

**UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
ESCUELA PROFESIONAL DE INGENIERÍA  
ELECTRÓNICA**



**TESIS**

**“Guía de prácticas de laboratorio con sistemas embebidos para la  
enseñanza de control digital en los estudiantes de la EPIE –  
UNPRG”**

**Para optar el título profesional de  
INGENIERO ELECTRÓNICO**

**ELABORADA POR:**

Bach. Amaya Huayama Segundo Raul

Bach. Olazabal Urbina Daniel Andres

**ASESOR:**

Mg. Ing. Romero Cortez Oscar Uchelly

ORCID: 0000-0002-7727-7900

**LAMBAYEQUE – PERÚ  
2023**

**“Guía de prácticas de laboratorio con sistemas embebidos para la enseñanza de  
control digital en los estudiantes de la EPIE – UNPRG”**

Para optar el título profesional de  
INGENIERO ELECTRÓNICO

**Aprobado por el siguiente jurado:**



---

Ing. Ramírez Castro Manuel Javier  
Presidente



---

Dra. Ing. Chamán Cabrera Lucía Isabel  
Secretario



---

Mg. Ing. Nombera Lossio Martín Augusto  
Vocal



**UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO**  
**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**  
**DECANATO**

Ciudad Universitaria - Lambayeque



**ACTA DE SUSTENTACIÓN VIRTUAL N° 021-2023-D/FACFyM**

Siendo las 11:20 am del día 26 de mayo del 2023, se reunieron vía plataforma virtual, <https://meet.google.com/cef-cdcd-ovi> los miembros del jurado evaluador de la Tesis titulada:

**GUÍA DE PRÁCTICAS DE LABORATORIO CON SISTEMAS EMBEBIDOS PARA LA ENSEÑANZA DE CONTROL DIGITAL EN LOS ESTUDIANTES DE LA EPIE – UNPRG**

Designados por Resolución N° 059-2022-VIRTUAL-D/FACFyM de fecha 10 de octubre de 2022. Con la finalidad de evaluar y calificar la sustentación de la tesis antes mencionada, conformada por los siguientes docentes:

<b>Ing. Manuel Javier Ramírez Castro</b>	Presidente
<b>Dra. Ing. Lucía Isabel Chamán Cabrera</b>	Secretario
<b>Mg. Ing. Martín Augusto Nombera Lossio</b>	Vocal

La tesis fue asesorada por el Mg. Ing. Oscar Uccelly Romero Cortez, nombrado por Resolución N° 059-2022-VIRTUAL -D/FACFyM de fecha 10 de octubre de 2022.

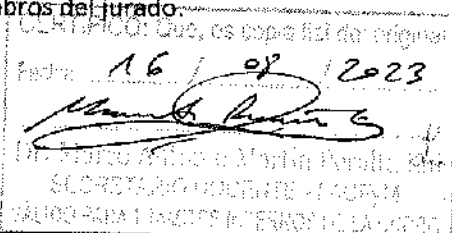
El Acto de Sustentación fue autorizado por Resolución N° 317-2023-VIRTUAL-D/FACFyM de fecha 02 de mayo de 2023.

La Tesis fue presentada y sustentada por los Bachilleres: Amaya Huayama Segundo Raul y Olazabal Urbina Daniel Andrés y tuvo una duración de 30 minutos.

Después de la sustentación, y absueltas las preguntas y observaciones de los miembros del jurado se procedió a la calificación respectiva, otorgándole el Calificativo de 16 (dieciséis) en la escala vigesimal, mención de Bueno.

Por lo que quedan aptos para obtener el Título Profesional de Ingeniero Electrónico, de acuerdo con la Ley Universitaria 30220 y la normatividad vigente de la Facultad de Ciencias Físicas y Matemáticas y la Universidad Nacional Pedro Ruiz Gallo.

Siendo las 12:20 pm se dio por concluido el presente acto académico, dándose conformidad al presente acto con la firma de los miembros del jurado.



**Ing. Manuel Javier Ramírez Castro**  
Presidente

**Dra. Ing. Lucía Isabel Chamán Cabrera**  
Secretaria

**Mg. Ing. Martín Augusto Nombera Lossio**  
Vocal

**Mg. Ing. Oscar Uccelly Romero Cortez**  
Asesor

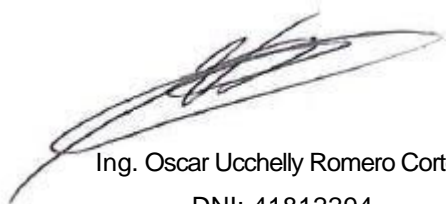
## CONSTANCIA DE VERIFICACIÓN DE ORIGINALIDAD

Yo, Oscar Uchelly Romero Cortez, usuario revisor del documento titulado **GUÍA DE PRÁCTICAS DE LABORATORIO CON SISTEMAS EMBEBIDOS PARA LA ENSEÑANZA DE CONTROL DIGITAL EN LOS ESTUDIANTES DE LA EPIE – UNPRG**. Cuyo autor es, Bach. Amaya Huayama, Segundo Raul, identificado con documento de identidad 74962729. Bach. Olazabal Urbina Daniel Andres, identificado con documento de identidad 73204475; declaro que la evaluación realizada por el programa informático ha arrojado un porcentaje de similitud de **18%**, verificable en el resumen de reporte automatizado de similitudes que se acompaña.

El suscrito analizó dicho reporte y concluyó que cada una de las coincidencias detectadas dentro del porcentaje de similitud permitido no constituyen plagio y que el documento cumple con la integridad científica y con las normas para el uso de citas y referencias establecidas en los protocolos respectivos.

Se cumple con adjuntar el Recibo Digital a efectos de la trazabilidad respectiva del proceso.

Lambayeque, 23 de febrero del 2023



Ing. Oscar Uchelly Romero Cortez

DNI: 41812294

Asesor

(Precisar si es docente, asesor, docente investigador, administrativo u otro)

Se adjunta:

\*Resumen de Reporte automatizado de similitudes

\*Recibo Digital

## DEDICATORIA

*El presente trabajo va dedicado a mis padres que con su amor, dedicación , enseñanzas y esfuerzo lograron en mi inculcar la pasión por el estudio y poder lograr mis metas propuestas, de igual manera dedicado a mi hermano que siempre me apoya e incentiva a seguir adelante, dedicado a mis familiares y amigos que siempre me apoyaron y me dieron ánimos para seguir adelante y una dedicatoria especial a mi abuelo Luis Urbina que siempre me apoyo en este proceso de mi vida profesional y que sé que desde el cielo está muy orgulloso de mí.*

*Olazabal Urbina Daniel Andres*

*Dedico este trabajo a mis padres y familiares, porque fueron ellos quienes, con su apoyo incondicional y cariño sincero, inculcaron en mi la pasión por el estudio, y a todas las personas que con especial cariño dedicaron un momento de sus vidas para brindarme su ayuda.*

*Amaya Huayama Segundo Raul*

## AGRADECIMIENTOS

*Agradecemos a Dios por permitirnos tener y disfrutar a nuestras familias, porque siempre nos han apoyado en cada decisión y proyecto, gracias por confiar y creer en nosotros, gracias nuestros padres por habernos forjado como las personas que somos en la actualidad, nuestros logros se los debemos a ellos, siempre nos motivaron constantemente a seguir nuestros sueños, no ha sido sencillo el camino hasta ahora, pero gracias a sus aportes, amor a su inmensa bondad y apoyo.*

*Agradecemos a nuestros amigos y maestros, personas de gran estima y sabiduría, agradecemos por su tiempo y esfuerzo que dedicaron a compartir sus conocimientos.*

*Agradecemos a Dios por este nuevo triunfo, gracias a todas las personas que nos apoyaron y creyeron en nosotros.*

# ÍNDICE DE CONTENIDO

DEDICATORIA .....	v
AGRADECIMIENTOS .....	vi
ÍNDICE DE CONTENIDO .....	vii
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE TABLAS .....	xiii
RESUMEN.....	xiv
ABSTRACT.....	xv
INFORMACIÓN GENERAL.....	xiii
INTRODUCCIÓN .....	xiv
RESUMEN.....	xviii
ABSTRACT.....	xix
I. DISEÑO TEÓRICO.....	1
1.1. Antecedentes .....	1
1.2. Bases teóricas .....	2
1.2.1. Control digital.....	2
1.2.2. Sistemas embebidos .....	11
1.3. Operacionalización de variables.....	16
II. DISEÑO METODOLÓGICO.....	17
2.1. Diseño de contrastación de hipótesis .....	17
2.2. Población y muestra .....	17

2.3.	Técnicas, instrumentos, materiales y equipos .....	17
2.3.1.	Técnicas e instrumentos de recolección de datos .....	17
III.	RESULTADOS.....	22
3.1.	Identificación del modelo matemático de la planta.....	22
3.2.	Prueba de los algoritmos de control PID digital .....	25
3.2.1.	Control PID digital con el primer método de Ziegler y Nichols .....	25
3.2.2.	Control PID digital con el método de Cohen y Coon.....	29
3.2.3.	Control PID digital con el método de la Integral del Error .....	33
3.3.	Aplicación y resultados del pretest.....	36
3.4.	Aplicación de la guía de prácticas de laboratorio .....	39
3.5.	Aplicación y resultados del postest .....	40
IV.	DISCUSIÓN DE RESULTADOS.....	41
V.	CONCLUSIONES.....	42
VI.	RECOMENDACIONES.....	43
REFERENCIAS		
ANEXOS		



## ÍNDICE DE FIGURAS

<b>Figura 1</b> Diagrama básico de un control digital con realimentación .....	4
<b>Figura 2</b> Diagrama básico de un controlador PID continuo .....	5
<b>Figura 3</b> Gráfica del método de integración trapezoidal.....	7
<b>Figura 4</b> Diagrama de Bloques de un Sistema Embebido .....	12
<b>Figura 5</b> Sistema embebido.....	15
<b>Figura 6</b> Sistema embebido seleccionado.....	19
<b>Figura 7</b> Conexiones de TCLAB .....	20
<b>Figura 8</b> Valores de temperatura y PMW al iniciar la prueba experimental .....	22
<b>Figura 9</b> Valores de temperatura y PMW al terminar la prueba experimental .....	23
<b>Figura 10</b> Gráficas de temperatura y PMW al terminar la prueba experimental .....	23
<b>Figura 11</b> Tiempo de respuesta del sistema luego de aplicar la señal escalón .....	24
<b>Figura 12</b> Gráfica de valores iniciales de temperatura y PWM – Ziegler y Nichols .....	25
<b>Figura 13</b> Lectura de valores iniciales de temperatura y PWM – Ziegler y Nichols.....	25
<b>Figura 14</b> Gráfica de respuesta del sistema ante el cambio de referencia a 30°C – Ziegler y Nichols .....	26
<b>Figura 15</b> Gráfica de respuesta del sistema igual a la referencia de 30°C – Ziegler y Nichols .....	26
<b>Figura 16</b> Lectura de valores de respuesta del sistema igual a la referencia de 30°C – Ziegler y Nichols .....	27
<b>Figura 17</b> Gráfica de respuesta del sistema ante el cambio de referencia de 40°C – Ziegler y Nichols .....	27
<b>Figura 18</b> Gráfica de respuesta del sistema igual a la referencia de 40°C – Ziegler y Nichols .....	28

<b>Figura 19</b> Lectura de valores de respuesta del sistema igual a la referencia de 40°C – Ziegler y Nichols .....	28
<b>Figura 20</b> Gráfica de valores iniciales de temperatura y PWM – Cohen y Coon.....	29
<b>Figura 21</b> Lectura de valores iniciales de temperatura y PWM – Cohen y Coon.....	29
<b>Figura 22</b> Gráfica de respuesta del sistema ante el cambio de referencia a 30°C – Cohen y Coon.....	30
<b>Figura 23</b> Gráfica de respuesta del sistema igual a la referencia de 30°C – Cohen y Coon...	30
<b>Figura 24</b> Lectura de valores de respuesta del sistema igual a la referencia de 30°C – Cohen y Coon.....	31
<b>Figura 25</b> Gráfica de respuesta del sistema ante el cambio de referencia de 40°C – Cohen y Coon.....	31
<b>Figura 26</b> Gráfica de respuesta del sistema igual a la referencia de 40°C – Cohen y Coon.....	32
<b>Figura 27</b> Lectura de valores de respuesta del sistema igual a la referencia de 40°C – Cohen y Coon.....	32
<b>Figura 28</b> Gráfica de valores iniciales de temperatura y PWM – Integral del Error .....	33
<b>Figura 29</b> Lectura de valores iniciales de temperatura y PWM – Integral del Error .....	33
<b>Figura 30</b> Gráfica de respuesta del sistema ante el cambio de referencia a 30°C – Integral del Error .....	34
<b>Figura 31</b> Gráfica de respuesta del sistema igual a la referencia de 30°C – Integral del Error .....	34
<b>Figura 32</b> Lectura de valores de respuesta del sistema igual a la referencia de 30°C – Integral del Error .....	35
<b>Figura 33</b> Gráfica de respuesta del sistema ante el cambio de referencia de 40°C – Integral del Error .....	35

<b>Figura 34</b> Gráfica de respuesta del sistema igual a la referencia de 40°C – Integral del Error .....	36
<b>Figura 35</b> Lectura de valores de respuesta del sistema igual a la referencia de 40°C – Integral del Error .....	36
<b>Figura 36</b> Sistema embebido TCLab utilizado en los talleres .....	40

## ÍNDICE DE TABLAS

<b>Tabla 1</b> Características de las acciones de control .....	5
<b>Tabla 2</b> Definición y operacionalización de variables .....	16
<b>Tabla 3</b> Análisis técnico y económico para seleccionar el sistema embebido.....	18
<b>Tabla 4</b> Principales referencias bibliográficas .....	21
<b>Tabla 5</b> Resultados del pretest .....	39
<b>Tabla 6</b> Resultados del postest .....	40

# INFORMACIÓN GENERAL

**Título:**

GUÍA DE PRÁCTICAS DE LABORATORIO CON SISTEMAS EMBEBIDOS  
PARA LA ENSEÑANZA DE CONTROL DIGITAL EN LOS ESTUDIANTES DE LA EPIE  
– UNPRG.

**Autores:**

- Bach. Amaya Huayama Segundo Raul.
- Bach. Olazabal Urbina Daniel Andres.

**Asesor de especialidad:**

Mg.. Ing. Romero Cortez Oscar Ucchelly.

**Área de investigación:**

Ingeniería electrónica

**Línea de investigación:**

Ingenierías y Tecnologías

**Lugar:**

La Victoria, Chiclayo, Lambayeque.

**Duración estimada del proyecto**

- Fecha de inicio: julio de 2022.
- Fecha de finalización: febrero de 2023.

## INTRODUCCIÓN

Moura Oliveira et al. (2020) indican que muchos estudiantes de ingeniería en Portugal, EE. UU. y el Reino Unido solo tienen un curso básico de control por retroalimentación en su plan de estudios, y que estos cursos suelen basarse completamente en los métodos tradicionales de control en el dominio del tiempo continuo. Sin embargo, señalan que la mayoría de los fundamentos de diseño de control se están incorporando ahora a los sistemas digitales. Para abordar esta situación, los autores desarrollaron controladores digitales educativos basados en TCLab de APMonitor. Como resultado, se creó una herramienta eficiente, de tamaño reducido y portátil para que los estudiantes puedan aprender algoritmos de control digital tanto básicos como avanzados. De esta forma, se logra una solución efectiva que permite a los estudiantes adquirir conocimientos actualizados en una materia clave para la ingeniería, lo que a su vez contribuye a mejorar su formación académica y su preparación para el mercado laboral.

Nguyen et al. (2022) llevaron a cabo un estudio en Vietnam en el que destacan la importancia de contar con instalaciones de laboratorio adecuadas para los estudiantes de ingeniería de control. No obstante, señalan que los principales instrumentos y equipos necesarios para llevar a cabo una práctica de laboratorio eficaz son costosos, lo que hace difícil su adquisición tanto para institutos y universidades como para los propios estudiantes. Ante esta situación, los autores decidieron crear un kit de laboratorio de control desde cero, utilizando componentes electrónicos, con el fin de enseñar a los estudiantes una serie de temas de ingeniería de control, como el modelado de sistemas, la identificación de sistemas, el control clásico y avanzado, así como varios algoritmos de control, como ON/OFF, PID, Fuzzy, entre otros. De esta forma, se logró desarrollar una solución más accesible y económica para los estudiantes y las instituciones educativas.

Sotelo et al. (2022), en México, afirman que la experiencia práctica en el laboratorio es fundamental para enseñar la ingeniería de control, ya que los estudiantes pueden cerrar las brechas entre la comprensión teórica y la experimentación. Los autores resaltan que, en los laboratorios virtuales, los estudiantes no están expuestos a las dificultades del mundo real, como el mal funcionamiento de los equipos o las interferencias, y que pueden surgir problemas de conectividad. Por esta razón, desarrollaron un proyecto educativo llamado Lab-Tec@Home, con el que se logró aumentar el acceso de los estudiantes a la enseñanza práctica de la ingeniería de control en temas relacionados con la identificación de modelos de sistemas y el desarrollo de algoritmos de control digital. De esta manera, se creó una solución efectiva que permite a los estudiantes adquirir habilidades prácticas en la materia, incluso en situaciones difíciles como las que se presentaron durante la pandemia del COVID-19.

Fonseca (2022), en Brasil, sostiene que la ingeniería de control es una disciplina que proporciona a los estudiantes los conocimientos necesarios para automatizar las operaciones industriales. Además, señala que en el contexto de la Industria 4.0, los nuevos controladores digitales incorporan tecnologías avanzadas y complejas, como la Inteligencia Artificial (AI), el Internet de las Cosas (IoT) y el Big Data, lo que representa una evolución significativa respecto a los antiguos controladores. Por lo tanto, recomienda que tanto los instructores como los estudiantes cuenten con las herramientas necesarias para adentrarse en el ámbito del control digital y sus complejos algoritmos. De esta forma, podrán comprender y aplicar adecuadamente los nuevos conceptos y tecnologías que están transformando la industria, lo que les permitirá estar preparados para afrontar los retos del mercado laboral actual y futuro.

Barbarán Castañeda y Chávarry Ysla (2019), en nuestro país, señalan que los instructores deben adaptarse a los cambios tecnológicos, capacitarse en el uso de nuevos recursos y equipos digitales, y transformar la metodología de enseñanza en las asignaturas relacionadas con la innovación y las nuevas tecnologías. En este sentido, han creado una guía

de prácticas de diseño de algoritmos de control digital para robótica utilizando sistemas embebidos. El objetivo es enseñar la programación necesaria para generar algoritmos de control digital, lo que permitirá a los estudiantes adquirir habilidades y competencias en el ámbito del control digital y la robótica. De esta forma, podrán estar preparados para afrontar los desafíos del mercado laboral actual y futuro, en el que la automatización y la robótica son cada vez más importantes.

En la Escuela Profesional de Ingeniería Electrónica (EPIE) de la Universidad Nacional Pedro Ruiz Gallo (UNPRG), ubicada en Lambayeque, los estudiantes carecen de guías de prácticas de laboratorio para la enseñanza de control digital, ya que no disponen de recursos tecnológicos portátiles y económicos para llevar a cabo dichas prácticas. A esta problemática se suma el cierre de los laboratorios debido a la pandemia del COVID-19. Aunque se imparten clases teóricas, se realizan ejercicios y pruebas, y se utilizan simuladores para cubrir temas relacionados con las técnicas de control convencionales en tiempo continuo, existe una carencia significativa de experimentación por parte de los estudiantes y de enseñanza de técnicas de control digital en la ingeniería de control. Esta situación puede tener consecuencias negativas en el desempeño profesional de los estudiantes, ya que no contarían con los conocimientos necesarios para aplicar estrategias de control digital apropiadas en el contexto de la Industria 4.0, la digitalización y las tecnologías emergentes.

Después de analizar la problemática, se formuló la siguiente pregunta de investigación: ¿Cómo se puede mejorar la enseñanza de control digital para los estudiantes de la EPIE - UNPRG? La hipótesis planteada establece que, si se desarrolla una guía de prácticas de laboratorio basada en un sistema embebido portátil y de bajo costo, se mejorará la enseñanza de control digital para los estudiantes de la EPIE - UNPRG.



Por lo tanto, como objetivo general se planteó desarrollar una guía de prácticas de laboratorio con un sistema embebido para mejorar la enseñanza de control digital para los estudiantes de la EPIE – UNPRG. Los objetivos específicos fueron: i) Aplicar el método de la curva de reacción para identificar el modelo matemático de la planta de la guía de prácticas de laboratorio; ii) diseñar los algoritmos de control PID digital para implementarlos en la planta de la guía de prácticas de laboratorio; iii) aplicar un pretest a los estudiantes de la EPIE – UNPRG para evaluar su nivel de conocimiento respecto a la identificación de sistemas y el diseño de algoritmos de control PID digital; iv) aplicar la guía de prácticas de laboratorio a los estudiantes de la EPIE – UNPRG para desarrollar los ejercicios propuestos; y v) aplicar un posttest para determinar si la guía de prácticas de laboratorio mejora la enseñanza de control digital para los estudiantes de la EPIE – UNPRG.

## RESUMEN

En la Escuela Profesional de Ingeniería Electrónica (EPIE) de la Universidad Nacional Pedro Ruiz Gallo (UNPRG), la falta de recursos tecnológicos portátiles y económicos ha impedido que los estudiantes tengan acceso a una guía de prácticas de laboratorio de control digital. Para abordar esta problemática, se llevó a cabo una investigación tecnológica aplicada con un diseño transversal, prospectivo y experimental, cuyo objetivo era desarrollar una guía de prácticas de laboratorio para mejorar la enseñanza de control digital para los estudiantes de la EPIE - UNPRG. La metodología empleada en la investigación consistió en la aplicación de un pretest, la implementación de la guía de prácticas de laboratorio a una muestra de 25 estudiantes, y un posttest. El pretest reveló que el nivel de conocimiento de los estudiantes en cuanto a la identificación de sistemas y el diseño de algoritmos de control PID digital era regular (promedio de notas 11.12). Posteriormente, se desarrollaron ejercicios prácticos utilizando la guía de prácticas de laboratorio para controlar un sistema termodinámico. Al aplicar el posttest, se determinó que la guía de prácticas de laboratorio mejoró la enseñanza de control digital para los estudiantes de la EPIE - UNPRG (promedio de notas 18). En conclusión, se demostró que el uso de una guía de prácticas de laboratorio basada en un sistema embebido portátil y de bajo costo mejora significativamente la enseñanza de control digital para los estudiantes de la EPIE - UNPRG.

**Palabras clave:** Guía de prácticas de laboratorio, sistema embebido, control PID digital, sistema de primer orden más retardo, TCLab.

## ABSTRACT

At the Professional School of Electronic Engineering (EPIE) of the National University Pedro Ruiz Gallo (UNPRG), the lack of portable and economic technological resources has prevented students from having access to a digital control laboratory practice guide. To address this problem, an applied technological research was carried out with a cross-sectional, prospective and experimental design, whose objective was to develop a laboratory practice guide to improve the teaching of digital control for students of the EPIE - UNPRG. The methodology used in the research consisted of the application of a pretest, the implementation of the laboratory practice guide to a sample of 25 students, and a posttest. The pretest revealed that the level of knowledge of the students regarding the identification of systems and the design of digital PID control algorithms was regular (average grade 11.12). Subsequently, practical exercises were developed using the laboratory practice guide to control a thermodynamic system. When applying the post-test, it was determined that the laboratory practice guide improved the teaching of digital control for the students of EPIE - UNPRG (average marks 18). In conclusion, it was demonstrated that the use of a laboratory practice guide based on a portable and low-cost embedded system significantly improves the teaching of digital control for students of the EPIE - UNPRG.

**Keywords:** Laboratory practice guide, embedded system, digital PID control, first order plus delay system, TCLab.

# **I. DISEÑO TEÓRICO**

## **1.1. Antecedentes**

En un artículo científico presentado por Hoyo et al. (2021) en la Universidad de Almería, España, se describe su experiencia con la plataforma portátil TCLab para enseñar técnicas de identificación de sistemas y estrategias de control en cursos de grado y postgrado de ingeniería de control. Los resultados del estudio demuestran que emplear laboratorios portátiles y de bajo costo, como TCLab, ayuda a mejorar la enseñanza de los estudiantes porque les permite aplicar estrategias de control en sistemas reales, así como familiarizarse con el ruido en las señales del sistema y las perturbaciones que pueden alterar su dinámica.

El equipo de investigadores compuesto por Moura Oliveira et al. (2020) de Portugal, Estados Unidos y Reino Unido, publicaron un artículo científico en el que presentaron una propuesta para desarrollar y evaluar controladores digitales. Además, llevaron a cabo una serie de experimentos de formación utilizando el laboratorio de control de temperatura TCLab, un kit portátil y de bajo coste basado en la plataforma Arduino. Según los resultados obtenidos, TCLab es una herramienta altamente efectiva para experimentar y mejorar el aprendizaje de los controladores digitales, ya que permite a profesores y estudiantes trabajar en un laboratorio compacto y portátil.

En Portugal, Marques Aguiar (2020) presentó un proyecto de tesis de maestría que abarcó numerosos casos de kits de laboratorio y los comparó en términos de accesibilidad, relevancia didáctica y enfoque. Los resultados indicaron que con el kit portátil y de bajo costo TCLab, los estudiantes pudieron identificar un modelo dinámico de primer orden con retardo para un sistema termodinámico. Además, los estudiantes implementaron técnicas de algoritmos de control como el control ON-OFF, control P, control PD, control PI, control PID, y se comparó el rendimiento obtenido en cada caso.

En Trujillo, Grados Rebaza (2022) llevó a cabo un proyecto de tesis en el que desarrolló e implementó un controlador PID digital utilizando un microcontrolador PIC para un módulo educativo de control de velocidad de motores en el curso de automatización industrial. Los resultados del proyecto demostraron que la inclusión de un controlador digital PID en un módulo de instrucción mejoró significativamente el aprendizaje de los estudiantes, ya que les permitió experimentar y ajustar los parámetros críticos de un sistema de control en el mundo real.

En Lima, Saravia Valverde (2019) desarrolló un proyecto de tesis en el que se diseñó un módulo educativo y una guía de prácticas de laboratorio para el campo de la automatización, con el fin de que tanto estudiantes como profesores pudieran crear y configurar un sistema de control utilizando las plataformas Arduino y LabVIEW. Los resultados de la investigación muestran que la guía de prácticas de laboratorio es efectiva en la comprensión de los principios fundamentales de los sistemas de control digital a través de la experimentación y manipulación de variables físicas en un sistema de control retroalimentado.

En Lima, Ruíz Pérez (2019) presentó un proyecto de tesis en el que desarrolló un módulo instructivo para que los estudiantes de ingeniería mecatrónica realicen prácticas de laboratorio y experimenten con un sistema de control de caudal. Los resultados de la investigación mostraron que el uso de la guía de prácticas de laboratorio en el módulo educativo de control de flujo tuvo una influencia positiva y significativa en el desempeño de los estudiantes de ingeniería de control.

## **1.2. Bases teóricas**

### **1.2.1. Control digital**

A medida que la tecnología avanza, las empresas están digitalizando y automatizando sus plantas de control. Los avances de la ciencia y la tecnología hacen más factible que los profesionales del sector industrial diseñen e innoven sistemas de control. Sin embargo, los

desarrollos y sistemas clave son cada vez más sofisticados y complejos, lo que exige formación y dominio de las nuevas tecnologías de los sistemas de control digital (Flores Avella et al., 2020).

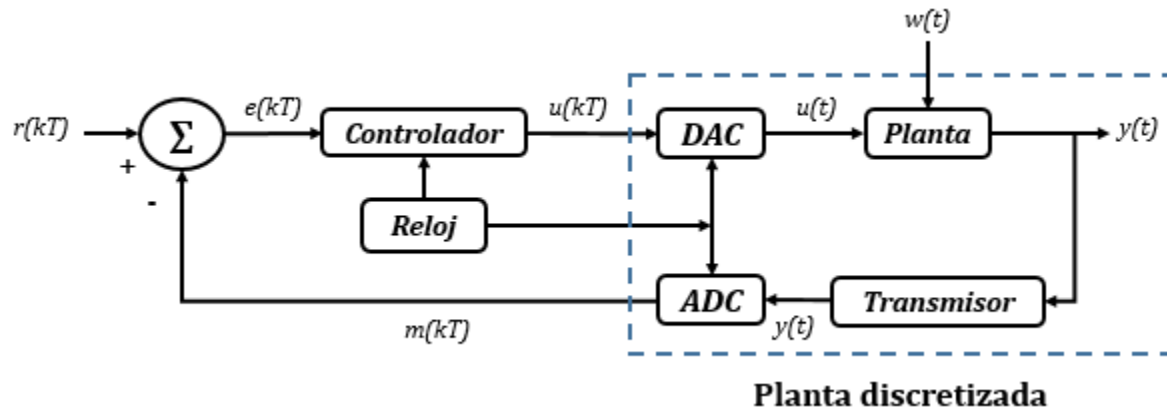
Cuando hablamos de control digital, hablamos de control en tiempo discreto, lo que significa que tendremos que discretizar las señales continuas que serán procesadas por el microprocesador. Esto implica que una o más variables pueden fluctuar sólo en puntos específicos del tiempo, conocidos como tiempos de muestreo, que se denotan por  $kT$  ( $k=0,1,2,\dots$ ) y pueden describir el momento en que se toma una medida física o el momento en que se accede a la memoria del ordenador (Flores Avella et al., 2020; Mosquera Quimbíta, 2020).

El diagrama fundamental del control digital con retroalimentación se muestra en la Figura 1. La señal de referencia o consigna especificada digitalmente por un ordenador es  $r(kT)$ . Un convertidor analógico-digital (ADC) convierte la señal de control  $u(t)$  en forma analógica y la mantiene constante durante el tiempo de muestreo. Un convertidor digital-analógico (DAC) convierte la señal de salida del sistema  $y(t)$  en forma digital en un periodo de muestreo  $kT$  determinado por los ciclos de reloj (Mosquera Quimbíta, 2020; Caicedo Grajales, 2018).

Como se trata de un sistema de lazo cerrado, las señales de los sensores también se discretizan para proporcionar una versión muestreada de la señal de salida de la planta  $m(kT)$ . La diferencia entre la consigna  $r(kT)$  y la respuesta discretizada de la planta  $m(kT)$  es interpretada como una secuencia de números enteros por el microprocesador, que la hace pasar por un algoritmo de control para crear la señal de control  $u(kT)$ . Se puede introducir una señal perturbadora  $w(t)$  en la planta, que intentará ajustarse mediante el algoritmo de control digital (Piñol Navarro, 2021; Mosquera Quimbíta, 2020).

**Figura 1**

*Diagrama básico de un control digital con realimentación*



*Fuente: Adaptado de Mosquera Quimbita (2020).*

Por su robustez y facilidad de comprensión, el controlador PID (Proporcional - Integral - Derivativo) es el más extendido en el sector industrial, y se emplea en el 95% de los procesos productivos. Asimismo, las técnicas de control PID o PID modificado son empleadas por más del 50% de los controladores industriales utilizados actualmente en los procesos térmicos. A pesar de los avances técnicos en materia de control, el controlador PID ha demostrado un rendimiento extremadamente bueno con la sintonización correcta de sus ganancias  $K_p$ ,  $T_i$  y  $T_d$  (Castaño, 2022; Mosquera Quimbita, 2020; Horna Martínez y Díaz Cercado, 2019).

Para comprender el control digital PID, primero debemos entender su forma continua. La señal de un controlador PID en tiempo continuo se describe de forma óptima mediante la siguiente ecuación (Mosquera Quimbita, 2020).

$$u(t) = K_p \left( e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \dots (1)$$

Dónde:  $u(t)$  es la señal de salida,  $e(t)$  es el error,  $K_p$  es la ganancia proporcional,  $T_i$  es el tiempo integral y  $T_d$  es el tiempo derivativo.

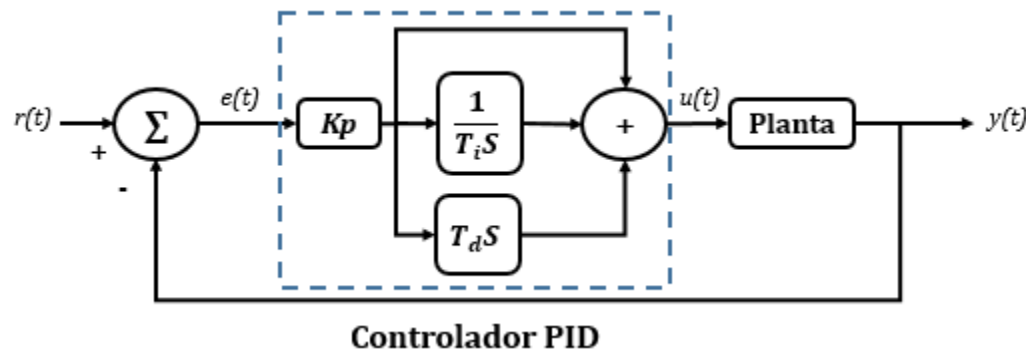
Al aplicar la transformada de Laplace a la ecuación (1) da como resultado la función de transferencia de un controlador PID continuo, que se construye como se indica en la ecuación

(2) y une la transformada de Laplace de la acción de control con la transformada de Laplace de la señal de error (Castaño, 2022).

$$G_{PID}(S) = \frac{U(S)}{E(S)} = K_p \left( 1 + \frac{1}{T_i S} + T_d S \right) \dots (2)$$

**Figura 2**

*Diagrama básico de un controlador PID continuo*



*Fuente: Adaptado de Castaño (2022).*

Es necesario tener una comprensión matemática del algoritmo de control PID para entender qué papel desempeña cada una de las acciones de control (proporcional, integral y derivativa). En la Tabla 1 se describen sus principales características (Castaño, 2022).

**Tabla 1**

*Características de las acciones de control*

Acción de control	Características
Proporcional	<ul style="list-style-type: none"> <li>Es proporcional al valor de la señal de error en cada instante de tiempo (actúa en el presente).</li> <li>Cuando la ganancia es baja, la acción de control es suave, con una respuesta lenta y sin oscilaciones.</li> <li>Cuando la ganancia es demasiado grande, la acción de control es contundente y la reacción es rápida pero inestable.</li> </ul>
Integral	<ul style="list-style-type: none"> <li>Acumula los errores con el tiempo y dejan de acumularlos hasta que se erradican (actúa en el pasado).</li> <li>Ayuda a reducir el error de estado estacionario creado por el control proporcional.</li> <li>Una ganancia grande mejora las características de funcionamiento en régimen permanente.</li> <li>En el régimen transitorio, donde la respuesta debe oscilar lo menos posible, un valor grande de ganancia no es adecuado, por lo que su valor debe ser bajo.</li> </ul>



*Continuación de la Tabla 1*

Derivativa	<ul style="list-style-type: none"> <li>• Emplea la derivada para prever la evolución de la señal de error, lo que le permite actuar sobre la planta y solucionar los fallos antes de que se produzcan (actúa en el futuro).</li> <li>• Actúa como un freno o amortiguador en el periodo transitorio, eliminando el sobreimpulso y las oscilaciones.</li> </ul>
------------	--

*Fuente: Adaptado de Mosquera Quimbita (2020).*

A continuación, se explica cómo al discretizar un controlador PID continuo se obtiene un controlador discreto (P, PI, PID).

Empezaremos con la aproximación discreta de un controlador proporcional. Como se ha mencionado, la señal de salida de esta acción de control es proporcional al error en un cierto instante de tiempo. La ecuación (3) representa su comportamiento en tiempo continuo (Castaño, 2022; Horna Martínez y Díaz Cercado, 2019).

$$u(t) = K_p e(t) \dots (3)$$

La ecuación (4) representa su comportamiento en tiempo discreto.

$$u(k) = K_p e(k) + u_0 \dots (4)$$

La señal de salida del controlador proporcional discreto en el instante  $(k-1)$  se representa en la ecuación (5).

$$u(k-1) = K_p e(k-1) + u_0 \dots (5)$$

Al realizar la diferencia entre la ecuación (4) y la ecuación (5) se obtiene la ecuación (6).

$$u(k) - u(k-1) = K_p [e(k) - e(k-1)] \dots (6)$$

Si a la ecuación (6) se le aplica la transformada Z se obtiene la ecuación (7).

$$(1 - z^{-1})U(z) = K_p (1 - z^{-1})E(z) \dots (7)$$

De la ecuación (7) se deduce que la función de transferencia que representa la dinámica del controlador proporcional discreto es la que se indica en la ecuación (8).

$$C(z) = \frac{U(z)}{E(z)} = q_0 \dots (8)$$

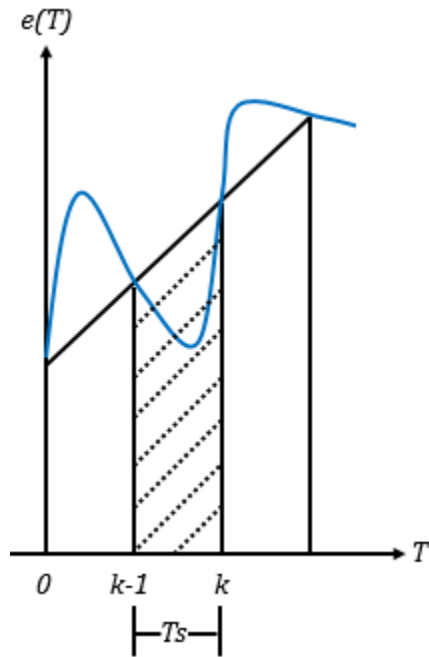
Continuamos con la aproximación del controlador proporcional integral discreto. Con el efecto de esta acción de control, la señal de salida se somete a un impulso inicial proporcional al error actual y luego varía progresivamente a un ritmo proporcional al error. La ecuación (9) representa su comportamiento en tiempo continuo (Castaño, 2022).

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt \dots (9)$$

Cuando los tiempos de muestreo son cortos, la ecuación (9) puede discretizarse a una ecuación en diferencias sustituyendo la acción integral por una suma. El método de integración trapezoidal es ideal para aproximar la integración continua (Castaño, 2022).

### Figura 3

*Gráfica del método de integración trapezoidal*



*Fuente: Adaptado de Castaño (2022).*

Como resultado de aplicar el método de integración trapezoidal y considerando a  $T_s$  como tiempo de muestreo se obtiene la ecuación (10).

$$A = \left( \frac{e(k) + e(k-1)}{2} \right) T_s \dots (10)$$

Por otro lado, se sabe que el área total bajo la curva de error es proporcional a la suma de las áreas de todos los trapecios que pueden dividir esta área. Aplicando este criterio al término integral de la ecuación (9) se obtiene la ecuación (11).

$$\int_0^t e(t)dt = \sum_{h=0}^k \left( \frac{e(h) + e(h-1)}{2} \right) T_s \dots (11)$$

Ahora transformaremos la ecuación (9) en su equivalente discreto. Debe notarse que la ecuación (11) es la contraparte discreta de la acción integral en tiempo continuo de la ecuación (9), por lo tanto, debe ser sustituida para producir la ecuación (12).

$$u(k) = K_p \left( e(k) + \frac{T_s}{2T_i} \sum_{h=0}^k [e(h) + e(h-1)] \right) \dots (12)$$

La señal de salida del controlador proporcional integral discreto en el instante (k-1) es:

$$u(k-1) = K_p \left( e(k-1) + \frac{T_s}{2T_i} \sum_{h=0}^k [e(h-1) + e(h-2)] \right) \dots (13)$$

Al realizar la diferencia entre la ecuación (12) y la ecuación (13) se obtiene la ecuación (14).

$$u(k) - u(k-1) = K_p \left( e(k) - e(k-1) + \frac{T_s}{2T_i} \sum_{h=0}^k [e(h) + e(h-2)] \right) \dots (14)$$

El término  $\sum_{h=0}^k [e(h) + e(h-2)]$  es equivalente a  $e(k) - e(k-1)$ , por lo tanto, la ecuación (14) se puede reescribir de la siguiente manera:

$$u(k) - u(k-1) = K_p \left( e(k) - e(k-1) + \frac{T_s}{2T_i} [e(k) - e(k-1)] \right) \dots (15)$$

$$u(k) - u(k-1) = K_p \left[ e(k) \left( 1 + \frac{T_s}{2T_i} \right) - e(k-1) \left( 1 - \frac{T_s}{2T_i} \right) \right] \dots (16)$$

La ecuación (16) se puede representar:

$$u(k) = u(k-1) + q_0 e(k) + q_1 e(k-1) \dots (17)$$

Dónde:

$$q_0 = K_c \left( 1 + \frac{T_s}{2T_i} \right) \dots (18)$$

$$q_1 = -K_c \left( 1 - \frac{T_s}{2T_i} \right) \dots (19)$$

La ecuación (20) es la función de transferencia que refleja la dinámica del controlador proporcional integral discreto y se obtiene aplicando la transformada Z a la ecuación (17).

$$C(z) = \frac{U(z)}{E(z)} = \frac{q_0 + q_1 z^{-1}}{1 - z^{-1}} \dots (20)$$

Esta sección concluye con una aproximación del controlador proporcional integral derivado (PID) discreto. El componente integral del controlador PID continuo, como se ha demostrado anteriormente, puede discretizarse utilizando la aproximación de integración trapezoidal. El término derivativo, por otro lado, puede ser discretizado usando la aproximación de Euler (Castaño, 2022; Mosquera Quimbata, 2020).

Por lo tanto, la ecuación (1) del controlador PID continuo es discretizada para obtener la ecuación (21) que representa al controlador PID discreto.

$$u(k) = K_p \left( e(k) + \frac{T_s}{2T_i} \sum_{h=0}^k [e(h) + e(h-1)] + \frac{T_d}{T_s} [e(k) + e(k-1)] \right) \dots (21)$$

La señal de salida del controlador PID discreto en el instante (k-1) es:

$$u(k-1) = K_p \left( e(k-1) + \frac{T_s}{2T_i} \sum_{h=0}^k [e(h-1) + e(h-2)] + \frac{T_d}{T_s} [e(k-1) + e(k-2)] \right) \dots (22)$$

Al realizar la diferencia entre la ecuación (21) y la ecuación (22) se obtiene la ecuación (23).

$$u(k) - u(k-1) = K_p \left[ e(k) \left( 1 + \frac{T_s}{2T_i} + \frac{T_d}{T_s} \right) - e(k-1) \left( 1 - \frac{T_s}{2T_i} + \frac{2T_d}{T_s} \right) + \frac{T_d}{T_s} e(k-2) \right] \dots (23)$$

La ecuación (23) se puede representar como:

$$u(k) = u(k-1) + q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \dots (24)$$

Dónde:

$$q_0 = K_c \left( 1 + \frac{T_s}{2T_i} + \frac{T_d}{T_s} \right) \dots (25)$$

$$q_1 = -K_c \left( 1 - \frac{T_s}{2T_i} + \frac{2T_d}{T_s} \right) \dots (26)$$

$$q_2 = \frac{K_c T_d}{T_s} \dots (27)$$

La ecuación (28) es la función de transferencia que refleja la dinámica del controlador PID discreto y se obtiene aplicando la transformada Z a la ecuación (24).

$$C(z) = \frac{U(z)}{E(z)} = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{1 - z^{-1}} \dots (28)$$

Si despejamos  $u(k)$  de la ecuación (28), entonces habremos determinado la señal de salida del controlador PID discreto.

$$u(k)(1 - z^{-1}) = q_0 e(k) + q_1 z^{-1} e(k) + q_2 z^{-2} e(k) \dots (29)$$

$$u(k) - u(k)z^{-1} = q_0 e(k) + q_1 z^{-1} e(k) + q_2 z^{-2} e(k) \dots (30)$$

$$u(k) = u(k)z^{-1} + q_0 e(k) + q_1 z^{-1} e(k) + q_2 z^{-2} e(k) \dots (31)$$

La ecuación en diferencias del controlador PID discreto se encuentra aplicando la transformada Z inversa a la ecuación (31).

$$u(k) = u(k-1) + q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \dots (32)$$

Dónde  $u(k)$  representa la ley de control actual,  $u(k-1)$  representa la ley de control anterior a un punto de muestreo,  $e(k)$  representa el error actual,  $e(k-1)$  representa el error anterior a un punto de muestreo, y  $e(k-2)$  representa el error anterior a dos puntos de muestreo.

Con la ecuación (32) es posible implementar un controlador PID discreto empleando sistemas embebidos como Arduino, Raspberry Pi Pico, microcontroladores (PIC, AVR, STM, ATMEL), entre otros.

### 1.2.2. Sistemas embebidos

Los sistemas embebidos se definen como dispositivos o equipos electrónicos que pueden realizar algún tipo de procesamiento de datos y están diseñados para realizar una función específica, a diferencia de los ordenadores de uso general (Tosini et al., 2021).

En los primeros tiempos de los sistemas embebidos, los recursos de comunicación de los microcontroladores eran muy limitados y, en la mayoría de los casos, la comunicación sólo se utilizaba para interactuar con el entorno y no para actualizar el software. Desde entonces, la comunicación inalámbrica y la conectividad a Internet están disponibles y los sistemas integrados han pasado a formar parte de sistemas más amplios, como el "Internet de los Cosas" (Valencia Luna, 2021).

Elegir un sistema embebido para un proyecto puede ser complicado. No sólo se deben considerar los criterios técnicos relacionados con el hardware y el software, sino también el tiempo y el coste, que pueden hacer fracasar el proyecto (Grajeda De los Santos, 2021).

La elección de los puertos de comunicación es una de las primeras cosas que un desarrollador debe tener en cuenta al evaluar un microcontrolador. La elección correcta de los puertos de E/S que manejan las señales de estos periféricos y de los sensores externos (por ejemplo, los de temperatura) utilizados en el diseño tiene un impacto significativo no sólo en la programación, sino también en el hardware y la cantidad de pines que necesita el sistema embebido (Restrepo Ríos y Morales Higueta, 2022).

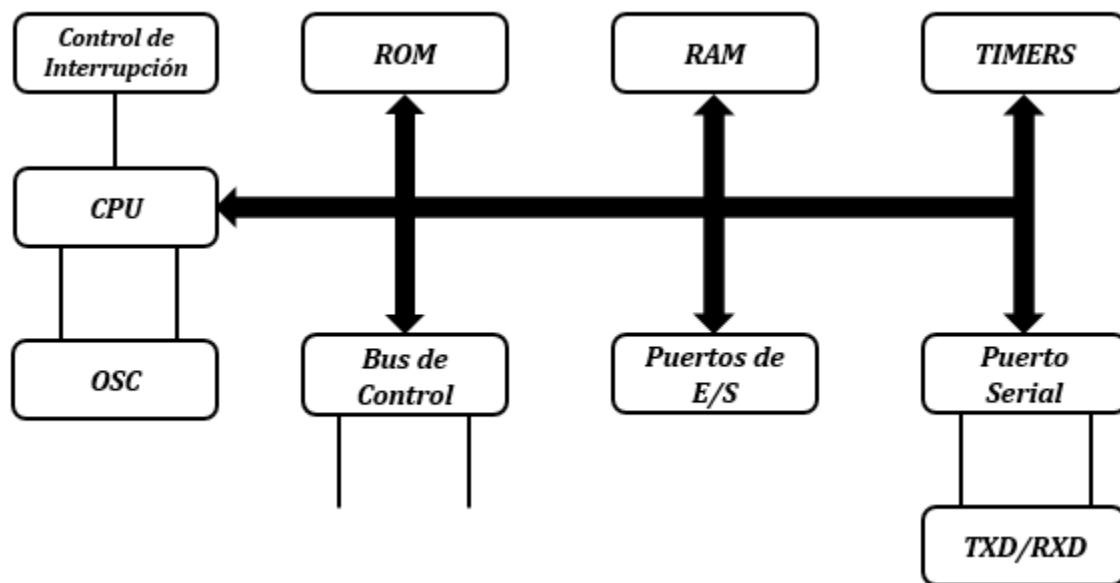
Una amplia gama de funciones de control y comandos analógicos y digitales permite utilizar sistemas embebidos en lugar de las caras y complejas placas electrónicas cableadas convencionales (Giler Cevallos, 2021).

En la Figura 4 se muestra un diagrama de bloques de un sistema embebido. La CPU se considera como el cerebro de todo equipo de procesamiento. Un bloque de interrupción se define como una subrutina que interrumpe el funcionamiento principal del sistema embebido

para iniciar otra acción. Un bus es una serie de cables que sirve de canal para transmitir datos y señales de control. Los temporizadores, en cambio, miden intervalos de tiempo para sincronizar los equipos. Los sistemas embebidos, al igual que otros circuitos digitales, son dispositivos que no pueden funcionar sin un reloj (OSC). El reloj genera una señal digital que cambia de nivel bajo (lógico 0) a nivel alto (lógico 1) muchas veces por segundo, y con cada impulso los circuitos internos realizan una operación o comando en el código compilado. El reloj sirve como referencia para sincronizar todas las tareas y transferencias de datos. Las altas frecuencias de reloj, como las de los microprocesadores, no permiten un gran consumo de energía en un solo paquete. Por otro lado, la frecuencia de reloj de un microcontrolador es inferior a la de un microprocesador convencional (GHz) (Chaljub Rojas y Troncozo Arenales, 2022).

**Figura 4**

*Diagrama de Bloques de un Sistema Embebido*



*Fuente: Adaptado de Tosini et al. (2021).*

Los sistemas embebidos tienen una amplia gama de aplicaciones y su uso depende de las necesidades del usuario. Estos dispositivos se encuentran en la mayoría de los equipos electrónicos que conocemos hoy en día. Algunos ejemplos son la industria del automóvil, la

aeronáutica y la construcción naval, los electrodomésticos, la domótica, la automatización, el control de procesos, y los sistemas de comunicación (Giler Cevallos, 2021).

Como vemos los sistemas embebidos proporcionan soluciones para resolver problemas y tomar decisiones basadas en variables del mundo real. Esto significa que el software proporciona flexibilidad y facilidad de uso, mientras que el hardware ofrece rendimiento y seguridad para apoyar la transición digital de la Industria 4.0 (Méndez de la Torre, 2022).

Una de las principales características de los sistemas embebidos es que pueden realizarse en un solo chip, ya que no requieren grandes recursos como la memoria. Además, como ya se ha mencionado, está equipado con numerosos periféricos, como temporizadores, convertidores analógico-digital (ADC), convertidores digital-analógico (DAC) e interfaces para la comunicación en serie, lo que lo hace suficientemente versátil para controlar todo tipo de equipos electrónicos (Enríquez Aguilera, 2021).

Otra característica de los sistemas embebidos es que el software que se ejecuta en él está diseñado para adaptarse al propio sistema, en lugar de ser dirigido por el usuario. El software que se ejecuta en un sistema embebido se denomina "firmware" para destacar que no es necesario cambiarlo con frecuencia. Incluso en el caso de algunos sistemas embebidos que están diseñados para ser actualizados con software, esto no suele ser así, es decir, existe una estrecha relación entre el software del sistema embebidos y el hardware sobre el que se ejecuta (Flores Jacome y Sillagana Guano, 2022).

Los sistemas embebidos también tienen un sistema operativo, pero no es comparable al IOS o a Windows de los ordenadores de uso general. En cambio, un sistema operativo integrado actúa como un programador de tareas que coordina las tareas en tiempo real, como la lectura de pulsadores y sensores. Los sistemas operativos integrados se conocen como "sistemas operativos en tiempo real" (Romero Valladares, 2021).



Hay muchos tipos diferentes de arquitecturas de sistemas embebidos y, en general, la arquitectura utilizada en un sistema está estrechamente relacionada con el acoplamiento del hardware y el software. Sin embargo, la gran variedad de arquitecturas de software plantea varios problemas, como la portabilidad, la reutilización, el mantenimiento y la fiabilidad. Los problemas de comunicación e intercambio entre el SO y las bibliotecas, así como la interfaz entre las bibliotecas y el SO, son también cuestiones importantes (Enríquez Aguilera, 2021).

Las arquitecturas tradicionales de los sistemas embebidos constan de seis módulos. Estos módulos, de menor a mayor, son: hardware, capas periféricas, sistema operativo, archivos de sistema, bibliotecas y aplicaciones. El uso de arquitecturas tradicionales plantea varios problemas para la propia arquitectura, como la reutilización de componentes, el intercambio de información entre capas, la portabilidad de los módulos de software, las dependencias entre plataformas y la flexibilidad entre sistemas (Chaljub Rojas y Troncozo Arenales, 2022).

Para resolver estos problemas, existe una capa en la arquitectura de un sistema embebido, comúnmente denominada middleware. Esta capa proporciona la interfaz para la comunicación entre las capas de aplicación y de servicio del sistema y resuelve los problemas de portabilidad, reutilización y dependencia del hardware al unificar las interfaces de la arquitectura (Restrepo Ríos y Morales Higueta, 2022).

En otras palabras, en nivel de middleware se inicializa el transceptor, se configura el puerto y se inicializa el controlador para que el hardware pueda transmitir datos de comunicación. Una vez inicializado el hardware de comunicación, se prepara la pila de comunicación. Esta pila proporciona interfaces y servicios para que las capas superiores, normalmente desarrolladas en middleware, puedan establecer la comunicación entre la capa de aplicación y el hardware. Esta separación de la aplicación y el hardware se consigue en lo que respecta a la comunicación: el hardware puede generarse o configurarse para utilizar protocolos

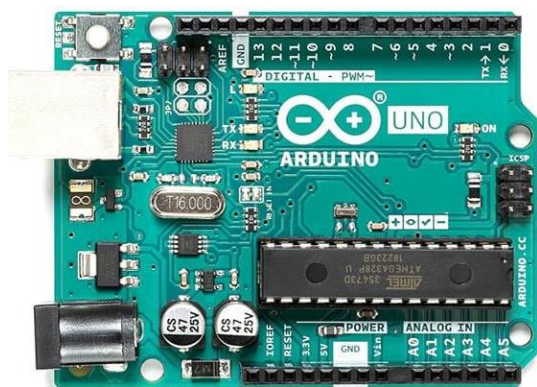
de comunicación como SPI, UART, I2C, etc., pero el middleware abstrae esta configuración para que no sea relevante (Valencia Luna, 2021).

Los avances en la arquitectura de hardware de los sistemas embebidos han permitido aprovechar plenamente la memoria e incrementar el rendimiento del código por unidad de tiempo, lo que permite ejecutar tareas complejas, pero manteniendo las características de bajo consumo de energía. Advanced Risc Machine (ARM) domina el mercado mundial de la electrónica y están integrados en dispositivos con los que entramos en contacto a diario, como smartphones, tabletas, electrónica de consumo, la industria biomédica y el desarrollo de hardware (Torres Correa y Valencia Gómez, 2021).

Por ello, los dispositivos electrónicos modernos no sólo son capaces de realizar múltiples tareas, sino que también cuentan con interfaces de operador que permiten una interacción del usuario final con el proceso. Debido a esta multifuncionalidad, los diseñadores de electrónica están obligados a tener amplios conocimientos de las distintas tecnologías para garantizar que sus productos cumplan los requisitos y necesidades actuales. Por lo tanto, es importante actualizar las herramientas propias para ser realmente competitivo en el campo del desarrollo electrónico (Torres Correa y Valencia Gómez, 2021).

## Figura 5

*Sistema embebido*



*Fuente: obtenido de*

<https://www.mouser.pe/ProductDetail/Arduino/A000066?qs=BC3YYPaifMrIue9b%252BHtKQg%3D%3D>

### 1.3. Operacionalización de variables

**Tabla 2**

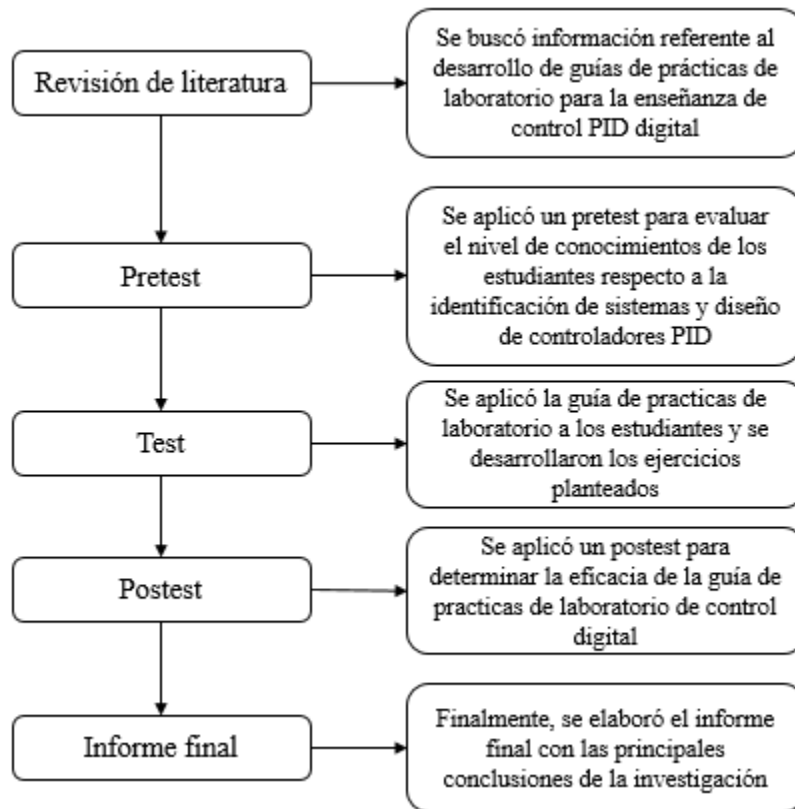
*Definición y operacionalización de variables*

Variables	Definición conceptual	Definición operacional	Dimensiones	Instrumento
Guía de prácticas de laboratorio (Independiente)	Se podría definir como un documento que proporciona instrucciones y directrices detalladas para llevar a cabo un experimento o práctica de laboratorio en un campo particular, como la ingeniería o la ciencia. (Bejerano Abarca, 2014).	Esta variable se evaluará por su efectividad en el logro de los objetivos de aprendizaje	Contenido	Análisis de la tasa de éxito
			Estructura	
			Enfoque pedagógico	
			Nivel de complejidad	
Mejorar la enseñanza de control digital (Dependiente)	Metodología que permite la experimentación práctica directa de los conceptos teóricos esenciales de un sistema de control real (Fortet Roura et al., 1996).	Esta variable se medirá mediante pruebas de conocimientos previos y posteriores a la implementación de la guía de prácticas de laboratorio de control digital.	Efectividad	Pretest y posttest
			Mejora en la capacidad de identificar sistemas de primer orden más retardo	
			Mejora en la capacidad de diseñar controladores PID digitales	

## II. DISEÑO METODOLÓGICO

### 2.1. Diseño de contrastación de hipótesis

A continuación, se muestra un diagrama de flujo indicando los pasos seguidos para comprobar la hipótesis planteado en la investigación.



### 2.2. Población y muestra

La población está formada por los alumnos de la EPIE – UNPRG. La muestra está compuesta por 25 estudiantes. El tipo de muestreo fue no probabilístico por conveniencia.

### 2.3. Técnicas, instrumentos, materiales y equipos

#### 2.3.1. Técnicas e instrumentos de recolección de datos

En este estudio se utilizaron dos técnicas de investigación para recopilar información relevante: la revisión de literatura y la encuesta a través de un cuestionario.

A través de la revisión de literatura se obtuvieron pautas valiosas para seleccionar un sistema embebido y desarrollar una guía de prácticas de laboratorio en el ámbito de la enseñanza del control digital.

En la Tabla 3 se muestran los criterios utilizados en el análisis técnico y económico para la selección del sistema embebido.

**Tabla 3**

*Análisis técnico y económico para seleccionar el sistema embebido*

Sistema embebido	CPU	GPIO	Comunicación	Costo
Arduino Leonardo	ATMega32u4	20	USB, UART, SPI, I2C.	S/ 159.00
Raspberry Pi 4	ARM Cortex A72	40	USB, UART, SPI, I2C, Wi-Fi, BLE.	S/ 395.00
FPGA DE10-Lite	Altera MAX10	40	USB, UART, SPI, I2C.	S/ 450.00

Fuente: <https://docs.arduino.cc/hardware/leonardo>; <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>; <https://www.mouser.pe/new/terasic-technologies/terasic-de10-lite-board/>

Arduino Leonardo es un microcontrolador basado en la arquitectura AVR de ATMEL. Tiene un bajo consumo de energía gracias a la tecnología "Picopower" y un oscilador de cristal de 16 MHz. Además, cuenta con 20 pines digitales configurables como E/S, pines de entrada analógica y de salida PWM, y múltiples interfaces para la comunicación con periféricos.

Raspberry Pi 4 cuenta con un procesador ARM Cortex A72 de cuatro núcleos de 64 bits a 1.5 GHz y un módulo SoC Tiene modelos de 1 GB a 8 GB de SDRAM, 40 GPIO y múltiples interfaces de comunicación como USB, UART, SPI, I2C, HDMI, Wi-Fi y Bluetooth. Requiere un sistema operativo basado en Linux para su configuración y funcionamiento.

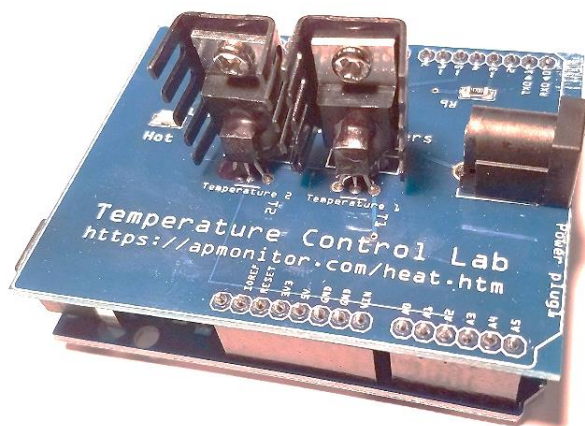
FPGA DE10-Lite incluye un procesador Altera MAX10 de dos núcleos, 64MB de SDRAM y un oscilador de cristal de 25 MHz. Tiene 40 GPIO configurables como E/S y un ADC de 12 bits. Dispone de interfaces USB, UART, SPI e I2C para la comunicación con los periféricos.

Después de realizar un análisis, se concluyó que el sistema embebido más adecuado para la guía de prácticas de laboratorio es Arduino Leonardo. Las otras dos opciones fueron descartadas, ya que su uso no está justificado para aplicaciones que no requieren demasiado procesamiento. Estas plataformas se utilizan principalmente en proyectos que involucran procesamiento digital de señales e imágenes, procesamiento de video, comunicaciones inalámbricas, Internet de las Cosas, entre otros.

Se decidió utilizar la herramienta TCLab debido a que está basada en Arduino. TCLab es un laboratorio portátil para la enseñanza de la teoría de control utilizando hardware y software libre para la identificación de modelos dinámicos, la diferencia entre el control manual y el automático, los procedimientos de prueba para generar datos dinámicos, el ajuste de los datos dinámicos a un modelo de primer orden más retardo (FOPDT), la determinación de los parámetros de control PID mediante reglas de sintonización estándar y su ajuste para mejorar su eficiencia.

### Figura 6

*Sistema embebido seleccionado*

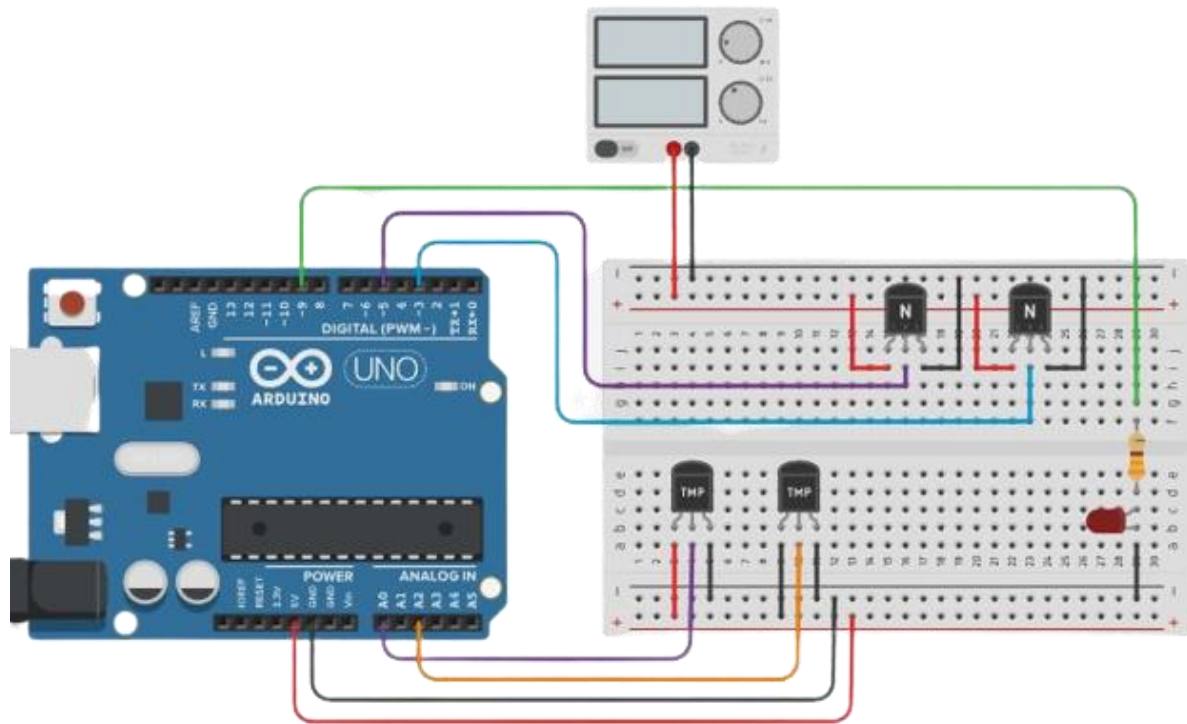


Fuente: obtenido de <https://www.amazon.in/TCLab-Temperature-Control-Lab/dp/B07QJSSFVC>

TCLab está compuesto por un Arduino Leonardo, un LED, dos transistores NPN TIP31C y dos sensores de temperatura TMP36. La fuente de energía de los transistores es externa (5V y 2A), como se muestra en la Figura 7-

**Figura 7**

*Conexiones de TCLAB*



TCLab funciona mediante la inyección de corriente en los pines PWM (3 y 5), lo que produce un calentamiento de los transistores NPN y un aumento en su temperatura interna. Para medir esta temperatura, se coloca un sensor en los disipadores de calor de los transistores (como se muestra en la Figura 7) y se envía la señal al Arduino UNO a través de los terminales de entrada analógica (A0 y A2).

Si bien TCLab es un sistema MIMO (Multiple Input Multiple Output), para la guía de prácticas de laboratorio se empleó como un sistema SISO (Single Input Single Output). En particular, se utilizó el pin A0 como entrada analógica para el sensor de temperatura, el pin 3 como salida PWM y el pin 9 para el LED. De esta manera, se simplificó el sistema para enfocarse en el control de la temperatura a través de la variación del ciclo de trabajo de la señal PWM.

En la Tabla 4 se muestran las principales referencias bibliográficas empleadas para el desarrollo de la guía de prácticas de laboratorio de control digital en esta investigación.

**Tabla 4**

*Principales referencias bibliográficas*

<b>Título</b>	<b>Autor(es)</b>	<b>Conclusión</b>
Uso del paradigma Take-Home Labs para la enseñanza del control automático en estudios de ingeniería.	Hoyo, Á., García Mañas, F., Ramos Teodoro, J., Sánchez Molina, J. A., Guzmán, J. L., y Rodríguez, F. (2021).	Los laboratorios portátiles de bajo costo mejoran la enseñanza de los estudiantes de ingeniería de control porque les permite aplicar estrategias de control en sistemas reales.
Introducing Digital Controllers to Undergraduate Students using the TCLab Arduino Kit.	Moura Oliveira, P. B., Hedengren, J. D., y Rossiter, J. A. (2020).	Los laboratorios portátiles de bajo costo son una herramienta efectiva para experimentar y mejorar el aprendizaje de los controladores digitales tanto para profesores como para estudiantes.
Tools and Control Experiences using TCLab Arduino Kit.	Marques Aguiar, B. F. (2020).	Los laboratorios portátiles de bajo costo permiten que los estudiantes identifiquen un modelo dinámico de primer orden más retardo y aplicar distintas técnicas de control digital.
Diseño e implementación de un prototipo de modulo didáctico de caudal para medir su influencia en el aprendizaje de la instrumentación y control industrial.	Ruíz Pérez, J. R. (2019).	Una guía de prácticas de laboratorio tiene una influencia positiva y estadísticamente significativa en el rendimiento de los estudiantes de ingeniería de control.

*Fuente: Hoyo et al. (2021), Oliveira et al. (2020), Marques Aguiar (2020) y Ruiz Pérez (2019)*



### III. RESULTADOS

#### 3.1. Identificación del modelo matemático de la planta

Se utilizó el método de identificación por curva de reacción para estimar la función de transferencia del sistema termodinámico TCLAB. Este método permite caracterizar la planta de forma experimental. La idea es aproximar el modelo matemático como un sistema de primer orden más retardo de la siguiente manera.

$$G(S) = \frac{K e^{-\theta s}}{\tau s + 1} \dots (33)$$

Dónde: K es la ganancia estática,  $\theta$  es retardo de tiempo y  $\tau$  es la constante de tiempo del sistema.

En este experimento, el valor de la temperatura del transistor se mantuvo igual a la temperatura ambiente y el valor del PWM se mantuvo en 0%, como se muestra en la Figura 8.

**Figura 8**

*Valores de temperatura y PMW al iniciar la prueba experimental*

```
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.65, PWM (%) : 0
Temperatura (°C) : 19.65, PWM (%) : 0
Temperatura (°C) : 19.65, PWM (%) : 0
Temperatura (°C) : 19.65, PWM (%) : 0
Temperatura (°C) : 19.65, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.65, PWM (%) : 0
Temperatura (°C) : 19.65, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 0
```

A continuación, se aplicó el 50% del PWM del Arduino en el transistor y se esperó durante un periodo de 1718 segundos hasta que el transistor se calentó y el valor de la

temperatura se estabilizó en 52.58 °C. Las gráficas y valores de la temperatura y el PWM se visualizaron en el Serial Monitor y el Serial Plotter, como se muestra en la Figuras 9 y 10.

**Figura 9**

*Valores de temperatura y PMW al terminar la prueba experimental*

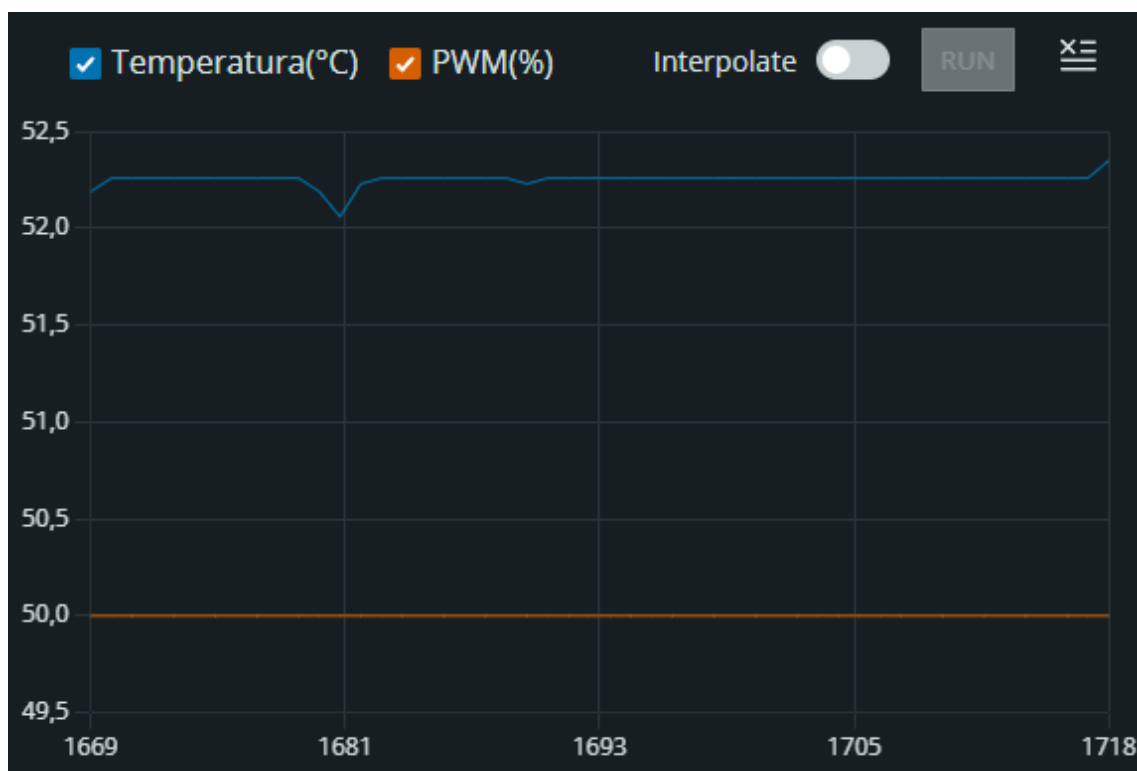
```

Temperatura (°C) : 52.26, PWM (%) : 50
Temperatura (°C) : 52.35, PWM (%) : 50
Temperatura (°C) : 52.52, PWM (%) : 50
Temperatura (°C) : 52.58, PWM (%) : 50
Temperatura (°C) : 52.58, PWM (%) : 50
Temperatura (°C) : 52.58, PWM (%) : 50
Temperatura (°C) : 52.58, PWM (%) : 50
Temperatura (°C) : 52.58, PWM (%) : 50
Temperatura (°C) : 52.58, PWM (%) : 50

```

**Figura 10**

*Gráficas de temperatura y PMW al terminar la prueba experimental*



De las Figuras 8, 9 y 10 se puede concluir que la temperatura inicial es de 19.65°C y la final de 52.58°C. Para conseguir este aumento de temperatura en el transistor, se aplicó el 50% de PWM del Arduino. Luego se utilizó la siguiente ecuación para determinar la ganancia estática del sistema:

$$K = \frac{T_{Final} - T_{Inicial}}{U_{Final} - U_{Inicial}} = \frac{52.58^{\circ}C - 19.65^{\circ}C}{50\% - 0\%} = 0.6586 \dots (34)$$

Para calcular la constante de tiempo se dividió el tiempo de estabilización por 4:

$$\tau = \frac{T_{ss}}{4} = \frac{1718}{4} = 429.5 \dots (35)$$

Para calcular el retardo de tiempo se empleó la información del Serial Monitor que se muestra en la Figura 11. Se puede apreciar que la temperatura inicial tarda aproximadamente 10 segundos para cambiar de 19.68°C a 19.81°C luego de aplicar el PWM al 50%. Por lo tanto:

$$\theta = 10 \dots (36)$$

### Figura 11

*Tiempo de respuesta del sistema luego de aplicar la señal escalón*

```

Temperatura (°C) : 19.68, PWM (%) : 0
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.65, PWM (%) : 50
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.68, PWM (%) : 50
Temperatura (°C) : 19.81, PWM (%) : 50
Temperatura (°C) : 19.97, PWM (%) : 50
Temperatura (°C) : 20.00, PWM (%) : 50
Temperatura (°C) : 20.00, PWM (%) : 50
Temperatura (°C) : 20.16, PWM (%) : 50
Temperatura (°C) : 20.32, PWM (%) : 50
Temperatura (°C) : 20.32, PWM (%) : 50
Temperatura (°C) : 20.52, PWM (%) : 50
Temperatura (°C) : 20.65, PWM (%) : 50

```

Basándose en la información anterior, el modelo matemático del sistema podría determinarse mediante la siguiente ecuación:

$$G(S) = \frac{K e^{-\theta s}}{\tau s + 1} = \frac{0.6586 e^{-10s}}{429.5s + 1} \dots (37)$$

### 3.2. Prueba de los algoritmos de control PID digital

En este apartado se comprueba el rendimiento los controladores PID en tiempo discreto, Ziegler y Nichols, Cohen y Coon y la Integral del Error. Los detalles del diseño se pueden encontrar en el apartado de ANEXOS.

#### 3.2.1. Control PID digital con el primer método de Ziegler y Nichols

En las Figuras 12 y 13 se observa que la temperatura inicial es 23.23°C, el valor de la referencia es 0°C y el valor de PMW es 0%.

**Figura 12**

*Gráfica de valores iniciales de temperatura y PWM – Ziegler y Nichols*



**Figura 13**

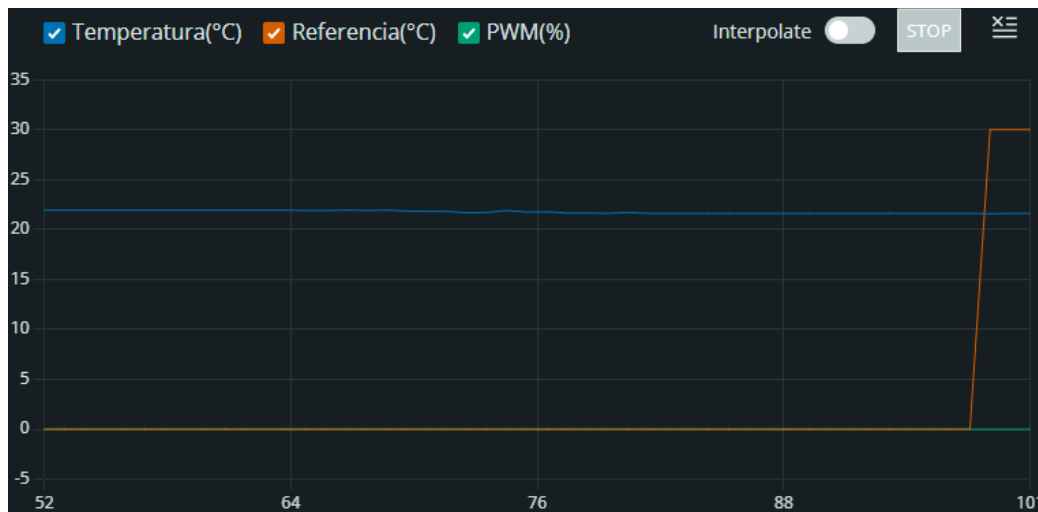
*Lectura de valores iniciales de temperatura y PWM – Ziegler y Nichols*

```
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 23.23, Referencia(°C): 0.00, PWM(%): 0
```

Después de unos 100 segundos, se cambió la referencia del sistema a 30°C. De esta manera, el valor del PWM cambió de acuerdo con la ley de control calculada por el sistema embebido, como se muestra en la Figura 14.

**Figura 14**

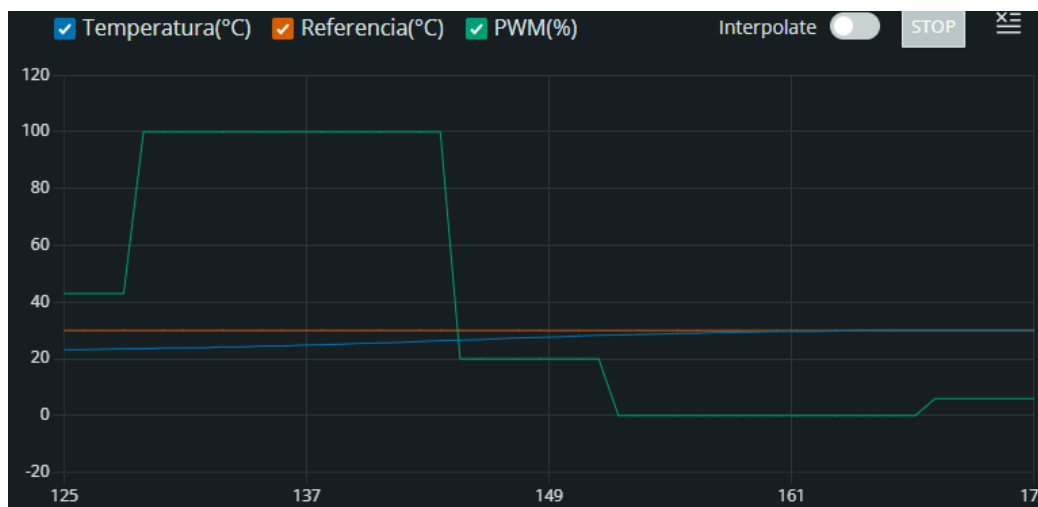
*Gráfica de respuesta del sistema ante el cambio de referencia a 30°C – Ziegler y Nichols*



Con el tiempo, la precisión del sistema mejoró. Así, antes de los 174 segundos, los valores de temperatura del sistema ya eran iguales al valor de referencia (30°C), como se muestra en las Figuras 15 y 16.

**Figura 15**

*Gráfica de respuesta del sistema igual a la referencia de 30°C – Ziegler y Nichols*



**Figura 16**

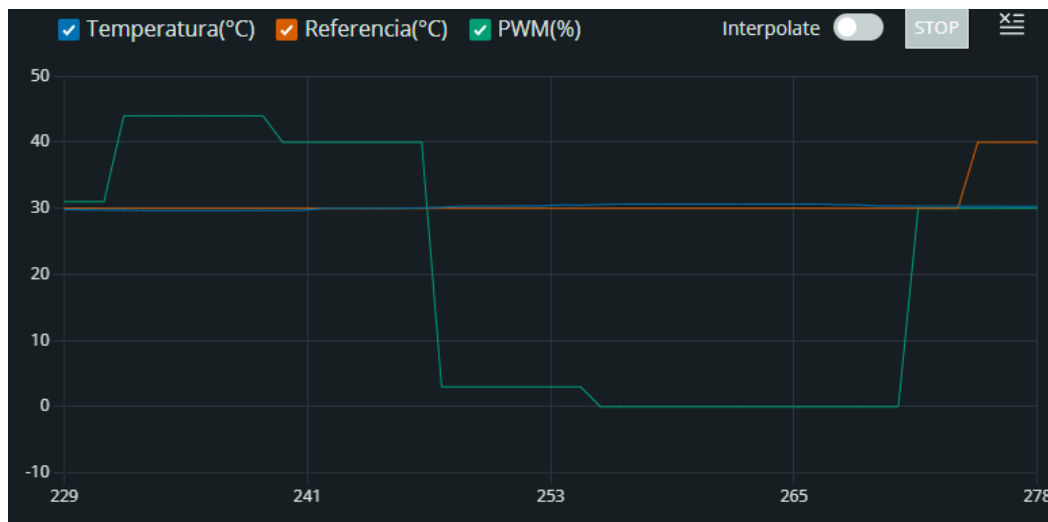
*Lectura de valores de respuesta del sistema igual a la referencia de 30°C – Ziegler y Nichols*

```
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 6
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 6
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%) : 32
```

A los 270 segundos, se cambió la referencia del sistema a 40°C. De esta manera, el valor del PWM cambió de acuerdo con la ley de control calculada por el sistema embebido, como se muestra en la Figura 17.

**Figura 17**

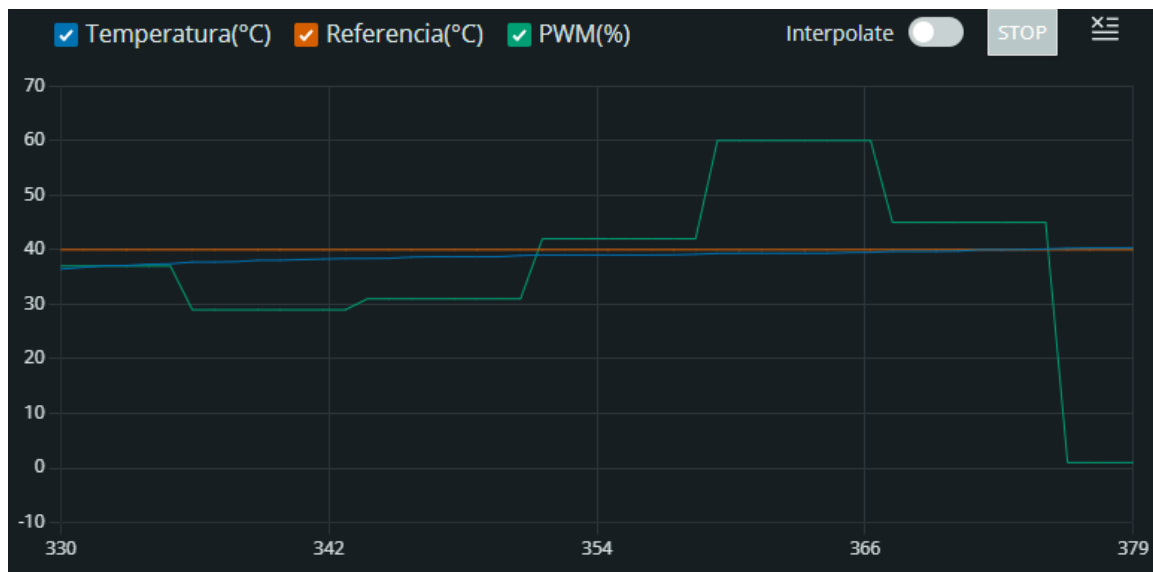
*Gráfica de respuesta del sistema ante el cambio de referencia de 40°C – Ziegler y Nichols*



Con el tiempo, la precisión del sistema mejoró. Así, antes de los 379 segundos, los valores de temperatura del sistema ya eran iguales al valor de referencia (40°C), como se muestra en las Figuras 18 y 19.

**Figura 18**

*Gráfica de respuesta del sistema igual a la referencia de 40°C – Ziegler y Nichols*



**Figura 19**

*Lectura de valores de respuesta del sistema igual a la referencia de 40°C – Ziegler y Nichols*

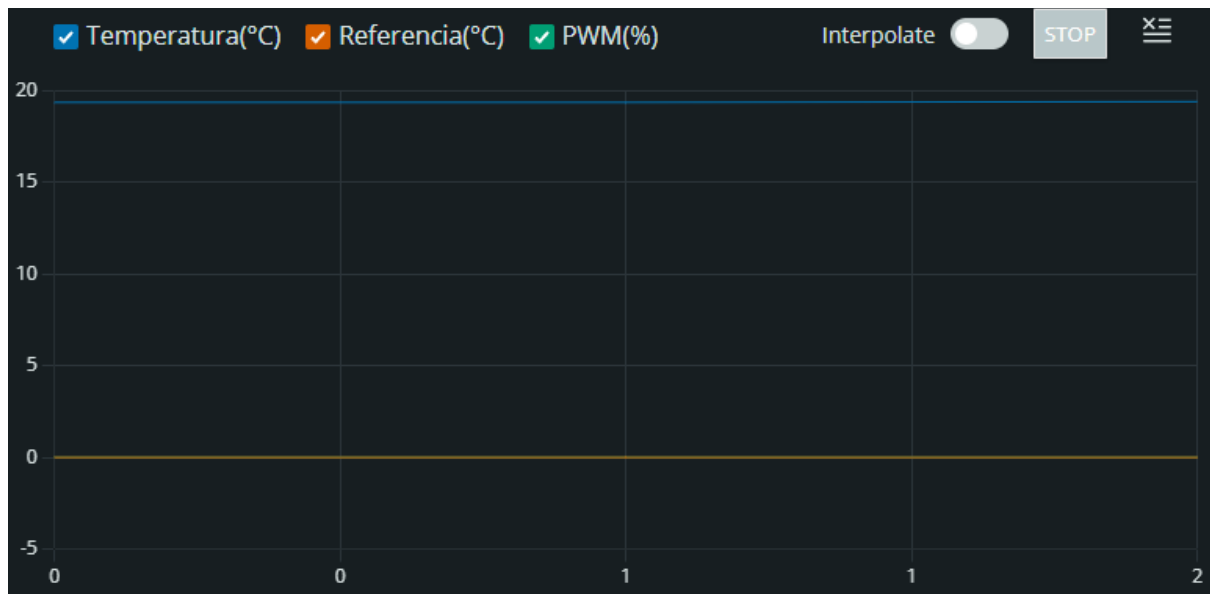
```
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 37
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 37
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 37
Temperatura(°C): 40.03, Referencia(°C): 40.00, PWM(%): 37
Temperatura(°C): 40.03, Referencia(°C): 40.00, PWM(%): 37
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 37
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 37
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 37
```

### 3.2.2. Control PID digital con el método de Cohen y Coon

En las Figuras 20 y 21 se observa que la temperatura inicial es 23.23°C, el valor de la referencia es 0°C y el valor de PWM es 0%.

**Figura 20**

*Gráfica de valores iniciales de temperatura y PWM – Cohen y Coon*



**Figura 21**

*Lectura de valores iniciales de temperatura y PWM – Cohen y Coon*

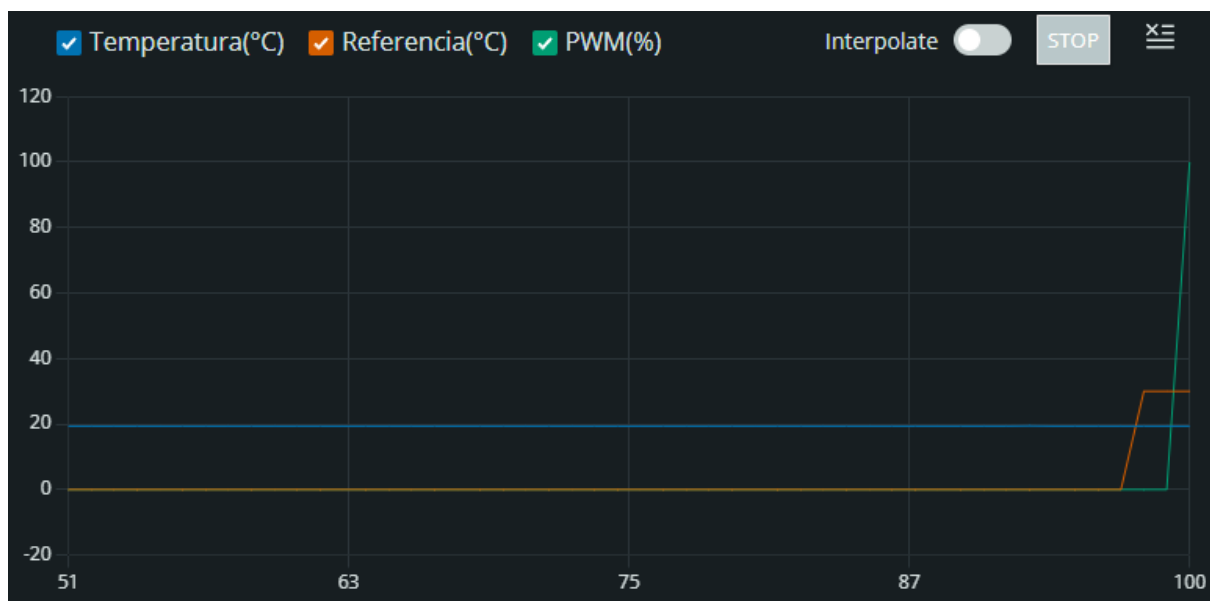
```
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
```

Antes de los 100 segundos, se cambió la referencia del sistema a 30°C. De esta manera, el valor del PWM cambió de acuerdo con la ley de control calculada por el sistema embebido, como se muestra en la Figura 22.



**Figura 22**

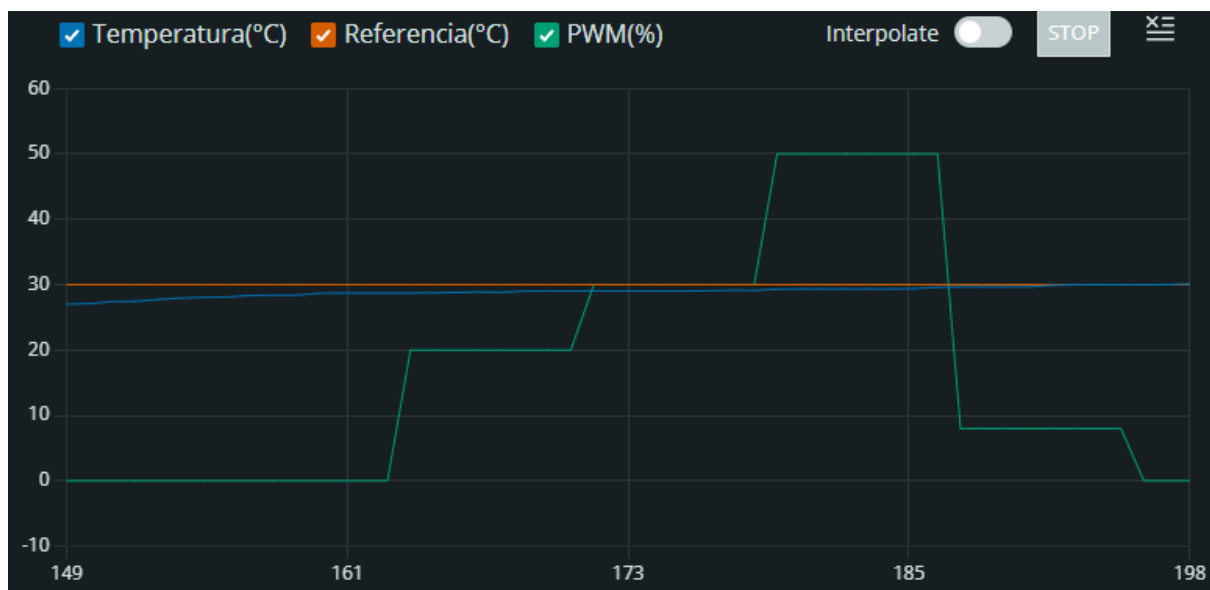
*Gráfica de respuesta del sistema ante el cambio de referencia a 30°C – Cohen y Coon*



Con el tiempo, la precisión del sistema mejoró. Así, antes de los 198 segundos, los valores de temperatura del sistema ya eran iguales al valor de referencia (30°C), como se muestra en las Figuras 23 y 24.

**Figura 23**

*Gráfica de respuesta del sistema igual a la referencia de 30°C – Cohen y Coon*



**Figura 24**

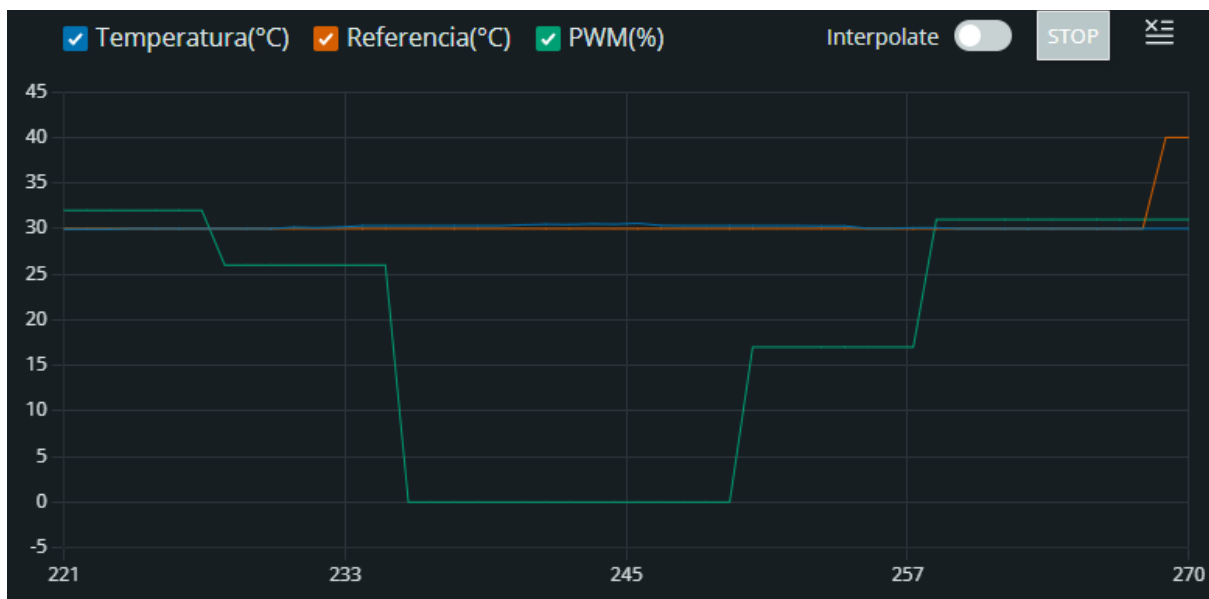
*Lectura de valores de respuesta del sistema igual a la referencia de 30°C – Cohen y Coon*

```
Temperatura(°C): 29.94, Referencia(°C): 30.00, PWM(%): 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 32
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 26
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 26
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 26
Temperatura(°C): 30.16, Referencia(°C): 30.00, PWM(%): 26
```

Antes de los 270 segundos, se cambió la referencia del sistema a 40°C. De esta manera, el valor del PWM cambió de acuerdo con la ley de control calculada por el sistema embebido, como se muestra en la Figura 25.

**Figura 25**

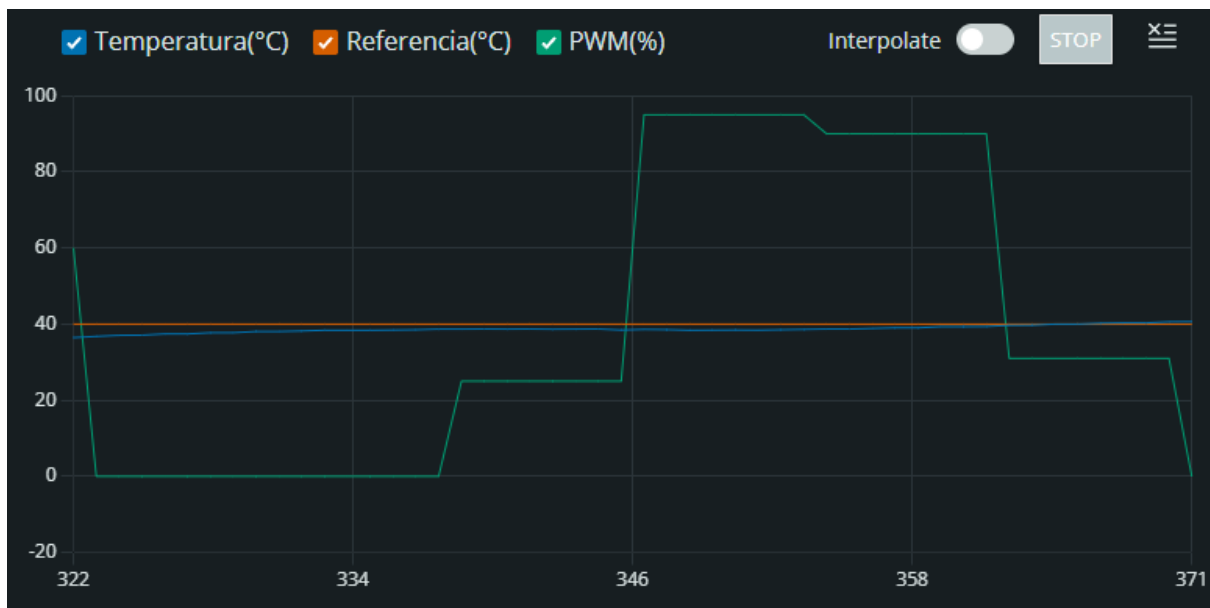
*Gráfica de respuesta del sistema ante el cambio de referencia de 40°C – Cohen y Coon*



Con el tiempo, la precisión del sistema mejoró. Así, antes de los 371 segundos, los valores de temperatura del sistema ya eran iguales al valor de referencia (40°C), como se muestra en las Figuras 26 y 27.

**Figura 26**

*Gráfica de respuesta del sistema igual a la referencia de 40°C – Cohen y Coon*



**Figura 27**

*Lectura de valores de respuesta del sistema igual a la referencia de 40°C – Cohen y Coon*

```
Temperatura(°C): 39.97, Referencia(°C): 40.00, PWM(%): 58
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 58
Temperatura(°C): 40.10, Referencia(°C): 40.00, PWM(%): 58
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 58
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 58
Temperatura(°C): 40.39, Referencia(°C): 40.00, PWM(%): 58
Temperatura(°C): 40.65, Referencia(°C): 40.00, PWM(%): 58
Temperatura(°C): 40.65, Referencia(°C): 40.00, PWM(%): 0
Temperatura(°C): 40.71, Referencia(°C): 40.00, PWM(%): 0
```

### 3.2.3. Control PID digital con el método de la Integral del Error

En las Figuras 28 y 29 se observa que la temperatura inicial es 19.35°C, el valor de la referencia es 0°C y el valor de PWM es 0%.

**Figura 28**

*Gráfica de valores iniciales de temperatura y PWM – Integral del Error*



**Figura 29**

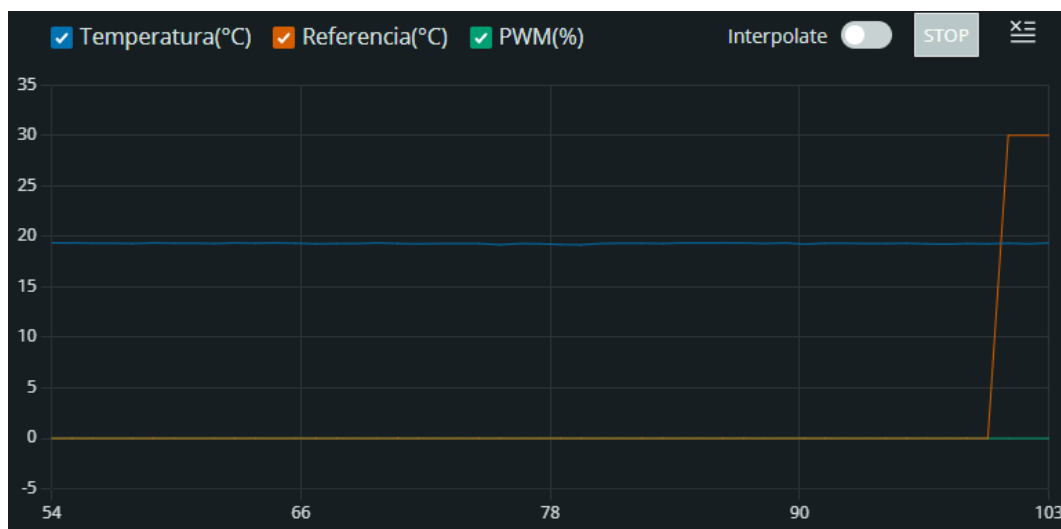
*Lectura de valores iniciales de temperatura y PWM – Integral del Error*

```
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
Temperatura(°C): 19.35, Referencia(°C): 0.00, PWM(%): 0
```

Después de unos 100 segundos, se cambió la referencia del sistema a 30°C. De esta manera, el valor del PWM cambió de acuerdo con la ley de control calculada por el sistema embebido, como se muestra en la Figura 29.

**Figura 30**

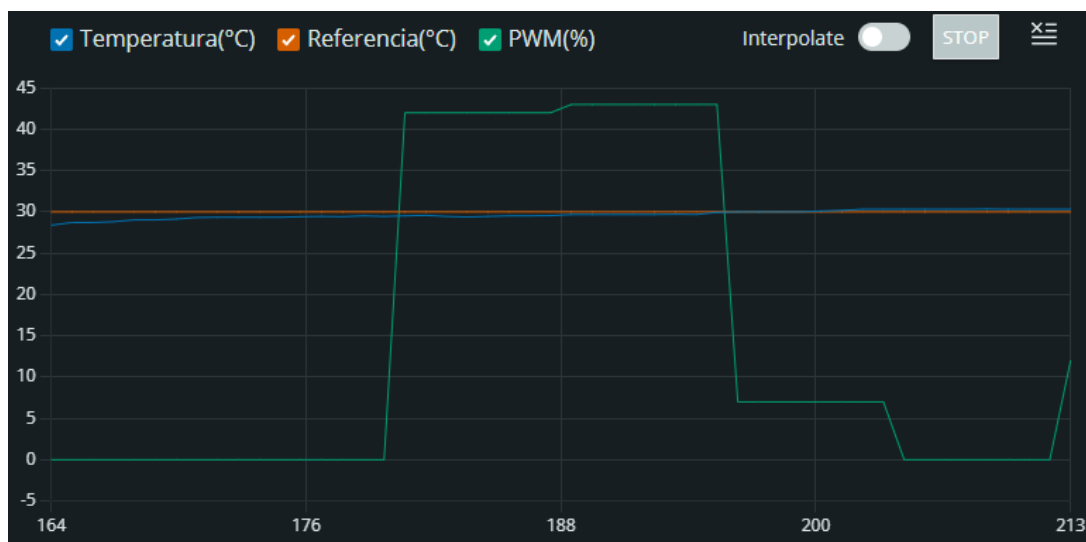
*Gráfica de respuesta del sistema ante el cambio de referencia a 30°C – Integral del Error*



Con el tiempo, la precisión del sistema mejoró. Así, antes de los 213 segundos, los valores de temperatura del sistema ya eran iguales al valor de referencia (30°C), como se muestra en las Figuras 31 y 32.

**Figura 31**

*Gráfica de respuesta del sistema igual a la referencia de 30°C – Integral del Error*



**Figura 32**

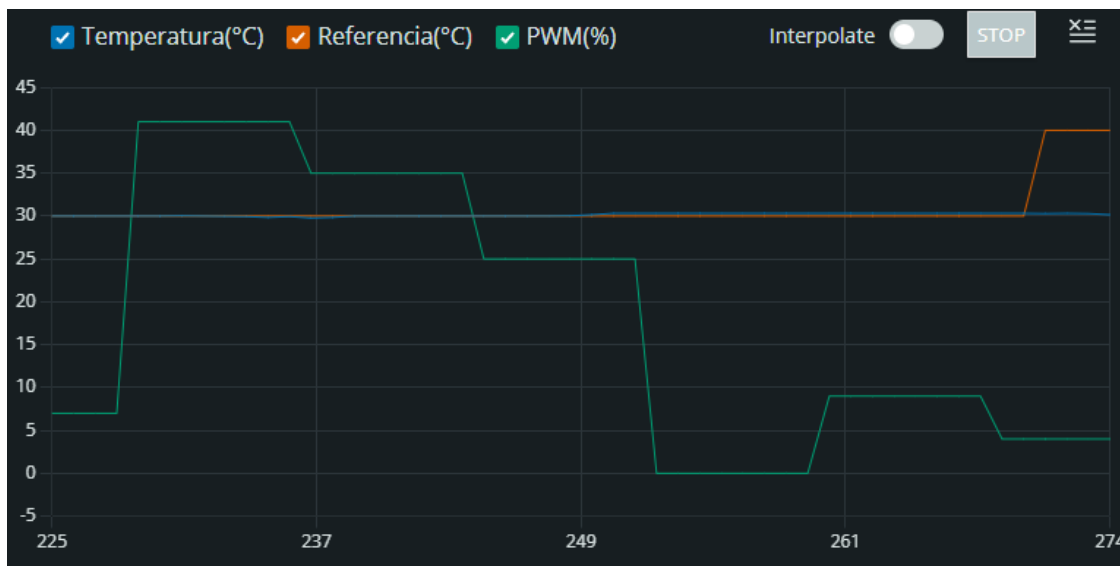
*Lectura de valores de respuesta del sistema igual a la referencia de 30°C – Integral del Error*

```
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 7
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 41
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 41
Temperatura(°C): 30.03, Referencia(°C): 30.00, PWM(%): 41
Temperatura(°C): 30.00, Referencia(°C): 30.00, PWM(%): 41
Temperatura(°C): 29.97, Referencia(°C): 30.00, PWM(%): 41
Temperatura(°C): 29.94, Referencia(°C): 30.00, PWM(%): 41
Temperatura(°C): 29.84, Referencia(°C): 30.00, PWM(%): 41
Temperatura(°C): 29.94, Referencia(°C): 30.00, PWM(%): 41
```

A los 270 segundos, se cambió la referencia del sistema a 40°C. De esta manera, el valor del PWM cambió de acuerdo con la ley de control calculada por el sistema embebido, como se muestra en la Figura 33.

**Figura 33**

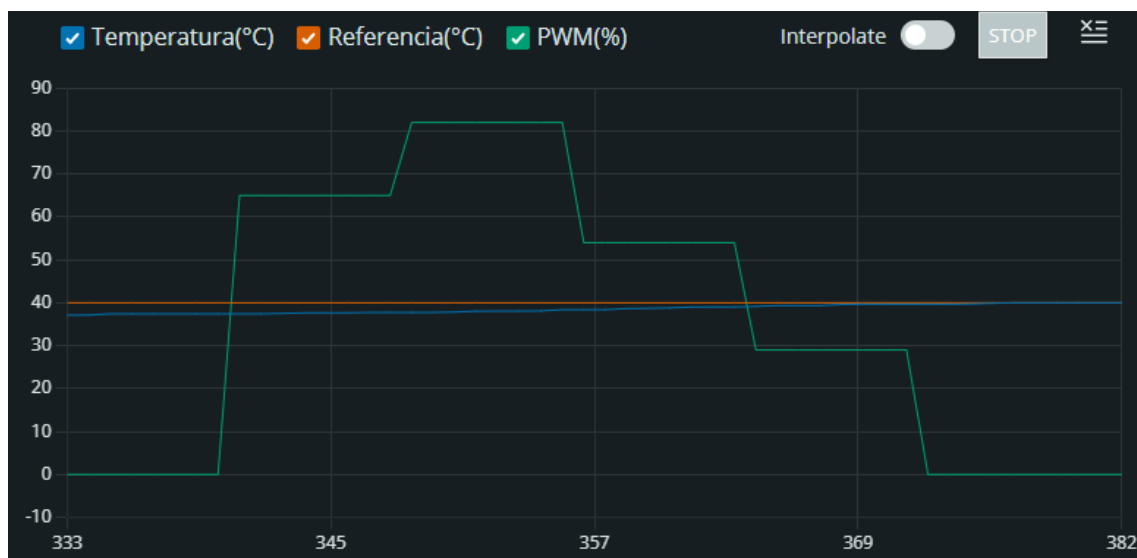
*Gráfica de respuesta del sistema ante el cambio de referencia de 40°C – Integral del Error*



Con el tiempo, la precisión del sistema mejoró. Así, antes de los 382 segundos, los valores de temperatura del sistema ya eran iguales al valor de referencia (40°C), como se muestra en las Figuras 34 y 35.

**Figura 34**

*Gráfica de respuesta del sistema igual a la referencia de 40°C – Integral del Error*



**Figura 35**

*Lectura de valores de respuesta del sistema igual a la referencia de 40°C – Integral del Error*

```
Temperatura(°C): 39.35, Referencia(°C): 40.00, PWM(%): 91
Temperatura(°C): 39.48, Referencia(°C): 40.00, PWM(%): 91
Temperatura(°C): 39.68, Referencia(°C): 40.00, PWM(%): 91
Temperatura(°C): 39.74, Referencia(°C): 40.00, PWM(%): 91
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 91
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 8
Temperatura(°C): 40.03, Referencia(°C): 40.00, PWM(%): 8
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 8
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 8
```

### 3.3. Aplicación y resultados del pretest

Se elaboró un cuestionario de 10 preguntas con respuestas de opción múltiple en la plataforma Google Forms. El objetivo de aplicar el pretest fue evaluar el nivel de conocimiento de los 25 estudiantes de la EPIE - UNPRG respecto a la identificación de sistemas y el diseño de algoritmos de control PID digital.

El cuestionario se envió por correo electrónico (institucional) y cada pregunta tuvo un puntaje de 2 puntos, lo que significa que la calificación más alta posible fue 20. A continuación, se muestran las 10 preguntas y las opciones de respuesta del cuestionario aplicado.

1. ¿Qué significa TCLAB? \*

- ☐ Technology Control Lab
- ☐ Teaching Control Lab
- ☐ Temperature Control Lab
- ☐ Technical Control Laboratory

2. ¿Qué tipo de sistemas se pueden identificar utilizando TCLAB? \*

- ☐ Sistemas de segundo orden
- ☐ Sistemas lineales
- ☐ Sistemas no lineales
- ☐ Sistemas de primer orden con retardo

3. ¿Cuál es el propósito de la identificación de modelos en un sistema de control? \*

- ☐ Optimizar el controlador PID
- ☐ Mejorar la precisión del sistema
- ☐ Determinar los parámetros del modelo del sistema
- ☐ Todas las anteriores

4. ¿Qué parámetros se deben ajustar en el modelo de primer orden con retardo para obtener una buena aproximación del sistema? \*

- ☐ El tiempo muerto y la constante de tiempo
- ☐ La ganancia y el tiempo muerto
- ☐ La ganancia y la constante de tiempo

5. ¿Cuál es la principal ventaja de utilizar el método de Ziegler-Nichols para sintonizar un controlador PID? \*

- ☐ Es el método más rápido y preciso
- ☐ Requiere poca o ninguna matemática
- ☐ Puede ser aplicado a cualquier sistema de control
- ☐ No tiene ninguna ventaja en particular



6. ¿Cuál es la principal ventaja del método de Cohen y Coon sobre el método de Ziegler-Nichols? \*

- ☐ El método de Cohen y Coon es más fácil de implementar
- ☐ El método de Cohen y Coon es más preciso
- ☐ El método de Cohen y Coon es más rápido
- ☐ El método de Cohen y Coon es más barato

7. ¿Qué parámetros del controlador se calculan con el método de la Integral del Error? \*

- ☐  $K_p$  y  $T_i$
- ☐  $K_p$  y  $T_d$
- ☐  $T_i$  y  $T_d$
- ☐  $K_p$ ,  $T_i$  y  $T_d$

8. ¿Qué se mide con los sensores de temperatura en TCLab para la identificación de sistemas de primer orden con retardo? \*

- ☐ La temperatura ambiente
- ☐ La temperatura interna de los transistores
- ☐ La temperatura de los disipadores de calor de los transistores

9. ¿Qué tipo de señal genera TCLab para regular la temperatura de sistemas de primer orden con retardo ? \*

- ☐ PWM
- ☐ 0-5V CC
- ☐ 4 a 20 mA

10. ¿Qué estrategias de control PID digital se puede aplicar TCLab para controlar sistemas de primer orden con retardo ? \*

- ☐ PID
- ☐ Cascada
- ☐ Lógica difusa
- ☐ Solo PID
- ☐ Todas las anteriores

Luego de evaluar las respuestas del pretest aplicado a los 25 estudiantes de la EPIE – UNPRG se obtuvieron los resultados de la Tabla 5.

**Tabla 5**

*Resultados del pretest*

Resultados del pretest	
Total de estudiantes evaluados	25
Estudiantes aprobados	13
Estudiantes desaprobados	12
Nota mínima	12
Nota máxima	14
Promedio de notas	11.12

12 estudiantes desaprobaron con nota igual a 10, 12 estudiantes aprobaron con nota igual a 12 y 1 estudiante aprobó con nota igual a 14.

### 3.4. Aplicación de la guía de prácticas de laboratorio

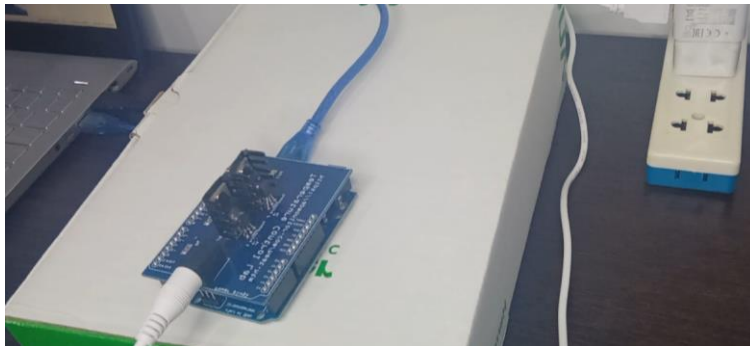
La guía de prácticas de laboratorio se dividió en 3 talleres y se aplicó en 3 días (3, 4 y 5 de febrero de 2023). Cada taller tuvo una duración de 2 horas aproximadamente y se realizó de forma virtual a través de la plataforma Google Meet.

El primer día se realizó la identificación de modelo matemático de la planta y se diseñó el algoritmo de control PID digital con el primer método de Ziegler y Nichols. El segundo día se diseñó el algoritmo de control PID digital con el método de Cohen y Coon. El tercer día se diseñó el algoritmo de control PID digital con el método de la Integral de Error.

En la sección de ANEXOS se encuentra la guía de prácticas de laboratorio con los fundamentos teóricos y prácticos empleados para desarrollar los ejercicios planteados.

**Figura 36**

*Sistema embebido TCLab utilizado en los talleres*



### 3.5. Aplicación y resultados del postest

Se utilizó el mismo cuestionario de 10 preguntas con respuestas de opción múltiple desarrollado en la plataforma Google Forms. El objetivo de aplicar el postest fue determinar si la aplicación de la guía de prácticas de laboratorio mejora la enseñanza de control digital para los estudiantes de la EPIE – UNPRG.

El cuestionario se envió por correo electrónico (institucional) y cada pregunta tuvo un puntaje de 2 puntos, lo que significa que la calificación más alta posible fue 20.

Luego de evaluar las respuestas del postest aplicado a los 25 estudiantes de la EPIE – UNPRG se obtuvieron los resultados de la Tabla 6.

**Tabla 6**

*Resultados del postest*

Resultados del postest	
Total de estudiantes evaluados	25
Estudiantes aprobados	25
Estudiantes desaprobados	0
Nota mínima	16
Nota máxima	20
Promedio de notas	18

23 estudiantes aprobaron con nota igual a 18, 1 estudiante aprobó con nota igual a 16 y 1 estudiante aprobó con nota igual a 20.

## **IV. DISCUSIÓN DE RESULTADOS**

En el artículo de Hoyo et al. (2021), se describe cómo la utilización de laboratorios portátiles y de bajo costo, como TCLab, ayuda a mejorar la enseñanza de los estudiantes de ingeniería de control en cursos de grado y postgrado. Por su parte, Moura Oliveira et al. (2020) llevaron a cabo una serie de experimentos utilizando el laboratorio de control de temperatura TCLab, y concluyeron que es una herramienta altamente efectiva para experimentar y mejorar el aprendizaje de los controladores digitales. Finalmente, Marques Aguiar (2020) comparó diferentes kits de laboratorio en términos de accesibilidad, relevancia didáctica y enfoque, y concluyó que TCLab permitió la implementación y prueba de técnicas de algoritmos de control digital.

Los resultados de esta investigación son similares a los antecedentes tomados como referencia en cuanto a la eficacia de la plataforma portátil TCLab en la enseñanza de las técnicas de control y la identificación de sistemas. Los autores Hoyo et al. (2021), Moura Oliveira et al. (2020) y Marques Aguiar (2020), destacan que la utilización de esta herramienta de portátil permite a los estudiantes aplicar estrategias de control en sistemas reales, familiarizarse con el ruido en las señales del sistema y las perturbaciones que pueden alterar su dinámica. Por lo tanto, con base en los resultados de la aplicación de la guía de prácticas de laboratorio a los 25 estudiantes de la EPIE – UNPRG, se puede afirmar que TCLab es una herramienta valiosa y efectiva en la enseñanza de técnicas de control y la identificación de sistemas en cursos de ingeniería. Además, la portabilidad y bajo costo de TCLab lo hacen accesible para una amplia gama de estudiantes y profesores, lo que puede contribuir a mejorar la calidad de la enseñanza en esta área.

## V. CONCLUSIONES

- Se logró identificar el modelo matemático del sistema de control de temperatura como un sistema de primer orden más retardo aplicando el método experimental de identificación por curva de reacción para la planta de la guía de prácticas de laboratorio.
- Se lograron diseñar tres controladores PID en tiempo discreto y se implementaron en la planta de la guía de prácticas de laboratorio.
- Se logró aplicar el pretest a los 25 estudiantes de la EPIE – UNPRG y se comprobó que su nivel de conocimiento respecto a la identificación de sistemas y diseño de algoritmos de control PID digital era regular (el promedio de notas fue 11.12).
- Se logró aplicar la guía de prácticas de laboratorio a los 25 estudiantes de la EPIE – UNPRG y se demostró que es posible controlar la temperatura del sistema de primer orden con retardo utilizando las estrategias de control digital: Ziegler y Nichols, Cohen y Coon e Integral de Error.
- Se logró aplicar el postest a los 25 estudiantes de la EPIE – UNPRG y se determinó que la guía de prácticas de laboratorio mejora la enseñanza de control digital para los estudiantes de la EPIE – UNPRG (el promedio de notas fue 18).
- Finalmente, se concluye que con una guía de prácticas de laboratorio basada en un sistema embebido portátil y de bajo costo, se mejora la enseñanza de control digital para los estudiantes de la EPIE - UNPRG.

## **VI. RECOMENDACIONES**

- Antes de realizar la guía de prácticas de laboratorio de control digital es necesario que los estudiantes adquieran los fundamentos teóricos necesarios para comprender el funcionamiento de los controladores diseñados.
- Integrar una interfaz gráfica basada en pantallas de operador (LCD, HMI, etc.) para visualizar las gráficas y valores de la temperatura, la referencia y el PWM.
- Integrar un sistema de ventilación para favorecer el descenso de la temperatura.
- Debido a que TCLAB es un sistema MIMO es posible agregar estrategias de control multivariable a la de la guía de prácticas de laboratorio de control digital.

## REFERENCIAS

- Barbarán Castañeda, M. D., y Chávarry Ysla, P. D. R. (2019). *Implementación del Algoritmo de Control Robótico PID Aplicado en la enseñanza de Robótica en los Niños*. INGENIERÍA: Ciencia, Tecnología e Innovación, 6(1).
- Bejarano Abarca, X. (2014). *Elaboración y Aplicación de una Guía de Prácticas de Laboratorio “Creando–Aprendo” para el aprendizaje de Ciencias Naturales en los estudiantes de séptimo año de Educación General Básica de la escuela “José María Román” de la ciudad de Riobamba. Período 2013* (Tesis de Grado, Universidad Nacional de Chimborazo).
- Bustos Pulluquitin, S. P. (2021). *Modelación matemática para un control robusto de la planta Festo MPS-PA Compact Workstation mediante la normativa IEC-61499* (Tesis de maestría, Universidad Técnica de Ambato. Facultad de Ingeniería en Sistemas, Electrónica e Industrial. Maestría en Matemática Aplicada).
- Caicedo Grajales, A. D. (2018). *Diseño e implementación de un sistema de control de temperatura* (Tesis de Grado, Universidad ECCI).
- Castaño, S. (2022, 19 enero). *Controladores PID Discreto*. Control Automático Educación. <https://controlautomaticoeducacion.com/control-realimentado/controladores-pid-discreto/>
- Chaljub Rojas, E. O., y Troncozo Arenales, B. D. (2022). *Sistema de monitoreo de consumo de energía (KWH) y control remoto a través de aplicaciones de mensajería*.
- Correa Torres, C. M., y Valencia Gómez, J. F. (2021). Desarrollo de una herramienta para la elaboración de interfaces gráficas para sistemas de monitoreo y control con la STM32F429I-DISC0 y. NET micro framework. *Ciencia e Ingeniería*, 5(2), e069-e069.

- Enríquez Aguilera, F. J. (2021). Plataforma de Programación Paralela en FPGA para Administrar Procesos de Enseñanza-Aprendizaje en Algoritmos de Visión por Computadora. *Departamento de Ingeniería Industrial y Manufactura*.
- Flores Avella, A., Gallegos Mendoza, G., y Hernández Sánchez, C. (2020). Modelado físico de un sistema de motores brushless basado en control digital PID para la enseñanza del control automático por medio de la programación en plataforma arduino (Physical modeling of a brushless motor system based on digital pid control for teaching automatic control through Arduino platform programming). *Pistas Educativas*, 42(137).
- Flores Jacome, E. V., y Sillagana Guano, A. V. (2022). *Implementación de un módulo didáctico multifuncional para la asignatura de electrónica de potencia* (Tesis de Grado, Ecuador: La Mana: Universidad Técnica de Cotopaxi (UTC)).
- Fonseca Yupa, J. D., y Soria Badillo, D. A. (2020). *Diseño e implementación de control domótico (con sistemas embebidos) para conectarse con aplicaciones adaptativas basados en IOT* (Tesis de grado).
- Fonseca, G. P. D. (2022). *Aprendizagem emocional em hardware: aplicação em controle térmico* (Tesis de grado, Universidad Federal do Rio Grande do Norte).
- Fortet Roura, P., García Aristegui, D., Illanes Moreno, A., Soto, J., y Santana Domínguez, A. (1996). *Sistema de ayuda a la enseñanza del control digital*.
- Giler Cevallos, K. K. (2021). *Sistemas electrónicos embebidos con microcontroladores PIC para mejorar la enseñanza de Electrónica Digital en la Carrera de Tecnologías de la Información de la Universidad Estatal del Sur de Manabí* (Tesis de Grado, Jipijapa. UNESUM).
- Grajeda De los Santos, G. A. (2021). *Metodología para el desarrollo de prácticas de programación de microcontroladores PIC*.



- Horna Martínez, L. M., y Díaz Cercado, C. K. (2019). *Diseño e Implementación de un Prototipo Automatizado para el Proceso de Conchado de Chocolate en la Empresa Industrias Sol Norteño—Jaén—Perú.*
- Hoyo, Á., García Mañas, F., Ramos Teodoro, J., Sánchez Molina, J. A., Guzmán, J. L., y Rodríguez, F. (2021). *Uso del paradigma Take-Home Labs para la enseñanza del control automático en estudios de ingeniería.* In *XLII Jornadas de Automática* (pp. 218-225). Universidade da Coruña, Servizo de Publicacións.
- Marques Aguiar, B. F. (2020). *Tools and Control Experiences using TCLab Arduino Kit.*
- Méndez de la Torre, R. E. (2022). *Diseño e Implementación de un Módulo de entrenamiento utilizando el procesador ESP32 para aplicaciones enfocadas a la Domótica* (Tesis de Grado).
- Moura Oliveira, P. B., Hedengren, J. D., y Rossiter, J. A. (2020). *Introducing Digital Controllers to Undergraduate Students using the TCLab Arduino Kit.* *IFAC-PapersOnLine*, 53(2), 17524-17529.
- Nguyen, N. T. A., Do, V. Q., Pham, T. T., y Tran Nguyen, T. (2022). *Application of different control algorithms on a 'home-made' temperature control lab kit.* *Can Tho University Journal of Science*, 14(1), 62-73. <https://doi.org/10.22144/ctu.jen.2022.007>
- Piñol Navarro, A. (2021). *Laboratori de control de temperatura per a aplicacions docents* (Bachelor's thesis, Universitat Politècnica de Catalunya).
- Real Garcés A. E., y Palacios Calderón A. W. (2021). *Estudio comparativo de control PID y control Fuzzy en un proceso de flujo* (Tesis de grado, Ecuador: Latacunga: Universidad Técnica de Cotopaxi: UTC.).
- Reyna Salas, R. R., y Puente Camasca, R. I. (2021). *Diseño del fitotron aplicando lógica difusa.*

- Ríos Restrepo, D., y Morales Higueta, M. A. (2022). Implementación de plataforma de hardware in the loop (HIL) aplicado a electrónica de potencia.
- Romero Valladares, C. J. (2021). *Diseño de circuitos con microcontroladores para las prácticas en el Laboratorio de Robótica de la Carrera de Ingeniería en Computación y Redes* (Tesis Grado, Jipijapa. UNESUM).
- Ruíz Pérez, J. R. (2019). *Diseño e implementación de un prototipo de modulo didáctico de caudal para medir su influencia en el aprendizaje de la instrumentación y control industrial*.
- Saravia Valverde, E. P. (2019). *Diseño de un módulo educativo para mejorar la enseñanza en el área de control y automatización de la Universidad de Ciencias y Humanidades*.
- Sotelo, D., Sotelo, C., Ramírez Mendoza, R. A., López Guajardo, E. A., Navarro Duran, D., Niño Juárez, E., y Vargas Martínez, A. (2022). *Lab-Tec@ Home: A Cost-Effective Kit for Online Control Engineering Education. Electronics, 11(6), 907*.
- Tosini, M., Leiva, L., Vázquez, M., Goñi, O., y Toloza, J. (2021). *Técnicas de optimización de soluciones en sistemas embebidos. In XXIII Workshop de Investigadores en Ciencias de la Computación (WICC 2021, Chilecito, La Rioja)*.
- Valencia Luna, L. J. (2021). *Sistemas Embebidos en el IoT*.

## ANEXOS

### ANEXO N.º01: CÓDIGO PID DIGITAL – ZIEGLER Y NICHOLS

```
// Definiciones de componentes de la tarjeta
#define sensorPin A0 // Entrada analógica para el sensor de temperatura TMP36
#define pwmPin 3 // Salida PWM para el transistor TIP31C
#define ledPin 9 // Salida digital para el LED

#include <TimerOne.h>
#include "tclab_lib.h"

/*=====
VARIABLES GLOBALES
=====
*/

float tempSensor = 0.0; // Variable para almacenar el
valor del sensor de temperatura
String setPoint = ""; // Variable para almacenar el
valor de la referencia desde el puerto serial
int PWM = 0; // Variable para almacenar el
valor del PWM
float w = 0; // Referencia para el control
PID discreto ZN
float e[3] = {0,0,0}; // Vector para almacenar
valores del error
float u[2] = {0,0}; // Vector para almacenar
valores de Ley de Control
int kU = sizeof(u)/sizeof(float)-1;
int kE = sizeof(e)/sizeof(float)-1;
float kp, ti, td, q0, q1, q2; // Parámetros del Controlador
PID discreto

// Parámetros de modelo matemático
float K = 0.6586, tau = 429.5, theta = 10; // Valores hallados en la
identificación por Curva de de Reacción
int Ts = 8; // Periodo de muestreo
float L = theta + Ts/2; // Modificación del retardo en
su equivalente discreto
```

```

/*=====
                                PERIODO DE MUESTREO - TIMER 1
=====
*/
void sampleTime(void)
{
    //Actualización los vectores u y e
    update_past(u,kU);
    update_past(e,kE);

    //Calcula el error actual
    e[kE] = w - tempSensor;
    // Calcula la Acción de Control PID
    u[kU] = PID_Controller(u, e, q0, q1, q2);    // Max = 100, Min =0
    PWM = u[kU];
    //Aplica la acción de control en el PWM
    analogWrite(pwmPin,map(PWM , 0,100, 0,255)); // Max = 100, Min =0
}

/*=====
                                FUNCIÓN SETUP
=====
*/
void setup() {

    // Configuración del LED y la referencia analógica
    pinMode(ledPin,OUTPUT);    //LED como salida
    digitalWrite(ledPin,LOW);   //LED inicia apagado
    analogReference(EXTERNAL);  //Referencia analógica PIN AREF (3.3V)

    // Configuración del puerto serial
    Serial.begin(9600);

    // El PWM inicia en cero
    analogWrite(pwmPin,0);

    // Cada 8 segundos (sampleTime) se ejecuta la interrupción
    Timer1.initialize(8000000);    //Configuración del TIMER en 8 Segundos
    Timer1.attachInterrupt(sampleTime) ; //Configuración de la interrupción del
Timer 1

    // ***** //
    // ***** SINTONIA DEL PID ZN ***** //
    // ***** //

    // Descomentar de acuerdo a la estrategia de control a aplicar. Por defecto
    PID.

```

```

/* P
kp = tau/(K*L);
ti = valor alto;
td = 0.0;
*/

/* PI
kp = (0.9*tau)/(K*L);
ti = L/3;
td = 0.0;
*/

// PID
kp = (1.2*tau)/(K*L);
ti = 2*L;
td = 0.5*L;

// ***** //
// **** CONTROLADOR PID DIGITAL **** //
// ***** //
q0 = kp*(1+Ts/(2*ti)+td/Ts);
q1 = -kp*(1-Ts/(2*ti)+(2*td)/Ts);
q2 = (kp*td)/Ts;
}

/*=====
                        FUNCIÓN PRINCIPAL
=====
*/
void loop() {

    //Lectura de temperatura
    tempSensor = tempRead(sensorPin);

    //Indicación Luminica
    if(tempSensor > 40.0)
        digitalWrite(ledPin, HIGH); //Transistor Caliente
    else
        digitalWrite(ledPin, LOW); //Transistor Frio

    // Recibir dato de referencia de temperatura por el puerto serial
    if(Serial.available() > 0){
        // Leer dato enviado al puerto serial y convertirlo a
        setPoint = Serial.readStringUntil('\n');
        w = setPoint.toFloat();
    }
}

```

```
Serial.print("Temperatura(°C): ");
Serial.print(tempSensor);
Serial.print(", ");
Serial.print("Referencia(°C): ");
Serial.print(w);
Serial.print(", ");
Serial.print("PWM(%): ");
Serial.println(PWM);
delay(1000);

}
```

## ANEXO N.º02: CÓDIGO PID DIGITAL – COHEN Y COON

```
// Definiciones componentes de la tarjeta
// Definiciones de componentes de la tarjeta
#define sensorPin A0 // Entrada analógica para el sensor de temperatura TMP36
#define pwmPin 3 // Salida PWM para el transistor TIP31C
#define ledPin 9 // Salida digital para el LED

#include <TimerOne.h>
#include "tclab_lib.h"

/*=====
                        VARIABLES GLOBALES
=====*/

float tempSensor = 0.0; // Variable para almacenar el
valor del sensor de temperatura
String setPoint = ""; // Variable para almacenar el
valor de la referencia desde el puerto serial
int PWM = 0; // Variable para almacenar el
valor del PWM
float w = 0; // Referencia para el control
PID discreto ZN
float e[3] = {0,0,0}; // Vector para almacenar valores
del error
float u[2] = {0,0}; // Vector para almacenar valores
de Ley de Control
int kU = sizeof(u)/sizeof(float)-1;
int kE = sizeof(e)/sizeof(float)-1;
float kp,ti,td,q0,q1,q2; // Parámetros del Controlador
PID discreto

// Parámetros de modelo matemático
float K = 0.6586, tau = 429.5, theta = 10; // Valores hallados en la
identificación por Curva de de Reacción
int Ts = 8; // Periodo de muestreo
float L = theta + Ts/2; // Modificación del retardo en
su equivalente discreto
```

```

/*=====
                                PERIODO DE MUESTREO - TIMER 1
=====
*/
void sampleTime(void)
{
    //Actualización los vectores u y e
    update_past(u,kU);
    update_past(e,kE);

    //Calcula el error actual
    e[kE] = w - tempSensor;
    // Calcula la Acción de Control PID
    u[kU] = PID_Controller(u, e, q0, q1, q2);    // Max = 100, Min = 0
    PWM = u[kU];
    //Aplica la acción de control en el PWM
    analogWrite(pwmPin,map(PWM , 0,100, 0,255)); // Max = 100, Min = 0
}

/*=====
                                FUNCIÓN SETUP
=====
*/
void setup() {

    // Configuración del LED y la referencia analógica
    pinMode(ledPin,OUTPUT);    //LED como salida
    digitalWrite(ledPin,LOW);  //LED inicia apagado
    analogReference(EXTERNAL); //Referencia analógica PIN AREF (3.3V)

    // Configuración del puerto serial
    Serial.begin(9600);

    // El PWM inicia en cero
    analogWrite(pwmPin,0);

    // Cada 8 segundos (sampleTime) se ejecuta la interrupción
    Timer1.initialize(8000000);    //Configuración del TIMER en 8 Segundos
    Timer1.attachInterrupt(sampleTime) ; //Configuración de la interrupción del
Timer 1

    // ***** //
    // ***** PID Cohen y Coon ***** //
    // ***** //

    // P
    //kp = (1.03+0.35*(L/tau))*(tau/(K*L));

```



```

//ti = 0.0; td = 0.0;

// PI
//kp = (0.9+0.083*(L/tau))*(tau/(K*L));
//ti = L*(0.9+0.083*(L/tau))/(1.27+0.6* (L/tau));
//td = 0;

// PID
kp = (1.35+0.25*(L/tau))*(tau/(K*L));
ti = L*(1.35+0.25*(L/tau))/(0.54+0.33*(L/tau));
td = 0.5*L/(1.35+0.25*(L/tau));

/// ***** //
// *** CONTROLADOR PID DIGITAL *** //
// ***** //
q0 = kp*(1+Ts/(2*ti)+td/Ts);
q1 = -kp*(1-Ts/(2*ti)+(2*td)/Ts);
q2 = (kp*td)/Ts;

}

/*=====
                        FUNCIÓN PRINCIPAL
=====*/

void loop() {
    //Lectura de temperatura
    tempSensor = tempRead(sensorPin);

    //Indicación Luminica
    if(tempSensor > 40.0)
        digitalWrite(ledPin, HIGH); //Transistor Caliente
    else
        digitalWrite(ledPin, LOW); //Transistor Frio

    // Recibir dato de referencia de temperatura por el puerto serial
    if(Serial.available() > 0){
        // Leer dato enviado al puerto serial y convertirlo a
        setPoint = Serial.readStringUntil('\n');
        w = setPoint.toFloat();
    }

    Serial.print("Temperatura(°C): ");
    Serial.print(tempSensor);
    Serial.print(", ");
    Serial.print("Referencia(°C): ");
    Serial.print(w);
    Serial.print(", ");

```

```
Serial.print("PWM(%): ");  
Serial.println(PWM);  
delay(1000);  
}
```

### ANEXO N.º03: CÓDIGO PID DIGITAL – INTEGRAL DEL ERROR

```
// Definiciones de componentes de la tarjeta
#define sensorPin A0 // Entrada analógica para el sensor de temperatura TMP36
#define pwmPin 3 // Salida PWM para el transistor TIP31C
#define ledPin 9 // Salida digital para el LED

#include <TimerOne.h>
#include "tclab_lib.h"

/*=====
VARIABLES GLOBALES
=====
*/

float tempSensor = 0.0; // Variable para almacenar el
valor del sensor de temperatura
String setPoint = ""; // Variable para almacenar el
valor de la referencia desde el puerto serial
int PWM = 0; // Variable para almacenar el
valor del PWM
float w = 0; // Referencia para el control
PID discreto ZN
float e[3] = {0,0,0}; // Vector para almacenar valores
del error
float u[2] = {0,0}; // Vector para almacenar valores
de Ley de Control
int kU = sizeof(u)/sizeof(float)-1;
int kE = sizeof(e)/sizeof(float)-1;
float kp,ti,td,q0,q1,q2; // Parametros del Controlador
PID discreto

// Parámetros de modelo matemático
float K = 0.6586, tau = 429.5, theta = 10; // Valores hallados en la
identificación por Curva de de Reacción
int Ts = 8; // Periodo de muestreo
float L = theta + Ts/2; // Modificación del retrado en su
equivalente discreto

/*=====
PERIODO DE MUESTREO - TIMER 1
=====
*/
void sampleTime(void)
{
//Actualización los vectores u y e
update_past(u,kU);
update_past(e,kE);
```

```

//Calcula el error actual
e[kE] = w - tempSensor;
// Calcula la Acción de Control PID
u[kU] = PID_Controller(u, e, q0, q1, q2);    // Max = 100, Min = 0
PWM = u[kU];
//Aplica la acción de control en el PWM
analogWrite(pwmPin,map(PWM , 0,100, 0,255)); // Max = 100, Min = 0
}

/*=====
                        FUNCIÓN SETUP
=====
*/
void setup() {

    // Configuración del LED y la referencia analógica
    pinMode(ledPin,OUTPUT);    //LED como salida
    digitalWrite(ledPin,LOW);  //LED inicia apagado
    analogReference(EXTERNAL); //Referencia analógica PIN AREF (3.3V)

    // Configuración del puerto serial
    Serial.begin(9600);

    // El PWM inicia en cero
    analogWrite(pwmPin,0);

    // Cada 8 segundos (sampleTime) se ejecuta la interrupción
    Timer1.initialize(8000000);    //Configuración del TIMER en 8 Segundos
    Timer1.attachInterrupt(sampleTime) ; //Configuración de la interrupción del
Timer 1

    // Parámetros del ITAE Rovira
    float iE[6]={0.965,-0.85,0.796,-0.147,0.308,0.929};

    //PID
    kp = (1/K)*(iE[0]*pow(L/tau, iE[1]));
    ti = tau/(iE[2]+iE[3]*(L/tau));
    td = tau*(iE[4]*pow(L/tau, iE[5]));

    /// ***** //
    // *** CONTROLADOR PID DIGITAL **** //
    // ***** //
    q0 = kp*(1+Ts/(2*ti)+td/Ts);
    q1 = -kp*(1-Ts/(2*ti)+(2*td)/Ts);
    q2 = (kp*td)/Ts;
}

```

```

/*=====
                                FUNCIÓN PRINCIPAL
=====*/
*/
void loop() {
    //Lectura de temperatura
    tempSensor = tempRead(sensorPin);

    //Indicación Luminica
    if(tempSensor > 40.0)
        digitalWrite(ledPin, HIGH); //Transistor Caliente
    else
        digitalWrite(ledPin, LOW); //Transistor Frio

    // Recibir dato de referencia de temperatura por el puerto serial
    if(Serial.available() > 0){
        // Leer dato enviado al puerto serial y convertirlo a
        setPoint = Serial.readStringUntil('\n');
        w = setPoint.toFloat();
    }

    Serial.print("Temperatura(°C): ");
    Serial.print(tempSensor);
    Serial.print(", ");
    Serial.print("Referencia(°C): ");
    Serial.print(w);
    Serial.print(", ");
    Serial.print("PWM(%): ");
    Serial.println(PWM);
    delay(1000);
}

```

## ANEXO N.º04: CÓDIGO PID DIGITAL – FUNCIONES TCLAB

```
#include "tclab_lib.h"

float tempRead(int sen)
{
    float S1,aux;
    int i;
    //Filtro de promedio móvil en la lectura ADC
    aux=0;
    for(i=0;i<10;i++){
        aux = aux + (float(analogRead(sen))*3.3/1023.0-0.5)/0.01; //TMP36
    }
    S1 = aux/10.0;
    return(S1);
}

/*=====*/
/
/*===== FUNCION DE ACTUALIZACIÓN =====*/
/*=====*/
/
// Esta función desplaza todo el vector una unidad hacia atrás para actualizar
// los valores pasados en las ecuaciones causales de los controladores
// float v: El vector que deseamos desplazar (Actualizar)
// int kT: Es la última posición del vector v
void update_past(float v[],int kT){
    int i;
    for(i=1;i<=kT;i++){
        v[i-1]=v[i];
    }
}

/*=====*/
/
/*===== FUNCION DEL CONTROL PID =====*/
/*=====*/
/
float PID_Controller(float u[], float e[3], float q0, float q1, float q2)
{
    float lu;
    // Control PID
    // e[2] = e(k)
    // e[1] = e(k-1)
    // e[0] = e(k-2)
    // u[0] = u(k-1)
    lu = u[0] + q0*e[2] + q1*e[1] + q2*e[0]; //Ley del controlador PID
    discreto
}
```

```
// Anti - Windup
if (lu >= 100.0)
    lu = 100.0;

if (lu <= 0.0)
    lu = 0.0;

return(lu);
}
```

## ANEXO N.º05: CÓDIGO PID DIGITAL – ENCABEZADO TCLAB

```
#include<Arduino.h>
// Encabezados de la librería tclab_lib

float tempRead(int sen);

void update_past(float v[],int kT);

float PID_Controller(float u[], float e[3], float q0, float q1, float q2);
```



## ANEXO N.º06: TCLAB

# Laboratorio de control de temperatura

El laboratorio de control de temperatura es una aplicación de control de retroalimentación con un Arduino, un LED, dos calentadores y dos sensores de temperatura. La potencia de salida del calentador se ajusta para mantener un punto de ajuste de temperatura deseado. La energía térmica del calentador se transfiere por conducción, convección y radiación al sensor de temperatura. El calor también se transfiere del dispositivo al entorno.



Este laboratorio es un recurso para la identificación de modelos y el desarrollo de controladores. Es un laboratorio de bolsillo con software en Python, MATLAB y Simulink con el propósito de reforzar la teoría de control para los estudiantes. Los módulos de aprendizaje incluyen:

- **Comience Python con TCLab**
- **Aprenda Python científico para ingenieros**
- **Control y Dinámica de Procesos con TCLab (este curso)**
- **Aprendizaje automático y optimización dinámica con TCLab**

## ANEXO N.º07: Guía de prácticas de laboratorio de control digital

Curso	
Docente	
Estudiante	
Título del laboratorio	Control PID en tiempo discreto

### Objetivo general

Diseñar tres controladores PID en tiempo discreto para regular un sistema de control de temperatura (TCLAB).

### Objetivos específicos

- Determinar el modelo matemático del sistema de control de temperatura (TCLAB).
- Diseñar un algoritmo de control PID digital basado en Ziegler y Nichols.
- Diseñar un algoritmo de control PID digital basado en Cohen y Coon.
- Diseñar un algoritmo de control PID digital basado en la Integral de Error.

### Fundamentos teóricos

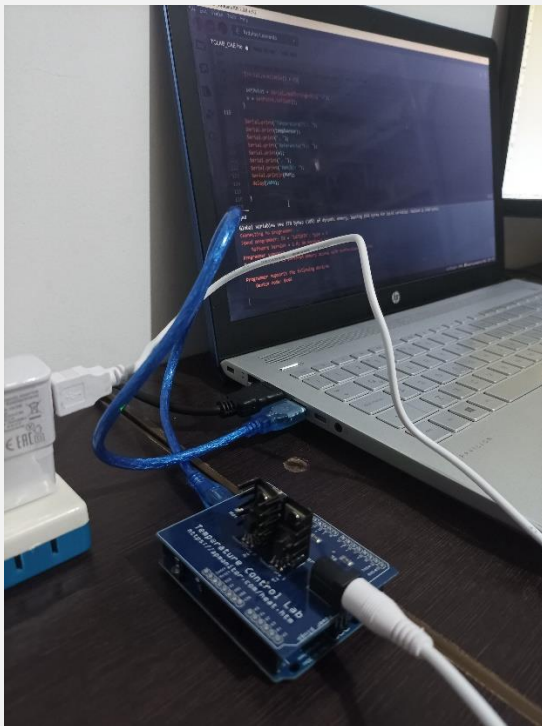
Fundamentos teóricos del laboratorio en Google Drive:

[https://docs.google.com/presentation/d/1BqS1PFTgOV7ludkgfMHtDRAbbVmd\\_r\\_u/edit#slide=id.p1](https://docs.google.com/presentation/d/1BqS1PFTgOV7ludkgfMHtDRAbbVmd_r_u/edit#slide=id.p1)

### Fundamentos prácticos

- Determinar el modelo matemático de la planta

#### 1. Conecte TCLAB al computador y ejecute el código Interfaz Gráfica.



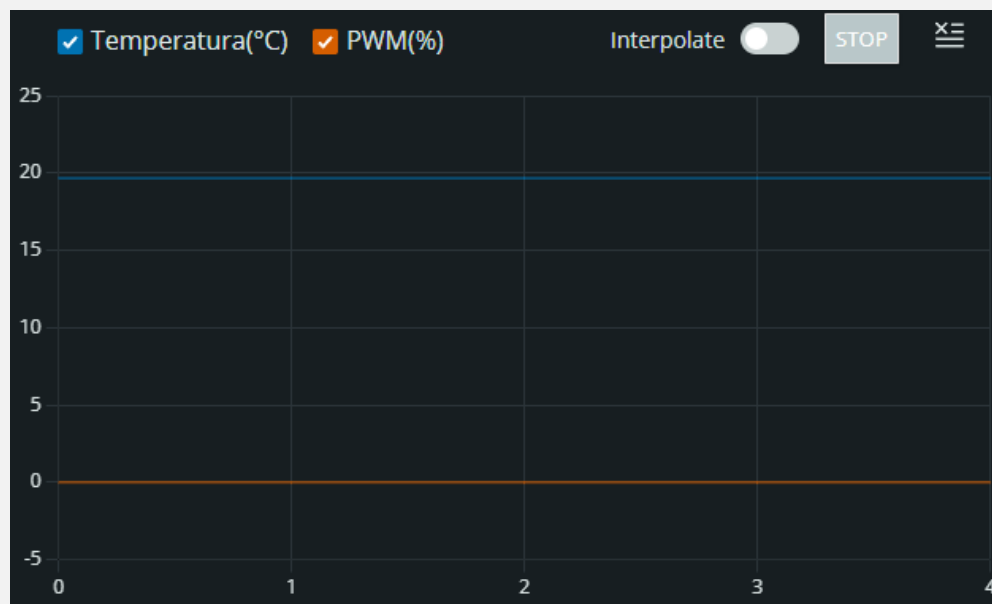
## 2. Abra el Serial Monitor

Espere a que la temperatura se estabilice. El valor de temperatura puede variar dependiendo del lugar y el horario en que se realice la prueba.

```
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
Temperatura(°C): 19.68, PWM(%) : 0
```

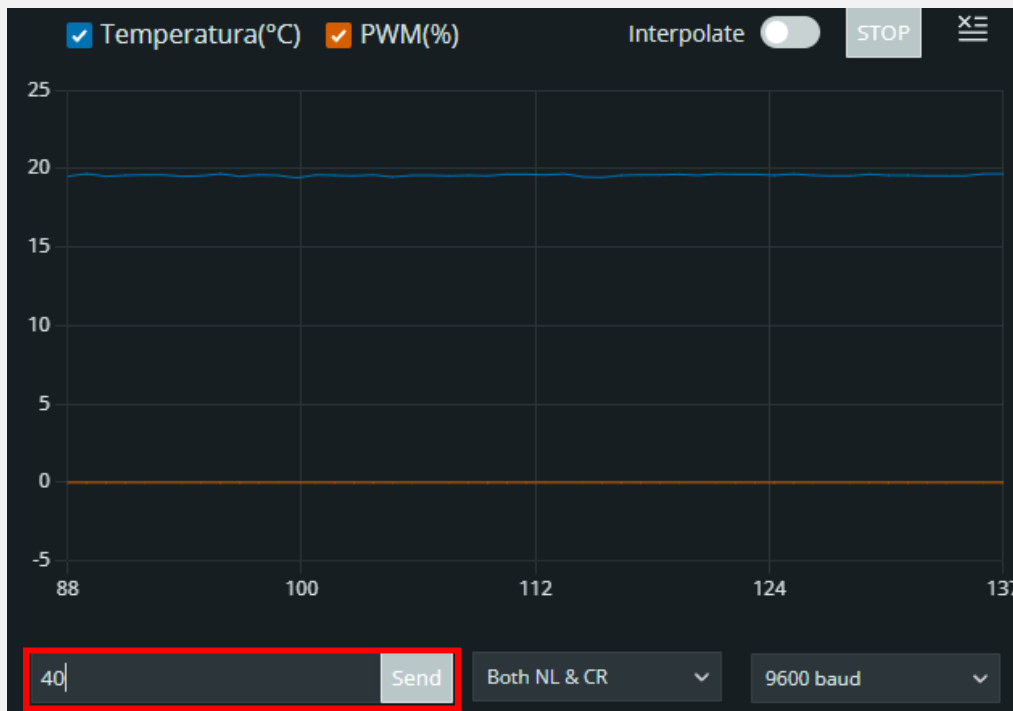
## 3. Abra el Serial Plotter

Gráfica de curvas para el monitoreo del valor de temperatura y PWM



#### 4. Aplique una señal escalón entre 0% a 100% de PWM.

En este ejemplo se aplicó el 40% de PWM.



#### 5. Esperar hasta que el transistor se caliente y el valor de la temperatura se estabilice.

En este ejemplo la temperatura se estabilizó en 45.16°C en un periodo de 1800 segundos (30 minutos).

```
Temperatura (°C) : 45.10, PWM (%) : 40
Temperatura (°C) : 45.13, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.13, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
Temperatura (°C) : 45.16, PWM (%) : 40
```

6. Registrar los valores de temperatura inicial y final, así como los valores de PWM inicial y final. Luego aplicar la siguiente ecuación para determinar la ganancia estática del sistema.

$$K = \frac{T_{Final} - T_{Inicial}}{U_{Final} - U_{Inicial}} = \frac{45.16^{\circ}\text{C} - 20.32^{\circ}\text{C}}{40\% - 0\%} = 0.621$$

7. Calcular la constante de tiempo dividiendo el tiempo de estabilización del paso 5 entre 4.

$$\tau = \frac{T_{ss}}{4} = \frac{1800}{4} = 450$$

8. Verificar en el Serial Monitor el tiempo transcurrido desde que se aplicó el PWM hasta que la temperatura tuvo un cambio significativo.

```

Temperatura (°C) : 19.68, PWM (%) : 40
Temperatura (°C) : 19.68, PWM (%) : 40
Temperatura (°C) : 19.68, PWM (%) : 40
Temperatura (°C) : 19.68, PWM (%) : 40
Temperatura (°C) : 19.68, PWM (%) : 40
Temperatura (°C) : 19.71, PWM (%) : 40
Temperatura (°C) : 19.87, PWM (%) : 40
Temperatura (°C) : 20.00, PWM (%) : 40
Temperatura (°C) : 20.00, PWM (%) : 40
Temperatura (°C) : 20.00, PWM (%) : 40

```

Para este ejemplo el retardo de tiempo fue:

$$\theta = 10 \text{ s}$$

9. Aplique el método de la curva de reacción para modelar el sistema mediante la siguiente ecuación:

$$G(S) = \frac{Ke^{-\theta s}}{\tau s + 1} = \frac{0.621e^{-10s}}{450s + 1}$$

Dónde: K es la ganancia estática,  $\theta$  es retardo de tiempo y  $\tau$  es la constante de tiempo del sistema de primer orden más retardo.

Nota: El modelo puede variar, aunque se trate de la misma planta. También pueden ser similares. Esto depende del lugar y el horario en que se realice la prueba.

- Aplicar las diferentes variantes de control PID digital

### 1. Control PID discreto por Ziegler y Nichols

#### a. Conecte TCLAB al computador y ejecute el código Ziegler y Nichols

Ingrese los valores del modelo matemático

```
// Parámetros de modelo matemático  
float K = 0.621, tau = 450, theta = 10;
```

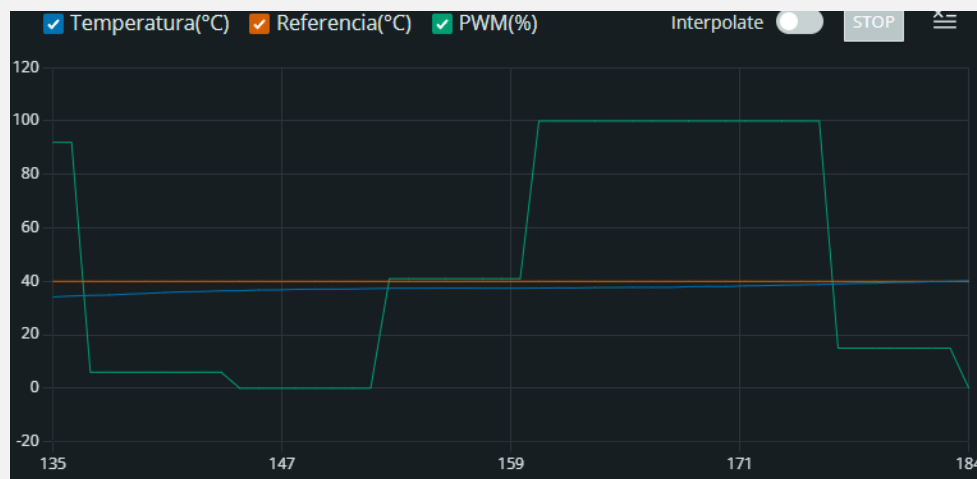
#### b. Abra el Serial Plotter e ingrese una referencia en el siguiente rango: $T_i$ - 150°C. Donde $T_i$ = Temperatura > temperatura ambiente.

Para este ejemplo se ingresó una referencia de 40°C.



#### c. Abra el Serial Monitor para visualizar el valor de temperatura, de la referencia y del PWM.

Para este ejemplo se obtuvieron los siguientes resultados.



```
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 36  
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 36  
Temperatura(°C): 40.29, Referencia(°C): 40.00, PWM(%): 36  
Temperatura(°C): 40.29, Referencia(°C): 40.00, PWM(%): 36  
Temperatura(°C): 40.29, Referencia(°C): 40.00, PWM(%): 36  
Temperatura(°C): 40.26, Referencia(°C): 40.00, PWM(%): 33  
Temperatura(°C): 40.26, Referencia(°C): 40.00, PWM(%): 33  
Temperatura(°C): 40.26, Referencia(°C): 40.00, PWM(%): 33  
Temperatura(°C): 40.23, Referencia(°C): 40.00, PWM(%): 33
```

## 2. Control PID discreto por Cohen y Coon

### a. Conecte TCLAB al computador y ejecute el código Cohen y Coon

Ingrese los valores del modelo matemático

```
// Parámetros de modelo matemático  
float K = 0.621, tau = 450, theta = 10;
```

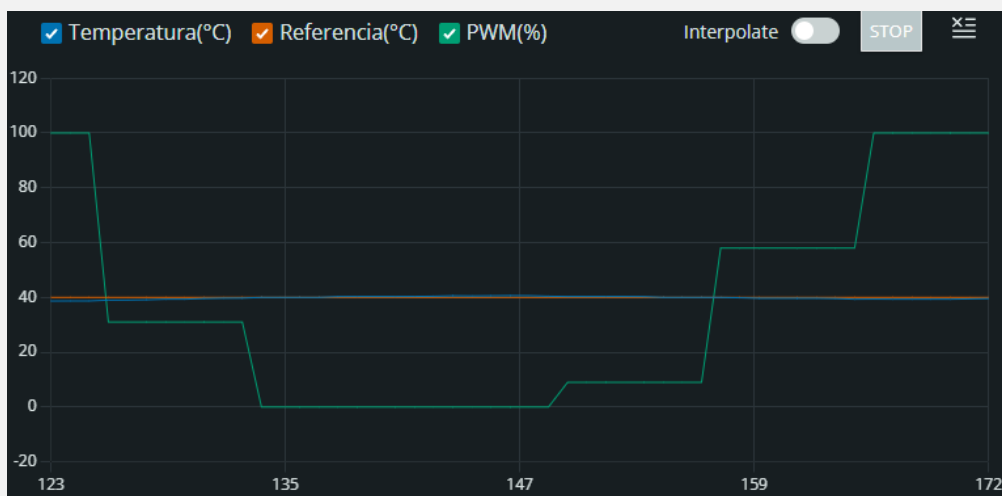
### b. Abra el Serial Plotter e ingrese una referencia en el siguiente rango: $T_i - 150^{\circ}\text{C}$ . Donde $T_i$ = Temperatura > temperatura ambiente.

Para este ejemplo se ingresó una referencia de  $40^{\circ}\text{C}$ .



### c. Abra el Serial Monitor para visualizar el valor de temperatura, de la referencia y del PWM.

Para este ejemplo se obtuvieron los siguientes resultados.





```
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 9  
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 9  
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 9  
Temperatura(°C): 40.29, Referencia(°C): 40.00, PWM(%): 9  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 9  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 9  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 9  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 58
```

### 3. Control PID discreto por Integral de Error

#### a. Conecte TCLAB al computador y ejecute el código Integral del Error

Ingrese los valores del modelo matemático

```
// Parámetros de modelo matemático  
float K = 0.621, tau = 450, theta = 10;
```

#### b. Abra el Serial Plotter e ingrese una referencia en el siguiente rango: $T_i - 150^{\circ}\text{C}$ . Donde $T_i$ = Temperatura > temperatura ambiente.

Para este ejemplo se ingresó una referencia de  $40^{\circ}\text{C}$ .



#### c. Abra el Serial Monitor para visualizar el valor de temperatura, de la referencia y del PWM.

Para este ejemplo se obtuvieron los siguientes resultados.



```
Temperatura(°C): 40.32, Referencia(°C): 40.00, PWM(%): 18  
Temperatura(°C): 40.26, Referencia(°C): 40.00, PWM(%): 18  
Temperatura(°C): 40.26, Referencia(°C): 40.00, PWM(%): 18  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 18  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 18  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 63  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 63  
Temperatura(°C): 40.00, Referencia(°C): 40.00, PWM(%): 63
```

**ANEXO N.º08: Estudiantes de la EPIE – UNPRG que participaron en los talleres de control digital**

<b>ALUMNO</b>	<b>CORREO</b>
Luis Enrique Manay Bardales	lmanayb@unprg.edu.pe
Carlos Oblitas Vásquez	coblitasv@unprg.edu.pe
Cesareo Wilfredo Sánchez Salazar	csanchezs@unprg.edu.pe
Luis Miguel Fenco Llanos	lfencol@unprg.edu.pe
Leo Danny Camizan Cabrera	lcamizan@unprg.edu.pe
Romers Jesús Damián Suclupe	rdamians@unprg.edu.pe
Ethel Sabrina Mejía Vilcabana	emejiavi@unprg.edu.pe
Angel Alfredo Jaramillo Ramírez	ajaramillor@unprg.edu.pe
Julián Saúl Dávila Reque	jdavilare@unprg.edu.pe
Miguel Mauricio Aguilar Verona	maguilarv@unprg.edu.pe
Guillermo Abel Bances Mehan	gbancesm@unprg.edu.pe
Luis Miguel Chaponán Baldera	lchaponanb@unprg.edu.pe
Gonzalo Arroyo Bernilla	garroyo@unprg.edu.pe
Edin Gean Carlos Vásquez Sempertegui	evasquezse@unprg.edu.pe
Brayan Yeison Vásquez Silva	bvasquezs@unprg.edu.pe
Cristopher Anthony Pisfil Mego	cpisfilm@unprg.edu.pe
Sheyla Naomi Krisia Quiñones Céspedes	squiñones@unprg.edu.pe
Carmen Waldir Julón Delgado	cjulonde@unprg.edu.pe
Carlos Alejandro Ayma Velásquez	cayma@unprg.edu.pe
Karina Trigos Mixan	ktrigoso@unprg.edu.pe
Jhackson Díaz Rojas	jdiazroja@unprg.edu.pe
Alexander Rojas Burga	arojasb@unprg.edu.pe
Arturo David Flores Giles	arturo.flores@unprg.edu.pe
Cristian Fernando Vásquez Vargas	cvasquezva@unprg.edu.pe
Jairo Rubén Cabrera Guevara	jcabrera@unprg.edu.pe

# GUÍA DE PRÁCTICAS DE LABORATORIO CON SISTEMAS EMBEBIDOS PARA LA ENSEÑANZA DE CONTROL DIGITAL EN LOS ESTUDIANTES DE LA EPIE – UNPRG

## INFORME DE ORIGINALIDAD

18%

INDICE DE SIMILITUD

18%

FUENTES DE INTERNET

3%

PUBLICACIONES

5%

TRABAJOS DEL  
ESTUDIANTE

## FUENTES PRIMARIAS

1

[controlautomaticoeducacion.com](http://controlautomaticoeducacion.com)

Fuente de Internet

3%

2

[hdl.handle.net](http://hdl.handle.net)

Fuente de Internet

2%

3

[www.dspace.uce.edu.ec](http://www.dspace.uce.edu.ec)

Fuente de Internet

1%

4

[repositorio.unprg.edu.pe](http://repositorio.unprg.edu.pe)

Fuente de Internet

1%

5

[dspace.unach.edu.ec](http://dspace.unach.edu.ec)

Fuente de Internet

1%

6

[alicia.concytec.gob.pe](http://alicia.concytec.gob.pe)

Fuente de Internet

1%

7

[qdoc.tips](http://qdoc.tips)

Fuente de Internet

1%

8

[repositorio.uta.edu.ec](http://repositorio.uta.edu.ec)

Fuente de Internet

1%

Ing. Oscar Uccelly Romero Cortez

DNI: 41812294

Asesor



## Recibo digital

Este recibo confirma que su trabajo ha sido recibido por Turnitin. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega: Segundo Raul Amaya Huayama  
Título del ejercicio: GUÍA DE PRÁCTICAS DE LABORATORIO CON SISTEMAS EMBE...  
Título de la entrega: GUÍA DE PRÁCTICAS DE LABORATORIO CON SISTEMAS EMBE...  
Nombre del archivo: PROYECTO\_FINAL\_AMAYA\_OLAZABAL.docx  
Tamaño del archivo: 2.89M  
Total páginas: 93  
Total de palabras: 13,158  
Total de caracteres: 73,599  
Fecha de entrega: 23-feb.-2023 07:50a. m. (UTC-0500)  
Identificador de la entrega... 2021200530

UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
ESCUELA PROFESIONAL DE INGENIERÍA  
ELECTRÓNICA



TESIS  
"GUÍA DE PRÁCTICAS DE LABORATORIO CON SISTEMAS  
EMBEBIDOS PARA LA ENSEÑANZA DE CONTROL  
DIGITAL EN LOS ESTUDIANTES DE LA EPIE – UNPRG"

PARA OBTENER EL TÍTULO PROFESIONAL DE  
INGENIERO ELECTRÓNICO

**ELABORADA POR:**

Bach. Amaya Huayama Segundo Raul  
Bach. Olazabal Urbina Daniel Andres

**ASESOR:**

Mg. Ing. Romero Cortez Oscar Uccelly  
ORCID:

LAMBAYEQUE – PERÚ  
2023

Ing. Oscar Uccelly Romero Cortez

DNI: 41812294  
Asesor