www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 16(4), Sep-Dec 2025, 61-78. ISSN: 2007-1558. https://doi.org/10.61467/2007.1558.2025.v16i4.1133

Model of a Low-Cost Educational Prototype of the Furuta Pendulum

Emmanuel Martínez Ramírez [1], Marcelino Alan Diaz Ledezma [1], Zuarth David López Portillo [1], Alex Yoel Vicente Ortiz [1], Nelson Rosendo Mera Sánchez [1], Mario Oscar Ordaz Oliver[1], María Angélica Espejel Rivera [1]

[1]Departamento de Ingeniería Eléctrica y Electrónica, Tecnológico Nacional de México, Campus Pachuca, México.

l22200604@pachuca.tecnm.mx,l22200588@pachuca.tecnm.mx,l22200602@pachuca.tecnm.mx, l22200627@pachuca.tecnm.mx, l22200609@pachuca.tecnm.mx, mario.oo@pachuca.tecnm.mx, maria.er@pachuca.tecnm.mx

Abstract. Didactic control systems are tools that facilitate the understanding of topics related to control and stabilisation. However, due to the high cost of these devices, their implementation in schools is limited. This work presents the design, construction, and testing of a low-cost didactic prototype of a Furuta-type pendulum for the analysis of topics related to this system. The construction included parts designed in SolidWorks and produced with 3D printing, threaded rods, a precision linear potentiometer, a DC motor, a BTS7960 H-bridge, and an Arduino Nano. The functionality of the project was verified through the implementation of two digital tools: a Kalman filter to improve the measured signal and a PID controller to regulate the angular speed of the motor. It was demonstrated that the prototype can be stabilised under the operational considerations of PID control without the use of an angular position sensor. Moreover, the Kalman filter must be kept small in order to avoid information loss.

Keywords: PID control, digital tools, Arduino, model, stability, system.

Article Info
Received May 8, 2025
Accepted July 15, 2025

1. Introduction

The Furuta Pendulum is a underactuated system with two degrees of freedom, commonly used in control theory due to its high instability and the challenge it represents. These systems are generally expensive, mainly due to the components used in their design. It is well known that not all students have the financial means to manage such projects. Therefore, it has been proposed to develop a low-cost Furuta Pendulum as a didactic tool in order to support students both theoretically and practically, enabling them to implement some of the strategies carried out in this work.

Below is a brief description of the background related to the development of the Furuta Pendulum. Álvarez Serrano (2021) chose to implement two control techniques, PID and state feedback, using FPGA and Arduino electronic boards. Both controllers responded correctly; however, the performance was superior on the FPGA platform, as it operates in parallel, while Arduino does so sequentially. It was concluded that state feedback is optimal when a model of the system is available, while PID is more suitable when such a model is not available. Similarly, in the work of Calderón Agudelo (2022), the stabilization of the Furuta Pendulum was studied using a PID controller and a fuzzy controller. The first showed an adequate response in simulations and good performance against disturbances, while the second presented better results in the face of disturbances, with smoother transitions. It was also demonstrated that a neural network was capable of effectively emulating the action of the controller to balance the pendulum. Likewise, Cárdenas Castañeda and Castiblanco Castañeda (2022) built a mathematical model of the Furuta Pendulum using a Newtonian analysis to characterize its nonlinear dynamics. The model was validated through simulation tools such as Simulink, MATLAB, and Python. Subsequently, they implemented experimental validation using the Hardware-in-the-Loop technique, conducting multiple tests in Simulink. It was identified that the PID controller was insufficient to maintain the vertical position of the pendulum during intervals of less than 1 second, due to the high inherent instability of the system. For their part, Guo et al. (2023) analyzed the stability of the Furuta Pendulum system, finding a solution to achieve full control of the internal and external loops, even when PD control does not provide stability. The authors demonstrated that LQR control can stabilize the pendulum arm more quickly and effectively, reducing both oscillations and their frequency. In this context, the work of Figuerola Aboy (2023) also implemented an optimal LQR control to stabilize the Furuta Pendulum in its resting configuration. The control

showed good results, even achieving stabilization through the lift control law.

On the other hand, the study by Rodríguez Rivera & Ruiz Bravo (2020) implemented several controllers, such as LQR, a state observer, gain scheduling, and a swing-up control, to control the Furuta Pendulum. These controls were virtually implemented on platforms such as Simulink and V-REP. It is noted that the V-REP platform showed better performance in stabilizing the system, while in Simulink it was not possible to control the virtual plant due to instability caused by linear observers. Following this line of research, the study by Mérida Rubio et al. (2020) aimed to stabilize the Furuta Pendulum using LQR control in the inverted position of the Furuta Pendulum, following the input reference signal. The control was designed so that the undisturbed system is stable in the inverted position, as long as the input signal is adequate. The results were obtained using software such as the Simscape Toolbox along with Simulink, allowing a 3D model. Likewise, Velásquez Marcelo (2024) proposed controlling the Furuta Pendulum by calculating the LQR feedback constant through matrices obtained from the Euler–Lagrange equations. Similarly, in the work of Moura et al. (2024), the nonlinear dynamics were obtained through the Euler–Lagrange formulation. These dynamics were used to feed a linear model, on which linear feedback control was applied. This control was applied both to the nonlinear dynamics and the virtual prototype, demonstrating through the obtained data that the system can be stabilized with this strategy.

Complementarily, Díaz et al. (2024) proposed a new controller design methodology based on observation for plants in Lagrange–Euler form. They provided examples illustrating the advantages of the proposed methodology, investigating its robustness as it allows lower energy consumption and lower sensitivity to noise, due to reduced dependence on measured signals from the plant. Some limitations include the convex rewriting of nonlinear expressions and the factorization of error signals, which are not unique. The feasibility of the LMIs depends on these rewritings. Furthermore, this approach requires more steps compared to traditional methodologies. Likewise, Martínez-Velázquez et al. (2021) developed a methodology to observe nonlinear systems. The proposal begins with the factorization of the error signal without using estimates or Lipschitz bounds. Once the appropriate error signal is obtained, it is rewritten in a convex and exact form. The observer, designed under LMI conditions, was validated using the Furuta Pendulum system. In addition, Hernández-Cortés et al. (2024) decided to propose a stabilizer obtained through LMI. The method proved effective in nonlinear systems, generating a bounded error that can be reduced through unmodeled trajectories suitable for the plant. The results were validated on the Furuta Pendulum system.

Following a different approach, Martínez et al. (2023) developed a MADRPC controller (Model-Assisted Predictive Control with Disturbance Rejection), which adequately estimates and adjusts the mismatch between the actual controlled process and an assumed prediction model, like a first-order dynamic plus an integrator. This strategy ensures system stability and feasibility. The controller was implemented in a multibody environment in Simscape 3D, which accurately reproduces the dynamics of the Furuta Pendulum. The results showed that the MADRPC strategy is valid for a mechanical plant, extending its applicability to SIMO processes. On the other hand, in the work by Coria et al. (2022), the use of Genetic Programming tools was proposed to stabilize a Furuta Pendulum. To do this, GP uses data obtained from a sliding mode controller, where it was determined that it is possible to improve a controller's performance through the correct application of Genetic Programming.

In the study by Acosta Villamil et al. (2021), they chose to use a NNARMAX model to mathematically represent the system dynamics from data. From the model, control strategies such as pole placement, optimal control, and minimum variance were developed. Similarly, Prado et al. (2020) developed and implemented a nonlinear model predictive control in tubes (T-NMPC), managing to balance the pendulum, stabilize the inverted position, and follow predefined trajectories, maintaining robustness against external disturbances. Unlike the above, in the work of Montes de Oca (2023), it was concluded that the implemented algorithms made it possible to confirm that reinforcement learning is a viable solution in control problems where the system model is unknown. It also highlighted the correct selection of the reward signal as a key factor in improving training and achieving functional control. Meanwhile, the study by Méndez et al. (2020) demonstrated an approximation methodology for UMS, which offers a fast system response and lower computational cost based on its mathematical expression. It was concluded that the method allows simpler implementation and even improvements in models through artificial intelligence. In contrast, Antonio-Cruz et al. (2023) presented a nonlinear controller with friction compensation to achieve a complete swing-up based on energy for a Furuta Pendulum subject to dynamic friction. To achieve the complete swing-up of the Furuta Pendulum, they did so in a short period of time, feasible for experimental implementation, and provided a formal analysis of the closed-loop system stability. Finally, Chen et al. (2020) designed a control system that integrated the trajectory tracking of the external subsystem and the trajectory stabilization of the internal subsystem. The trajectory of the external subsystem was designed using a Model Predictive Control (MPC) approach, while an inverse dynamics controller was used to simultaneously stabilize the trajectory of the internal subsystem. Gaussian Processes (GP) were used to estimate the system dynamics and provide a predictive distribution of

uncertainties, ensuring closed-loop control stability and performance. In contrast to the reviewed works, this project proposes the development of the Furuta Pendulum as a didactic tool, aiming to achieve a low cost in most of the design. The goal is to stabilize the Furuta Pendulum through PID control, as well as implement a Kalman Filter due to noise that may be generated, in order to improve the plant's functioning. Both the control and the filter will be programmed and implemented using the Arduino platform.

Therefore, the following strategy is implemented to achieve the development of the low-cost Furuta Pendulum, consisting of a series of aspects that allow optimizing the design, with the essential use of low-cost materials. Created using 3D printing, with a very limited number of parts, and screws used to maintain the rigid structure. In addition, a PID control is programmed, which must calculate the error between the desired and estimated position of the Pendulum. For the PID programming, trapezoidal integration is used. The control result is applied directly to the motor shaft. Regarding the Kalman filter, it is programmed in its simplified or first-order form to improve the Pendulum's sensor signal, which combines a prediction based on the previous state and a correction based on the new measurement. Its simplified form does not require matrices or complex operations. In both cases, the Arduino platform is implemented to program the filter and the PID, as it is an economical, accessible, and commonly used option in the academic field. The implementation of both strategies is carried out to improve the overall performance of the control system. The filter eliminates the inherent noise from the sensors, allowing the control to calculate a more effective control signal by having a cleaner signal. A fundamental aspect of system stabilization is tuning, which will be carried out using the heuristic method, which consists of manually adjusting the PID control parameters (kp, kd, ki), testing better constants, taking the system's behavior as a reference. The method is very accessible and economical, requiring less use of software or complex mathematical models.

The combination of these objectives, such as low-cost design and construction, implementing PID control, applying the Kalman filter, and performing heuristic tuning, aims to achieve the stabilization of the Furuta Pendulum, allowing its use as an affordable and accessible educational tool.

It is well known that control subjects are fundamental in the development of future students, mainly university students in engineering careers. The theoretical aspect is important in student development, but practical procedures add weight to understanding the fundamentals. However, the application of controls to underactuated systems, such as the Furuta Pendulum, requires significant financial investment from students. It is known that not all students can afford the development of systems with high economic value. At the same time, the use of highly rigorous prototypes creates both intellectual and economic barriers. This results in a contrast between performing well academically or facing financial limitations. That is why this alternative was developed to provide students with the necessary tools to optimally develop their knowledge.

Thus, the development of this project, which aims to achieve a low-cost design through the use of appropriate materials, also highlights the decision not to use an encoder to measure the motor's signal. This is an important aspect of the research, as the previously studied works mention the use of this encoder. This is why the relevance of this work lies in this distinction.

The following sections to describe the organization of the present work: In the Methodology section, to define the concept of the Furuta Pendulum, its background, and the approach to take as a teaching tool in this work. Additionally, the factors considered in the design and construction of the prototype. In the Results and Discussion, to present the performance indices obtained from the Furuta Pendulum, along with a critical analysis of the aspects needing improvement and, finally, the aspects effectively achieved in the development. Finally, to present the Conclusions, where to summarize the results obtained, indicating whether the proposed objectives were achieved, specifically if the Furuta Pendulum was stabilized and developed at low cost.

2. Methodology

Among underactuated systems, the most commonly used and extensively studied example is the inverted pendulum. This system consists of a cart moving along a straight line on a rail of limited length, a pendulum hinged to the cart that moves within the plane of that line, and a pulley-based transmission system connected to a direct current motor. (Mori et al., 1973)

Nevertheless, inverted pendulum systems present the drawback of a bounded cart travel range, which may restrict the execution of control maneuvers. To overcome this limitation, a modification of the original system was proposed by replacing the cart's linear displacement with a closed circular trajectory. This led to the development of the so-called Furuta pendulum, an underactuated system with two degrees of freedom. (Valera et al., 2025)

The experimental study of this system was introduced in 1991 by Katsuhisa Furuta at the Tokyo Institute of Technology, under the name "TITech Pendulum." The rotational inverted pendulum system, or Furuta pendulum, developed by Dr. K. Furuta, is defined as an underactuated system with only two degrees of freedom, both rotational, referred to as the arm and the pendulum. (Calderón & J, 2022)

An underactuated system is characterized by its inability to directly control all of its motions, which makes its control more challenging. This complexity turns it into a classic case for applying linear and nonlinear control theories. These systems are particularly relevant because, through the use of robust control, it is possible to govern their behavior using fewer actuators than degrees of freedom, which in turn results in improved energy efficiency. (Valera et al., 2025; Arias, 2018)

The objective of this project is to validate control theory as applied to a dynamic system. In the case of the Furuta pendulum, a physical system is used that is commonly employed for teaching and studying automatic control due to its complexity and nonlinear behavior.

Validating the control theory involves:

- Verifying the controller's effectiveness: Checking whether the selected control system (PID) allows the pendulum to remain balanced and stable.
- Evaluating the system's behavior: Observing if the system's response to the controller's inputs, such as variations in the pendulum's angle or velocity, is adequate.
- Adjusting the parameters: Modifying the PID gains if optimal control is not achieved, in order to improve performance. • Demonstrating control theory concepts: Illustrating how to apply principles such as feedback, stability, and response time in a real physical system.

To design and construct a Furuta pendulum, it is necessary to integrate a set of mechanical, electronic, and control components (Fig. 1) that enable the study and validation of advanced stabilization and regulation techniques for nonlinear systems. This kind of pendulum is an excellent didactic test bench for promoting studies in Automatic Control, Robotics and System Dynamics, since it presents an inherent instability and it has to do with control algorithms that are developed in a practical context.

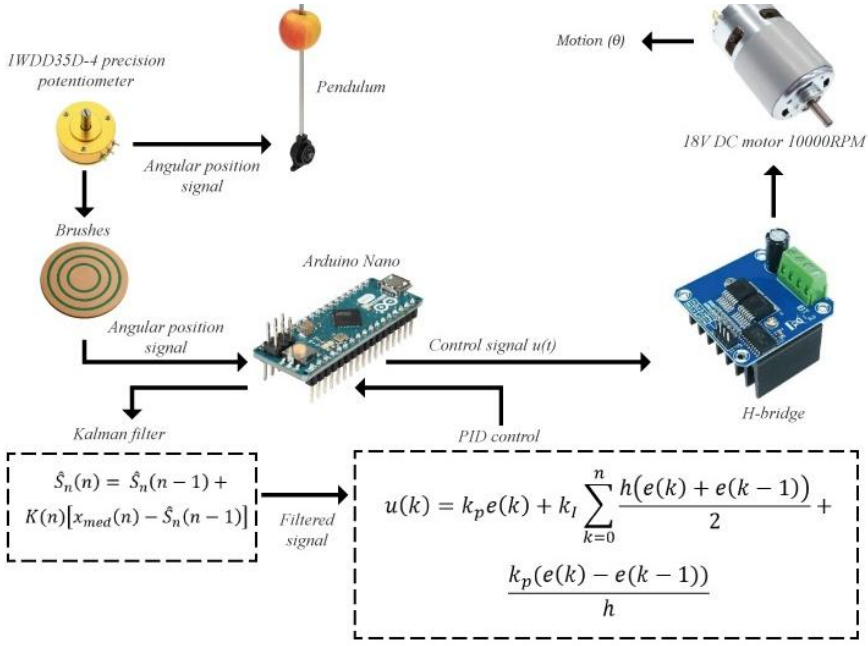

Fig. 1. Diagram of the control of the pendulum using a PID controller and Kalman filter implemented on Arduino Nano

The aim of this itemized bill of materials is to determine the necessary parts to assemble an analytical system prototype at minimum cost, but without sacrificing the system's analytical and control capabilities. Both available choices and possible improvements

to consider that can increase the model's capabilities and favor its adoption in educational and research institutions. These are the components used to build the prototype:

1. Arduino Nano, used to acquire and analyze the pendulum's position signals, and subsequently to control them to a desired position, with a discrete design to enhance aesthetics.
2. 3D prints, designed using SolidWorks software.
3. Motor of 18V and 10,000 rpm.
4. Counterweight, used to manipulate the pendulum's center of mass.
5. H-bridge BTN7960B, operating voltage of 5.5V to 26 V DC, compatible with 3.3 V and 5 V systems, output current 43 A.
6. 10kΩ precision potentiometer, employed for sensing angular displacement, with an error margin of 0.5%.
7. Fan, used to allow the H-bridge to reach its maximum operating current while maintaining a low temperature.
8. Screws, used to join the 3D-printed parts, maintaining structural rigidity.

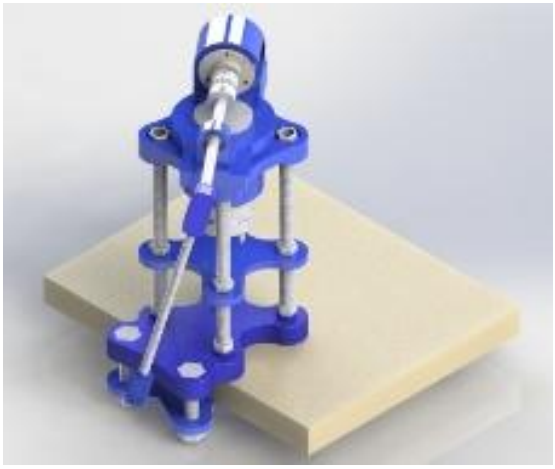

Fig. 2. Design of the Furuta pendulum created in SolidWorks.

The prototype of the Furuta pendulum is designed in SolidWorks (Fig. 1), and all the parts are subsequently 3D printed to achieve high precision during assembly.

Fig. 3. Prototype of the Furuta pendulum designed and built for this work, in operation.

The Furuta pendulum prototype must be constructed with robustness, avoiding vibrations during motor actuation. As shows in Fig. 2, the base supporting the sensor (a precision angular potentiometer), the transmission shaft linking the sensor to the pendulum mass, and the mass itself, has to maintain an almost perfectly leveled alignment in order to ensure a stable and consistent equilibrium point throughout the rotation of the base.

To measure the pendulum's position error, a precision angular position potentiometer was used, installed on the pendulum's transmission shaft. The obtained measurement is sent to the Arduino platform, where it is subsequently compared with the desired position of the pendulum (equilibrium point), thus generating the error variable; then, the Kalman Filter is used in order which produces an estimate of the system state as an average of the previous system state and the new measurement using a weighted average. The purpose of the weights is to make values with a better (i.e., lower) estimated uncertainty more reliable.

Once the prototype is assembled, the PID controller's gain values are heuristically adjusted in order to observe the physical model's behavior and fine-tune these values until the pendulum is satisfactorily stabilized.

Table 1. Price comparison of this project.

Low-cost prototype		Popular brand	+$30,000.00 MXN +$1438.00 USD
Arduino Nano	$120 MXN		
3D prints	$500 MXN		
Motor DC	$300 MXN		
Counterweight	$20 MXN		
Screws	$100 MXN		
H-Bridge	$150 MXN		
Fan	$70 MXN		
Wires	$20 MXN		
Precision potentiometer	$500 MXN		
Total	$1,780.00 MXN $85.00 USD		

As can be seen in the comparison table, the low-cost model is significantly cheaper than the commercial model, with our prototype being approximately sixteen times cheaper, which fulfills the purpose of its construction.

2.1 Kalman Filter

Is an algorithm capable of process observed measurements over a defined period, including signals corrupted by noise, with the aim of estimating unknown variables with greater accuracy than could be achieved from a single measurement. Given the delay between the issuance of motor commands and the reception of sensory feedback, employing a Kalman filter enables the construction of a realistic model to estimate the current state of a motor system and to generate adjusted control commands.

To recursively estimate the state of a linear dynamic system provided with noisy measurements from uncertain information theory sensors. The algorithm works in a loop with two primary processing steps:

Prediction: From the state estimate at the previous time step, calculate the state estimate at the current time step using the system model. This estimate comes with an inherent uncertainty.

Update: Fuse in a new noisy, uncertain measurement, correlated with the current state. Compare that measurement to the earlier prediction and improve your estimate by weighing in both bits of data. Give higher priority to either the prediction or the measurement, whichever was less certain, to make a weighted estimate that was closer to the true condition than either value alone.

The mathematical expressions defining the calculation of uncertainty, estimation error, Kalman gain, and state update are presented below, following the classic approach described in Li, Q., Li, R., Ji, K., & Dai, W. (2015) & Hernández Pérez, J., Gutiérrez Moreno, E., Ordaz Oliver, J. P., & Ordaz Oliver, M. O. (2024).

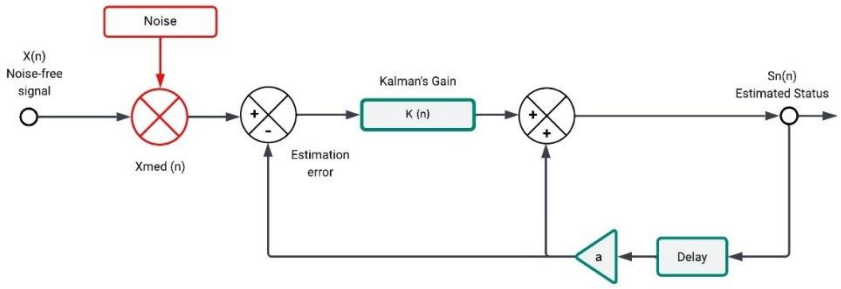

Fig. 4. Block diagram of Kalman filter.

The standard deviation is associated with the noise-free signal (σ_u). This standard deviation is related to the uncertainty of the process model, that is, to how confident one is in how the system evolves over time. This deviation is calculated by:

$$\sigma_u^2 = \sum_{k=1}^{n-1} \frac{\left(x(k) - \bar{x}_{n-1}\right)^2}{n} . \tag{1}$$

Where:

$\bar{x}$: The average value of the measurements in the absence of disturbances.

$x(k)$: Represents the value of the noise-free signal at time k, that is, each individual sample unaffected by disturbances at that specific moment.

Additionally, the standard deviation of the noisy signal (σ_n) is associated with the measurement noise covariance matrix. Knowing how noisy the signal is allows adjusting the weight assigned to the measurement during the update step. This deviation is calculated as follows:

$$\sigma_n^2 = \sum_{k=1}^{n-1} \frac{(x(k)_{med} - \bar{x}_{med})^2}{n} . \tag{2}$$

Where:

$\bar{x}_{med}$: The average value of the noisy measurements.

$x(k)_{med}$: Represents the value of the noisy signal at time k.

The mean squared error (MSE) is used as an optimization criterion to generate the most accurate possible estimates by combining noisy information in a statistically optimal manner. It is calculated as follows:

$$M_{n-1}(n) = a^2 M_{n-1}(n-1) + \sigma_u^2 . \tag{3}$$

Where:

a : Represents the prediction factor that models how the estimated state is expected to change between time steps. When $a=1$, the model assumes the state remains unchanged; if $a<1$, it models a gradual attenuation of the state.

$M_{n-1}(n-1)$: Represents the variance of the estimation error at step $n-1$, representing the uncertainty after incorporating the latest available measurement.

The estimation error quantifies the difference between the filter's prediction and the actual observed measurement. Its purpose is to correct and refine the current state estimate, improving the filter's accuracy at each iteration. This value is calculated as follows:

$$Error = x_{med}(n) - \hat{S}_n(n-1) . \tag{4}$$

Where:

$\hat{s}_n(n-1)$: Refers to the prediction of the state at step n, based solely on the information available up to time $n-1$.

The Kalman gain determines the weight assigned to the new noisy measurement relative to the prior estimate. Its value defines the extent of correction applied to the estimated state and is calculated as follows:

$$K(n) = \frac{M_{n-1}(n)}{M_{n-1}(n) + \sigma_n^2} . \tag{5}$$

$\hat{s}_n(n-1)$ serves as the foundation for predicting the next state of the system. It is calculated as follows:

$$\hat{S}_n(n-1) = \hat{S}_{n-1}(n) + K(n)\left[x_{med}(n) - \hat{S}_{n-1}(n-1) \right] . \tag{6}$$

Finally, the estimated state $\hat{S}_n(n)$ represents the best estimate of the system's true value at time n. It is calculated as follows:

$$\hat{S}_n(n) = \hat{S}_n(n-1) + K(n)\left[x_{med}(n) - \hat{S}_n(n-1) \right] . \tag{7}$$

```
// Kalman filter variables
float shat = 0.0, M = 0.0, num = 0.0, media = 0.0, rest = 0.0, a = 1.0;
float errork = 0.0, K = 0.0;
int k = 1;
float var_n = 0.0;
float sigma_u = 0.0011;
float sigma_n = 0.0;
```

The variable shat represents the current state estimation, while `M` corresponds to the variance of the associated estimation error. The latest measurement is stored in `xn`, and the innovation, denoted by `errork`, is computed as the difference between this measurement and the previous estimate. The parameter a, generally set to `a` value of 1, defines the system model used to predict the next state.

The Kalman gain `K` is calculated dynamically as a function of the process variance (`sigma_u`) and the total estimation uncertainty (`M`). The code also includes an adaptive mechanism to estimate the measurement noise variance (`sigma_n`), based on a cumulative computation of the mean (`media`) and the variance (`num`). The variable `k` functions as an iteration counter, allowing the noise estimation to be progressively refined as new measurements are incorporated.

```
// Variance compute

  var_n = xn + var_n;

  media = var_n / k;

  rest = pow((xn - media), 2);

  num = (num + rest);

  sigma_n = num / k;
```

The calculation of the variance begins by accumulating the variance in the variable `var_n`, which is incremented with the current value of `xn`. The mean (`media`) is then updated by dividing `var_n` by the iteration counter `k`. To compute the variance, the squared difference between the current measurement (`xn`) and the mean is calculated and stored in the variable rest. This value is added to the cumulative sum (`num`), which tracks the squared deviations over time. Finally, the variance (`sigma_n`) is obtained by dividing num by the current iteration `k`.

```
// Kalman filter

  shat = a * shat;

  errork = xn - shat;

  M = (pow(a, 2)) * (M + sigma_u);

  K = M / (sigma_u + M);

  shat = (shat) + (K * errork);

  M = (1 - K) * M;

  k = k + 1;
```

The Kalman filter operates by first predicting the current state (`shat`) based on the previous state and a system model represented by the parameter a. The error (`errork`) is then calculated as the difference between the current measurement (`xn`) and the predicted state (`shat`). The variance of the error (`M`) is updated using the system model, incorporating the noise variance (`sigma_u`). The Kalman gain (`K`) is computed dynamically, based on the total estimation uncertainty (`M`) and the process noise (`sigma_u`).

The state estimation (`shat`) is then corrected by adding the product of the Kalman gain (`K`) and the innovation (`errork`). Finally, the variance of the error (`M`) is updated by applying the Kalman gain, and the iteration counter k is incremented to allow for the continuous refinement of the filter as new measurements are processed.

The Kalman filter can play a decisive role in inverted pendulum control systems, as its value can significantly affect the performance of the control system. If the value is higher than necessary, it will smooth out the peaks in the position error measurements, making it impossible for the control system to detect them and, as a result, stabilize the pendulum.

2.2 Design of PID Controller

To calculate a control signal, the PID controller employs three control terms, each of which is multiplied by a corresponding gain to properly tune their contribution within the control equation.

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau)\, d\tau + K_D \frac{de(t)}{dt} \,. \tag{8}$$

The proportional part of the PID controller can be described in two components: the gain (K_P), to be multiplied by the measured signal e(t).

The integral part of the PID controller involves a gain (K_I), multiplied to the integral of the measured signal, which is calculated through the trapezoidal rule Ortiz, M., & Popov, E. P. (1985) & Alvarado Hernández, J. R., Domínguez Ortega, O., & Ordaz Oliver, M. O. (2024) in this implementation.

$$\int_0^t e(\tau)\, d\tau \approx \sum_{k=0}^{n} \frac{h\left(e(k) + e(k-1)\right)}{2} \,. \tag{9}$$

This term sums the error over time, so it has a cumulative effect. Its main function is to eliminate steady-state error.

The derivative part of the PID controller includes a gain (K_D), multiplied by the derivative of the measured signal. In this case, the derivative of the signal is obtained using Euler's method Burden, R. L., & Faires, J. D. (2011) & Chapra, S. C., & Canale, R. P. (2015).

$$\frac{de(t)}{dt} \approx \frac{e(k) - e(k-1)}{h} \,. \tag{10}$$

This term predicts future error behavior and helps smooth the system response.

It is necessary to implement numerical approximations to transform the control laws used in the pendulum into a discrete form due to digital systems like Arduino operate sequentially in finite time intervals instead of operating in the continuous time domain which is where control laws for these systems are normally formulated; with these approximations is possible to evaluate the system and apply control action at discrete instants, ensuring a practical and functional implementation.

3. Results and Discussion

This section presents the results obtained from both performance tests and observations made during the implementation of the prototype together with the control laws. These insights provide a comprehensive understanding of how the system responds in real-world conditions and how practical factors influence theoretical control performance.

In this type of pendulum, it is necessary to incorporate an encoder in order to send a signal to the motor and stop its rotation once the pendulum reaches a balanced position. However, with the goal of minimizing costs, it was decided to omit this component. For the same reason, the capability of this prototype to maintain balance despite such a disturbance is acknowledged. The presence of this disturbance not only tends to complicate the control of the system but also modifies its dynamics, as in the case of mechanical wear.

After conducting many performances testing by heuristically tuning the gains of both the PID controller and the Kalman filter, the following set of parameters was identified as yielding the best result:

Control variables: X_{des}=542.5, K_p=170, K_d=1.09, Ki=0.0, h (sampling time) = 0.001.

Kalman filter variables: a (prediction factor) =1.0, k=1, sigma_u=0.0011.

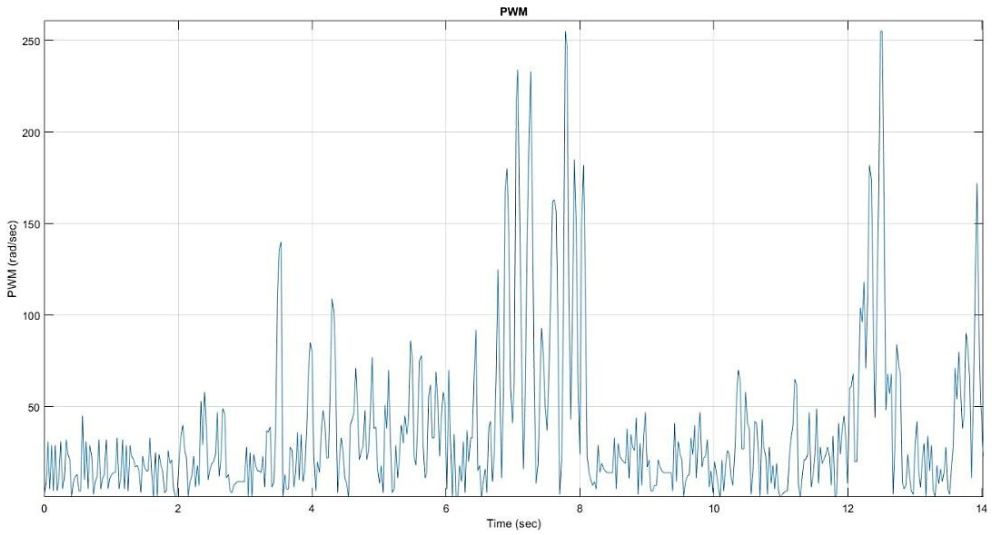

Fig. 5. Graph of the motor speed (PWM).

Graphic of the pulse-width modulation (PWM) signal generated by the controller versus time, used to displace the pendulum position. Variability in the PWM values is observed, reflecting the controller's actions to maintain the stability of the pendulum.

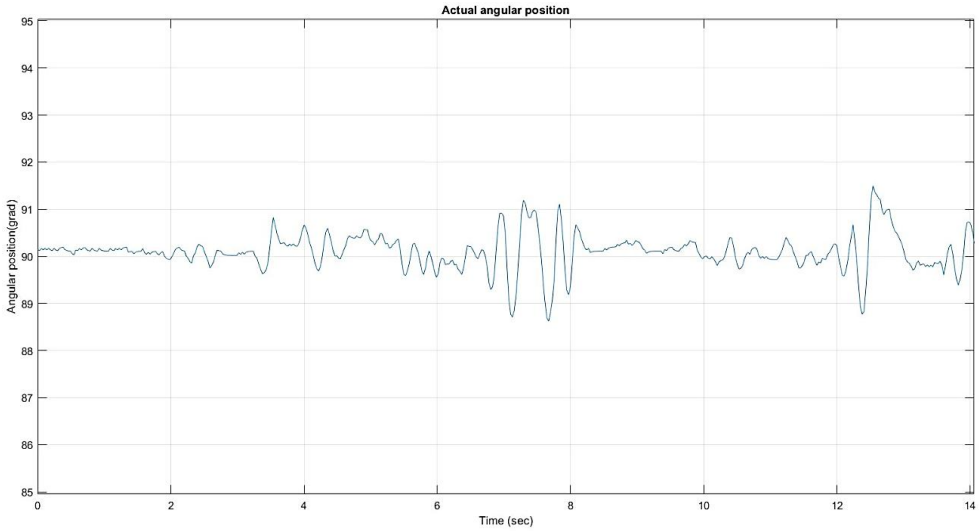

Fig. 6. Graph of the pendulum's angular position.

Graphic of the angular position of the pendulum, fluctuations around the equilibrium position indicates the ability of the control to keep the pendulum stability despite causing disturbances to it.

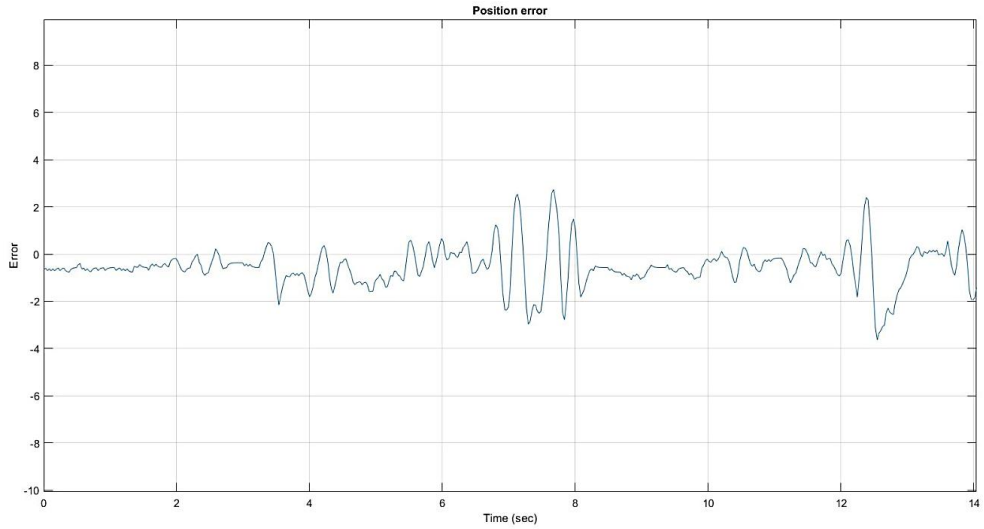

Fig. 7. Graph of the pendulum's position error.

Graphic of the pendulum's position error, a similarity is observed with Fig. 6, however, the signal is shown inverted to the actual angular position, demonstrating the system's capability to accurately estimate the position error.

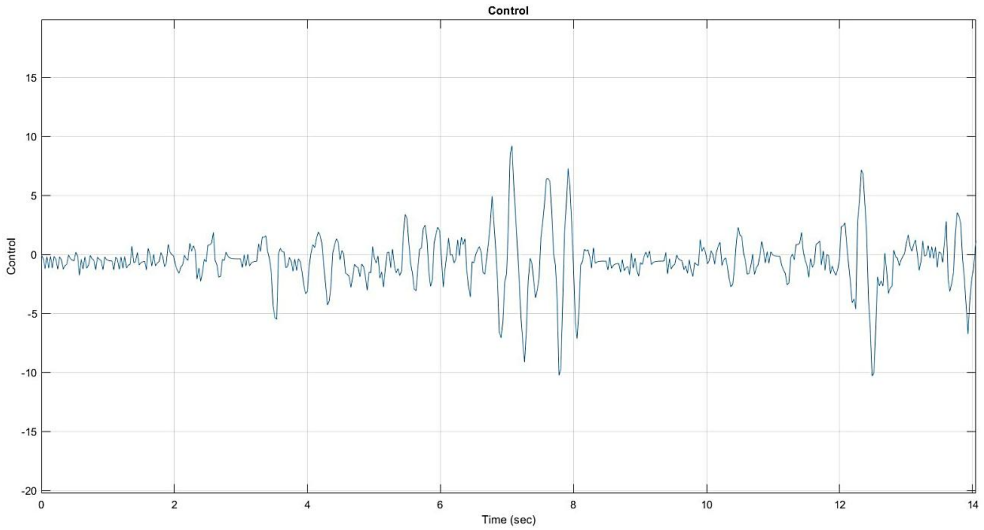

Fig. 8. Graph of the system's control value.

Graphic of the control value in this system, a comparison between this graph and the previous one reveals that an increase in the position error corresponds to a higher control output, as a function of the pendulum's angular position.

To evaluate the performance of the implemented control system, three main signals were analyzed: the actual angular position of the pendulum (Fig. 6), the position error (Fig. 7), and the control signal applied by the actuator (Fig. 8). These signals allow for the observation of both the system's stability in steady-state and its response to disturbances.

In the position error graph (Fig. 7), it can be seen that during the test, the system maintains a low-magnitude ($\pm$ 2.5) oscillation around the zero value. This indicates that the pendulum is relatively close to its equilibrium position, although with slight fluctuations. Furthermore, in the same figure, it can also be observed that despite the application of disturbances of varying magnitudes, the pendulum is capable of maintaining a constant state of equilibrium.

The real angular position graph (Fig. 6.), confirms these actions: a sudden variation in position is observed, with a peak above 91 degrees followed by a drop below 89 degrees. These oscillations indicate that the system momentarily loses balance and that the pendulum's dynamics are affected.

The control signal graph (Fig. 7.) helps to understand the system's reaction to these events. During the initial stage (before 6 seconds), the control signal remains with low-amplitude oscillations, reflecting that small corrections are being made to maintain balance. However, after the disturbance, a mayor response from the controller is evident, with peaks reaching values close to 10 and below -10. This response suggests that the controller is attempting to correct this error quickly, applying a large control action.

This behavior can be interpreted in two ways:

- On the one hand, the controller reacts effectively to an instability condition, which is expected in nonlinear systems such as the Furuta pendulum.
- On the other hand, the excessive magnitude of the control signal could indicate that the system is at the actuator's saturation limit, or that the PID controller gains are configured with values that produce an aggressive response.

Implementing the Kalman filter in underactuated control systems is useful as it allows for precise estimation of system states that cannot be directly measured or are affected by noise. This contributes to improving the controller's decision-making and, therefore, the overall system performance. However, in this specific case, the precision potentiometer demonstrates high effectiveness in sensing the angular position of the pendulum, which is highly beneficial for this type of system, as it relies on the accurate detection of small error measurements to be interpreted by the controller to actuate the motor. However, when the Kalman filter is applied, these small error measurements are smoothed out, preventing the controller from interpreting them as position errors, which significantly hinders the ability to stabilize the pendulum.

To demonstrate the impact of incorporating the Kalman filter with a higher gain into the system, several performance tests were conducted. The following values correspond to those that exhibited the lowest error in these tests:

Control variables: Xdes=542.5, Kp=170, Kd=1.09, Ki=0.0, h(iterations)=0.001.

Kalman filter variables: a (prediction factor) =1.0, k=1, sigma_u=0.1.

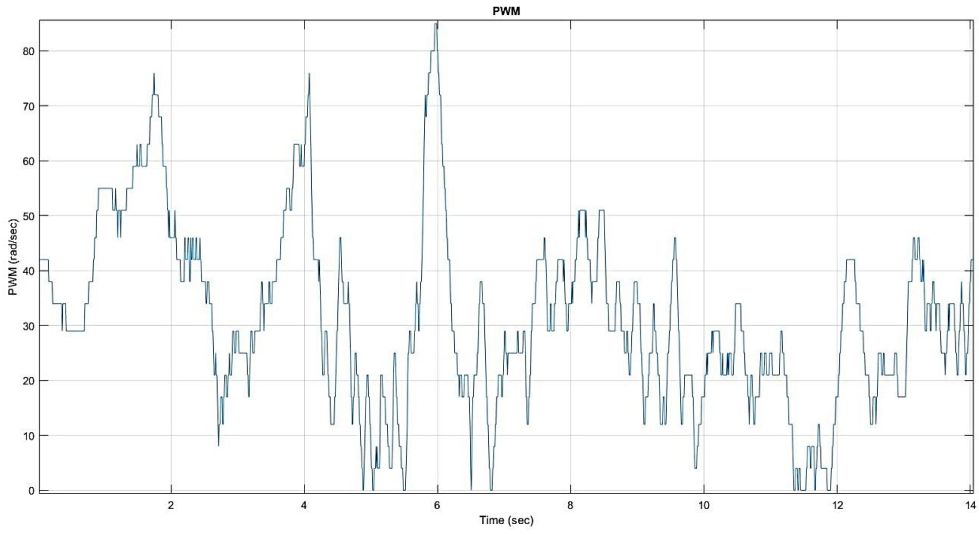

Fig. 9. Graph of the motor speed (PWM) with Kalman filter.

In fig. 9 a graphic of motor speed (PWM) versus time is presented; however, in this case, it can be observed that the motor speed does not even reach 50% of its maximum capacity, a behavior that contrasts significantly with the previous performance tests

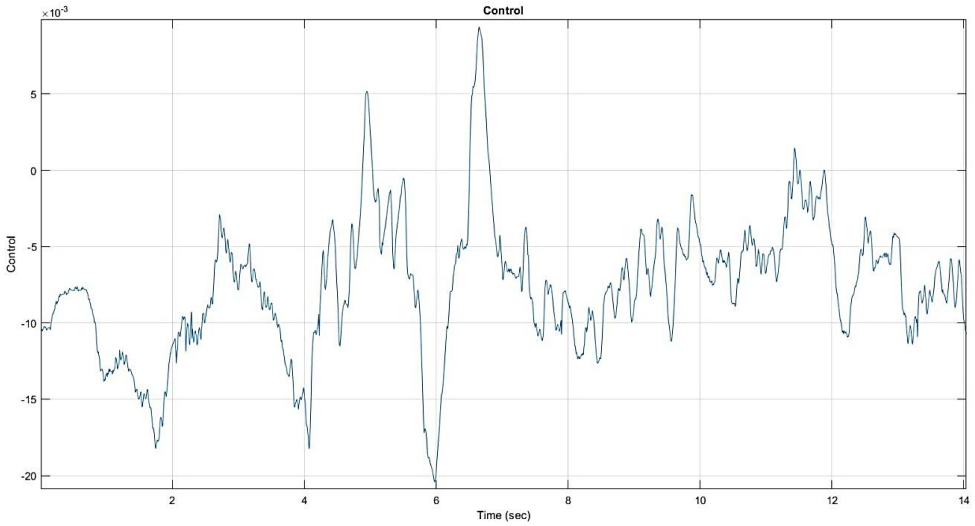

Fig. 10. Graph of the system's control value with the Kalman filter.

Now, by analyzing the control signal graph, another significant change can be observed. Due to the influence of the Kalman filter, the controller is unable to detect small error measurements; as a result, the control values recorded are considerably lower compared to those obtained in the previous performance tests.

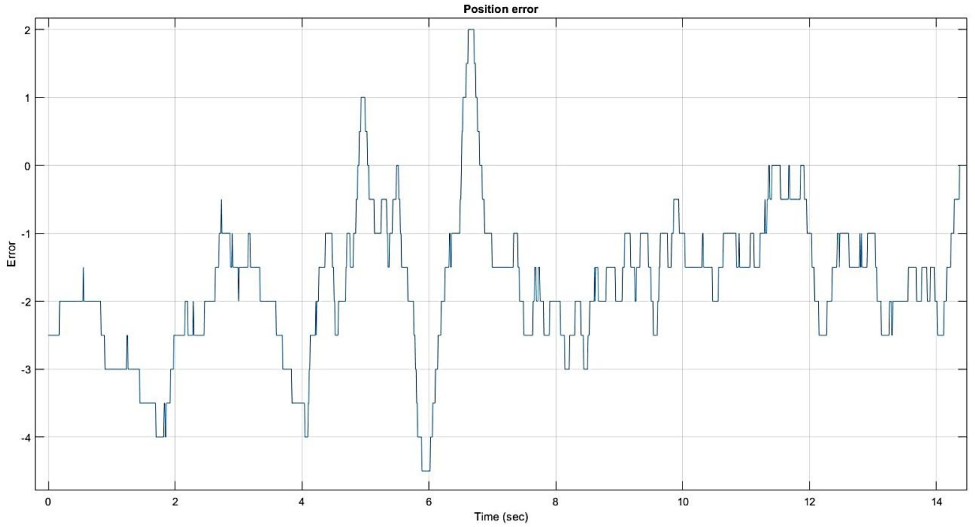

Fig. 11. Graph of the pendulum's position error with Kalman filter.

This graph reveals another important change in the prototype's behavior. Although the pendulum repeatedly passes through the equilibrium position, it remains oscillating outside of it for most of the duration of the test. Additionally, the squared peaks observed in the graph indicate time intervals during which the control system makes no adjustments to the pendulum's position.

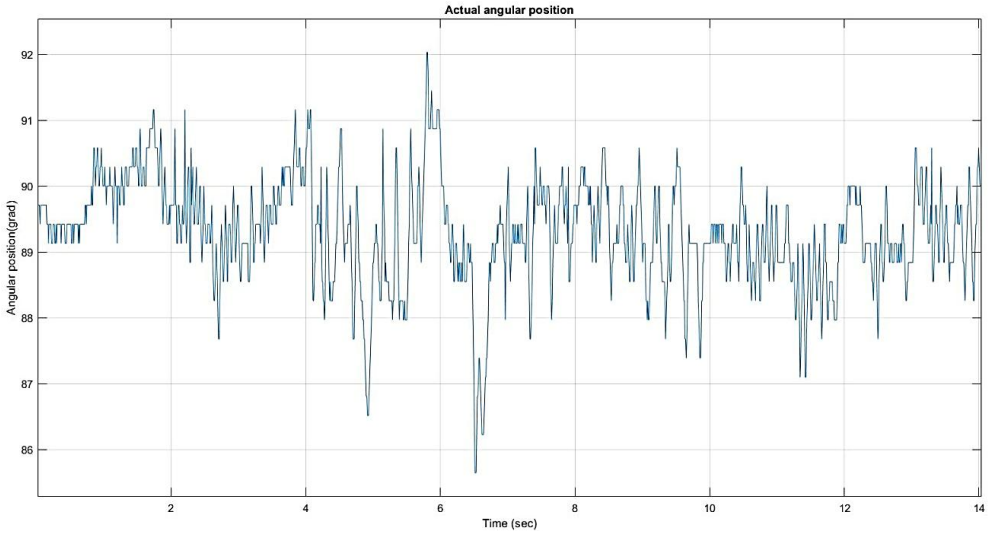

Fig. 12. Graph of the pendulum's angular position with Kalman filter.

Figure 12 shows the graph of the pendulum's angular position during the test, where the relationship with Figure 11 can be observed, along with a clear contrast compared to the tests conducted with a lower Kalman filter gain, which made stabilization significantly difficult.

Table 2. Comparison of the related works.

Related works		
Author and year	Characteristics	Development
Calderón Agudelo, J.M. (2021)	In this work, an Inverted Furuta Pendulum is developed, in which three different controls are implemented, whose mathematical development as well as their simulation in the MatLab platform are specified. In addition, a 3D prototype is created in the MatLab software and its Simulink tool to have a better visual perspective of the system. The mathematical development of the Furuta Pendulum is specified. This work focuses on being accessible.	The mathematical model that represents the dynamic behavior of the system is formulated, in addition to studying the mathematical development of the three controllers and their subsequent simulation as described below. The PID control is carried out by means of the transfer function of the Furuta Pendulum in its linear format and is simulated in the MatLab Software. The fuzzy control can be based solely on the experience gained from the system; Therefore, a Mamdani-type fuzzy control is used with the help of the MatLab Fuzzy Logic Toolbox. The ANN control is developed in the MatLab software using newff function, which creates a backpropagation-type artificial neural network. The virtual plant of the system is developed to better understand its behavior with the help of MatLab and Simulink. Finally, the physical Furuta Pendulum is developed with the help of SolidWorks software; the design mainly focuses on being replicable and accessible.

Alvarez Serrano, I. M. (2021)	In this document, two control techniques, PID and State Feedback, are implemented. Both controls are implemented of the embedded platforms of Arduino and Field Programable Gate Array (FPGA). Data regarding the behavior of the control son both embedded platforms are specified and shown in comparative tables.	A theorical review of the control techniques applied to the Furuta Pendulum (PID) and state Feedback is carried out. In the second phase, PID control and the state feedback control are designed and implemented. For the second controls, the process begins with mathematics model of the system, which is validated with the help of MatLab. Once the model is validated and parameterized, it is linearized, and the feedback vector is obtained using Ackermann's formula. Both controls are implemented on the embedded platforms of Arduino and FPGA. The results obtained from the experiments of both controls are acquired by a National Intruments data acquisition card. The interpretation of the data is carried out using the MatLab software in order to calculate the average error indices, average energy, and total energy, and a table is presented comparing the data with different operating characteristics.
Velásquez Marcelo, A. N. (2024)	The project focuses on improving the acquisition of the mathematical model of the system through the equations of motion obtained by the Euler-Lagrange method. The virtual model was based on a prototype from the University of Vic. With the necessary equations, an LQR control scheme can be implemented in Simulink.	The aim is to stabilize the Furuta Pendulum by means of LQR control. To fulfill the general objective, the following strategy Will be followed: the mathematical equations for the dynamics of the system must be found or formulated; after calculating the two types of energy (potential and kinetic), they are developed by means of the Lagrange equations to obtain the model f the pendulum´s motion. The equations are linearized to generate a change in the format of the variables that are incorporated into the control program. To stabilize the system, the LQR feedback constants must be found. Finally, the MatLab program is implemented with the constants obtained from the control, in order to stabilize the Dynamic system.

To provide a broader perspective, Table 2 aims to compare the development and methodologies adopted in related works by various authors, to facilitate the understanding of the proposals and the development of this study.

4. Conclusions

This work presents the design, construction, and testing of a low-cost didactic prototype of the Furuta-type pendulum to analyze control and stability issues within this system. Software such as SolidWorks was implemented for part design and Arduino for the incorporation of PID control and the Kalman filter.

During the construction of the prototype we were able to notice several physical and virtual impediments: it was observed that both the surface of the base and the model cannot contain unevenness since if the weight gives way towards some point the stability of the entire system is compromised causing the counterweight to fall, in the same way, the height of the model and counterweight can make the system more unstable due to the position of the gravity scepter, which if they are higher they are more unstable. Virtually, it was found that there is a loss of information about the angular position of the counterweight, necessary to activate the motor in the event of small vibrations, related to the implementation of the Kalman filter. This generates a failure in the calculated error and, consequently, in the system's response, resulting in a delay, causing an imbalance in the model. The prototype achieves stability in a period of 13 seconds, a time that could be increased if an angular position sensor for the motor was implemented, which facilitates responses to system errors, a more effective control for this type of systems such as LQR control.

The presented model, compared to commercial models, shows a 16-fold reduction in acquisition cost, thus fulfilling the objective of prototype development. This model can contribute to the academic development of students with limited resources interested in topics related to controllability and stability of highly nonlinear systems, opening up the possibility of improvements to this system and prototype.

5. References

Calderón Agudelo, J. M. (2022). *Desarrollo de un péndulo de Furuta como herramienta didáctica para la aplicación de diferentes métodos de control regulatorio y control inteligente en la Universidad de Pamplona* (Tesis de maestría).

Universidad de Pamplona. http://repositoriodspace.unipamplona.edu.co/jspui/handle/20.500.12744/4484

Alvarez Serrano, I. M. (2021). *Evaluación del desempeño de dos algoritmos de control en tiempo real usando Arduino y FPGA. Caso de estudio: péndulo de Furuta* (Tesis de pregrado).

Universidad de Cuenca. http://dspace.ucuenca.edu.ec/handle/123456789/36702

Aboy, J. R. F. *Estudios sobre el Péndulo Invertido Rotacional y el Péndulo Doble Invertido Rotacional*.

Martínez-Velázquez, F. J., Estrada-Manzo, V., & Bernal-Reza, M. (2021). Diseño de observadores no lineales para plantas mecatrónicas por medio de LMIs. *Pädi Boletín Científico de Ciencias Básicas e Ingenierías del ICBI, 8*(16), 75-81.

Prado, A., Herrera, M., & Menéndez, O. (2020). Levantamiento inteligente y estabilización robusta de un sistema de péndulo invertido rotatorio vía control predictivo basado en modelo no-lineal y tubos. *Revista Politécnica, 45*(1), 49-64.

Cárdenas Castañeda, B. A., & Castiblanco Castañeda, A. V. (2022). Implementación de plataforma Hardware in the Loop para la enseñanza del control sobre sistema embebido de bajo costo.

Yudho Montes de Oca, E. (2023). *Control del péndulo invertido rotativo por aprendizaje reforzado* (Tesis de maestría, Centro de Investigación y de Estudios Avanzados del IPN, Departamento de Control Automático).

Velasquez Marcelo, A. N. (2024). *Control de un péndulo invertido*.

Canales Martínez, J. (2020). *Diseño de un controlador híbrido predictivo adaptivo para la subida y estabilización de un péndulo invertido* (Tesis de pregrado). Universitat Politècnica de València. https://riunet.upv.es/handle/10251/161534

Coria, L. N., Ramírez-Villalobos, R., Aldrete-Maldonado, C., & Sanchez-Martínez, E. (2022). Control del péndulo de Furuta aplicando programación genética inspirado en un controlador por modos deslizantes. *Revista Aristas, 281*-287.

Acosta Villamil, D., Pacheco Bolívar, J., Noguera Polania, J., & Sanjuan Mejía, M. (2021). Modelo de red neuronal ARMAX para un péndulo de Furuta. *Ingeniare. Revista chilena de ingeniería, 29*(4), 668-682.

Rodriguez Rivera, L. M., & Ruiz Bravo, H. F. (2020). Desarrollo de plantas virtuales para el estudio de sistemas de control usando v-rep y Matlab.

Hernandez-Cortes, T., Amador-Macias, M., Tapia-Herrera, R., & Meda-Campana, J. (2024). Output regulation for descriptor systems with high-gain observer used as exosystem for unmodeled references. *IEEE Latin America Transactions, 22*(2), 156-165

Martínez, B., Sanchis, J., & García-Nieto, S. (2023). A model independent constrained predictive control for the Furuta Pendulum. In *XLIV Jornadas de Automática* (pp. 323-328). Universidade da Coruña. Servizo de Publicacións.

Yudho Montes de Oca, E. (2023). *Control del péndulo invertido rotativo por aprendizaje reforzado* (Tesis de maestría, Centro de Investigación y de Estudios Avanzados del IPN, Departamento de Control Automático).

Díaz, J. A., Estrada-Manzo, V., & Bernal, M. (2024). Diseño de par calculado robusto no lineal basado en observación: una solución por medio de desigualdades matriciales lineales. *Revista Iberoamericana de Automática e Informática Industrial, 21*(3), 218-230.

Moura, L. R. C., Montezuma, M. A. F., Mendonça, M., & Oliveira, E. R. (2024). Controlling the Furuta pendulum: Proof of concept through virtual prototyping. *Journal of Applied Research and Technology, 22*(1), 80–93. https://orcid.org/0000-0003-1364-1483

Guo, H., Wu, J., Yin, Z., & Zhang, W. (2023). Comparative study of Furuta pendulum based on LQR and PID control. *Journal of Physics: Conference Series, 2562*(1), 012075. https://doi.org/10.1088/1742-6596/2562/1/012075

Mendez, E., Baltazar-Reyes, G., Macias, I., & Gómez, A. (2020). ANN based MRAC-PID controller implementation for a Furuta pendulum system stabilization. *Advances in Science, Technology and Engineering Systems Journal (ASTESJ), 5*(3), 369–374. https://doi.org/10.25046/aj050346

Mérida Rubio, J. O., Chávez Vázquez, P. A., Coria de los Ríos, L. N., & Aguilar Landa, J. A. (2020). Optimal control design for Furuta's pendulum. *Revista de Ciencia y Tecnología (RECIT), 2*(2), 15–26. https://doi.org/10.37636/recit.v124953

Chen, K., Yi, J., & Song, D. (2020). Gaussian processes model-based control of underactuated balance robots. *arXiv*. https://arxiv.org/abs/2010.15320

Antonio-Cruz, M., Hernandez-Guzman, V. M., Merlo-Zapata, C. A., & Marquez-Sanchez, C. (2023). Nonlinear control with friction compensation to swing-up a Furuta pendulum. *Practice Article*. Available online: https://doi.org/10.1016/j.jfranklin.2023.05.021

Valera, A., Vallés, M., & Cardo, M. (2002). *Desarrollo y control de un péndulo de Furuta*. Departamento de Ingeniería de Sistemas y Automática. Recuperado el 23 de marzo de 2025, de https://scholar.google.com.mx/scholar?hl=es&as_sdt=0%2C5&q=DESARROLLO+Y+CONTROL+DE+UN+P%C3%89NDULO+DE+FURUTA&btnG=

Mori, S., Nishihara, H., & Furuta, K. (1976). Control of unstable mechanical system: Control of pendulum. *International Journal of Control*. Taylor & Francis.

Calderón Agudelo, J. M. (2022). *Desarrollo de un péndulo de Furuta como herramienta didáctica para la aplicación de diferentes métodos de control regulatorio y control inteligente en la Universidad*. Recuperado el 23 de marzo de 2025, de https://repositoriodspace.unipamplona.edu

Arias Blanco, V. (2018). *Control y fabricación de un péndulo invertido mediante un volante de inercia* (Trabajo de grado, Grado en Ingeniería Electromecánica, Itinerario Electrónico). Escuela Técnica Superior de Ingeniería (ICAI), Madrid, España.

Li, Q., Li, R., Ji, K., & Dai, W. (2015, November). Kalman filter and its application. In 2015 8th international conference on intelligent networks and intelligent systems (ICINIS) (pp. 74-77). IEEE. https://doi.org/10.1109/ICINIS.2015.35

Hernández Pérez, J., Gutiérrez Moreno, E., Ordaz Oliver, J. P., & Ordaz Oliver, M. O. (2024). LQR-Based Control with Gravity Compensation for a Wind Turbine Pendulum System. International Journal of Combinatorial Optimization Problems and Informatics, 15(5), 161–180. https://doi.org/10.61467/2007.1558.2024.v15i5.428

Ogata, K. (1995). Discrete-time control systems (5th ed.). Prentice-Hall, Inc.

Åström, K. J., & Hägglund, T. (2006). PID control. IEEE Control Systems Magazine, 1066, 30-31.

Ortiz, M., & Popov, E. P. (1985). Accuracy and stability of integration algorithms for elastoplastic constitutive relations. International journal for numerical methods in engineering, 21(9), 1561-1576. https://doi.org/10.1002/nme.1620210902

Alvarado Hernández, J. R., Domínguez Ortega, O., & Ordaz Oliver, M. O. (2024). Estabilización de un balancín eólico de bajo costo mediante un contro-lador LQR con compensación gravitatoria: Resultados experimentales. Memorias Del Concurso Lasallista De Investigación, Desarrollo E innovación, 11(1), 112–116. https://doi.org/10.26457/mclidi.v11i1.4210

Burden, R. L., & Faires, J. D. (2011). *Numerical analysis* (9th ed.). Brooks/Cole, Cengage Learning.

Chapra, S. C., & Canale, R. P. (2015). *Métodos numéricos para ingenieros* (7.ª ed.). McGraw-Hill Education.