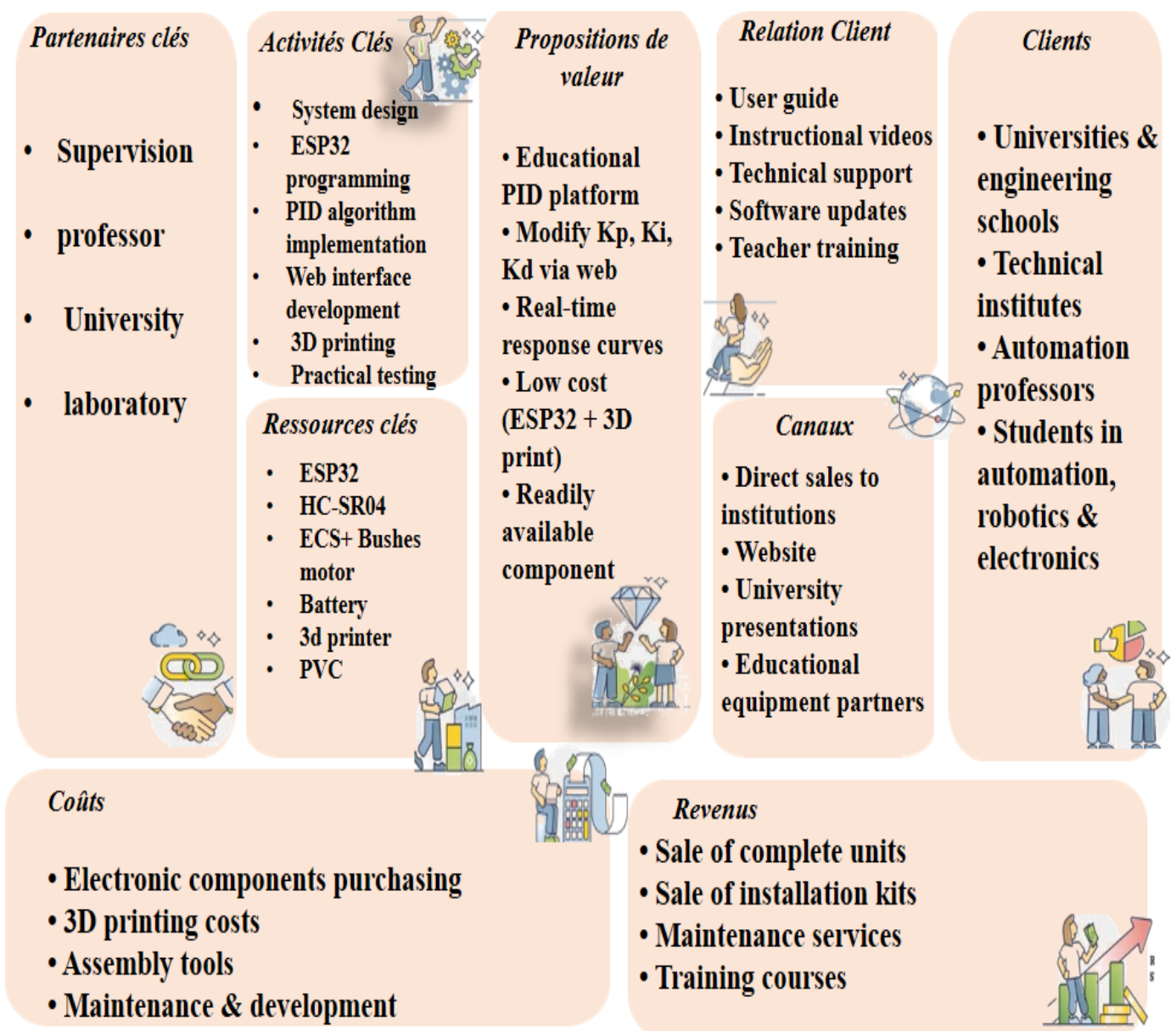
• Product:

ESP32 Based Intelligent Didactic PID Control Platform

• Description:

It is an educational platform developed to teach automatic control, PID regulation, embedded systems and real time monitoring with practical experimentation.

2-BMC:



3-Business plan:

Back to



INVESTMENT

	year 01	Total Period
computer	100,000	100,000
3d printer	30,000	30,000
desk	30,000	30,000
swivel chairs*2	60,000	60,000
guest cabinets*2	8,000	8,000
maintenance tools	30,000	30,000
first aid kit	6,000	6,000
large table	60,000	60,000
air conditioners	54,000	54,000
Equipment N°10	0	0
Equipment N°11	0	0
Equipment N°12	0	0
Equipment N°13	0	0
Equipment N°14	0	0
Equipment N°15	0	0
Equipment N°16	0	0
Equipment N°17	0	0
Equipment N°18	0	0
Equipment N°19	0	0
Sub-total (01)	378,000.00	378,000

DIRECT PURCHASES

	Prestation	Total
Product / service N°01		720,000
Product / service N°02		160,000
Product / service N°03		1,800,000
Product / service N°04		1,400,000
Product / service N°05		4,520,000
Sub-total (03)		8,600,000

TOTAL PAYROLL

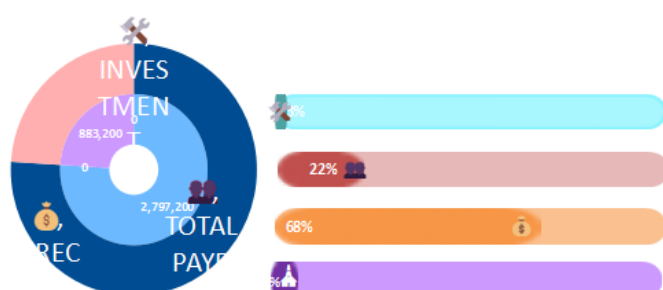
	FTE	Total
project manager	1.00	1,209,600
technical manager	1.00	907,200
electrician	1.00	680,400
Position N°04	0.00	0
Position N°05	0.00	0
Position N°06	0.00	0
Position N°07	0.00	0
Position N°08	0.00	0
Position N°09	0.00	0
Position N°10	0.00	0
Sub-total (2)		2,797,200

EXTERNAL CHARGES

	Wording	Total
Outsourcing		30,000
Rent		300,000
Energy/water/gas		66,000
Marketing expenses		30,000
Lawyer's fees		0
Notary's fees		0
Accountant's fees		0
Auditor's fees		0
Transit charges		400,000
Telecom costs		37,200
Various supplies		20,000
Training fees		0
R&D		0
Sub-total (04)		883,200.00

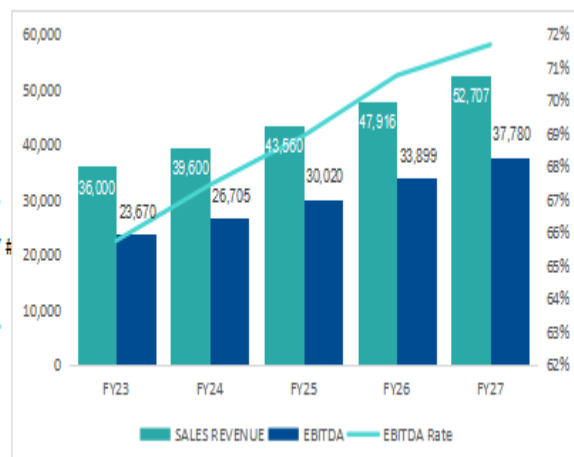
OVERVIEW

Category	Amount	
INVESTMENT	378,00	3%
TOTAL PAYROLL	2,797,200	22%
DIRECT PURCHASES	8,600,000	68%
EXTERNAL CHARGES	883,200	7%
	12,658,400	



Financial KPIs

	FY23	FY24	FY25	FY26	FY27
SALES REVENUE	36,000	39,600	43,560	47,916	52,707
EBITDA	23,670	26,705	30,020	33,899	37,780
EBITDA Rate	66%	67%	69%	71%	72%
FCF	23	26,705	30,020	33,899	37,780



General conclusion:

This dissertation presents work on the design and creation of an intelligent didactic platform aimed at teaching about PID control systems. The overall goal was to create a teaching tool that can show how automatic control works by providing a hands-on, interactive experience.

To achieve this goal, the project included a review and application of theoretical foundation for control systems, PID controllers, and embedded systems in order to create an experimental bench using an ESP32 microcontroller. The platform is comprised of several components including an ultrasonic sensor, brushless motor, electronic speed controller (ESC), and a web-based monitoring interface creating a system that provides real-time control and monitoring.

Testing of the experimental platform demonstrates that the proposed solution is valid and provides insight into how changing various controller settings affects the performance of the system. Thus, the experimental platform can serve as a very useful educational tool to learn about and experiment with control strategies.

Along with providing the technical components of a project, this project was developed in an entrepreneurial context. The evidence from the models showed that there is a good opportunity for commercializing this type of product within universities, training centers, and educational labs; they are currently looking for affordable and effective means to implement their learning approach.

Ultimately, this project met all of its goals of creating a combined solution of engineering, innovation, and business in one system by providing a quality educational experience and being a viable startup in the automated/embedded systems industry.

The platform can be further developed by using advanced control approaches; implementing mobile applications, connecting to the internet (cloud-based), and adding new modules can help expand the number of capabilities and benefits this product can provide.

Bibliographies :

- [1] P. PROUVOST, « Instrumentation et régulation en 30 fichiers », édition Dunod, Paris, France, 2010
- [2] D. LEQUESNE, « La régulation PID analogique et numérique », édition Eyrolles, France, 2006.
- [3] S.Sekhsokh et K.Oukili «Etude d'une boucle de régulation de niveau», mémoire de master en génie des procédés, école supérieure de technologie _ FES _ 2010/2011.
- [7] M.Kareb et L.Gadi, «Étude et implémentation d'une commande PID en utilisant le PIC18F2550 pour le contrôle de vitesse d'un MCC», mémoire de master en automatique, UMMTO, Tizi-Ouzou, 2016.
- [9] E.Magarotto, «cours de régulation», département de génie chimique et procédés, université de Caen, France, version septembre 2004.
- [10] C.Bessieres, « Systèmes asservis analogiques et échantillonnés», cours, France.
- [11] W.Aous et G.Sedaoui «conception et réalisation d'un régulateur PID numérique à base d'un microcontrôleur PIC16F877A», mémoire de master en électronique industrielle, UMMTO, Tizi-Ouzou, 2014.
- [12] K. Ogata, Modern Control Engineering, 5th Edition, Prentice Hall, 2010.
- [14] K. J. Åström and T. Hägglund, PID Controllers: Theory, Design and Tuning, 2nd Edition, Instrument Society of America (ISA), 1995.
- [15] B. C. Kuo and F. Golnaraghi, Automatic Control Systems, 9th Edition, Wiley, 2009
- [16] G. F. Franklin, J. D. Powell and A. Emami-Naeini, Feedback Control of Dynamic Systems, 7th Edition, 2015.
- [17] M. J. Pont, Embedded C, 2nd Edition, Pearson Education, 2002
- [18] K. J. Åström and B. Wittenmark, Computer-Controlled Systems: Theory and Design, 3rd Edition, Prentice Hall, 1997.
- [19] M. Banzi and M. Shiloh, Getting Started with Arduino, 4th Edition, Maker Media, 2022

[22] J. Boxall, *Arduino Workshop: A Hands-On Introduction with 65 Projects*, No Starch Press, 2013.

[23] R. Krishnan, *Permanent Magnet Synchronous and Brushless DC Motor Drives*, CRC Press, 2017.

[24] A. I. Pressman, K. Billings and T. Morey, *Switching Power Supply Design*, 3rd Edition, McGraw-Hill, 2009.

Webography:

[5] <https://en.wikipedia.org/wiki/Proportional>

[6] [https://fr.wikipedia.org/wiki/Asservissement_\(automatique\)#Articles_connexes](https://fr.wikipedia.org/wiki/Asservissement_(automatique)#Articles_connexes).

[8] [https://fr.wikipedia.org/wiki/Régulation automatique](https://fr.wikipedia.org/wiki/R%C3%A9gulation_automatique).

[13] <https://www.bh-automation.fr/Ressources/Automaticiens>.

[20] <https://www.elprocus.com/what-is-a-closed-loop-control-system-its-working/>.

Accessed: Jun. 2026.

[21] <https://fr.wikipedia.org/wiki/ESP32>