



**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE GUAYAQUIL**  
**CARRERA DE INGENIERÍA ELECTRÓNICA**

**“DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO BASADO  
EN REDES NEURONALES APLICADO A UN BANCO  
DIDÁCTICO PARA EL CONTROL DE TEMPERATURA EN  
LABVIEW NXG”**

Trabajo de titulación previo a la obtención del  
Título de Ingeniero Electrónico

**AUTORES: JULIA LISSETH MOREIRA MANZABA**  
**STEVEN ENOC RODRÍGUEZ GÓMEZ**

**TUTOR: ING. TEDDY JHENNSE NEGRETE PEÑA MSC.**

**GUAYAQUIL – ECUADOR**

**2022**

## CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA

Nosotros, Julia Lisseth Moreira Manzaba con documento de identificación N° 0951357250 y Steven Enoc Rodríguez Gómez con documento de identificación N° 0926633090; manifestamos que:

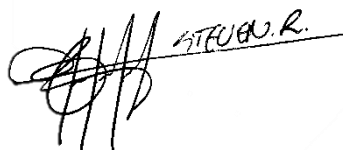
Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Guayaquil, 20 de septiembre del año 2022.

Atentamente,



-----  
Julia Lisseth Moreira Manzaba  
0951357250



-----  
Steven Enoc Rodríguez Gómez  
0926633090

## CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR

Nosotros, , **Julia Lisseth Moreira Manzaba**, con documento de identificación N° 0951357250 y **Steven Enoc Rodríguez Gómez**, con documento de identificación N° 0926633090, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de grado titulado: “**DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO BASADO EN REDES NEURONALES APLICADO A UN BANCO DIDÁCTICO PARA EL CONTROL DE TEMPERATURA EN LABVIEW NXG**”, el cual ha sido desarrollado para optar por el título de: **INGENIERO ELECTRÓNICO**, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

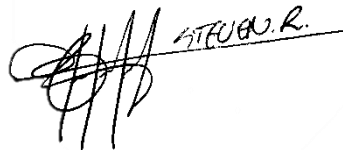
Guayaquil, 20 de septiembre del año 2022.

Atentamente,



-----  
Julia Lisseth Moreira Manzaba

0951357250



-----  
Steven Enoc Rodríguez Gómez

0926633090

## CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN

Yo, **Teddy Jhennse Negrete Peña MSc.** con documento de identificación N° , docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: “**DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO BASADO EN REDES NEURONALES APLICADO A UN BANCO DIDÁCTICO PARA EL CONTROL DE TEMPERATURA EN LABVIEW NXG**”, realizado por **Julia Lisseth Moreira Manzaba** con documento de identificación N° 0951357250 y por **Steven Enoc Rodríguez Gómez** con documento de identificación N° 0926633090, obteniendo como resultado final el trabajo de titulación bajo la opción de trabajo de grado que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 20 de septiembre del año 2022.

Atentamente,



-----  
**Ing. Teddy Jhennse Negrete Peña MSc.**

0912419611

## DEDICATORIA

Mi título universitario se lo dedico a Dios que sé que sin Él hubiera sido imposible lograrlo, ya que Él proveyó de los medios para que mis padres puedan ayudarme con mis estudios.

Después de Dios dedico mi título a mis padres, quienes con paciencia y sabiduría supieron animarme día a día a seguir adelante hasta obtener mi tan deseado título de Ingeniera Electrónica.

No puedo expresar el orgullo que siento al saber que un eslabón más ha sido superado y que puedo celebrar con mis padres y mis seres queridos que han estado pendiente de mi progreso en mi vida estudiantil. Es por esta razón que dedico mi esfuerzo y mis logros a Dios y a mis padres.

Gracias Yucipina Manzaba por tus palabras de aliento y de ánimo, por demostrarme con tu ejemplo lo que es tener un título universitario.

|Gracias Vicente Moreira por enseñarme que en la vida todo trae su recompensa si eres dedicado y te esfuerzas en lo que haces.

Para ustedes mis amados padres mis victorias y alegrías, sin ustedes nada de esto se hubiera hecho realidad.

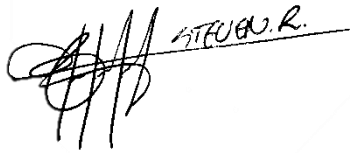
*Julia Moreira Mo.*

-----

**Julia Lisseth Moreira Manzaba**

## DEDICATORIA

Este proyecto se lo dedico en especial a todas esas personas que creyeron en mí y mi familia entera por ser el apoyo que necesité durante toda esta etapa de mi vida, en especial a mi madre, padre y abuela que fueron siempre los que estuvieron a mi lado en todo momento ayudándome a continuar en este duro camino, a mis amigos y profesores que también fueron guías en esta etapa.

A handwritten signature in black ink, consisting of a stylized, cursive script. To the right of the main signature, the initials "S.E.R." are written in a simpler, more legible font. The signature is positioned above a horizontal dashed line.

**Steven Enoc Rodríguez Gómez**

## AGRADECIMIENTO

La culminación del ciclo de los estudios universitarios atrae una amalgama de sentimiento, en donde me embarga la tristeza por ya no estar junto a nuestros amigos y profesores que me han ayudado en esta etapa tan importante, me llena de alegría el saber que obtuve mi título universitario y hasta me llena de incertidumbre pues sé que me voy a enfrentar al campo laboral para ejercer mi carrera.

A pesar de todos los sentimientos que me embarga el obtener mi título universitario es necesario agradecer a:

Agradezco a mis padres Yucipina Yamilet Manzaba Carranza y Vicente Aníbal Moreira Tortorelly por su apoyo para que este sueño se haga realidad, les dedico todo mi esfuerzo, en reconocimiento a todo el sacrificio puesto para que yo pueda estudiar, se merecen esto y mucho más.

Agradezco a mis hermanos Kevin Aníbal Moreira Manzaba y Jerico Damián Moreira Manzaba que me han brindado su apoyo, amor y confiando que lograré esta etapa estudiantil.

A mi novio Jorge Olaya, a quien amo tanto y agradezco por tenerme tanta paciencia, estar a mi lado en todo momento y por darme su amor todos los días, lo que me motiva a cumplir todo lo que me proponga.

Agradezco a cada uno de los docentes que han compartido sus conocimientos y sabiduría conmigo y con mis compañeros.

Agradezco a la prestigiosa institución que acoge en sus aulas estudiantes que anhelan conocimientos y superación.

Para mí es un honor salir siendo una ingeniera de la Universidad y espero ser una excelente ingeniera que enaltezca el nombre de mi querida universidad.

*Julia Moreira Mo.*

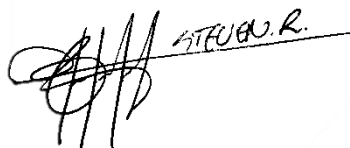
-----  
**Julia Lisseth Moreira Manzaba**

## AGRADECIMIENTO

Agradezco a Dios por ser el que guió el camino de mi vida, me dio la fuerza que me hizo falta cada vez que me quería dar por vencido, fue la luz de mi sendero para cada paso que he dado en el transcurso de mi vida, a su lado lo he podido todo y sé que siempre caminarás junto a mi sin importar a donde me dirija.

A mis padres, Mónica Gómez y José Rodríguez por ser las personas que me dieron la vida y estuvieron ahí en cada paso de mi vida, y la oportunidad de avanzar con mis estudios, a mi tía Mariana Jiménez, la cual fue un pilar en todo este proceso, por su apoyo incondicional en cada etapa de mi vida, a mi abuela, Zulema Jiménez por la crianza que me dio y los valores que me inculco.

A mis hermanos, Stalyn Rodríguez, José Rodríguez y Marilyn Rodríguez por ser los pilares fundamentales en el transcurso de esta dura etapa, apoyándome en cada momento de mi vida.

A handwritten signature in black ink, consisting of stylized, overlapping loops and lines. To the right of the signature, the initials "S.E.R." are written in a simple, straight font.

-----  
**Steven Enoc Rodríguez Gómez**



## RESUMEN

| AÑO  | ALUMNOS   | DIRECTOR DE PROYECTO TÉCNICO    | TEMA DE PROYECTO TÉCNICO  |
|------|---|---------------------------------|---|
| 2020 | <ul style="list-style-type: none"><li>Julia Lisseth Moreira Manzaba</li><li>Steven Enoc Rodríguez Gómez</li></ul> | Ing. Teddy Jhennse Negrete Peña | Diseño e Implementación de un algoritmo basado en redes neuronales aplicado a un banco didáctico para el control de temperatura en LabVIEW NXG. |

El presente trabajo está en el desarrollo y diseño de un banco de pruebas para el control de temperatura mediante la creación de algoritmos que basados en lógica neuronal teniendo como generador de calor un foco dicróico y una lectura de temperatura utilizando un sensor Im35 que ayuden a los estudiantes de Ingeniería Electrónica y Automatización puedan ejecutar prácticas en el programa de instrumentación virtual LabVIEW NXG teniendo como tarjeta de adquisición de datos una tarjeta NI Daq 6009, permitiendo desarrollar destrezas y experiencias que puedan aplicar en su vida profesional.

Este trabajo aporta a los estudiantes una herramienta de aprendizaje práctico ya que permite la simulación de procesos industriales a través de la planta que facilita las actividades didácticas mediante una guía para los estudiantes.

En el presente documento en la parte de marco metodológico incluye los elementos que se tomaron en cuenta para la creación del proyecto, como instrumentación electrónica, diseño electrónico y programación en LabVIEW.

Para el estudio y la eficacia del controlador basado en lógica neuronal se estableció realizar una comparación entre un control en lazo cerrado PID, tomando en cuenta los datos que se muestran en el apartado de resultados

como sus graficas de temperaturas de la planta en ambos controladores trabajando de manera independiente en el banco de pruebas.

A través del programa Matlab se obtendrán los valores del entrenamiento de la red neuronal tomando en cuenta sus pesos y vías establecidos mediante la integración de la herramienta Neuronal Network Training, mediante los resultados del entrenamiento se crearán sub vi en LabVIEW NXG donde se podrán realizar tareas de administración y algoritmos como los algoritmos inteligentes basado en lógica neuronal.

El proyecto consta con una interfaz desarrollada en LabVIEW NXG que permite la visualización de los valores en tiempo real de los datos de temperatura y la salida de los controladores o panel frontal donde se visualiza las prácticas y se realizan las pruebas.

**PALABRAS CLAVES:** LabVIEW, LabVIEW NXG, DAQ, NEURONAL, CONTROL INTELIGENTE, TEMPERATURA.

## ABSTRACT

| YEAR | STUDENTS  | DIRECTOR OF TECHNICAL PROJECT   | TECHNICAL PROJECT THEME  |
|------|---|---------------------------------|--|
| 2020 | <ul style="list-style-type: none"><li>• Julia Lisseth Moreira Manzaba</li><li>• Steven Enoc Rodríguez Gómez</li></ul> | Ing. Teddy Jhennse Negrete Peña | Design and Implementation of an algorithm based on neural networks applied to a didactic bench for temperature control in LabVIEW NXG. |

The present work is in the development and design of a test bench for temperature control through the creation of algorithms based on neural logic having a dichroic focus as a heat generator and a temperature reading using an Im35 sensor that help the Electronic Engineering and Automation students can carry out practices in the virtual instrumentation program LabVIEW NXG having a NI Daq 6009 card as a data acquisition card, allowing them to develop skills and experiences that they can apply in their professional life.

This work provides students with a practical learning tool since it allows the simulation of industrial processes through the plant that facilitates educational activities through a guide for students.

In this document, in the methodological framework part, it includes the elements that were considered for the creation of the project, such as electronic instrumentation, electronic design and programming in LabVIEW.

For the study and the efficiency of the controller based on neural logic, it was established to make a comparison between a PID closed-loop control, considering the data shown in the results section as its temperature graphs of the plant in both controllers working independently on the test bench.

Through the MATLAB program, the training values of the neural network will be obtained considering their weights and established pathways through the integration of the Neural Network Training tool, through the results of the training sub vi will be created in LabVIEW NXG where tasks can be performed management and algorithms such as intelligent algorithms based on neural logic.

The project consists of an interface developed in LabVIEW NXG that allows the visualization of the real-time values of the temperature data and the output of the controllers or front panel where the practices are visualized, and the tests are conducted.

KEY WORDS: LabVIEW, LabVIEW NXG, DAQ, NEURONAL, SMART CONTROL, TEMPERATURE.

## ÍNDICE GENERAL

|   |      |
|---|------|
| CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA.....           | II   |
| CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR .....        | III  |
| CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN ..... | IV   |
| DEDICATORIA .....                                       | V    |
| DEDICATORIA .....                                       | VI   |
| AGRADECIMIENTO .....                                    | VII  |
| AGRADECIMIENTO .....                                    | VIII |
| RESUMEN .....   | IX   |
| ABSTRACT.....   | XI   |
| ÍNDICE GENERAL.....                                     | XIII |
| ÍNDICE DE FIGURAS.....                                  | XVI  |
| ÍNDICE DE TABLAS .....                                  | XIX  |
| INTRODUCCIÓN .....                                      | 1    |
| 1. EL PROBLEMA .....                                    | 2    |
| 1.1. Antecedentes .....                                 | 2    |
| 1.2. Importancia y Alcances .....                       | 2    |
| 1.3. Delimitación.....                                  | 3    |
| 1.3.1. Temporal .....                                   | 3    |
| 1.3.2. Espacial.....                                    | 3    |
| 1.3.3. Académica.....                                   | 3    |
| 1.4. Innovación .....                                   | 4    |
| 1.5. Objetivos .....                                    | 4    |
| 1.5.1. Objetivo general .....                           | 4    |
| 1.5.2. Objetivos específicos.....                       | 4    |
| 2. FUNDAMENTOS TEÓRICOS .....                           | 5    |

|   |    |
|---|----|
| 2.1. El principio de funcionamiento del sensor de temperatura ..... | 5  |
| 2.2. Coeficiente de temperatura .....                               | 5  |
| 2.3. Sensor LM35 .....  | 6  |
| 2.4. Tipos de sensores de temperatura .....                         | 6  |
| 2.4.1. Termopar .....   | 6  |
| 2.4.2. Dispositivos resistivos (RTD y Termistores) .....            | 7  |
| 2.4.3. Dispositivos bimetálicos .....                               | 7  |
| 2.5. Dispositivos de dilatación de líquido .....                    | 8  |
| 2.6. Sistema de adquisición de datos (DAS o DAQ) .....              | 9  |
| 2.7. Daq USB 6009 .....   | 10 |
| 2.8. Relé de estado sólido (SSR) .....                              | 10 |
| 2.9. MOSFET .....   | 11 |
| 2.10. Controlador PID .....   | 12 |
| 2.10.1. Constante Proporcional (P) .....                            | 13 |
| 2.10.2. Constante Integral (I) .....                                | 13 |
| 2.10.3. Constante Derivativa (D) .....                              | 13 |
| 2.11. Redes Neuronales .....  | 13 |
| 2.12. Redes Neuronales artificial .....                             | 14 |
| 2.13. Enlaces de conexión .....                                     | 16 |
| 2.14. Sumador ( $\Sigma$ ): .....                                   | 16 |
| 2.15. Función de activación ( $\varphi$ ) .....                     | 16 |
| 2.16. LabVIEW .....   | 16 |
| 2.17. Diagrama de bloques .....                                     | 17 |
| 2.18. Complemento de LabVIEW PID Control y simulación .....         | 17 |
| 2.19. LabVIEW NXG .....   | 19 |
| 3. MARCO METODOLÓGICO .....   | 20 |

|   |    |
|---|----|
| 3.1. Sistema propuesto de control para el modulo .....                  | 20 |
| 3.1.1. Esquema de conexiones de la planta .....                         | 20 |
| 3.1.2. Conexión del DAQ6009 con LabVIEW NXG .....                       | 21 |
| 3.1.3. Configuración del DAQ6009 en LabVIEW NXG .....                   | 21 |
| 3.1.4. Configuración de una entrada analógica .....                     | 22 |
| 3.1.5. Configuración de una salida analógica .....                      | 23 |
| 3.1.6. Configuración de una entrada digital .....                       | 24 |
| 3.1.7. Configuración de una salida digital .....                        | 25 |
| 3.1.8. Controlador PID en LabVIEW NXG .....                             | 26 |
| 3.1.9. Obtención de la función de transferencia en MATLAB .....         | 27 |
| 3.1.10. Obtención de las ganancias mediante pid tuner MATLAB .....      | 29 |
| 3.1.11. Entrenamiento de la red neuronal en Matlab .....                | 32 |
| 3.1.12. Implementación en LabVIEW NXG .....                             | 35 |
| 4. RESULTADOS .....   | 37 |
| 4.1. RESULTADOS DEL CONTROLADOR PID .....                               | 38 |
| 4.2. RESULTADOS DEL CONTROLADOR NEURONAL.....                           | 40 |
| 4.3. COMPARACIÓN DE CONTROLADORES.....                                  | 42 |
| CONCLUSIONES .....  | 44 |
| RECOMENDACIONES.....  | 45 |
| REFERENCIAS BIBLIOGRAFICAS.....   | 46 |
| Código para graficas en Matlab individuales del PID y Neuronal.....     | 48 |
| Código para graficas en matlab comparación entre el PID y Neuronal..... | 51 |
| Enlace para descarga de documentos.....                                 | 52 |

## ÍNDICE DE FIGURAS

|  |    |
|--|----|
| Figura 1-1:: Delimitación espacial del módulo del Proyecto Técnico ..... | 3  |
| Figura 2-1: Sensor LM35 .....  | 6  |
| Figura 2-2 : Sensor de temperatura Termopar.....                         | 7  |
| Figura 2-3 : Sensor de temperatura RTD.....                              | 7  |
| Figura 2-4 : Sensor Bimetálico.....                                      | 8  |
| Figura 2-5: Sensor de dilatación por liquido.....                        | 8  |
| Figura 2-6: Diagrama de un sistema con Daq o Das .....                   | 9  |
| Figura 2-7: Daq USB 6009.....  | 10 |
| Figura 2-8: Relé de estado sólido .....                                  | 11 |
| Figura 2-9: MOSFET.....  | 11 |
| Figura 2-10: Controlador PID.....  | 12 |
| Figura 2-11:: Redes Neuronales biológicas .....                          | 14 |
| Figura 2-12:: Modelo de una red neuronal artificial.....                 | 15 |
| Figura 2-13:: Componentes de una red neuronal artificial .....           | 15 |
| Figura 2-14 : Labview2019 .....  | 17 |
| Figura 2-15 : LabVIEW 2019 Diagrama de Bloques .....                     | 17 |
| Figura 2-16 : Complemento de diseño y simulación .....                   | 18 |
| Figura 2-17 : Labview2019 .....  | 19 |
| Figura 3-1 : Esquema de la planta de temperatura.....                    | 20 |
| Figura 3-2 : Ventana de diseño de proyecto en LabVIEW NXG .....          | 21 |
| Figura 3-3 : Lista de bloques para comandos en LabVIEW NXG .....         | 22 |
| Figura 3-4 : Bloque de entrada analógica en LabVIEW NXG .....            | 22 |
| Figura 3-5 : Configuración de entrada analógica en LabVIEW NXG .....     | 23 |
| Figura 3-6 : Bloque de salida analógica en LabVIEW NXG .....             | 23 |
| Figura 3-7 : Configuración de salida analógica en LabVIEW NXG .....      | 24 |
| Figura 3-8 : Bloque de entrada digital en LabVIEW NXG .....              | 24 |



|  |    |
|--|----|
| Figura 3-9 : Configuración de entrada digital en LabVIEW NXG .....   | 25 |
| Figura 3-10 : Bloque de salida digital en LabVIEW NXG .....  | 25 |
| Figura 3-11 : Configuración de salida digital en LabVIEW NXG .....   | 26 |
| Figura 3-12 : Controlador PID en LabVIEW NXG .....   | 26 |
| Figura 3-13 : Datos de entrada y salida para obtención de función de<br>transferencia .....                | 27 |
| Figura 3-14 : Herramienta de identificación de sistemas de MATLAB .....                                    | 27 |
| Figura 3-15 : Herramienta de identificación de sistemas - ingreso de datos                                 | 28 |
| Figura 3-16 : Herramienta de identificación de sistemas – configuración para<br>obtención del modelo ..... | 28 |
| Figura 3-17 : Herramienta de identificación de sistemas – estimaciones para<br>el modelo .....             | 29 |
| Figura 3-18 : Obtención de la función de transferencia mediante el comando<br>paso. ....                   | 29 |
| Figura 3-19 : Herramienta Pid tuner de MATLAB .....  | 30 |
| Figura 3-20 : Herramienta Pid tuner de MATLAB – configuración .....  | 30 |
| Figura 3-21 : Ganancias del controlador PID a partir de la función de<br>transferencia .....               | 31 |
| Figura 3-22 : Ganancias del controlador PID a partir de la función de<br>transferencia .....               | 31 |
| Figura 3-23 : Estructura de la neurona artificial a entrenar en Matlab .....                               | 32 |
| Figura 3-24 : Programa para el entrenamiento de la red neuronal en<br>Matlab.....                          | 32 |
| Figura 3-25 : Resultados del entrenamiento en Matlab .....   | 33 |
| Figura 3-26 : Seguimiento de los valores de entrada previo al<br>entrenamiento.....                        | 33 |
| Figura 3-27 : Entrenamiento de la red neuronal y las épocas.....   | 34 |
| Figura 3-28 : Seguimiento de los valores de entrada previo al<br>entrenamiento.....                        | 34 |

|  |    |
|--|----|
| Figura 3-29 : Historió grama del error en la red neuronal .....                        | 35 |
| Figura 3-30 : Desarrollo del control neuronal en LabVIEW NXG .....                     | 35 |
| Figura 3-31 : Función en LabVIEW NXG para perceptrón.....                              | 36 |
| Figura 3-32 : Implementación del control neuronal en LabVIEW NXG .....                 | 36 |
| Figura 4-1 : Panel frontal del controlador PID y del controlador neuronal....          | 37 |
| Figura 4-2 : Controlador Pid con referencia de 40°C .....                              | 38 |
| Figura 4-3 : Controlador Pid con referencia de 60°C .....                              | 39 |
| Figura 4-4 : Controlador Pid con referencia de 100°C .....                             | 39 |
| Figura 4-5 : Controlador neuronal con referencia de 40°C .....                         | 40 |
| Figura 4-6 : Controlador neuronal con referencia de 60°C .....                         | 41 |
| Figura 4-7 : Controlador neuronal con referencia de 100°C .....                        | 41 |
| Figura 4-8 : Comparación de controladores PID y Neuronal con referencia de 40°C .....  | 42 |
| Figura 4-9 : Comparación de controladores PID y Neuronal con referencia de 60°C.....   | 43 |
| Figura 4-10 : Comparación de controladores PID y Neuronal con referencia de 100°C..... | 43 |

## ÍNDICE DE TABLAS

No se encuentran elementos de tabla de ilustraciones.

## INTRODUCCIÓN

El proyecto está basado en la implementación de un sistema de control basado en lógica difusa aplicado a un control de temperatura por medio de una tarjeta de adquisición de datos de National Instrument NI DAQ 6009 teniendo como programa de instrumentación virtual el LabVIEW NXG permitiendo programar de manera práctica y visualizar los resultados en la interfaz gráfica o panel frontal donde se encontrarán los indicadores gráficos y las respuestas del sistema.

El proyecto está enfocado en los estudiantes de la carrera de Ingeniería en Automatización Industrial de la Universidad Politécnica Salesiana Sede Guayaquil.

El objetivo principal de este proyecto es implementar un banco didáctico en el cual el estudiante tenga un equipo el cual le permita desarrollar rutinas de aprendizaje y adquirir conocimiento enfocados a sistemas de control inteligentes.

Se proponen prácticas básicas y otras enfocadas en el tema de titulación. Se explican varios procesos y aplicaciones que un estudiante puede lograr con el módulo didáctico demostrando sus destrezas para crear e innovar más procesos a nivel industrial.

## **1. EL PROBLEMA**

### **1.1. Antecedentes**

Los sistemas de temperatura son necesarios en la mayoría de las aplicaciones industriales. Los procesos manejados en las industrias tienen factores que influyen en los procesos y muchas veces no son tenidos en cuenta en los modelos lineales.

Para el diseño del control PID convencional es necesario linealizar el sistema en un punto de operación, garantizando que el control trabaje bien en esta zona, pero cuando se aleja del punto de operación pierde efectividad el controlador, es por ello que es necesario diseñar un control que garantice el buen funcionamiento del control sin importar las no linealidades del proceso, por consiguiente, se diseña un control con redes neuronales MPC (Model Predictive Control). Para comprobar el funcionamiento de este proyecto se realizó prácticas acordes a los temas indicados por el docente para ver la viabilidad de este proyecto de titulación en el cual deberemos utilizar conocimientos tanto de teoría de control I y II., microprocesadores y electiva.

### **1.2. Importancia y Alcances**

Con el presente proyecto el estudiante constara con una herramienta la cual permita desarrollar conocimientos prácticos y teóricos en materias como Teoría de control I y II, Redes de computadoras, electiva I y I, siendo una herramienta de estudio para los docentes y estudiantes de la Universidad Politécnica Salesiana Sede Guayaquil de la carrera Ingeniería electrónica.

Se podrá realizar pruebas físicas, simulaciones e investigaciones que les permita realizar aplicaciones para futuros proyectos a nivel académico e industrial utilizando software especializado en sistemas de control basados en lógica neuronal y LabVIEW NXG.

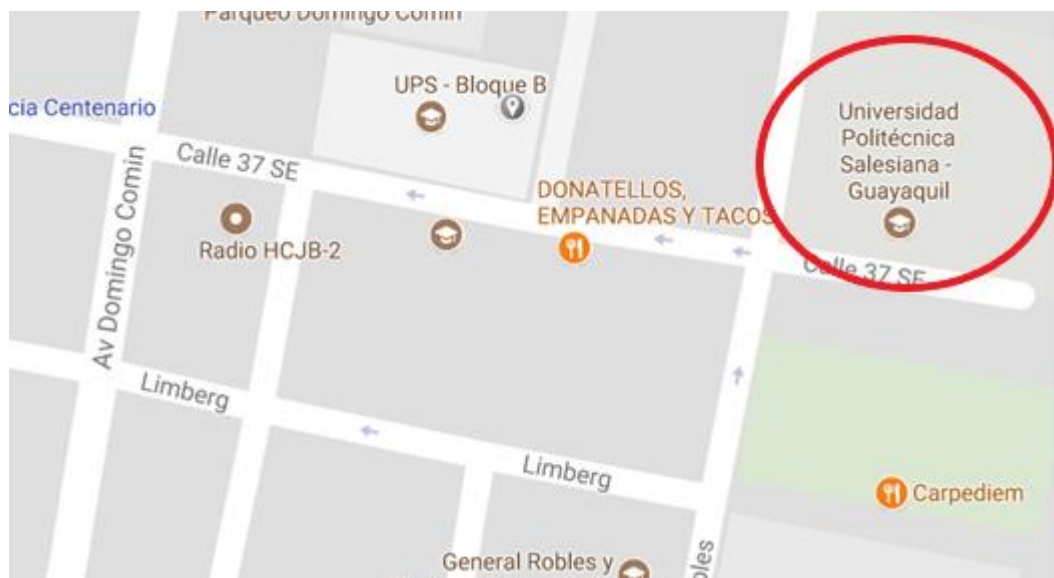
### 1.3. Delimitación

#### 1.3.1. Temporal

La implementación de este proyecto se realizó en un intervalo de seis meses a partir de la aprobación de este.

#### 1.3.2. Espacial

El proyecto se desarrolla en las instalaciones del Laboratorio de Automatización Industrial de la carrera Ingeniería Electrónica de la Universidad Politécnica Salesiana de la Sede Guayaquil. Figura 1.



**Figura 1-1** Delimitación espacial del módulo del Proyecto Técnico

Fuente: (Moreira, Rodríguez)

#### 1.3.3. Académica

El módulo diseñado, provee un alcance académico que involucran conocimientos obtenidos en las siguientes materias: Electiva I, Electiva II, Teoría del Diseño, Electrónica de Potencia, Circuitos Eléctricos I, Procesamiento Digital de Señales, Informática Industrial.

## **1.4. Innovación**

El módulo y su aplicación con el control de temperatura, ayuda al entendimiento de algoritmos basados en lógica difusa mediante nuevas herramientas de instrumentación virtual como el LabVIEW NXG.

## **1.5. Objetivos**

### **1.5.1. Objetivo general**

- Desarrollar un algoritmo basado en redes neuronales aplicado a un banco didáctico para el control de temperatura.

### **1.5.2. Objetivos específicos**

- Desarrollar una guía para la creación de un algoritmo neuronal para el control de la temperatura en un módulo didáctico.
- Realizar una comparación entre un control pid convencional y el sistema neuronal.
- Desarrollar un módulo didáctico para el entrenamiento de proceso de temperatura utilizando un DAQ6008 y los actuadores.
- Realizar un análisis de las ventajas de utilizar sistemas neuronales.
- Realizar una comparación entre el tiempo de estabilización entre un controlador neuronal y un control pid clásico.

## 2. FUNDAMENTOS TEÓRICOS

### 2.1. El principio de funcionamiento del sensor de temperatura

Los sensores de temperatura funcionan aprovechando una característica física de algunos materiales conductores y semiconductores, estos materiales son capaces de variar la resistencia eléctrica en función de la temperatura ambiente, gracias a este principio podemos describir el funcionamiento de un sensor de temperatura de cualquier tipo (Veloso, 2016).

Existen materiales semiconductores con coeficiente de temperatura negativo (Cuando aumenta la temperatura la resistencia disminuye) y otros con coeficiente de temperatura positivo (Cuando aumenta la temperatura la resistencia aumenta) (Veloso, 2016)..

### 2.2. Coeficiente de temperatura

El coeficiente de temperatura, habitualmente simbolizado como  $\alpha$ , es una propiedad intensiva de los materiales que cuantifica la relación entre la variación de la propiedad física de un material y el cambio de temperatura.

Por tanto, es el cambio relativo de una propiedad física cuando la temperatura se cambia un K. Este coeficiente se expresa según el Sistema Internacional de Unidades en K (Paul, 2008). Su expresión matemática toma la forma de la ecuación 1.

$$\alpha(T) = \frac{1}{R(T_0)} \cdot \frac{\partial R(T)}{\partial T} \quad (1)$$

Donde:

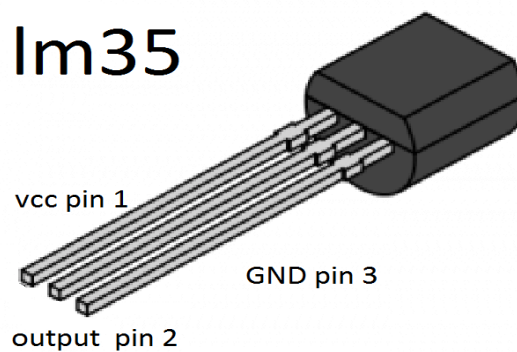
- $\alpha$  es el coeficiente de temperatura, que puede variar con la temperatura;
- $R(T)$  es la magnitud de la propiedad física del material a la temperatura T



- $R(T_0)$  es la magnitud de la propiedad física del material a la temperatura de referencia  $T_0$

### 2.3. Sensor LM35

La serie LM35 son circuitos integrados de precisión dispositivos de temperatura con un voltaje de salida linealmente proporcional a la temperatura en grados centígrados, tiene una ventaja sobre el lineal sensor de temperatura calibrados en Kelvin, no requiere ninguna calibración externa o recorte para proporcionar precisiones de  $\pm \frac{1}{4}^{\circ} \text{C}$  a temperatura ambiente y  $\pm \frac{3}{4}^{\circ} \text{C}$  en un rango de temperatura completo de  $-55^{\circ} \text{C}$  a  $150^{\circ} \text{C}$ . Inferior (Texas Instruments, 2017).



**Figura 2-1:** Sensor LM35

Fuente: (hetpro-store, 2017)

## 2.4. Tipos de sensores de temperatura

### 2.4.1. Termopar

Este tipo de sensores como se muestra en la figura 2-2, se basan en dos tiras fabricadas de diferentes metales unidos en un extremo. Los cambios de temperatura que se producen en esa junta inducen una variación en la fuerza electromotriz (FEM) entre los extremos (Benavides, 2017).



**Figura 2-2 :** Sensor de temperatura Termopar

Fuente: (Benavides, 2017).

### **2.4.2. Dispositivos resistivos (RTD y Termistores)**

Estos dispositivos funcionan debido a que la resistencia eléctrica de un material cambia al variar su temperatura. Entre sus dispositivos metálicos más comunes se encuentran los RTD y los termistores como se evidencia en la figura 2-3 (Benavides, 2017).



**Figura 2-3 :** Sensor de temperatura RTD

Fuente: (Benavides, 2017).

### **2.4.3. Dispositivos bimetálicos**

Debido a la diferencia en la tasa de dilatación térmica entre los diferentes metales, estos sensores aprovechan esa función para poder realizar la medición. Se agrupan entre si tiras o dos metales. Si un lado se calienta más se dilata más que el otro resultando en una curvatura que se traduce a una

lectura de temperatura a través de una articulación mecánica a un apuntador como se observa en la figura 2-4 (Sánchez, 2016)



**Figura 2-4 :** Sensor Bimetálico

Fuente: (Sánchez, 2016).

## 2.5. Dispositivos de dilatación de líquido

Son los sensores de temperatura más comunes que existen como lo son los domésticos, se divide en dos, el que contiene mercurio considerado un riesgo ambiental y el de líquido orgánico como se muestra en la figura 2-5 (Sánchez, 2016)



**Figura 2-5:** Sensor de dilatación por líquido

Fuente: (Sánchez, 2016)

## 2.6. Sistema de adquisición de datos (DAS o DAQ)

Un sistema de adquisición de datos (DAS o DAQ) es un grupo de sensores, componentes de hardware y programas de software que trabajan juntos con el fin de recoger las señales del mundo físico o real. El DAS puede ser tan simple como una computadora personal con software y periféricos de hardware de adquisición de datos. Si el DAS es como un sistema independiente con software dedicado que solo es capaz de realizar la adquisición de datos, puede denominarse un registrador de datos.

La primera etapa de cualquier DAS son los sensores o transductores. Estos dispositivos convierten algún tipo de señal física del mundo real (por ejemplo, sonido, presión, temperatura ambiente, tasa de flujo de fluidos, etc.) en una señal eléctrica. Generalmente, el DAS tendrá algunos circuitos de acondicionamiento de señal, como el filtrado analógico en la cadena de señales. Este acondicionamiento prepara la señal para la conversión de los datos de analógico a digital a través del convertidor analógico digital (ADC).

Estos datos convertidos se envían a la unidad de procesamiento principal, como una computadora, a través de una ranura o puerto. La unidad de procesamiento principal manipula los datos mediante el uso de software y hardware dedicados (Arrow , 2015).



**Figura 2-6:** Diagrama de un sistema con Daq o Das

Fuente: (Agüero, 2017)

## 2.7. Daq USB 6009

Tarjeta de adquisición de datos con interfaz USB posee entradas y salidas multifunción, 8 entradas analógicas de 14 Bits, 2 salidas analógicas, es un dispositivo de bajo costo, brinda funcionalidad básica para aplicaciones como registro de datos simple, medidas portátiles y experimentos académicos de laboratorio (Ni, 2019)



**Figura 2-7:** Daq USB 6009

Fuente: (Ni, 2019)

## 2.8. Relé de estado sólido (SSR)

Es un dispositivo de conmutación electrónico que se enciende o apaga cuando se aplica un voltaje externo (CA o CC) a través de sus terminales de control. Tiene la misma función que un relé electromecánico, pero no tiene partes móviles y, por lo tanto, da como resultado una vida útil más larga. Los SSR consisten en un sensor que responde a una entrada apropiada (señal de control), un dispositivo de conmutación electrónico de estado sólido que cambia la energía al circuito de carga y un mecanismo de acoplamiento para permitir que la señal de control active este interruptor sin partes mecánicas. El relé puede estar diseñado para conmutar AC o cargas de CC (Pulido, 2000)



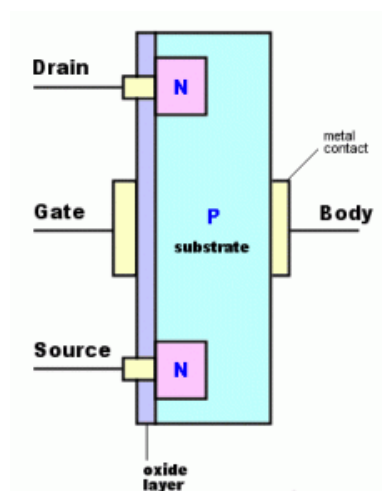
**Figura 2-8:** Relé de estado sólido

Fuente: (Pulido, 2000)

## 2.9. MOSFET

El transistor de efecto de campo semiconductor de óxido metálico, también conocido como transistor de óxido metálico y silicio (transistor MOS o MOS), es un tipo de puerta aislada transistor de efecto de campo que se fabrica mediante la oxidación controlada de un semiconductor, normalmente silicio.

El voltaje de la puerta cubierta determina la conductividad eléctrica del dispositivo; esta capacidad de cambiar la conductividad con la cantidad de voltaje aplicado se puede utilizar para amplificación o conmutación de señales electrónicas (Kang, 2000) .



**Figura 2-9:** MOSFET

Fuente: (Kang, 2000) .

## 2.10. Controlador PID

El control pid es un algoritmo el cual consta de tres parámetro o ganancias proporcional, integral y derivativo , el valor proporcional depende del error entre el valor medido y la referencia representada por la fórmula 1, el valor integral consta de los valores de los errores pasados representados por la fórmula 2 y el valor derivativo depende de una predicción de los valores futuros representado por la fórmula 3, fue patentado en 1939 por Albert Calender y Allan Stevenson de la firma Imperial Chemicoles (Åström, 1995), la formula del algoritmo pid está representado por la fórmula 4.

$$K_p e(t) \quad (1)$$

$$\frac{1}{T_i} \int_0^t e(\tau) d\tau \quad (2)$$

$$T_d \frac{de(t)}{dt} \quad (3)$$

$$u(t) = K_p [e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt}] \quad (4)$$

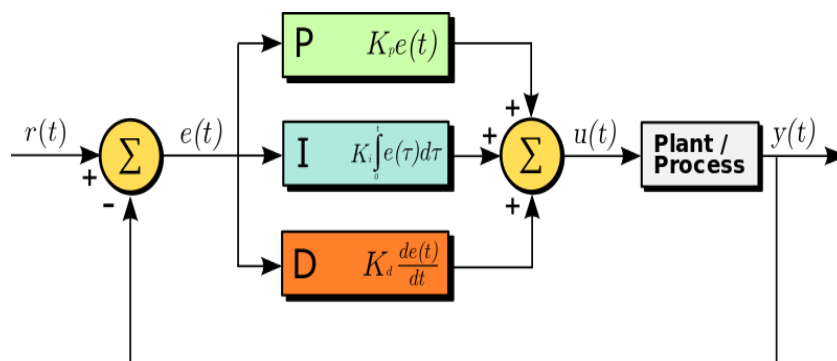
Donde:

$K_p$  es la acción proporcional

$T_i$  es la acción intergral

$T_d$  es la acción derivativa

La representación del pid en diagrama se muestra en la figura 2-10



**Figura 2-10:** Controlador PID.

Fuente: (Manuel Baltieri, 2018)

### **2.10.1. Constante Proporcional (P)**

La constante proporcional es el producto entre la señal de error y la constante proporcional conteniendo como resultante el error de estado estacionario para su aproximación a cero véase en la figura 19. (Astrom, 1995).

### **2.10.2. Constante Integral (I)**

La constante integral ayuda a disminuir el error en estado estacionario generado por perturbaciones exteriores, integrando la derivación en el tiempo que ocurre entre la variable y el punto de referencia véase en figura 20. (Astrom, 1995).

### **2.10.3. Constante Derivativa (D)**

La constante derivativa funciona para mantener el error de estado estacionario al mínimo corrigiéndole periódicamente que se genera en el cambio en un valor absoluto véase en Figura 21. (Astrom, 1995).

## **2.11. Redes Neuronales**

Ya en 1911, con los primeros estudios de Santiago Ramón y Cajal, las neuronas se definieron como el componente básico del sistema nervioso, incluyendo al cerebro, el cual está formado por un enorme número de ellas.

Así mismo, existen numerosos estudios que muestran que estas neuronas están conectadas de forma masiva entre sí, y aunque esta cualidad está presente en todos los animales es mucho más acentuada y evidente en los seres humanos, característica que nos permite ser inteligentes y nos discrimina del resto de seres vivos (Hernández, 2010). (“Control Por Redes Neuronales | Neurona | Red neuronal artificial”)



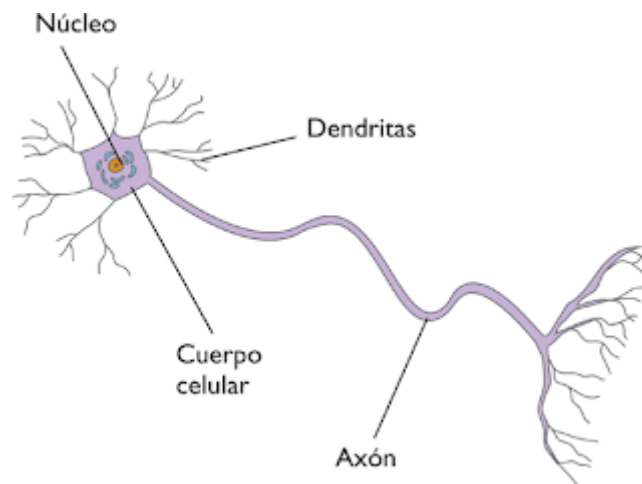
## 2.12. Redes Neuronales artificial

Una red neuronal artificial (RNA) es un elemento capaz de procesar gran cantidad de información de forma paralela y distribuida, inspirada de las redes neuronales biológicas (ver Figura 2-11) las cuales pueden almacenar conocimiento experimental y tenerlo disponible para su uso (Moreno Armendariz, 2001) .

Esta tiene algunas similitudes con el cerebro, como lo son:

"1. El conocimiento es adquirido a través del proceso de aprendizaje."  
("Redalyc. Control adaptable indirecto usando Redes Neuronales Dinámicas")

2. La conectividad entre neuronas es llamada pesos sinápticos y son utilizados para almacenar el conocimiento ("Muy Bueno Levitación | Red neuronal artificial | Neurona - Scribd")



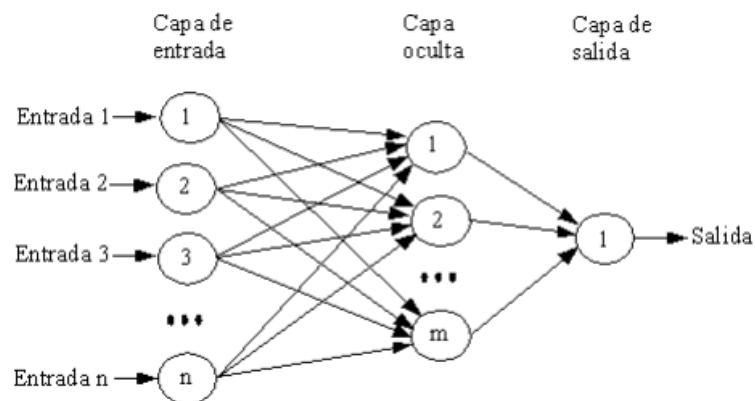
**Figura 2-11::** Redes Neuronales biológicas

Fuente: (Moreno Armendariz, 2001) .

Una red neuronal artificial está formada por una capa de entrada, un determinado número de capas ocultas (que también puede ser cero) y una capa de salida. ("Control Por Redes Neuronales | Neurona | Red neuronal artificial") Entre cada una de las capas que forman la red neuronal existen matrices de pesos sinápticos, que son los que le otorgan a la red neuronal su potencial de cálculo y capacidad de aprendizaje, ya que son estos los que

varían a medida que el sistema aprende (Moreno Armendariz, 2001). (“Redes neuronales en control de sistemas”)

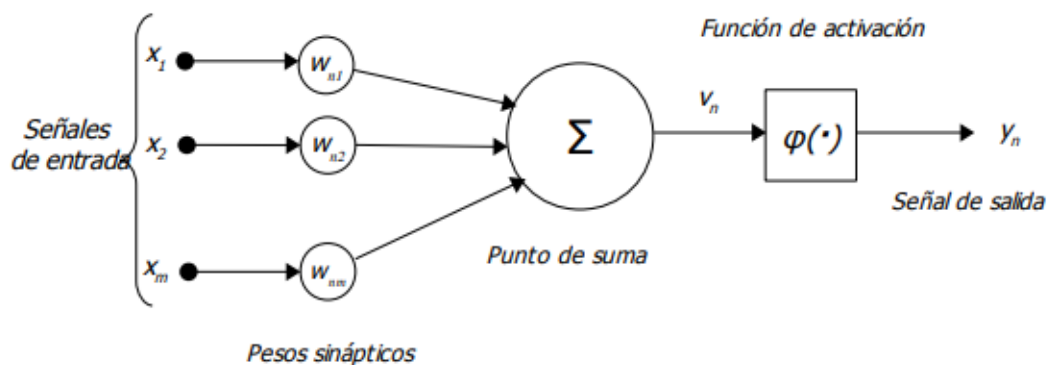
Como vemos en la figura 9 cada una de las capas está formada por nodos interconectados entre sí de forma masiva (cada una de las neuronas de una capa está conectada con todas las demás neuronas de las capas adyacentes), cada uno de estos nodos son neuronas artificiales y son el elemento básico de una red neuronal (Moreno Armendariz, 2001)..



**Figura 2-12::** Modelo de una red neuronal artificial.

Fuente: (Moreno Armendariz, 2001) .

En el modelo de neurona artificial mostrado en la Figura 10 se pueden identificar tres elementos que componen una red neuronal:



**Figura 2-13::** Componentes de una red neuronal artificial

Fuente: (Moreno Armendariz, 2001) .

### **2.13. Enlaces de conexión**

Estos están parametrizados por los pesos sinápticos  $w_{ij}$ . El primer subíndice se refiere a la neurona receptora y el segundo a la neurona emisora. Si  $w_{ij}$  es positivo se trata de una conexión excitadora, si es negativo entonces es inhibitoria. (“Redes Neuronales... Solo Un Capítulo | Red neuronal artificial - Scribd”) En una neurona biológica la dendrita y el axón funcionan como la neurona receptora y emisora respectivamente en una red neuronal artificial (Báez, 2016).

### **2.14. Sumador ( $\Sigma$ ):**

Es el resultado de sumar cada uno de los componentes de las señales de entrada multiplicadas por  $w_{ij}$ . El sumador realiza la misma función de procesamiento de datos que el núcleo de una neurona biológica (Báez, 2016)..

### **2.15. Función de activación ( $\phi$ )**

Se trata de una función de transformación generalmente no lineal. Este elemento es equivalente a la sinapsis en la neurona biológica, la cual otorgaba a la red neuronal biológica la capacidad de cálculo no lineal (si la salida es no-lineal) (Báez, 2016).. (“Redes Neuronales... Solo Un Capítulo | Red neuronal artificial - Scribd”)

### **2.16. LabVIEW**

Es el acrónimo de Laboratory Virtual Engineering Workbench, es un lenguaje de programación con un entorno gráfico en el cual se pueden desarrollar aplicaciones de control (Vizcaíno, 2011).

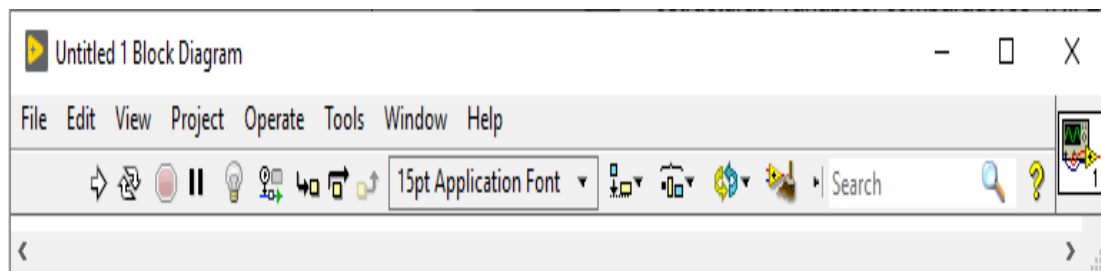


**Figura 2-14** : Labview2019

Fuente: (Vizcaíno, 2011).

## 2.17. Diagrama de bloques

En la figura 2-18, el diagrama de bloques es el entorno donde se programa incluyen funciones o bloques de control los cuales realizan operaciones como el ser constantes, estructuras, variables, comparadores (Vizcaíno, 2011).



**Figura 2-15** : LabVIEW 2019 Diagrama de Bloques

Fuente: (Vizcaíno, 2011).

## 2.18. Complemento de LabVIEW PID Control y simulación

El módulo LabVIEW de control diseño y simulación, permite la simulación de sistemas dinámicos, diseñar controladores e implementar sistemas de control en hardware en tiempo real (Vizcaíno, 2011). ("LabVIEW Para Simulación y Diseño de Control")

El Módulo LabVIEW de control diseño y simulación es un software complementario que se integra con el entorno de programación de LabVIEW

para ofrecer habilidades como tecnologías integradas de paralelismo, multi núcleo y de velocidad variable, así como herramientas para implementar en hardware en tiempo real. Usted puede integrar medidas con diseño para identificación del sistema, calibración del modelo o validación del modelo.

(“Módulo LabVIEW Control Design and Simulation - NI”) Después usted puede implementar su algoritmo en hardware embebido en tiempo real para rápida generación de prototipos de control y aplicaciones de hardware en el ciclo como se muestra en la figura 2-19 (Vizcaíno, 2011).



**Figura 2-16** : Complemento de diseño y simulación

Fuente: (Vizcaíno, 2011).

## 2.19. LabVIEW NXG

Es el Software de Desarrollo Web G le ayuda a crear interfaces de usuario basadas en web para aplicaciones de pruebas y medidas sin necesidad de conocimientos de desarrollo web. (“Software de Desarrollo Web G - NI”) (Vizcaíno, 2011).



**Figura 2-17** : Labview2019

Fuente: (Vizcaíno, 2011).

### 3. MARCO METODOLÓGICO

#### 3.1. Sistema propuesto de control para el modulo

El sistema propuesto de control es un HMI SCADA asistido con redes neuronales para monitorear y controlar los cambios de temperatura utilizando una tarjeta de adquisición de datos DAQ6009 mediante el software LabVIEW NXG.

El proyecto consiste en el diseño de un módulo de pruebas para el control de temperatura mediante el uso de un foco dicroico y un sensor Im35 como retroalimentación del sistema en donde se ejecuta un control de lazo cerrado pid y un controlador basado en lógica neuronal.

##### 3.1.1. Esquema de conexiones de la planta

En la Figura 3-1 se muestran los componentes de la planta de temperatura teniendo como núcleo una tarjeta DAQ6009 conectándose a un relé solido para la activación del foco dicroico que trabaja con voltaje alterno, una entrada analógica conectada al sensor LM35 para la lectura de la temperatura y una salida digital para la activación del motor mediante una tarjeta controladora de motores L923D.

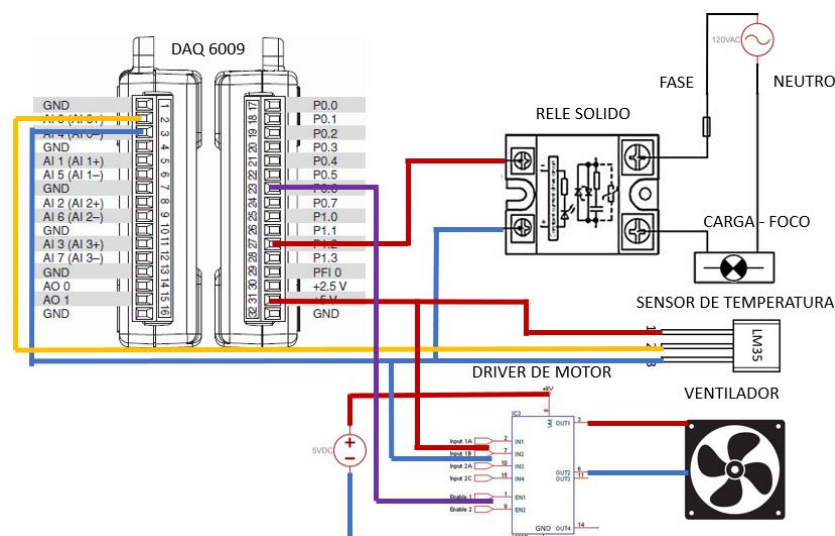
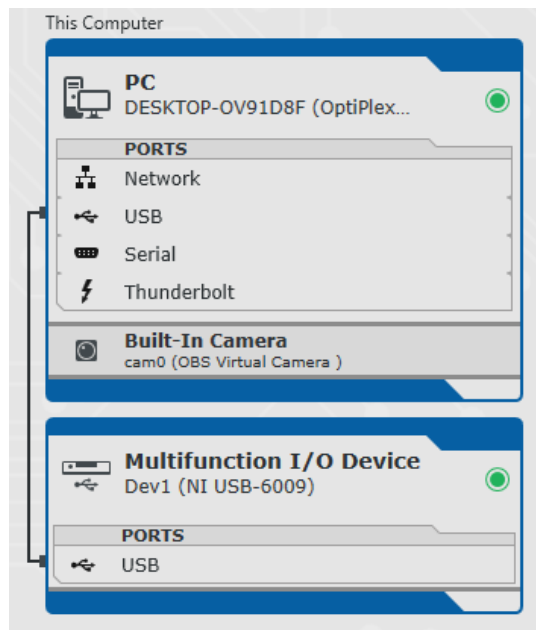


Figura 3-1 : Esquema de la planta de temperatura

Fuente: (Moreira, Rodríguez)

### 3.1.2. Conexión del DAQ6009 con LabVIEW NXG

En la implementación de un controlador en LabVIEW NXG es necesario tener el equipo conectado de manera que la detección del dispositivo sea automática y se muestre en la ventana del diseño del proyecto como se muestra en la Figura 3-1.



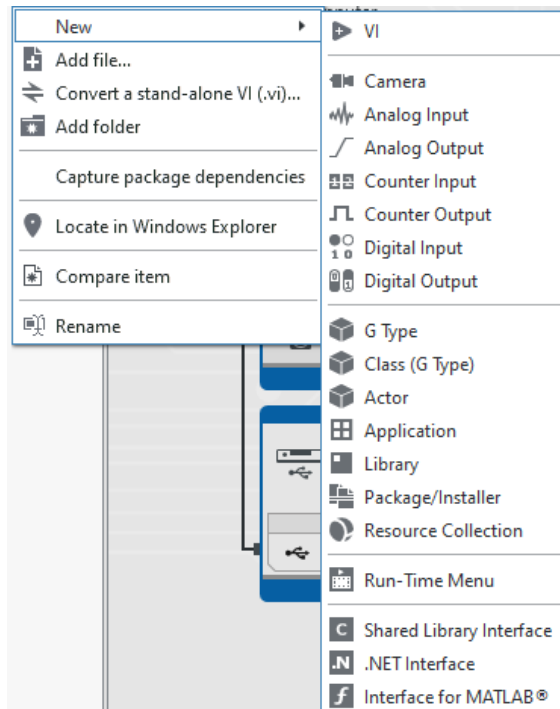
**Figura 3-2 :** Ventana de diseño de proyecto en LabVIEW NXG

Fuente: (Moreira, Rodríguez)

### 3.1.3. Configuración del DAQ6009 en LabVIEW NXG

Para utilizar la tarjeta de adquisición de datos Daq es necesario crear un bloque de memoria que contenga el elemento ya sea entrada o salida como digital o analógica, para esto se tendrá que dar clic en el proyecto como se muestra en la Figura 3-3.

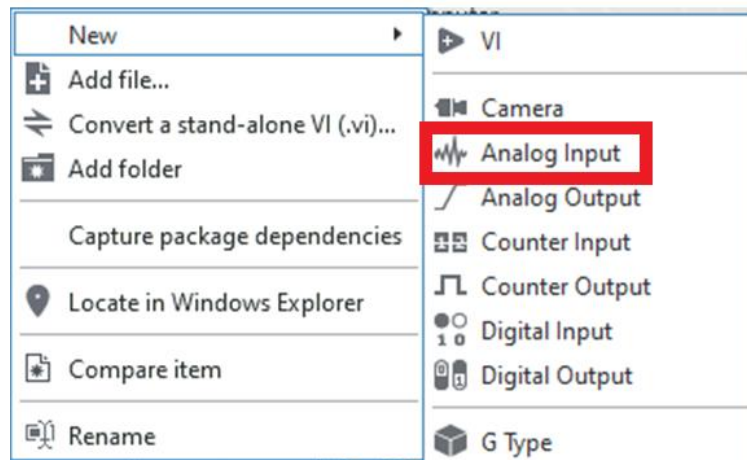




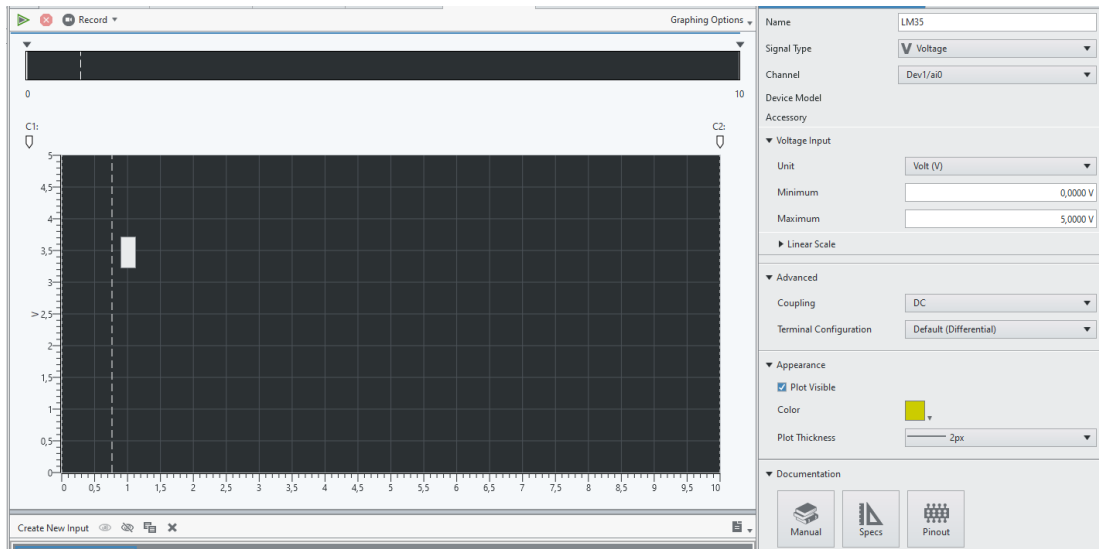
**Figura 3-3 :** Lista de bloques para comandos en LabVIEW NXG  
Fuente: (Moreira, Rodríguez)

### 3.1.4. Configuración de una entrada analógica

En la Figura 3-4 se detalla que para agregar una entrada análoga es necesario dar clic en el proyecto y seleccionar analog input, luego se debe seleccionar el pin de la entrada, rango de voltaje y tipo de acople como se muestra en la Figura 3-5



**Figura 3-4 :** Bloque de entrada analógica en LabVIEW NXG  
Fuente: (Moreira, Rodríguez)

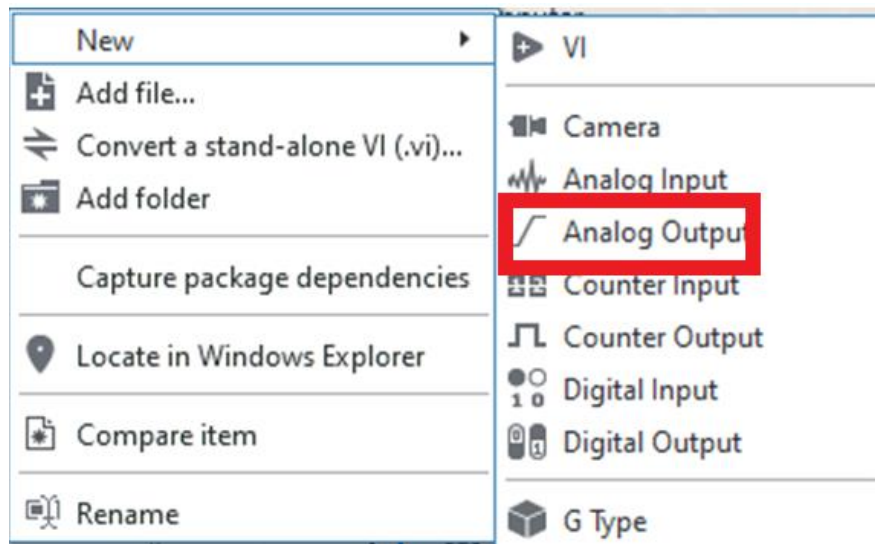


**Figura 3-5 :** Configuración de entrada analógica en LabVIEW NXG

Fuente: (Moreira, Rodríguez)

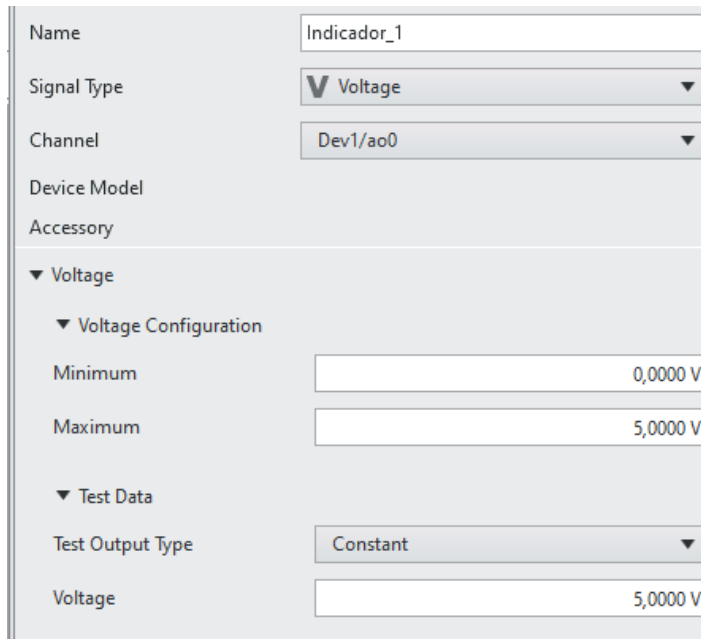
### 3.1.5. Configuración de una salida analógica

En la Figura 3-6 se detalla que para agregar una salida analógica es necesario dar clic en el proyecto y seleccionar analog output, luego se selecciona el pin de salida, rango de voltaje como se muestra en la Figura 3-7



**Figura 3-6 :** Bloque de salida analógica en LabVIEW NXG

Fuente: (Moreira, Rodríguez)

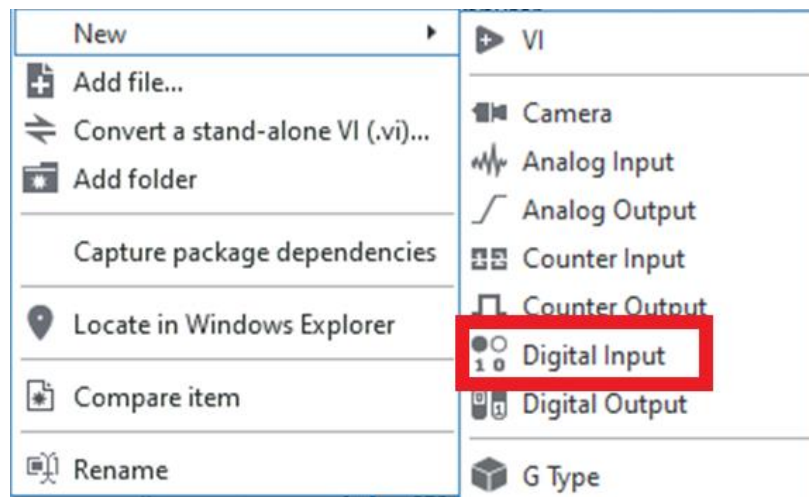


**Figura 3-7 :** Configuración de salida analógica en LabVIEW NXG

Fuente: (Moreira, Rodríguez)

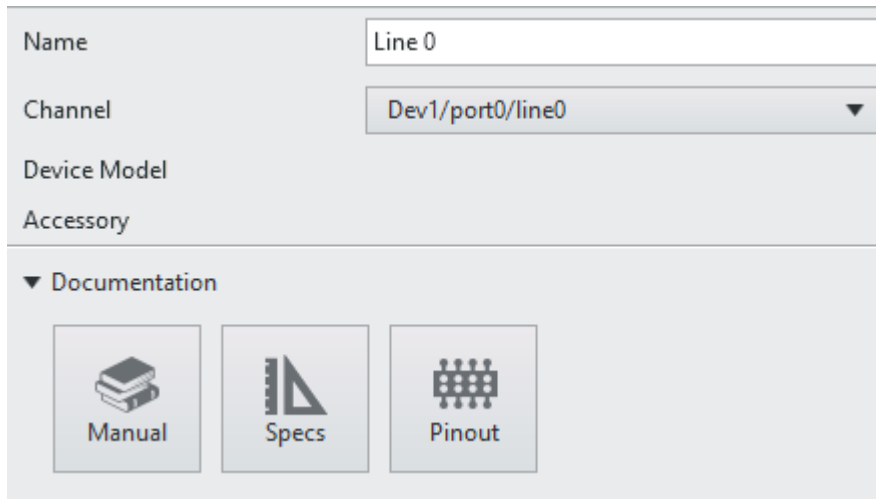
### 3.1.6. Configuración de una entrada digital

En la Figura 3-8 se detalla que para agregar una entrada digital es necesario dar clic en el proyecto y seleccionar digital input, luego se selecciona el pin de entrada como se muestra en la Figura 3-9



**Figura 3-8 :** Bloque de entrada digital en LabVIEW NXG

Fuente: (Moreira, Rodríguez)

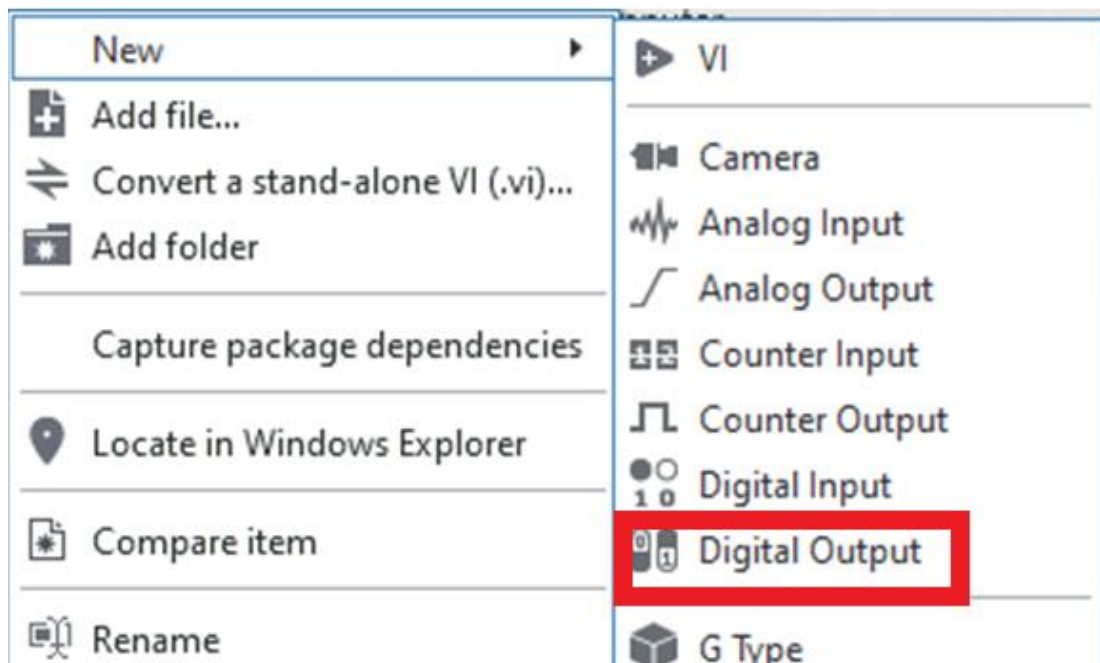


**Figura 3-9** : Configuración de entrada digital en LabVIEW NXG

Fuente: (Moreira, Rodríguez)

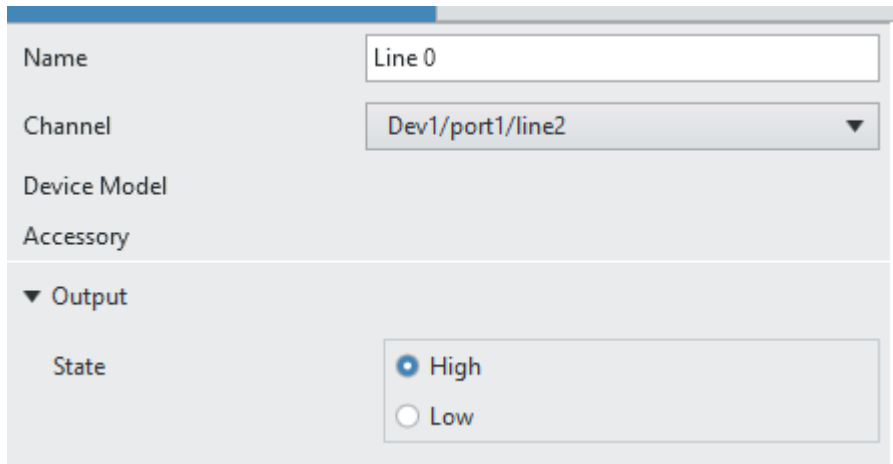
### 3.1.7. Configuración de una salida digital

En la Figura 3-10 se detalla que para agregar una entrada digital es necesario dar clic en el proyecto y seleccionar digital input, luego se selecciona el pin de salida como se muestra en la Figura 3-11



**Figura 3-10** : Bloque de salida digital en LabVIEW NXG

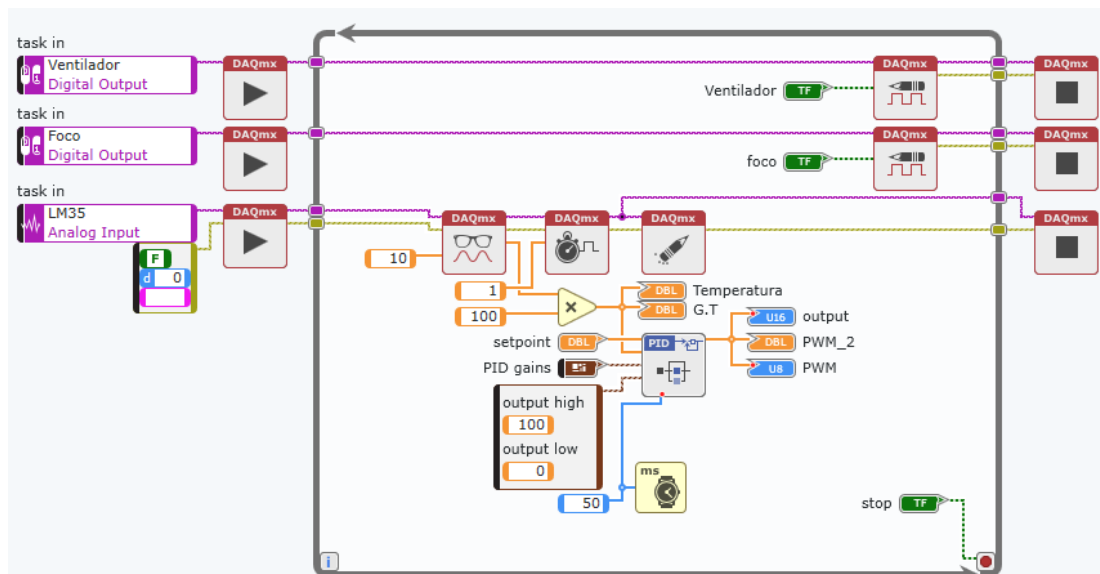
Fuente: (Moreira, Rodríguez)



**Figura 3-11 :** Configuración de salida digital en LabVIEW NXG  
Fuente: (Moreira, Rodríguez)

### 3.1.8. Controlador PID en LabVIEW NXG

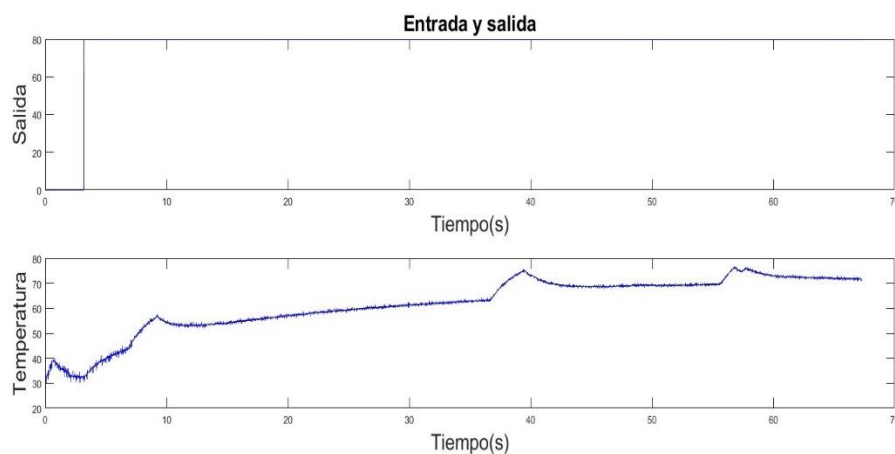
Para la implementación de un controlador pid es necesario programar en el bloque de programas que es una característica del LabVIEW basada en código G mediante la utilización de bloques, donde se llaman a los bloques de salida analógica para el control del foco como actuador, teniendo el sensor de temperatura como una entrada analógica como variable de proceso, en el bloque pid se conectan las variables de referencia, proceso y su salida como se muestra en la Figura 3-12.



**Figura 3-12 :** Controlador PID en LabVIEW NXG  
Fuente: (Moreira, Rodríguez)

### 3.1.9. Obtención de la función de transferencia en MATLAB

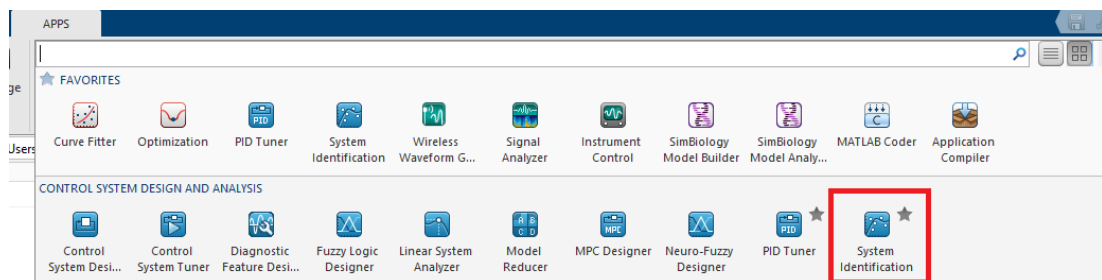
En el desarrollo de un controlador pid es fundamental la sintonización del controlador configurando las ganancias de este como el proporcional, integral y derivativo, para obtener los valores aproximados se utilizó la herramienta de Matlab ident teniendo como dato de entrada la temperatura y el dato de salida el valor de excitación del actuador o foco mediante una señal de modulación PWM, para los valores se optó por realizar una señal paso como se muestra en la Figura 3-13.



**Figura 3-13 :** Datos de entrada y salida para obtención de función de transferencia

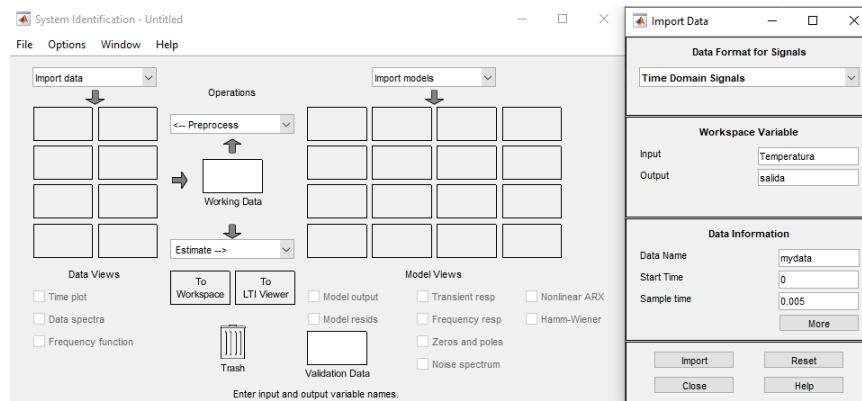
Fuente: (Moreira, Rodríguez)

Para la obtención de los valores en MATLAB se utiliza la herramienta de identificación de sistemas y se agregan las variables de entrada siendo la temperatura y la salida la modulación del Pwm como se muestra en la Figura 3-14 y 3-15.



**Figura 3-14 :** Herramienta de identificación de sistemas de MATLAB

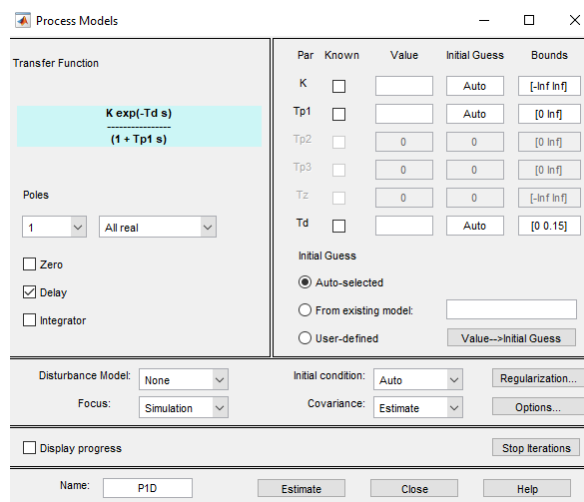
Fuente: (Moreira, Rodríguez)



**Figura 3-15 :** Herramienta de identificación de sistemas - ingreso de datos

Fuente: (Moreira, Rodríguez)

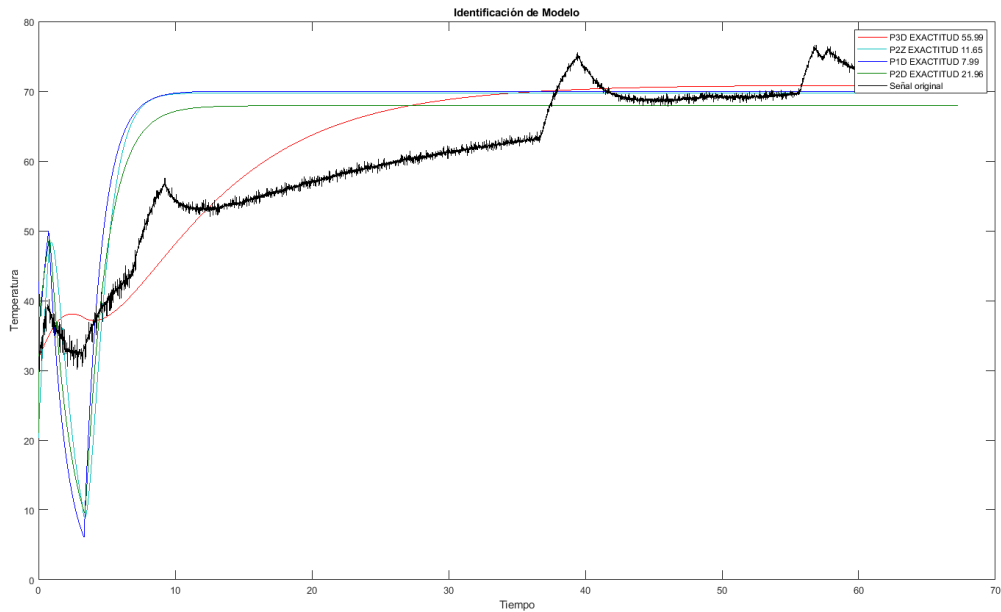
Teniendo los valores de entrada y salida se configura el método para la obtención del modelo que domina al sistema, en la Figura 3-16 se muestra las opciones donde se selecciona la cantidad de polos, ceros, retrasos e integradores para evaluar el sistema.



**Figura 3-16 :** Herramienta de identificación de sistemas – configuración para obtención del modelo

Fuente: (Moreira, Rodríguez)

En la Figura 3-17 se muestra las estimaciones para encontrar el modelo de la planta, se debe seleccionar el que tenga el valor más alto de aproximación a la planta de temperatura y exportar dicha estimación al espacio de trabajo en Matlab.



**Figura 3-17 :** Herramienta de identificación de sistemas – estimaciones para el modelo  
Fuente: (Moreira, Rodríguez)

En el espacio de trabajo se ejecuta el comando de paso como se muestra en la Figura 3-18 obteniendo la función de transferencia del sistema.

```
>> H=tf(P3D)

H =

From input "u1" to output "y1":

          0.8861
exp(-0.15*s) * -----
          0.0248 s^3 + 23.63 s^2 + 10.95 s + 1

Name: P3D
Continuous-time transfer function.
```

**Figura 3-18 :** Obtención de la función de transferencia mediante el comando paso.  
Fuente: (Moreira, Rodríguez)

### 3.1.10. Obtención de las ganancias mediante pid tuner MATLAB

Para el proceso de obtención de las ganancias del controlador pid se utiliza la herramienta pid tuner que se muestra en la Figura 3-19.

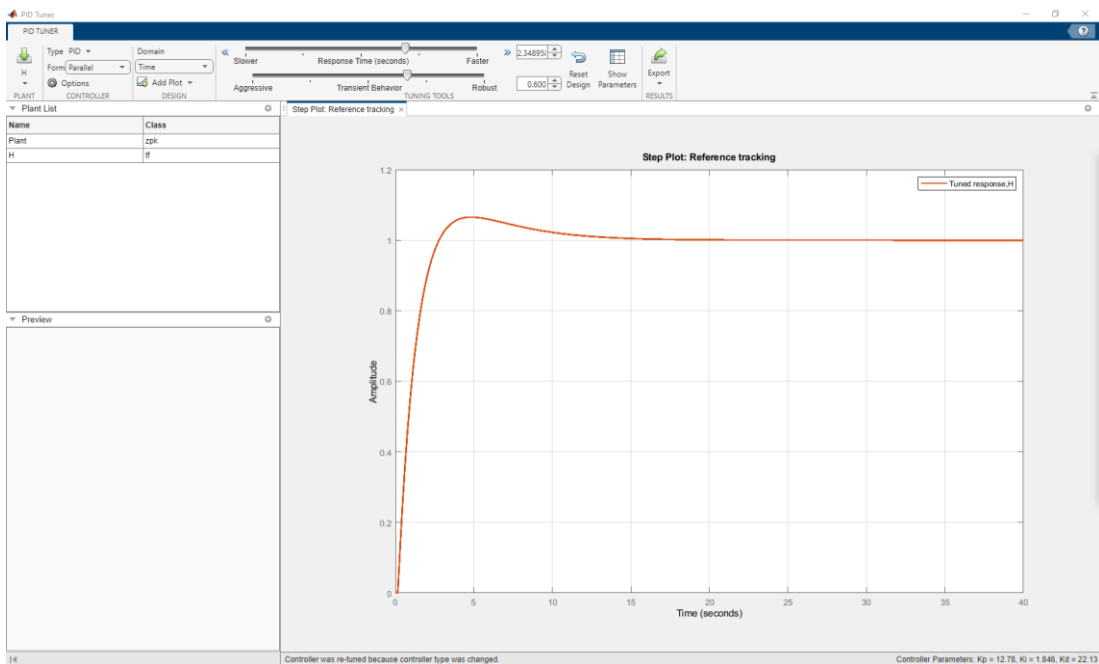




**Figura 3-19** : Herramienta Pid tuner de MATLAB

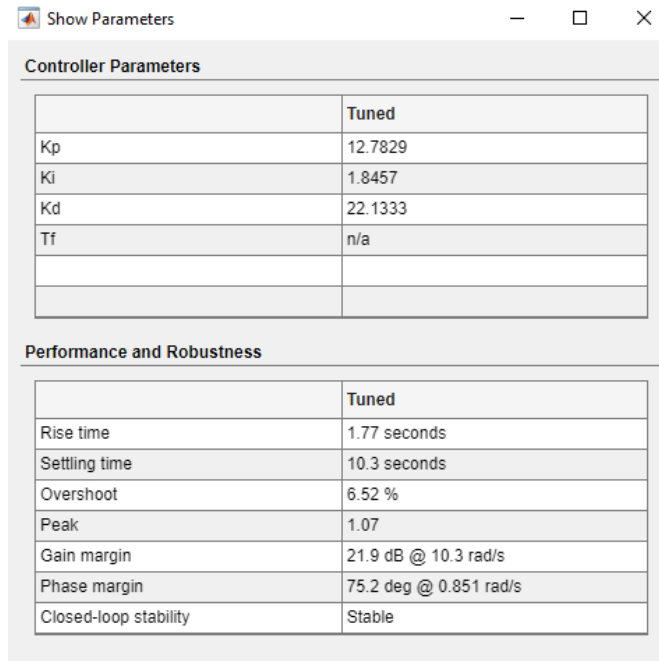
Fuente: (Moreira, Rodríguez)

En la herramienta pid tuner se configura el tiempo de respuesta de la señal a partir de la función de transferencia del sistema, también el valor que indicara si dicha respuesta es más rápida o lenta interviniendo en los valores de las ganancias como se muestra en la Figura 3-20 y en la Figura 3-21 se muestran los valores para el pid.

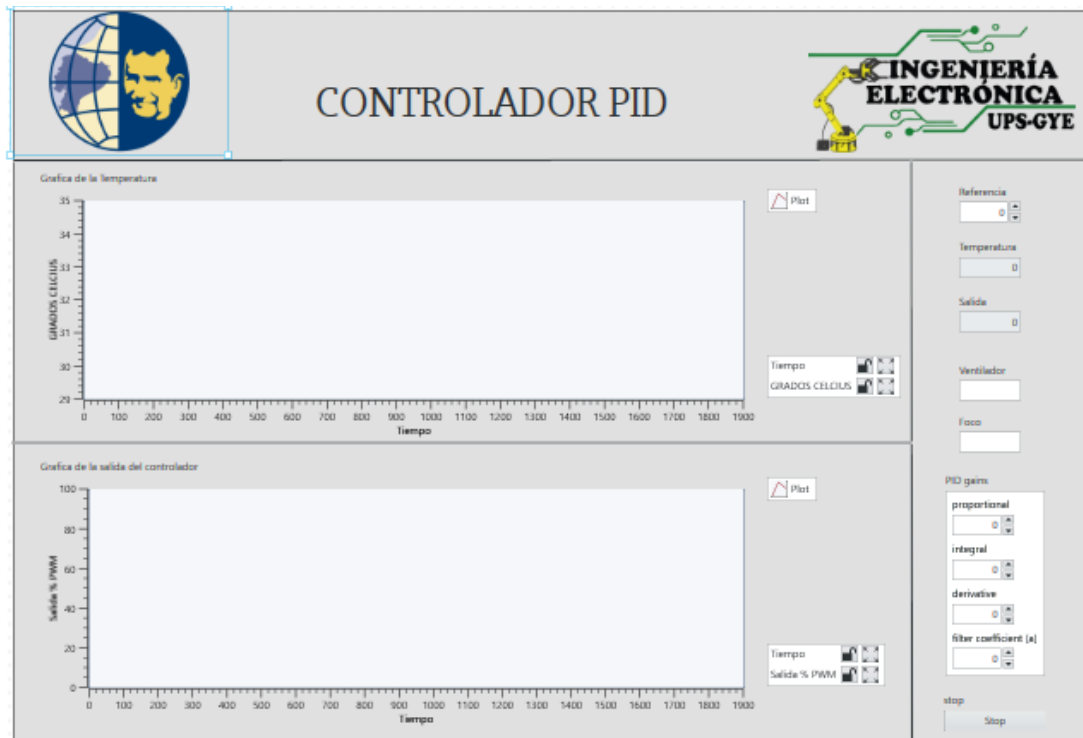


**Figura 3-20** : Herramienta Pid tuner de MATLAB – configuración

Fuente: (Moreira, Rodríguez)



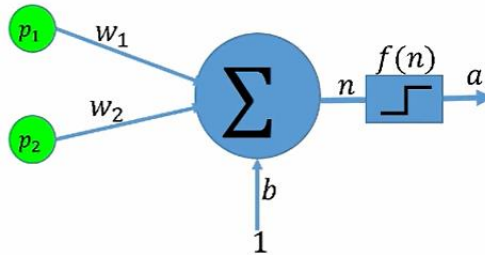
**Figura 3-21** : Ganancias del controlador PID a partir de la función de transferencia  
Fuente: (Moreira, Rodríguez)



**Figura 3-22** : Ganancias del controlador PID a partir de la función de transferencia  
Fuente: (Moreira, Rodríguez)

### 3.1.11. Entrenamiento de la red neuronal en Matlab

En el proceso de implementación del control neuronal se optó por la creación de una neurona entrada en Matlab tal como se muestra en la Figura 3-23 la estructura de la neurona artificial para obtener los valores de los pesos y las vías para LabVIEW nxg.



**Figura 3-23 :** Estructura de la neurona artificial a entrenar en Matlab

Fuente: (Moreira, Rodríguez)

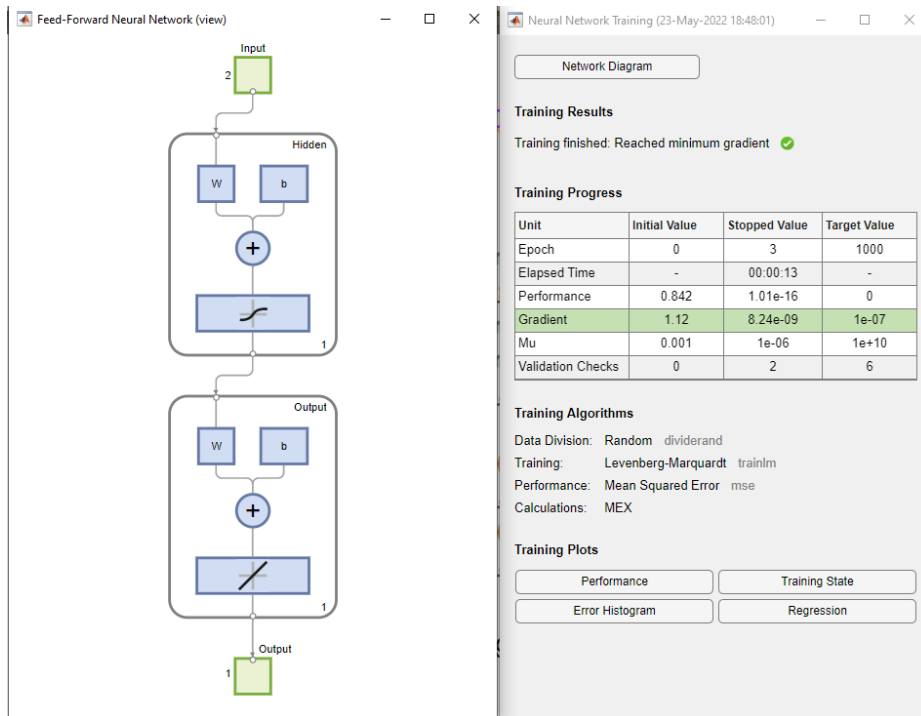
En el script de Matlab se crea los vectores de entrada y salida, se define las épocas, alcance mínimo del entrenamiento y se designa las variables donde se guardan los valores de los bias y los pesos como se observa en la figura 3-24.

```
clc, clear all,
p=[0 0 1 1 ;
  0 1 0 1];
t=[1 0 0 0];
net=newff(minmax(p),[2,1],('tansig','purelin'),'traincgf');
net=init(net);
net.trainParam.epoch=500;
net.trainParam.goal=1e-6;
net.train(net,p,t);
a=sim(net,p);
pesos1=net.iw(1,1)
bias1=net.b(1,1)
pesos2=net.lw(2,1)
bias2=net.b(2,1)
```

**Figura 3-24 :** Programa para el entrenamiento de la red neuronal en Matlab

Fuente: (Moreira, Rodríguez)

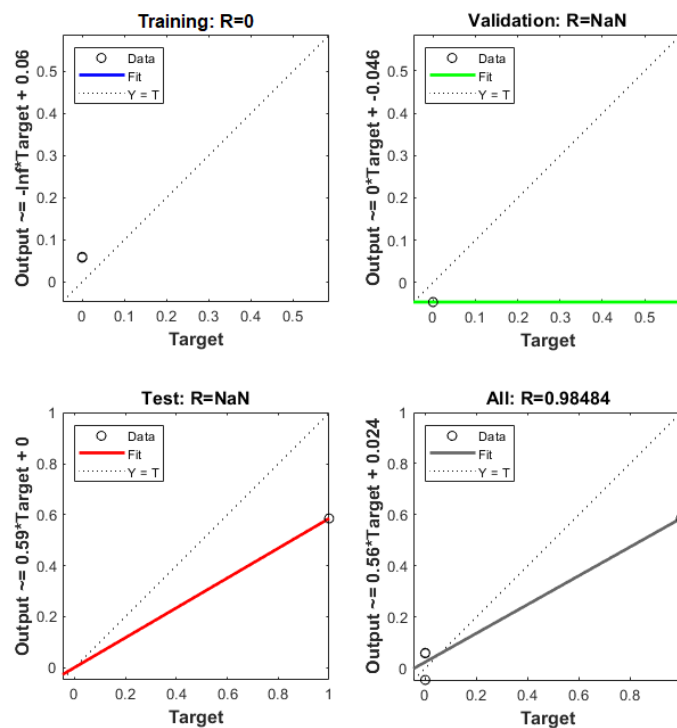
En la figura 3-25 se muestra el entrenamiento de la red neuronal y los resultados de los valores del entrenamiento.



**Figura 3-25 :** Resultados del entrenamiento en Matlab

Fuente: (Moreira, Rodríguez)

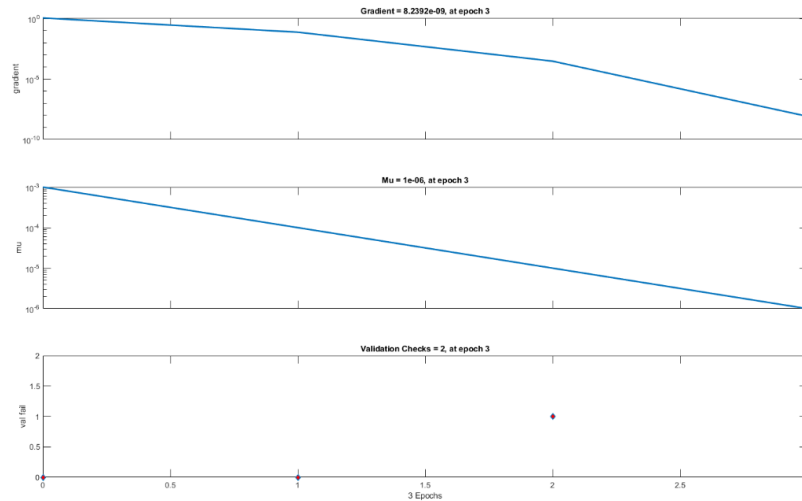
En la figura 3-26 el seguimiento de los valores de entrada hasta tener como meta el valor de 0.98.



**Figura 3-26 :** Seguimiento de los valores de entrada previo al entrenamiento

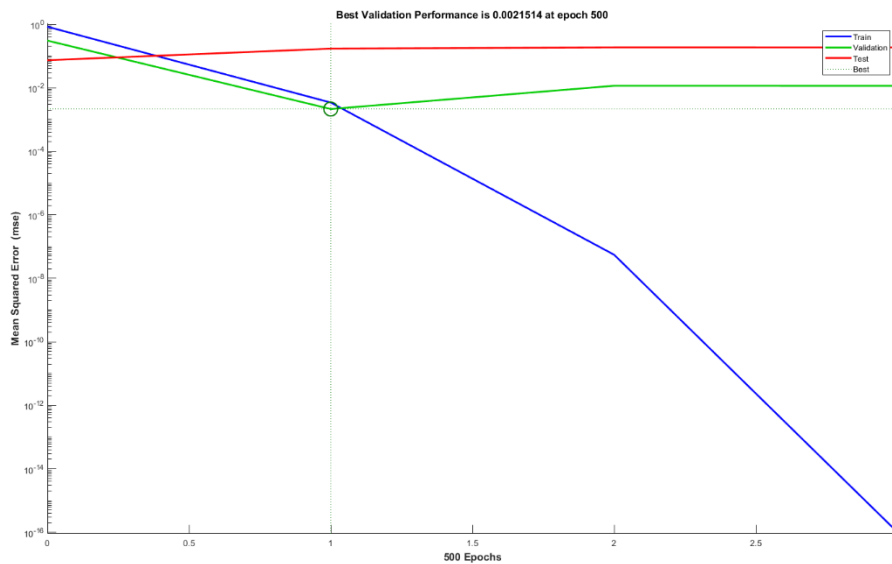
Fuente: (Moreira, Rodríguez)

En la figura 3-27 se detalla el número de entrenamiento a partir de las épocas teniendo como resultante el número total de entrenamientos para la red neuronal.

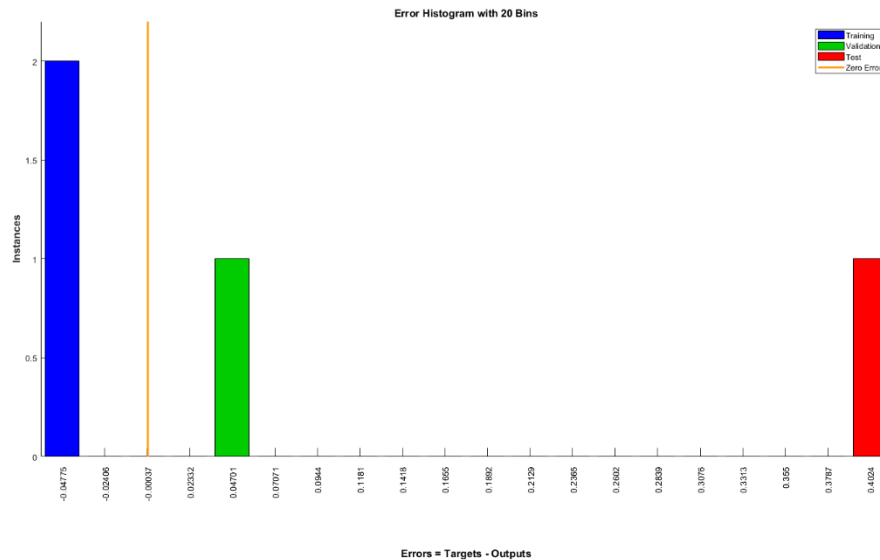


**Figura 3-27 :** Entrenamiento de la red neuronal y las épocas  
Fuente: (Moreira, Rodríguez)

En la figura 3-28 se muestra la validación del entrenamiento a partir del número de épocas máximo configurado.



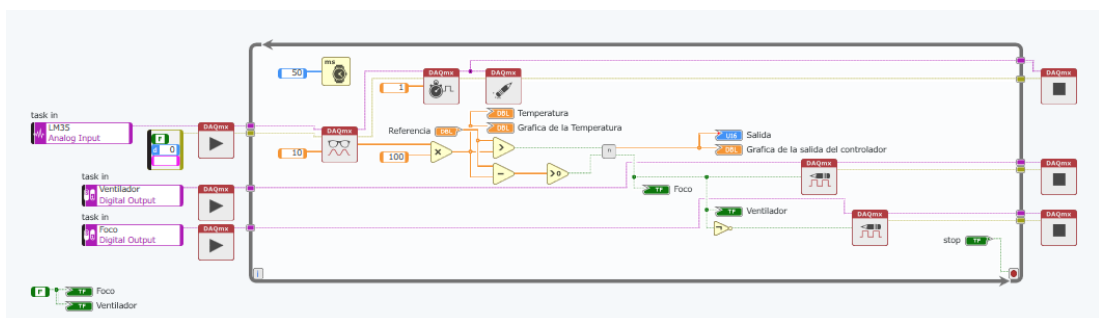
**Figura 3-28 :** Seguimiento de los valores de entrada previo al entrenamiento  
Fuente: (Moreira, Rodríguez)



**Figura 3-29** : Historiografía del error en la red neuronal  
Fuente: (Moreira, Rodríguez)

### 3.1.12. Implementación en LabVIEW NXG

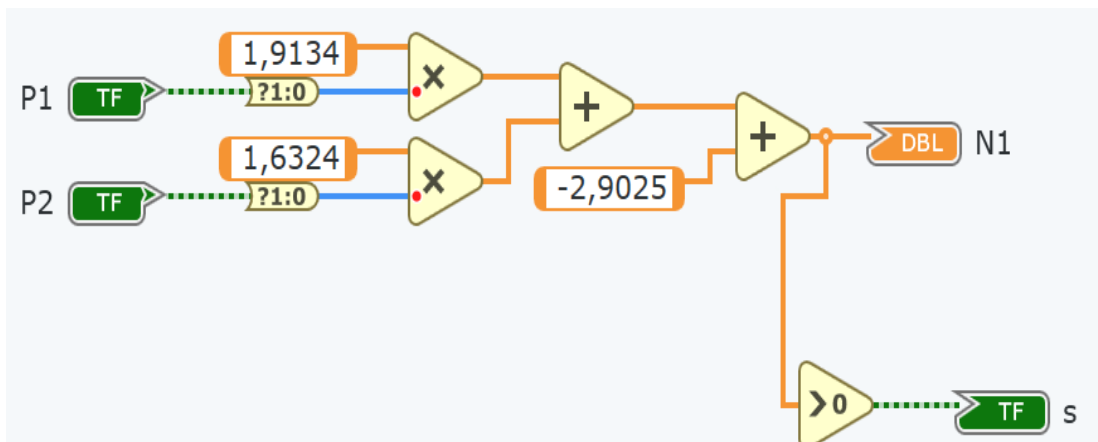
El método empleado en LabVIEW NXG para el desarrollo de un controlador mediante una red neuronal es declarar los objetos tecnológicos necesarios para las lecturas de las entradas tal como el sensor LM35 y el control de los periféricos mediante un relé de estado sólido para la fuente de luz por el foco microico y la parte de potencia con una tarjeta de puente de diodos para el motor del ventilador como se muestra en la figura 3-30 teniendo una función llamada n donde esta alojada la neurona.



**Figura 3-30** : Desarrollo del control neuronal en LabVIEW NXG  
Fuente: (Moreira, Rodríguez)

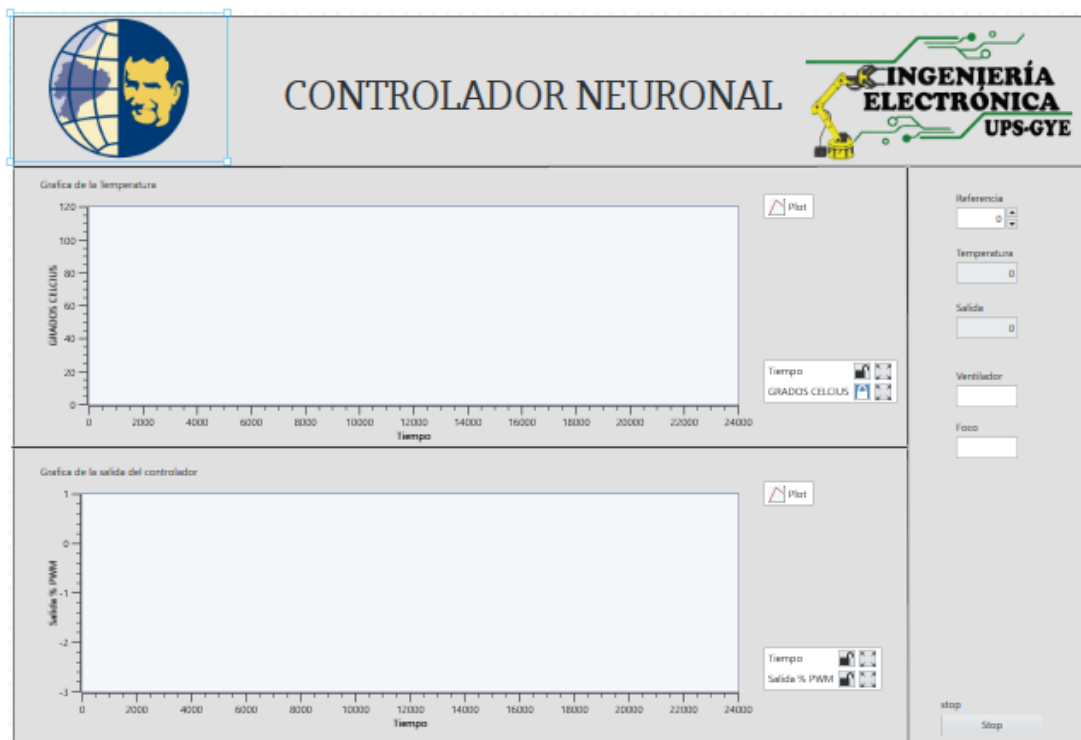
En la función donde se programa la neurona se utiliza la Ecuación (1), donde se toma los valores del mundo físico y mediante la codependencia de los valores de los pesos y las vías da como resultante al valor de la salida de la red como se muestra en la Figura 3-31.

$$n = w_1p_1 + w_2p_2 + b \quad (1)$$



**Figura 3-31** : Función en LabVIEW NXG para perceptrón

Fuente: (Moreira, Rodríguez)

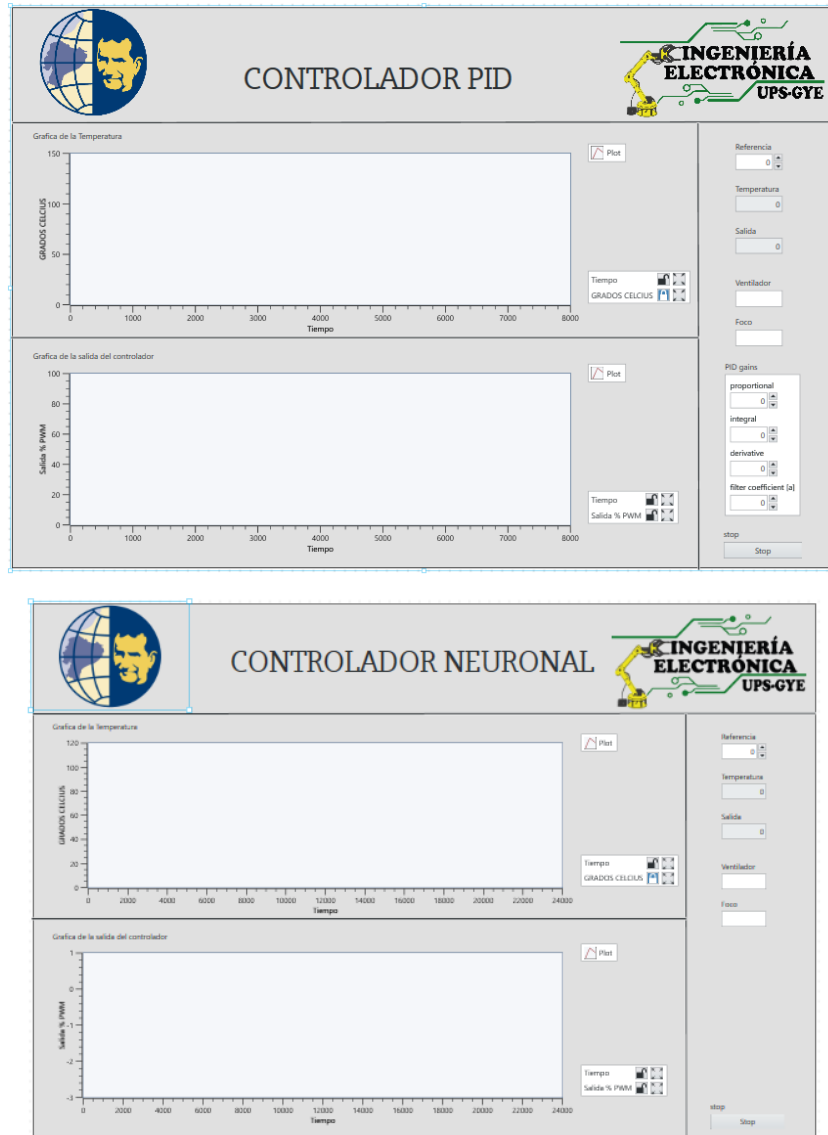


**Figura 3-32** : Implementación del control neuronal en LabVIEW NXG

Fuente: (Moreira, Rodríguez)

## 4. RESULTADOS

En este capítulo se demuestran los resultados al momento de la ejecución de cada controlador independiente uno del otro y posterior a la toma de datos una comparación entre ambos de la planta de procesos que se muestra en la Figura 4-1.



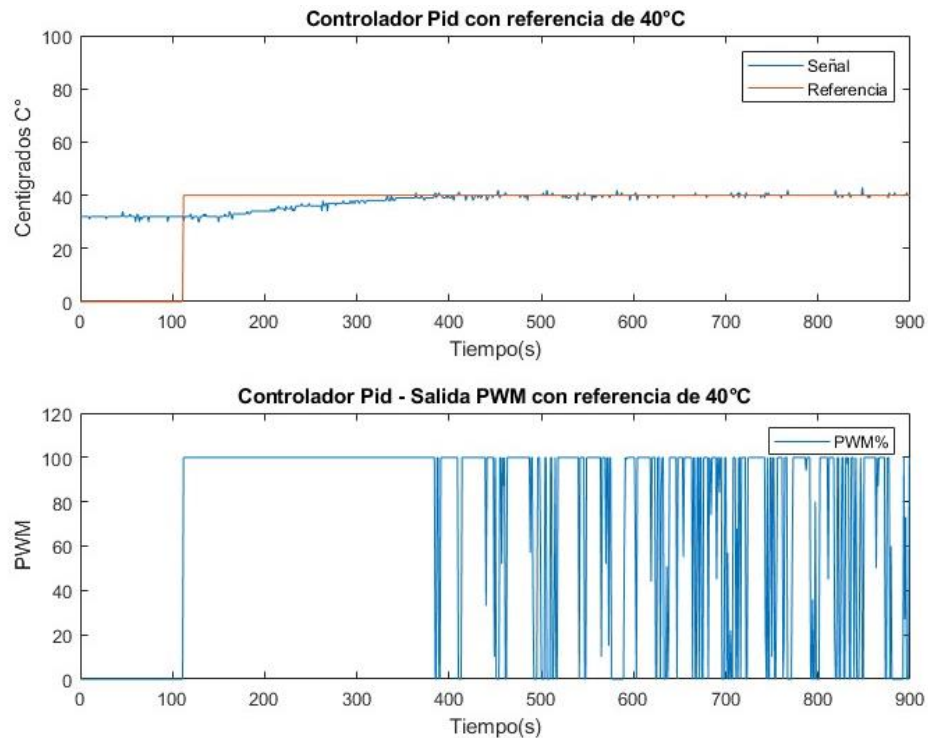
**Figura 4-1 :** Panel frontal del controlador PID y del controlador neuronal

Fuente: (Moreira, Rodríguez)



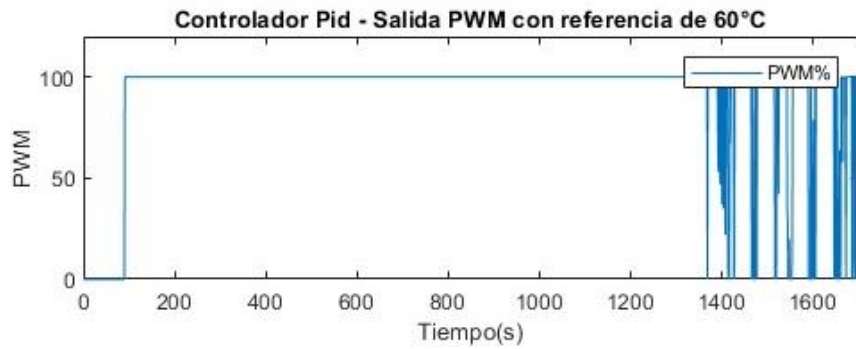
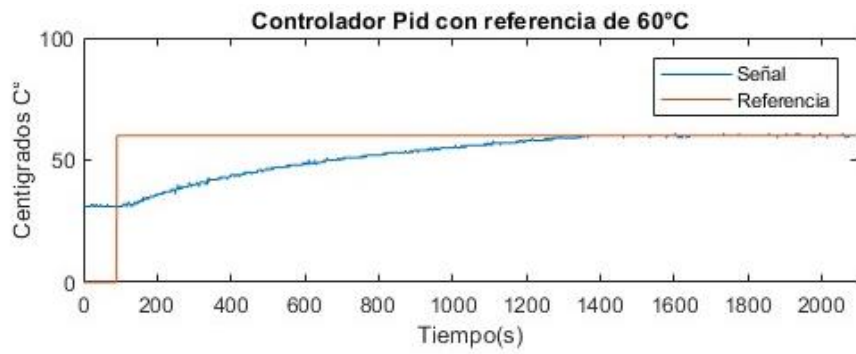
#### 4.1. RESULTADOS DEL CONTROLADOR PID

En el experimento se ha realizado pruebas con algunos puntos de referencias teniendo como datos los que se muestran en las Figura 4-2, 4-3 y 4-4 con su respectivo seguimiento del valor del sensor de temperatura y sus valores de referencia en 40, 60 y 100 grados Celsius, teniendo su respuesta de salida del controlador en cada una.



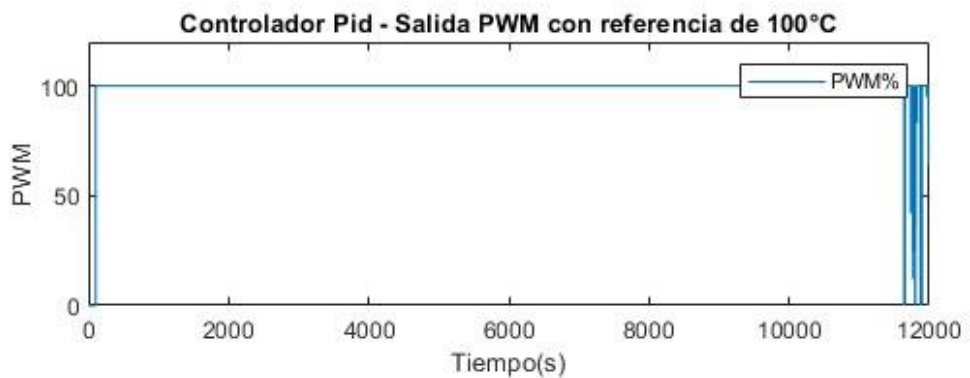
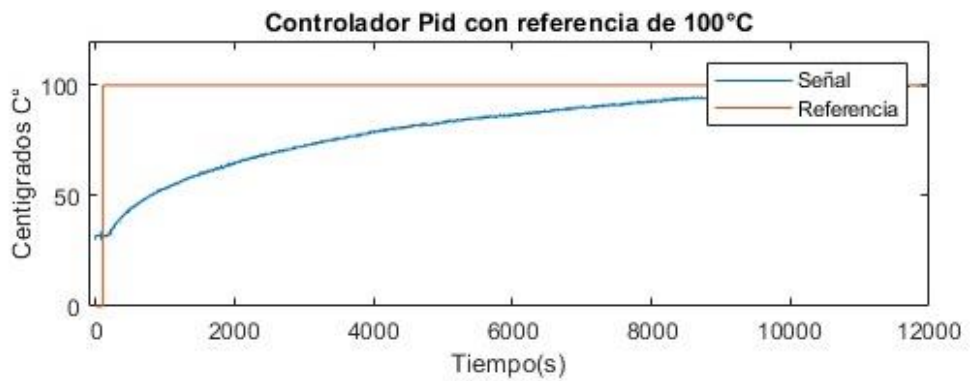
**Figura 4-2 :** Controlador Pid con referencia de 40°C

Fuente: (Moreira, Rodríguez)



**Figura 4-3 :** Controlador Pid con referencia de 60°C

Fuente: (Moreira, Rodríguez)

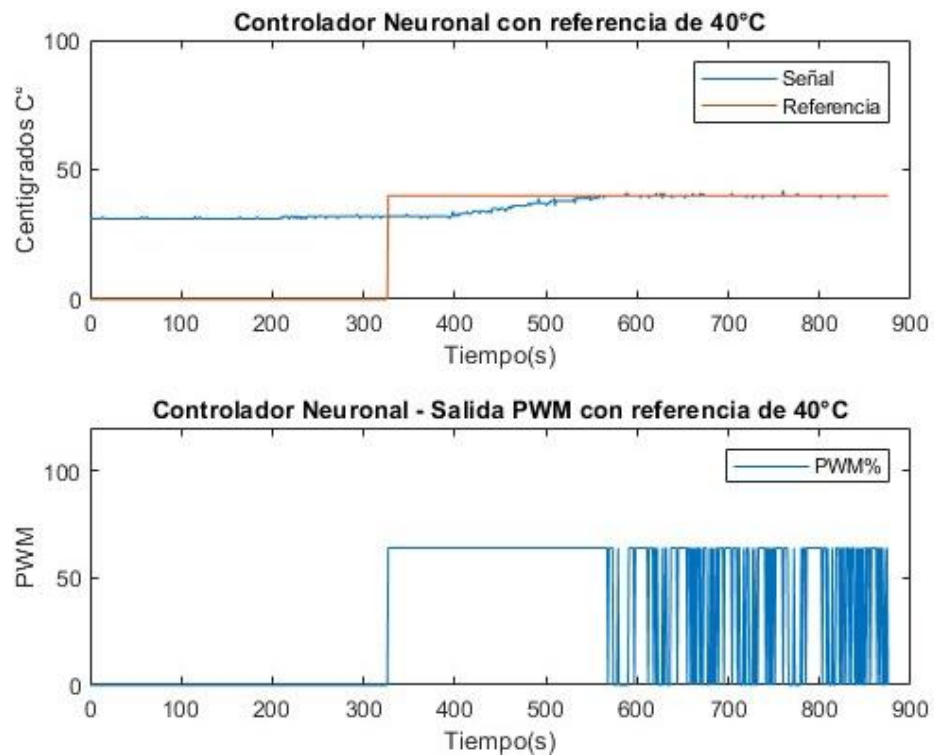


**Figura 4-4 :** Controlador Pid con referencia de 100°C

Fuente: (Moreira, Rodríguez)

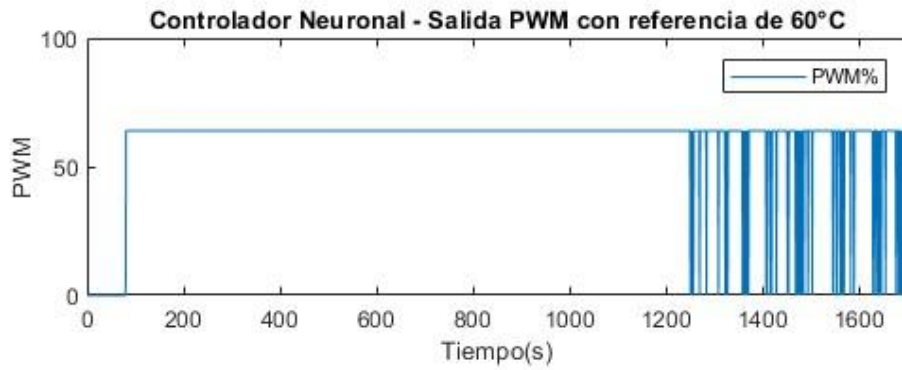
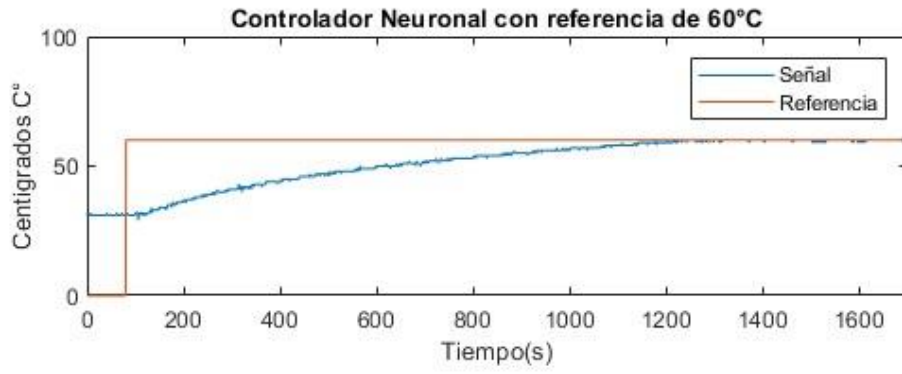
## 4.2. RESULTADOS DEL CONTROLADOR NEURONAL

En el experimento se ha realizado pruebas con algunos puntos de referencias teniendo como datos los que se muestran en las Figura 4-5, 4-6 y 4-7 con su respectivo seguimiento del valor del sensor de temperatura y sus valores de referencia en 40, 60 y 100 grados Celsius, teniendo su respuesta de salida del controlador en cada una.

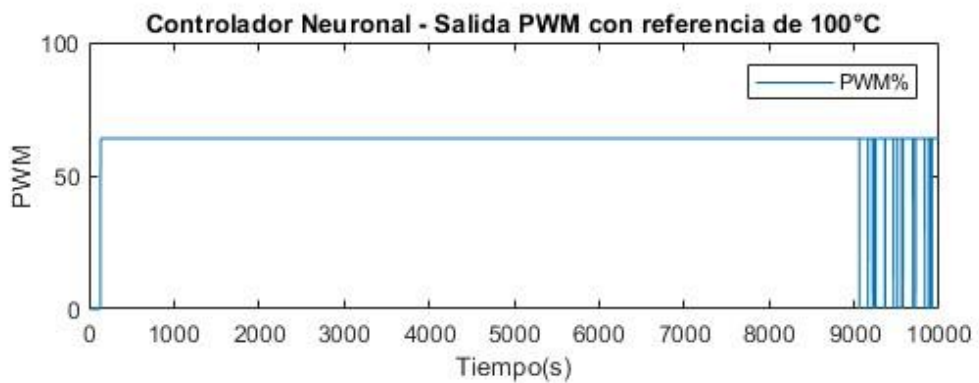
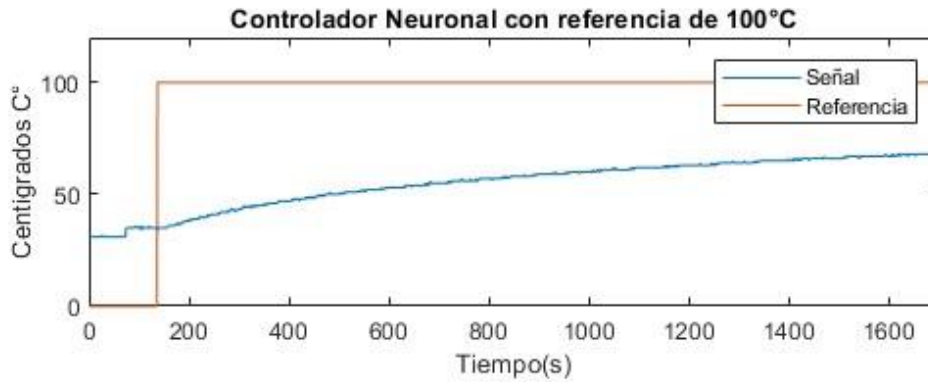


**Figura 4-5 :** Controlador neuronal con referencia de 40°C

Fuente: (Moreira, Rodríguez)



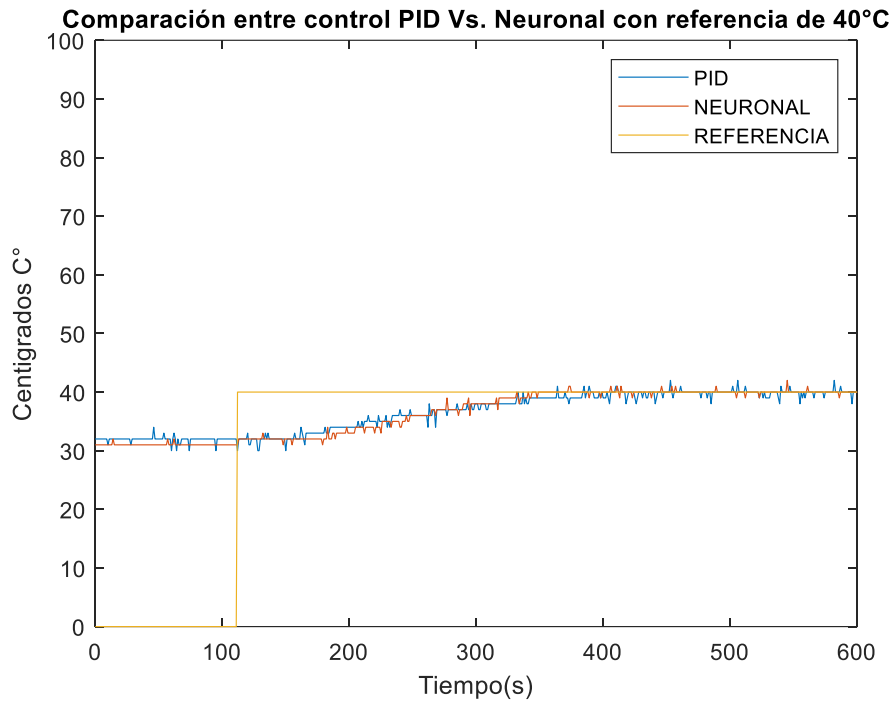
**Figura 4-6 :** Controlador neuronal con referencia de 60°C  
Fuente: (Moreira, Rodríguez)



**Figura 4-7 :** Controlador neuronal con referencia de 100°C  
Fuente: (Moreira, Rodríguez)

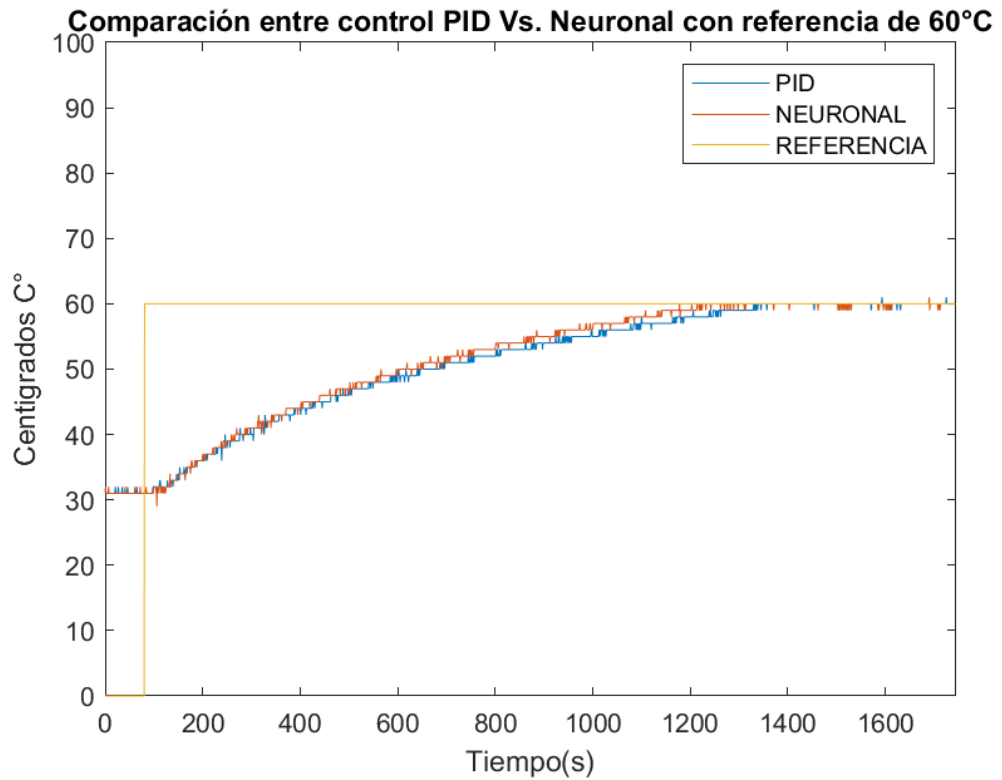
### 4.3. COMPARACIÓN DE CONTROLADORES

Los valores que se muestran en las siguientes Figuras 4-8, 4-9 y 4-10 es la sobreposición de los valores tomados en las pruebas con diferentes puntos de referencias utilizando un controlador PID y en otra ejecución un controlador con lógica difusa.

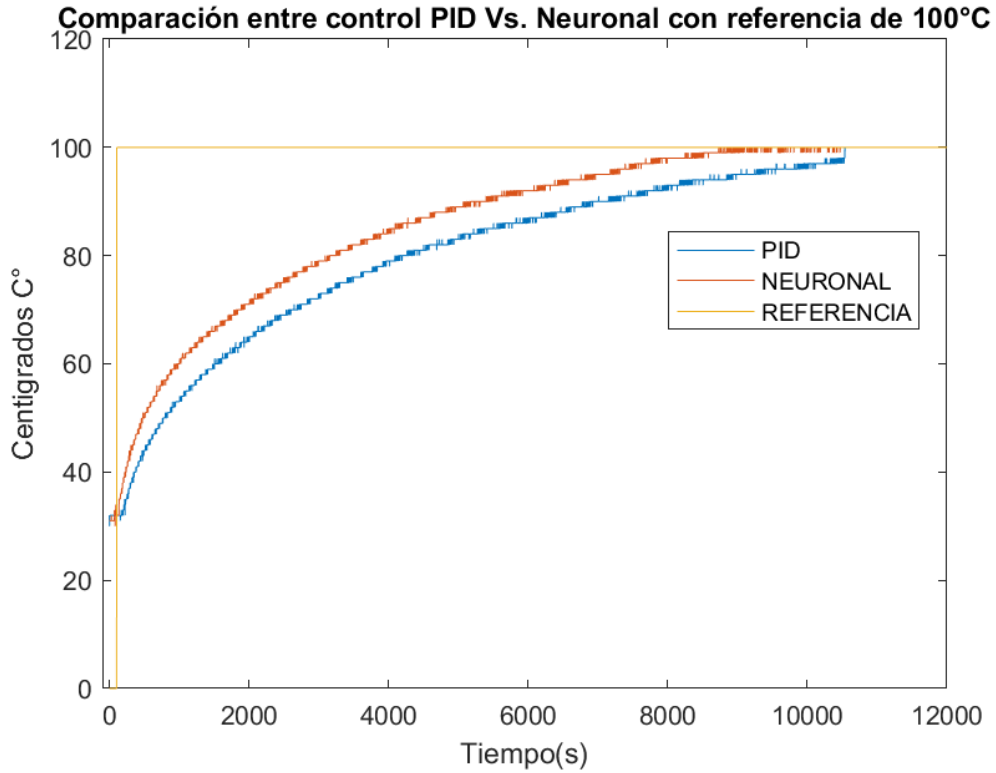


**Figura 4-8 :** Comparación de controladores PID y Neuronal con referencia de 40°C

Fuente: (Moreira, Rodríguez)



**Figura 4-9 :** Comparación de controladores PID y Neuronal con referencia de 60°C  
 Fuente: (Moreira, Rodríguez)



**Figura 4-10 :** Comparación de controladores PID y Neuronal con referencia de 100°C  
 Fuente: (Moreira, Rodríguez)

## CONCLUSIONES

- En el presente proyecto se desarrolló en detalle los pasos a seguir para la implementación de una red neuronal en la sección de marco metodológico teniendo como interfaz el uso de LabVIEW NXG y la herramienta de Matlab la cual se toma en cuenta el nivel de procesamiento de épocas para lograr el objetivo del entrenamiento.
- En la comparación de un controlador basado en redes neuronales y un control en lazo cerrado PID se toman los valores de entrada y salida, donde se adquiere los valores de temperatura mediante el uso de funciones permitiendo que la toma de datos sea precisa, teniendo como interfaz para el plóter de los datos la herramienta Matlab.
- En el diseño del módulo didáctico se utiliza como interfaz para la etapa de potencia un relé de estado sólido permitiendo la conmutación con el generador de calor o foco dicróico en corriente alterna, mediante el Daq6009.
- En el desarrollo del proyecto, se utilizó la comparación de los valores en Matlab, teniendo que el controlador basado en redes neuronales demostró menos atenuación de temperatura en contraste con el pid convencional.
- Para la comparación de los controladores se tiene en cuenta el almacenar datos y luego procesarlos en Matlab para un mejor análisis y observar mejor los resultados.

## RECOMENDACIONES

- Tomar en cuenta lo descrito en el presente documento respecto a las conexiones para su correcto funcionamiento.
- El programa LabVIEW Nxg necesita complementos para realizar la conexión con la DAQ6009.
- Trabajar con la versión del LabVIEW NXG 4.0 y la versión de MATLAB 2018b.
- Realizar comparaciones con un controlador neuro difuso y un control difuso, para la creación de documento que permitan publicaciones en revistas científicas.



## REFERENCIAS BIBLIOGRAFICAS

- Agüero, M. B. (2017). Obtenido de Sistema de adquisición de datos:  
<https://materias.df.uba.ar/mta2019c1/files/2014/08/Adquisicion-de-datos-Aguero.pdf>
- Arrow . (2015). Obtenido de DAS o DAQ: <https://www.arrow.com/es-mx/categories/data-acquisition/data-converters/data-acquisition-systems>
- Astrom, K. J. (1995). *PID Controllers: Theory, Design and Tuning*. ISA; Edición: Second Edition.
- Åström, K. J. (1995). *PID Controllers: Theory, Design, and Tuning. Second Edition. Research Triangle Park*. Tore: Instrument Society of America.
- Báez, P. G. (2016). Introducción a las Redes Neuronales y su aplicación a la Investigación Astrofísica. *Investigación Astrofísica*.
- Benavides, K. R. (2017). Locomoción de Robot Móvil.
- Hernández, A. I. (2010). REDES NEURONALES MULTIMODELO APLICADAS AL CONTROL DE SISTEMAS. *Sabadell*, 79. Universidad Autonoma de Barcelona.
- hetpro-store. (2017). Obtenido de <https://hetpro-store.com/TUTORIALES/wp-content/uploads/2016/05/lm35-img.png?x18372>
- Kang. (2000). MOSFET devices with polysilicon on single-layer. (págs. 35-38). IEEE Technical Digest. IEDM.
- Manuel Baltieri, C. L. (Marzo de 2018). <http://dx.doi.org/10.1101/284562>. Obtenido de University of Sussex:  
<https://www.biorxiv.org/content/biorxiv/early/2018/03/19/284562.full.pdf>
- Moreno Armendariz, A. S. (2001). Sistema de Información Científica. *Red de Revistas Científicas de América Latina, el Caribe, España y Portugal*, 13-32.
- Ni. (2019). *USB-6009*. Obtenido de <https://www.ni.com/es-cr/support/model.usb-6009.html>
- Paul, R. (2008). Temperature effects. *Neutron Physics*. Francia: EDP Sciences.

- Pulido, M. Á. (2000). *Convertidores de frecuencia, controladores de motores y SSR*. Marcombo.
- ROBÓTICA INDUSTRIAL. (2017). Obtenido de <https://www.interempresas.net/Robotica-industrial/Articulos/150824-Robotica-guiada-por-vision-Bin-Picking-una-solucion-para-los-procesos-productivos.html>
- Sánchez, L. F. (2016). Control cinemático y dinámico. *Laboratorio de Robótica Móvil y Sistemas Automatizados*.
- Texas Instruments. (2017). *LM35 Precision Centigrade Temperature Sensors*. Obtenido de Texas Instruments: <https://www.ti.com/lit/ds/symlink/lm35.pdf>
- Veloso, C. (2016). *FUNCIONAMIENTO DE UN SENSOR DE TEMPERATURA*. Obtenido de electrontools: <https://www.electrontools.com/Home/WP/funcionamiento-de-un-sensor-de-temperatura/>
- Vizcaíno, J. R. (2011). LabView. En *Entorno gráfico de programación* (págs. 3-10). Marcombo.

## ANEXOS

### Código para graficas en Matlab individuales del PID y Neuronal

```
clc, clear all, close all
load('matlab.mat')
%% FIG 1
figure(1)
ax1=nexttile;
y1=round(pid_in_40)
plot(ax1,x1,y1,x1,pid_ref_40)
xlim([0 900])
ylim([0 100])
title(ax1,'Controlador Pid con referencia de 40°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'Centigrados C°')
legend('Señal','Referencia')
% FIG 2
ax1=nexttile;
y2=round(pid_out_40)
plot(ax1,x1,y2)
xlim([0 900])
ylim([0 120])
title(ax1,'Controlador Pid - Salida PWM con referencia de 40°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'PWM')
legend('PWM%')

% FIG 5 ENTRADA DEL CONTROLADOR PID CON REFERENCIA DE 60
figure(5)
ax1=nexttile;
y5=round(pid_in_60)
plot(ax1,x3,y5,x3,pid_ref_60)
xlim([0 2100])
ylim([0 100])
title(ax1,'Controlador Pid con referencia de 60°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'Centigrados C°')
legend('Señal','Referencia')
% FIG 6 SALIDA DEL CONTROLADOR PID CON REFERENCIA DE 60

ax1=nexttile;
y6=round(pid_out_60)
plot(ax1,x3,y6)
xlim([0 1700])
ylim([0 120])
title(ax1,'Controlador Pid - Salida PWM con referencia de 60°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'PWM')
legend('PWM%')

% FIG 9 ENTRADA DEL CONTROLADOR PID CON REFERENCIA DE 100
figure(9)
ax1=nexttile;
y9=round(pid_in_100)
plot(ax1,x5,y9,x5,pid_ref_100)
xlim([-100 12000])
ylim([0 120])
title(ax1,'Controlador Pid con referencia de 100°C')
xlabel(ax1,'Tiempo(s)')
```

```

ylabel(ax1,'Centigrados C°')
legend('Señal','Referencia')

% FIG 6 SALIDA DEL CONTROLADOR PID CON REFERENCIA DE 100

ax1=nexttile;
y10=round(pid_out_100)
plot(ax1,x5,y10)
xlim([0 12000])
ylim([0 120])
title(ax1,'Controlador Pid - Salida PWM con referencia de 100°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'PWM')
legend('PWM%')

%% FIG 3
figure(3)
ax1=nexttile;
y3=round(ner_in_40)
plot(ax1,x2,y3,x2,ner_ref_40)
xlim([0 900])
ylim([0 100])
title(ax1,'Controlador Neuronal con referencia de 40°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'Centigrados C°')
legend('Señal','Referencia')
% FIG 4

ax1=nexttile;
y4=round(ner_out_40)
plot(ax1,x2,y4)
xlim([0 900])
ylim([0 120])
title(ax1,'Controlador Neuronal - Salida PWM con referencia de 40°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'PWM')
legend('PWM%')

%FIG 7
figure(7)
ax1=nexttile;
y7=round(ner_in_60)
plot(ax1,x4,y7,x4,ner_ref_60)
xlim([0 1700])
ylim([0 100])
title(ax1,'Controlador Neuronal con referencia de 60°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'Centigrados C°')
legend('Señal','Referencia')

%FIG 8
ax1=nexttile;
y8=round(ner_out_60)
plot(ax1,x4,y8)
xlim([0 1700])
ylim([0 100])
title(ax1,'Controlador Neuronal - Salida PWM con referencia de 60°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'PWM')

```

```
legend('PWM%')
```

```
%NEURONAL 100 FIG 11
```

```
figure(11)  
ax1=nexttile;  
y11=round(ner_in_100)  
plot(ax1,x6,y11,x6,ner_ref_100)  
xlim([0 1700])  
ylim([0 120])  
title(ax1,'Controlador Neuronal con referencia de 100°C')  
xlabel(ax1,'Tiempo(s)')  
ylabel(ax1,'Centigrados C°')  
legend('Señal', 'Referencia')
```

```
% NEURONAL 100 FIG 12
```

```
ax1=nexttile;  
y12=round(ner_out_100)  
plot(ax1,x6,y12)  
xlim([0 10000])  
ylim([0 100])  
title(ax1,'Controlador Neuronal - Salida PWM con referencia de 100°C')  
xlabel(ax1,'Tiempo(s)')  
ylabel(ax1,'PWM')  
legend('PWM%')
```

## Código para graficas en matlab comparación entre el PID y Neuronal

```
clc, clear all,close all
load('matlab.mat')
%% FIG 1
figure(1)
hold on
ax1=nexttile;
c_y1=round(c_p_40)
c_y2=round(c_n_40)
plot(ax1,c_x_40,c_y1,c_x_40,c_y2,c_x_40,c_r_40)
xlim([0 600])
ylim([0 100])
title(ax1,'Comparación entre control PID Vs. Neuronal con referencia de 40°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'Centigrados C°')
legend('PID','NEURONAL','REFERENCIA')

%% FIG 2
figure(2)
hold on
ax1=nexttile;
c_y3=round(c_p_60)
c_y4=round(c_n_60)
plot(ax1,c_x_60,c_y3,c_x_60,c_y4,c_x_60,c_r_60)
xlim([0 1745])
ylim([0 100])
title(ax1,'Comparación entre control PID Vs. Neuronal con referencia de 60°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'Centigrados C°')
legend('PID','NEURONAL','REFERENCIA')

%% FIG 3
figure(3)
hold on
ax1=nexttile;
c_y5=round(c_p_100)
c_y6=round(c_n_100)
plot(ax1,c_x_100,c_y5,c_x_100,c_y6,c_x_100,c_r_100)
xlim([-100 12000])
ylim([0 120])
title(ax1,'Comparación entre control PID Vs. Neuronal con referencia de 100°C')
xlabel(ax1,'Tiempo(s)')
ylabel(ax1,'Centigrados C°')
legend('PID','NEURONAL','REFERENCIA')
```

## **Enlace para descarga de documentos**

<https://mega.nz/folder/mIF3gKyA#IJU8fPHmd5luX0cJWfkuMA>