UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE INGENIERÍA ELECTRÓNICA

Trabajo de titulación previo a la obtención del título de Ingeniero Electrónico e Ingeniera Electrónica

.

PROYECTO TÉCNICO CON ENFOQUE GENERAL:

“DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE PROCESO TÉRMICO DIDÁCTICO ORIENTADO AL ANÁLISIS Y CONTROL DE SISTEMAS DINÁMICOS”

AUTORES:

DANIEL EDUARDO GRANDA ORTEGA

ANA PAULA JADÁN CAMPOVERDE

TUTOR:

ING. JULIO CÉSAR ZAMBRANO ABAD

CUENCA - ECUADOR

2020

CESIÓN DE DERECHOS DE AUTOR

Nosotros, Daniel Eduardo Granda Ortega con documento de identificación N° 0105871289 y Ana Paula Jadán Campoverde con documento de identificación N° 0105813869, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación: **"DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE PROCESO TÉRMICO DIDÁCTICO ORIENTADO AL ANÁLISIS Y CONTROL DE SISTEMAS DINÁMICOS"**, mismo que ha sido desarrollado para optar por el título de: *Ingeniero Electrónico e Ingeniera Electrónica*, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, noviembre del 2020

Daniel Eduardo Granda Ortega
C.I. 0105871289

Ana Paula Jadán Campoverde
C.I. 0105813869

CERTIFICACIÓN

Yo, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **"DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE PROCESO TÉRMICO ORIENTADO AL ANÁLISIS Y CONTROL DE SISTEMAS DINÁMICOS"**, realizado por Daniel Eduardo Granda Ortega y Ana Paula Jadán Campoverde, obteniendo el *Proyecto Técnico con enfoque general*, que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, noviembre del 2020

Ing. Julio César Zambrano Abad

C.I. 0301489696

DECLARATORIA DE RESPONSABILIDAD

Nosotros, Daniel Eduardo Granda Ortega con documento de identificación N° 0105871289 y Ana Paula Jadán Campoverde con documento de identificación N° 0105813869, autores del trabajo de titulación: **"DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE PROCESO TÉRMICO ORIENTADO AL ANÁLISIS Y CONTROL DE SISTEMAS DINÁMICOS"**, certificamos que el total contenido del *Proyecto Técnico con enfoque general,* es de nuestra exclusiva responsabilidad y autoría

Cuenca, noviembre del 2020

Daniel Eduardo Granda Ortega
C.I. 0105871289

Ana Paula Jadán Campoverde
C.I. 0105813869

ÍNDICE GENERAL

ÍNDICE GENERAL I

ÍNDICE DE FIGURAS III

ÍNDICE DE FIGURAS VII

AGRADECIMIENTOS VIII

DEDICATORIA IX

RESUMEN X

INTRODUCCIÓN XI

ANTECEDENTES DEL PROBLEMA DE ESTUDIO XII

JUSTIFICACIÓN XIII

OBJETIVOS XIV

OBJETIVO GENERAL XIV
OBJETIVOS ESPECÍFICOS XIV

CAPÍTULO 1: DESCRIPCIÓN DE PROCESO TÉRMICO 15

1.1 ARQUITECTURA GENERAL DEL PROCESO 15
1. ELEMENTO CONTROLADOR 15
2. INTERFAZ DE 0-10 VOLTIOS 15
3. PROCESO TÉRMICO 16
a. Elemento calefactor *16*
b. Sensor de Temperatura *17*
c. Transmisor *17*
d. Ventilador *18*
1.2 DISEÑO DEL PROTOTIPO DE PROCESO TÉRMICO 18
1.3 DISEÑO DEL CIRCUITO IMPRESO 22

CAPÍTULO 2: IDENTIFICACIÓN DEL PROCESO 29

1.1 INSTRUMENTO VIRTUAL PARA LA IDENTIFICACIÓN 29
1.1.1 Identificación del sistema *31*
1.2 DETECCIÓN DE LA HISTÉRESIS. 36

CAPÍTULO 3: ESQUEMAS DE CONTROL 38

1.3 CONTROLADOR PID 38
1.3.1 PID Tuner *39*
1.3.2 Implementación de un controlador PID con LabVIEW *42*
1.3.3 Implementación de un controlador PID con módulo TK4S-T4RC *48*
1.3.4 PID con auto sintonía utilizando un PLC S7-1200 ..... *70*

CAPÍTULO 4: CONTROLADOR EN ESPACIO DE ESTADOS 82

2.1 CONTROLADOR EN ESPACIO DE ESTADOS POR EL MÉTODO DE ASIGNACIÓN DE POLOS 82

2.1.1 Implementación de un controlador en espacio de estados por asignación de polos *..... 83*

CAPÍTULO 5: CONTROL PREDICTIVO BASADO EN MODELO 89

3.1 CONTROL PREDICTIVO BASADO EN MODELO (MPC) 89

3.1.1 Control Predictivo basado en el Modelo (MPC) implementado en Simulink con NI myDAQ *..... 89*

3.1.2 Controlador Predictivo basado en el Modelo MPC implementado en Simulink mediante un PLC S7-1200 *..... 95*

CAPÍTULO 6: CONCLUSIONES Y RECOMENDACIONES 99

REFERENCIAS BIBLIOGRÁFICAS 101

APÉNDICES 106

APÉNDICE A: GUÍAS DE PRÁCTICAS DE LABORATORIO PARA EL ANÁLISIS Y CONTROL DEL PROCESO TÉRMICO 106

- ***GUÍA DE PRÁCTICA 1: ADQUISICIÓN DE DATOS Y ANÁLISIS DE HISTÉRESIS DE LA PLANTA*** *..... 106*
- ***GUÍA DE PRÁCTICA 2: IDENTIFICACIÓN DE LA PLANTA*** *..... 138*
- ***GUÍA DE PRÁCTICA 3: DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR PID USANDO UN COMPUTADOR CON LABVIEW Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA*** *..... 173*
- ***GUÍA DE PRÁCTICA 4: IMPLEMENTACIÓN DE UN CONTROLADOR PID CON AUTO SINTONÍA UTILIZANDO UN PLC S7-1200 CON MONITOREO EN TIEMPO REAL EN LABVIEW USANDO EL PROTOCOLO OPC*** *..... 199*
- ***GUÍA DE PRÁCTICA 5: IMPLEMENTACIÓN DE UN CONTROLADOR PID AUTO SINTONIZADO USANDO EL MÓDULO TK4S-T4RC Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA*** *..... 239*
- ***GUÍA DE PRÁCTICA 6: DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR EN ESPACIO DE ESTADOS UTILIZANDO UNA NI myDAQ Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA*** *..... 269*
- ***GUÍA DE PRÁCTICA 7: CONTROLADOR PREDICTIVO BASADO EN EL MODELO (MPC) IMPLEMENTADO EN EL EQUIPO DE ENSAYO DE CONTROL DE TEMPERATURA MEDIANTE SIMULINK Y NI myDAQ.*** *..... 287*
- ***GUÍA DE PRÁCTICA 8: CONTROLADOR PREDICTIVO BASADO EN EL MODELO (MPC) IMPLEMENTADO EN EL EQUIPO DE ENSAYO DE CONTROL DE TEMPERATURA MEDIANTE SIMULINK Y PLC.*** *..... 307*

APÉNDICE B: PLANOS DEL PROTOTIPO 338

ÍNDICE DE FIGURAS

Figura 1. Arquitectura del proceso 15
Figura 2. Diagrama de bloques del proceso 16
Figura 3. Elemento calefactor [5] 16
Figura 4. PT100 de 3 hilos [13] 17
Figura 5.Trasmisor JUMO [15] 18
Figura 6. Ventilador [16] 18
Figura 7. Vista frontal del prototipo 19
Figura 8. Vista lateral del prototipo 19
Figura 9. Vista posterior del prototipo 20
Figura 10. Vista Superior del prototipo 20
Figura 11. Vista isométrica del prototipo 21
Figura 12. Esquemático del circuito principal 23
Figura 13. Esquemático del circuito de entrada de señales 25
Figura 14. Esquemático del circuito de alimentación 25
Figura 15. Esquemático del circuito de potencia 26
Figura 16. Circuito impreso 28
Figura 17. Modelo 3D 28
Figura 18. Diagrama de bloques de la adquisición de datos para la identificación de la planta 30
Figura 19. Panel frontal de la adquisición de datos para la identificación de la planta 30
Figura 20. Datos obtenidos para la identificación de la planta 31
Figura 21. Obtención de datos para la identificación 31
Figura 22. Datos para identificación sin offset de la temperatura de la cámara 32
Figura 23. Tiempo muerto 33
Figura 24. Sistemas de primer, segundo, tercer y cuarto orden. 33
Figura 25. Poner solo el mapa de polos y ceros 34
Figura 26. Sistemas de primer, segundo, tercer y cuarto orden con retardo de 17 segundos. 35
Figura 27. Respuesta del Sistema 36
Figura 28.Histeresis de la temperatura en la cámara 37
Figura 29. Esquema de controlador PID 39
Figura 30. Respuesta al escalón 40
Figura 31. Ventana PID Tuner 41
Figura 32. Respuesta del sistema con controlador PID sintonizado 41
Figura 33. Bloque PID LabVIEW 43
Figura 34. VI controlador PID 43
Figura 35. Configuración del bloque PID 44
Figura 36. Configuración bloque de entrada analógica 45
Figura 37. Configuración de salida analógica 45
Figura 38. Interfaz gráfica controlador PID LabView 46
Figura 39. Ensayo de control de temperatura usando un controlador PID implementado en LabVIEW 47
Figura 40. Repuesta del sistema con cambio de referencia de 50 a 55 grados Celsius 47
Figura 41. Módulo Autonics TK4S-T4RC 48

Figura 42. Restaurar valores por defecto 48
Figura 43. Grupo de parámetros 1 49
Figura 44. Grupo de parámetros 3 49
Figura 45. Menú de opciones de entrada 49
Figura 46. Límite inferior de la entrada analógica 50
Figura 47. Límite superior de la entrada analógica 50
Figura 48. Ubicación del punto decimal 50
Figura 49. Límite inferior de la escala de la entrada 51
Figura 50. Límite superior de la escala de la entrada 51
Figura 51. Unidades mostradas en el indicador 51
Figura 52. Comportamiento de la salida del dispositivo 52
Figura 53. Grupo de parámetros 5 52
Figura 54. Nivel de usuario 53
Figura 55. Grupo de parámetros 4 53
Figura 56. Comportamiento de alarma 1 53
Figura 57. Usar salida 2 como salida de corriente 54
Figura 58. Grupo de parámetros 1 54
Figura 59. Pantalla principal 54
Figura 60. Establecer una referencia 55
Figura 61. Grupo de parámetros 2 55
Figura 62. Activar autotuning 55
Figura 63. Pantalla principal con autotuning activado 56
Figura 64. Ventana de inicio LabVIEW 2019 56
Figura 65. Crear nuevo proyecto en blanco 57
Figura 66. Agregar un dispositivo 57
Figura 67. Agregar myRIO-1900 58
Figura 68. Agregar un VI 58
Figura 69. Estructura while 59
Figura 70. Acceder al buscador de LabVIEW 59
Figura 71. Configuración de entradas analógicas 60
Figura 72. Configuración del bloque "Set Dynamic Data Atributes" 60
Figura 73. Conexión de bloques "Set Dynamic Data Atributes" 61
Figura 74. Agregar el bloque "Write to measurement file" 61
Figura 75. Selección de dirección de archivo de datos 62
Figura 76. Nombre del archivo de datos 62
Figura 77. Configuración del bloque "Write to measurement file" 63
Figura 78. Conexión de secciones del programa 63
Figura 79. Agregar un "Waveform Chart" 64
Figura 80. Conexión del bloque "Waveform Chart" 64
Figura 81. Diagrama de conexión 65
Figura 82. Ventana de inicio de NI myRIO 65
Figura 83. Propiedades de NI myRIO 66
Figura 84. Asignar dirección IP 66
Figura 85. Ejecutar código 67
Figura 86. Ingresar dirección IP 67
Figura 87. Ingresar credenciales 67
Figura 88. Importar datos 68

Figura 89. Configurar parámetros ____ *68*
Figura 90. Variables en el workspace ____ *69*
Figura 91.Ensayo de control de temperatura usando un controlador PID implementado en módulo TK4S-T4RC de Autonics ____ *69*
Figura 92. Nuevo bloque ____ *71*
Figura 93. Configuración del bloque nuevo ____ *71*
Figura 94. Añadiendo nuevo objeto ____ *72*
Figura 95. Ventana de configuración del objeto ____ *72*
Figura 96. Selección del tipo de controlador ____ *73*
Figura 97. Límite del valor de proceso ____ *73*
Figura 98. Escala del valor de proceso ____ *74*
Figura 99. Agregando bloque PID_Compact ____ *74*
Figura 100. Nombre del bloque PID_Compact ____ *75*
Figura 101. Bloque PID_Compact ____ *75*
Figura 102. Bloques de variables para el monitoreo ____ *76*
Figura 103. Puesta en marcha ____ *77*
Figura 104. Ventana para auto sintonización del bloque PID_Compact ____ *77*
Figura 105. Inicio de lectura de variables ____ *77*
Figura 106. Inicio de autotuning ____ *77*
Figura 107. Parámetros de la auto sintonía ____ *78*
Figura 108. Variables creadas para el monitoreo y recolección de datos ____ *78*
Figura 109. Configuración de las variables según su función ____ *79*
Figura 110. Programa para monitoreo y adquisición de datos. ____ *79*
Figura 111. Panel de control ____ *80*
Figura 112. Ensayo de control de temperatura con PLC S7-1200 y controlador PID ____ *80*
Figura 113. Esquema general de un controlador en espacio de estados con observador e integrador embebido ____ *82*
Figura 114. Esquema de asignación de polos por realimentación de estados con integrador embebido ____ *86*
Figura 115. Esquema de asignación de polos por realimentación de estados con integrador embebido con entradas y salidas físicas. ____ *86*
Figura 116. Configurar bloque Real-Time Synchronization ____ *87*
Figura 117. Ensayo de control de temperatura con controlador en espacio de estados ____ *87*
Figura 118. Representación del control predictivo [16] ____ *89*
Figura 119. Importando datos para diseño del controlador ____ *90*
Figura 120. Selección de datos para el diseño del controlador ____ *91*
Figura 121. Planta ingresada ____ *91*
Figura 122. Tiempo de simulación ____ *92*
Figura 123. Condiciones iniciales del controlador ____ *92*
Figura 124. Parámetros para la estabilidad del sistema ____ *92*
Figura 125. Restricciones para el diseño del controlador ____ *93*
Figura 126. Performance Tuning ____ *93*
Figura 127. Diseño del controlador MPC ____ *93*
Figura 128. Generando Modelo de Simulink ____ *94*
Figura 129. Modelo Generado con la herramienta MPC Designer ____ *94*
Figura 130. Programa generado para la implantación del controlador MPC ____ *94*
Figura 131. Implementación del controlador MPC mediante NI-DAQ ____ *95*

Figura 132. Configuración de bloques _____ *96*
Figura 133. Configuración del bloque "OPC Read" _____ *97*
Figura 134. Configuración del bloque "OPC Write" _____ *97*
Figura 135. Modelo de Simulink del controlador MPC implementado con PLC S7-1200 *98*
Figura 136. Respuesta del prototipo al implementar un controlador MPC con PLC S7-1200 _____ *98*
Figura 16137. Configuración del bloque de entradas analógico _____ *280*

ÍNDICE DE FIGURAS

Tabla 1. Lista de componentes. _____ *22*
Tabla 2. Separación mínima entre pistas de un circuito impreso [10] _____ *27*
Tabla 3. Respuesta del sistema con entrada escalonada _____ *37*

AGRADECIMIENTOS

Agradezco a mi padre por ser mi ejemplo de trabajo constante y generosidad. A mi madre por siempre brindarme calidez y ser ejemplo de fe. A mis hermanas por siempre contar con su apoyo y consejos. A mis sobrinos por ser uno de los motivos de mi inspiración. A mis abuelitos por siempre ver por el bienestar de la familia. A mi persona especial por todos los momentos que hemos podido aprender y crecer juntos a lo largo de esta experiencia universitaria.

Agradezco a nuestro tutor el Ing. Julio Zambrano por brindarnos su tiempo al orientarnos en este proyecto.

Ana Paula Jadán Campoverde

DEDICATORIA

Dedico este proyecto a mis abuelitos Mamá y Papá por ser un pilar fundamental en mi vida.

Ana Paula Jadán Campoverde

RESUMEN

En el siguiente trabajo de titulación se describe el diseño, la construcción, la identificación y el control de un prototipo de proceso térmico destinado a realizar prácticas de control automático a nivel de ingeniería. El prototipo se diseña con un enfoque didáctico teniendo en consideración la flexibilidad como una característica fundamental. Bajo este sentido el usuario podrá interactuar con el prototipo utilizando diversas plataformas de instrumentación y control.

Los materiales de construcción en su mayoría son translucidos de tal forma que el estudiante pueda observar sin restricciones la estructura interna y el funcionamiento del prototipo. Para la medición de temperatura se ha seleccionado un sensor RTD PT100 y un trasmisor JUMO industrial, mientras que para la actuación se utiliza una niquelina comandada por un sistema electrónico de potencia.

Producto de este trabajo de titulación se ha generado un conjunto de guías de práctica para que el estudiante pueda usar de manera adecuada el prototipo y pueda experimentar sobre tópicos de control que normalmente se analizan en el aula a nivel teórico. El trabajo de titulación incluye una propuesta de modelado a través de técnicas computacionales conocidas como identificación de sistemas. Posteriormente se plantean algunas propuestas de control que incluyen PIDs, espacio de estados y Control Predictivo basado en Modelo.

Los esquemas de control fueron implementados buscando dar una mayor diversidad en cuanto al dominio de ciertas plataformas. Por ejemplo, en el caso de los PIDs, se planteó diversas propuestas de implementación utilizando un PLC un módulo industrial TK4S-T4RC y un computador personal bajo el programa informático de Labview. En complemento para las otras propuestas de control se utilizó Simulink de Matlab. Para todos los sistemas, en aras de verificar el desempeño de los controladores se implementó un interfaz humano máquina utilizando un sistema de adquisición NI myDAQ y para el caso de los controladores sobre PLC se utilizó una comunicación OPC.

INTRODUCCIÓN

El control de temperatura tiene un papel muy importante en muchas industrias, puesto que permite que muchos de los procesos se lleven a cabo con total éxito.

El conocimiento de diversos controladores tanto teórico como práctico nos da la ventaja en el ámbito laboral, por ello es necesario adquirir este conocimiento durante nuestra vida universitaria. Se adquiere este conocimiento al realizar trabajos prácticos en equipos que reflejen situaciones reales en la industria.

En el presente trabajo de titulación se describe la elaboración del prototipo de proceso térmico didáctico y guías de laboratorio con las cuales los estudiantes puedan observar el comportamiento de la temperatura ante diversos controladores.

En el capítulo 1 se presenta una arquitectura general del proceso, así como los elementos utilizados para el proceso térmico (elemento calefactor, sensor de temperatura, transductor y ventilador); además, se describe el diseño y las características de construcción del prototipo y la etapa de potencia.

En el capítulo 2 se describe el proceso de recolección de datos que son útiles para la para la identificación del sistema; se describe el procedimiento llevado a cabo para la identificación de la planta del prototipo en el programa informático Matlab; por último, se realiza un análisis de la histéresis del sistema.

En el capítulo 3 se expone el controlador PID y sus diferentes propuestas utilizando diversas plataformas e interfaces. Se detalla el uso Matlab para el diseño de uno de los controladores; además, se hace uso de la auto sintonización de controladores PID; por último, se compara la respuestas de los controladores al implementarlos en el prototipo.

En el capítulo 4 se desarrolla el diseño de un controlador en espacio de estados y se describe su implementación en el prototipo utilizando Simulink de Matlab y una NI my DAQ.

En el capítulo 5 se describe el Control Predictivo basado en Modelo. Se detalla el diseño del controlador en la herramienta MPC Designer de Matlab y su implementación con Simulink utilizando dos interfaces una NI myDAQ y un PLC S7-1200.

ANTECEDENTES DEL PROBLEMA DE ESTUDIO

El estudio de los sistemas de control es de gran importancia en la Carrera de Ingeniería Electrónica. En la industria existen diversos tipos de sistemas de control, uno de los más comunes es aquel en el que la variable principal a controlar es la temperatura [1].

En el ámbito educativo, la teoría y la práctica constituyen dos realidades independientes que gestionan conocimientos de diferente índole y se desenvuelven en contextos distintos, sin embargo, se necesitan y se justifican mutuamente y con frecuencia se ignoran la una a la otra, generando problemas de aprendizaje [2].

Los módulos didácticos brindan la oportunidad al estudiante de entender la teoría de una forma más simple puesto que se pueden realizar varios ensayos que permiten presenciar situaciones reales que muchas veces los simuladores no permiten observar [3].

El prototipo propuesto en este proyecto de titulación utilizara el rango de voltaje industrial estándar, 0 a 10 voltios y el rango de corrientes 4 a 20 mA Esto con la finalidad de que pueda ser comandado por múltiples interfaces que posean esos rangos de operación, entre ellos tenemos NI myDAQ, Ni myRIO, módulos de Autonics, PLCs de Siemens, entre otros. De esta manera se podrá alcanzar a estudiantes de diferentes carreras que tengan en su malla curricular materias prácticas vinculadas al control de sistemas dinámicos, pudiendo ser: Electrónica y Automatización, Telecomunicaciones, Ingeniería en Control, Mecatrónica, Sistemas, Aeronáutica, etc.

El prototipo poseerá una entrada para el control del elemento calefactor tanto por voltaje como por corriente, de igual forma una entrada para el control del ventilador y una salida de la señal del sensor PT100 de tres hilos acondicionada con un transmisor industrial JUMO dTRANS T03 BU.

Con la configuración propuesta el equipo podrá ser utilizado para prácticas de control de temperatura por métodos como: PID, asignación de polos en espacio de estados, control predictivo basado en el modelo, entre otros.

JUSTIFICACIÓN

Actualmente no es común encontrar equipos comerciales orientados al sector académico destinados a realizar ensayos del control de temperatura que contengan elementos industriales. Cabe recalcar que la temperatura es una de las cuatro variables más importantes de la industria, razón por la cual es indispensable que se planteen prácticas referentes al control de esta variable física. Este proyecto tiene como objetivo desarrollar un prototipo de proceso térmico didáctico para realizar ensayos en el área de control automático. El prototipo planteado podrá ser controlado por un PLC S7-1200 haciendo uso de TIA Portal y por LabVIEW haciendo uso de la NI MyDAQ lo que brindará a los estudiantes una gran flexibilidad a la hora de implementar un controlador. Este tipo de dispositivos cobran gran importancia porque permiten apreciar el comportamiento de un sistema real en el que intervienen varios factores que generalmente no son considerados en una simulación como el efecto de saturación, perturbaciones no modeladas, ruido, entre otras.

Un módulo didáctico que permita controlar la temperatura permite observar el funcionamiento de los diferentes tipos de controladores, además, se pueden comparar las técnicas de control y analizar cuál de ellas es la más eficaz al momento de controlar la temperatura.

El prototipo de proceso térmico didáctico es un sistema abierto puesto que no se limita a trabajar con un único software a diferencia de dispositivos similares existentes en el mercado. Los componentes (resistencia calefactora, sensor de temperatura y transmisor) considerados para el prototipo serán los comúnmente empleados en procesos industriales. Además, se plantea utilizar señales continuas para realizar el proceso de control de temperatura a diferencia de dispositivos similares que utilizan una señal de PWM. Se considera importante mencionar este detalle porque los libros generalmente no muestran señales de control implementadas por PWM.

OBJETIVOS

OBJETIVO GENERAL

Diseñar y construir un prototipo de proceso térmico didáctico orientado al análisis y control de sistemas dinámicos.

OBJETIVOS ESPECÍFICOS

1. Analizar los diversos materiales y componentes electrónicos empleados a nivel industrial para el diseño y construcción de un prototipo de proceso térmico didáctico.
2. Diseñar y construir el prototipo de proceso térmico didáctico.
3. Modelar el prototipo de proceso térmico didáctico para conocer su dinámica.
4. Diseñar prácticas de laboratorio para el control del prototipo de proceso térmico didáctico.

CAPÍTULO 1: DESCRIPCIÓN DE PROCESO TÉRMICO

1.1 Arquitectura general del proceso.

La arquitectura del prototipo propuesto se muestra en la Figura 1. Esta arquitectura comprende tres componentes principales: un elemento controlador, un interfaz de 0-10 voltios y el proceso térmico como tal.

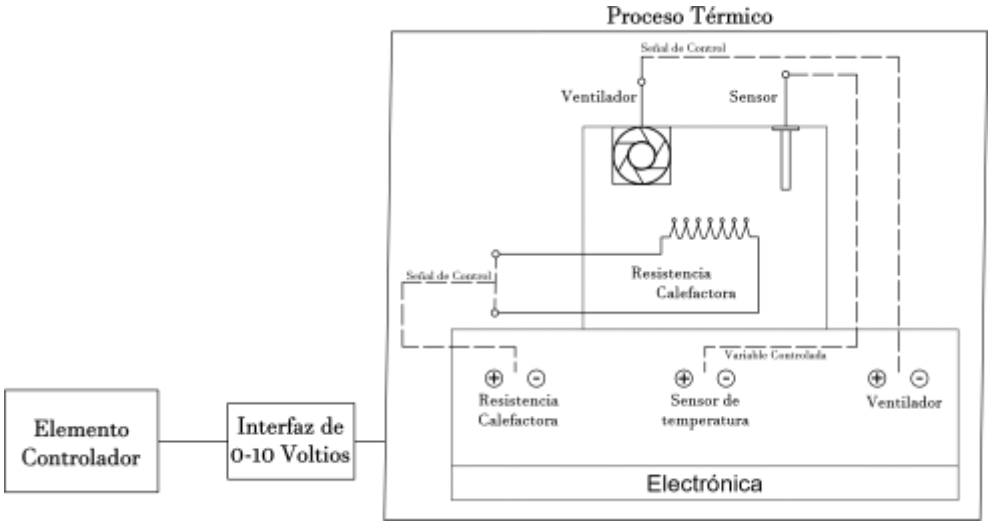


Figura 1. Arquitectura del proceso

A continuación, se describe cada uno de estos componentes:

1. Elemento controlador

El elemento controlador es el encargado de ejecutar las tareas de control para el proceso. Este elemento controlador puede ser un computador con software dedicado como LabVIEW o Matlab o cualquier otro dispositivo capaz de recibir señales del proceso, procesarlas y generar una acción de control, como por ejemplo un PLC o un sistema microprocesador en general.

2. Interfaz de 0-10 Voltios

Este módulo permite adquirir las señales del proceso en un formato establecido. Además, este módulo sirve como interfaz para enviar señales generadas por el elemento controlador hacia el proceso. En el caso de este prototipo el módulo

de interfaz se lo puede crear con una tarjeta de adquisición de datos de propósito general como la NI myDAQ, o también se puede utilizar una Ni myRIO o un PLC.

3. Proceso térmico

El proceso térmico comprende una cámara que encierra a una resistencia calefactora controlada junto con un sensor de temperatura PT100 y un elemento enfriador. La Figura 2 muestra el diagrama de bloques en lazo cerrado del proceso, donde se incluye el elemento controlador y el módulo de interfaz ambos elementos interactúan directamente con el proceso.

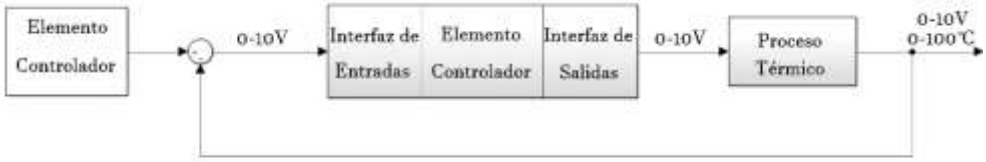


Figura 2. Diagrama de bloques del proceso

A continuación, se realiza una descripción de cada uno de los elementos que intervienen en el proceso térmico:

a. Elemento calefactor

El elemento calefactor es el encargado de generar un incremento de temperatura en la cámara, es importante que el elemento pueda variar su potencia en función de una señal de control y de ser posible exhiba un comportamiento lineal para facilitar las tareas de control [4]. El elemento calefactor seleccionado para este proyecto es una resistencia en espiral diseñada para que entregue una potencia de 60 vatios y funcione con 24 voltios.

Figura 3. Elemento calefactor [5]

b. Sensor de Temperatura

La temperatura es la magnitud física que nos muestra el valor de la cantidad de calor existente en un objeto o ambiente, para nuestros propósitos es importante el sensor posea una precisión similar o superior a 0.5 grados Celsius y que reaccione con suficiente velocidad como para realizar ensayos de teoría de control. En el mercado existen una amplia gama de alternativas para medir temperatura. De esta gama destacan los termopares y los termómetros de resistencia también conocidos como RTD (véase la Figura 4). Esta última opción es la que se ha seleccionado para medir la temperatura del proceso térmico. Este tipo de sensores basan su funcionamiento en la variación de resistencia a medida que varía la temperatura. El RTD que se utiliza en el mercado con mayor frecuencia es el PT100, el cual consiste en un alambre de platino que a 0 °C tiene una resistencia de 100 ohm y al aumentar la temperatura esta aumenta su valor [6]. Este tipo de sensores se les encuentra en industrias de alimentos, químicos y secado [7].

Figura 4. PT100 de 3 hilos [13]

c. Transmisor

Los transmisores son elementos industriales que toman la señal de los sensores, las transducen, las adecuan y las trasmite a un instrumento de control o indicador en un formato adecuado [8]. Generalmente a nivel industrial se manejas señales de corriente de 4-20mA o señales de voltaje de 0-10v. El transmisor es un complemento del sensor por lo que tiene que existir una compatibilidad entre estos dos elementos. Para el presente proyecto se ha seleccionado un transmisor industrial *JUMO dTRANS T03 BU Three-Wire Transmitter* (véase la Figura 5). La señal de salida de este transmisor es de 0 a 10 V y es lineal con la temperatura de medida de 0 a 100 ºC. El voltaje de alimentación es de 15 a 30V DC y se lo puede configurar mediante una herramienta informática especializada. En este caso el transmisor que utilizaremos se encuentra configurado para entregar una señal de salida de 0 a 10V [9].

Figura 5.Trasmisor JUMO [15]

d. Ventilador

El ventilador es un dispositivo que nos ayuda a refrigerar o ventilar un ambiente. Está compuesto por un motor DC o AC, el cual tiene en su eje una hélice que genera un flujo de aire.

Figura 6. Ventilador [16]

En este proyecto se ha seleccionado un ventilador de 30x30 milímetros que trabaje con 24 voltios y ofrezca una potencia de 2 vatios.

1.2 Diseño del prototipo de proceso térmico

Para el diseño del prototipo se ha establecido las siguientes condiciones:

1. La cámara del prototipo debe estar aislada de la electrónica.
2. La posición del sensor de temperatura debe ser horizontal y se ubica en la parte superior de la cámara.
3. Las borneras de alimentación y de señales deben estar en la parte frontal del equipo.

El diseño del prototipo se ha realizado en el software SolidWorks. En las figuras 8, 9, 10, 11 y 12 muestran las perspectivas del prototipo.

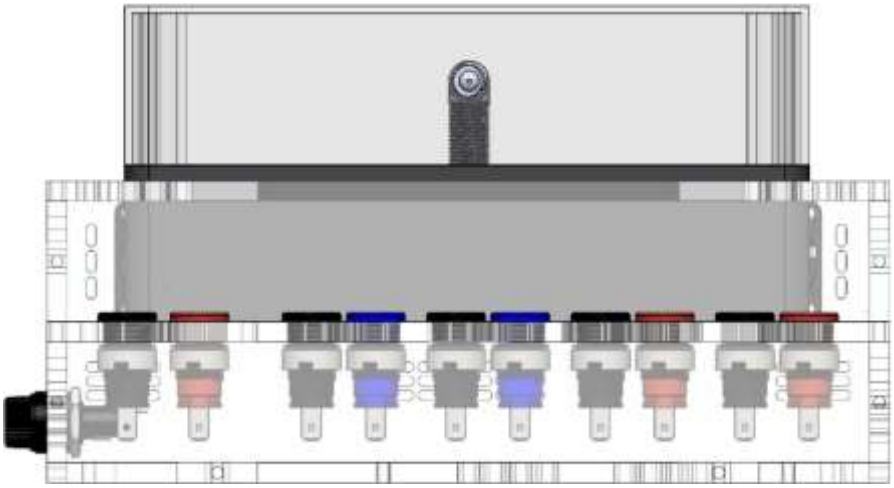


Figura 7. Vista frontal del prototipo

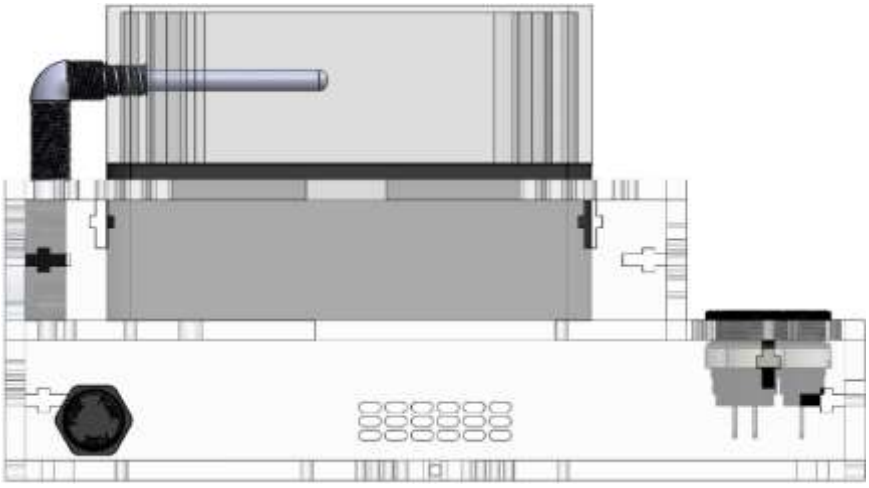


Figura 8. Vista lateral del prototipo

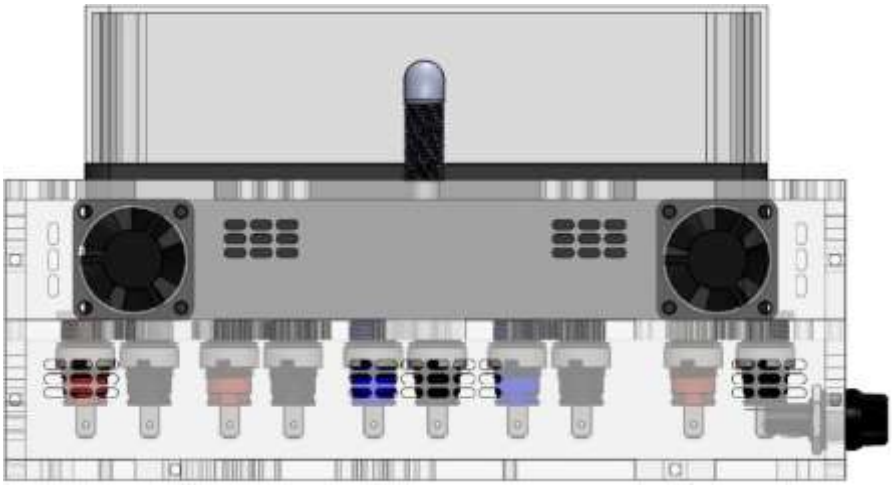


Figura 9. Vista posterior del prototipo

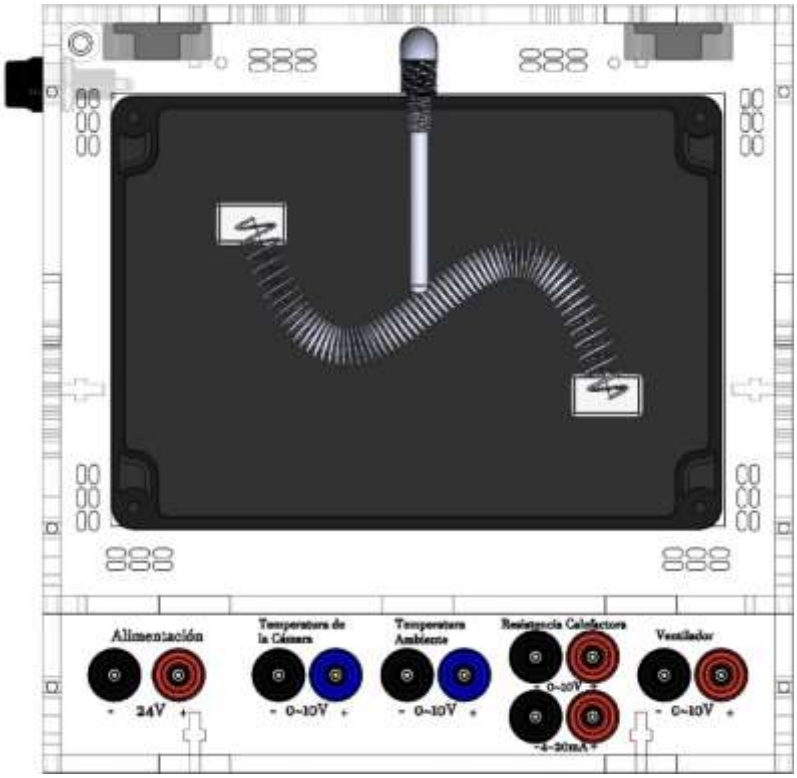


Figura 10. Vista Superior del prototipo

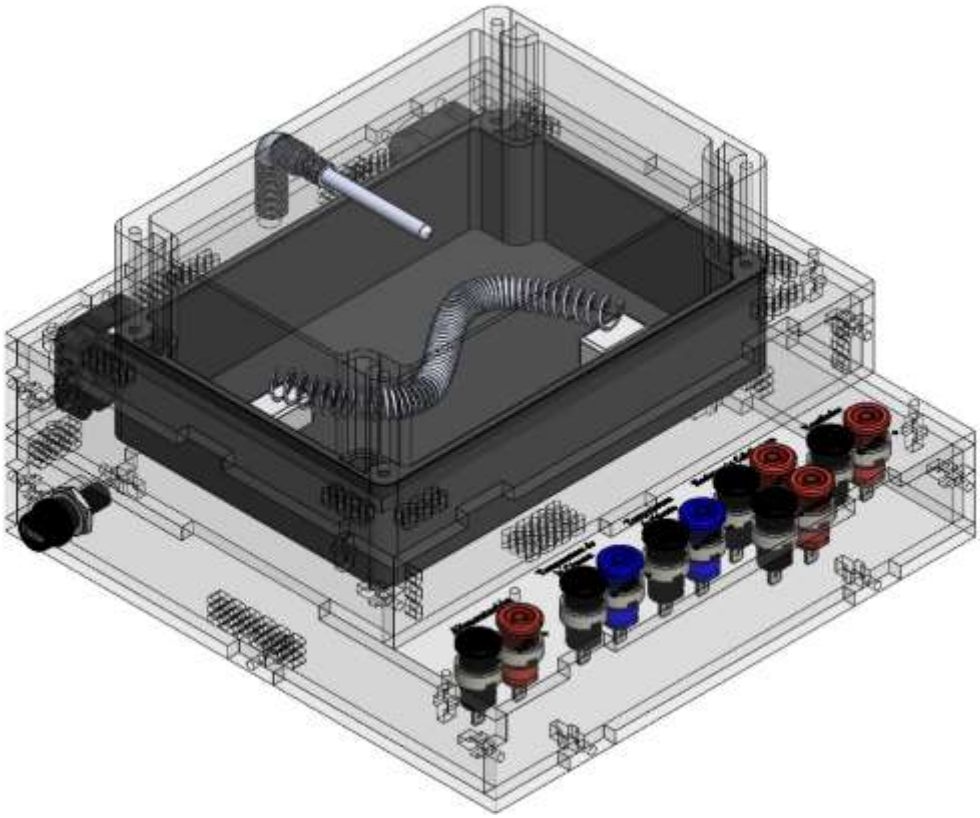


Figura 11. Vista isométrica del prototipo

En cada una de las vistas se puede ver la disposición de los componentes del proceso (elemento de medición y actuación). El volumen total de la recámara que actuará como horno es de 1423 cm^3. Esta recámara se la puede identificar en el prototipo ya que el color del material utilizado es más obscuro comparado con el resto de material que ha sido utilizado únicamente con el propósito de crear un prototipo con buena estética, funcional y que sobre todo proteja a los circuitos electrónicos.

A través de este diseño se pudo realizar una distribución óptima de los módulos o dispositivos que integran el sistema. Además, se generaron los planos necesarios para la implementación.

De izquierda a derecha tenemos 6 pares de borneras, las cuales serán descritas a continuación:

Alimentación: estos dos terminales reciben la energía de una fuente de 24 voltios DC a 3 amperios, posee una protección por diodos que impiden la circulación de corriente en caso de una conexión de polaridad inversa.

Temperatura de la cámara: En estos dos terminales podemos obtener la temperatura de la cámara expresada con el rango de voltaje estándar 0-10 voltios que representa la escala de 0-100 grados Celsius.

Temperatura ambiente: En estos dos terminales podemos obtener la temperatura ambiente expresada con el rango de voltaje estándar 0-10 voltios que representa la escala de 0-100 grados Celsius. En algunos ensayos puede llegar a ser interesante tener como información adicional los cambios en la temperatura ambiente.

Resistencia Calefactora (0-10V): En estos dos terminales ingresamos la señal de control de la resistencia calefactora, admite el estándar industrial de tensión, 0-10 voltios que representa 0-100% de la potencia.

Resistencia Calefactora (4-20mA): En estos dos terminales ingresamos la señal de control de la resistencia calefactora expresada en intensidad, admite el estándar industrial de corriente, 4-20 miliamperios que representa 0-100% de la potencia.

Ventilador (0-10V): En estos dos terminales ingresamos la señal de control del ventilador, admite el estándar industrial de tensión, 0-10 voltios que representa 0-100% de la potencia.

1.3 Diseño del circuito impreso

En la Tabla 1 se detallan los componentes que fueron utilizados en el circuito impreso. En adición en esta misma tabla se muestran las etiquetas que se han utilizado para distinguir a cada elemento dentro del diseño.

Tabla 1. Lista de componentes.

Cantidad	Componente	Etiqueta
3	Capacitor electrolítico	C1, C2, C3
2	Capacitor cerámico	C4, C5
4	Diodo rectificador	D1, D2, D3, D4
1	Led RGB	DS1
17	Terminales	J1, J2, J3, J4, J5, J6, J7, J8, J9, J10, J11, J12, J13, J14, J15, J16, J17
1	Mosfet IRF540	Q1
1	Transistor TIP31C	Q2
20	Resistores 1/4 W 1k	R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14,

		R15, R16, R17, R18, R19, R20
2	Resistencias de precisión 1k	R21, R22
1	Regulador LM7812	U1
1	Regulador LM7805	U2
1	Microcontrolador PIC18F4550	U3
3	Amplificador operacional LM358P	U4, U5, U6
1	Cristal 16 MHz	Y1

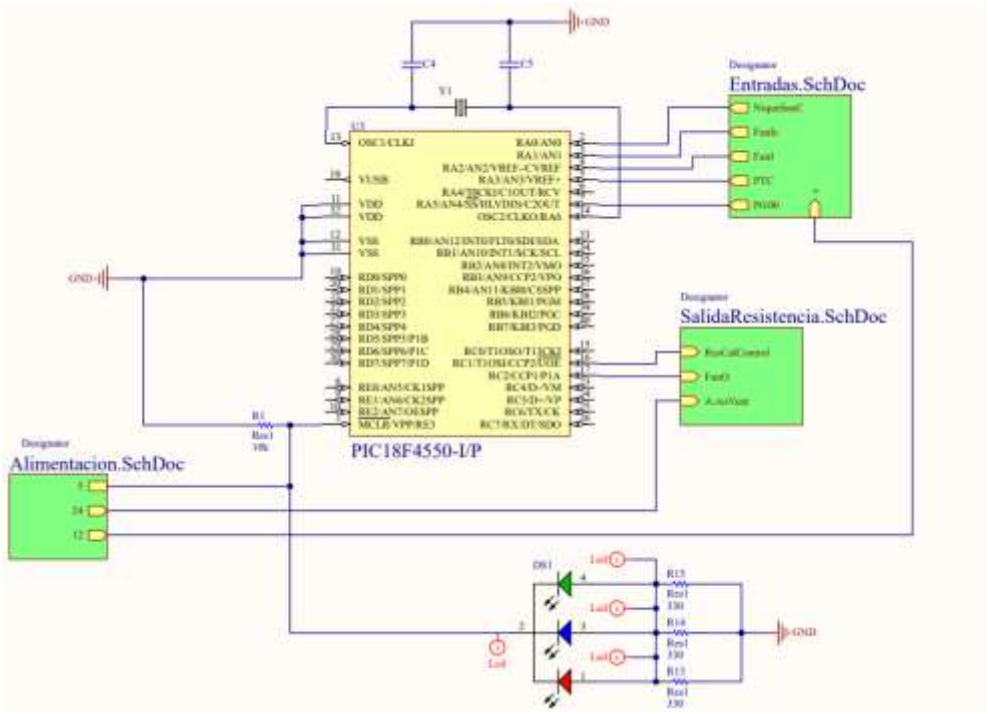


Figura 12. Esquemático del circuito principal

En la Figura 12 se muestra el esquemático del circuito principal, para el cual se utilizó un PIC18f4550 para el manejo de los periféricos utilizados en el dispositivo. Las tareas gestionadas por el microcontrolador se listan a continuación:

- Establecer la temperatura máxima de funcionamiento en 85 grados.
- Controlar la potencia entregada a la resistencia calefactora.
- Controlar la potencia entregada a los ventiladores.
- Gestionar las señales de entrada, corriente y voltaje, de la resistencia calefactora.
- Mostrar alertas luminosas.

Las señales de entrada del dispositivo utilizan un rango de 0 a 10 voltios, las cuales han sido acondicionadas para ser procesadas por un dispositivo TTL lo que significa que su valor debe ser escalado a un rango de 0 a 5 voltios, para esto, se ha utilizado divisores de tensión y amplificadores operacionales. El esquema de conexión se muestra en la Figura 13.

En la Figura 14 se observa el bloque de alimentación, este es un circuito simple que utiliza las configuraciones estándar recomendadas por los fabricantes para la polarización de los regulares de tensión. Se utiliza el integrado LM7812 para obtener 12 voltios y el integrado LM7805 para obtener 5 voltios. D1 es un diodo utilizado como protección contra conexión de polaridad inversa.

Para controlar la potencia entregada a la resistencia calefactora se utiliza un Mosfet IRF540. La señal proveniente del microcontrolador polariza directamente al dispositivo. Los ventiladores son controlados por el transistor de potencia TIP31C (véase Figura 15).

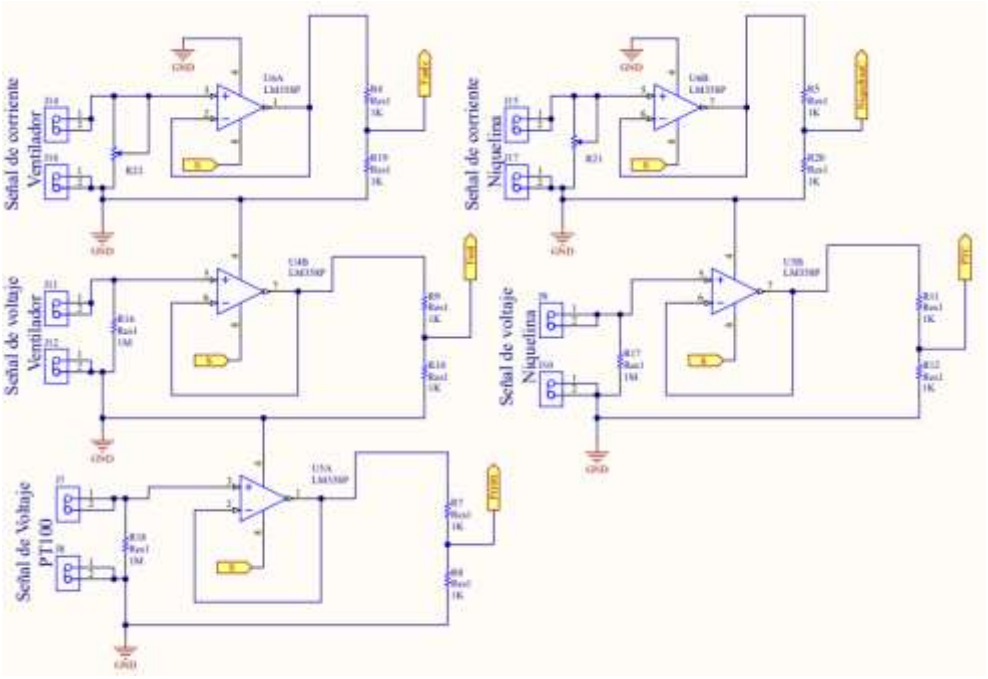


Figura 13. Esquemático del circuito de entrada de señales

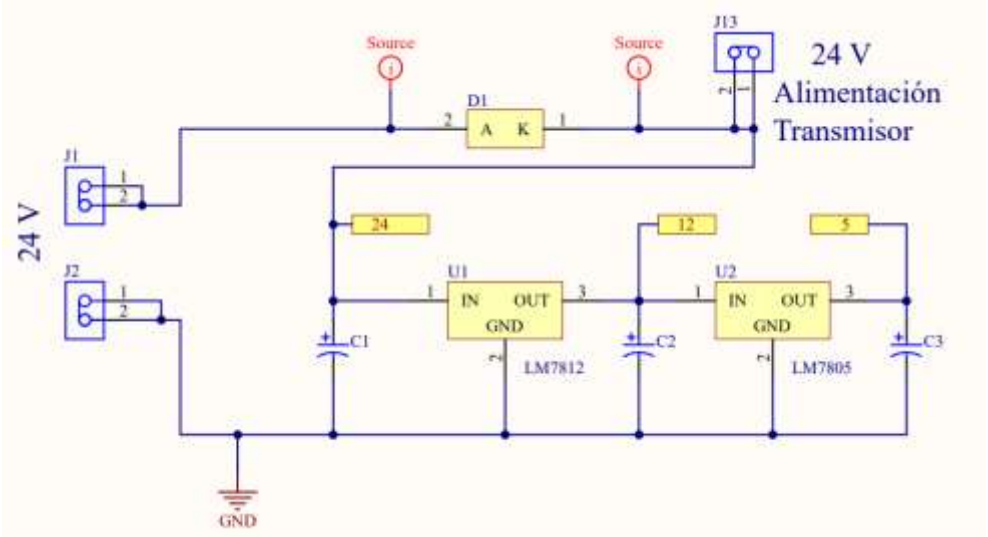


Figura 14. Esquemático del circuito de alimentación

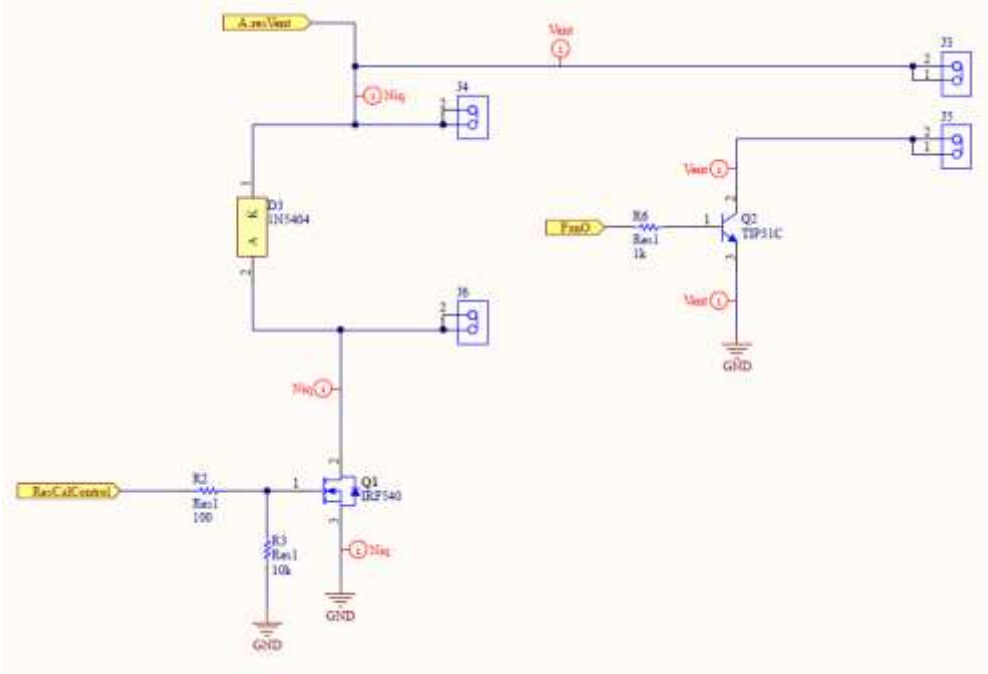


Figura 15. Esquemático del circuito de potencia

El ancho de pistas se calcula con la siguiente ecuación:

$$Ancho = \frac{\left(\frac{I}{k_1 \Delta T^{k_2}}\right)^{\frac{1}{k_3}}}{L \cdot 1.378} \qquad (1)$$

Donde

I: corresponde a la corriente máxima en amperios

ΔT: corresponde al incremento de temperatura máximo

L: corresponde al grosor de la capa de cobre, en onzas por pie cuadrado.

k_1: corresponde a una constante de 0.0150 para pistas internas y 0.0647 para pistas externas.

k_2: corresponde a una constante de 0.5453 para pistas internas y 0.4281 para pistas externas.

k_3: corresponde a una constante de 0.7349 para pistas internas y 0.6732 para pistas externas [10].

El resultado se expresa en milésimas de pulgada.

Se utiliza la Ecuación (1) para encontrar el ancho de las pistas para la resistencia calefactora y los ventiladores.

En la Ecuación (2) se muestra el cálculo para las pistas de potencia de la resistencia calefactora las cuales deberán conducir una corriente máxima de 4.8 amperios.

$$Ancho = \frac{\left(\frac{4.8}{0.0647 \cdot (60-22)^{0.4281}}\right)^{\frac{1}{0.6732}}}{1 \cdot 1.378} \qquad (2)$$

$$Ancho = 43.0939\, mil = 1.0946\, mm$$

La Ecuación (3) para el cálculo de las pistas de potencia de los ventiladores con una corriente máxima de 150mA, entonces:

$$Ancho = \frac{\left(\frac{0.15}{0.0647 \cdot (60-22)^{0.4281}}\right)^{\frac{1}{0.6732}}}{1 \cdot 1.378} \qquad (3)$$

$$Ancho = 0.250\, mil = 0.00635\, mm$$

Para asegurar que no existan problemas de calentamientos no considerados, se ha optado por usar 0.9mm para las todas las pistas con una corriente menor a 0.5 amperios y 1.5mm para las pistas con una corriente mayor a 3 amperios.

Para la separación entre pistas nos referimos a la Tabla 2, en la que nos dice que debemos utilizar una separación mínima de 0.317mm.

Tabla 2. Separación mínima entre pistas de un circuito impreso [10]

Tensión (V)	Separación mínima (mm)
0 a 30	0.317
31 a 50	0.444
51 a 150	0.571

El PCB se muestra en la Figura 16. Este fue realizado utilizando la herramienta Ruteo Automático de Altium Designer donde previamente se establecieron reglas de ancho y separación de pistas. La Figura 17 muestra una vista panorámica en tres dimensiones de la tarjeta electrónica con cada uno de sus componentes.

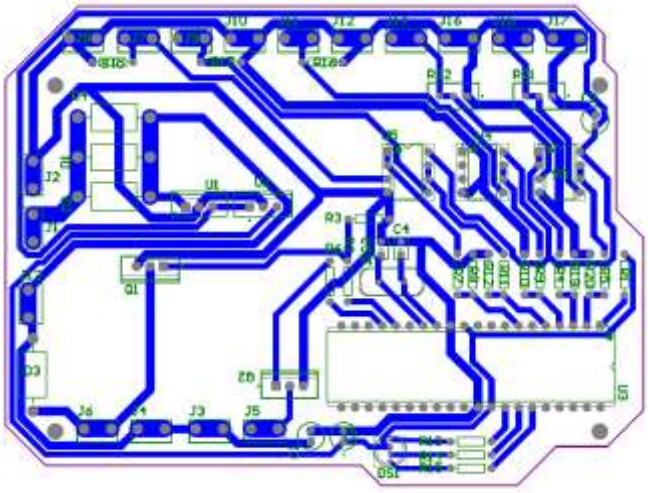


Figura 16. Circuito impreso

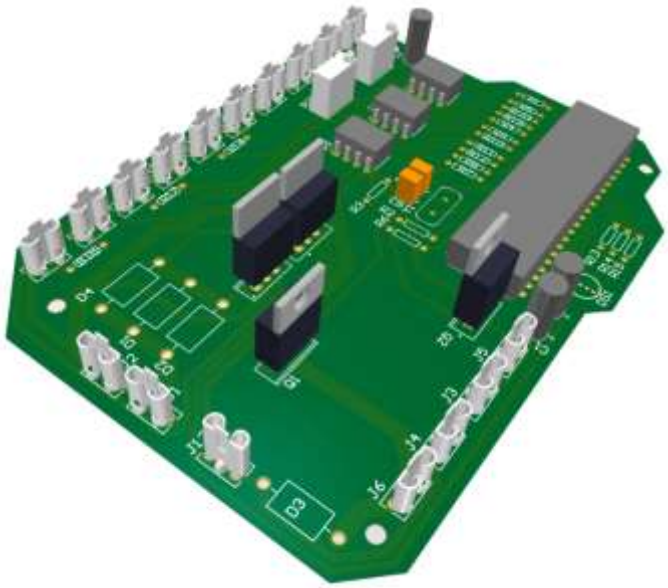


Figura 17. Modelo 3D

CAPÍTULO 2: IDENTIFICACIÓN DEL PROCESO

La identificación es un proceso mediante el cual se puede obtener un modelo. Este será de mucha utilidad para poder diseñar y sintonizar los controladores que se pretenden implementar en esta plataforma. Actualmente existen diversos métodos de identificación y muchos de estos se pueden aplicar con la plataforma Matlab, la cual se utilizará en este proyecto para identificar el modelo del proceso. Este capítulo ha sido estructurado en varios puntos. En primera instancia se hablará sobre el diseño del VI que nos servirá para realizar la identificación; luego de que se recolecten los datos, utilizaremos el software Matlab. En complemento se dará una guía paso a paso de cómo sacar el modelo; por último, como complemento se realizará un pequeño análisis para verificar y observar la no linealidad y la histéresis que puede tener el sistema.

1.1 Instrumento virtual para la identificación

El instrumento virtual o VI es el panel frontal que permite al usuario interactuar con el proceso en conjunto con un módulo de adquisición de datos. Para el diseño del VI se ha utilizado el software LabVIEW y para la adquisición de datos se ha utilizado un dispositivo NI myRIO.

El instrumento virtual consta de un bloque de entrada analógica, una salida analógica, un bloque para almacenamiento de datos y un bloque para visualización de datos en tiempo real (véase Figura 18). A continuación, se describen cada uno de estos bloques.

Entrada analógica: este bloque permite obtener los datos medidos por el sensor de temperatura de la cámara. La señal obtenida tiene un rango de 0 a 10 V la cual tendría un equivalente de 0 a 100ºC en temperatura.

Salida analógica: mediante este bloque es posible comandar la resistencia calefactora con una señal de 0 a 10 V. Lo que se pretende es excitar al sistema con señales de entrada definidas, para obtener la dinámica de este.

Almacenamiento de datos: puesto que este es un proceso lento se ha tomado una muestra cada segundo. Para la identificación de la planta se ha guardado los datos en su equivalente a voltaje. Este bloque nos entrega archivos en formato. lvm, los cuales posteriormente son utilizados para obtener el modelo del sistema.

Visualización de datos: este bloque permite observar en tiempo real la temperatura de la cámara al darle un voltaje de referencia a la resistencia calefactora.

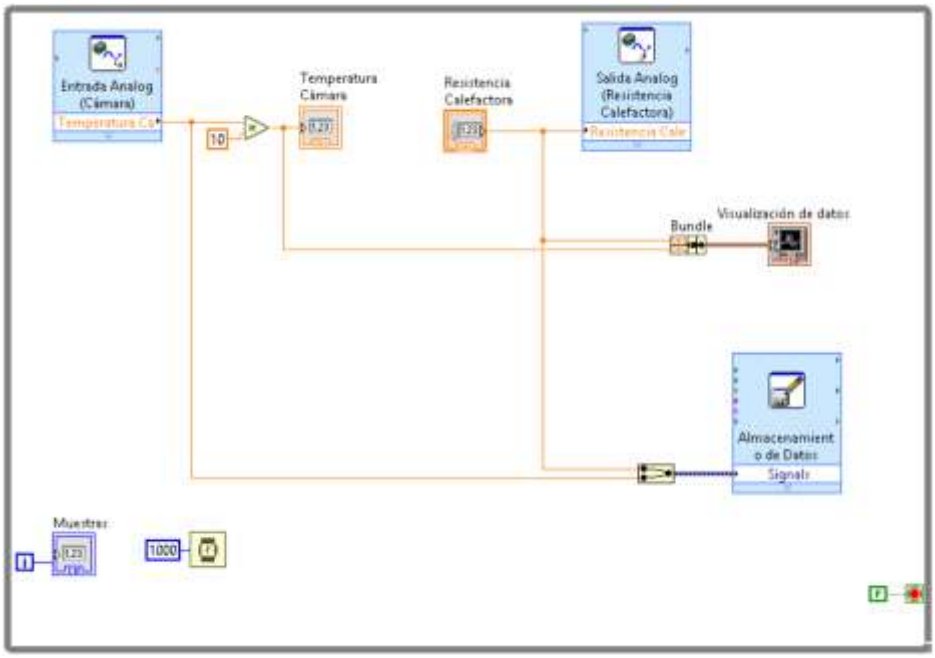


Figura 18. Diagrama de bloques de la adquisición de datos para la identificación de la planta

El panel frontal implementado en el software se muestra en la Figura 19. Este panel cuanta con un indicador que muestra la temperatura de la cámara, un control para comandar la resistencia calefactora y un gráfico para visualizar las señales.

Figura 19. Panel frontal de la adquisición de datos para la identificación de la planta

1.1.1 *Identificación del sistema*

Para la identificación del sistema, excitamos al sistema con una entrada escalón de amplitud definida hasta obtener estabilidad, luego repetimos el proceso desde ese punto de estabilidad, lo que se consigue con esto es conseguir un comportamiento mayormente provocado por la resistencia calefactora y no por la temperatura ambiente. En la Figura 20 se muestra los datos obtenidos en una gráfica.

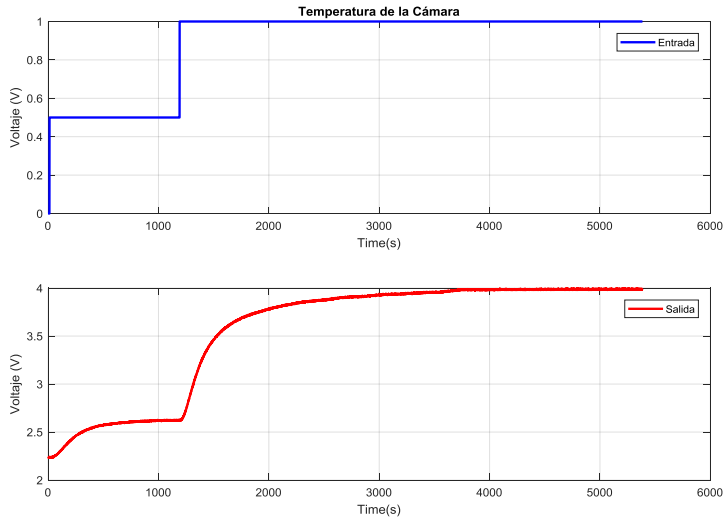


Figura 20. Datos obtenidos para la identificación de la planta

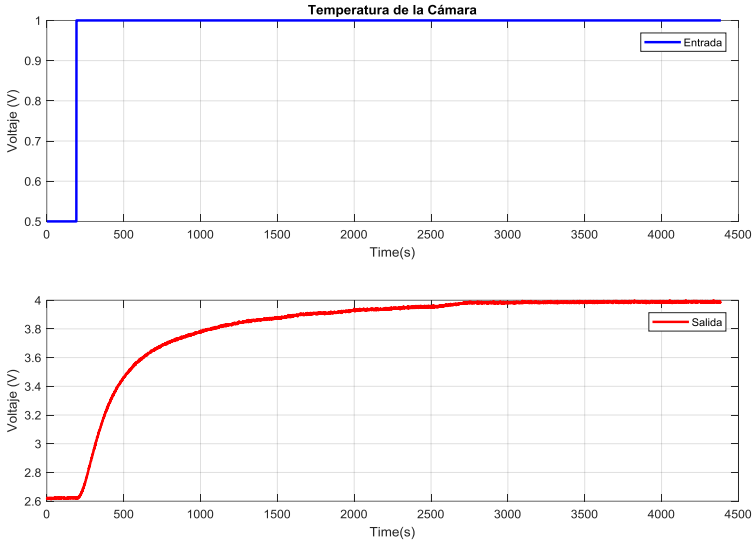


Figura 21. Obtención de datos para la identificación

Para el experimento de identificación se debe seleccionar el conjunto de datos de entrada y salida. Para este caso se ha seleccionado el segundo escalón. (Véase Figura 21)

A continuación, es necesario eliminar el offset en los datos. Se debe recalcar que lo que interesa es la dinámica del sistema por lo que el offset no debe ser considerado, más bien debe ser eliminado para facilitar el trabajo a la herramienta de identificación (véase Figura 22).

En la Figura 23 se observa que el sistema tiene una demora o tiempo muerto por lo que para la identificación del sistema se debe considerar ya sea un retardo o uno o más ceros para la identificación (véase Apéndice A, guía de prácticas 2).

Se ha identificado el sistema con modelos de primer, segundo, tercer y cuarto orden, esto con el propósito de seleccionar un modelo que consiga un equilibrio en similitud y complejidad. Es importante mencionar que mientras más polos y ceros tenga el modelo más complejo será controlarlo. Los resultados del proceso de identificación se muestran en la Figura 24.

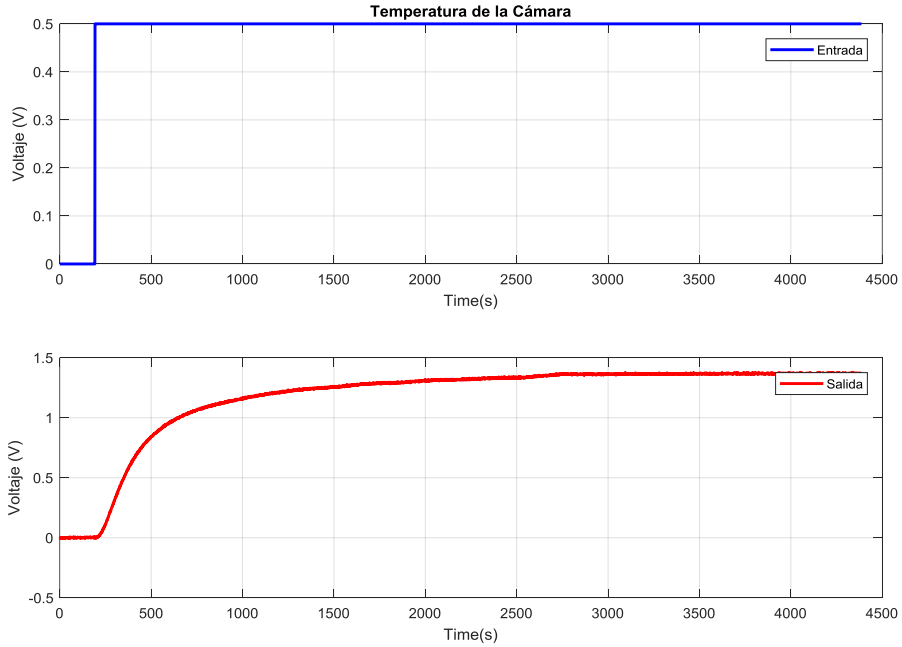


Figura 22. Datos para identificación sin offset de la temperatura de la cámara

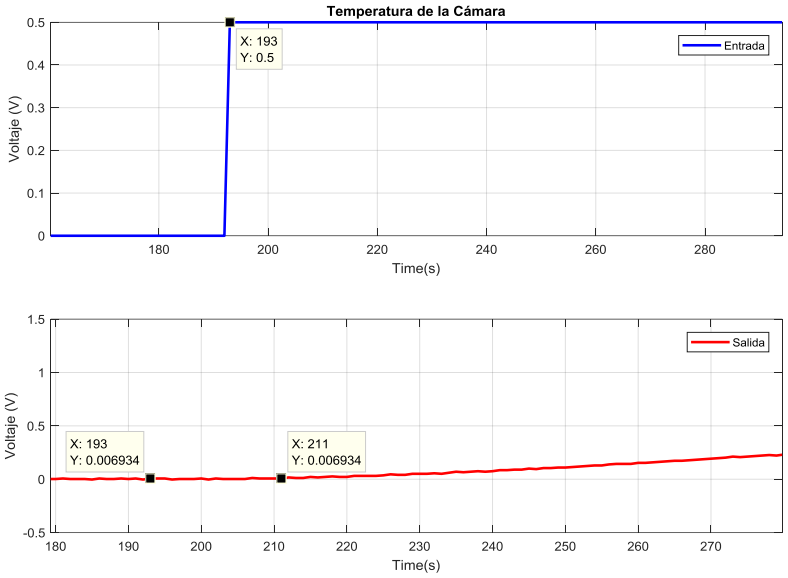


Figura 23. Tiempo muerto

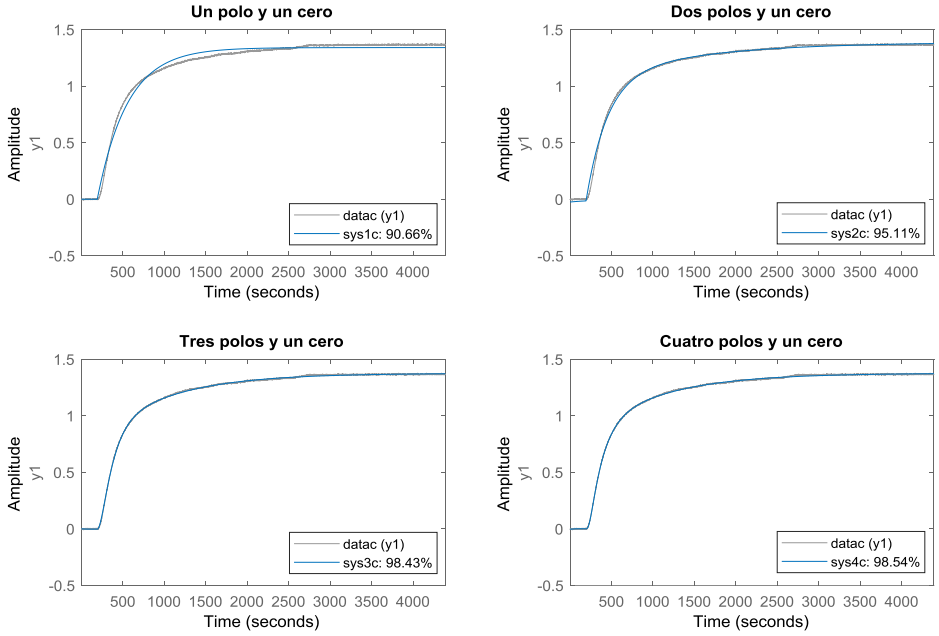


Figura 24. Sistemas de primer, segundo, tercer y cuarto orden.

Al analizar los resultados se puede ver que los mejores desempeños en cuanto a precisión se obtienen con los modelos de tres y cuatro polos. También se puede ver

que un solo polo es insuficiente para representar la dinámica del sistema con gran precisión. Un siguiente paso para seleccionar el mejor modelo sería trazar el mapa de polos y ceros de cada uno de estos.

Al analizar los polos y los ceros se observa que el sistema de tercer orden es el más adecuado ya que en el sistema de cuatro polos prácticamente existe una cancelación de un polo y un cero (véase Figura 25). Por lo antes expuesto se seleccionará un modelo de tercer orden.

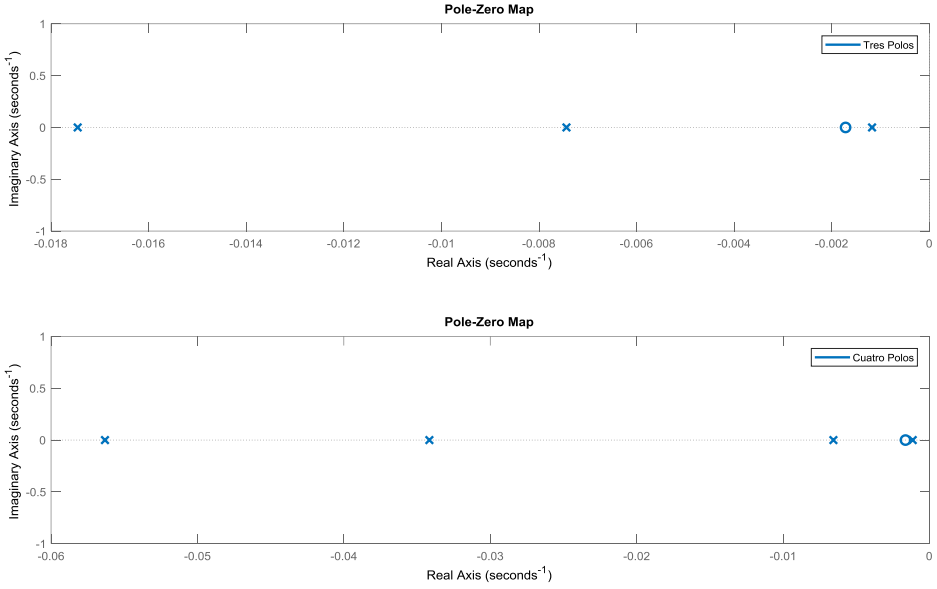


Figura 25. Poner solo el mapa de polos y ceros

La respuesta al escalón del sistema de tercer orden tiene los siguientes parámetros

```
S3c =
  struct with fields:

        RiseTime: 1.1248e+03
    SettlingTime: 2.5523e+03
     SettlingMin: 2.4002
     SettlingMax: 2.7481
       Overshoot: 0
      Undershoot: 0
            Peak: 2.7481
        PeakTime: 4.4130e+03
```

La función de transferencia que mejor se ajusta a los datos de la temperatura de la cámara se muestra en la Ecuación (4).

$$s3 = \frac{0.0002453s + 4.212 * 10^{-7}}{s^3 + 0.02612s^2 + 0.0001594s + 1.529 * 10^{-7}} \qquad (4)$$

Para generalizar el modelo se procede a identificar nuevamente el sistema, pero utilizando un tiempo muerto en lugar de un cero. Esto será útil al implementar controladores con integradores embebidos ya que de otra forma el cero podría cancelarse con el integrador debido a la naturaleza lenta del sistema.

Se identifica la planta con el tiempo muerto de 17 segundos (véase Figura 23) y se obtienen 4 modelos.

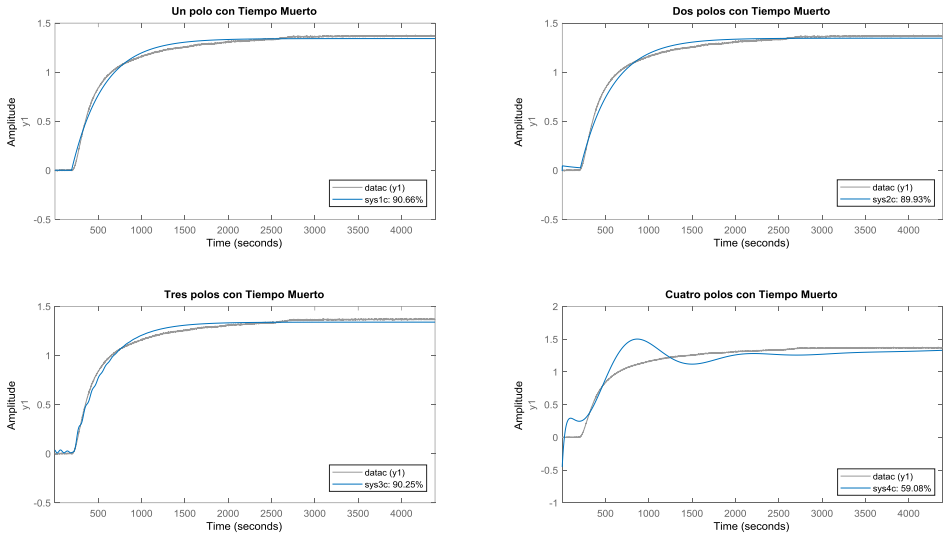


Figura 26. Sistemas de primer, segundo, tercer y cuarto orden con retardo de 17 segundos.

La respuesta al escalón del sistema de tercer orden tiene los siguientes parámetros

```
S3cr =
  struct with fields:

        RiseTime: 754.4157
    SettlingTime: 1.3643e+03
     SettlingMin: 2.4212
     SettlingMax: 2.6787
       Overshoot: 0
      Undershoot: 0
            Peak: 2.6787
        PeakTime: 2.6775e+03
```

La función de transferencia que mejor se ajusta a los datos de la temperatura de la cámara utilizando un retardo se muestra en la Ecuación (5).

$$s3 = e^{-17s} * \frac{4.936 * 10^{-5}}{s^3 + 0.01448s^2 + 0.00637s + 1.842 * 10^{-5}} \quad (5)$$

Para ver una guía detallada de la identificación del modelo véase guía de practica 1.

1.2 Detección de la histéresis.

El análisis de histéresis se realizó llevando al sistema a su estado estacionario utilizando como entrada una sucesión de escalones positivos y posteriormente una sucesión de escalones negativos. Lo que se pretende con este ensayo es determinar si la salida difiere su respuesta para una entrada ascendente respecto a una descendente.

La señal de entrada tiene un valor máximo de 2.5 voltios y está compuesta por escalones de 0.5 voltios de amplitud. En la Figura 27 se muestra la respuesta del sistema.

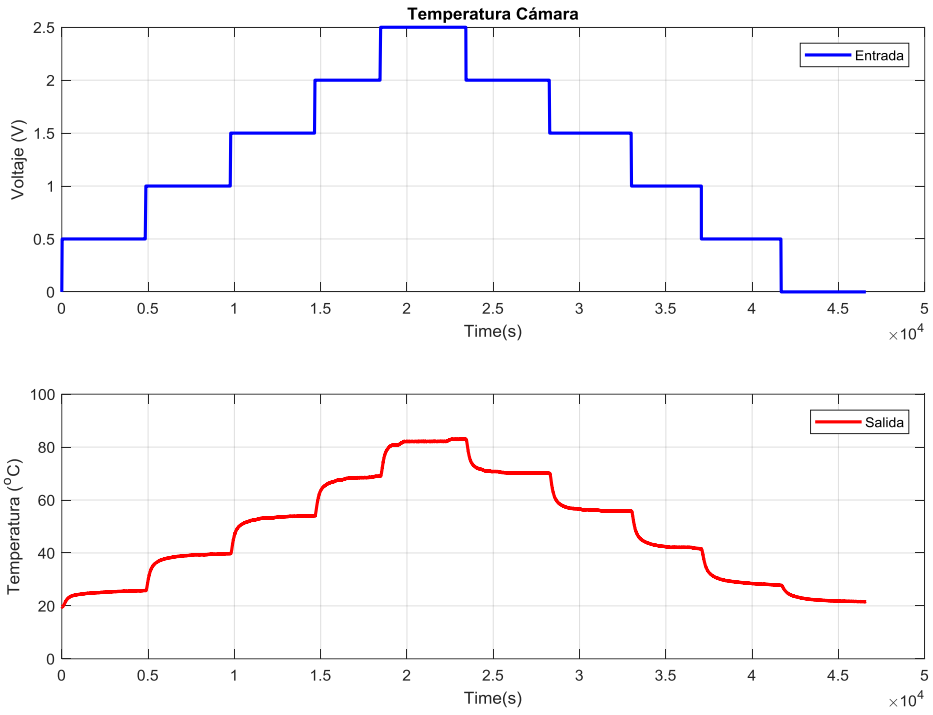


Figura 27. Respuesta del Sistema

Los valores de la señal en estado estacionario se muestran en la Tabla 3

Tabla 3. Respuesta del sistema con entrada escalonada

Voltaje	Temperatura
0	19.63
0.5	25.73
1	39.65
1.5	54.05
2	68.99
2.5	82.96
2	70.17
1.5	55.91
1	41.8
0.5	27.93
0	21.63

En la Figura 28 se muestra las curvas de respuesta del sistema para cuando la entrada es ascendente y descendente. Debido al efecto de histéresis observamos que la salida varía alrededor de dos grados centígrados para una misma entrada.

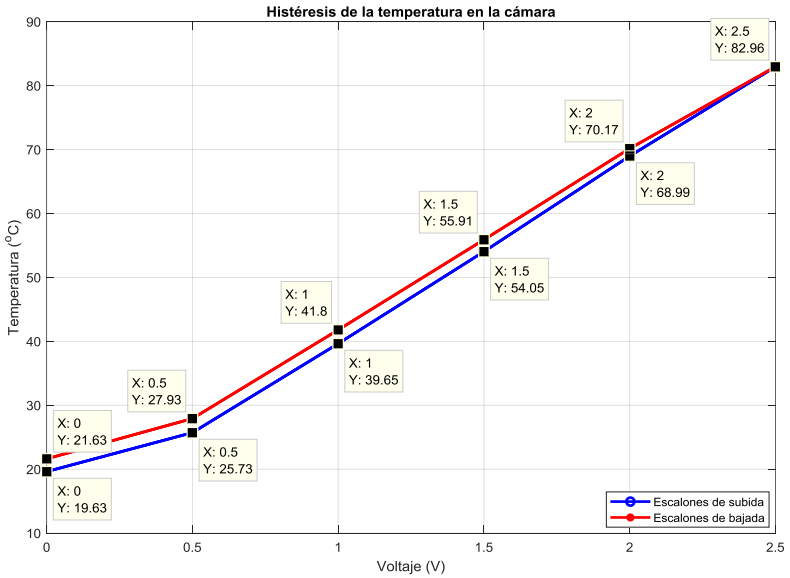


Figura 28.Histeresis de la temperatura en la cámara

CAPÍTULO 3: ESQUEMAS DE CONTROL

En el presente trabajo de investigación se realizarán algunos esquemas de control aplicados al proceso térmico construido. Estos esquemas de control se basan el uso de tres herramientas de hardware con gran aplicabilidad en el entorno industrial. En este caso se implementarán los controladores utilizando un módulo PID industrial TK4S-T4RC, adicionalmente se utilizará un PLC y finalmente también se desarrollarán esquemas de control utilizando un computador personal.

1.3 Controlador PID

Un controlador PID es una estrategia de control que agrega al sistema dos ceros, un polo en el origen y una ganancia estática. El diseñador es libre de modificar la posición de los ceros y el valor de la ganancia. La expresión que describe su comportamiento se muestra en la Ecuación (6):

$$C(s) = K_p \left[1 + \frac{1}{T_i s} + sT_d\right] E(s) \qquad (6)$$

Donde:

C(s): representa la respuesta del controlador en el dominio de Laplace

K_p: ganancia proporcional

T_i: constante de tiempo integral en segundos

T_d: constante de tiempo derivativa en segundos

E(s): error entre la referencia y la salida

En la Figura 29 se muestra el esquema de un sistema con un controlador PID. Como se puede observar el controlador actúa sobre la diferencia entre la entrada y la salida.

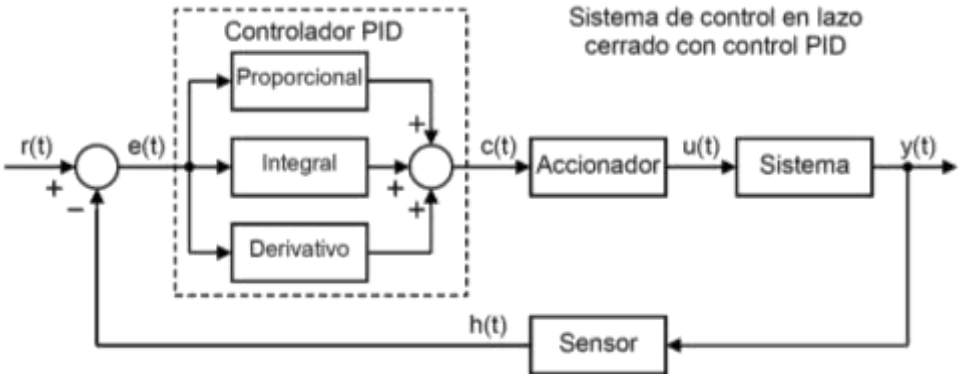


Figura 29. Esquema de controlador PID

Si la ley de control es meramente proporcional al error, podemos otorgarle más velocidad al sistema aumentando la ganancia k_p, sin embargo, es probable que la respuesta oscile más en torno a la referencia. Para evitar las oscilaciones se puede añadir la actuación proporcional a la derivada del error, con esto el sistema obtiene una capacidad de anticipación que le permite de cierta manera reducir la velocidad con la que se acerca a la referencia. En ocasiones puede pasar que tanto la acción proporcional como la derivativa tengan un valor cercano a cero cuando la respuesta se acerca a la referencia, lo que puede provocar un error en estado estacionario. En estos casos es necesario implementar la acción integral para que el error se vuelva cero paulatinamente. [11]

Los controladores PID son ampliamente utilizados, razón por la cual es muy común encontrar una amplia gama de dispositivos listos para la implementación del controlador. En este proyecto se han utilizado un PLC s7-1200, un módulo TK4S-T4RC de Autonics y un computador con Labview para su implementación, no obstante, para cuestiones de diseño y sintonización también se ha utilizado la herramienta PID Tuner de Matlab.

1.3.1 PID Tuner

PID Tuner es una herramienta de Matlab, diseñada para la sintonización de controladores PID y sus variaciones a partir de un modelo o un conjunto de datos.

Para este proyecto cargamos el modelo en un script de Matlab, para ello ejecutamos las siguientes líneas de código, la respuesta al escalón se muestra en la Figura 30.

```
% Definicion de la planta
clear all
clc
%Numerador de la funcion de transferencia
num=[0.0002453 4.2012e-7];
%Denominador de la funcion de transferencia
den=[1 0.02612 0.0001594 1.529e-7];
%Definicion de la planta
G=tf(num,den)
%Respuesta al escalon
step(G)
grid on
```

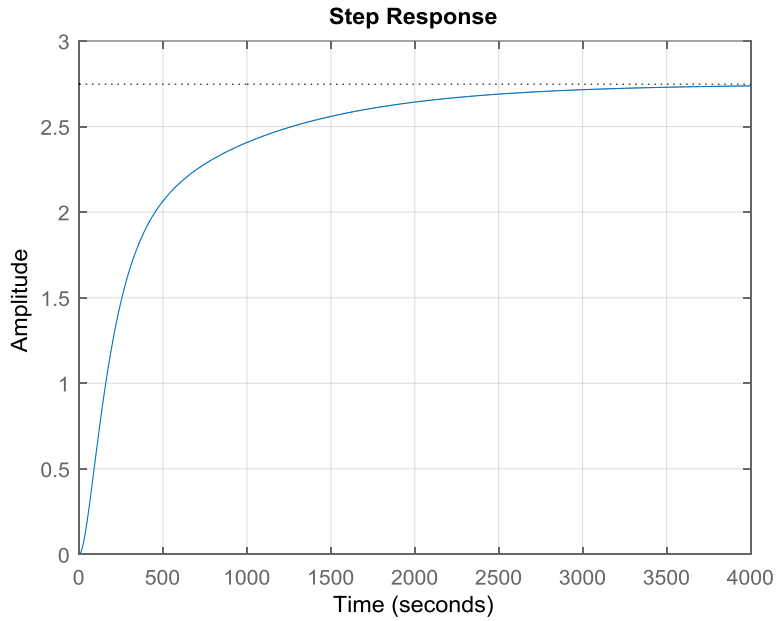


Figura 30. Respuesta al escalón

Iniciamos la herramienta desde la paleta de complementos de Matlab y cargamos el modelo. Seleccionamos el tipo de controlador a utilizar, en este caso un PID en su forma paralela.

Las barras deslizantes de la parte superior nos permiten sintonizar el controlador definiendo la velocidad y comportamiento. (véase Figura 31)

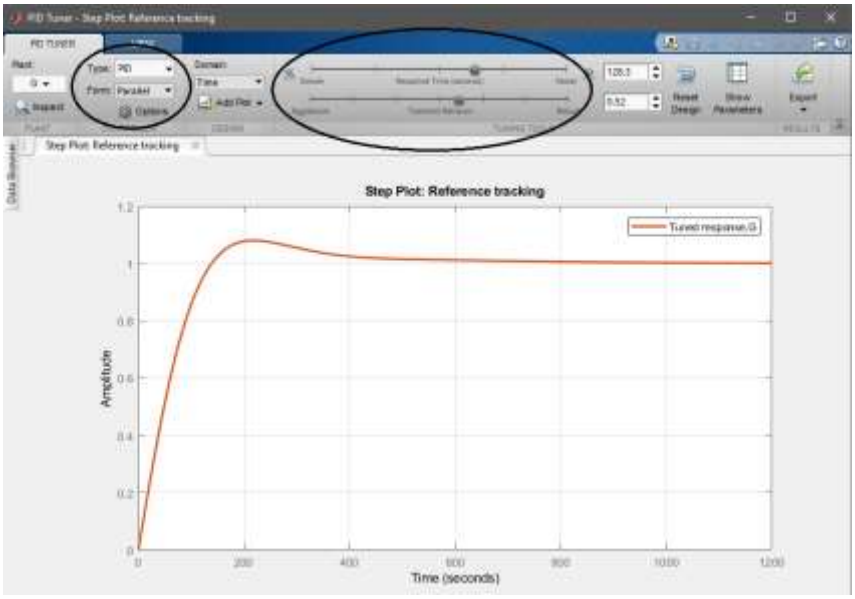


Figura 31. Ventana PID Tuner

Para este proyecto hemos seleccionado el comportamiento de la Figura 32. Consideramos importante mencionar que debido a las limitaciones físicas del dispositivo podría haber configuraciones que no se puedan alcanzar, razón por la que es necesario encontrar un equilibrio entre velocidad y robustez del sistema.

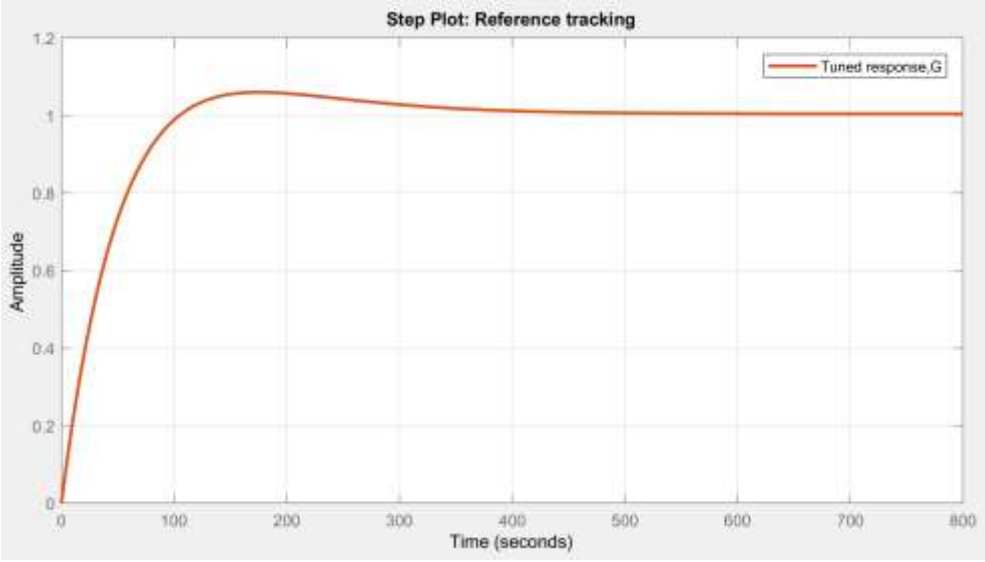


Figura 32. Respuesta del sistema con controlador PID sintonizado

De acuerdo con la respuesta obtenida, el sistema de control presenta las siguientes características:

$$Tiempo\ de\ establecimiento = 317\ segundos$$

$$Sobrepaso = 4.25\%$$

Las constantes del controlador sintonizado son las siguientes:

$$Kp = 2.7849$$

$$Ki = 0.02138$$

$$Kd = 90.6567$$

1.3.2 Implementación de un controlador PID con LabVIEW

Para la implementación de este controlador se utilizó un computador con el software LabVIEW 2020, la tarjeta NI myRIO y el prototipo de ensayos de control de temperatura.

La función de transferencia del prototipo de ensayos de control de temperatura se obtuvo sin realizar ninguna conversión de unidades por lo que debemos considerar que las señales con las que el controlador trabajará deberán estar expresadas en voltios, tanto para la entrada como para la salida.

LabVIEW ofrece un bloque PID con funcionamiento en tiempo real el cual será utilizado en este proyecto. La Figura 33 muestra los atributos del bloque, mientras que a continuación se describen los componentes de este bloque:

Setpoint: El setpoint corresponde a la temperatura expresada en voltios que deseamos que el sistema alcance, recordamos que cada voltio equivale a 10 grados Celsius partiendo de cero para ambas unidades.

Process variable: Este atributo corresponde a la variable que queremos controlar, en este caso la temperatura del interior de la cámara térmica del prototipo expresada en voltios.

Ganancias del controlador: Estos atributos corresponden a las contantes de la sintonización del controlador PID expresadas como ganancia proporcional, tiempo integral y tiempo derivativo. Estas constantes dependen de la respuesta deseada por el diseñador.

Output: Esta es la señal de control que comandará el funcionamiento del dispositivo.

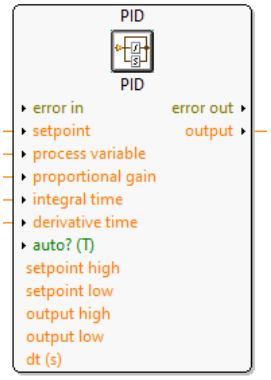


Figura 33. Bloque PID LabVIEW

Para este caso el dispositivo NI myRIO se utilizará como interfaz para comunicar el computador con el prototipo de ensayos de control de temperatura. El VI referente al controlador se muestra en la Figura 34, donde podemos observar la correcta conexión del bloque PID.

Además, se ha añadido un bloque de conversión de estructura del controlador PID, de esta forma damos libertad al usuario de seleccionar la estructura que le sea más conveniente para ingresar las constantes en función del método de sintonización de su preferencia.

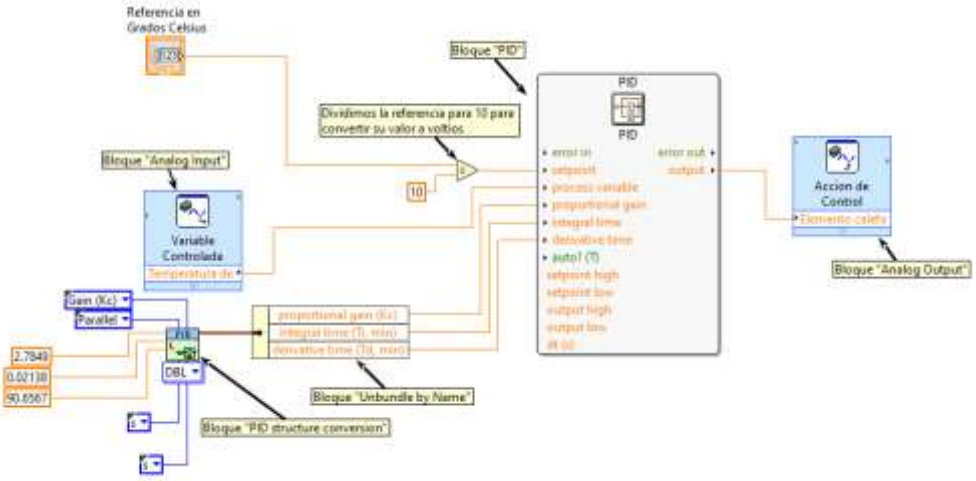


Figura 34. VI controlador PID

La configuración de los parámetros del bloque PID se muestra en la Figura 35. Los bloques de entrada y salida analógica se conectan directamente al bloque PID, esto debido a que el modelo se obtuvo con las señales expresadas en voltios, en concordancia la señal de referencia también deberá estar expresada en voltios.

Para modificar los parámetros de cualquiera de los bloques de LabVIEW debemos dar doble clic sobre el bloque a modificar. La configuración de los bloques de entrada y salida se muestran en la Figura 36 y la Figura 37. La configuración es similar en ambos casos, en el campo "Channel" vamos a seleccionar el puerto físico que será utilizado en el ensayo y en "Custom Channel name" el nombre asociado a los datos de entrada o salida.

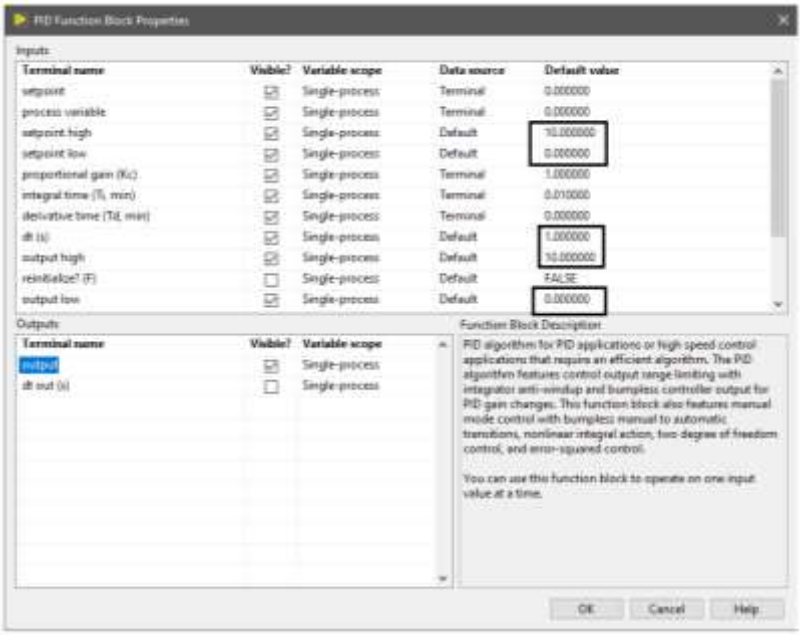


Figura 35. Configuración del bloque PID

Figura 36. Configuración bloque de entrada analógica

Figura 37. Configuración de salida analógica

La interfaz gráfica requiere básicamente de un campo para ingresar la referencia, un botón de stop y un graficador que nos muestre el comportamiento del sistema. (Véase Figura 38)

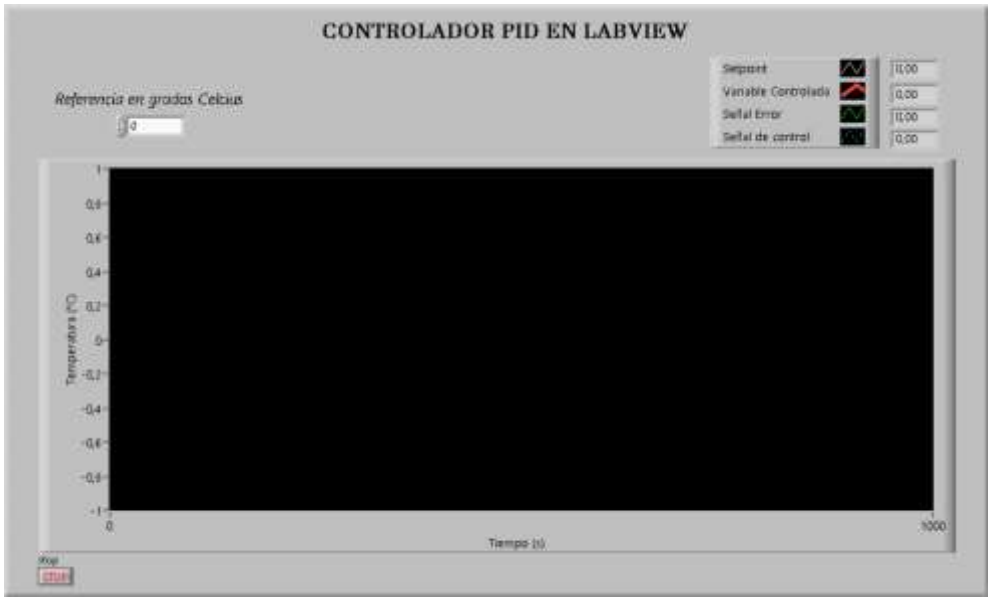


Figura 38. Interfaz gráfica controlador PID LabView

En la Figura 39 se muestra la respuesta del sistema utilizando un controlador PID sintonizado con la herramienta de Matlab, "pidTuner". En este ensayo se llevó al proceso hasta un punto de operación estable arbitrario y se dio tres escalones con amplitud de 5 grados y una duración aproximada de 600 segundos. Podemos observar la acción del controlador en la curva de color azul.

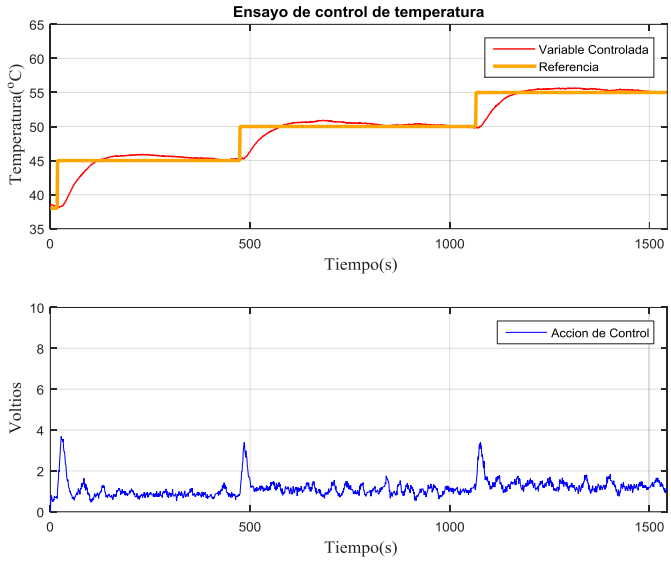


Figura 39. Ensayo de control de temperatura usando un controlador PID implementado en LabVIEW

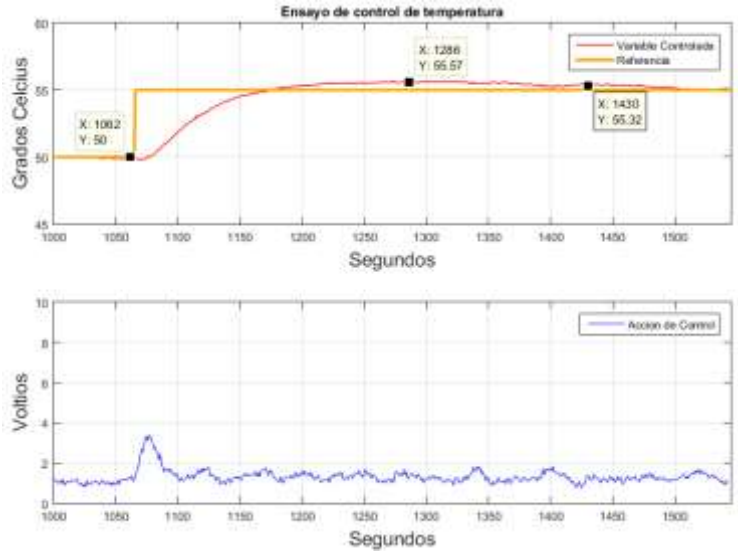


Figura 40. Repuesta del sistema con cambio de referencia de 50 a 55 grados Celsius

En la Figura 40 se puede observar la respuesta del sistema al cambiar la referencia de 50 a 55 grados Celsius, el sobrepaso máximo es del 1.03% y el tiempo de establecimiento es de 330 segundos, lo cual es muy cercano a lo que se obtuvo en la simulación realizada con PID Tuner.

1.3.3 Implementación de un controlador PID con módulo TK4S-T4RC

El módulo TK4S-T4RC es un dispositivo diseñado por Autonics específicamente para el control de temperatura. Este módulo puede generar una señal de control de tipo ON-OFF o de tipo variable expresada como corriente en el rango 4-20mA. El dispositivo controlador se puede observar en la Figura 41. Este consta de 4 teclas para la configuración.

Figura 41. Módulo Autonics TK4S-T4RC

Para nuestras pruebas hemos configurado el módulo para que reciba una señal de temperatura expresada en voltios en la escala de 0-10 voltios que corresponde a 0-100 grados Celsius y una señal de control expresada como corriente en la escala 4-20 miliamperios que corresponde a 0-100% de la potencia entregada.

Para configurar el módulo acorde a nuestras necesidades es recomendable restaurar los valores por defecto del dispositivo, para ello mantenemos pulsadas las teclas de navegación hasta que visualicemos “Init” (Inicializar) seleccionamos “yes” y confirmamos pulsando la tecla “mode” (véase la Figura 42).

Figura 42. Restaurar valores por defecto

La configuración se realiza modificando los valores de los 5 grupos de parámetros del módulo. Iniciamos ingresando al grupo de parámetros 1 (véase la Figura 43), para ello mantenemos pulsada la tecla “Mode” durante 3 segundos hasta visualizar “Par1” (Grupo de parámetros 1).

Figura 43. Grupo de parámetros 1

Pulsamos la tecla "Arriba" hasta ubicarnos en "Par3" (Grupo de parámetros 3) y pulsamos "mode" para acceder a los parámetros del grupo 3 (véase la Figura 44).

Figura 44. Grupo de parámetros 3

Dentro de la sección "In-t" (Entrada), pulsamos "abajo" hasta visualizar "Au1" (Entrada analógica 0-10v) y guardamos cambios con "mode" (véase la Figura 45).

Figura 45. Menú de opciones de entrada

Pulsamos mode hasta visualizar "L-rG" (límite inferior de la entrada analógica) y establecemos su valor en 00.00 (véase la Figura 46).

Figura 46. Límite inferior de la entrada analógica

Nos ubicamos en "H-rG" (límite superior de la entrada analógica) y establecemos su valor en 10.00 (véase la Figura 47).

Figura 47. Límite superior de la entrada analógica

Pulsamos "mode", hasta visualizar "dot" (punto decimal) y establecemos su valor 0.0 (véase la Figura 48).

Figura 48. Ubicación del punto decimal

Pulsamos "mode" y nos ubicamos en "L-SC" (límite inferior de la escala) y establecemos su valor en 000.0 (véase la Figura 49).

Figura 49. Límite inferior de la escala de la entrada

Pulsamos "mode" y nos ubicamos en "H-SC" (límite superior de la escala) y establecemos su valor en 100.0 (véase la Figura 50).

Figura 50. Límite superior de la escala de la entrada

Pulsamos "mode" y nos ubicamos en "d.Unt" (unidades mostradas) y establecemos su valor en °C (véase Figura 51).

Figura 51. Unidades mostradas en el indicador

Pulsamos “mode” hasta visualizar “o-Ft” (tipo de control) y establecemos su valor en “PI d” (véase Figura 52).

Figura 52. Comportamiento de la salida del dispositivo

Algoritmo de auto sintonía

Se ha utilizado el algoritmo de auto sintonía del módulo para establecer el comportamiento del controlador, para ello se llevó al sistema a un punto estable de operación y se ejecutó el algoritmo con una referencia de 55 grados Celsius, una vez alcanzada la estabilidad se dio una referencia de 65 grados Celsius.

Para ejecutar este proceso debemos mantener pulsado “mode” durante 3 segundos e ingresamos al grupo de parámetros 5 “Par5” (véase Figura 53).

Figura 53. Grupo de parámetros 5

Pulsamos “mode” hasta visualizar “USEr” (Nivel de Usuario) y establecemos su valor en “HIGH” (véase Figura 54).

Figura 54. Nivel de usuario

Mantenemos pulsado la tecla "mode" durante 3 segundos para regresar al menú anterior, y nos ubicamos en el grupo de parámetros 4 (véase Figura 55).

Figura 55. Grupo de parámetros 4

Pulsamos la tecla "mode" hasta visualizar "Al-1" (Operación de la salida de alarma 1) y establecemos su valor en "off" (véase Figura 56).

Figura 56. Comportamiento de alarma 1

Pulsamos “mode” hasta visualizar “Ao.ñ2” y establecemos su valor en “H-ñu” (véase Figura 57).

Figura 57. Usar salida 2 como salida de corriente

Mantenemos pulsado la tecla “mode” durante 3 segundos para regresar al menú anterior (véase Figura 58).

Figura 58. Grupo de parámetros 1

Mantenemos pulsado la tecla “mode” durante 3 segundos para regresar al estado de espera (véase Figura 59).

Figura 59. Pantalla principal

Utilizando las flechas establecemos una referencia entre 30 y 50 ℃ (véase Figura 60).

Figura 60. Establecer una referencia

Pulsamos la tecla "mode" durante 3 segundos y nos ubicamos en el "par2" (grupo de parámetros 2).

Figura 61. Grupo de parámetros 2

Pulsamos la tecla "mode" hasta visualizar "At" (Autotuning) y establecemos su valor en "on" (véase Figura 62).

Figura 62. Activar autotuning

Esperamos unos minutos hasta que se termine el proceso de autotuning (véase Figura 63).

Figura 63. Pantalla principal con autotuning activado

Monitoreo de señales con LabVIEW

Para monitorear las señales en tiempo real se diseñará un VI en LabVIEW, se utilizará la tarjera myRIO 1900 para la adquisición de la señal de control y la de temperatura. Se ha optado por esta solución debido a su facilidad de implementación.

Abrimos LabVIEW e iniciamos un nuevo proyecto (véase Figura 64).

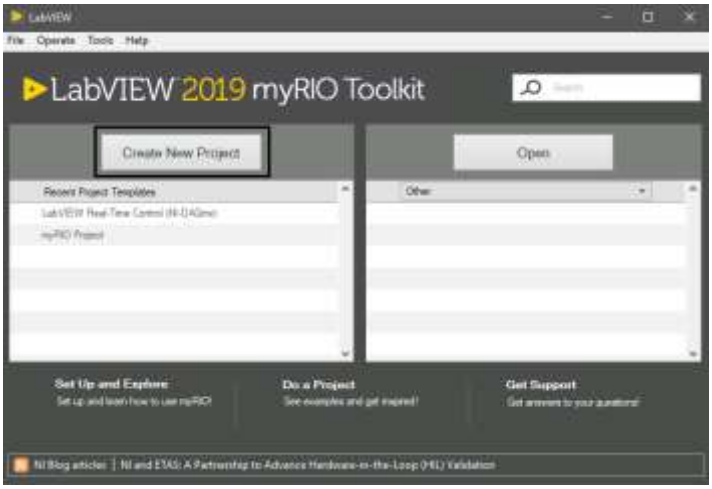


Figura 64. Ventana de inicio LabVIEW 2019

Creamos un proyecto en blanco

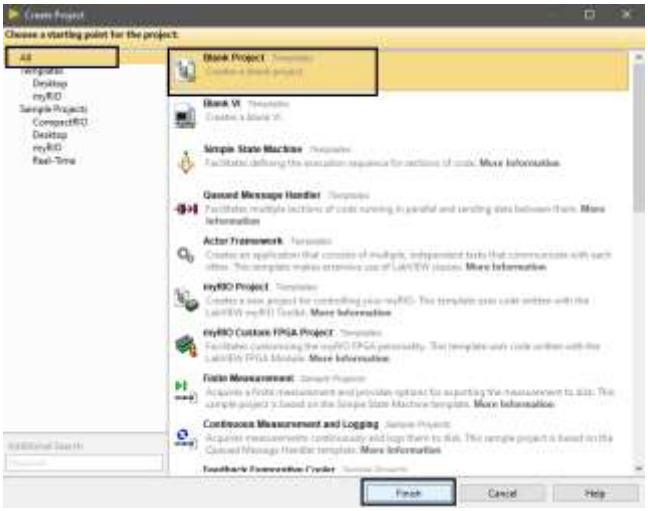


Figura 65. Crear nuevo proyecto en blanco

En el explorador del proyecto, damos clic derecho en la carpeta raíz, seleccionamos "New" (Nuevo) y damos clic en "Targets and Devices" (Objetivos y dispositivos) (Véase Figura 66).

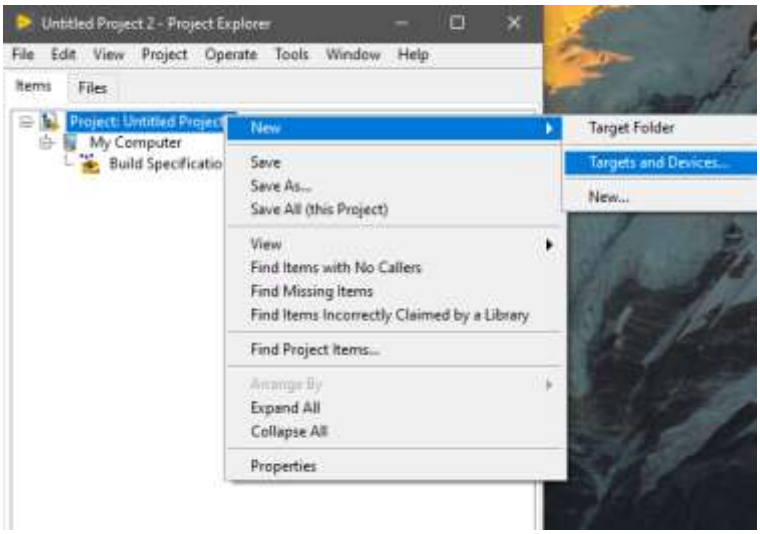


Figura 66. Agregar un dispositivo

Seleccionamos "New Target or Device" (Nuevo objetivo o dispositivo), desplegamos la carpeta myRIO, seleccionamos myRIO-1900 y damos clic en "OK" (véase Figura 67).

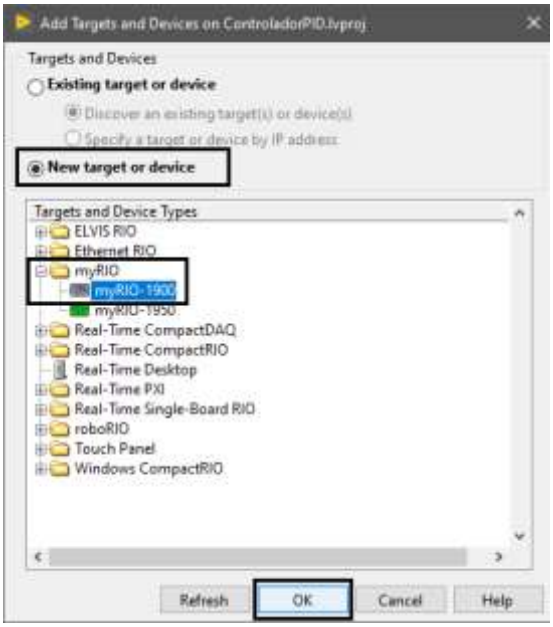


Figura 67. Agregar myRIO-1900

Damos clic derecho en "RT myRIO Target" (Dispositivo myRIO), seleccionamos "New" (Nuevo) y damos clic en "VI" (Interfaz Gráfica) (véase Figura 68).

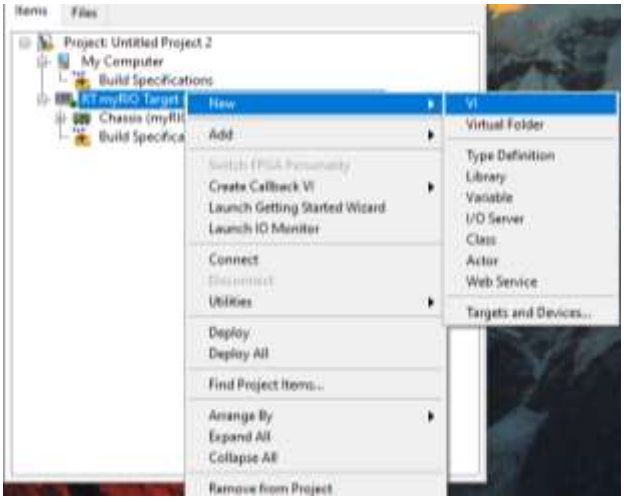


Figura 68. Agregar un VI

Agregamos una estructura while, una demora y un control para detener el proceso. Al agregar el bloque "Wait" (Esperar), aparecerá una ventana. Debemos configurar el "Counter units" (Unidades del contador) en milisegundos (véase Figura 69).

Figura 69. Estructura while

Damos clic derecho y accedemos al buscador de bloques de LabVIEW. (Véase Figura 70).

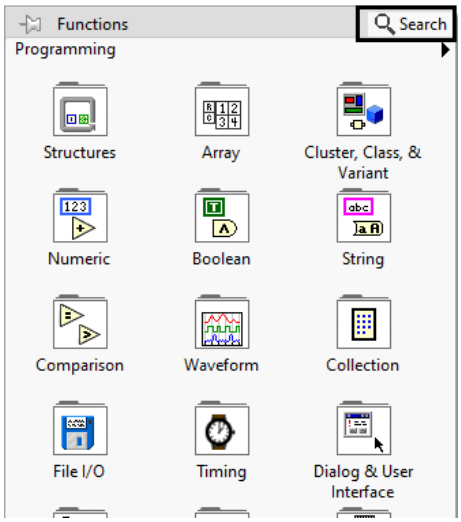


Figura 70. Acceder al buscador de LabVIEW

Usando el buscador agregamos el bloque "Analog input" (Entrada Analógica), al agregarlo aparecerá un asistente en el que modificaremos el nombre del bloque, así como las entradas físicas que vamos a utilizar, al finalizar pulsamos "OK". (Véase Figura 71).

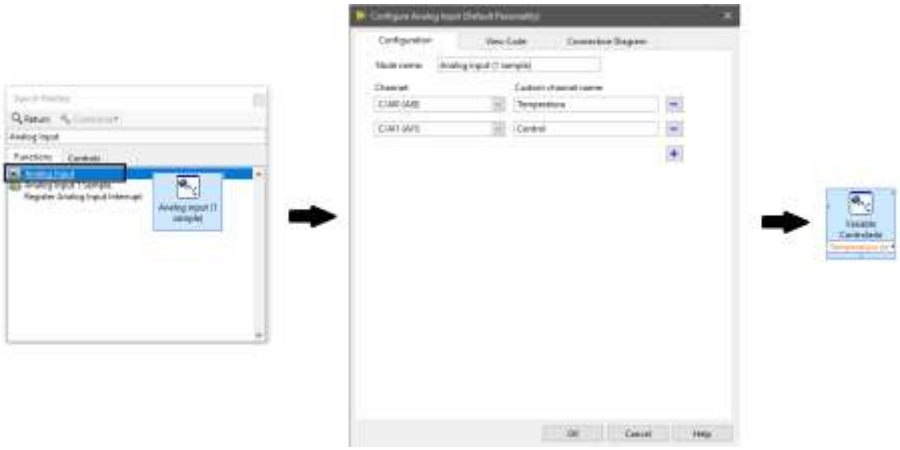


Figura 71. Configuración de entradas analógicas

Agregamos el bloque "Set Dynamic Data Atributes" (Establecer atributos de los datos) y modificamos sus parámetros (véase Figura 72).

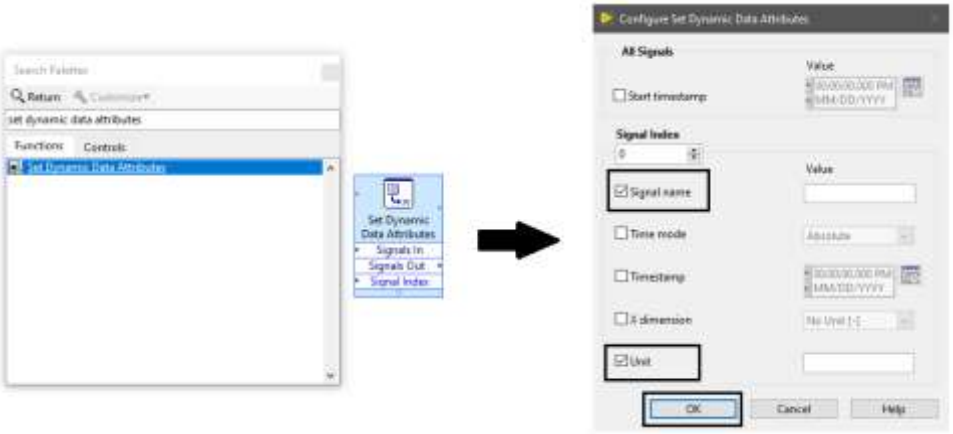


Figura 72. Configuración del bloque "Set Dynamic Data Atributes"

Conectamos los bloques "Set Dynamic Data Atributes" y añadimos los atributos requeridos en las señales (véase Figura 73)

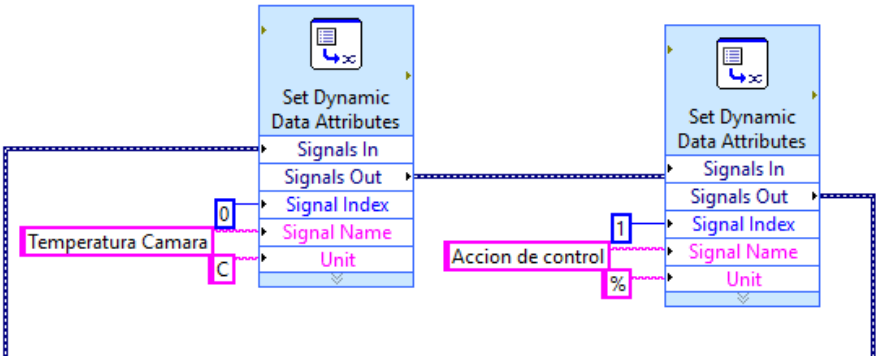


Figura 73. Conexión de bloques "Set Dynamic Data Atributes"

Agregamos el bloque "Write to measurement file" (Escribir a archivo de datos) (véase Figura 74).

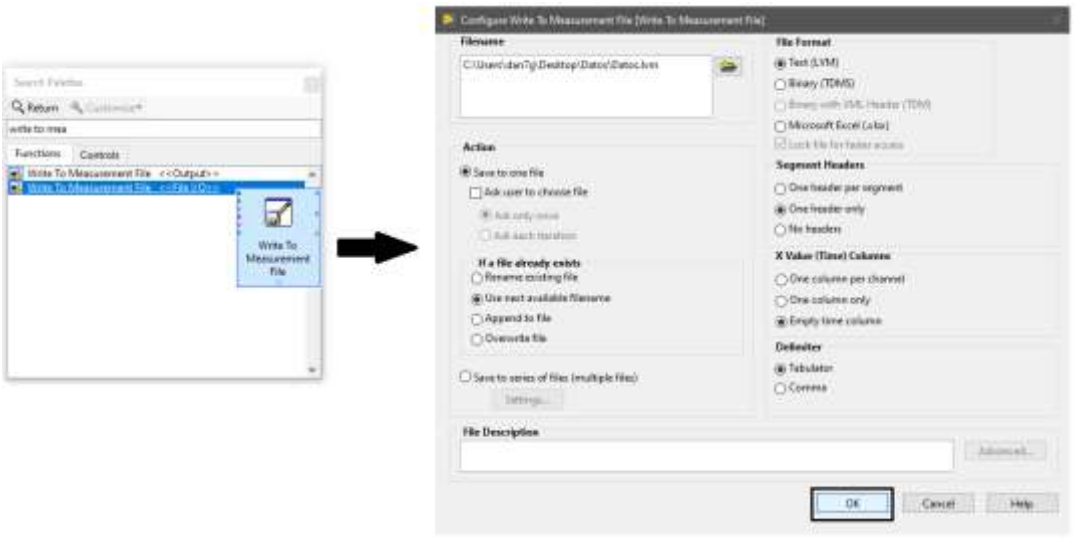


Figura 74. Agregar el bloque "Write to measurement file"

En la ventana de configuración seleccionaremos la dirección de destino del archivo de datos (véase Figura 75).

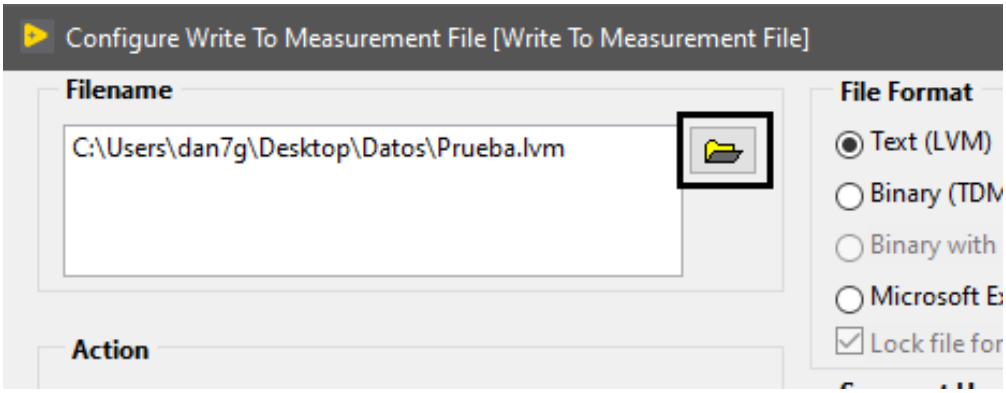


Figura 75. Selección de dirección de archivo de datos

Se debe asignar un nombre al archivo de datos (véase Figura 76)

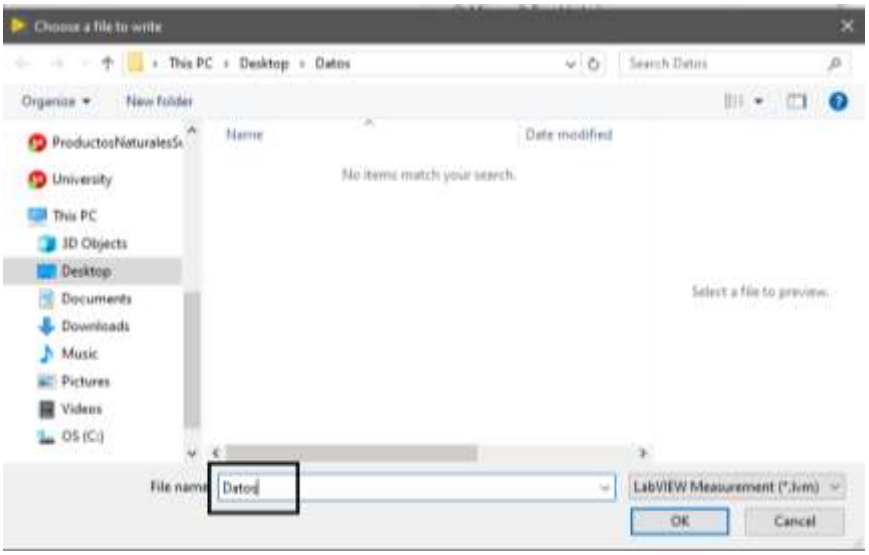


Figura 76. Nombre del archivo de datos

Configuramos los parámetros restantes (véase Figura 77).

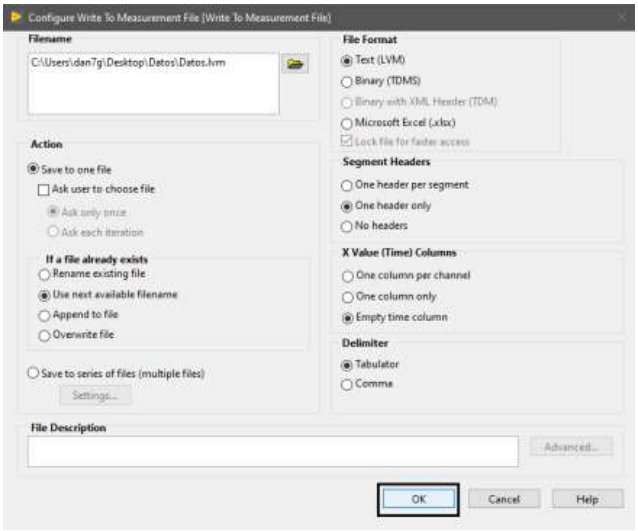


Figura 77. Configuración del bloque "Write to measurement file"

Utilizando el bloque "Merge Sygnals" agrupamos las señales de interés y las conectamos el conjunto como se muestra en la Figura 78.

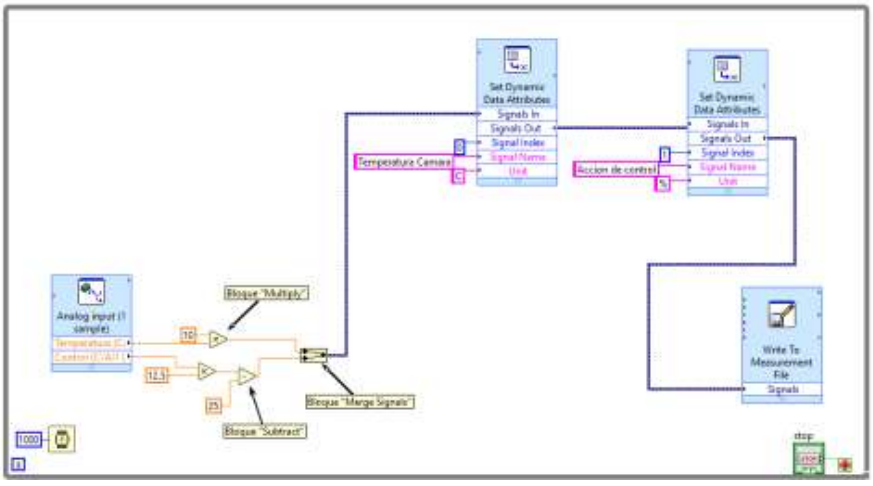


Figura 78. Conexión de secciones del programa

Hacemos clic derecho, nos ubicamos en la carpeta "Graph" y seleccionamos "Waveform Chart" (véase Figura 79).

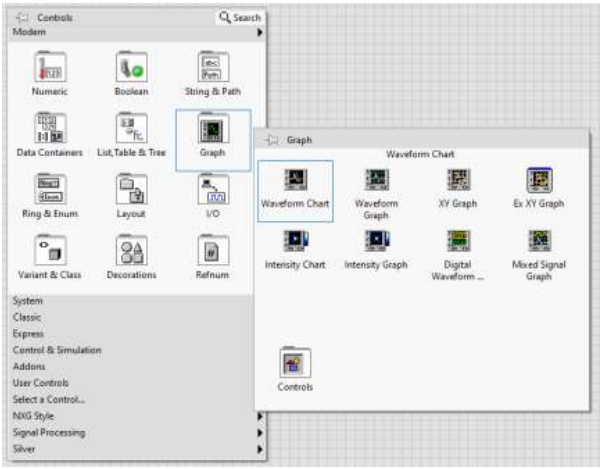


Figura 79. Agregar un "Waveform Chart"

En el diagrama de bloques conectamos el bloque "Waveform Chart" con las señales de interés (véase Figura 80)

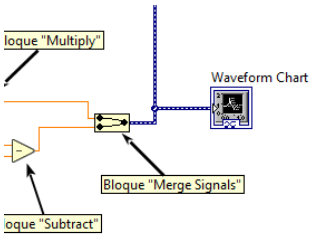


Figura 80. Conexión del bloque "Waveform Chart"

Cableado de dispositivos

El banco de trabajo en el que está montado el módulo TK4S T4RC tiene disponibles 8 pares de borneras, su función de detalla a continuación:

- 1, 2: Entradas digitales
- 3, 4: Salida 1 (Relé)
- 5, 6: Fuente de alimentación 100-240V
- 7, 8, 9: Salidas de control directo a cargas
- 10,11,12: Entradas Analógicas de sensor de temperatura
- 13, 14: Salida 2 (Relé o Salida de corriente)
- 15, 16: Salidas RS485.

El esquema de conexión se muestra a continuación (véase Figura 81).

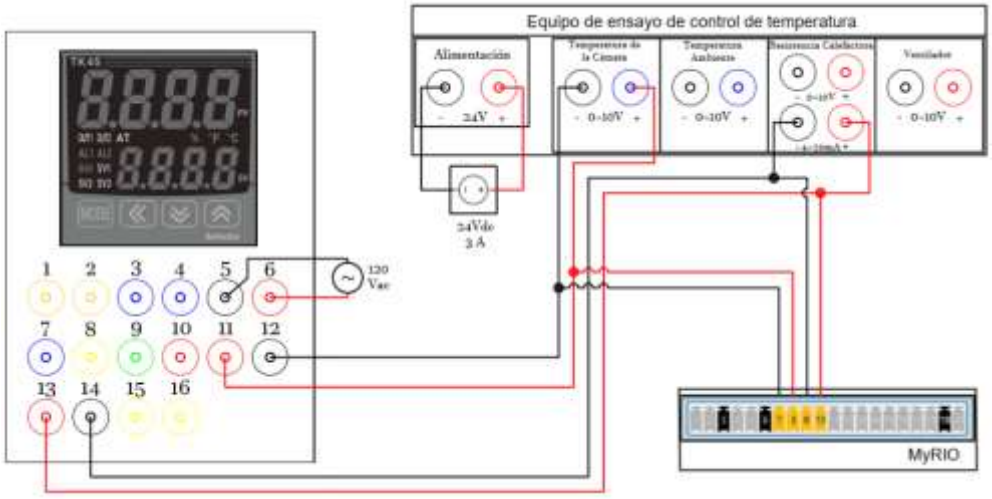


Figura 81. Diagrama de conexión

Al conectar Ni myRIO al PC aparecerá la ventana mostrada en la figura 45, debemos seleccionar en "Do Nothing" (No hacer nada). La dirección IP mostrada será utilizada más adelante.

Figura 82. Ventana de inicio de NI myRIO

En el explorador del proyecto, damos clic derecho en "RT myRIO target 0.0.0.0" y seleccionamos "Properties" (véase Figura 83).

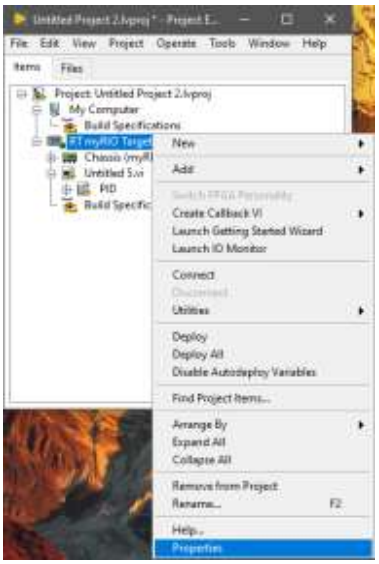


Figura 83. Propiedades de NI myRIO

En la sección “General” cambiamos la dirección IP con la mostrada anteriormente y damos clic en “OK” (véase Figura 84).

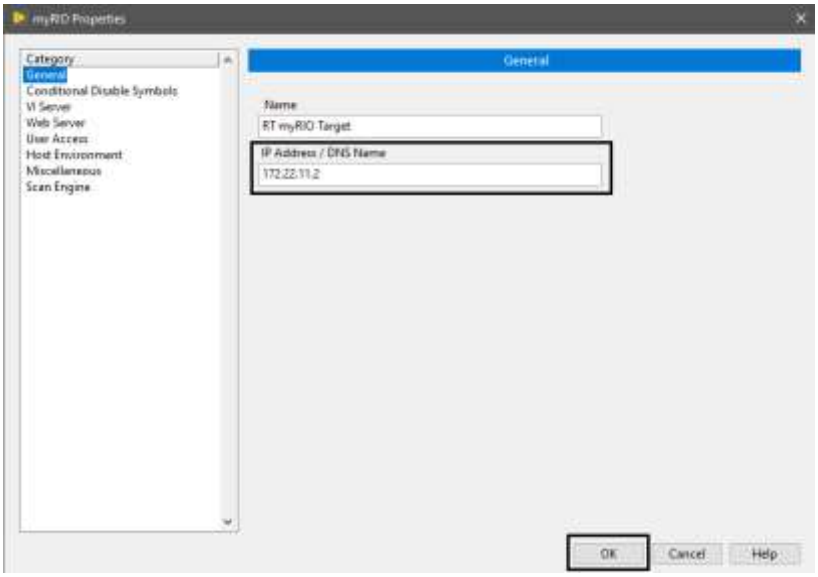


Figura 84. Asignar dirección IP

Ejecutamos el código cuando se desee capturar las señales (véase Figura 85).

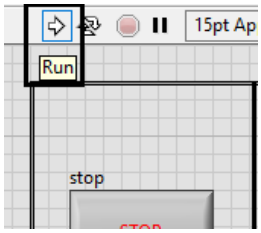


Figura 85. Ejecutar código

Para acceder al archivo de datos debemos utilizar un navegador de internet, donde ingresaremos la dirección Ip de la NI myRIO seguido de "/files" (véase Figura 86).

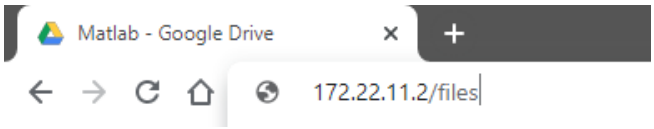


Figura 86. Ingresar dirección IP

En el campo de usuario ingresamos "admin" y dejamos el campo de contraseña en blanco (véase Figura 87).

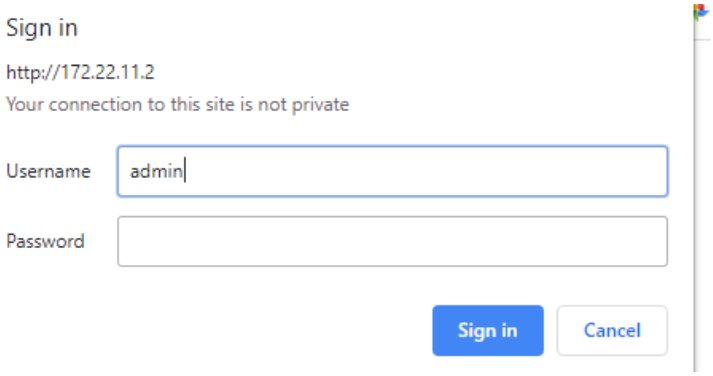


Figura 87. Ingresar credenciales

Accedemos al directorio seleccionado anteriormente y descargamos el archivo. En caso de que simplemente se muestre en el navegador debemos dar clic derecho sobre los datos y guardarlo como texto.

Grafica de resultados en Matlab

Abrimos Matlab r2017b y en hacemos clic en el botón "Import Data" (Importar datos). (Véase Figura 88)

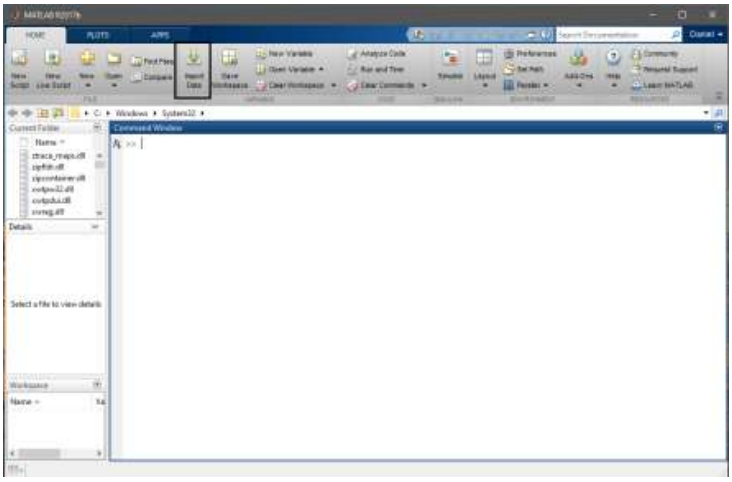


Figura 88. Importar datos

En la ventana emergente nos ubicamos en la pestaña “Import” (Importar) y seleccionamos el delimitador seleccionado en LabVIEW en este caso “Tab” (Tabulador).

Seleccionamos el rango de la matriz que contiene los datos junto con la fila en la que se encuentran los nombres de las variables

Seleccionamos el tipo de salida como “Column vectors” (Vectores Columna) y finalmente damos clic en “Import Selection” (Importar selección) (véase Figura 89).

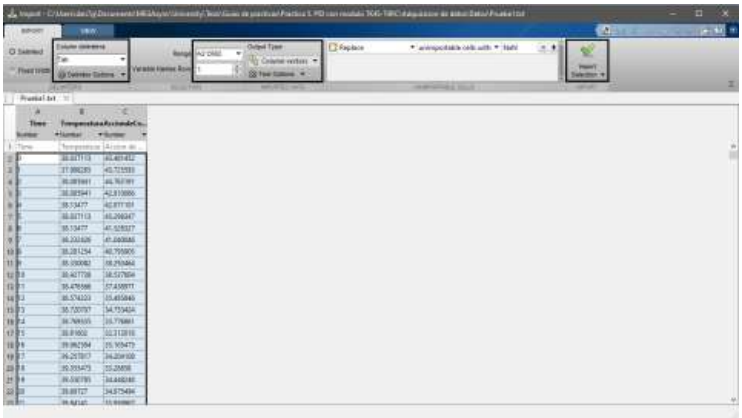


Figura 89. Configurar parámetros

Las variables aparecerán en el workspace

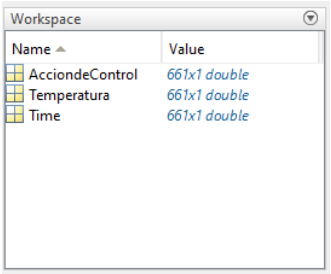


Figura 90. Variables en el workspace

Una vez ejecutado el algoritmo de auto sintonía se obtiene un tiempo de establecimiento de alrededor de 100 segundos sin sobrepaso.

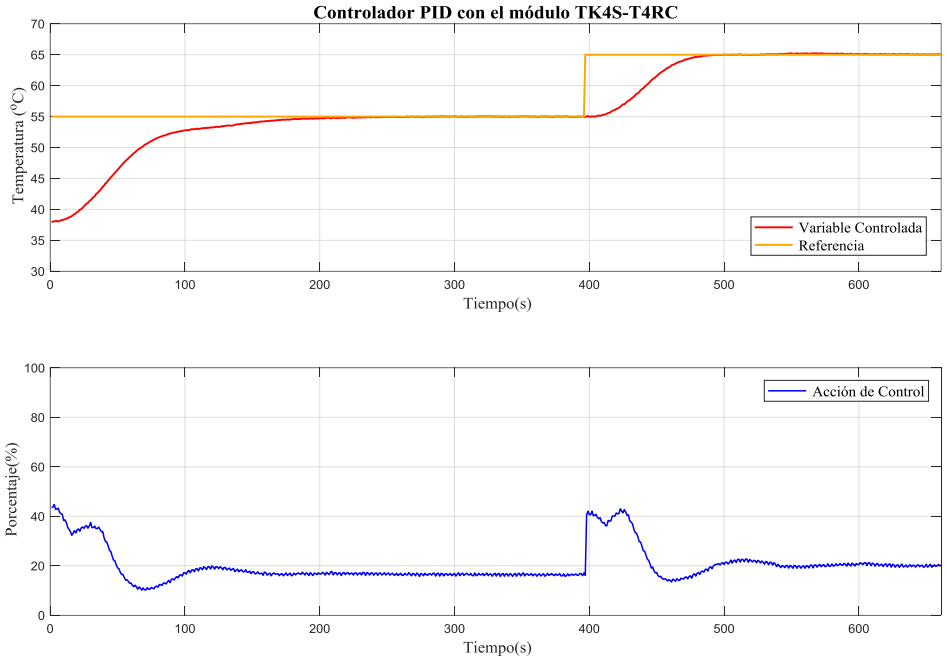


Figura 91.Ensayo de control de temperatura usando un controlador PID implementado en módulo TK4S-T4RC de Autonics

Los valores de las constantes que se obtienen luego de que se ejecute el algoritmo de auto sintonía son:

$$Kp = 7.023$$

$$Ki = 0.0921$$

$$Kd = 105.232$$

1.3.4 PID con auto sintonía utilizando un PLC S7-1200

Para la implementación de este controlador se utilizó el bloque PID_Compact del software TIA Portal, por lo cual se utilizó un PLC S7-1200 para adquisición de datos. La interfaz virtual se programa en el software LabVIEW. Se utilizó un servidor OPC del software KEPServerEX para la comunicación entre el PLC y LabVIEW. Finalmente se implementó el controlador en el prototipo de ensayos de control de temperatura. La guía detallada de la implementación se encuentra en el Apéndice A (guía de prácticas 4).

- **Programación en TIA Portal**

TIA Portal v13 contiene el objeto tecnológico PID_Compact que permite implementar controladores PID con una optimización integrada [12]. Se ha utilizado este bloque para llevar a cabo una auto sintonización del controlador PID con el equipo de ensayos.

1. *Agregando bloque PID_Compact*

Se agrega un bloque nuevo al programa de TIA Portal dando clic en "Add new block" (véase Figura 92). A continuación, en la ventana emergente se selecciona "Organization block" y luego "Cyclic interrupt" (véase la Figura 93), donde se coloca el tiempo del ciclo en 1000 ms, este valor se coloca para el tiempo de muestreo del controlador.

Figura 92. Nuevo bloque

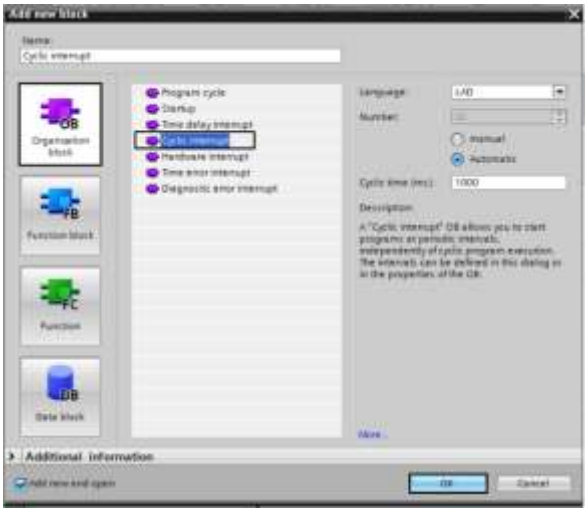


Figura 93. Configuración del bloque nuevo

A continuación del árbol del proyecto se añade un objeto (véase la Figura 94), donde se abrirá la ventana emergente “Add new object” y se selecciona “PID_Compact”, se escoge la versión “V1.0”, que es compatible con el dispositivo (véase la Figura 95).

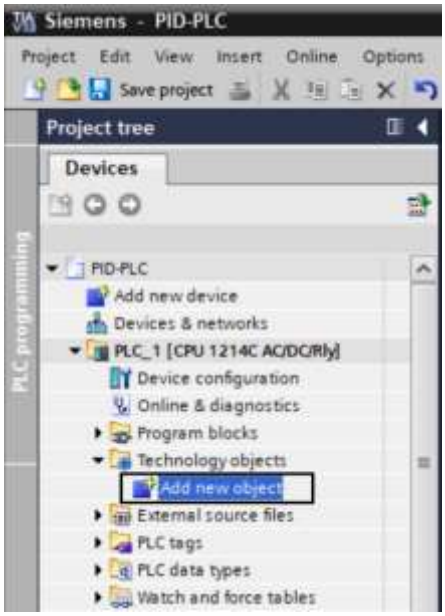


Figura 94. Añadiendo nuevo objeto

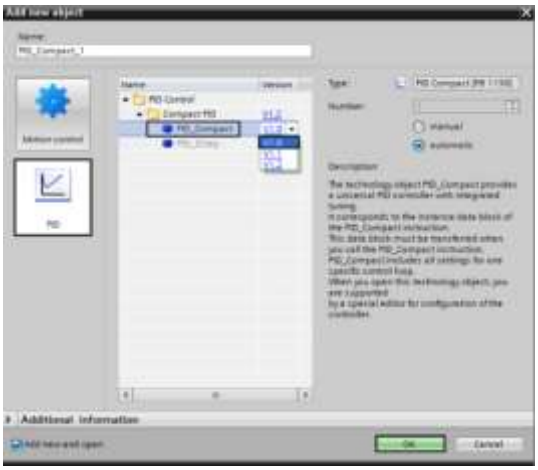


Figura 95. Ventana de configuración del objeto

Enseguida aparece las opciones del controlador en la pestaña “Functional view”. En el ítem “Controller type”, se despliega el menú de opciones y se escoge “Temperature” (véase Figura 96). Verificar que la escala de temperatura sea grados Celsius (ºC). En la opción “Process value limits” se debe colocar un rango de 0 a 100 ºC. Con este valor nos referimos al rango de 0 a 10 V de funcionamiento del prototipo

(véase la Figura 97). En la opción “Process value scaling” nos aseguramos de que los valores medidos de la cámara tengan un rango de 0 a100 ºC (véase Figura 98).

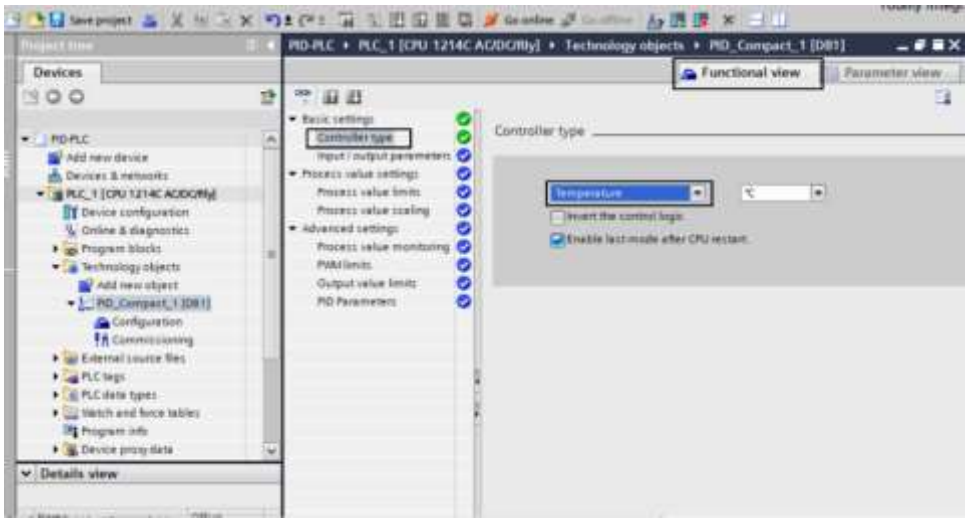


Figura 96. Selección del tipo de controlador

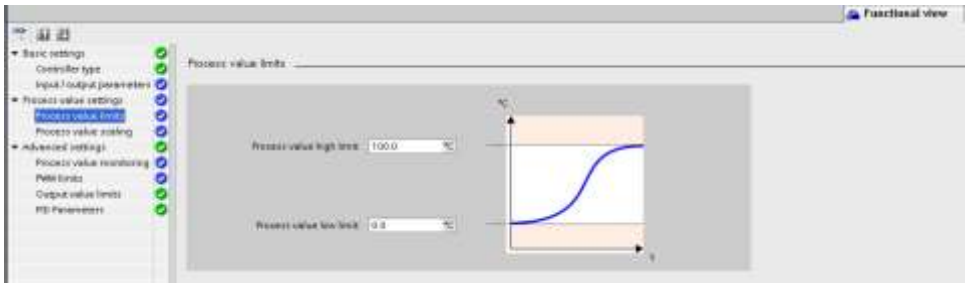


Figura 97. Límite del valor de proceso

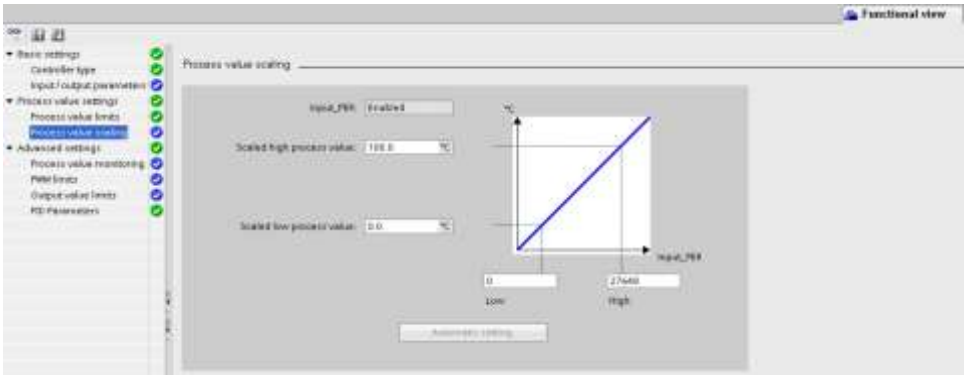


Figura 98. Escala del valor de proceso

En el bloque "Cyclic interrupt" agregado anteriormente, se añade el bloque "PID_Compact" de la pestaña de Instrucciones ubicado al lado derecho de la ventana (véase Figura 99). En seguida aparecerá la ventana "Call Options" donde se despliega el menú de "Name" y se escoge "PID_Compact_1" (véase Figura 100).

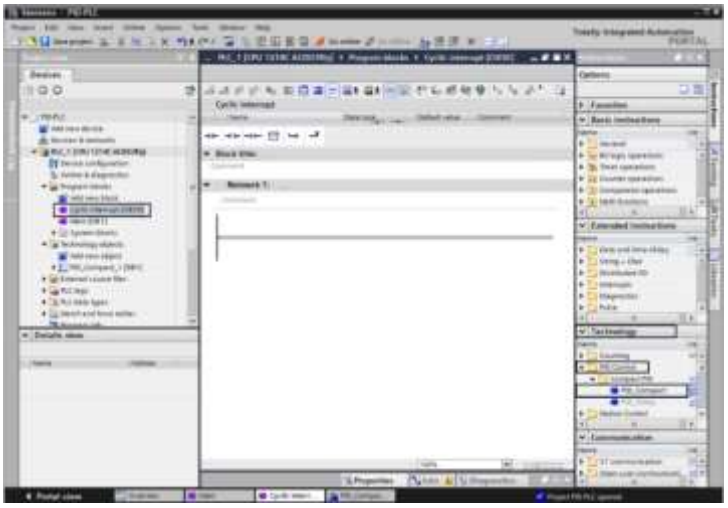


Figura 99. Agregando bloque PID_Compact

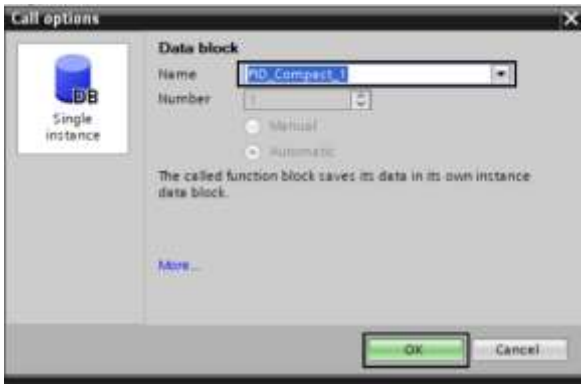


Figura 100. Nombre del bloque PID_Compact

La Figura 101 muestra el bloque PID_Compact agregado con sus respectivas entradas y salidas. A continuación, se describen los parámetros del bloque utilizados:

Setpoint: Establece la referencia del controlador. Se utilizó la variable %MD115 para establecer la comunicación OPC entre LabVIEW y el PLC.

Input_PER: Esta entrada permite valores de 0 a 10 V, por ello para la medición de la cámara del prototipo se ha utilizado la entrada analógica %IW66.

Output: Se utilizó la variable %MD1 para obtener la señal del controlador o acción de control, con la cual se comanda al equipo de ensayos mediante la salida analógica %QW80 y a su vez permite visualizar su comportamiento en LabVIEW mediante la comunicación OPC.

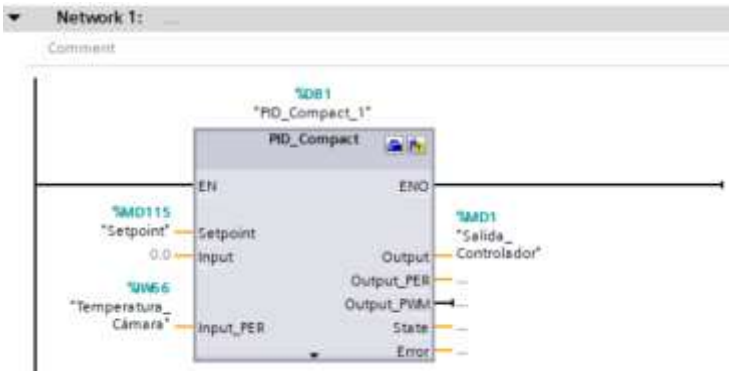


Figura 101. Bloque PID_Compact

A continuación, se procede a escalar la variable %MD1 (acción de control) con los bloques “NORM_X” y “SCALE_X”. El bloque “NORM_X” toma el valor real de %MD1 que está en un rango entre 0 a 100 y lo normaliza a un valor entre 0 a 1 (variable %MD2). Con el bloque “SCALE_X” se escala el valor normalizado a un valor entero que se encuentra en un rango definido por MIN y MAX del bloque en este caso 0 a 27648 que representa a la salida analógica %QW80 que se usa para la implementación del controlador es de 0 a 10 V (véase Figura 102).

Se obtiene la variable controlada mediante la entrada analógica %IW66, la cual admite señales de 0 a 10V. De la misma manera se escala la entrada analógica con los bloques “NORM_X” y “SCALE_X”. Con el bloque “NORM_X” se normaliza el valor entero de la variable de entrada que está en el rango de 0 a 27648 a un valor real entre 0 a 1 (variable %MD3). Este valor se escala a un rango entre 0 a 100 mediante el bloque “SCALE_X”. Este valor escalado es la variable %MD110 la cual se utiliza para el monitoreo de las señales véase Figura 102).

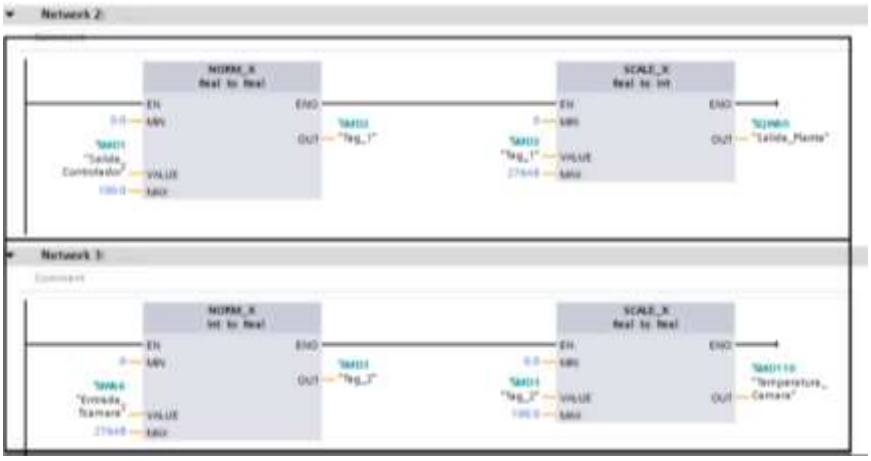


Figura 102. Bloques de variables para el monitoreo

2. Ejecución del bloque PID_Compact para la auto sintonización

Una vez ejecutado el programa y cargado en el PLC S7-1200 se selecciona el botón puesta en marcha del bloque (véase Figura 103). La Figura 104 muestra la ventana “PID_Compact_1”, en la cual se realiza la auto sintonización. Para empezar, se selecciona el tiempo de muestro de 1 segundo y se selecciona “Start” (véase Figura 105). Para llevar a cabo la auto sintonía se selecciona “Fine tuning” de las opciones de “Tuning mode”, enseguida se selecciona “Start” (véase Figura 106). Cabe recalcar que el valor de referencia para la auto sintonización es de 60 grados

Celsius y es establecido en el software LabVIEW, el cual se verá más adelante. Este proceso puede tardar varios minutos.

Figura 103. Puesta en marcha

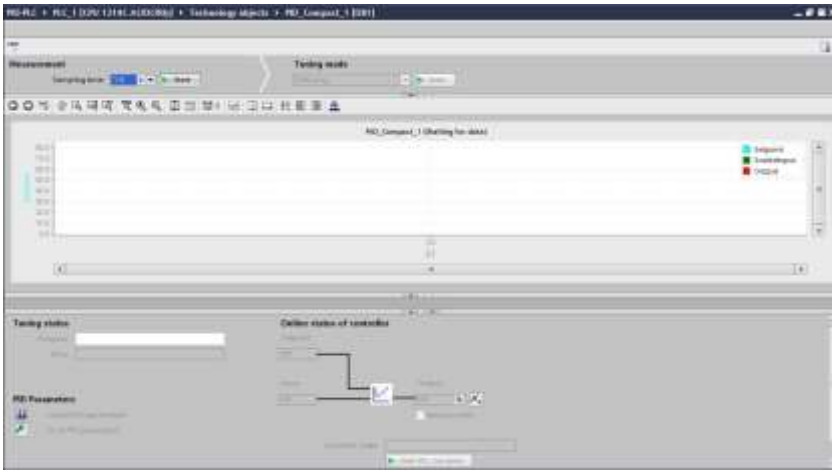


Figura 104. Ventana para auto sintonización del bloque PID_Compact

Figura 105. Inicio de lectura de variables

Figura 106. Inicio de autotuning

Cuando el bloque termina la auto sintonización se pueden observar los parámetros del controlador en “Go to PID parameters”, para utilizar las variables obtenidas por el bloque se selecciona la opción “Upload PID parameters” (véase Figura 107).

Figura 107. Parámetros de la auto sintonía

Las constantes del controlador obtenidas con la auto sintonización son las siguientes:

$$Kp = 6.937197$$

$$Ki = 0.11056$$

$$Kd = 110.23955$$

- **Implementación virtual en LabVIEW para el monitoreo y recolección de datos**

En el software de LabVIEW se crea un proyecto con las variables creadas en el software KEPServerEX (véase Apéndice A, guía de prácticas 4). La Figura 108 muestra el explorador de proyectos de LabVIEW con las variables a utilizar las cuales son SalidaControlador o acción de control, Setpoint o referencia y Tcamara o variable controlada.

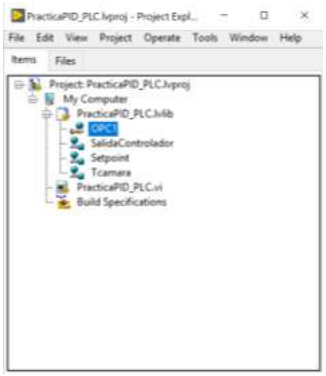


Figura 108. Variables creadas para el monitoreo y recolección de datos

Las variables agregadas se arrastran a la ventana “Block Diagram” donde se configura según el modo ya sea escritura o lectura. La variable Tcamara (variable controlada) y SalidaControlador (señal de control) se coloca en modo lectura (Read) y Setpoint (referencia) se coloca en modo escritura (Write) (véase Figura 109).

Figura 109. Configuración de las variables según su función

La Figura 110 muestra el arreglo de bloques para la lectura de los datos de la temperatura de la cámara y la señal de control. Además, se crea un control para colocar los valores de referencia. El bloque "Write To Measuremente File" se utiliza para almacenar los datos. El bloque "Waveform Chart" permite visualizar las gráficas en tiempo real del equipo de ensayos de temperatura.

El VI nos permite establecer la referencia o setpoint. Contiene indicadores que permite observar la temperatura de la cámara y a su vez muestra en la gráfica las señales de referencia, temperatura de la cámara y la señal del controlador (véase Figura 111).

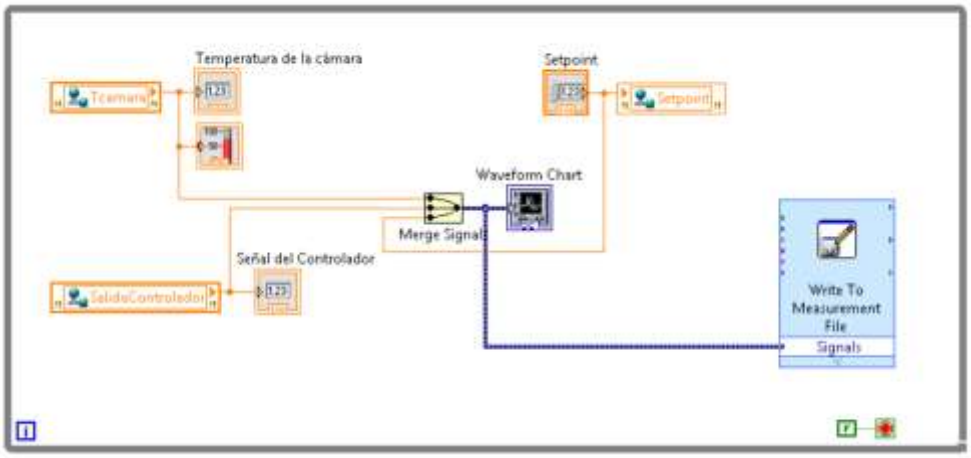


Figura 110. Programa para monitoreo y adquisición de datos.

Figura 111. Panel de control

La Figura 112 muestra el comportamiento del equipo de ensayo al implementarse la auto sintonía del controlador PID. Se aplicó tres escalones. El primer escalón se aplicó cuando la temperatura de cámara estaba a temperatura ambiente, se puede apreciar que a pesar de tener una gran amplitud el tiempo de establecimiento ha sido pequeño. Se dio un segundo escalón de bajada y luego un tercer escalón de subida para completar una correcta auto sintonización del controlador PID.

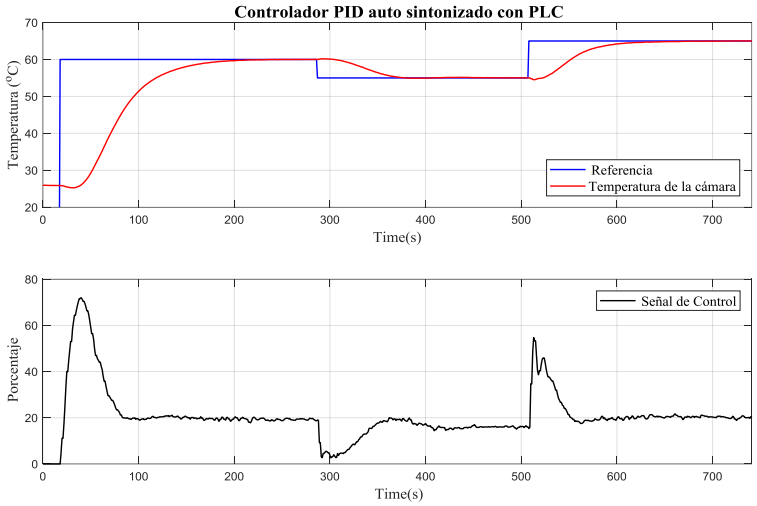


Figura 112. Ensayo de control de temperatura con PLC S7-1200 y controlador PID

Comparación de los esquemas de control propuestos.

A continuación, se muestra una tabla de resumen del comportamiento del sistema cuando se implementa un controlador PID con diferentes equipos. Cabe recalcar que la sintonización del controlador implementado en LabVIEW fue realizada manualmente mientras que en los otros dispositivos se utilizó un algoritmo de auto sintonía.

	LabVIEW	TK4S T4RC	PLC s7-1200
Mp (%)	<2%	<2%	<1%
Ts (s)	330	110	100
Kp	2.7849	7.023	6.937
Ki	0.02138	0.0921	0.11056
Kd	90.6567	105.232	110.239

El dispositivo que obtuvo el mejor comportamiento fue el PLC s71200, en este caso se obtuvo el menor tiempo de establecimiento y el sobrepaso de menor proporción, esto se logra gracias al algoritmo de auto sintonía que utiliza siemens. Cabe mencionar que TIA Portal ofrece más opciones de configuración que el módulo de Autonics por lo que no sorprende que se consigan mejores resultados.

CAPÍTULO 4: CONTROLADOR EN ESPACIO DE ESTADOS

En este capítulo se presenta un controlador en espacio de estados que se implantará en el prototipo de proceso térmico. En primera instancia se hablará sobre el controlador y las características que se debe tomar en cuenta en su diseño. Luego se describirá el diseño como tal del controlador y por último se detallará la implementación en el prototipo.

2.1 Controlador en espacio de estados por el método de asignación de polos

Esta estructura de controlador tiene la capacidad de ubicar los polos en lazo cerrado del sistema en cualquier posición mediante una matriz de ganancias de la realimentación del estado, siempre y cuando el sistema sea completamente controlable y observable. En la Figura 113 se muestra el diagrama del controlador y observador.

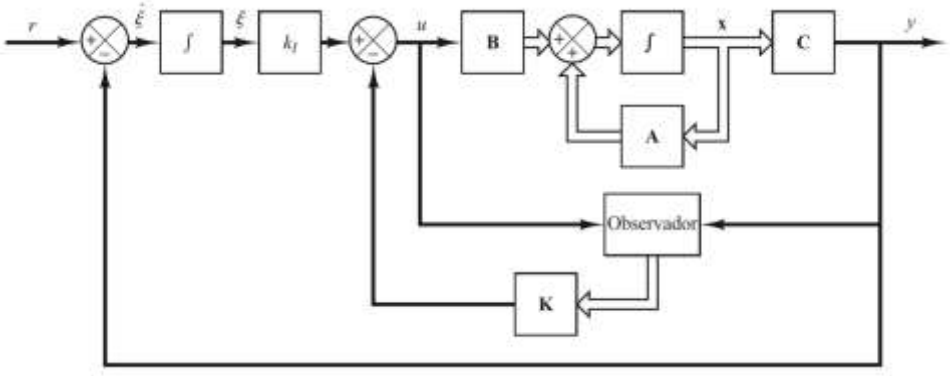


Figura 113. Esquema general de un controlador en espacio de estados con observador e integrador embebido

El primer paso en el diseño del controlador es seleccionar la ubicación de los polos en lazo cerrado. El método más utilizado consiste en seleccionar dos polos dominantes y ubicar los otros polos lo suficientemente alejados hacia la izquierda de modo que no tengan mayor influencia en el sistema. Por contrapartida mientras más alejados estén los polos, mayor será la señal de control. [13]

Una vez seleccionados los polos del controlador, debemos seleccionar los polos del observador. Por norma general los polos del observador deben ser de entre 2 a 5 veces más rápidos que los del controlador [13].

Una vez seleccionada la ubicación de los polos debemos calcular la matriz de ganancias del controlador y el observador, esto ha sido realizado utilizando el software Matlab.

2.1.1 Implementación de un controlador en espacio de estados por asignación de polos

Para la implementación de este controlador es necesario un computador con Matlab r2017b o superior, el equipo NI myDAQ y el prototipo de ensayos de control de temperatura.

La técnica de diseño empieza con la determinación de los polos en lazo cerrado deseados a partir de los requerimientos establecidos por el diseñador tales como velocidad, razón de amortiguamiento, o ancho de banda, al igual que los requisitos en estado estacionario. Para este ejemplo hemos dispuesto lo siguiente:

- Tiempo de establecimiento

$$ts = 650\ s$$

- Sobre impulso máximo

$$Mp = 8\%$$

Para calcular el coeficiente de amortiguamiento utilizamos la ecuación 2.

$$Mp = e^{\frac{-\pi\xi}{\sqrt{1-\xi^2}}} \qquad [2]$$

$$0.08 = e^{\frac{-\pi\xi}{\sqrt{1-\xi^2}}}$$

$$\xi = 0.62657$$

Para calcular la frecuencia natural utilizamos la ecuación 3.

$$ts = \frac{4}{\xi w_n} \qquad [3]$$

$$650 = \frac{4}{0.6265 w_n}$$

$$w_n = 0.009822$$

Los polos dominantes del sistema se obtienen al resolver la ecuación 4.

$$s^2 + 2\xi w_n s + w_n^2 = 0 \qquad [4]$$

$$s^2 + 0.0123s + 0.00009647 = 0$$

$$s_{1,2} = -0.00615 \pm 0.007658j$$

A pesar de que se ha asumido un modelo de tercer orden para el proceso térmico, es suficiente con calcular los dos polos más lentos ya que estos regirán el comportamiento del sistema, en este caso se trata de dos polos complejos conjugados. El polo restante debe estar relativamente alejado del eje jw, para disminuir su influencia.

Para asegurar las condiciones en estado estacionario se ha incrementado un integrador. Los polos no dominantes del sistema se muestran a continuación.

$$s_{3,4,5} = -0.123$$

Debido a que las variables de estado del sistema no están disponibles, es necesario la implementación de un observador de estados de orden completo. Por norma general los polos del observador deben ser de dos a cinco veces más rápidos que los del controlador. Para este ejemplo se han escogido los siguientes polos:

$$s_{1,2,3} = -0.3075$$

El código necesario para el cálculo de las matrices de ganancias, con el integrador embebido, se muestra a continuación:

```
%%
% Integrador
Ai=[A zeros(4,1);-C 0];
Bi=[B;0];

% Matriz de ganancias del controlador
pC=[-0.00615+0.007658i -0.00615-0.007658i -0.123 -0.123 -0.123];
k=acker(Ai,Bi,pC)

% Matriz de ganancias del observador
pO=[-0.3075 -0.3075 -0.3075 -0.3075];
ke=acker(A',C',pO)'
```

Donde A, B y C son las matrices del sistema:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2.1671e-06 & -0.0008 & -0.0081 & -0.1321 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$C = [5.8071e-06 \quad -4.9360e-05 \quad 0 \quad 0]$$

$$D = [0]$$

- Las matrices resultantes son:

$$k = [-2.0989 \cdot 10^{-6} \quad -0.0008 \quad -0.0071 \quad -0.0829 \quad -7.5683 \cdot 10^{-5}]$$

$$ke = \begin{bmatrix} -4.1783 \cdot 10^{3} \\ -306.6581 \\ -11.8833 \\ 7.1119 \end{bmatrix}$$

El esquema del controlador se muestra en la Figura 114.

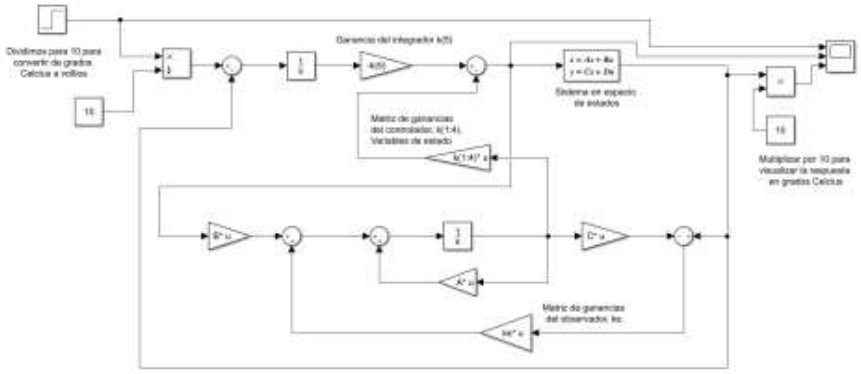


Figura 114. Esquema de asignación de polos por realimentación de estados con integrador embebido

Para la implementación del controlador es necesario incluir el bloque "Real-Time Sync" y sustituir el sistema en espacio de estados por una entrada y salida analógica.

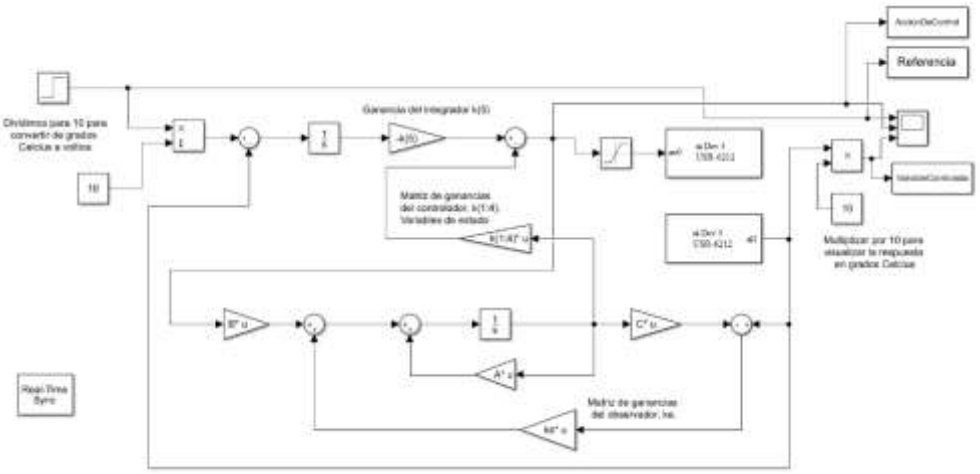


Figura 115. Esquema de asignación de polos por realimentación de estados con integrador embebido con entradas y salidas físicas.

En la Figura 116 se muestra la configuración del bloque "Real Time Sync"

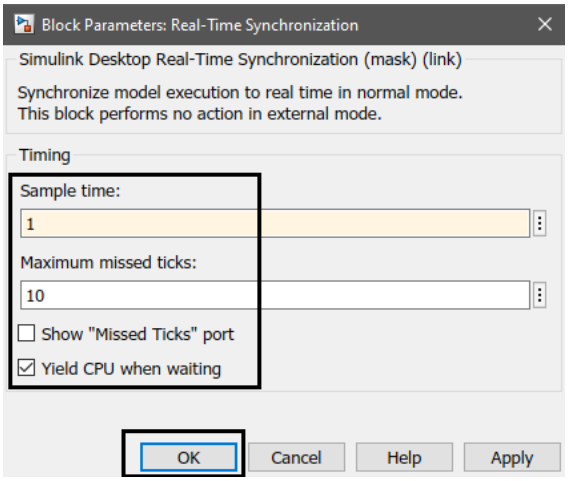


Figura 116. Configurar bloque Real-Time Synchronization

En la Figura 117 se muestra la respuesta del sistema con un controlador en espacio de estados con integrador embebido. En este ensayo se lleva al sistema a un punto estable de operación y se le da una referencia, en este caso de 34.5 grados Celsius.

El sistema tiene un tiempo de establecimiento de 662 segundos y un sobrepaso máximo del 8%, lo cual cumple con los parámetros de diseño establecidos previamente.

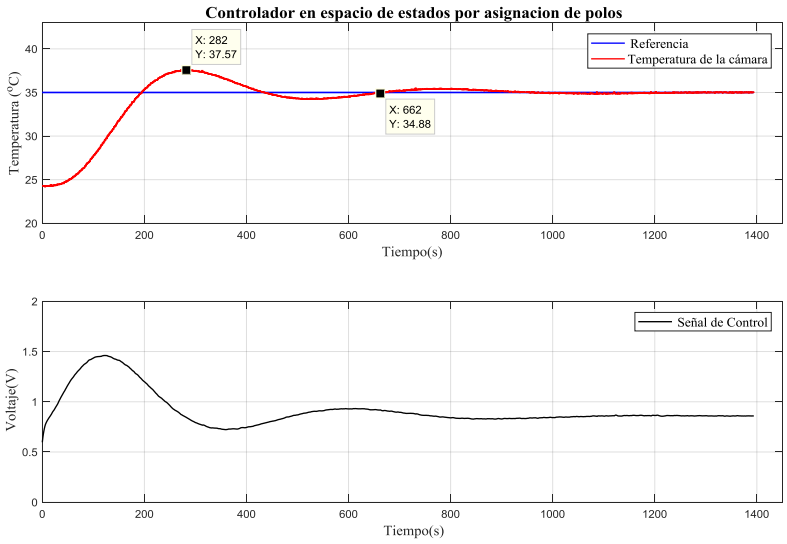


Figura 117. Ensayo de control de temperatura con controlador en espacio de estados

La intencionalidad de este proyecto de titulación no es comparar los controladores PID y en espacio de estados. Más bien lo que se pretende es dar una guía de implementación de estos controladores utilizando el proceso térmico de laboratorio. En este sentido, se debe tener en cuenta que, es altamente posible que se pueda obtener un mejor desempeño en cuanto al controlador en espacio de estados, tomando como referencia lo obtenido en la Figura 118.

CAPÍTULO 5: CONTROL PREDICTIVO BASADO EN MODELO

3.1 Control predictivo basado en modelo (MPC)

El control predictivo basado en modelo es una técnica de control que calcula valores futuros de la señal de control utilizando información pasada y presente de un sistema, para este cálculo el controlador se basa en el modelo matemático de la planta [14] [15]. Para un determinado periodo de tiempo N denominado horizonte de predicción se calculan los valores futuros en cada instante k_i [16]. MPC define la acción de control minimizando una función de costos que describe el comportamiento deseado del sistema. Esta función compara la salida anticipada del sistema con la referencia. Por lo general se permite un determinado valor de variaciones futuras de la señal de control, a estas variaciones se le conoce como horizonte de control [15]. Para cada tiempo de muestreo, el controlador calcula un vector con la acción de control la cual minimiza la función de costo y su dimensión depende del horizonte de control, el primer elemento del vector es que será enviado a la planta [17] [16]. Se puede tiene una mejor apreciación de cómo funciona el control predictivo en la Figura 118.

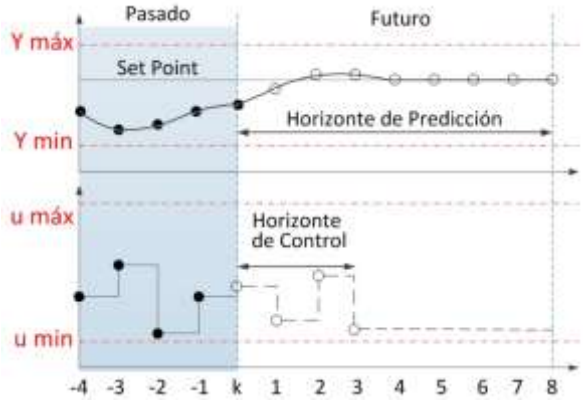


Figura 118. Representación del control predictivo [16]

3.1.1 Control Predictivo basado en el Modelo (MPC) implementado en Simulink con NI myDAQ

Para la implementación del controlador MPC se ha utilizado un computador con la herramienta Simulink del software Matlab, el equipo NI myDAQ para adquisición de datos y el prototipo de ensayos de control de temperatura. La guía detallada se encuentra en el Apéndice A, guía de prácticas 7.

- **Diseño del controlador MPC y programación en la herramienta Simulink**

Matlab provee la herramienta MPC Designer, la cual permite elaborar un controlador MPC a partir de la planta del sistema. Para este trabajo, el diseño del controlador MPC se realiza con la planta del sistema identificada anteriormente en el Capítulo 2, la cual se encuentra expresada en la ecuación (5).

Para el diseño del controlador se obtiene el modelo en espacio de estados y se discretiza con el programa que se encuentra en el Apéndice A, guía de prácticas 7. Se importa la planta seleccionado "MPC Structure" en la herramienta MPC Desingner (véase Figura 119). A continuación, se selecciona el modelo de la planta y se coloca el tiempo de muestreo de 1 segundo, los demás campos se dejan con los valores por default (véase Figura 120). La Figura 121 muestra las gráficas de la respuesta de entrada y salida.

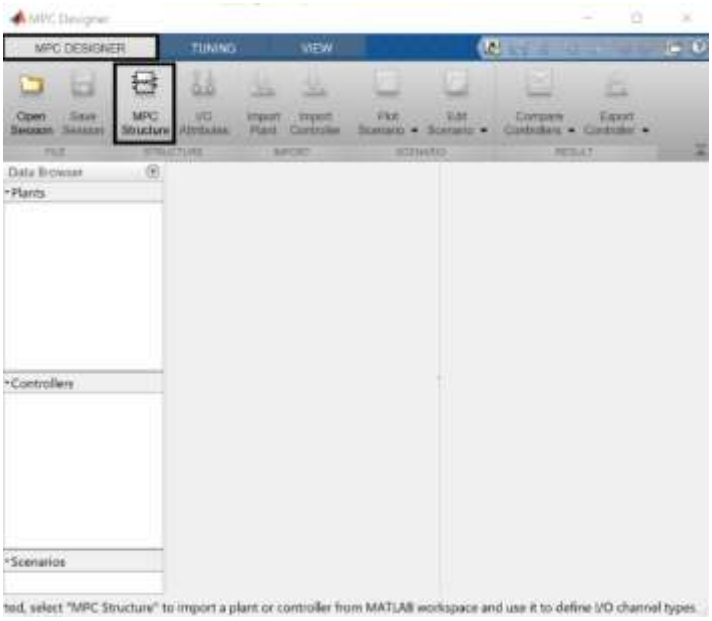


Figura 119. Importando datos para diseño del controlador

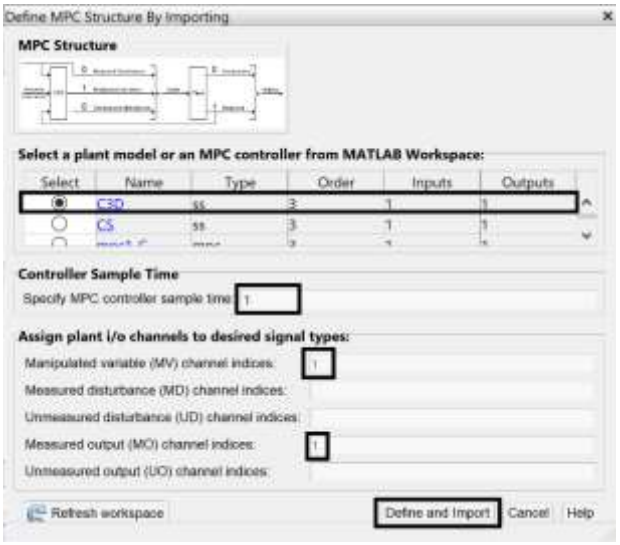


Figura 120. Selección de datos para el diseño del controlador

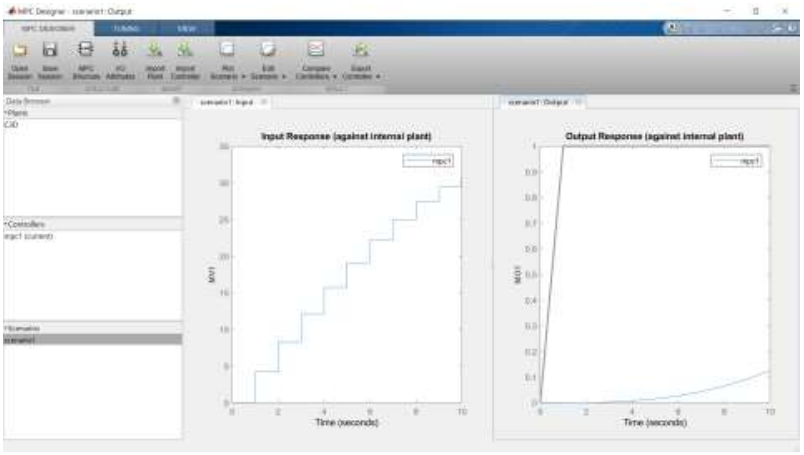


Figura 121. Planta ingresada

Se edita el tiempo de simulación del controlador en la opción “Edit Scenario” y se coloca un tiempo de 500 segundos (véase Figura 122). Se selecciona “I/O Atributes” y se coloca 0.5 en el valor nominal de entrada “u(1)” y 2.6 en el valor nominal de salida “y(1)” (vease Figura 123). Estos valores se colocan puesto que en los ensayos para la identificacion del sistema se ha concluido que con el valor de 0.5 voltios en la senal de la resistencia calefactora, el sensor de temperatura marca un voltaje de 2.6 voltios. Con un valor menor a 0.5 voltios, la resistencia calefactora no se expone a grandes perturbaciones.

Figura 122. Tiempo de simulación

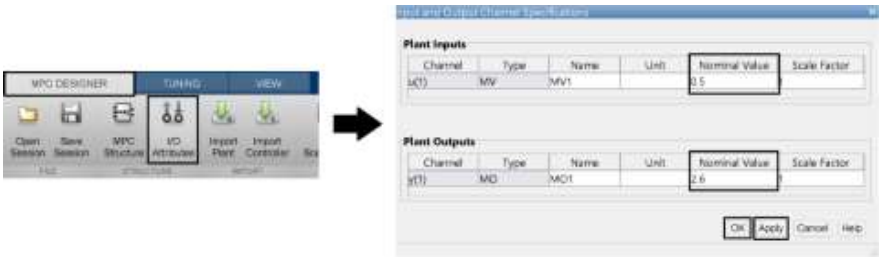


Figura 123. Condiciones iniciales del controlador

En la pestaña "Tuning" de la herramienta se puede modificar varios parametros del controlador para obtener la respuesta deseada del sistema. La Figura 124 muestra los valores de los parametros de sintonía del controlador: horizonte de prediccion y horizonte de control). Se colocan la resticciones de entrada de 0 a 10, puesto que estos valores son permitidos para el prototipo de ensayos de temperatura (véase Figura 125). Se puede mejorar la respuesta del sistema modificando los parámetros de "PERFORMANCE TUNING" (véase Figura 126). La Figura 127 muestra la mejor respuesta que se ha obtenido del controlador.

Figura 124. Parámetros para la estabilidad del sistema

Figura 125. Restricciones para el diseño del controlador

Figura 126. Performance Tuning

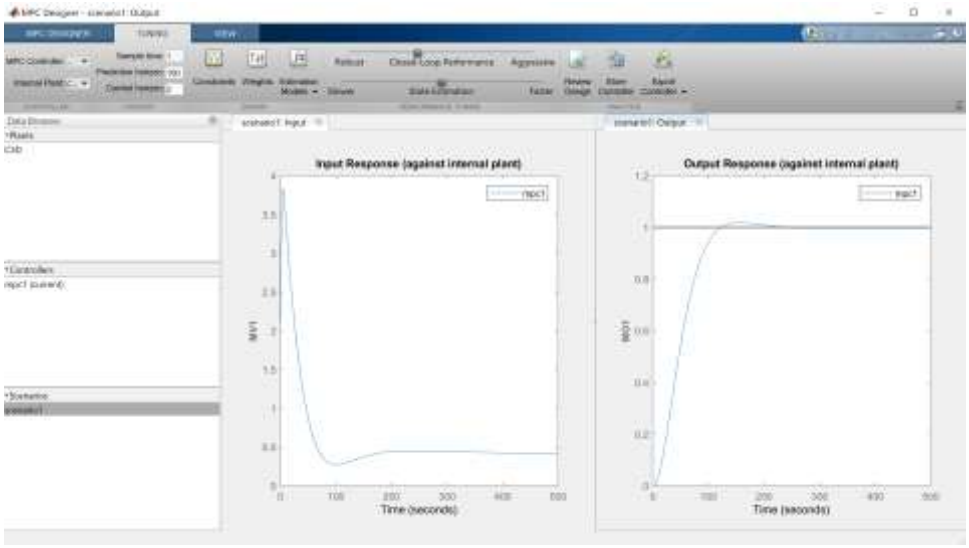


Figura 127. Diseño del controlador MPC

La herramienta MPC Designer permite generar un modelo de Simulink con los datos del controlador (véase Figura 128 y Figura 129). Con el modelo generado se configuró la conexión de los bloques DAQ y del controlador. Para la entrada analógica o variable controlada se ha escogido el puerto ai1 y para la salida analógica o señal de control el puerto ao1. El tiempo de muestreo para estos bloques es de 1 segundo. Esta configuración de bloques es la interfaz para monitoreo y adquisición

de datos en la implementación del controlador (véase Figura 130). Cabe recalcar que los datos se visualizan en escala de grados Celsius.

Figura 128. Generando Modelo de Simulink

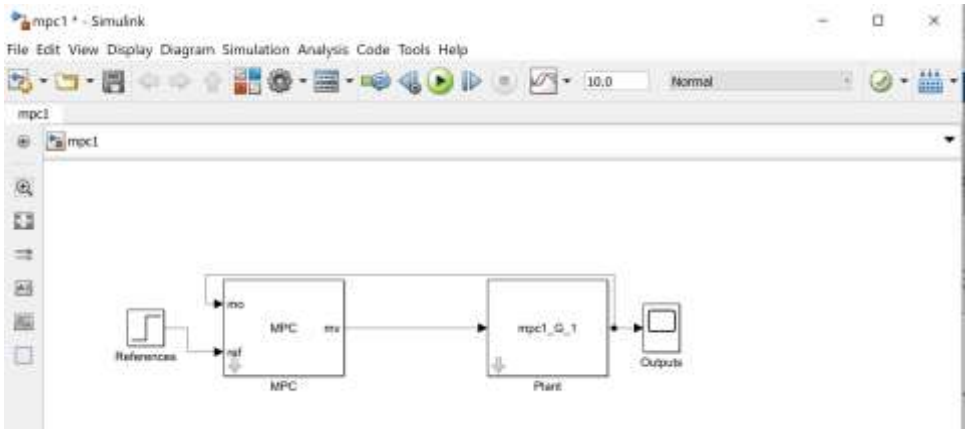


Figura 129. Modelo Generado con la herramienta MPC Designer

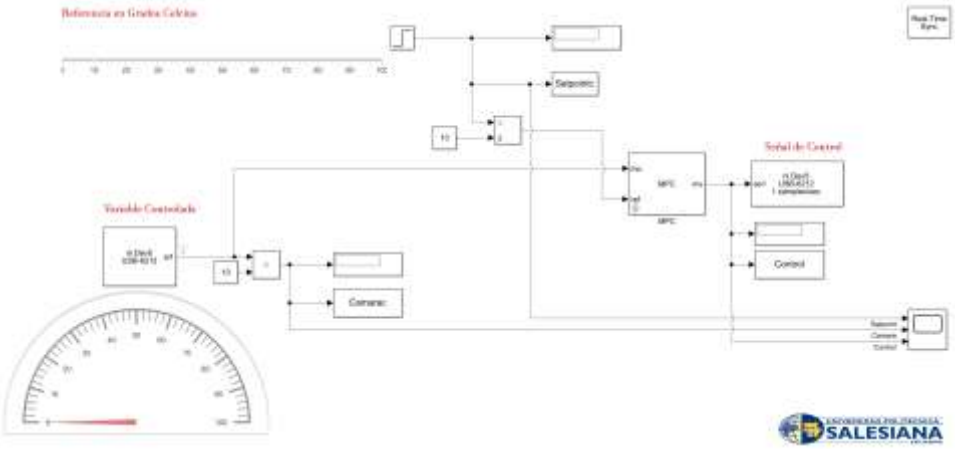


Figura 130. Programa generado para la implantación del controlador MPC

La Figura 131 muestra la respuesta del sistema ante la implementación del controlador MPC mediante un dispositivo NI myDAQ. Para este ensayo se dio al sistema dos escalones de diferente magnitud. Se puede apreciar que el segundo escalón con mayor amplitud el sobrepaso y el tiempo de establecimiento son mayores que con el primer escalón de menor amplitud.

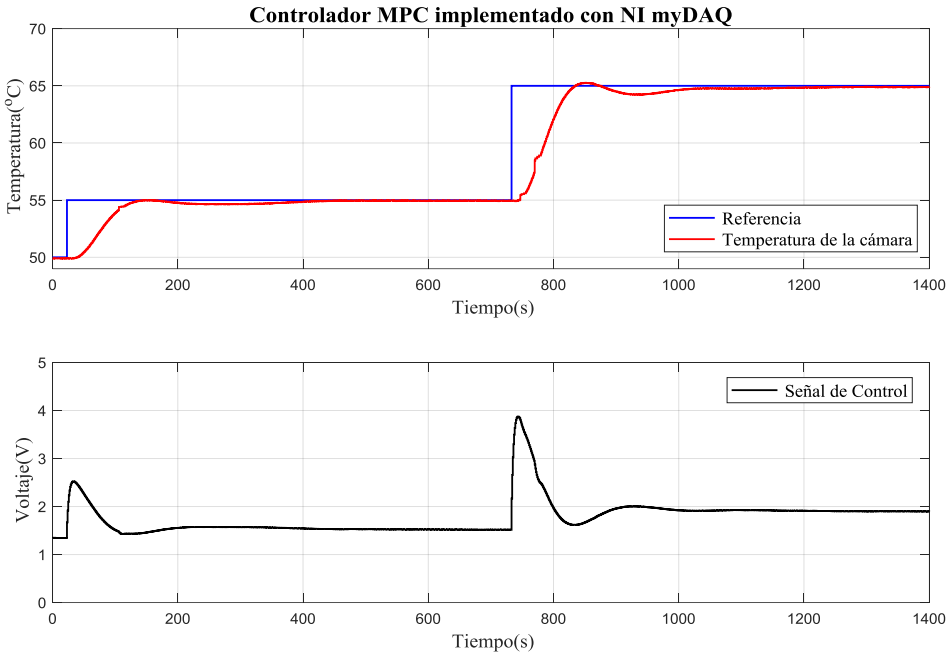


Figura 131. Implementación del controlador MPC mediante NI-DAQ

3.1.2 Controlador Predictivo basado en el Modelo MPC implementado en Simulink mediante un PLC S7-1200

Para la implementación de este controlador se utilizó la herramienta Simulink de Matlab, el servidor OPC del software KEPServerEX, el software TIA Portal v13, un PLC S7-1200 y el prototipo de ensayos de control de temperatura. La guía paso a paso del programa se encuentra en el Apéndice A, guía de prácticas 8.

- Programación en TIA Portal

Se ha creado un programa en TIA Portal en el que se estableció la entrada y salida analógica correspondiente comandar el prototipo de ensayos (véase Figura 132). Se coloca la entrada %IW66 (ETCamara) para medir la temperatura de la cámara y la

salida analógica %QW80 (SControl) para comandar la resistencia calefactora. Las variables virtuales %MD1 (VTemperatura_Camara), la cual será la variable controlada, y %MD100 (SControlador), la cual será la acción de control, se utilizan para la comunicación OPC entre Matlab y el PLC. Estas se configuran en el software KepServerEX (véase Apéndice A, guía de prácticas 8).

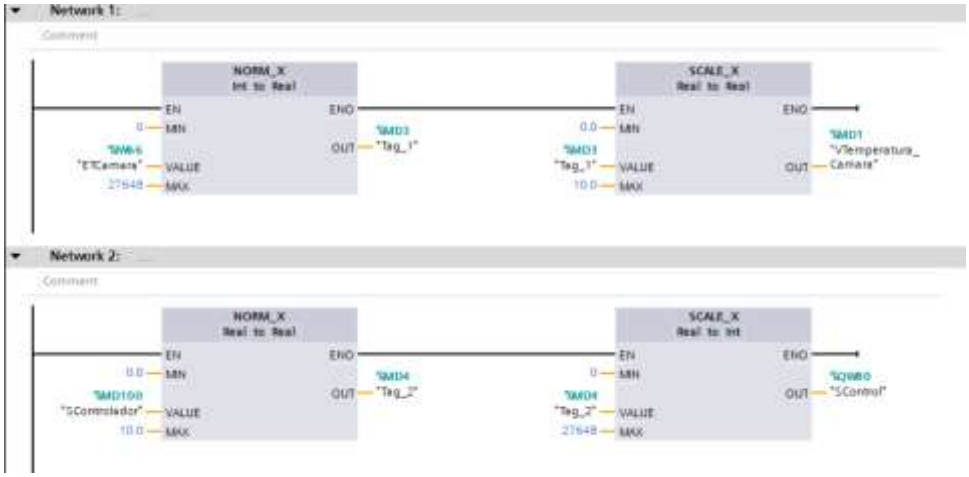


Figura 132. Configuración de bloques

- **Diseño del controlador MPC y programación en la herramienta Simulink**

Para el controlador se utilizó el mismo diseño del controlador MPC implementado en Simulink con un dispositivo NI myDAQ visto anteriormente en la sección 3.3.1 (véase Figura 129).

Se agrega al modelo de Simulink y los bloques OPC para leer (OPC Read) y escribir (OPC Write) los datos obtenidos mediante el PLC. El bloque "OPC Read" se sincroniza con la variable controlada (TemperaturaCamara) (véase Figura 133) y el bloque "OPCWrite" con la acción de control (Controlador) (véase Figura 134). La configuración de cada bloque se realiza al hacer doble clic en los mismos y se coloca el tiempo de muestreo de 1 segundo. La Figura 135 muestra el programa para monitoreo y adquisición de datos del ensayo.

En la Figura 136 se puede observar el comportamiento del controlador MPC implementado en el prototipo al darle un escalón. Se puede un pequeño sobrepaso en la respuesta, a pesar del descenso de temperatura el sistema se estabiliza.

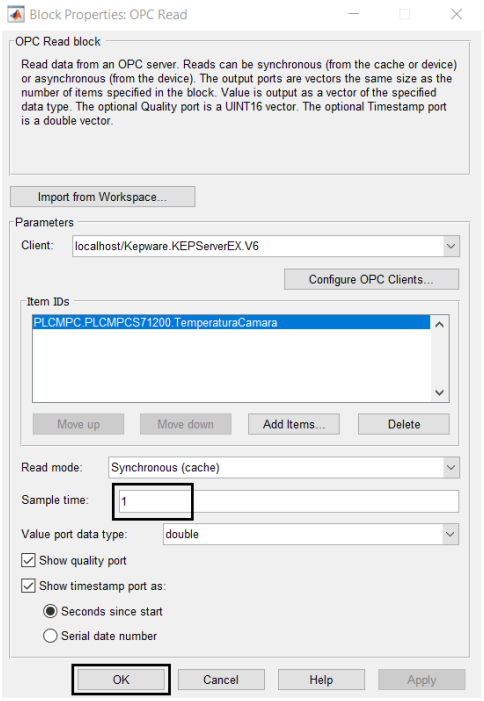


Figura 133. Configuración del bloque "OPC Read"

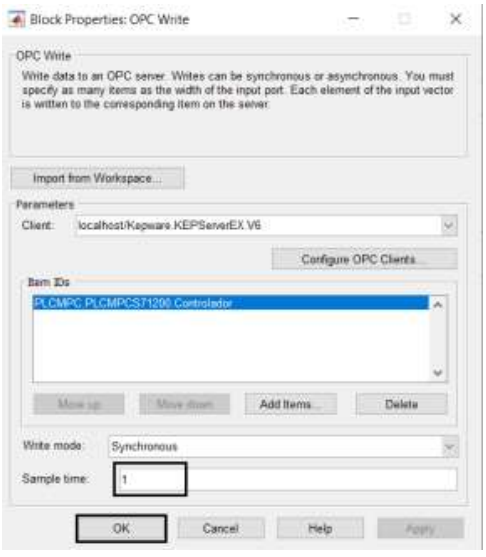


Figura 134. Configuración del bloque "OPC Write"

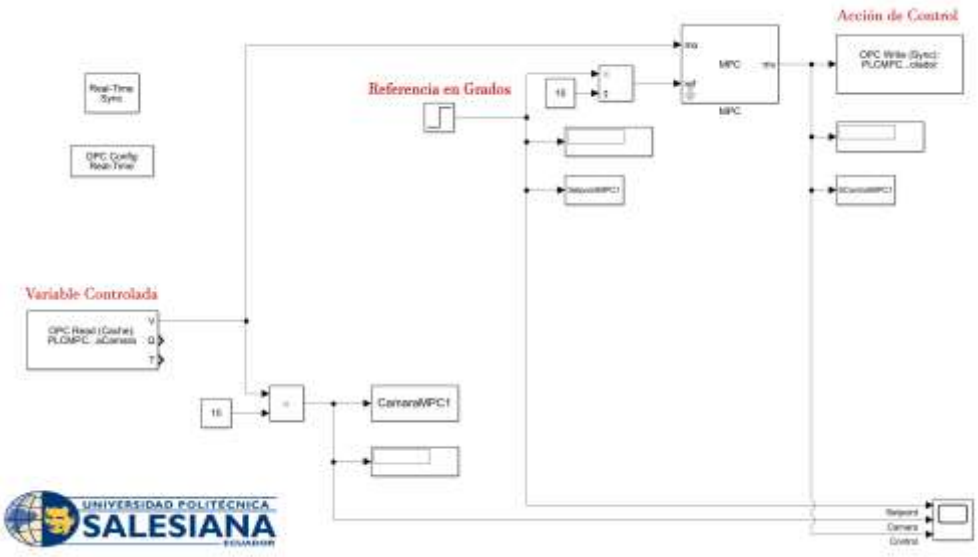


Figura 135. Modelo de Simulink del controlador MPC implementado con PLC S7-1200

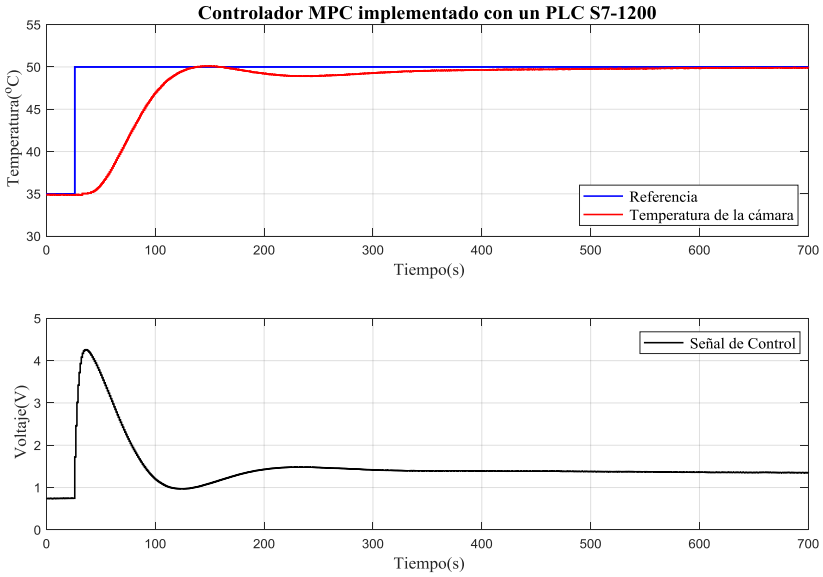


Figura 136. Respuesta del prototipo al implementar un controlador MPC con PLC S7-1200

CAPÍTULO 6: CONCLUSIONES Y RECOMENDACIONES

La construcción del prototipo de proceso térmico se realizó de acuerdo con las especificaciones establecidas, haciendo que este sea compatible con cualquier dispositivo de control donde el actuador tenga una señal de 0 a 10 V o de 4 a 20 mA y para el sensor una señal de 0 a 10 V. Además, se logró un diseño práctico y fácil de transportar.

El controlador PID con LabVIEW fue implementado utilizando el equipo NI myRIO como interfaz de 0-10 voltios. Es importante destacar que las unidades de las señales que se manejan en el software deben ser interpretadas de la misma manera que en el modelo matemático, de otra forma el sistema no funcionara correctamente. Hay que considerar que el control de la temperatura se realiza únicamente con un elemento calefactor, debido a esto la señal para el actuador no deberá ser negativa en ningún momento. Con estas consideraciones se obtiene un comportamiento muy cercano a la simulación.

El módulo TK4S-T4RC requiere de una configuración inicial la cual se realiza fácilmente siguiendo las indicaciones del manual, es importante destacar que la señal de control estará expresada como una intensidad que variará de 0-20 mA. El algoritmo PID incorporado puede ser configurado para que su comportamiento presente o no un sobresalto en función de los requerimientos y preferencias del diseñador. El sistema se estabiliza rápidamente y no presenta oscilaciones.

En la implementación del controlador PID auto sintonizado utilizando un PLC S7-1200 se evidencia el funcionamiento del bloque PID_Compact del software TIA Portal. Este controlador presenta un tiempo de establecimiento corto por lo que proporciona rapidez al sistema.

El controlador con asignación de polos con matriz de realimentación de estados fue implementado utilizando Simulink de Matlab empleando la librería de procesamiento en tiempo real que nos permite realizar operaciones matriciales en función del estado actual del sistema. Se requiere de un computador con una potencia de procesamiento alta para evitar el pantallazo azul o problemas similares. Debido a que no es factible medir todas las variables de estado es imprescindible el uso de un observador de estados. Para corregir el error en estado estacionario se utiliza un integrador embebido sin embargo debido a la naturaleza lenta del sistema obliga a utilizar un modelo con retardo en lugar de ceros. Con estas consideraciones el sistema tendrá un comportamiento cercano a la simulación.

CAPÍTULO 4. CONCLUSIONES Y RECOMENDACIONES

El Controlador Predictivo basado en Modelo se implementó mediante un dispositivo NI myDAQ y un PLC S7-1200, estos no presentan una mayor diferencia en la respuesta del sistema. Sin embargo, en la implementación con el dispositivo NI myDAQ es necesario verificar el tiempo de muestreo en tiempo real de la herramienta Simulink, puesto que si este no es el adecuado la respuesta del sistema siempre será errónea.

De manera general, se ha demostrado que a través de este prototipo que se ha construido para fortalecer el laboratorio de control de la universidad, es posible probar diferentes tipos de controladores y sobre todo utilizar diversas herramientas que pueden ser indispensables para el ejercicio profesional. En este sentido cabe resaltar la importancia de establecer las guías de prácticas para utilizar un PLC y los módulos industriales para implementar controladores PID. Por otra parte y atendiendo a las necesidades del control moderno se ha establecido un conjunto de guías para que los estudiantes puedan experimentar en el manejo de controladores MPC y en espacio de estados.

REFERENCIAS BIBLIOGRÁFICAS

[1] S. R. CEPEDA y F. O. ERAZO, «DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DIDÁCTICO PARA REPRESENTACIÓN DE SEÑALES ANALÓGICAS CON PLC Y SCADA,» Riobamba, 2014.

[2] C. Alvarez, «La relacion teorica-practica en los procesos de enseñanza-aprendizaje».

[3] J. C. H. G. R. L. S. Q. T. R. D. Z. J. P. S. M. Bañuelos S., «CONTROLADOR PID DE TEMPERATURA DE TIPO DIDACTICO,» México, 2016.

[4] I. A. Pozo Espín y P. R. Zurita Baquero, «Diseño y construcción de una presa calefactora para la unión de bandas planas sinteticas mediante resistencias planas,» Quito, 2011.

[5] «Thermolaser,» [En línea]. Available: http://www.resistenciaslaser.com/. [Último acceso: Septiembre 2019].

[6] V. R. Cañazaca Limachi y C. W. Jamachi Espillico, «Módulo didáctico para el análisis de sensores, mediante un PLC, controlada desde un HMI, en el laboratorio de Control y Automatización de la EPIME,» Puno, 2018.

[7] J. S. Marin Torres, «Diseño de una estación didáctica para el control de temperatura mediante un controlador industrial,» Pereira, 2016.

[8] C. A. Vinan Velasco y R. D. Alvarado Bajana, «Diseno e implementa de un sistema de control de temperatura y nivel de agua en la planta de llenado de tanques del laboratorio de automatización industrial utilizando un algoritmo PID,» Guayaquil, 2018.

[9] J. G. &. C. KG, «www.jumo.es,» [En línea]. Available: http://www.jumo.es/productos/temperatura/convertido-de-medici%C3%B3n/707030/jumo-dtrans-t03---two-wire-three-wire-transmitter-707030.html?parentId=2884. [Último acceso: 8 Septiembre 2019].

[10] M. P. Aparicio, Diseño y desarrollo de circuitos impresos con KICAD, RC Libros.

[11] D. A. D.-C. Dr. Jorge Gil Nobajas, Control de sistemas continuos, 2004.

[12] Siemens, «www.siemens.com,» [En línea]. Available: https://cache.industry.siemens.com/dl/files/036/108210036/att_916495/v1/s71500_pid_control_function_manual_esES_es-ES.pdf.

[13] K. Ogata, Ingenieria de control moderna, Madrid: Pearson Education, 2010.

[14] E. Aceros, E. Camargo y O. Parabavire, «Comparacion Cualitatica del Desempeno de la Aplicacion del Control Predictivo Basado en Modelo y PID para el Control de Nivel en Pozos,» Caracas, 2016.

[15] M. Duarte-Mermoud y F. Milla, «Estabilizador de Sistemas de Potencia usando Control Predictivo basado en Modelo,» *Revista Iberoamericana de Automática e Informática Industrial*, vol. Vol. 15, nº Núm. 3, pp. 286-296, 2018.

[16] J. C. Zambrano y A. I. González, «Implementación de un algoritmo de control predictivo en espacio de estados sobre una plataforma de simulación desarrollada en Matlab,» *Ingenius*, nº 9, pp. 5-14, 2013.

[17] S. Vazquez, J. Rodriguez, M. Rivera, L. Franquelo y M. Norambuena, «Model Predictive Control for Power Converters and Drives: Advances and Trends,» *IEEE Transactions on Industrial Electronics*, vol. Vol. 64, nº 2, pp. 935-947, 2017.

[18] National Instruments , «National Instruments,» 2019. [En línea]. Available: https://www.ni.com/es-cr/shop/select/myrio-student-embedded-device. [Último acceso: 11 Noviembre 2019].

[19] «National Instruments,» Junio 2018. [En línea]. Available: https://www.ni.com/pdf/manuals/376047d.pdf. [Último acceso: 08 Septiembre 2019].

[20] «Siemens,» [En línea]. Available: https://new.siemens.com/global/en/products/automation/systems/industrial/plc/s7-1200.html. [Último acceso: 05 Enero 2020].

[21] Autonics , «Modulo de control de temperatura TK4S-T4RC,» [En línea]. Available: https://www.autonicsonline.com/. [Último acceso: 10 01 2020].

[2 Wikipedia, «es.wikipedia.org,» 10 02 2011. [En línea]. Available:
2] http://es.wikipedia.org/wiki/Karel_%C4%8Capek. [Último acceso: 03 03 2011].

[2 R. S. J. S. O. R. Rafael Aracil, «arvc.umh.es,» 01 2006. [En línea]. Available:
3] http://arvc.umh.es/documentos/articulos/RIAI%202006.pdf. [Último acceso: 03 03 2011].

[2 J. A. B. León, *Diseño, análisis y construcción de un robot paralelo traslacional*,
4] Querétaro, 2009.

[2 J. R. Isidro Zabalza, «www.imem.unavarra.es,» 23 10 2007. [En línea].
5] Available: http://www.imem.unavarra.es/isidro/articles/Zabalza-Cuzco.pdf. [Último acceso: 28 02 2011].

[2 P. Palitroquez, «www.picmania.garcia-cuervo.net,» 27 12 2008. [En línea].
6] Available: http://picmania.garcia-cuervo.net/invitados_primer18f4550.php. [Último acceso: 12 03 2011].

[2 Amazon, «www.amazon.com,» 2011. [En línea]. Available:
7] http://www.amazon.com/gp/aw/d.html?is=200&a=B000VTQ3LU&in=3. [Último acceso: 12 03 2011].

[2 L. Reznik, Fuzzy Controllers, Melbourne, Australia: NEWNES, 1997, p. 94.
8]

[2 K. Ogata, Ingeniería de Control Moderno, Minnesota: Prentice-Hall, 1997.
9]

[3 IFR, «www.cfievalladolid2.net,» 2011. [En línea]. Available:
0] http://cfievalladolid2.net/tecno/cyr_01/robotica/industrial.htm. [Último acceso: 01 Marzo 2011].

[3 R. Varbuchta, «www.capek.misto.cz,» 2004. [En línea]. Available:
1] http://capek.misto.cz/english/robot.html. [Último acceso: 28 02 2011].

[3 Robotmania, «www.robotionary.com,» 18 11 2010. [En línea]. Available:
2] http://robotionary.com/robotics/world%E2%80%99s-fastest-robot-quattro.php. [Último acceso: 26 02 2011].

[33] M. S. D. Hernán González, «www.catic.unab.edu.co,» 2006. [En línea]. Available: http://catic.unab.edu.co/2congresomecatronica/images/docum/14.pdf. [Último acceso: 28 02 2011].

[34] U. P. d. Madrid, «www.sadesisam.etsii.upm.es,» 2007. [En línea]. Available: http://sadesisam.etsii.upm.es/trepa/es/project.asp. [Último acceso: 26 02 2011].

[35] L. A. Silva, «www.oa.upm.es,» 2005. [En línea]. Available: http://oa.upm.es/378/1/luis_angel_silva.pdf. [Último acceso: 29 02 2011].

[36] U. P. d. Cataluña, «www.tdr.cesca.es,» 2008. [En línea]. Available: http://www.tdr.cesca.es/TESIS_UPC/AVAILABLE/TDX-0207105-105056//04Rpp04de11.pdf. [Último acceso: 04 03 2011].

[37] S. K. Alonso, «eMathTeacher,» 2009. [En línea]. Available: http://www.dma.fi.upm.es/research/FundMatSoftComputing/fuzzyinf/mamdani1.htm. [Último acceso: 21 02 2011].

[38] I. H. F. Velasco, «www.docentes.unal.edu.co,» 2006. [En línea]. Available: http://www.docentes.unal.edu.co/hfvelascop/docs/OTROS/LOGICADIFUSA/5%20An%E1lisis%20de%20Controladores%20Difusos.pdf. [Último acceso: 21 02 2011].

[39] «DataLights,» [En línea]. Available: https://www.datalights.com.ec/site2/index.php?option=com_content&view=article&id=54:epc&catid=37:epc&Itemid=57.

[40] X. F. Ochoa Parra, «Elaboracion de un modulo guia para el desarrollo de practicas en el laboratorio de control de la Universidad Politecnica Salesiana,» Cuenca, 2016.

[41] M. G. Ortiz Escobar, «“Diseño y construcción de una matriz de embutido y corte para productir un casquillo bipolar de resistencias eléctricas industriales en la empresa Redanse S.A,» Ambato, 2018.

[42] W. W. Narváez Garzón y J. L. Suaréz Terán, «Diseño e implementación de un sistema de control de temperatura para la máquina extrusora de granos de laboratorio de ciencia de alimentos y biotecnología de la Escuela Politécnica Nacional,» Quito, 2014.

[43] «Direct Industry,» [En línea]. Available: https://www.directindustry.es/prod/h-b-sensors/product-71436-1643647.html. [Último acceso: 2019 09 25].

[44] «Hubot,» [En línea]. Available: https://hubot.cl/product/sonda-temperatura-pt100-rtd-sku-518/. [Último acceso: 2019 09 25].

[45] «www.amazon.com,» [En línea]. Available: https://www.amazon.es/WINSINN-Ventilador-escobillas-rodamiento-refrigeraci%C3%B3n/dp/B07DB59XP8. [Último acceso: 08 Septiembre 2019].

[46] «Maxpolheat,» [En línea]. Available: http://www.maxpolheaters.com/productos-es/Resistencias-espiral.html. [Último acceso: 30 Sept 2019].

APÉNDICES

APÉNDICE A: GUÍAS DE PRÁCTICAS DE LABORATORIO PARA EL ANÁLISIS Y CONTROL DEL PROCESO TÉRMICO

- **GUÍA DE PRÁCTICA 1: ADQUISICIÓN DE DATOS Y ANÁLISIS DE HISTÉRESIS DE LA PLANTA**

<table>
<tr><td colspan="3">FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES</td></tr>
<tr><td colspan="2">CARRERA: INGENIERÍA ELECTRÓNICA</td><td>ASIGNATURA: TEORÍA DE CONTROL</td></tr>
<tr><td>NRO. PRÁCTICA:</td><td>1</td><td>TÍTULO PRÁCTICA:

ADQUISICIÓN DE DATOS Y ANÁLISIS DE HISTÉRESIS DE LA PLANTA</td></tr>
<tr><td colspan="3">OBJETIVO

1. Proporcionar las herramientas necesarias para que el estudiante pueda crear un proyecto en el software LabVIEW 2019 que permita el manejo del Equipo de ensayo de control de temperatura mediante el equipo NI myRIO.
2. Conocer las características generales del equipo de ensayo de control de temperatura, su rango de trabajo y el equipo NI myRIO.
3. Conocer con las herramientas del software LabVIEW y Matlab.</td></tr>
<tr><td colspan="2">INSTRUCCIONES
(Detallar las instrucciones que se dará al estudiante):</td><td>1. Requisitos y conocimientos previos

a) Creación de un proyecto con el software LabVIEW 2019 y el dispositivo NI-MyRIO.</td></tr>
</table>

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Computadora S.O. Windows 10, 64 bits	1	-	-
Software LabVIEW 2019.	1	-	-
Software Matlab 2017	1	-	-
NI myRIO	1	Nacional Instrument	-
Equipo de ensayos de control de temperatura	1	CTILab	-
Cables	Varios	-	-

Tabla 1. Equipos, instrumentos y software

3. Exposición

Kit NI myRIO: proporciona entradas y salidas configurables que permite ensenar e implementar múltiples conceptos de diseño con un dispositivo.

Incluye entradas analógicas, salidas analógicas, líneas E/S digitales, LEDs, un push-Button, un acelerómetro interno, un FPGA Xilinx y un procesador dual-core ARM Cortex-A9. [18]

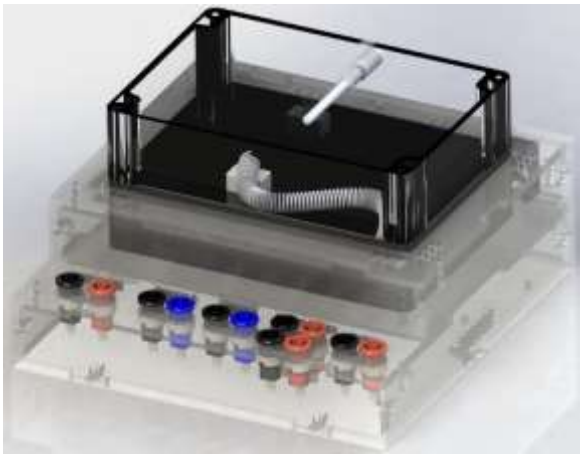
Figura 1. NI myRIO [19]

Datos técnicos:

NI myRIO	
Memoria	No volátil (512 MB) DDR3 (256 MB)
FPGA	Xilinx Z-7010
Wireless Characteristics	Radio (IEEE 802.11 b,g,n). Frecuencia de banda (ISM 2.4 GHz). Ancho de banda (20MHz). Canales (USA 1 A 11, Internacional 1 a 13). TX (+10dBm max). Outdoor range (Hasta 150m). Directividad de la antena (Omnidireccional). Seguridad (WPA, WPA2, WPA2-Enterprise).
USB Ports	USB host-device port (2.0 Hi-Speed)
Acelerómetro	Número de ejes (3). Rango (± 8g). Resolución (12bits). Taza de muestreo (800 S/s). Ruido (3.9 mg(rms) a 25ºC)
Requerimientos de potencia	Rango de voltaje de la fuente de alimentación (6 a 16 VCC). Consumo máximo de energía (14 W). Consumo de energía inactivo típico (2.6 W).

Tabla 1. Características del dispositivo NI-myRIO

***Equipo de ensayos de temperatura**:* este dispositivo permite realizar ensayos de control de temperatura con picos de hasta 80 °C, equipado con una niquelina fabricada con alambre de nicromo, dos ventiladores 3010 para perturbaciones, sensores de temperatura tipo

	RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C. El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.  *Figura 2. Equipo de ensayo de control de temperatura*
ACTIVIDADES POR DESARROLLAR (Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)	
1. Programación en el software LabVIEW • Para abrir LabVIEW 2019 myRIO Toolkit dirigirse al siguiente link http://www.ni.com/product-documentation/14603/es/ y descargar e instalar "Módulo LabVIEW Real-Time" y "LabVIEW myRIO Toolkit". • Abrir el acceso directo del software LabVIEW 2019. Conectar la NI myRIO al computador. • Una vez abierto el software, dentro de la pantalla de inicio dar clic en "Launch the Getting Started Wizard", aquí configurará el equipo NI myRIO (véase Figura 3).	

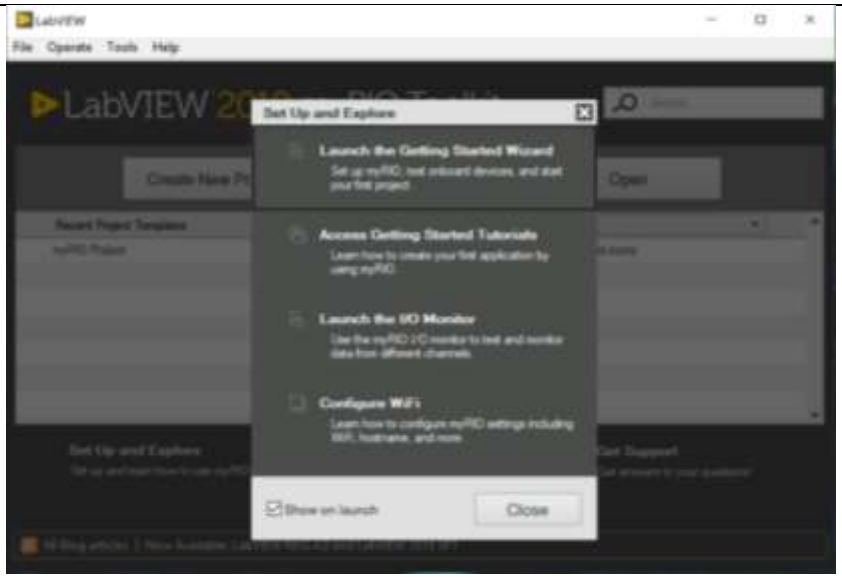


Figura 3. Pantalla de inicio.

- Se abre la ventana "Get Started with NI myRIO" (Iniciando con NI myRIO) donde aparecerá el nombre del dispositivo "NI-MyRIO-1900-03058D001" o similar (véase Figura 4).
- Clic en "Next" (Siguiente).

Figura 4. Pantalla de configuración del NI MyRIO.

- La siguiente configuración permite cambiar el nombre del dispositivo. Se deja el nombre por defecto.
- Clic en "Next".

Figura 5. Pantalla de configuración del NI MyRIO.

- Se puede comprobar el funcionamiento del dispositivo con el panel visto en pantalla (véase Figura 6). Al dar clic en los botones del panel ubicados al lado derecho, es posible observar en el dispositivo el cambio de estado de los LEDs. Al presionar el botón físico del dispositivo ubicado al costado inferior del dispositivo, se puede observar en pantalla el cambio de estado del LED ubicado en la parte inferior. Se puede comprobar el funcionamiento del acelerómetro del dispositivo al moverlo, esto se observa en la parte izquierda de la ventana.

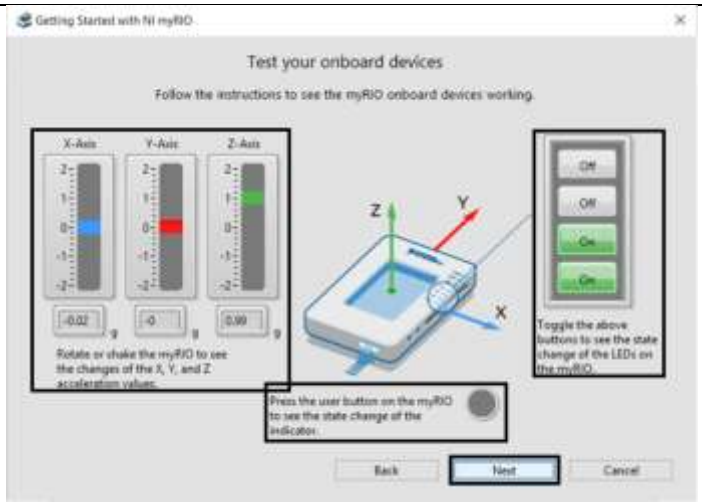


Figura 6. Pantalla de configuración del NI MyRIO.

- Una vez comprobada la conexión del dispositivo dar clic en "Next".
- Se inicia con un nuevo Proyecto dando clic en "Start your first Project now [Recommended]" (Comienza tu primer Proyecto ahora) (véase Figura 7).

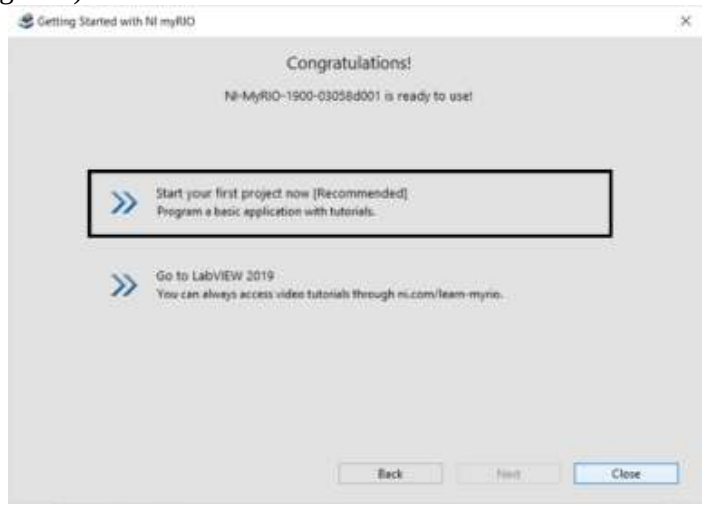


Figura 7. Pantalla de configuración del NI MyRIO.

- Dar clic en el botón "Create New Project" (Crear Nuevo Proyecto) (véase Figura 8).

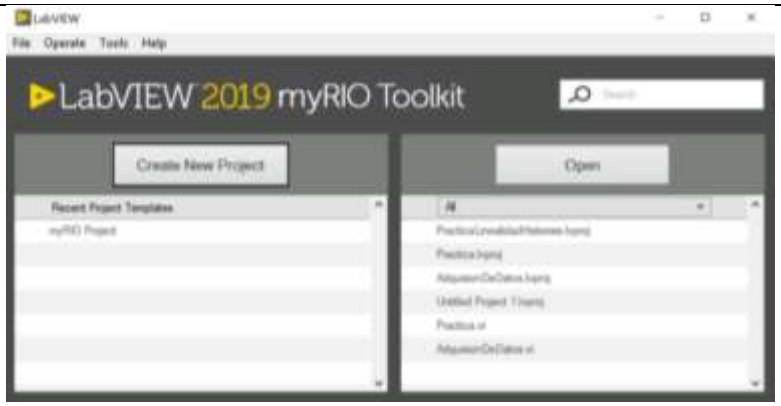


Figura 8. Pantalla de inicio.

- En la ventana "Create Project" (Crear Proyecto) seleccionar la opción "myRIO"
- Elegir la opción "myRIO Project" (myRIO Proyecto)
- Clic en "Next" (véase Figura 9)

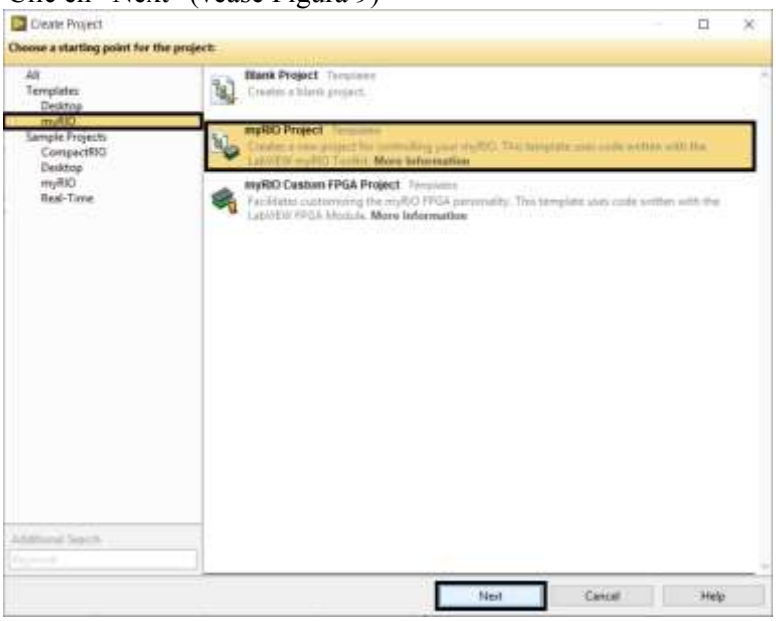


Figura 9. Pantalla de Creación de Proyecto.

- A continuación, se configura el proyecto colocando el nombre y la ubicación del archivo (Véase Figura 10).
- Seleccionar la opción "Plugged into USB" (Conectado a USB), se muestra el nombre y la dirección IP del dispositivo.

- Clic en “Finish” (Finalizar).

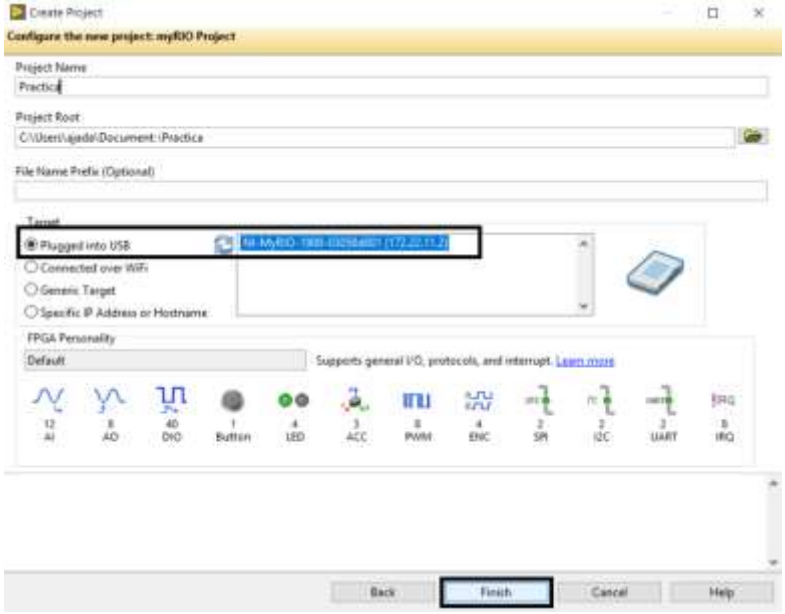


Figura 10. Pantalla para agregar un nuevo dispositivo.

La ventana con el árbol del proyecto contiene todos los archivos agregados en el mismo (véase Figura 11).

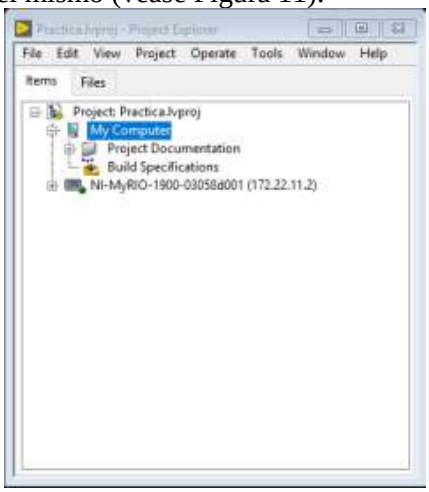


Figura 11. Ventana con el proyecto creado.

- Dar clic derecho sobre “NI-myRIO-1900-03058d001 (172.22.11.2)”, seleccionar “New” (Nuevo) y “VI”.

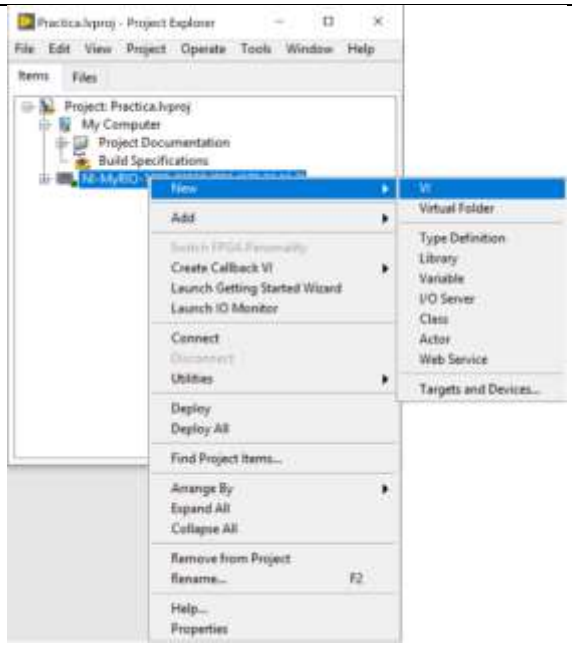


Figura 12. Agregar VI.

Al escoger "VI" (Interfaz Gráfica) se abre las ventanas "Block Diagram" (Diagrama de Bloques) y "Front Panel" (Panel Frontal) (véase Figura 13).

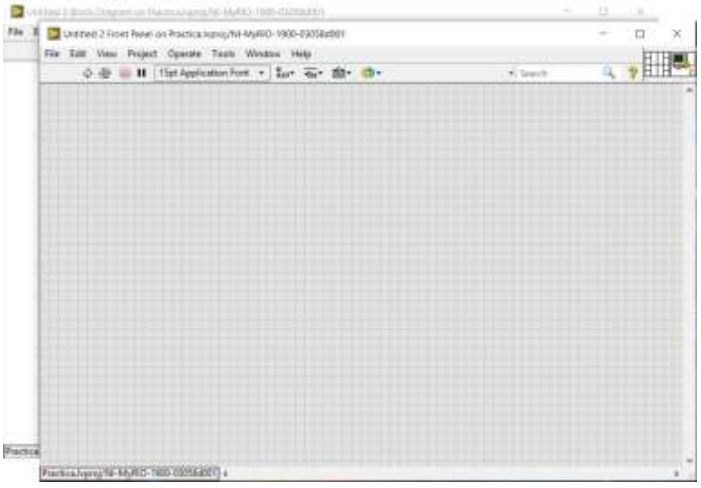


Figura 13. Ventanas del VI.

- Clic derecho en la ventana "Block Diagram" (Diagrama de Bloques).
- Seleccionar "Structures" (Estructuras).

- Arrastrar el bloque “While Loop” (Bucle While) a la ventana. (Véase Figura 14 y Figura 15)

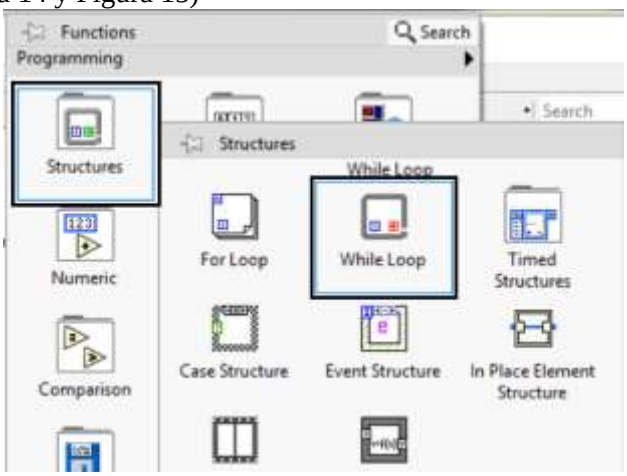


Figura 14. While Loop.

Figura 15. Bloque Bucle While

A continuación, se coloca los bloques de la MyRIO para leer las entradas analógicas del bloque C.

- Se hace clic derecho en la pantalla, seleccionar “MyRIO” (Véase Figura 16) y arrastrar el bloque “Analog In” (Entradas Analógicas) dentro del “While Loop”, donde se abrirá la ventana “Configure Analog Input” (Configuración de Entradas Analógicas) (Véase Figura 17).

Figure 16. Configuración de las entradas analógicas.

- En el cuadro “Channel” (Canal) seleccionar “C/AI0 (AI0)” (Véase Figura 17). Esta entrada se destinará al sensor de Temperatura de la Cámara del Equipo de ensayo de control de temperatura.

Figura 17. Ventana de Configuración de Entradas Analógicas

- Para el sensor de Temperatura Ambiente, dar clic en "+" (Véase Figura 18).

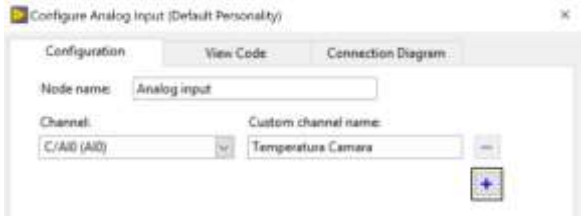


Figura 18. Configuración entrada analógica "C/AI1 (AI1)".

- En "Channel" (Canal) la opción "C/AI1 (AI1)" (Véase Figura 19).
- Clic en "OK"

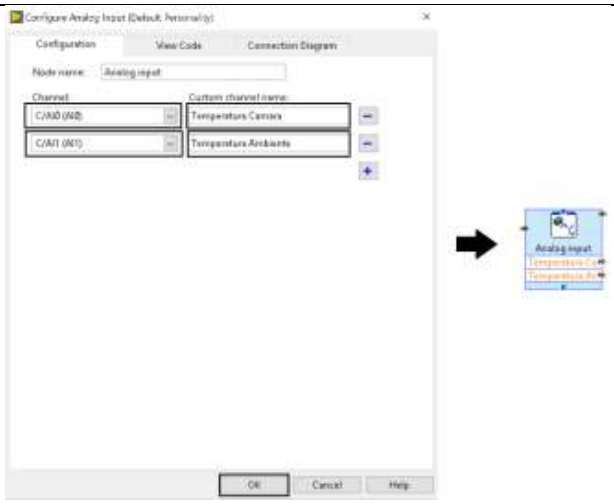


Figura 19. Configuración entrada analógica "C/AI1 (AI1)"

La configuración de salidas analógicas se realiza a continuación:

- Clic derecho en la ventana de "Block Diagram" (Diagrama de Bloques), seleccionar "MyRIO" (véase Figura 20) y arrastrar el bloque "Analog Output" (Salidas Analógicas) dentro del "While Loop" (Bucle While).

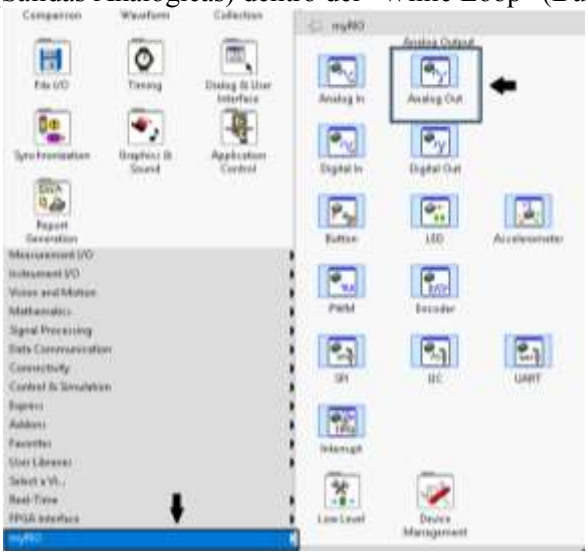


Figura 20. Bloque Analog Output

- En la ventana "Configure Analog Output", desplegar el menú de "Channel" (Canal) (Véase Figura 21).

- Seleccionar la opción “C/AO0 (AO0)”. Esta se usará para comandar la Resistencia Calefactora del Equipo de ensayo de control de temperatura.

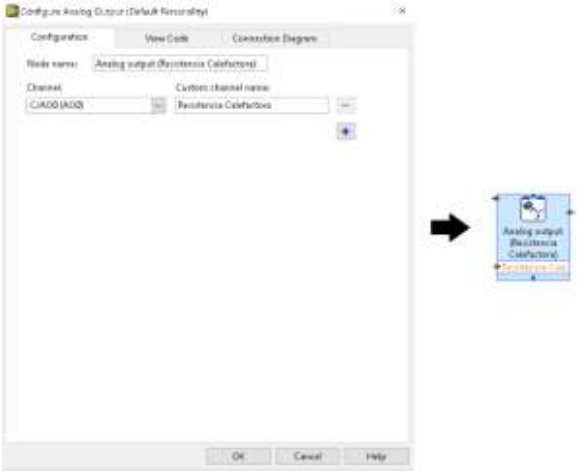


Figura 21. Configuración de salida analógica

Se coloca los indicadores y las multiplicaciones respectivas para convertir de voltajes a grados Celsius.

- Clic derecho en la pantalla de “While Loop”, seleccionar “Numeric” (Numérico) (véase Figura 22) y arrastrar bloque “Multiply” (Multiplicar)

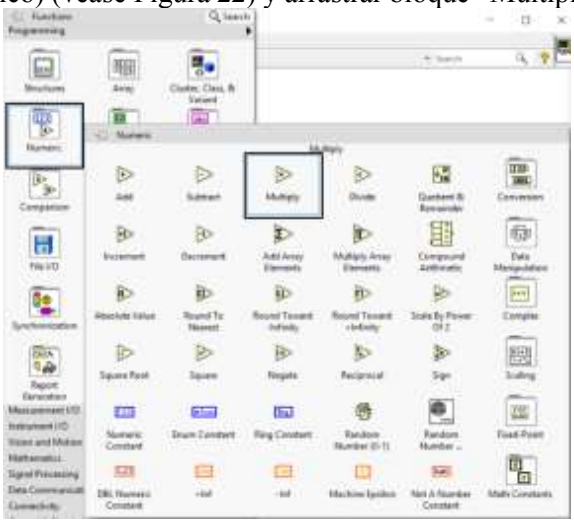


Figura 22. Bloque para multiplicación

Figura 23. Multiplicación para las entradas analógicas

- En el bloque de "Multiply" dar clic derecho en la señal de salida.
- Clic en "Create Indicator" (Véase Figura 24)

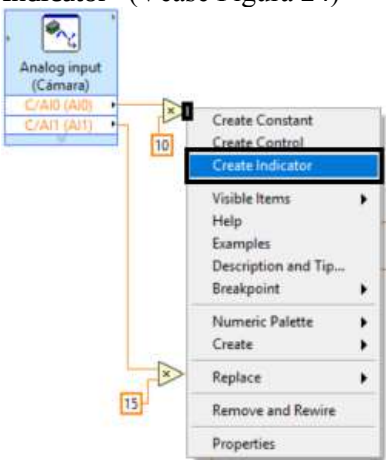


Figura 24. Creando indicadores

Esto se repite para colocar todos los indicadores señalados en la Figura 25.

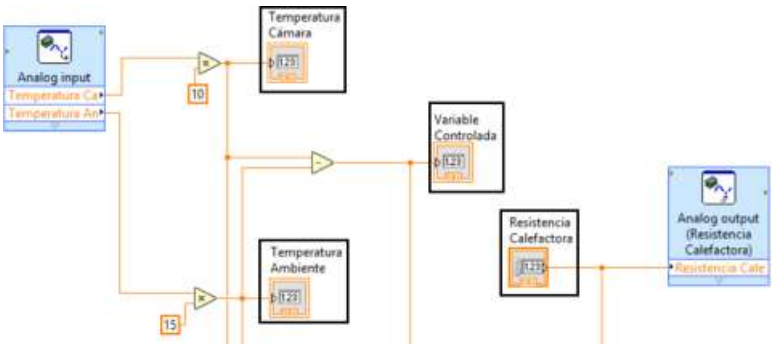


Figura 25. Creando indicadores

Para guardar los datos se realizan los siguientes pasos:

- Clic derecho en “Block Diagram”, seleccionar “File I/O” (Archivo E/S) y arrastrar el bloque “Write To Measurement File” (Escribir a archivo de datos) dentro del “While Loop” (véase Figura 26).

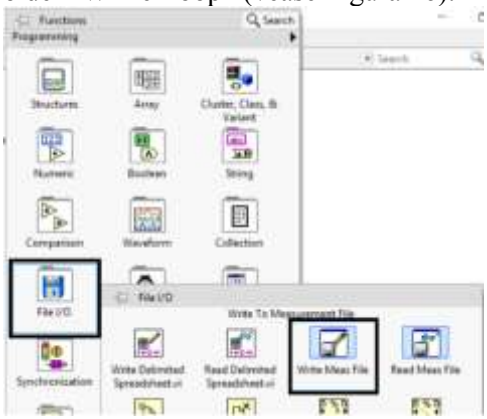


Figura 26. Bloque para guardar datos

- En la ventana “Configure Write To Measurement File” (Configuración del bloque escribir a archivos de datos) se coloca la dirección del archivo, el formato, entre otros.

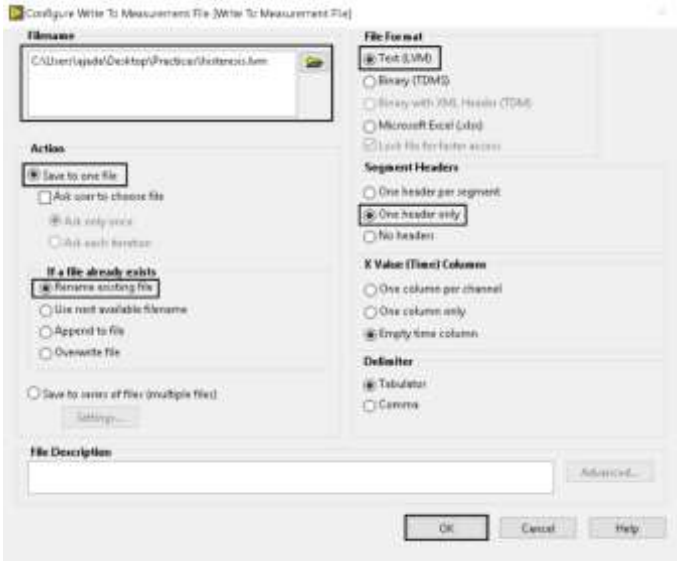


Figura 27. Ventana para Configuración de Archivo para guardar los datos

- Clic en el icono de la carpeta (Véase figura 28)

Para guardar los datos se debe crear una carpeta en el Escritorio, una vez creada la carpeta se coloca el nombre del archivo en este caso "histeresis.lvm".

Figura 28. Dirección para guardar los datos

La programación de bloques se muestra en la figura 29.

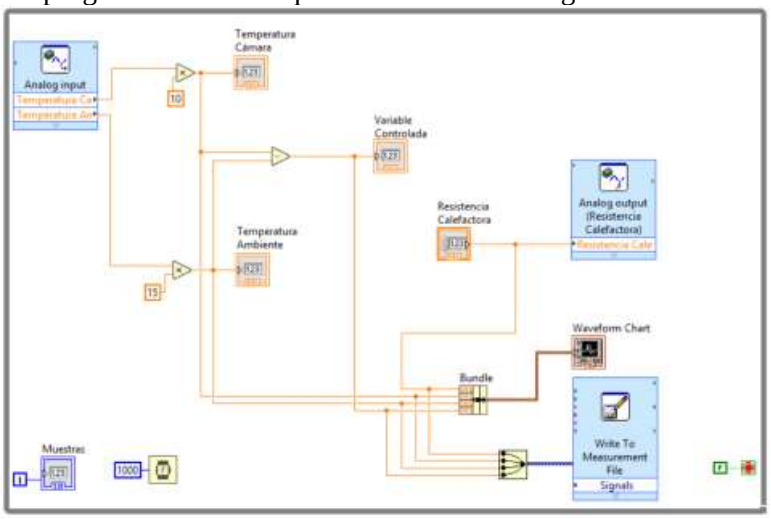


Figura 29. Programación de bloques

Nota:

- La señal de entrada analógica de la Cámara se multiplica por 10 puesto que el transmisor da un voltaje de 0 a 10 V para 0 a 100 ºC. La señal de la Temperatura Ambiente se multiplica por 15 puesto que el transmisor da un voltaje de 0 10V para 0 a 100 ºC.
- La señal de salida para la Resistencia Calefactora tiene un rango de 0 a 10V. Cabe recalcar que las señales de entrada y salida del Equipo de ensayo de control de temperatura son voltajes continuos.
- Para encontrar un bloque rápidamente hacer clic derecho en la ventana y luego clic en "Search". Aquí se escribe lo que se desea encontrar (véase Figura 30 y 31).

APÉNDICES

Figura 30. Search

Figura 31. Search

La ventana "Front Panel" se muestra en la figura 32.

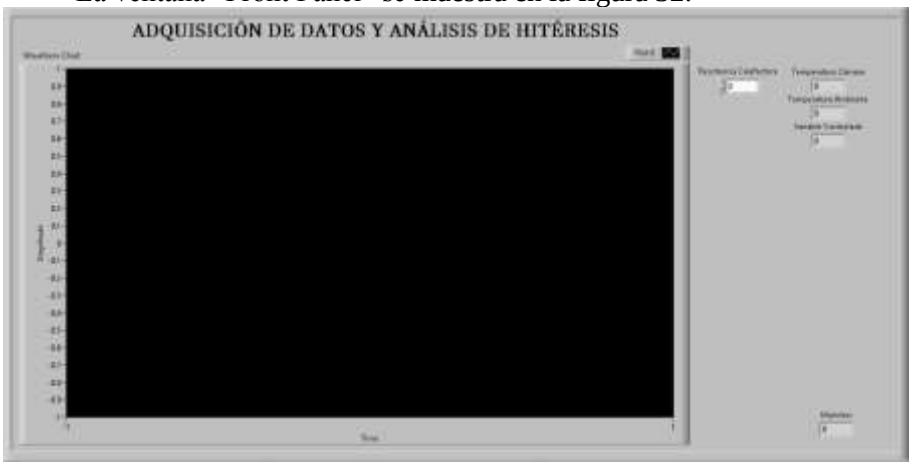


Figura 32. Ventana Front Panel

Adquisición de datos

- Ejecutamos el código cuando se desee capturar las señales (véase Figura 33).

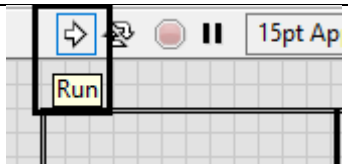


Figura 33. Ejecutar código

- Para el análisis de histéresis del equipo se dará un rango de voltaje de 0 a 2.5V a la Resistencia Calefactora. Cada escalón tendrá una amplitud de 0.5V. Los valores serán 0.5, 1, 1.5, 2 y 2.5.
- En cada escalón dado se debe esperar a que el sistema alance su estabilidad.
- Se hace lo mismo para los escalones de bajada, en este caso los valores serán 2.5, 2, 1.5, 1, 0.5 y 0.

Accediendo a los datos obtenidos

- Para acceder al archivo de datos debemos utilizar un navegador de internet, donde ingresaremos la dirección IP de la NI myRIO seguido de "/files" (véase Figura 34).

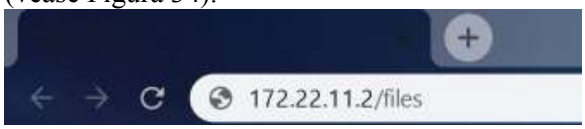


Figura 34. Ingresar dirección IP

- En el campo de usuario ingresamos "admin" y dejamos el campo de contraseña en blanco (véase Figura 35).

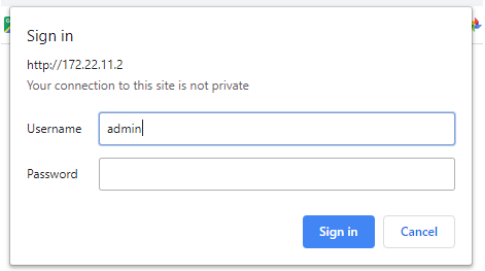


Figura 35. Ingresar credenciales

- Accedemos al directorio seleccionado anteriormente y descargamos el archivo. En caso de que simplemente se muestre en el navegador debemos dar clic derecho sobre los datos y guardarlo como texto. (véase Figura 36)

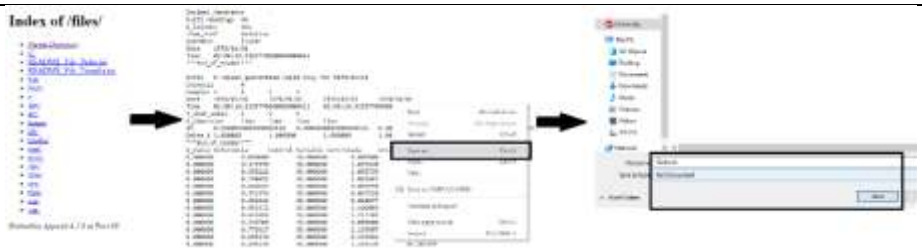


Figura 36. Guardar datos

2. Importar los datos en Matlab

Se abre el software Matlab R2017b.

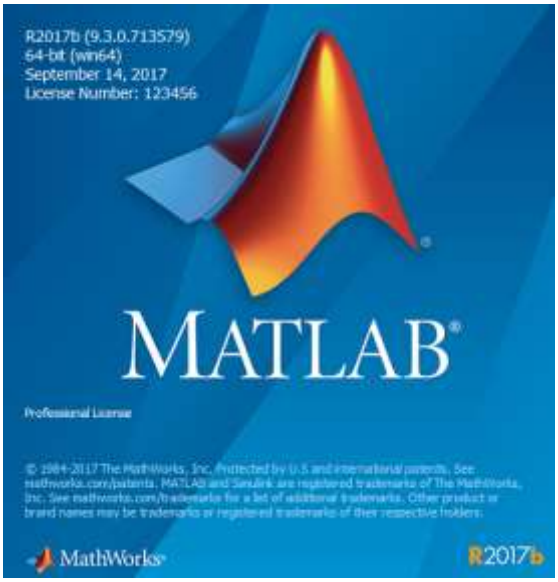


Figura 37. Software MATLAB R2017b.

- Clic en la ventana "HOME" (véase Figura 38)
- En la sección "VARIABLE" (Variable) seleccionar "Import Data" (Importar Datos)

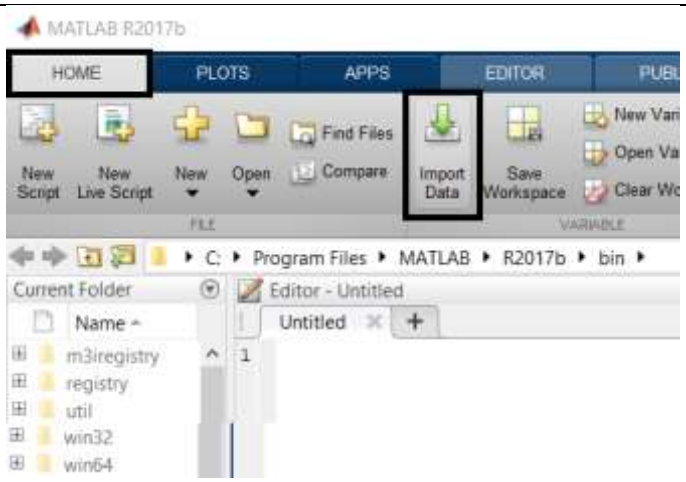
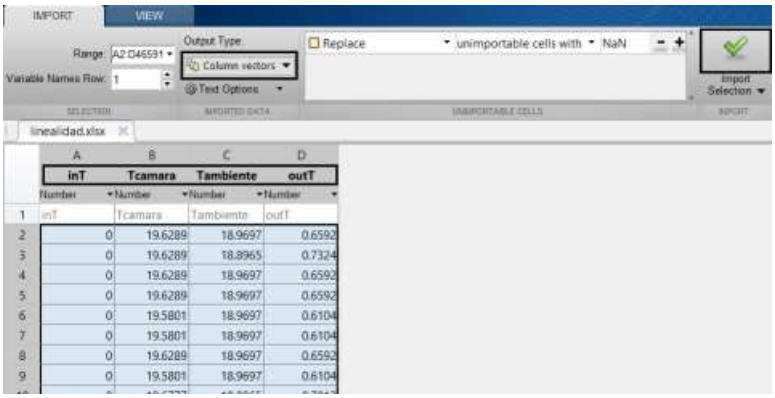


Figura 38. Importar datos adquiridos.

Se abre la ventana "IMPORT"

- En la sección "IMPORTED DATA", en la opción "Output Type" se escoge "Column vectors" (véase Figura 39).
- Cambiar el nombre de las columnas colocándose en la cada columna y escribir el nombre respectivo (véase Figura 39).
- Dar clic en "Import Vector".



	inT	Tcamara	Tambiente	outT
1	inT	Tcamara	Tambiente	outT
2	0	19.6289	18.9697	0.6592
3	0	19.6289	18.8965	0.7324
4	0	19.6289	18.9697	0.6592
5	0	19.6289	18.9697	0.6592
6	0	19.5801	18.9697	0.6104
7	0	19.5801	18.9697	0.6104
8	0	19.6289	18.9697	0.6592
9	0	19.5801	18.9697	0.6104

Figura 39. Importando datos

Guardar los datos ubicados en el Workspace

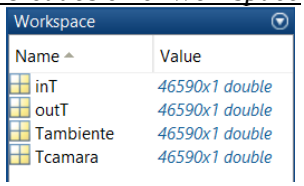


Figura 40. Variables Workspace

- En la pestaña "HOME", en la sección "VARIABLE", seleccionar "Save Workspace" (Guardar Espacio de Trabajo) (véase Figura 41).

Nota: Guardar los datos en la misma carpeta que el "Script", el nombre del archivo en este caso se llamará datos.mat.

Figura 41. Guardando los datos

- Se crea un nuevo proyecto escogiendo "New Script"

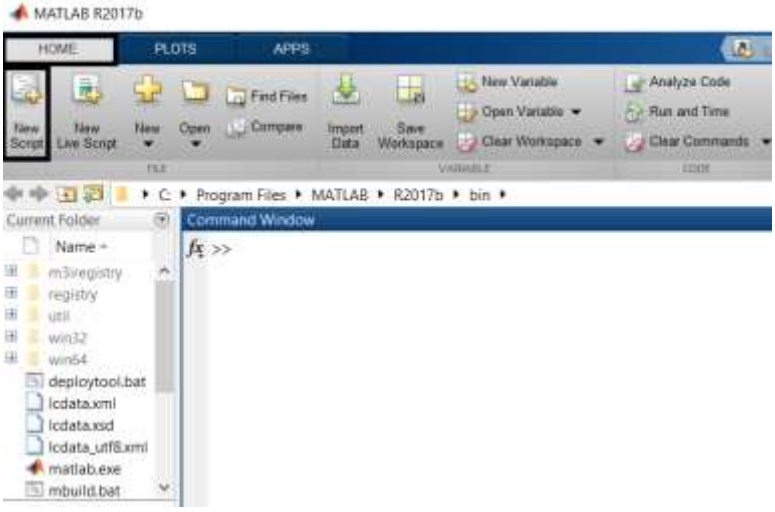


Figura 42. Creando "New Script"

3. Análisis de histéresis de la diferencia entre la temperatura de la cámara y la temperatura ambiente

- Con el siguiente código se procede a cargar y graficar los datos de la diferencia entre la temperatura de la cámara y la ambiente (outT) (véase Figura 43).

```
load datos.mat
ds=1; %Tiempo de muestro
t=0:ds:length(inT)-ds;
figure(1)
subplot(211);
plot(t,inT,'b','LineWidth',2);
xlabel('Time(s)'); ylabel('Voltaje (V)'); legend('Entrada');
title('Temperatura Cámara - Temperatura Ambiente'); grid on;
subplot(212);
plot(t,outT,'r','LineWidth',2);
xlabel('Time(s)');ylabel('Temperatura (^oC)');legend('Salida');
grid on;
```

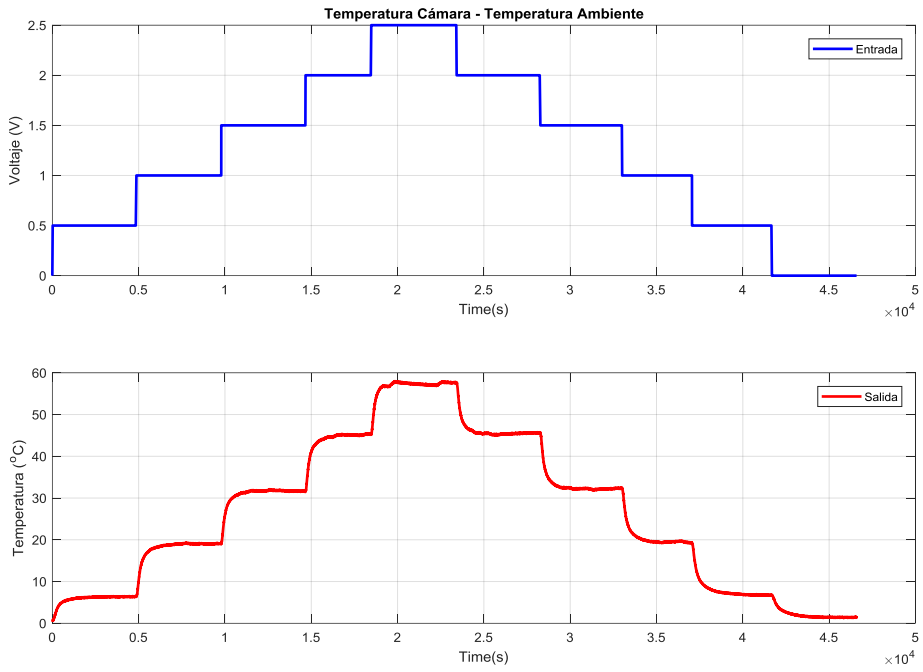


Figura 43. Datos tomados con el LabVIEW de la diferencia entre la temperatura de la cámara y la temperatura ambiente

- Para observar la histéresis se toma un dato del régimen permanente en cada escalón de la diferencia entre la temperatura de la cámara y la ambiente (outT) (véase Figura 43).

APÉNDICES

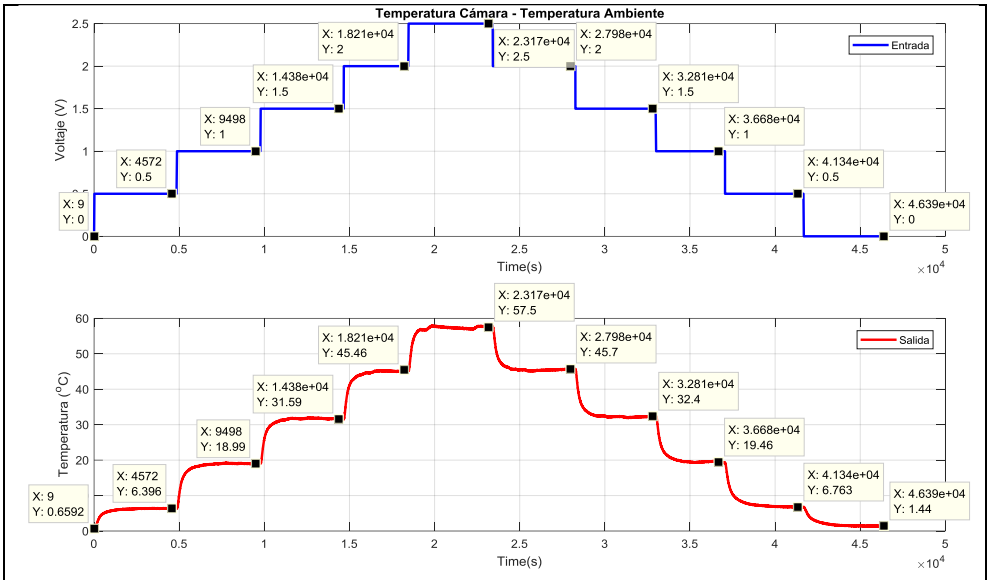


Figura 44. Datos para análisis de histéresis de la diferencia entre la temperatura de la cámara y la temperatura ambiente

Datos para los escalones de subida

Voltaje	**Temperatura**
0	0.6592
0.5	6.396
1	18.99
1.5	31.59
2	45.46
2.5	57.5

Tabla 2. Datos de Temperatura para escalones de subida

Datos para los escalones de bajada

Voltaje	**Temperatura**
2.5	57.5

2	45.7
1.5	32.4
1	19.46
0.5	6.763
0	1.44

Tabla 3. Datos de Temperatura para escalones de bajada

Al graficar los datos de bajada y de subida se observa que el sistema tiene un comportamiento líneal con voltajes mayores 0.5 V (véase Figura 45). El sistema presenta una histéresis mínima.

```
%Datos de temperatura para cada escalon diferencia
v=0:0.5:2.5; %Voltajes de entrada
%Subida
Ts=[0.6592 6.396 18.99 31.59 45.46 57.5];
%Bajada
Tb=[1.44 6.763 19.46 32.4 45.7 57.5];
figure(3)
plot(v,Ts,'b -o','LineWidth',2); hold on; grid on;
plot(v,Tb,'r -*','LineWidth',2);
xlabel('Voltaje (V)');ylabel('Temperatura (^oC)')
title('Histéresis diferencia entre temperatura en la cámara y temperatura ambiente')
legend('Escalones de subida','Escalones de bajada')
```

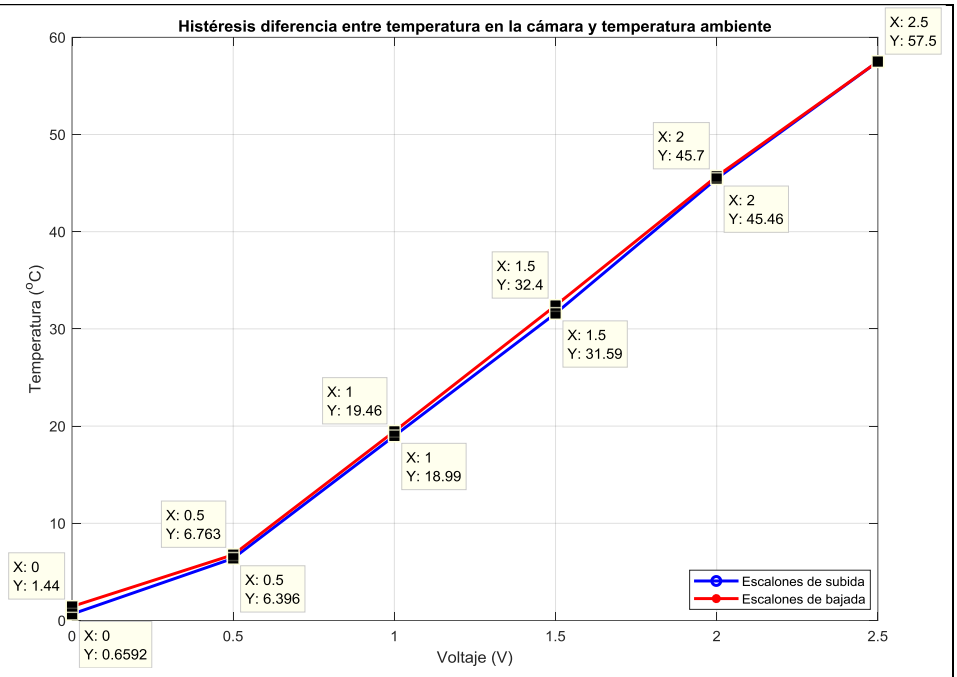


Figura 45. Histéresis de la diferencia entre la temperatura de la cámara y la ambiente (outT)

4. Histéresis de la temperatura de la cámara

- Se grafica los datos obtenidos de la temperatura de la cámara (Tcamara) (véase Figura 45).

```
figure(2)
subplot(211);
plot(t,inT,'b','LineWidth',2);
xlabel('Time(s)'); ylabel('Voltaje (V)'); legend('Entrada');
title('Temperatura Cámara'); grid on;
subplot(212);
plot(t,Tcamara,'r','LineWidth',2);
xlabel('Time(s)');ylabel('Temperatura (°C)');legend('Salida');
grid on;
```

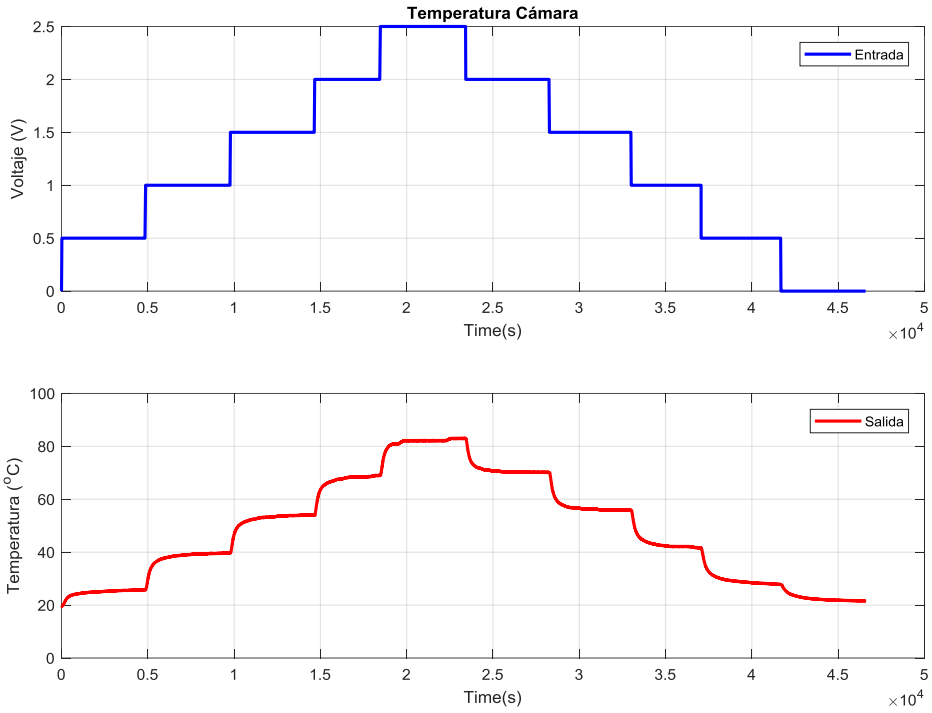


Figura 46. Datos tomados con el LabVIEW de la temperatura ambiente

Para observar la histéresis se toma un dato del régimen permanente para cada escalón de la temperatura de la cámara (véase Figura 47).

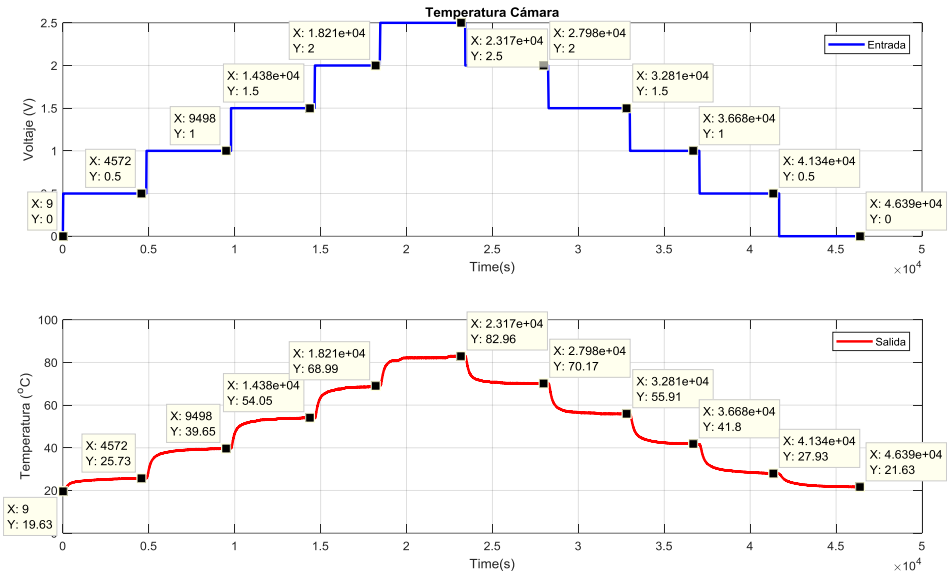


Figura 47. Datos para análisis de histéresis de la temperatura de la cámara

Datos para los escalones de subida

Voltaje	**Temperatura**
0	19.63
0.5	25.73
1	39.65
1.5	54.05
2	68.99
2.5	82.96

Tabla 4. Datos de Temperatura para escalones de subida

Datos para los escalones de bajada

Voltaje	**Temperatura**
2.5	82.96
2	70.17

1.5	55.91
1	41.8
0.5	27.93
0	21.63

Tabla 5. Datos de Temperatura para escalones de bajada

Al graficar los datos de bajada y de subida obtenemos que el sistema es lineal al trabajar con voltajes mayores 0.5 (véase Figura 48). El sistema presenta una histéresis mínima.

```
%Datos de temperatura para cada escalon temperatura camara
vc=0:0.5:2.5; %Voltajes de entrada
%Subida
Tsc=[19.63 25.73 39.65 54.05 68.99 82.96];
%Bajada
Tbc=[21.63 27.93 41.8 55.91 70.17 82.96];
figure(4)
plot(vc,Tsc,'b -o','LineWidth',2); hold on; grid on;
plot(vc,Tbc,'r -*','LineWidth',2);
xlabel('Voltaje (V)');ylabel('Temperatura (^oC)')
title('Histéresis de la temperatura en la cámara')
legend('Escalones de subida','Escalones de bajada')
```

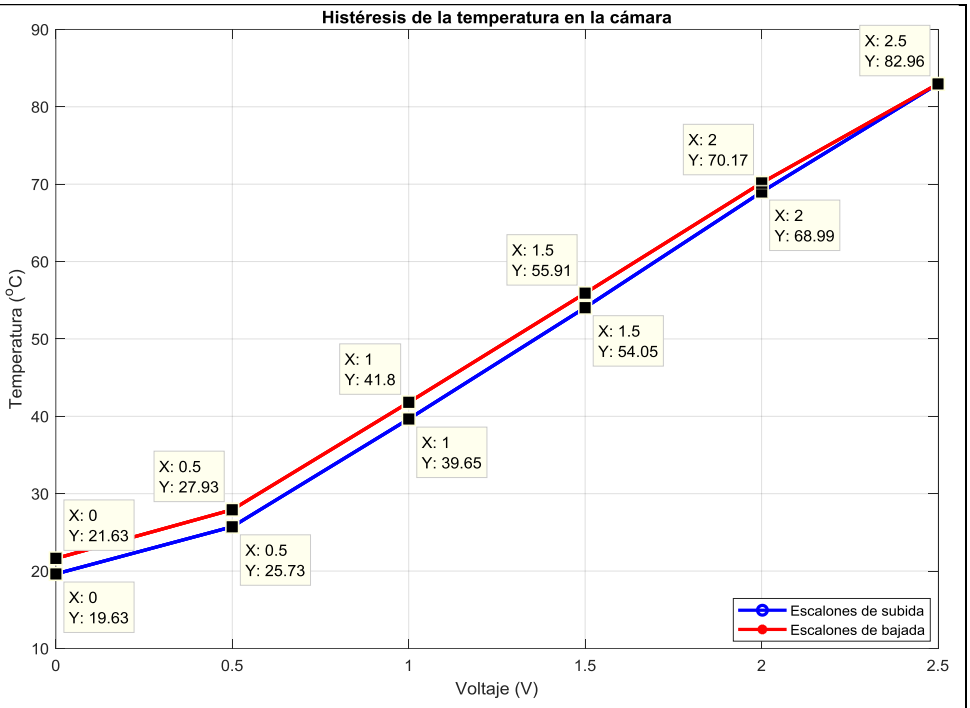


Figura 49. Histéresis de la temperatura de la cámara

5. Diagrama de conexión

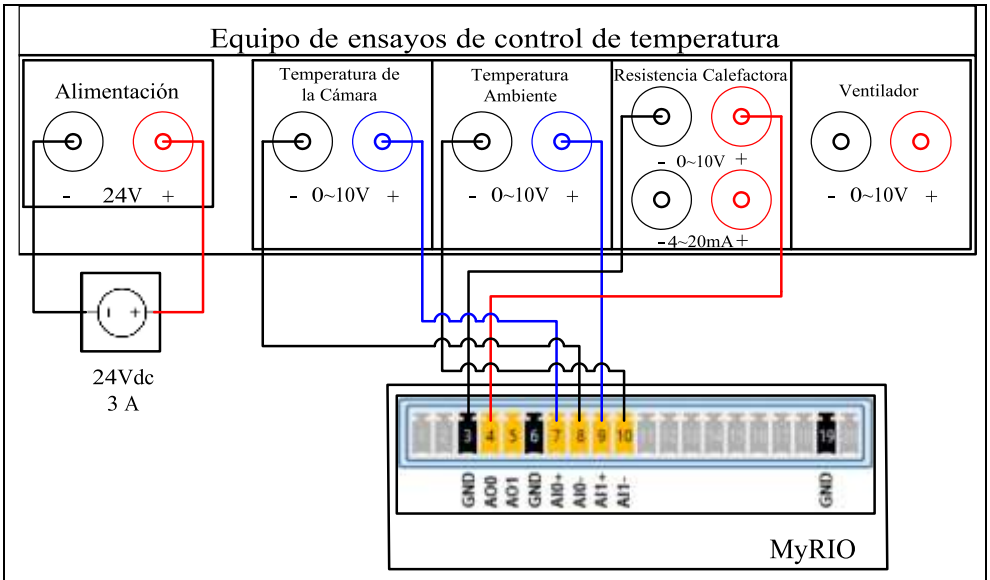


Figura 48. Diagrama de conexión

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

APÉNDICES

- **GUÍA DE PRÁCTICA 2: IDENTIFICACIÓN DE LA PLANTA**

FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: INGENIERÍA ELECTRÓNICA	**ASIGNATURA**: TEORÍA DE CONTROL

NRO. PRÁCTICA:	2	**TÍTULO PRÁCTICA**: **IDENTIFICACIÓN DE LA PLANTA**

OBJETIVO

4. Proporcionar las herramientas necesarias para que el estudiante pueda crear un proyecto en el software LabVIEW 2019 que permita el manejo de un equipo de ensayo de control de temperatura mediante el equipo NI-myRIO.
5. Conocer las características generales del equipo de ensayo de control de temperatura y NI myRIO.
6. Utilizar el software Matlab como herramienta para la obtención de función de transferencia del equipo de ensayos de control de temperatura

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

a) Modelado de sistemas dinámicos.

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Computadora S.O. Windows 10, 64 bits	1	-	-
Software Labview 2019.	1	-	-
Software Matlab 2017	1	-	-
NI myRIO	1	Nacional Instrument	-
Equipo de ensayos de control de temperatura	1	CTILab	-
Cables	Varios	-	-

Tabla 1. Equipos, instrumentos y software

3. Exposición

Kit NI myRIO: proporciona entradas y salidas configurables que permite ensenar e implementar múltiples conceptos de diseño con un dispositivo.

Incluye entradas analógicas, salidas analógicas, líneas E/S digitales, LEDs, un push-Button, un acelerómetro interno, un FPGA Xilinx y un procesador dual-core ARM Cortex-A9. [18]

Figura 1. NI myRIO [19]

Datos técnicos:

NI myRIO	
Memoria	No volátil (512 MB) DDR3 (256 MB)
FPGA	Xilinx Z-7010
Wireless Characteristics	Radio (IEEE 802.11 b,g,n). Frecuencia de banda (ISM 2.4 GHz). Ancho de banda (20MHz). Canales (USA 1 A 11, Internacional 1 a 13). TX (+10dBm max). Outdoor range (Hasta 150m). Directividad de la antena (Omnidireccional).

	Seguridad (WPA, WPA2, WPA2-Enterprise).
USB Ports	USB host-device port (2.0 Hi-Speed)
Acelerómetro	Numero de ejes (3). Rango (± 8g). Resolución (12bits). Taza de muestreo (800 S/s). Ruido (3.9 mg(rms) a 25ºC)
Requerimientos de potencia	Rango de voltaje de la fuente de alimentación (6 a 16 VCC). Consumo máximo de energía (14 W). Consumo de energía inactivo típico (2.6 W).

Tabla 2. Características del dispositivo NI-myRIO

***Equipo de ensayos de control temperatura**:* este dispositivo permite realizar ensayos de control de temperatura con picos de hasta 80 °C, equipado con una niquelina fabricada con alambre de nicromo, dos ventiladores 3010 para perturbaciones, sensores de temperatura tipo RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C.

El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.

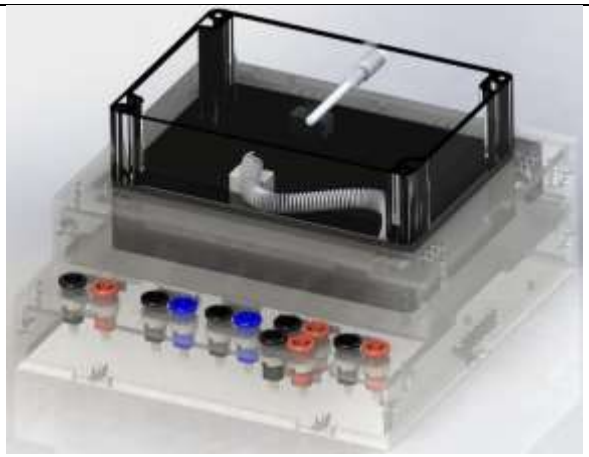


Figura 2. Equipo de ensayos de control de temperatura

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. **Programación en el software LabVIEW**

- Para abrir LabVIEW 2019 myRIO Toolkit dirigirse al siguiente link http://www.ni.com/product-documentation/14603/es/ y descargar e instalar "Módulo LabVIEW Real-Time" y "LabVIEW myRIO Toolkit".
- Abrir el acceso directo del software LabVIEW 2019. Conectar la NI myRIO al computador.
- Una vez abierto el software, dentro de la pantalla de inicio dar clic en "Launch the Getting Started Wizard", aquí configurará el equipo NI myRIO (véase Figura 3).

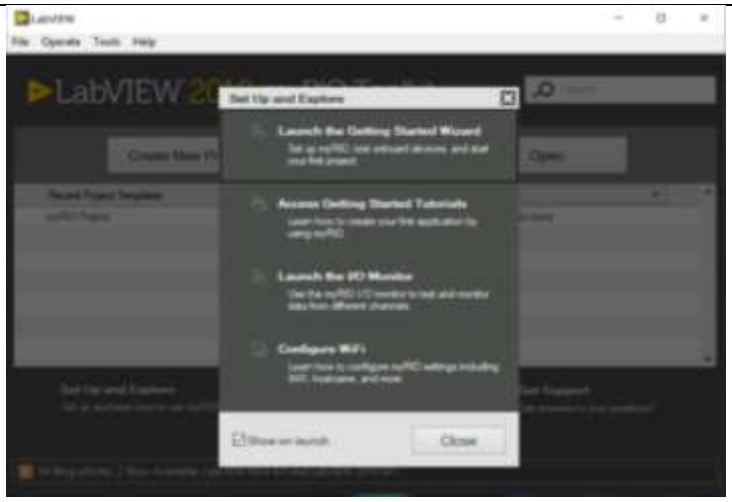


Figura 3. Pantalla de inicio.

- Se abre la ventana "Get Started with NI myRIO" (Iniciando con NI myRIO) donde aparecerá el nombre del dispositivo "NI-MyRIO-1900-03058D001" o similar (véase Figura 4).
- Clic en "Next" (Siguiente).

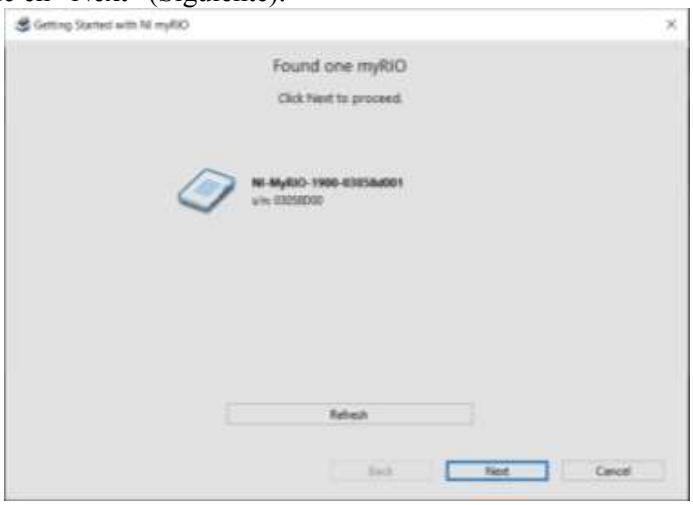


Figura 4. Pantalla de configuración del NI MyRIO.

- La siguiente configuración permite cambiar el nombre del dispositivo. Se deja el nombre por defecto.
- Clic en "Next".

Figura 5. Pantalla de configuración del NI MyRIO.

- Se puede comprobar el funcionamiento del dispositivo con el panel visto en pantalla (véase Figura 6). Al dar clic en los botones del panel ubicados al lado derecho, es posible observar en el dispositivo el cambio de estado de los LEDs. Al presionar el botón físico del dispositivo ubicado en el costado inferior del dispositivo, se puede observar en pantalla el cambio de estado del LED ubicado en la parte inferior. Se puede comprobar el funcionamiento del acelerómetro del dispositivo al moverlo, esto se observa en la parte izquierda de la ventana.

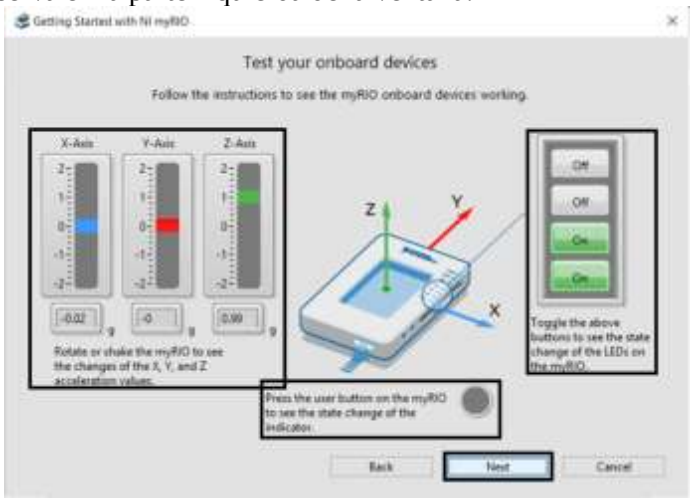


Figura 6. Pantalla de configuración del NI MyRIO.

- Una vez comprobada la conexión del dispositivo dar clic en “Next”.

- Se inicia con un nuevo Proyecto dando clic en “Start your first Project now [Recommended]” (Comienza tu primer Proyecto ahora) (véase Figura 7).

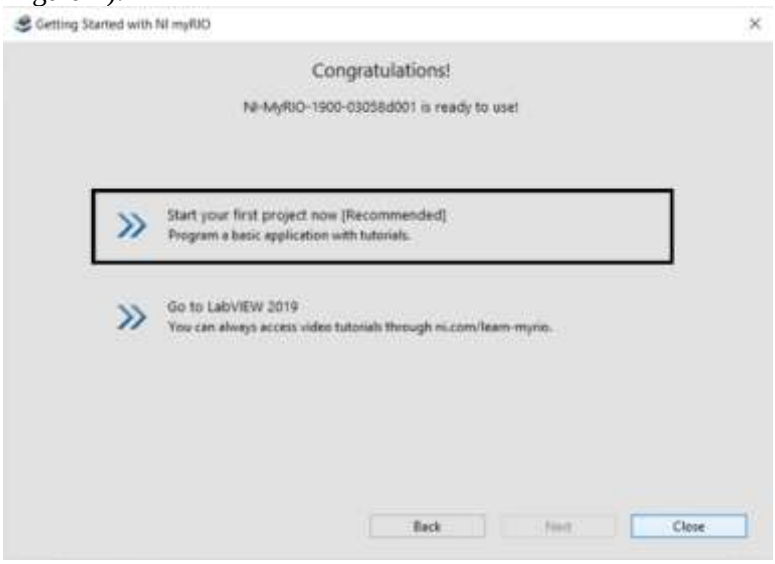


Figura 7. Pantalla de configuración del NI MyRIO.

- Dar clic en el botón “Create New Project” (Crear Nuevo Proyecto) (véase Figura 8).

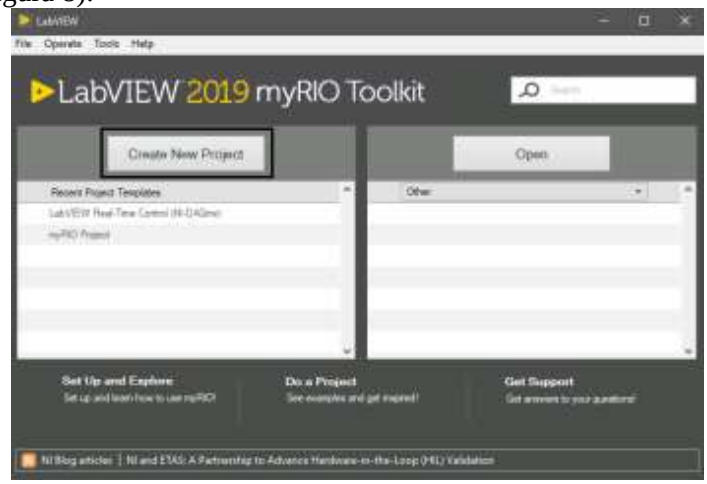


Figura 8. Pantalla de inicio.

- En la ventana “Create Project” (Crear Proyecto) seleccionar la opción “myRIO”
- Elegir la opción “myRIO Project” (myRIO Proyecto)

- Clic en “Next” (véase Figura 9)

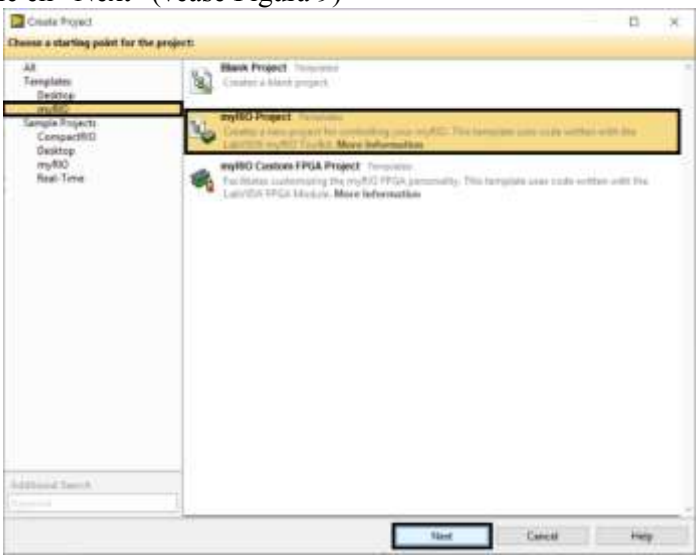


Figura 9. Pantalla de Creación de Proyecto.

- A continuación, se configura el proyecto colocando el nombre y la ubicación del archivo (Véase Figura 10).
- Seleccionar la opción “Plugged into USB” (Conectado a USB), se muestra el nombre y la dirección IP del dispositivo.
- Clic en “Finish” (Finalizar).

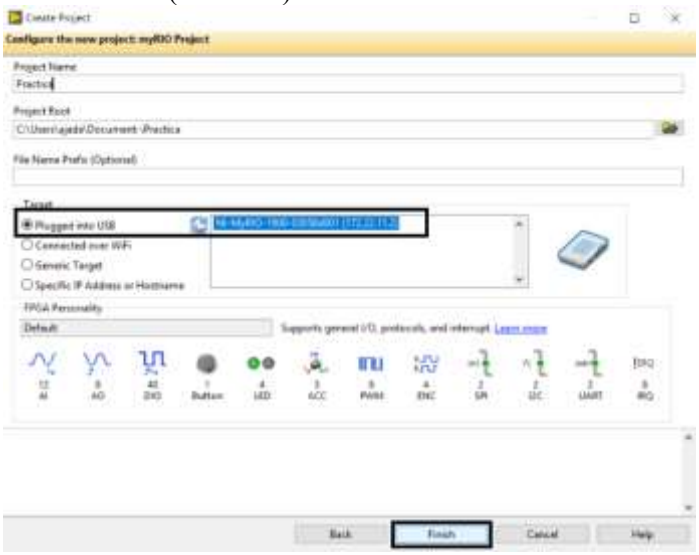


Figura 10. Pantalla para agregar un nuevo dispositivo.

La ventana con el árbol del proyecto contiene todos los archivos agregados en el mismo (véase Figura 11).

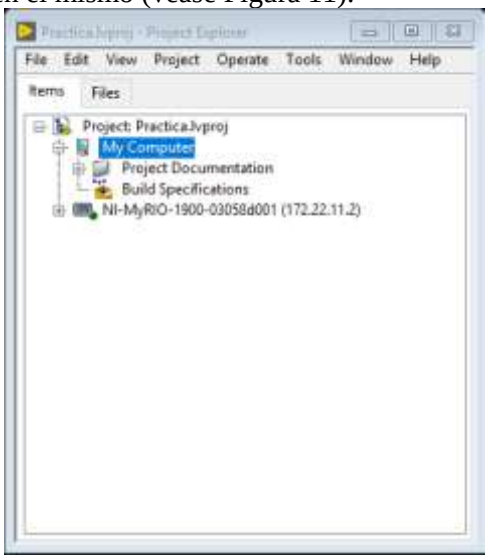


Figura 11. Ventana con el proyecto creado.

- Dar clic derecho sobre "NI-myRIO-1900-03058d001 (172.22.11.2)", seleccionar "New" (Nuevo) y "VI".

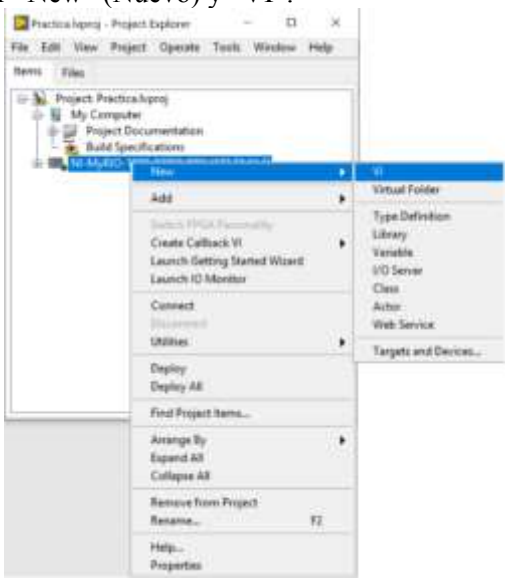


Figura 12. Agregar VI.

Al escoger “VI” (Interfaz Gráfica) se abre las ventanas “Block Diagram” (Diagrama de Bloques) y “Front Panel” (Panel Frontal) (véase Figura 13).

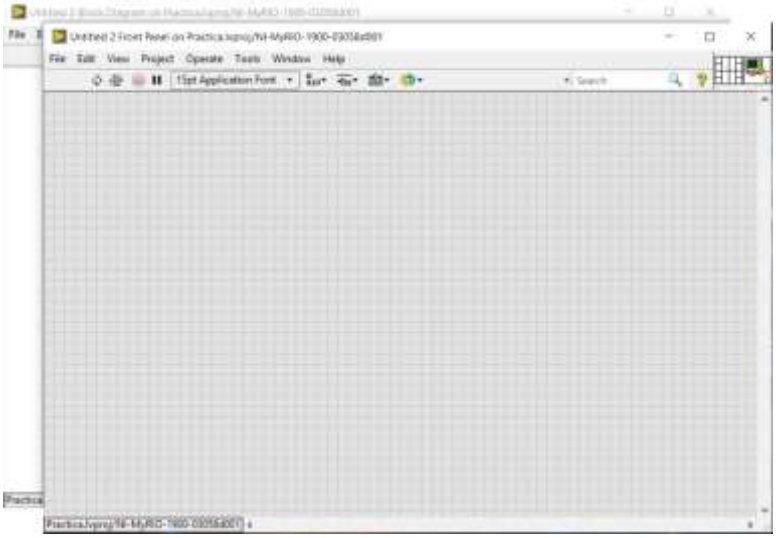


Figura 13. Ventanas del VI.

- Clic derecho en la ventana “Block Diagram” (Diagrama de Bloques).
- Seleccionar “Structures” (Estructuras).
- Arrastrar el bloque “While Loop” (Bucle While) a la ventana. (Véase Figura 14 y Figura 15)

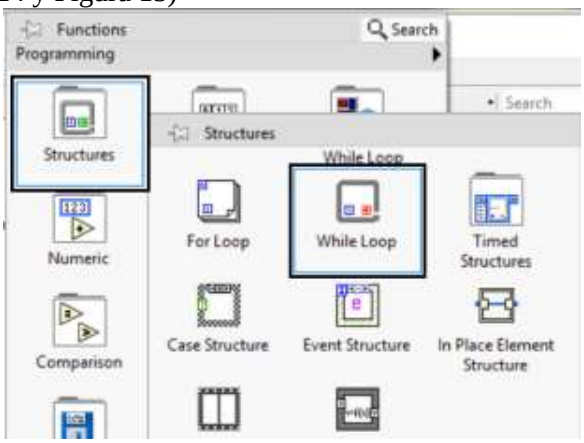


Figura 14. While Loop.

Figura 15. Bloque Bucle While

A continuación, se coloca los bloques de la MyRIO para leer las entradas analógicas del dispositivo:

- Se hace clic derecho en la pantalla.
- Se selecciona “MyRIO” (Véase Figura 16).
- Arrastrar el bloque “Analog In” (Entradas Analógicas) dentro del “While Loop”, donde se abrirá la ventana “Configure Analog Input” (Configuración de Entradas Analógicas) (Véase Figura 17).

Figure 16. Configuración de las entradas analógicas.

- En el cuadro “Channel” (Canal) seleccionar “C/AI0 (AI0)” (Véase Figura 17). Esta entrada se destinará al sensor de Temperatura de la Cámara del Equipo de ensayo de control de temperatura.

Figura 17. Ventana de Configuración de Entradas Analógicas

- Para el sensor de Temperatura Ambiente, dar clic en “+” (Véase Figura 18).

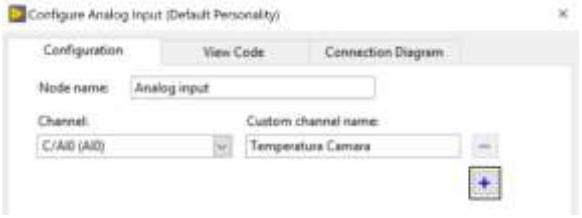


Figura 18. Configuración entrada analógica “C/AI1 (AI1)”.

- En “Channel” (Canal) la opción “C/AI1 (AI1)” (Véase Figura 19).
- Clic en “OK”

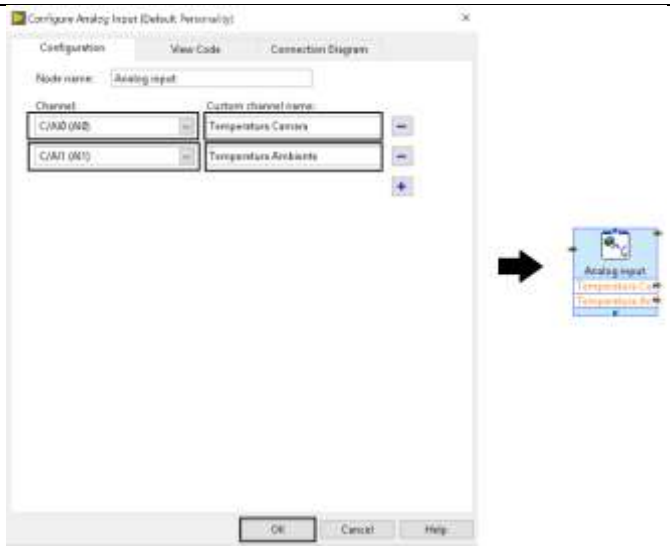


Figura 19. Configuración entrada analógica "C/AI1 (AI1)"

La configuración de salidas analógicas se realiza a continuación:

- Clic derecho en la ventana de "Block Diagram" (Diagrama de Bloques).
- Clic en "MyRIO" (véase Figura 20).
- Arrastrar el bloque "Analog Output" (Salidas Analógicas) dentro del "While Loop" (Bucle While).

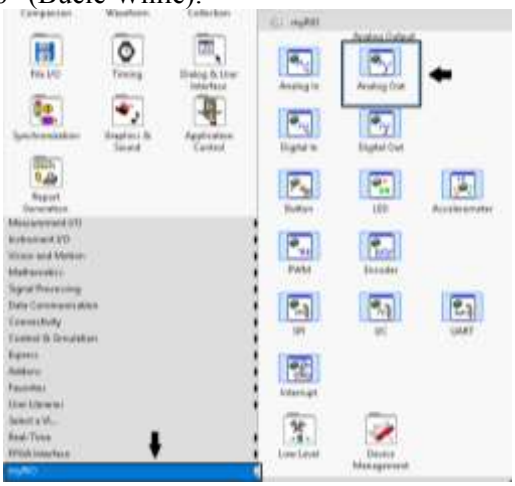


Figura 20. Bloque Analog Output

- En la ventana "Configure Analog Output", desplegar el menú de "Channel" (Canal) (Véase Figura 21).

- Seleccionar la opción “C/AO0 (AO0)”. Esta se usará para comandar la Resistencia Calefactora del Equipo de ensayo de control de temperatura.

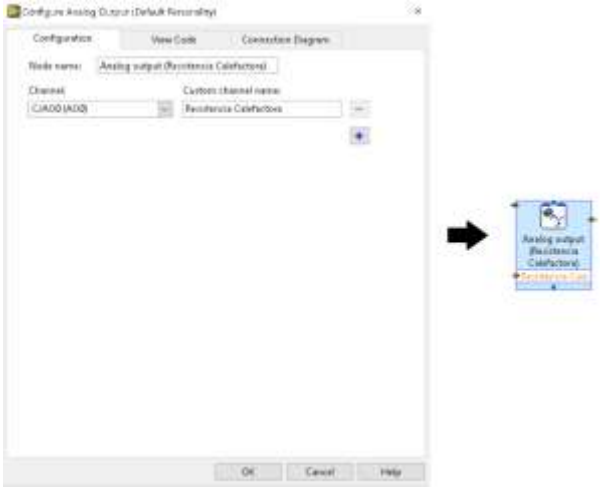


Figura 21. Configuración de salida analógica

Se coloca los indicadores y las multiplicaciones respectivas para convertir de voltajes a grados Celsius.

- Clic derecho en la pantalla de “While Loop”
- Seleccionar “Numeric” (Numérico) (véase Figura 22)
- Arrastrar bloque “Multiply” (Multiplicar)

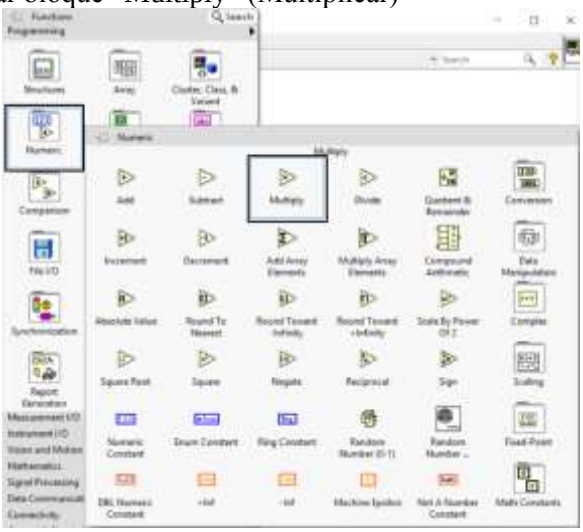


Figura 22. Bloque para multiplicación

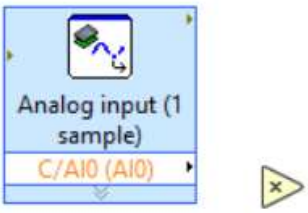


Figura 23. Multiplicación para las entradas analógicas

- En el bloque de "Multiply" dar clic derecho en la señal de salida, seleccionar "Create Indicator" (Véase Figura 24)

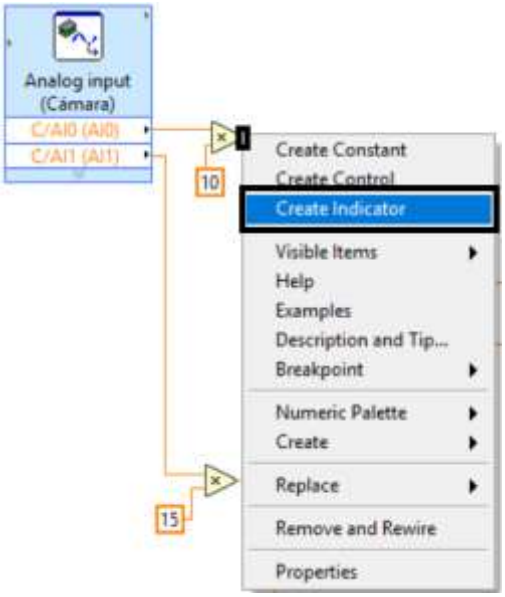


Figura 24. Creando indicadores

Esto se repite para colocar todos los indicadores señalados en la siguiente figura:

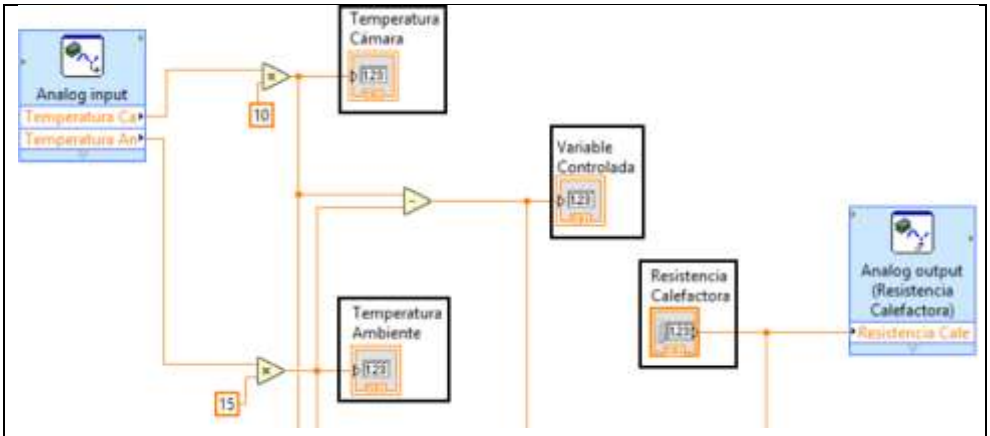


Figura 25. Creando indicadores

Para guardar los datos se realizan los siguientes pasos:

- Clic derecho en "Block Diagram", selecciona "File I/O" (Archivo E/S) y arrastrar el bloque "Write To Measurement File" (Escribir a archivo de datos) dentro del "While Loop" (véase Figura 26).

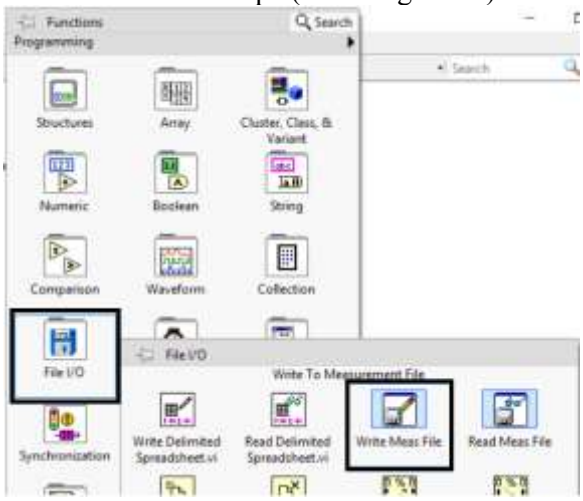


Figura 26. Bloque para guardar datos

- En la ventana "Configure Write To Measurement File" (Configuración del bloque escribir a archivos de datos) se coloca la dirección del archivo, el formato, entre otros.

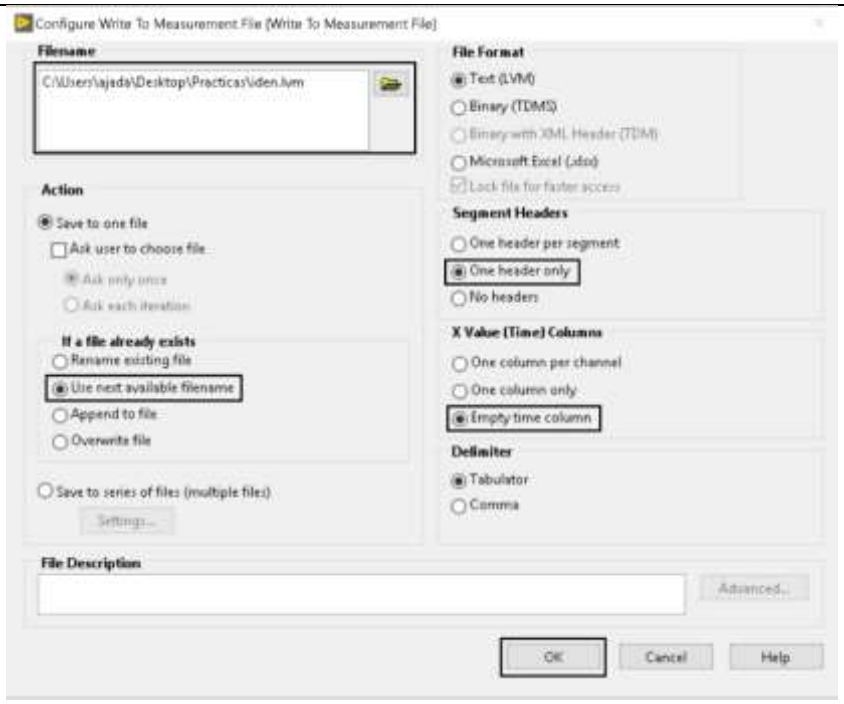


Figura 27. Ventana para Configuración de Archivo para guardar los datos

- Clic en el icono de la carpeta (Véase figura 28)
  Para guardar los datos se debe crear una carpeta en el Escritorio, una vez creada la carpeta se coloca el nombre del archivo en este caso "iden.lvm".

Figura 28. Dirección para guardar los datos

La programación de bloques se muestra en la figura 29.

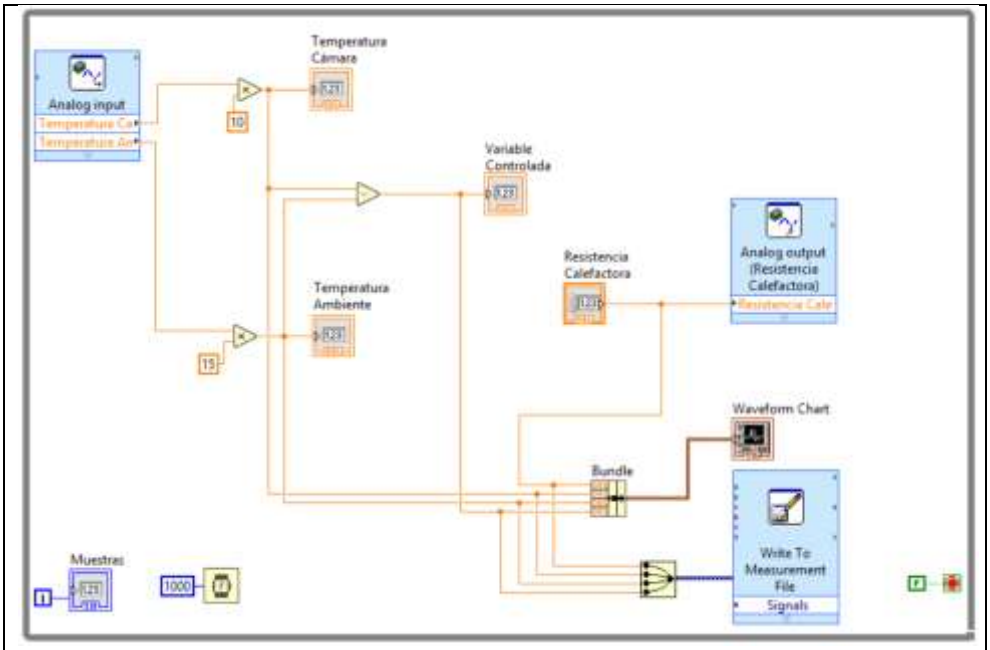


Figura 29. Programación de bloques

Nota:

- La señal de entrada analógica de la Cámara se multiplica por 10 puesto que el transmisor da un voltaje de 0 a 10 V para 0 a 100 ºC. La señal de la Temperatura Ambiente se multiplica por 15 puesto que el transmisor da un voltaje de 0 10V para 0 a 100 ºC.
- La señal de salida para la Resistencia Calefactora tiene un rango de 0 a 10V. Cabe recalcar que las señales de entrada y salida del Equipo de ensayo de control de temperatura son voltajes continuos.
- Para encontrar un bloque rápidamente hacer clic derecho en la ventana y luego clic en "Search". Aquí se escribe lo que se desea encontrar (véase Figura 30 y 31).

Figura 30. Search

Figura 31. Search

La ventana "Front Panel" se muestra en la figura 32.

Figura 32. Ventana Front Panel

Adquisición de datos

- Ejecutamos el código cuando se desee capturar las señales (véase Figura 33).

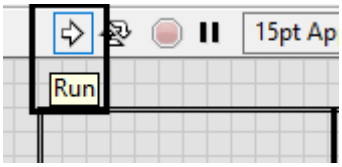


Figura 33. Ejecutar código

- Para el análisis de la linealidad del equipo se dará un rango de voltaje de 0.5 y 1 V a la Resistencia Calefactora.
- En cada escalón dado se debe esperar a que el sistema alance su estabilidad.

Accediendo a los datos obtenidos

- Para acceder al archivo de datos debemos utilizar un navegador de internet, donde ingresaremos la dirección IP de la NI myRIO seguido de "/files" (véase Figura 34).

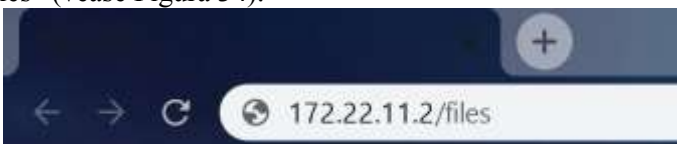


Figura 34. Ingresar dirección IP

- En el campo de usuario ingresamos "admin" y dejamos el campo de contraseña en blanco (véase Figura 35).

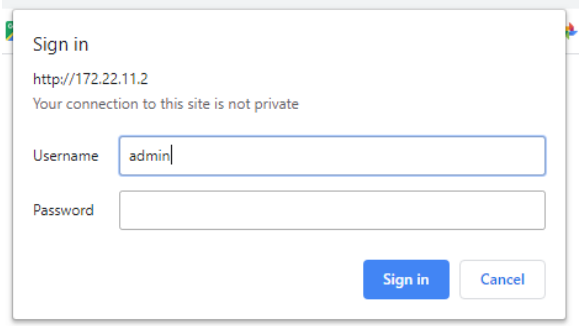


Figura 35. Ingresar credenciales

- Accedemos al directorio seleccionado anteriormente y descargamos el archivo. En caso de que simplemente se muestre en el navegador debemos dar clic derecho sobre los datos y guardarlo como texto. (véase Figura 36)

Figura 36. Guardar datos

2. Importando datos en Matlab

- Abrir el software Matlab R2017b.

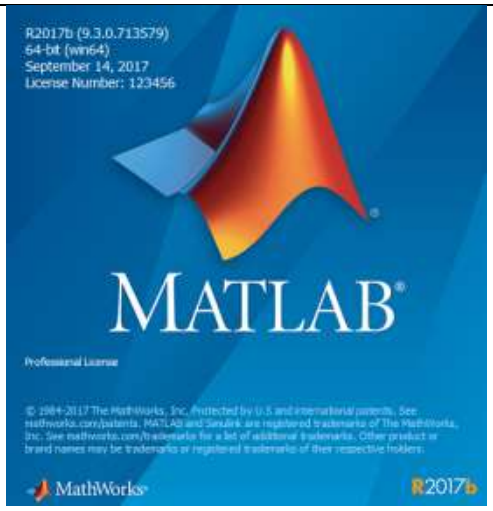


Figura 37. Software MATLAB R2017b.

- Clic en la ventana "HOME"
- En la sección "VARIABLE" (Variable), seleccionar "Import Data" (Importar Datos) (véase Figura 38).

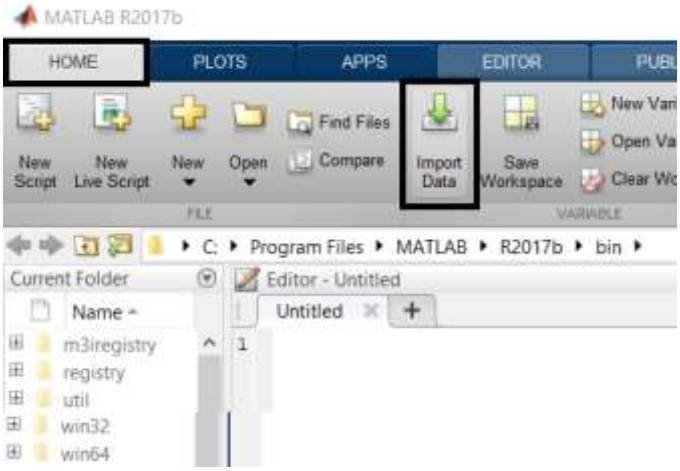


Figura 38. Importar datos adquiridos.

Se abre la ventana "IMPORT"

- En la sección "IMPORTED DATA", en la opción "Output Type" se escoge "Column vectors" (véase Figura 39).

- Para cambiar el nombre de cada columna, clic sobre la celda del nombre y escribir el nombre respectivo (véase Figura 39).
- Dar clic en “Import Vector”.

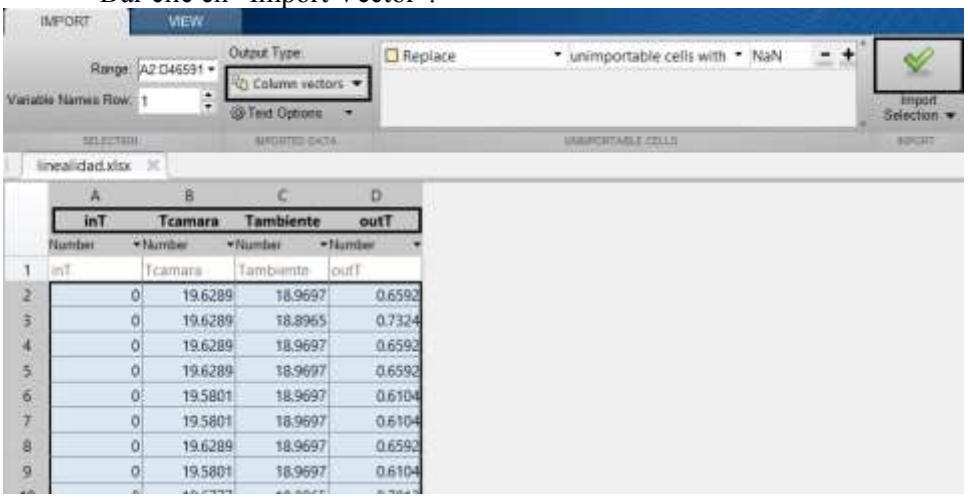


Figura 39. Importando datos

<u>Guardar los datos ubicados en el Workspace</u>

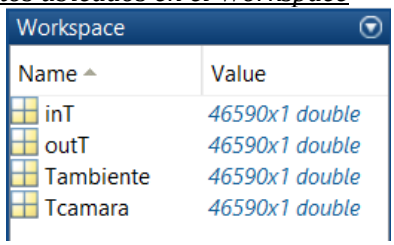


Figura 40. Variables Workspace

- En la pestaña “HOME”, en la sección “VARIABLE”, seleccionar “Save Workspace” (Guardar Espacio de Trabajo) (véase Figura 41).

Nota: Guardar los datos en la misma carpeta que el “Script”, el nombre del archivo en este caso se llamará datos.mat.

Figura 41. Guardando los datos

- Se crea un nuevo proyecto escogiendo "New Script" (Nuevo Script)

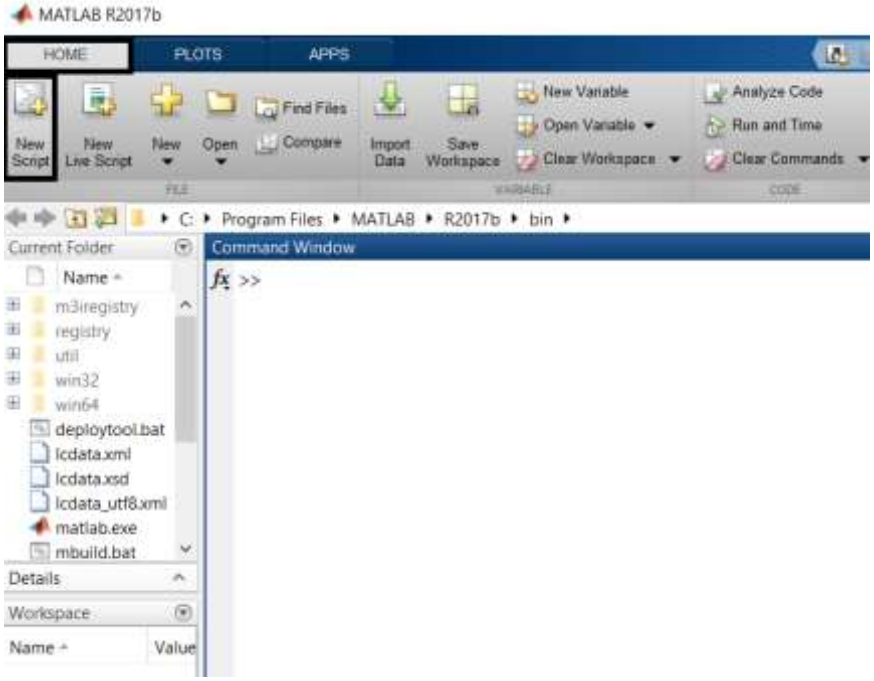


Figura 42. Creando "New Script"

3. Identificación de la planta con la diferencia entre temperatura del cámara y temperatura ambiente

- Se procede a cargar y graficar los datos de variable controlada la cual será diferencia entre la temperatura de la cámara y la temperatura ambiente (inT) (véase Figura 43).

```
load datos_ident.mat
ds=1; %Tiempo de muestreo
t=0:ds:length(inT)-ds; %vector de tiempo
figure(1)
subplot(211); %grafica de la variable controlada
plot(t,inT,'b','LineWidth',2); xlabel('Time(s)'); ylabel('Voltaje (V)');legend('Entrada')
grid on
title('Diferencia entre Temperatura Cámara y Temperatura Ambiente')
subplot(212); %grafica de la señal de control
plot(t,outT,'r','LineWidth',2);xlabel('Time(s)');ylabel('Voltaje (V)');legend('Salida')
grid on
```

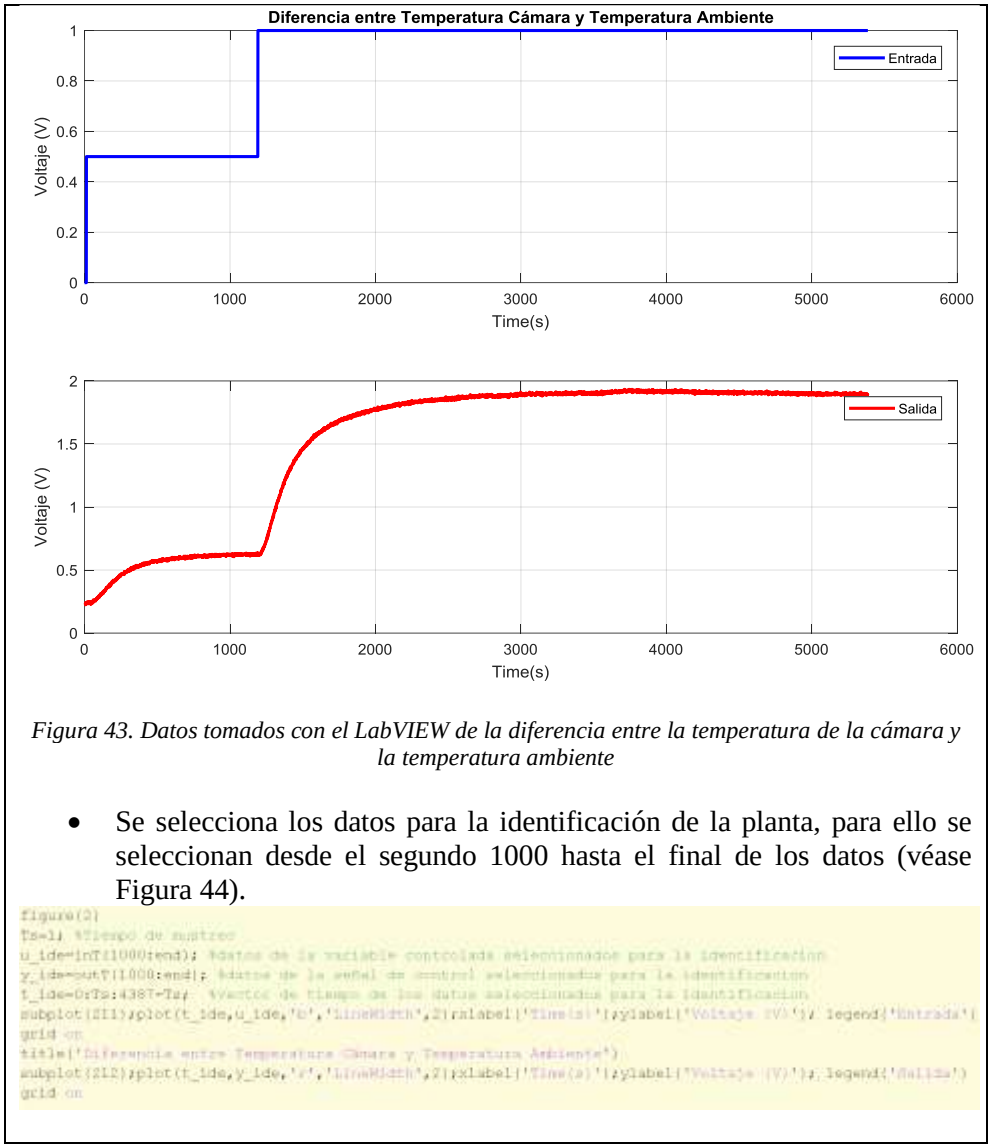


Figura 43. Datos tomados con el LabVIEW de la diferencia entre la temperatura de la cámara y la temperatura ambiente

- Se selecciona los datos para la identificación de la planta, para ello se seleccionan desde el segundo 1000 hasta el final de los datos (véase Figura 44).

```
figure(2)
Ts=1; %Tiempo de muestreo
u_ide=inT(1000:end); %datos de la variable controlada seleccionados para la identificacion
y_ide=outT(1000:end); %datos de la señal de control seleccionados para la identificacion
t_ide=0:Ts:4387-Ts;  %vector de tiempo de los datos seleccionados para la identificacion
subplot(211);plot(t_ide,u_ide,'b','LineWidth',2);xlabel('Time(s)');ylabel('Voltaje (V)');legend('Entrada')
grid on
title('Diferencia entre Temperatura Cámara y Temperatura Ambiente')
subplot(212);plot(t_ide,y_ide,'r','LineWidth',2);xlabel('Time(s)');ylabel('Voltaje (V)');legend('Salida')
grid on
```

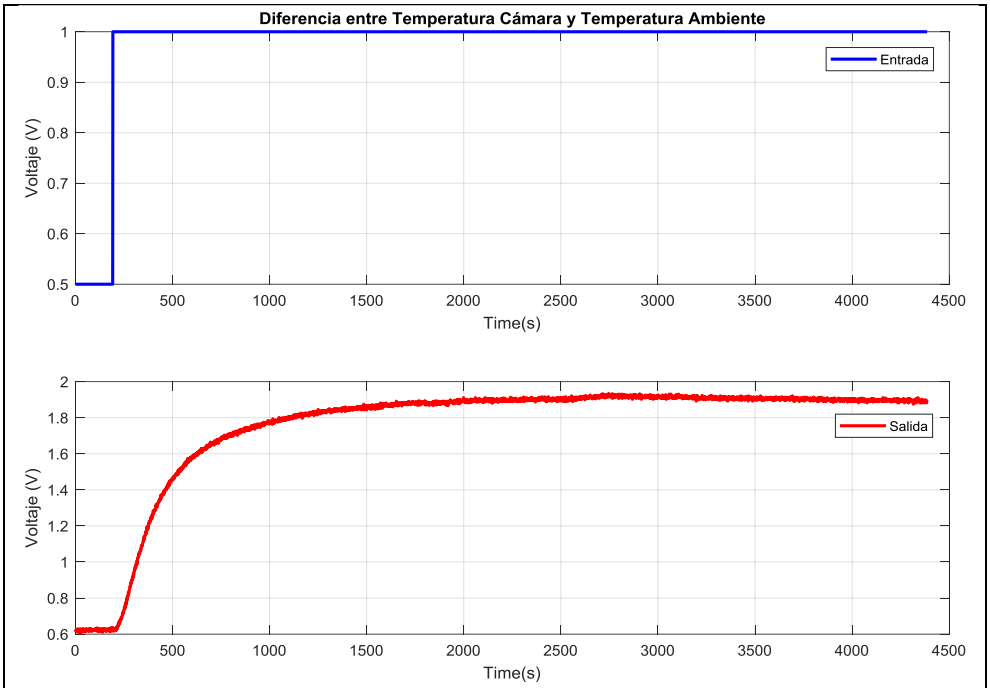


Figura 44. Datos para la identificación de la diferencia entre la temperatura de la cámara y la temperatura ambiente

- Se remueve el offset tanto de los datos de entrada y los de salida. (A los datos de entrada se le remueve el valor de la primera muestra. A los datos de salida se le resta el promedio de las primeras 50 muestras (véase Figura 45).

```
u_ide=u_ide-u_ide(1); %resta de la primera muestra
y_ide=y_ide-mean(y_ide(1:50)); %resta del promedio de las primera 50 muestras
figure(3)
subplot(211);plot(t_ide,u_ide,'b','LineWidth',2);xlabel('Tiempo(s)');ylabel('Voltaje (V)'); legend('Entrada')
grid on
title('Diferencia entre Temperatura de la Cámara y Temperatura Ambiente')
subplot(212);plot(t_ide,y_ide,'r','LineWidth',2);xlabel('Tiempo(s)');ylabel('Voltaje (V)'); legend('Entrada')
grid on
```

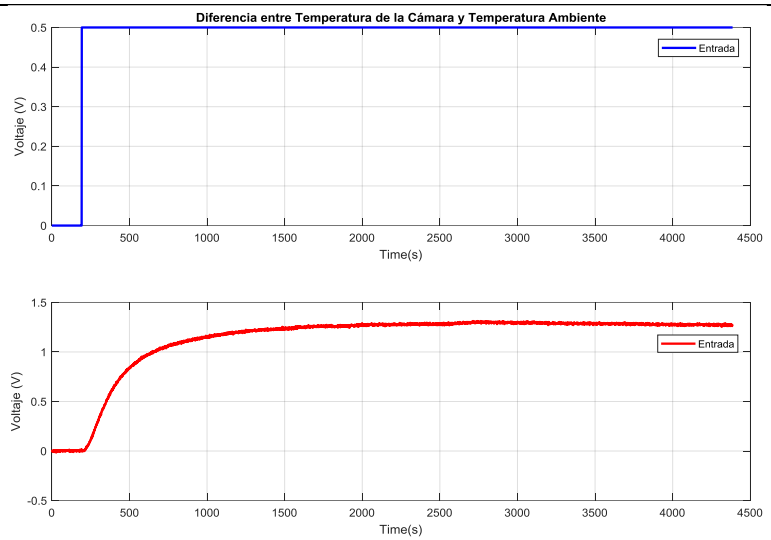


Figura 45. Datos para identificación sin offset de la diferencia entre la temperatura de la cámara y la temperatura ambiente

- El sistema tiene un retardo desde que se le da un escalón hasta que entra en funcionamiento por lo tanto para obtener la funcione de trasferencia se coloca un cero.

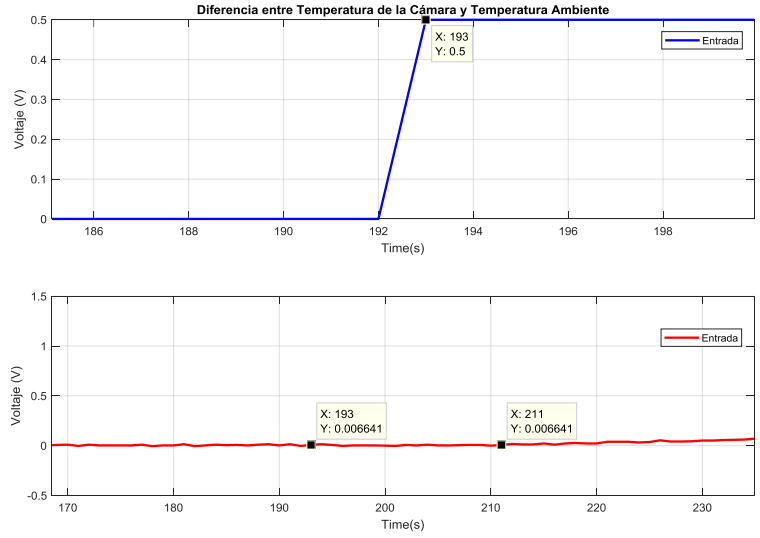


Figura 46. Tiempo muerto

- El comando tfest nos devuelve la función de transferencia al colocar los datos de entrada y salida, el número de polos y ceros y el tiempo muerto de la planta (datos, #polos, #ceros, tiempo muerto).

```
data=iddata(y_ide,u_ide,Ts); %se coloca en un solo objeto los datos
sys1=tfest(data,1,1,0) % Sistema con un polo y un cero
figure
subplot(221);
compare(data,sys1);title('Un polo')

sys2=tfest(data,2,1,0) %Sistema con dos polos y un cero
subplot(222);
compare(data,sys2);title('Dos polos')

sys3=tfest(data,3,1,0) %Sistema con tres polos y un cero
subplot(223);
compare(data,sys3);title('Tres polos')

sys4=tfest(data,4,1,0) %Sistema con cuatro polos y un cero
subplot(224);
compare(data,sys4);title('Cuatro polos')
```

- Comparando los datos obtenidos se puede observar que los sistemas con un mayor porsentaje son el de tercer y cuarto orden (vease Figura 47).

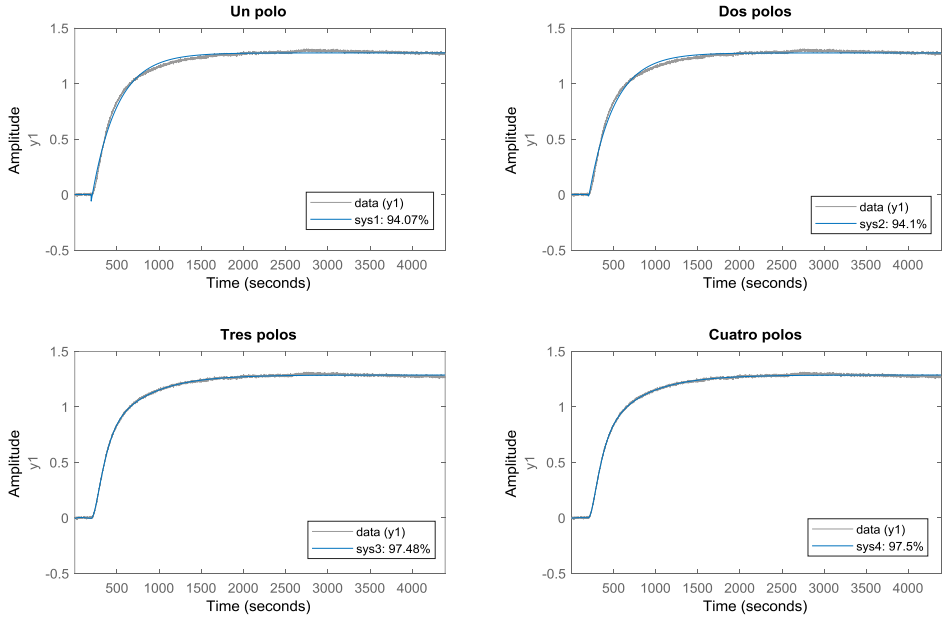


Figura 47. Comparación entre los datos tomas y obtenidos con el comando tfest

- Al analizar los polos y los ceros se observa que el sistema de tercer orden es el más adecuado (véase Figura 48).

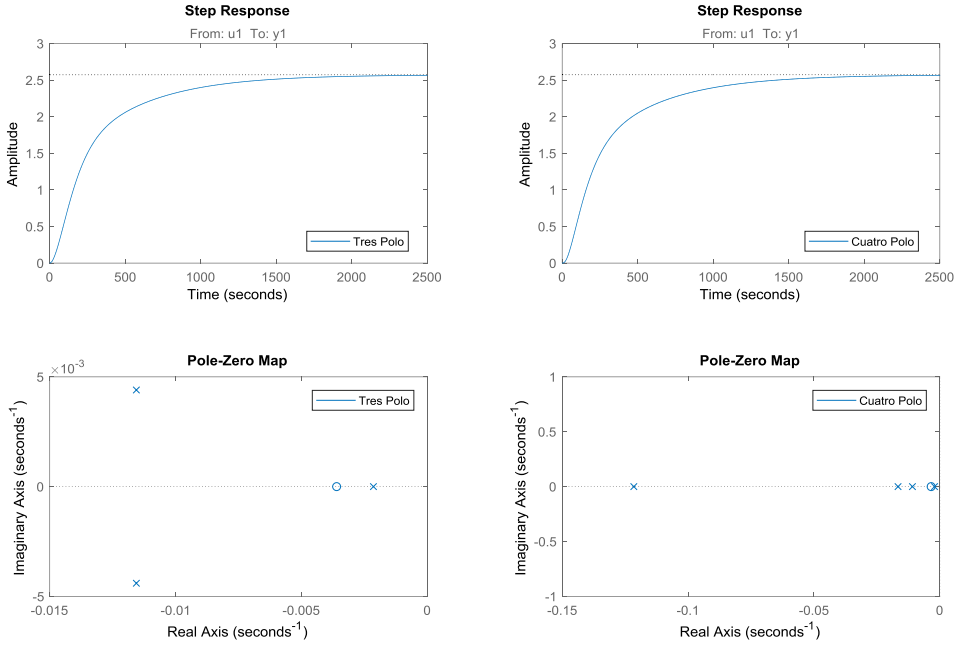


Figura 48. Polos y ceros de los sistemas de tercer y cuarto orden

La respuesta al escalón del sistema de tercer orden tiene los siguientes parámetros

```
s3 =
  struct with fields:

        RiseTime: 758.6720
    SettlingTime: 1.5681e+03
     SettlingMin: 23.1844
     SettlingMax: 25.7018
       Overshoot: 0
      Undershoot: 0
            Peak: 25.7018
        PeakTime: 2.8593e+03
```

La función de transferencia que mejor se ajusta a los datos de la diferencia entre la temperatura de la cámara y la temperatura ambiente se muestra en la ecuación 1.

$$s3 = \frac{0.0002341s + 8.433 * 10^{-7}}{s^3 + 0.02527s^2 + 0.0002026s + 3.277 * 10^{-7}} \qquad (1)$$

4. Identificación de la planta con la temperatura de la cámara

- Para la identificación de la planta con los datos de la temperatura de la cámara se repite el paso 3. Para los datos de temperatura de la cámara se utiliza (Tcamara) y los datos de entrada (inT).

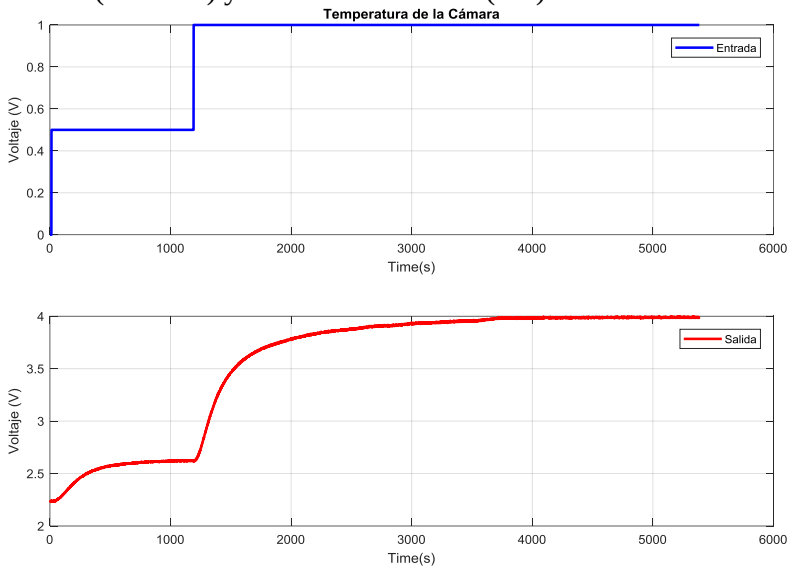


Figura 49. Datos tomados con el LabVIEW de la temperatura de la cámara

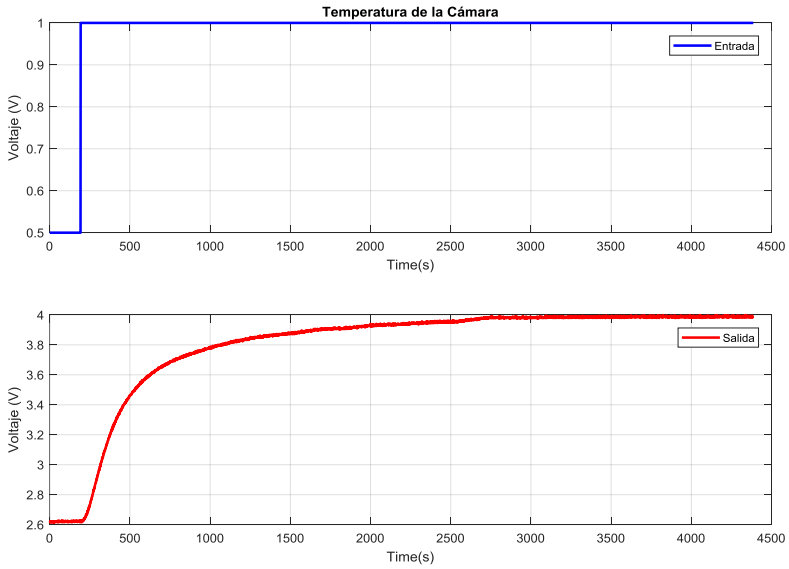


Figura 50. Datos recortados para la identificación de la temperatura de la cámara

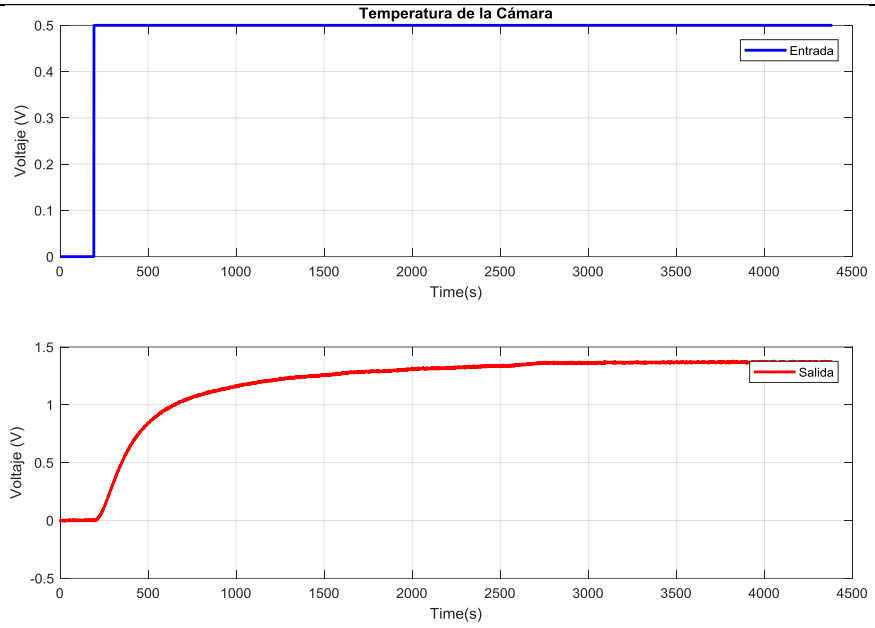


Figura 51. Datos para identificación sin offset de la temperatura de la cámara

- El sistema tiene un retardo desde que se le da un escalón hasta que entra en funcionamiento por lo tanto para obtener las funciones de trasferencia se coloca un cero.

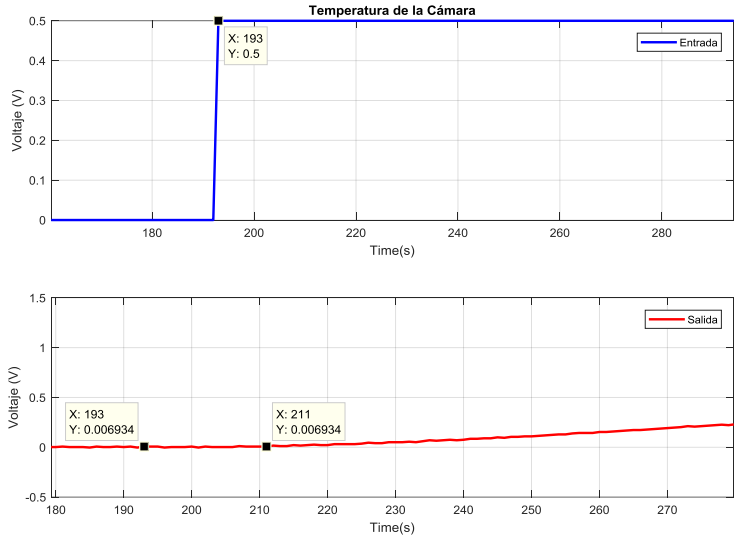


Figura 52. Tiempo muerto

- Se obtiene los modelos para primer, segundo, tercer y cuarto orden. Los sistemas de tercer y cuarto orden tienen mayor porcentaje de ajuste de datos (véase Figura 53).

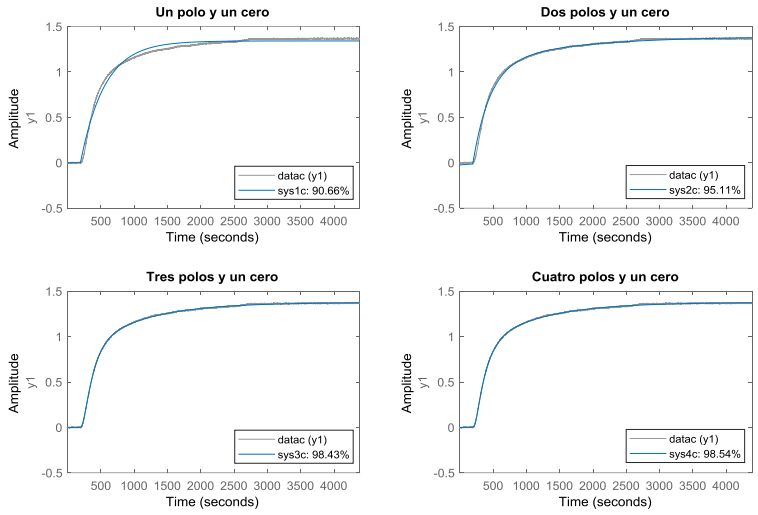


Figura 53. Sistemas de primer, segundo, tercer y cuarto orden.

- Al analizar los polos y los ceros se observa que el sistema de tercer orden es el más adecuado (véase Figura 54).

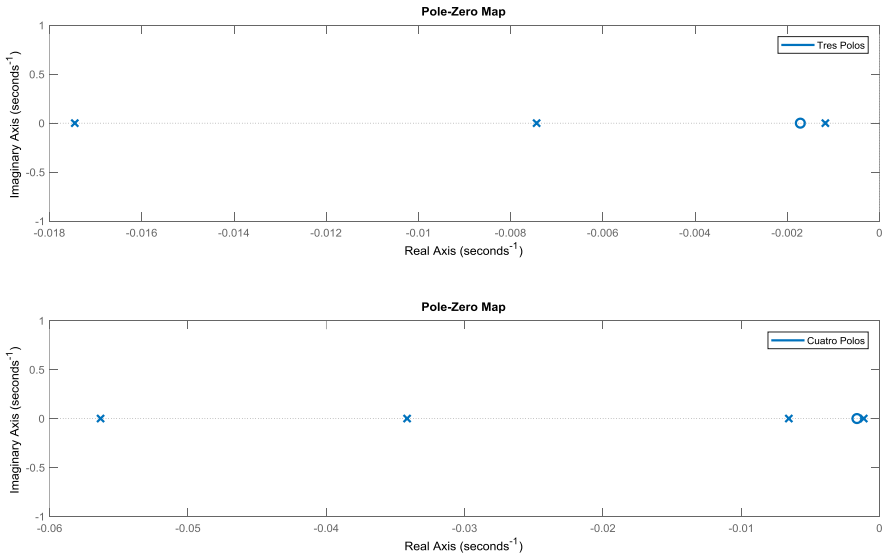


Figura 54. Polos y ceros de los sistemas de tercer y cuarto orden

La respuesta al escalón del sistema de tercer orden tiene los siguientes parámetros

```
S3c =
  struct with fields:

        RiseTime: 1.1248e+03
    SettlingTime: 2.5523e+03
     SettlingMin: 2.4002
     SettlingMax: 2.7481
       Overshoot: 0
      Undershoot: 0
            Peak: 2.7481
        PeakTime: 4.4130e+03
```

La función de transferencia que mejor se ajusta a los datos de la temperatura de la cámara se muestra en la ecuación 2.

$$s3 = \frac{0.0002453s + 4.212 * 10^{-7}}{s^3 + 0.02612s^2 + 0.0001594s + 1.529 * 10^{-7}} \qquad (2)$$

5. Identificación de la planta con la temperatura de la cámara con tiempo muerto

- Para la identificación de la planta con los datos de la temperatura de la cámara se repite el paso 3. Se identifica la planta con el tiempo muerto de 17s (véase Figura 55).

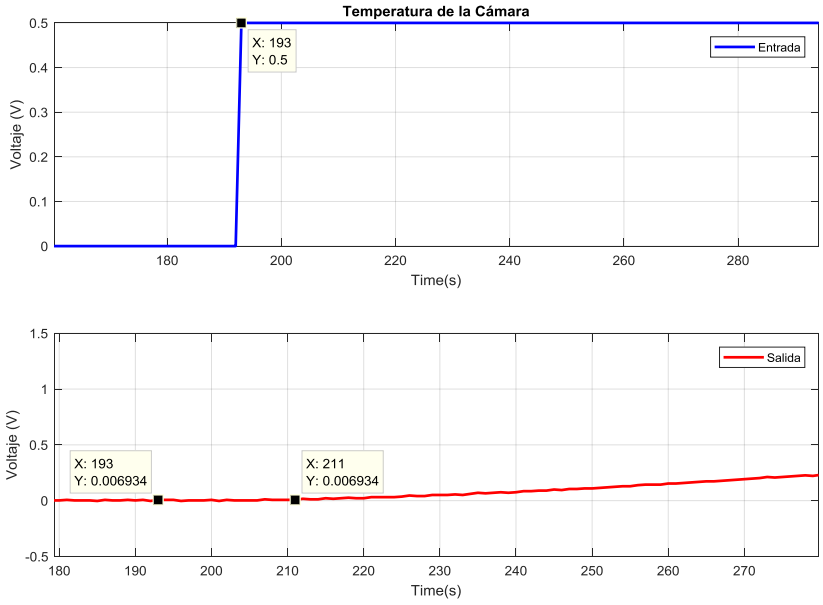


Figura 55. Tiempo muerto

- Se obtiene los modelos para primer, segundo, tercer y cuarto orden. Los sistemas de tercer y cuarto orden tienen mayor porcentaje de ajuste de datos (véase Figura 55). Colocando el retardo de 17s en el código para la identificación.

```
datac=iddata(y_idec,u_idec,Ts); %se coloca en un solo objeto los datos
sys1c=tfest(datac,1,1,0) % Sistema con un polo y un cero
figure
subplot(221);
compare(datac,sys1c);title('Un polo con Tiempo Muerto')

sys2c=tfest(datac,2,0,17) %Sistema con dos polos y un cero
subplot(222);
compare(datac,sys2c);title('Dos polos con Tiempo Muerto')

sys3c=tfest(datac,3,0,17) %Sistema con tres polos y un cero
subplot(223);
compare(datac,sys3c);title('Tres polos con Tiempo Muerto')

sys4c=tfest(datac,4,0,17) %Sistema con cuatro polos y un cero
subplot(224);
compare(datac,sys4c);title('Cuatro polos con Tiempo Muerto')
```

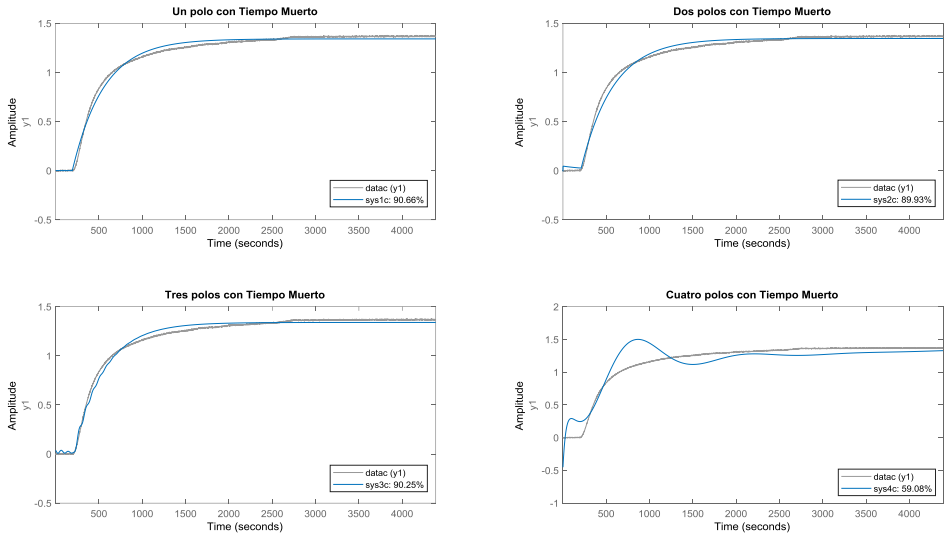


Figura 56. Sistemas de primer, segundo, tercer y cuarto orden.

- Al analizar los polos se observa que los sistemas de segundo y tercer orden son los más adecuado (véase Figura 57).

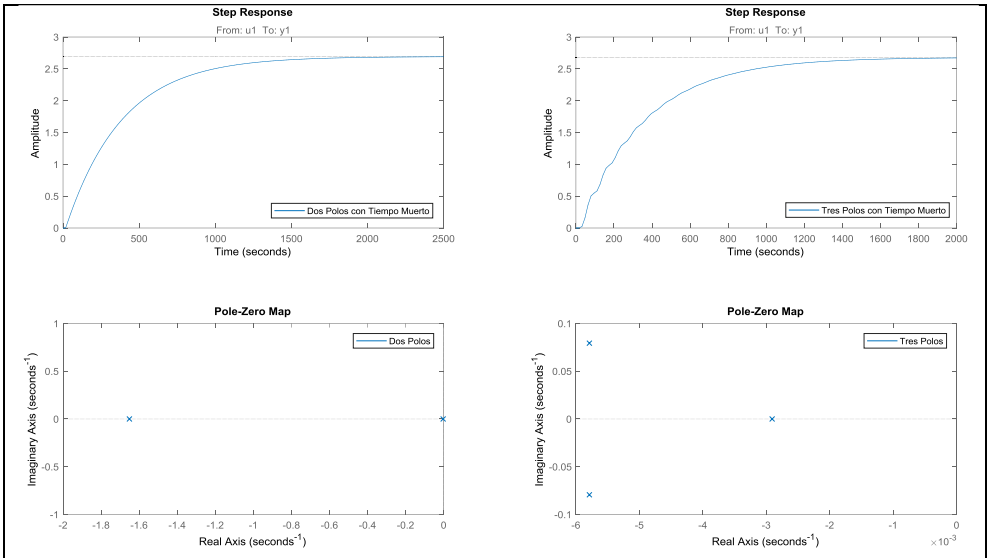


Figura 57. Polos y ceros de los sistemas de segundo y tercer orden.

La respuesta al escalón del sistema de tercer orden tiene los siguientes parámetros

```
S3cr =
  struct with fields:

        RiseTime: 754.4157
    SettlingTime: 1.3643e+03
     SettlingMin: 2.4212
     SettlingMax: 2.6787
       Overshoot: 0
      Undershoot: 0
            Peak: 2.6787
        PeakTime: 2.6775e+03
```

La función de transferencia que mejor se ajusta a los datos de la temperatura de la cámara se muestra en la ecuación 3.

$$s3 = e^{-17s} * \frac{4.936 * 10^{-5}}{s^3 + 0.01448s^2 + 0.00637s + 1.842 * 10^{-5}} \qquad (3)$$

6. Diagrama de conexión

En la figura 58 se muestra el diagrama de conexión de dispositivos.

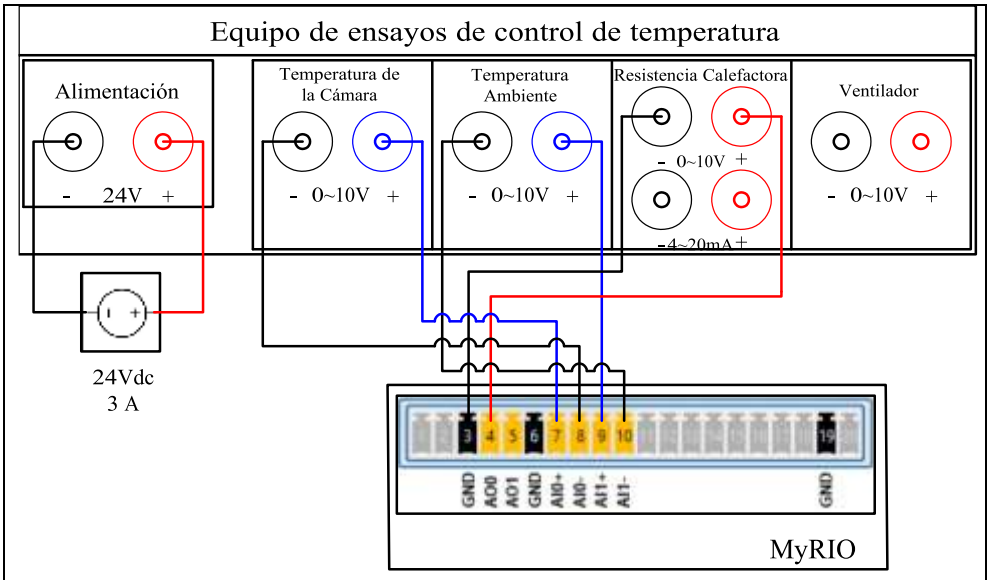


Figura 58. Diagrama de conexión

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

APÉNDICES

- **GUÍA DE PRÁCTICA 3: DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR PID USANDO UN COMPUTADOR CON LABVIEW Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA**

FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: Electrónica y automatización		**ASIGNATURA**: Teoría de control
NRO. PRÁCTICA	3	**TÍTULO PRÁCTICA**: **DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR PID USANDO UN COMPUTADOR CON LABVIEW Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA**

OBJETIVO

1. Proporcionar las herramientas necesarias para que el estudiante pueda implementar un controlador en el software LabVIEW v2019 que permita el manejo de un sistema térmico.
2. Conocer las características generales del equipo NI myRio 1900
3. Conocer las herramientas del software Matlab r2017b para el diseño de controladores PID.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

a) Creación de un proyecto y familiarización con el software LabVIEW 2019
b) Estructura de un sistema de control en lazo cerrado
c) Sintonización de un controlador PID

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Computadora S.O. Windows 7 o superior, 32 o 64 bits	1	-	-
Software Matlab r2017b	1	MathWorks	-

Software LabVIEW 2019	1	Nacional Instruments	
Kit NI MyRio 1900	1	National Instruments	-
Equipo de ensayos de control de temperatura	1	CTI Lab	-
Fuente DC 24 volts 3.2 Amps	1	Siemens	-

Tabla 1. Equipos, instrumentos y software

3. Exposición

***Kit NI MyRio*:** proporciona entradas y salidas configurables que permite ensenar e implementar múltiples conceptos de diseño con un dispositivo.

Incluye entradas analógicas, salidas analógicas, líneas E/S digitales, Leds, un push-Button, un acelerómetro interno, un FPGA Xilinx y un procesador dual-Core ARM Cortex-A9. [18]

Figura 1. Módulo NI MyRio. [18]

Datos técnicos:

NI myRIO	
Memoria	No volátil (512 MB) DDR3 (256 MB)

FPGA	Xilinx Z-7010
Wireless Characteristics	Radio (IEEE 802.11 b,g,n). Frecuencia de banda (ISM 2.4 GHz). Ancho de banda (20MHz). Canales (USA 1 A 11, Internacional 1 a 13). TX (+10dBm max). Outdoor range (Hasta 150m). Directividad de la antena (Omnidireccional). Seguridad (WPA, WPA2, WPA2-Enterprise).
USB Ports	USB host-device port (2.0 Hi-Speed)
Acelerómetro	Número de ejes (3). Rango (± 8g). Resolución (12bits). Taza de muestreo (800 S/s). Ruido (3.9 mg(rms) a 25ºC)
Requerimientos de potencia	Rango de voltaje de la fuente de alimentación (6 a 16 VCC). Consumo máximo de energía (14 W). Consumo de energía inactivo típico (2.6 W).

Tabla 2. Características del dispositivo NI-myRIO

***Equipo de ensayos de temperatura**:* este dispositivo permite realizar ensayos de control de temperatura con picos de hasta 55 °C, equipado con una niquelina fabricada con alambre de nicromo, dos ventiladores 3010 para perturbaciones, sensores de temperatura tipo RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C.

El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.

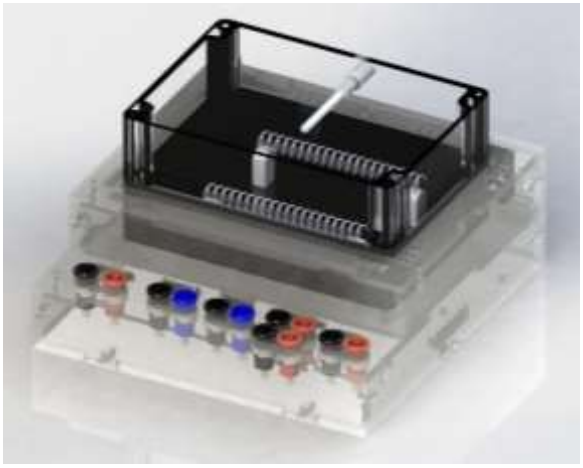


Figura 2. Equipo de ensayo de control de temperatura.

ACTIVIDADES POR DESARROLLAR

1. **Sintonizar controlador PID**

- Abrir el software Matlab r2017b e ingresamos la función de transferencia del sistema, para ello ejecutamos las siguientes líneas de código, la respuesta al escalón se muestra en la Figura 3.

```
% Definicion de la planta
clear all
clc
%Numerador de la funcion de transferencia
num=[0.0002453 4.2012e-7];
%Denominador de la funcion de transferencia
den=[1 0.02612 0.0001594 1.529e-7];
%Definicion de la planta
G=tf(num,den)
%Respuesta al escalon
step(G)
grid on
```

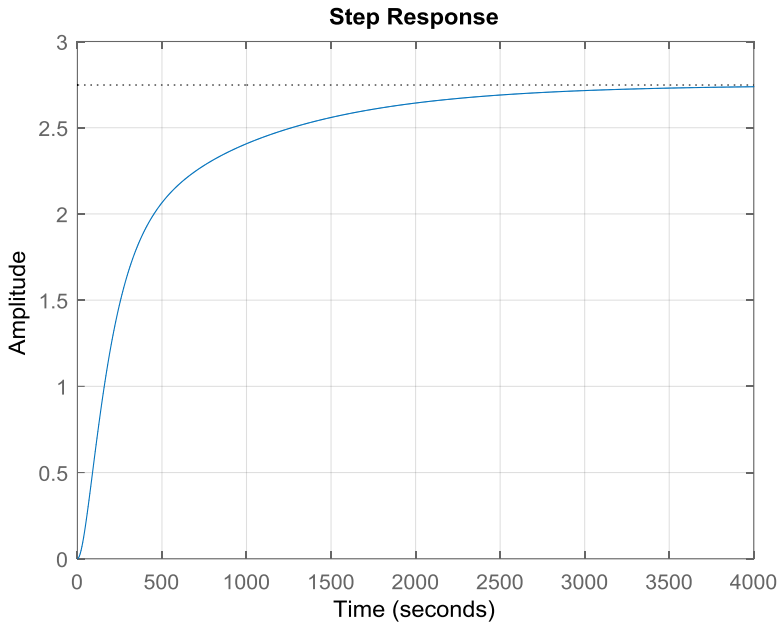


Figura 3. Respuesta al escalón

- Nos ubicamos en la pestaña "Apps" (Aplicaciones), sección "Control system design and analysis" (Diseño y análisis de sistemas de control) y seleccionamos la aplicación PID Tuner (véase Figura 4).

Figura 4. Abrir aplicación PID Tuner.

- En la pestaña "PID TUNER" (Sintonización PID) desplegamos el menú "Plant" (Planta) y seleccionamos "Import a linear plant from Workspace" (Importar una planta lineal desde el espacio de trabajo) (véase Figura 5).

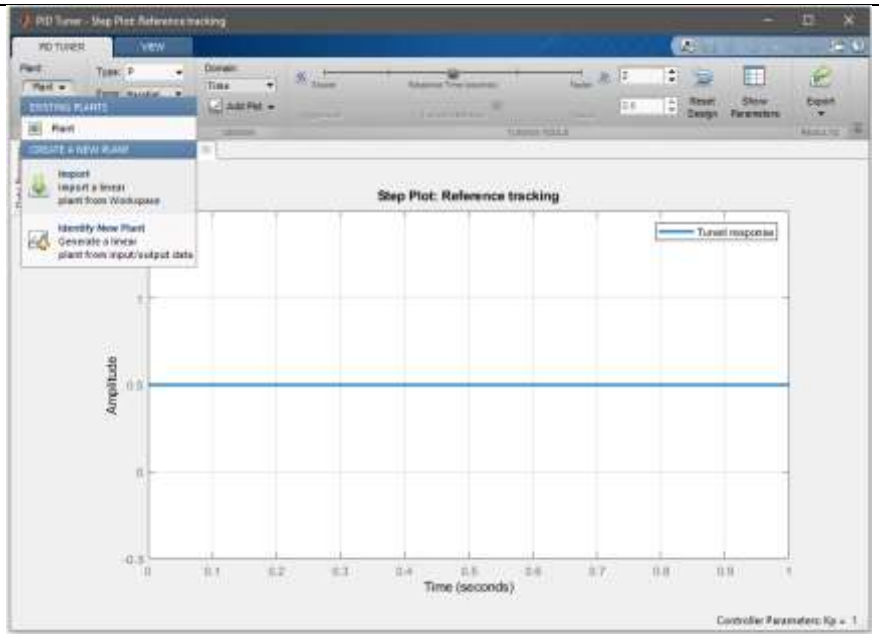


Figura 5. Ventana PID Tuner

- Seleccionamos la función de transferencia G y damos clic en el botón "Import" (Importar) (véase Figura 6).

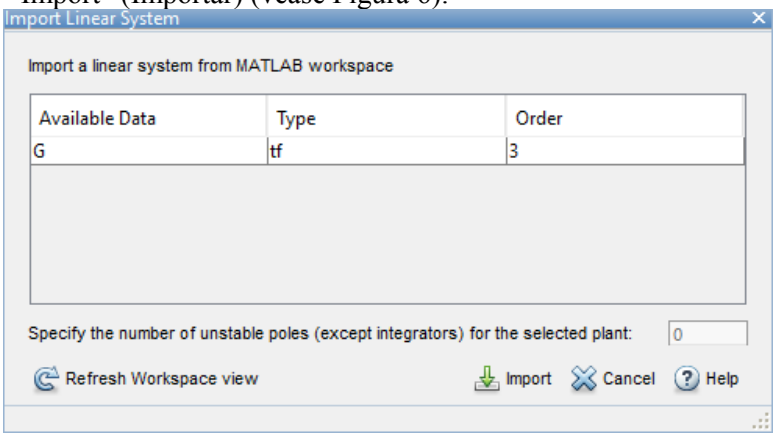


Figura 6. Importar sistema lineal.

- Seleccionamos un controlador PID de forma paralela, cambiamos el tiempo de respuesta y el comportamiento del transitorio hasta obtener la salida deseada (véase Figura 7).

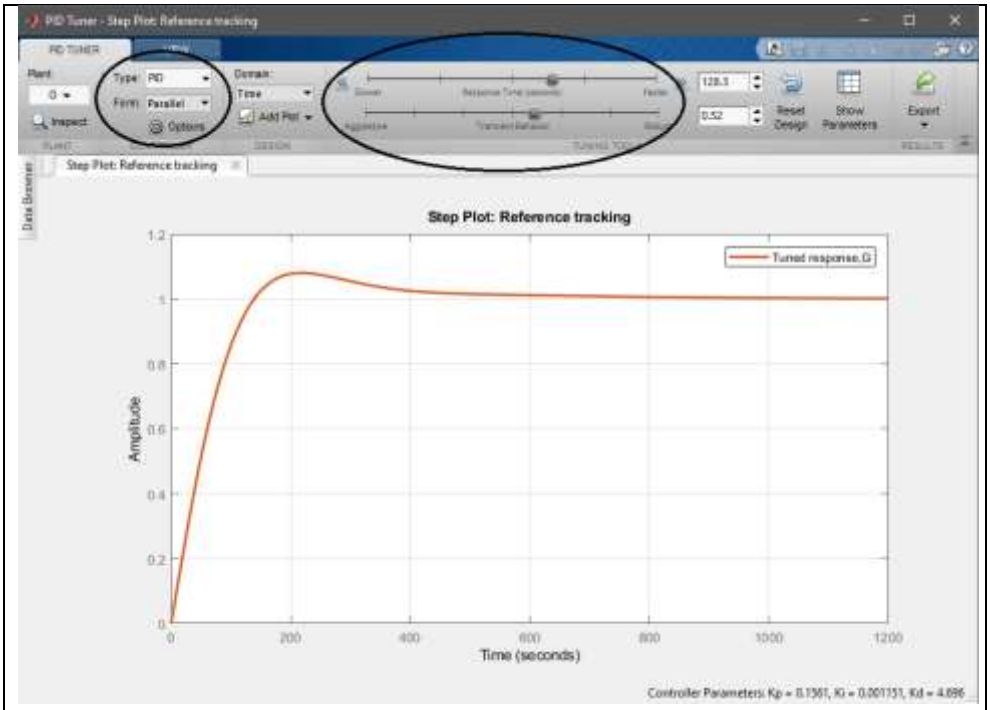


Figura 7. Sintonización del controlador.

- Las constantes del controlador se pueden obtener al dar clic al botón "Show parameters" (Mostar Parámetros) (véase Figura 8).

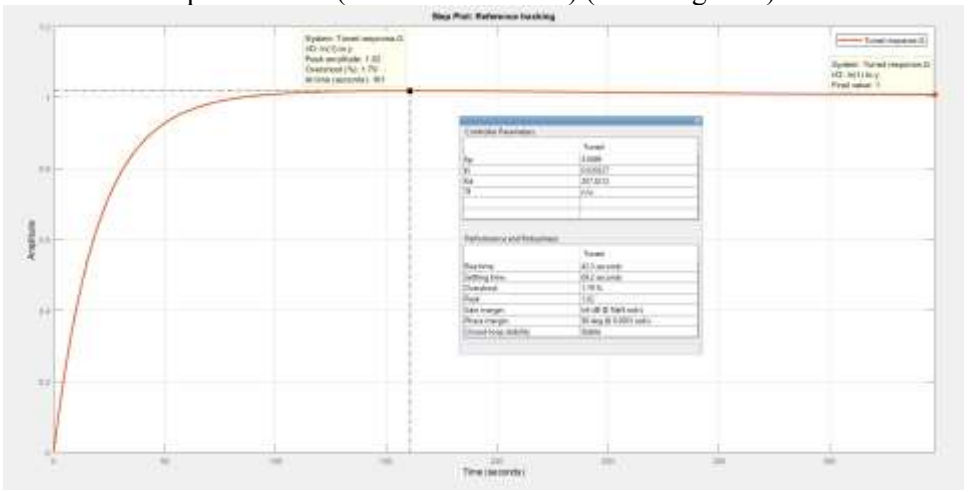


Figura 8. Mostrar parámetros del controlador.

2. Programación del diagrama de bloques en LabVIEW

- Abrimos LabVIEW 2019 myRIO Toolkit (véase Figura 9).

Figura 9. Portada LabVIEW 2019.

- Creamos un nuevo proyecto (véase Figura 10).

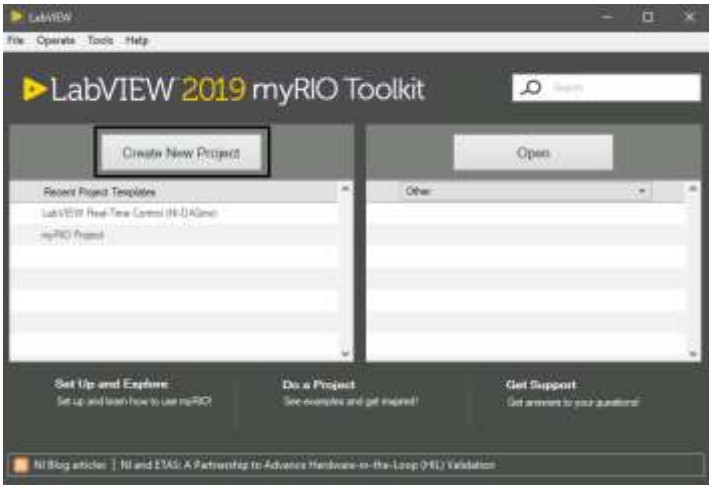


Figura 10. Crear nuevo proyecto

- Dentro de la sección "All" (Todos), seleccionamos "Blank Project" (Proyecto en blanco) y damos clic en "Finish" (Finalizar) (Véase Figura 11).

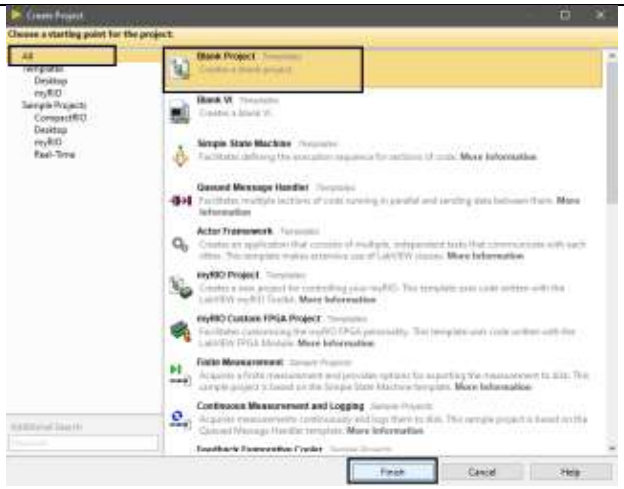


Figura 11. Crear nuevo proyecto en blanco

- En el explorador del proyecto, damos clic derecho en la carpeta raíz, seleccionamos "New" (Nuevo) y damos clic en "Targets and Devices" (Objetivos y dispositivos) (Véase Figura 12).

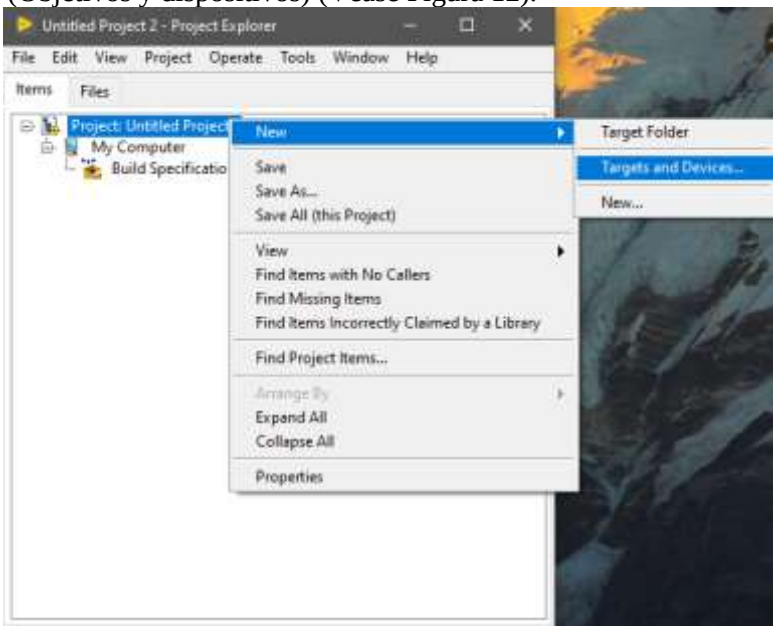


Figura 12. Crear nuevo proyecto en blanco

- Seleccionamos “New Target or Device” (Nuevo objetivo o dispositivo), desplegamos la carpeta myRIO, seleccionamos myRIO-1900 y damos clic en “OK” (véase Figura 13).

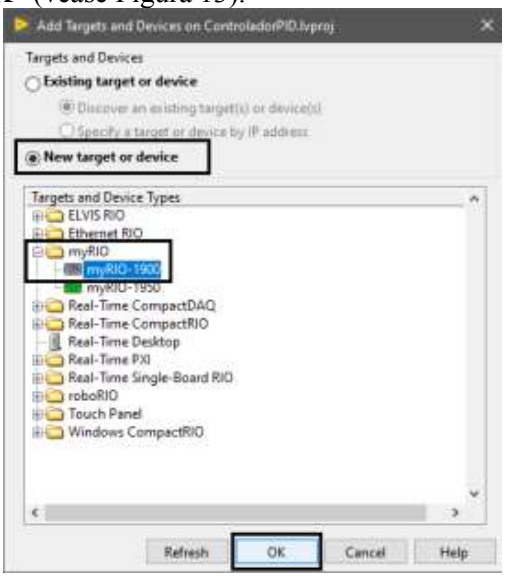


Figura 13. Crear nuevo proyecto en blanco

- Damos clic derecho en “RT myRIO Target” (Dispositivo myRIO), seleccionamos “New” (Nuevo) y damos clic en “VI” (Interfaz Gráfica) (véase Figura 14).

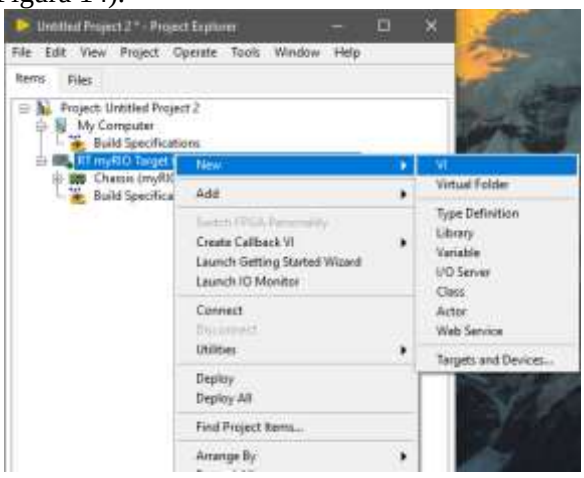


Figura 14. Agregar un VI

- Agregamos una estructura while, una demora y un control para detener el proceso. Al agregar el bloque “Wait” (Esperar), aparecerá una ventana.

Debemos configurar el "Counter units" (Unidades del contador) en milisegundos (véase Figura 15).

Figura 15. Estructura while

- Damos clic derecho y accedemos al buscador de bloques de LabVIEW. (véase Figura 16).

Figura 16. Acceder al buscador de LabVIEW

- Usando el buscador agregamos el bloque "Analog input" (Entrada Analógica), al agregarlo aparecerá un asistente en el que modificaremos el nombre del bloque, así como las entradas físicas que vamos a utilizar, al finalizar pulsamos "OK". (véase Figura 17).

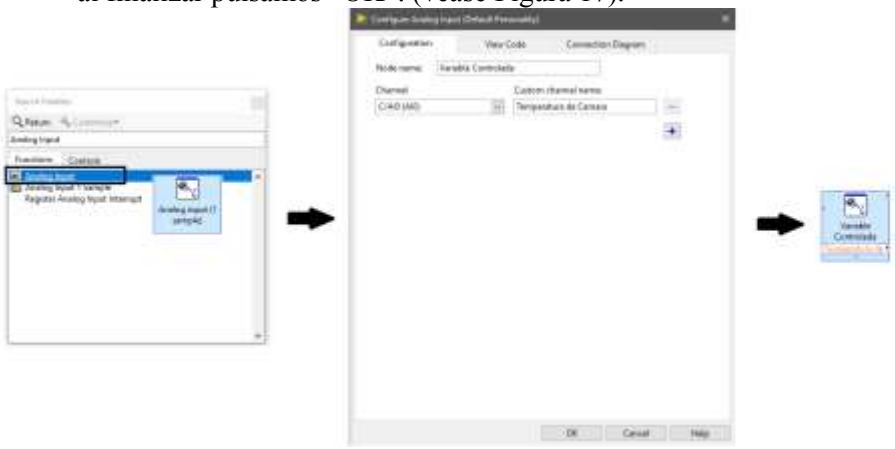


Figura 17. Configuración de entradas analógicas

- Realizamos un procedimiento similar para el bloque "Analog Output" (Salida Analógica) (véase Figura 18).

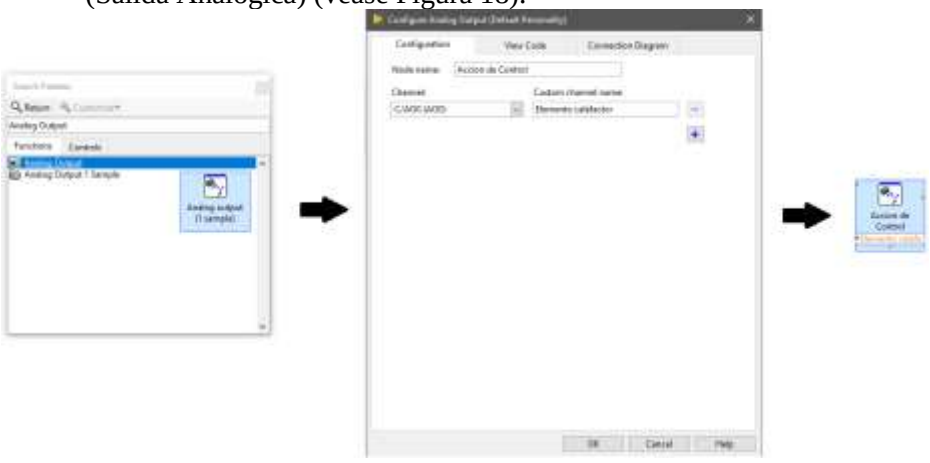


Figura 18. Configuración de salidas analógicas

- Agregamos el bloque PID y damos doble clic para modificar los parámetros: "Setpoint high" (Límite superior de la referencia), "Setpoint low" (Límite inferior de la referencia), "dt(s)" (Tiempo de muestreo) y

"Output high" (Límite superior de la salida), "Output low" (Límite inferior de la salida) (véase Figura 19).

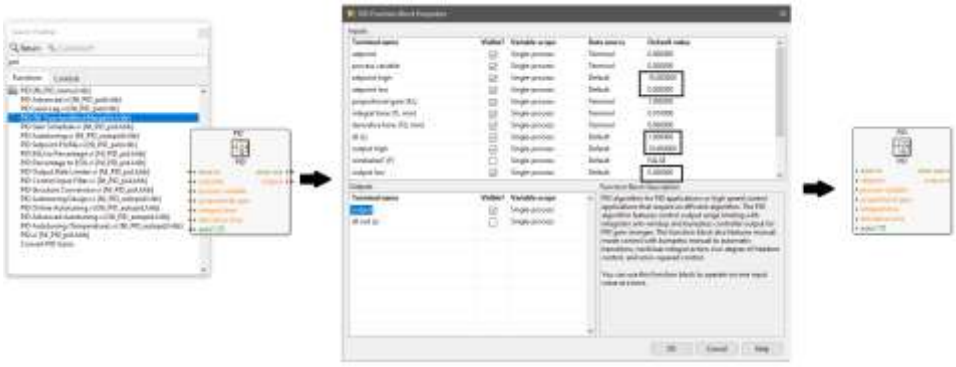


Figura 19. Configuración del bloque PID

- Agregamos el bloque "Write to measurement file" (Escribir a archivo de datos) (véase Figura 20).

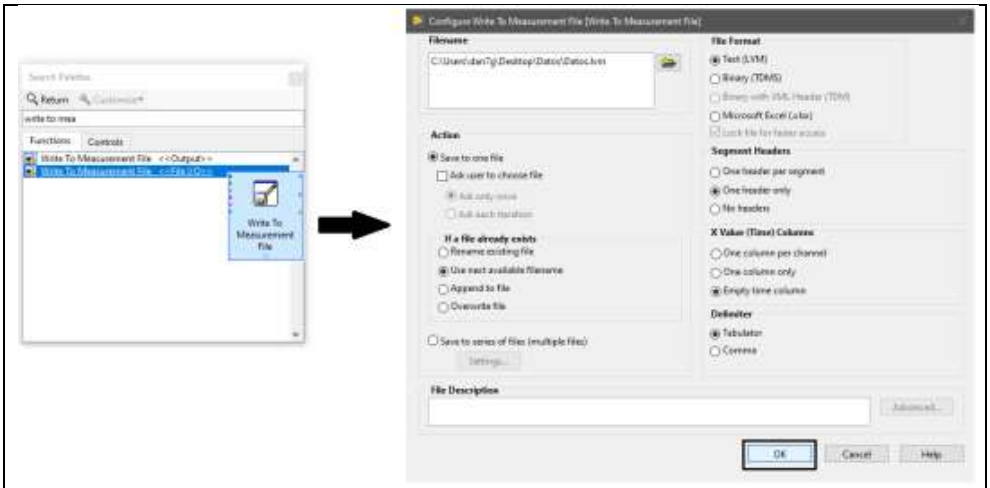


Figura 20. Agregar el bloque "Write to measurement file"

- En la ventana de configuración seleccionaremos la dirección de destino del archivo de datos (véase Figura 21).

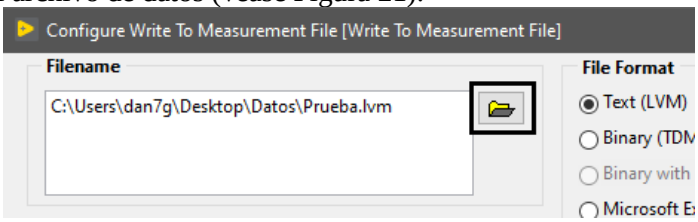


Figura 21. Selección de dirección de archivo de datos

- Se debe asignar un nombre al archivo de datos (véase Figura 22)

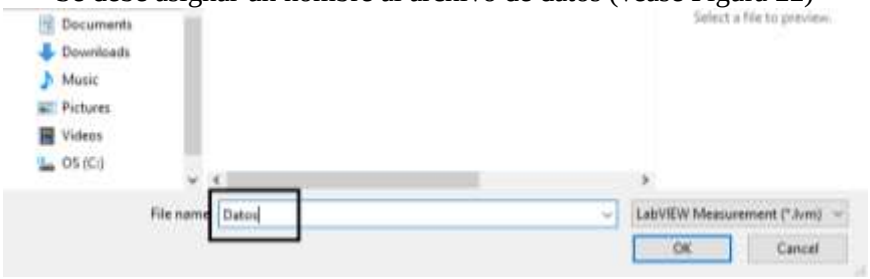


Figura 22. Nombre del archivo de datos

- Configuramos los parámetros restantes (véase Figura 23).

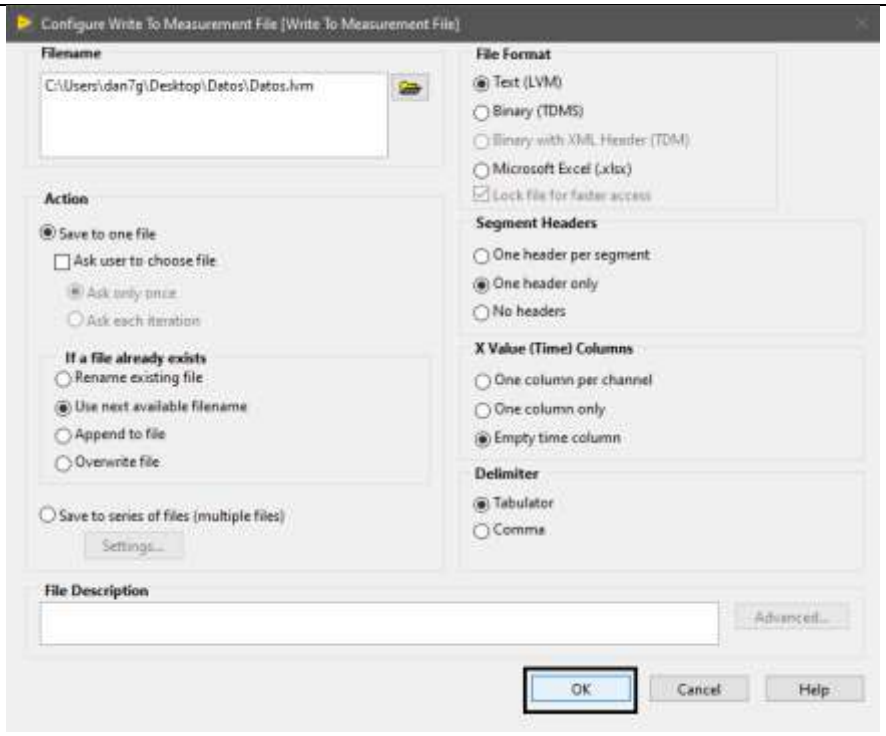


Figura 23. Configuración del bloque "Write to measurement file"

- Añadimos el bloque "Set Dynamic Data Atributes" (Establecer atributos de los datos) y modificamos sus parámetros (véase Figura 24).

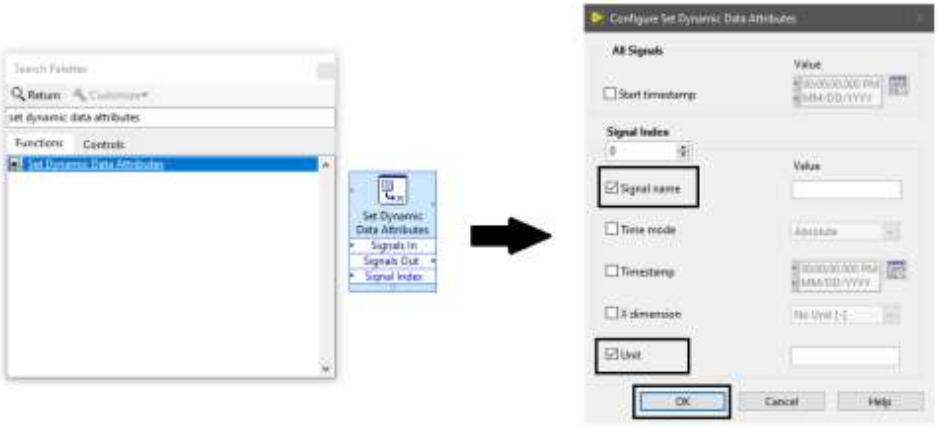


Figura 24. Configuración del bloque "Set Dynamic Data Atributes"

- Junto con los bloques anteriormente configurados se muestran bloques adicionales necesarios para la implementación del controlador, los elementos repetidos han sido copiados (véase Figura 25).

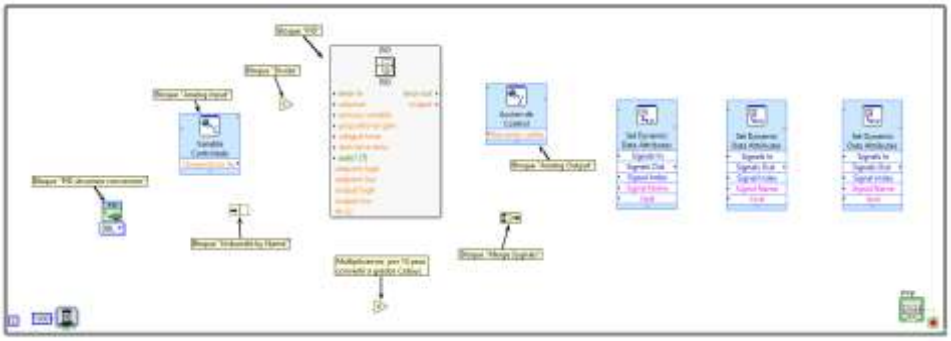


Figura 25. Bloques necesarios para la implementación del controlador

- Conectamos el bloque PID como se muestra en la figura 26, para agregar constantes, controles o indicadores debemos hacer clic derecho sobre una entrada o salida y seleccionar "Create Constant" (Crear constante), "Create Control" (Crear control) o "Create Indicator" (Crear indicador) (véase figura 27)

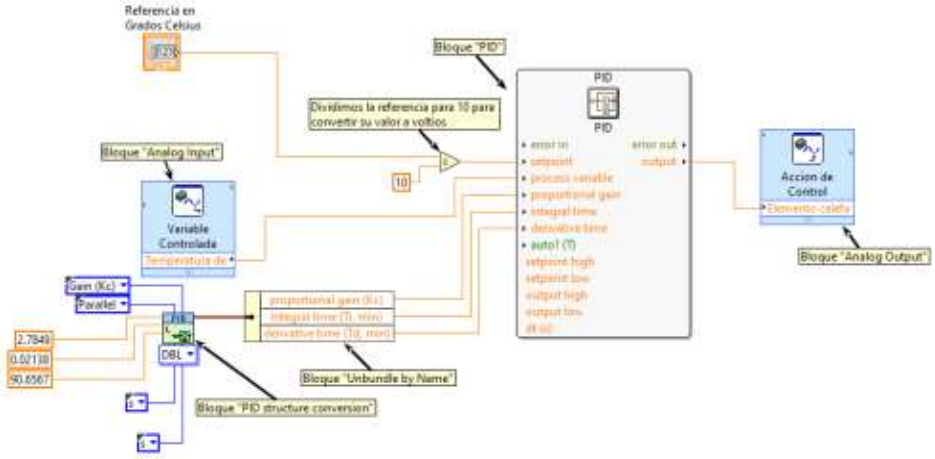


Figura 26. Conexión de bloques del controlador

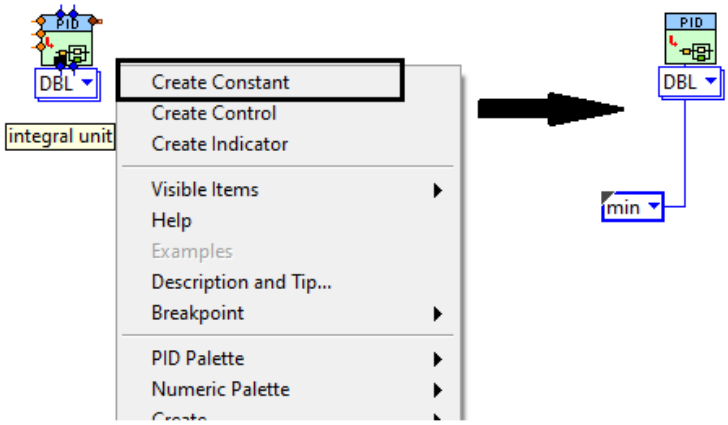


Figura 27. Agregar una constante

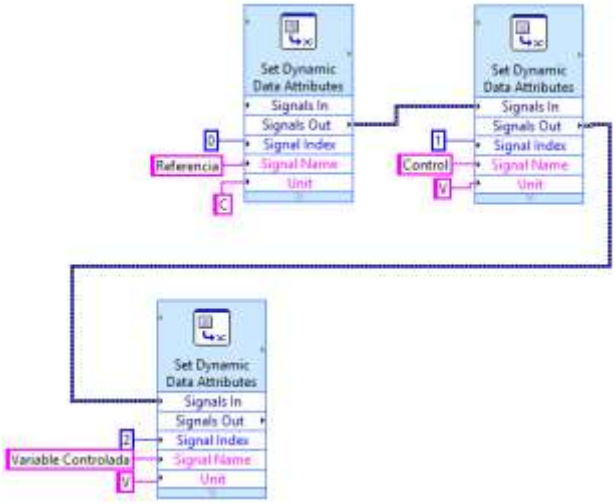


Figura 28. Conexión de bloques "Set Dynamic Data Atributes"

- Conectamos los bloques "Set Dynamic Data Atributes" y añadimos los atributos requeridos en las señales (véase Figura 28)

- Utilizando el bloque "Merge Sygnals" agrupamos las señales de interés y las conectamos el conjunto como se muestra en la figura 29.

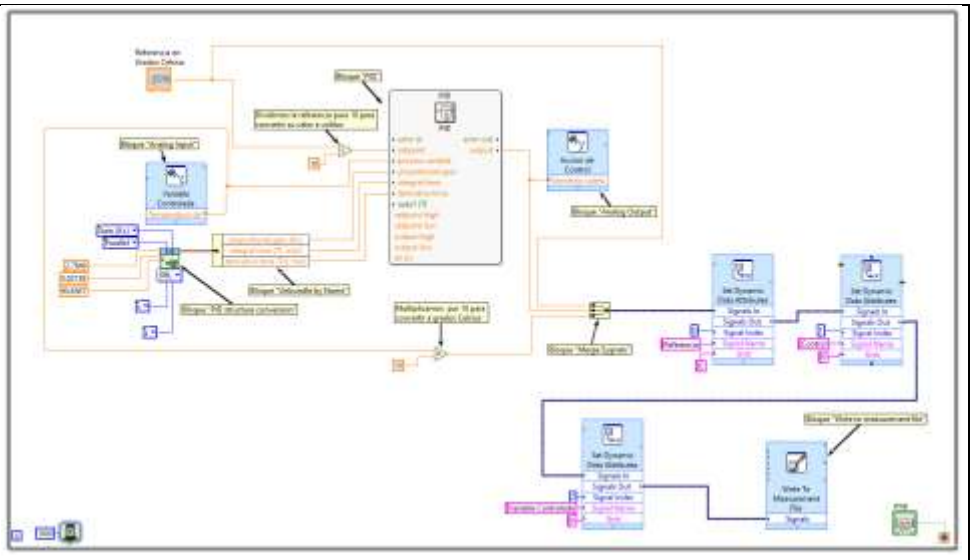


Figura 29. Conexión del diagrama de bloques

3. **Diseño de la interfaz grafica**

- Hacemos clic derecho, nos ubicamos en la carpeta "Graph" y seleccionamos "Waveform Chart" (véase figura 30).

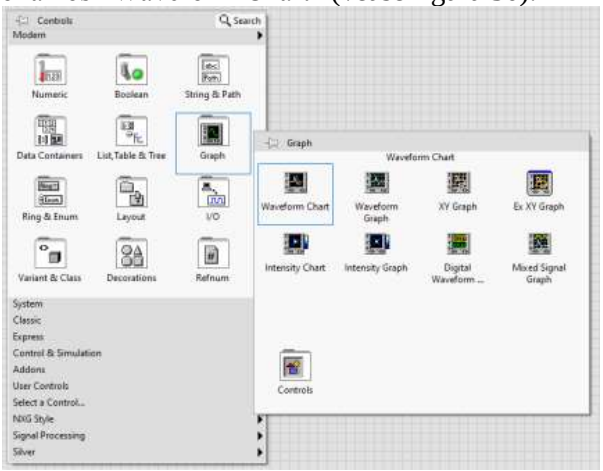


Figura 30. Agregar un "Waveform Chart"

- En el diagrama de bloques conectamos el bloque "Waveform Chart" con las señales de interés (véase figura 31)

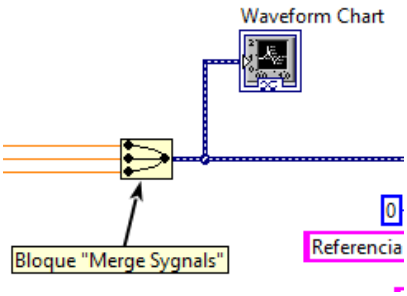


Figura 31. Conexión del bloque "Waveform Chart"

- Acomodamos los elementos de la interfaz gráfica para una mejor visualización (véase figura 32)

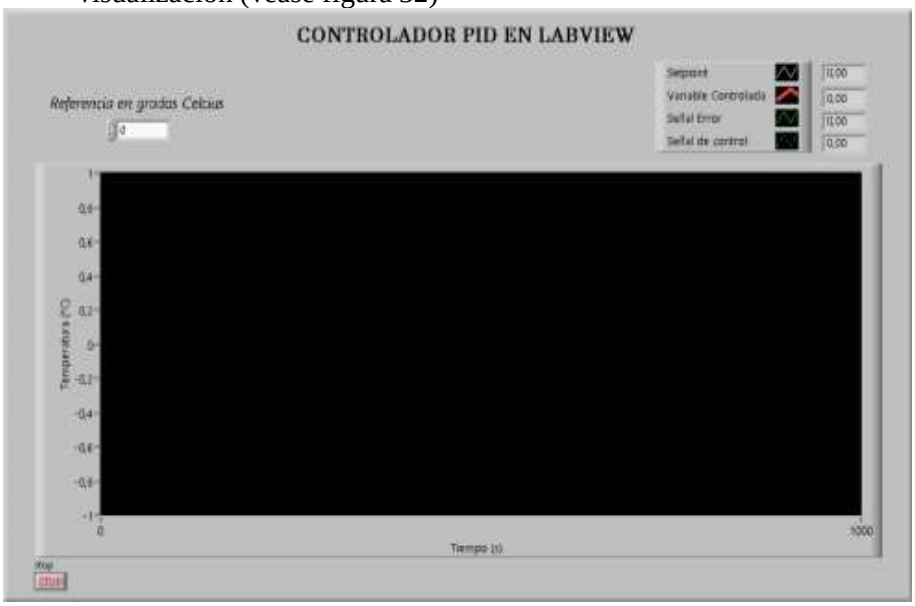


Figura 32. Interfaz grafica

- En el diagrama de bloques, cargamos las ganancias del controlador obtenidas anteriormente (véase figura 33)

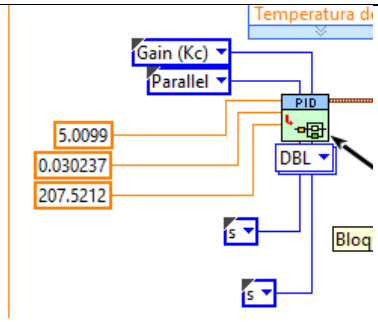


Figura 33. Ganancias del controlador

4. Cableado de dispositivos

- El esquema de conexión se muestra a continuación (véase figura 34).

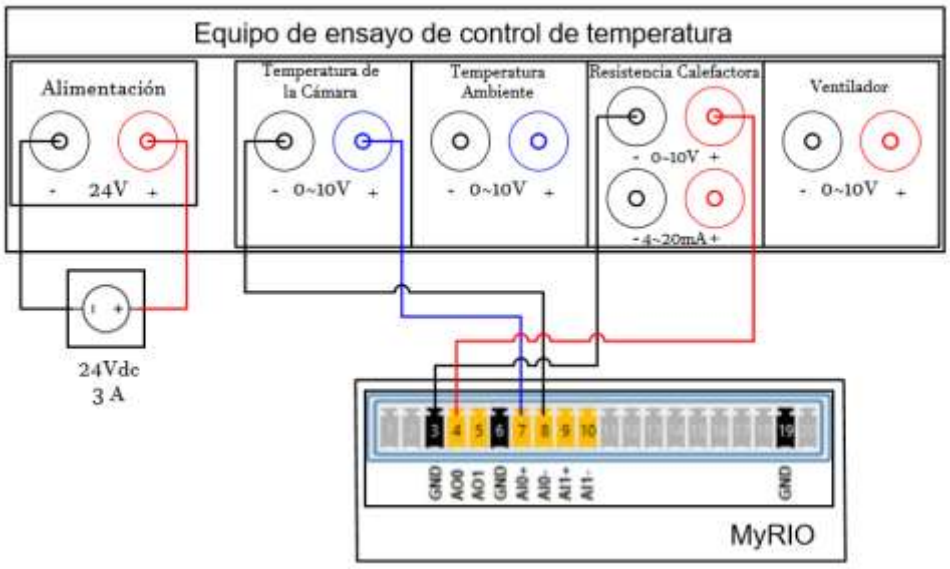


Figura 34. Diagrama de conexión

- Al conectar Ni myRIO aparecerá la ventana mostrada en la figura 35, debemos seleccionar en "Do Nothing". La dirección IP mostrada será utilizada más adelante.

Figura 35. Ventana de inicio de NI myRIO

- En el explorador del proyecto, damos clic derecho en "RT myRIO target 0.0.0.0" y seleccionamos "Properties" (véase figura 36).

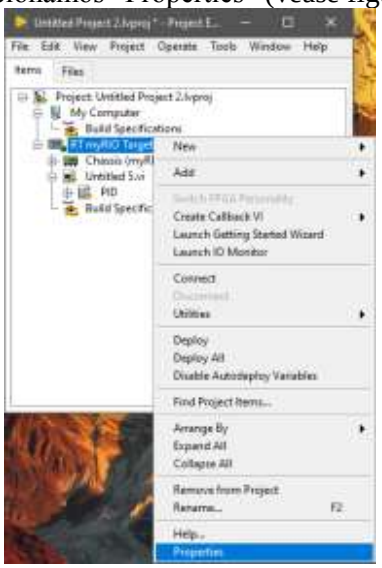


Figura 36. Propiedades de NI myRIO

- En la sección "General" cambiamos la dirección IP con la mostrada anteriormente y damos clic en "OK" (véase figura 37).

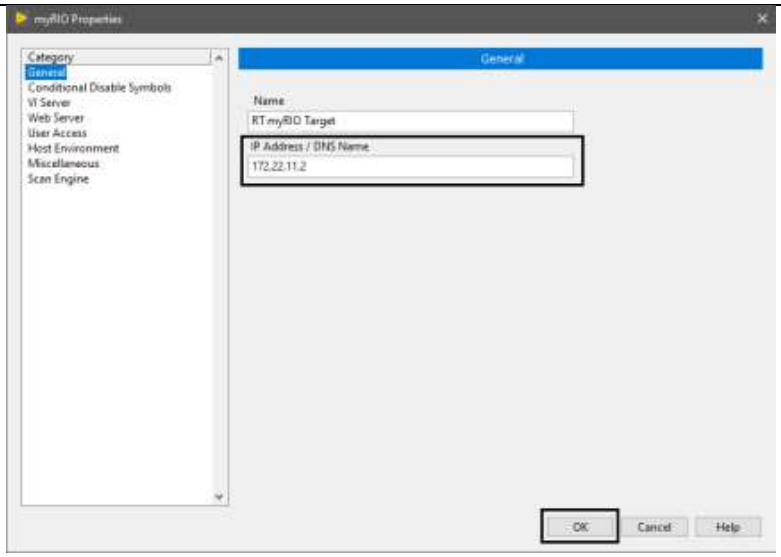


Figura 37. Asignar dirección IP

- Ejecutamos el código (véase figura 38).

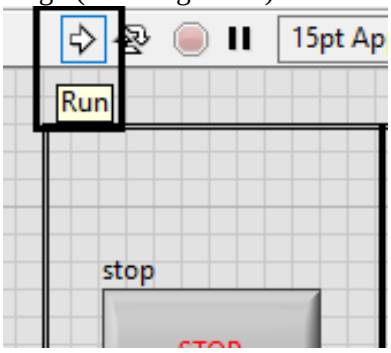


Figura 38. Ejecutar código

- Ingresamos una referencia en el rango de 30-70 grados Celsius y observamos el comportamiento del controlador.
- Para acceder al archivo de datos debemos utilizar un navegador de internet, donde ingresaremos la dirección IP de la NI myRIO seguido de "/files" (véase figura 39).

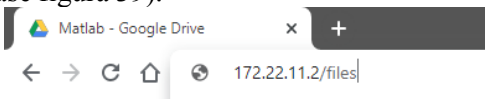


Figura 39. Dirección IP myRIO

- En el campo de usuario ingresamos “admin” y dejamos el campo de contraseña en blanco (véase figura 40).

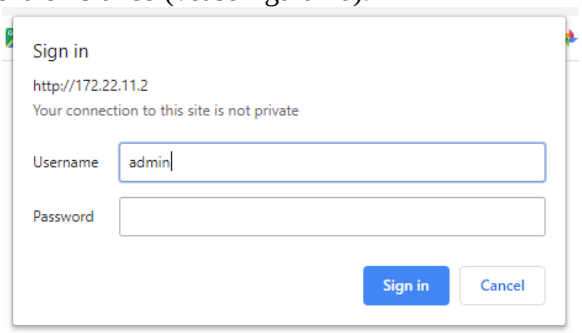


Figura 40. Credenciales NI myRIO

- Accedemos al directorio seleccionado anteriormente y descargamos el archivo. En caso de que simplemente se muestre en el navegador debemos dar clic derecho sobre los datos y guardarlo como texto.

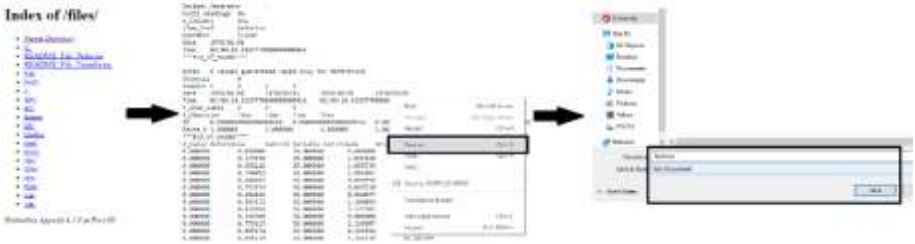


Figura 41. Descargar datos NI myRIO

5. Grafica de resultados en Matlab

- Abrimos Matlab r2017b y en hacemos clic en el botón “Import Data” (Importar datos). (véase Figura 42)

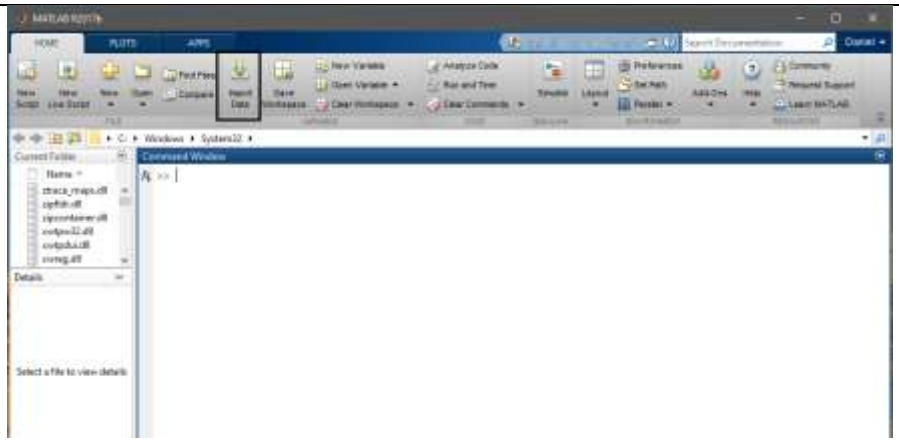


Figura 42. Importar datos

- En la ventana emergente nos ubicamos en la pestaña “Import” (Importar) y seleccionamos el delimitador seleccionado en LabVIEW en este caso “Tab” (Tabulador).
- Seleccionamos el rango de la matriz que contiene los datos junto con la fila en la que se encuentran los nombres de las variables
- Seleccionamos el tipo de salida como “Column vectors” (Vectores Columna) y finalmente damos clic en “Import Selection” (Importar selección) (véase figura 43).

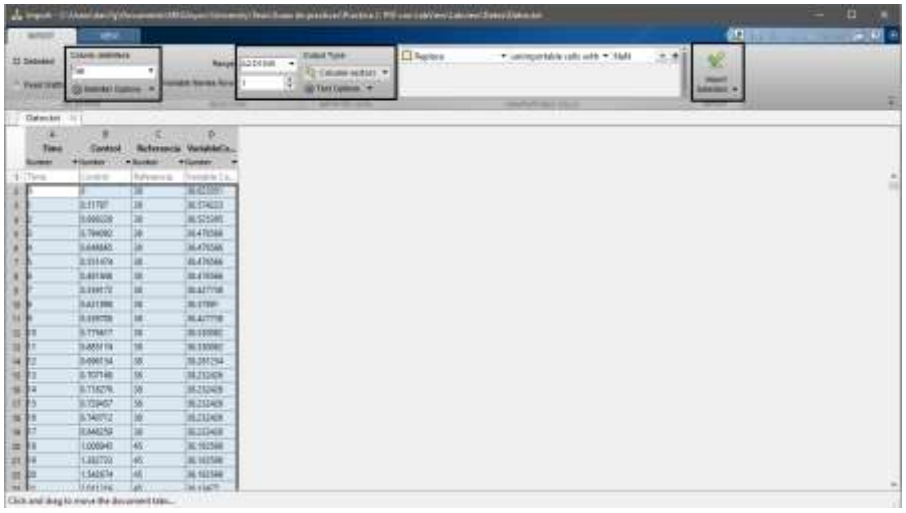


Figura 43. Configurar parámetros

- Las variables aparecerán en el workspace

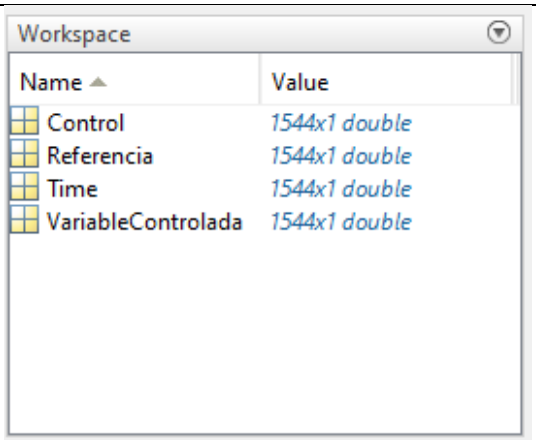


Figura 44. Variables en el workspace

- Con los datos importados podemos graficar el comportamiento del sistema con controlador, en la gráfica 45.

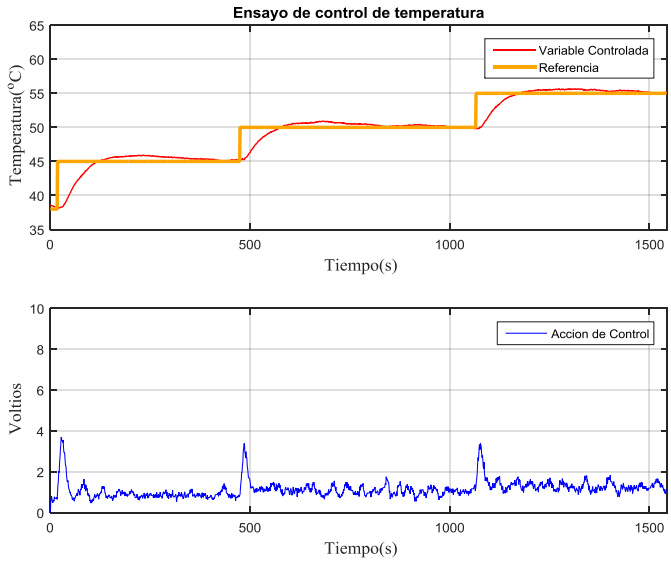


Figura 45. Respuesta del sistema

RESULTADO(S) OBTENIDO(S):

APÉNDICES

CONCLUSIONES:
RECOMENDACIONES:

APÉNDICES

- **GUÍA DE PRÁCTICA 4: IMPLEMENTACIÓN DE UN CONTROLADOR PID CON AUTO SINTONÍA UTILIZANDO UN PLC S7-1200 CON MONITOREO EN TIEMPO REAL EN LABVIEW USANDO EL PROTOCOLO OPC**

FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: INGENIERÍA ELECTRÓNICA	**ASIGNATURA**: TEORÍA DE CONTROL

NRO. PRÁCTICA :	4	**TÍTULO PRÁCTICA:** **IMPLEMENTACIÓN DE UN CONTROLADOR PID CON AUTO SINTONÍA UTILIZANDO UN PLC S7-1200 CON MONITOREO EN TIEMPO REAL EN LABVIEW USANDO EL PROTOCOLO OPC**

OBJETIVO

1. Proporcionar las herramientas necesarias para que el estudiante pueda crear un proyecto en el software TIA Portal V13.
2. Conocer las características generales del equipo de ensayo de control de temperatura y PLC S7-1200.
3. Conocer las herramientas del software LabVIEW y TIA Portal V13.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

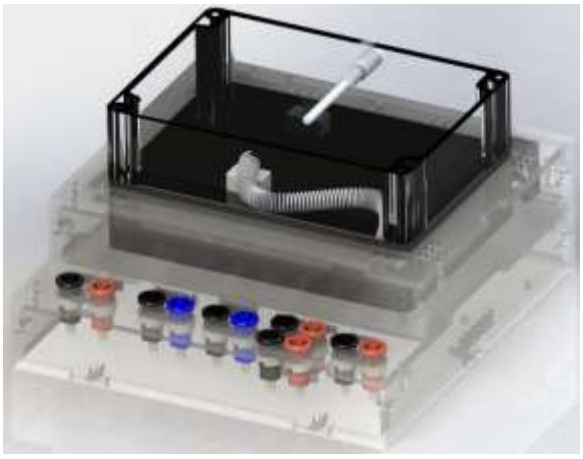
a) Protocolo de comunicación OPC

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Computadora S.O. Windows 10, 64 bits	1	-	-
Software Labview 2019.	1	-	-
Software TIA Portal V13	1	-	-
Equipo de ensayo de control de temperatura	1	CTILab	-
Cables bananas	Varios	-	-

Tabla 1. Equipos, instrumentos y software

3. Exposición

	Equipo de ensayos de control temperatura: este dispositivo permite realizar ensayos de control de temperatura con picos de hasta 80 °C, equipado con una niquelina fabricada con alambre de nicromo, dos ventiladores 3010 para perturbaciones, sensores de temperatura tipo RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C. El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.  *Figura 1. Equipo de ensayos de control de temperatura.* **SIMATIC S7-1200:** estos dispositivos ofrecen la automatización de varias aplicaciones industriales gracias a su flexibilidad. Tiene los más altos requisitos de la industria para aplicaciones que requieren de alta tecnología. Cuenta con un diseño compacto para el ahorro de espacio puesto que contiene entradas y salidas integradas, además, permite incrementar placas de montaje adicionales como entradas y salidas, la comunicación RS485 y la placa de batería al controlador. El controlador SIMATIC S7-1200 es capaz de manejar tareas estándar y relacionadas con la seguridad. Es posible programar el CPU para la comunicación con equipos como SIMATIC HMI. El

	equipo SIMATIC S7-1200 admite conmutación serial, PROFIBUS, IO-Link, AS-Interface, entre otros [20]. *Figura 2. Equipo SIEMENS S7-1200.*
ACTIVIDADES POR DESARROLLAR (Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)	
1. Descargar de software Descargar el software KEPServerEX Free Trail del siguiente link https://www.kepware.com/en-us/kepserverex-6-7-release/	
2. Dirección IP del computador • En el buscador de Windows escribir “Change adapter options” (Cambiar opciones del adaptador) • En la ventana “Network Connections” clic derecho en el puerto de red conectado al PLC, seleccionar “Propiedades”	

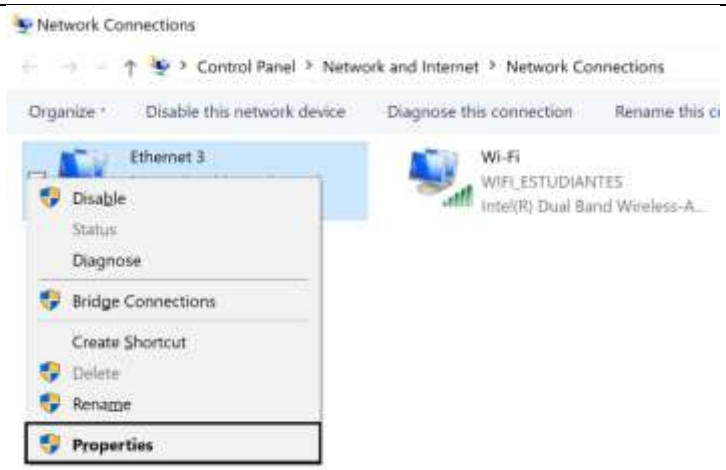


Figura 3. Propiedades del puerto ethernet

- Doble clic en "Internet Protocol Version 4 (TCP/IPv4)"

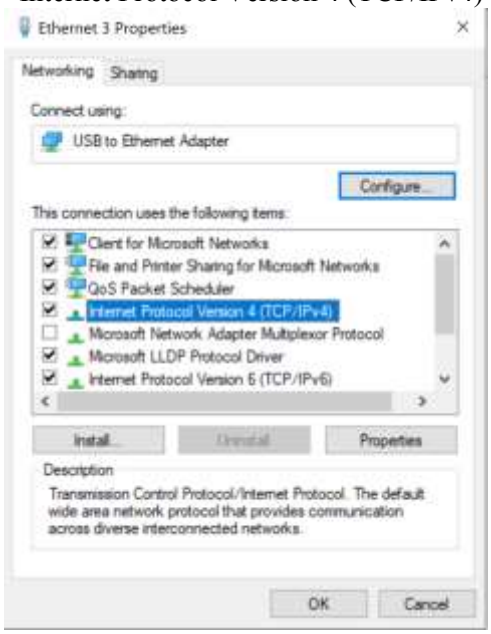


Figura 4. Propiedades del puerto

- Colocar la dirección IP "192.168.0.2", la máscara de red "255.255.255.0" y la puerta de enlace "192.168.0.1" (véase Figura 5).
- Clic en "OK"

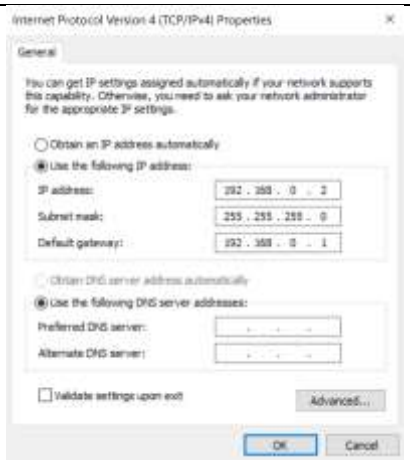


Figura 5. Configuración de dirección IP

3. Creación del proyecto en el software TIA Portal v13

- Abrir el acceso directo del software TIA PORTAL V13.
- Al lado derecho de la ventana se encuentran las opciones para colocar el nombre del proyecto y su ubicación. Una vez colocado los datos del proyecto clic en "Create" (Crear) (véase Figura 6).

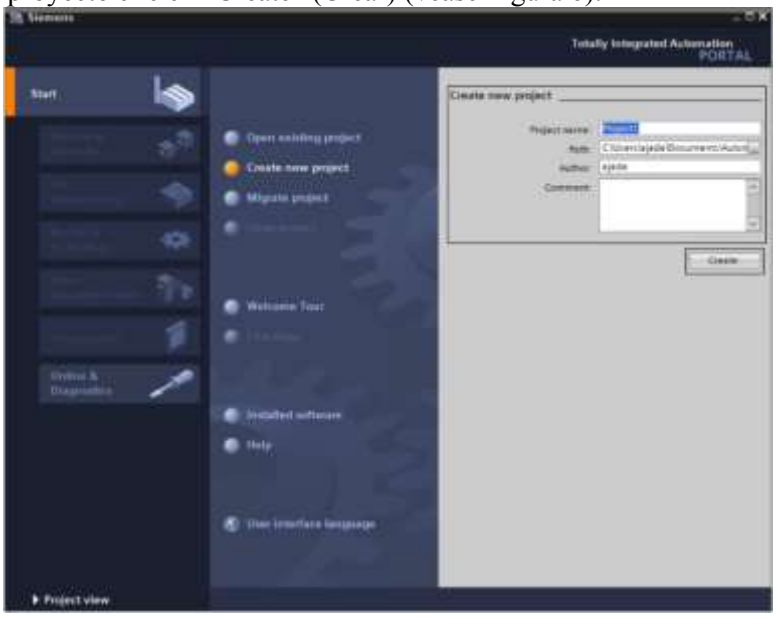


Figura 6. Datos del proyecto.

- En la siguiente ventana dar clic en "PLC programming" (Programación del PLC) (véase Figura 7).

Figura 7. Pantalla de Creación de Proyecto.

- En la siguiente pantalla dar clic en "Click here to add a new device…" (Clic aquí para añadir un nuevo dispositivo) (véase Figura 8).

Figura 8. Pantalla para agregar un nuevo dispositivo.

- En la ventana "Add new device" (Añadir nuevo dispositivo), desplegar la opción "Controllers" (Controladores), escoger la opción "SIMATIC S7-1200". Desplegar la opción "CPU" y escoger "CPU 1214 C AC/DC/Rly", clic en "6ES7 214-1BE30-0XB0". En la parte derecha de la ventana se observará información sobre el CPU del PLC, desplegar el menú de "Version" y escoger "V1.0" (véase Figura 9). Clic en "OK"

Figura 9. Ventana "Add new device"

- Una vez agregado el dispositivo dar doble clic en "Main"

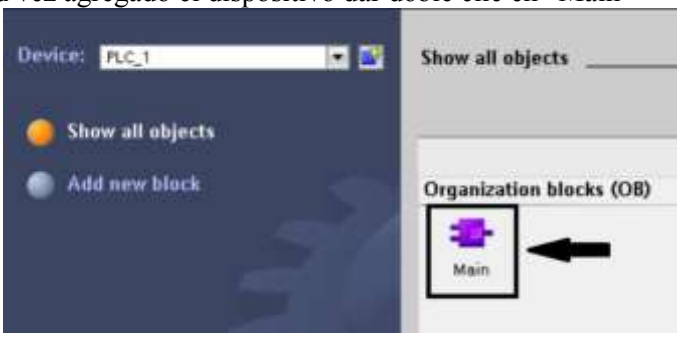


Figura 10. Bloque Main

APÉNDICES

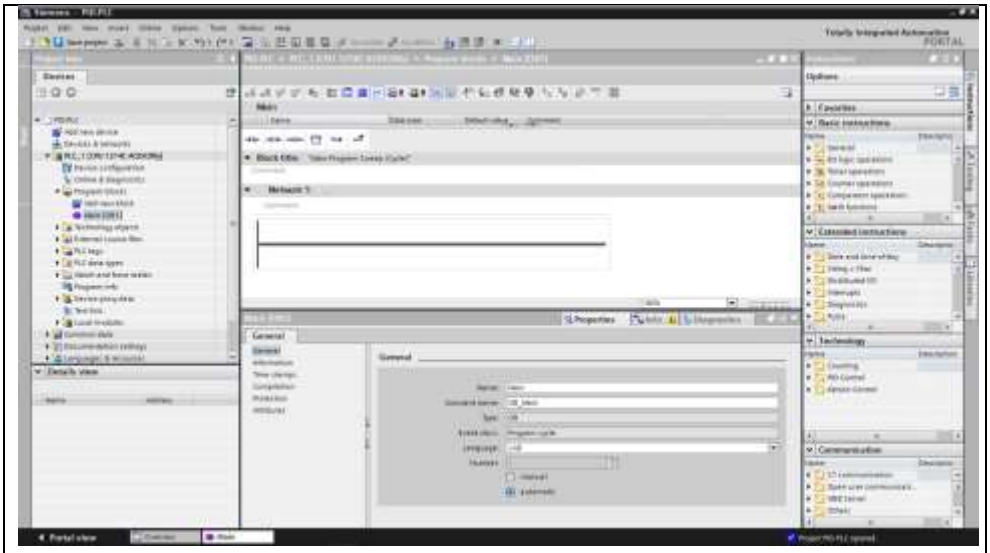


Figura 11. Ventana del nuevo proyecto creado

- En el Árbol del Proyecto dar clic en "Device configuration" (Configuración del dispositivo) (véase Figura 12)

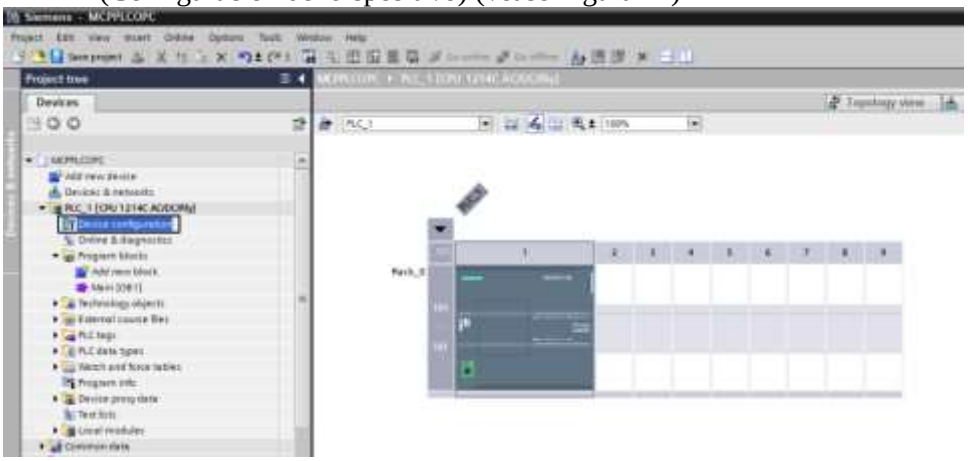


Figura 12. Device configuration

- Ubicarse al lado derecho de la ventana y escoger "Hardware catalog" (Catálogo de hardware), seleccionar "Signal boards", escoger "AQ", seguido de "AQ 1x12BIT", doble clic en "6ES7 232-4HA30-0XB0". Se habilitará las salidas analógicas del PLC.

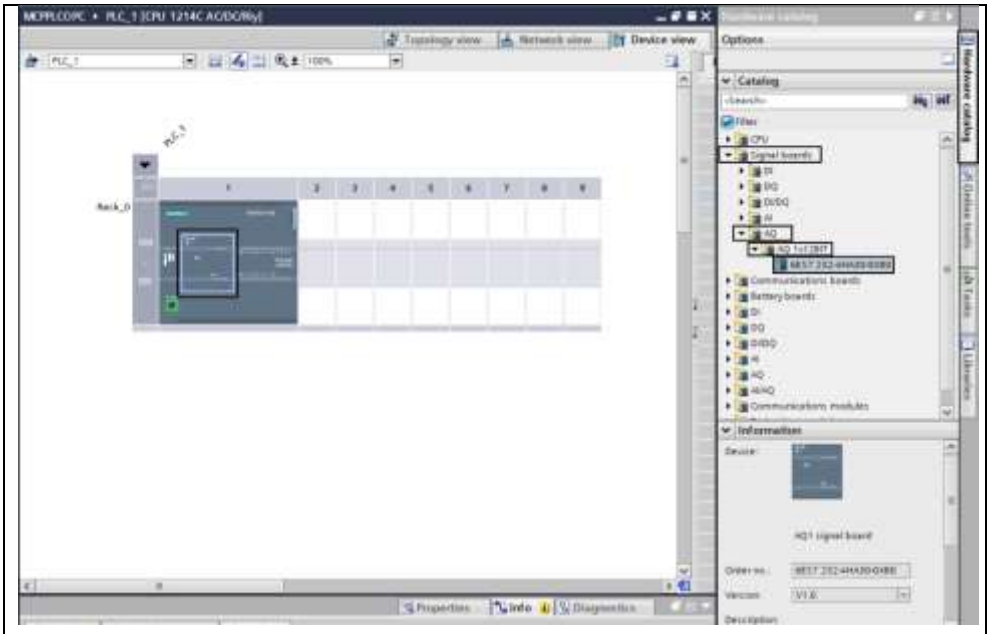


Figura 13. Añadiendo bloque de salidas analógicas

- Para la programación del PLC dar doble clic en “Add new block” (Añadir nuevo bloque) (véase Figura 14).

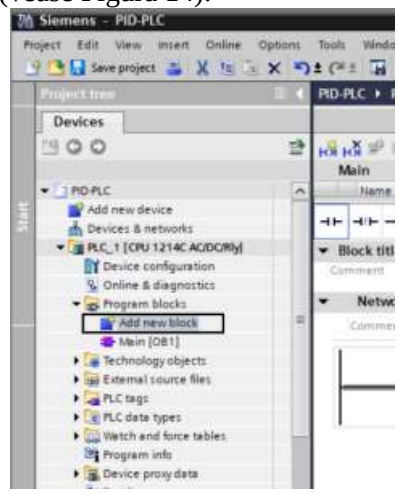


Figura 14. Agregar “new block”

- En la nueva ventana emergente escoger la opción “Organization block” (Bloque de Organización), seleccionar “Cyclic interrupt” (Interrupción Cíclica), clic en “OK” (véase Figura 15).

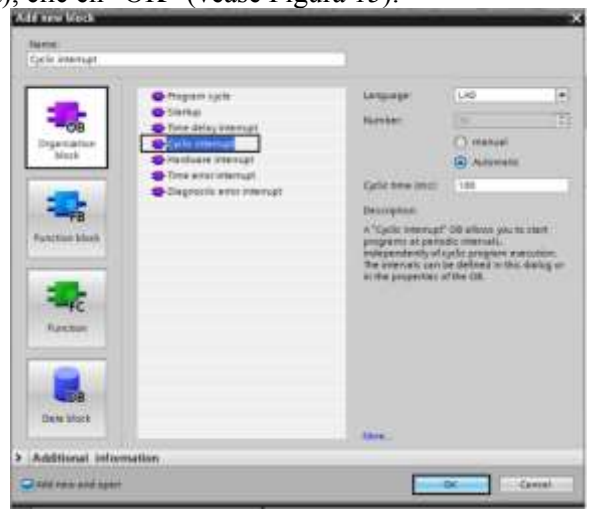


Figura 15. Ventana “Add new block”

- Doble clic en “Add new object” (Añadir nuevo objeto), ubicado en la pestaña “Technology object”, ubicado en el árbol del proyecto (véase Figura 16).

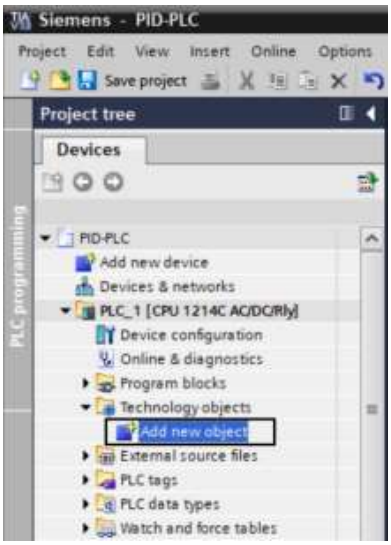


Figura 16. Añadiendo nuevo objeto

- En la ventana emergente “Add new object” (Añadir nuevo objeto) seleccionar en “PID”, escoger “PID_Compact” y desplegar el menú “Version” y escoger “V1.0”, clic en “OK” (véase Figura 17).

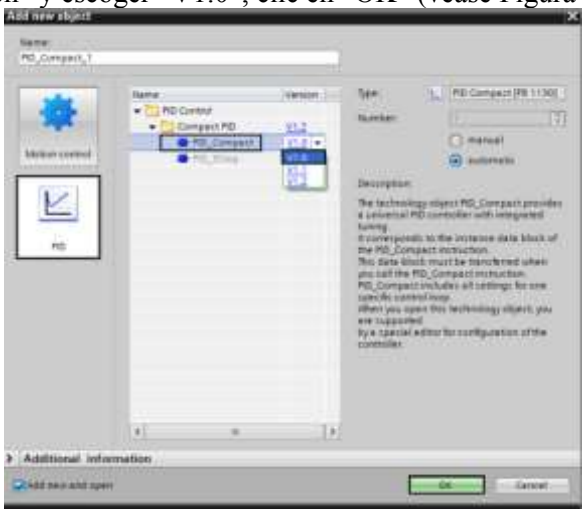


Figura 17. Ventana “Add new object”

- Al agregar el nuevo objeto aparece las opciones del controlador, en la pestaña “Functional view” (Vista funcional) dar clic en “Controller type” (Tipo de controlador), desplegar el menú de opciones y escoger la opción “Temperature” (Temperatura) (véase Figura 18).

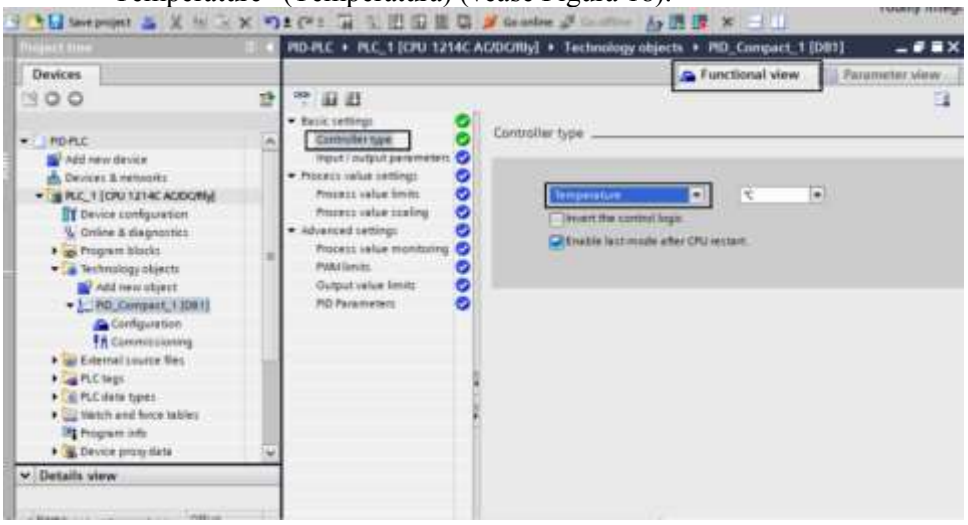


Figura 18. Pestaña “Functional view”

- Se regresa al bloque de programa “Cyclic interrupt”, en la parte derecha de la ventana dar clic en “Technology”, seleccionar “PID Control”, escoger “Compact PID” y arrastrar el bloque “PID_Compact” (véase Figura 19).

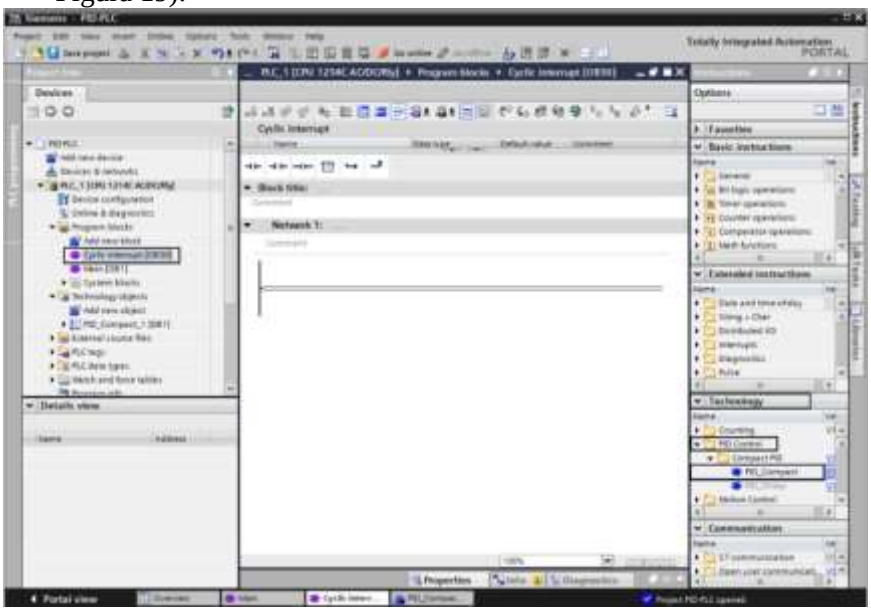


Figura 19. Agregando el bloque PID_Compact

- Al arrastrar el bloque “PID_Compact”, se abre la ventana “Call options” (Opciones de llamada), desplegar el menú de “Name” y escoger “PID_Compact_1”, clic en “OK” (véase Figura 20).

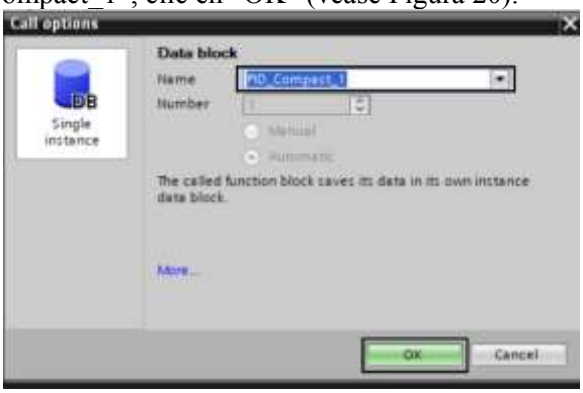


Figura 20. Nombre del bloque “PID_Compact”

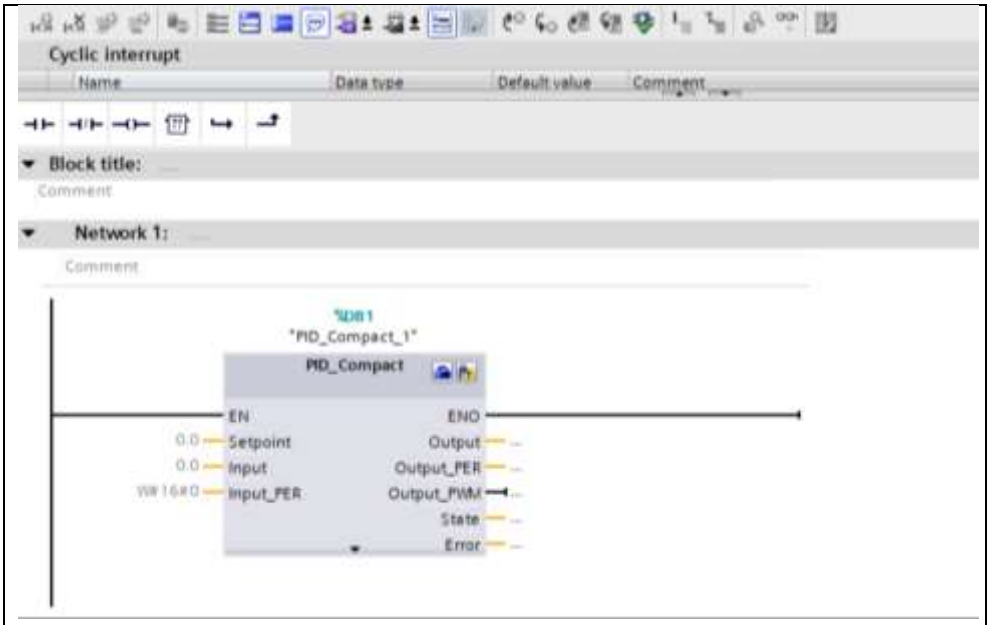


Figura 21. Objeto PID_Compact agregado

- Para añadir variables seleccionar “PLC tags” y doble clic en “Show all tags” (véase Figura 22).

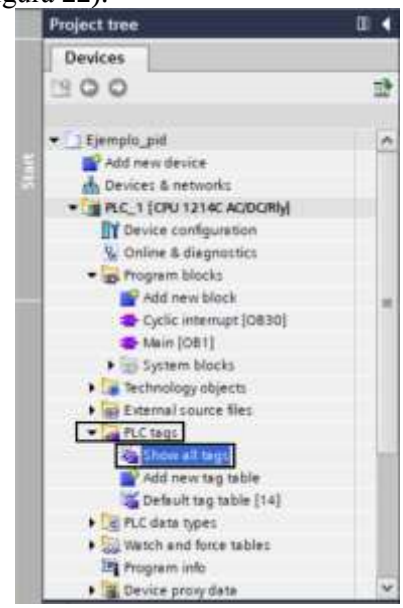


Figura 22. Añadiendo variables

- En la pestaña “Tags” (etiquetas), clic en “Add new” (Añadir nueva), colocar el nombre de la variable “Entrada_Tcamara”, clic en “Data type” (Tipo de dato) y seleccionar “Word”, clic en “Address” (Dirección) y colocar “%IW66” (véase Figura 23).

Figura 23. Añadiendo variables

- Colocar las todas las variables de la misma forma (véase Figura 24).

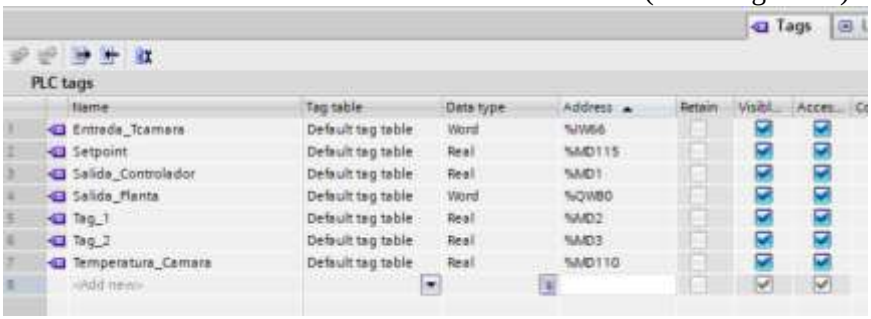


Figura 24. Variables añadidas

- Se programa el controlador (véase Figura 25)

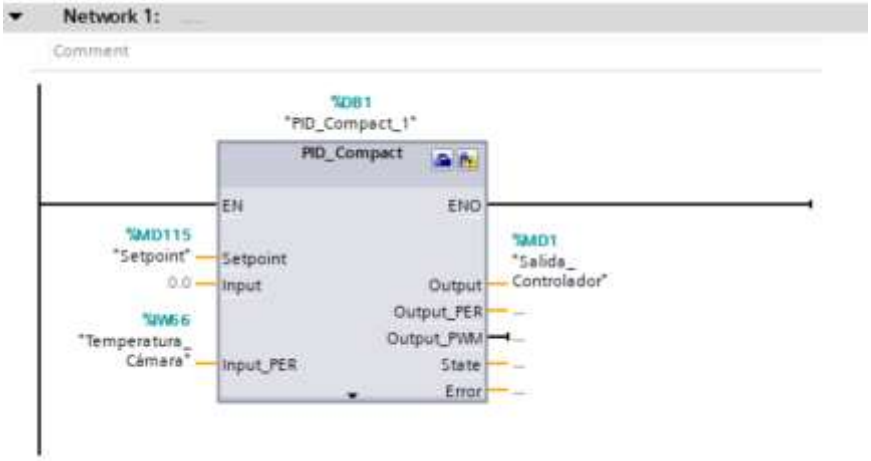


Figura 25. Entradas y salidas del controlador PID

La variable virtual Salida_Controlador (%MD1) se adecua para obtener la señal por la salida "%QW80" del PLC, esta también se utiliza para observar la señal en el software LabVIEW.

- Clic en "Conversion operations" ubicado al lado derecho del proyecto (véase Figura 26).

Figura 26. Bloques "NORM_X" y "SCALE_X"

- Arrastrar el bloque "NOMR_X" y el bloque "SCALE_X" al "Network 2" del bloque "Cyclic Interrupt" y colocar las variables correspondientes (véase Figura 27).

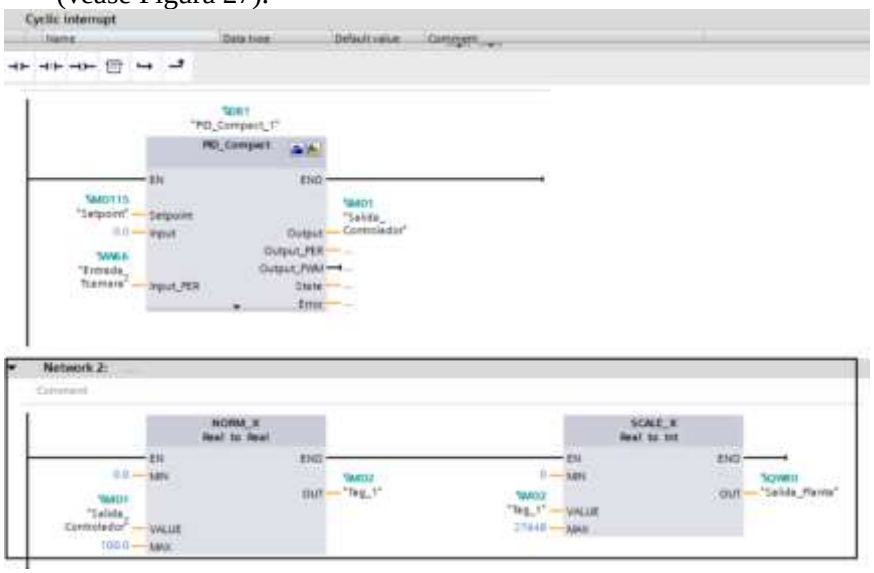


Figura 27. Programación del bloque "Cyclic Interrupt"

- Se adecua la señal de Entrada_Tcamara para el monitoreo con LabVIEW, para ello se coloca los bloques “NORM_X” y “SCALE_X” en el “Network 3” del bloque “Cyclic Interrupt” y se coloca las respectivas variables (véase Figura 28).

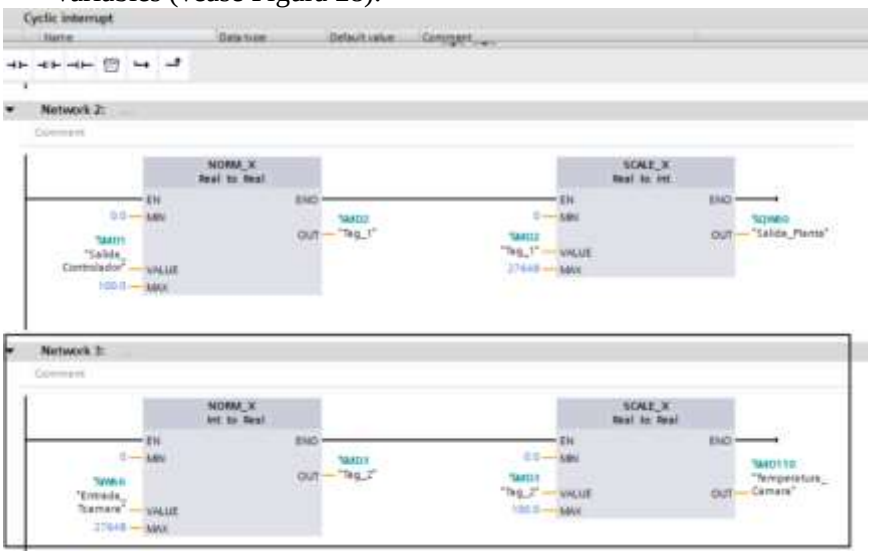


Figura 28. Programación de la Entrada de temperatura de la cámara para monitoreo con LabVIEW

- Se carga el programa en el PLC S7-1200 (véase Figura 29)

Figura 29. Cargando el programa en el PLC S7-1200

- En “Type of the PG/PC interface” seleccionar “PN/IE”, en “PG/PC interface” seleccionar el adaptador conectado al PLC en este caso “USB to Ethernet Adapter”, seleccionar “Start search”, esperar hasta que los dispositivos disponibles se muestren. Seleccionar “192.168.0.1”, clic en “Load” (véase Figura 30)

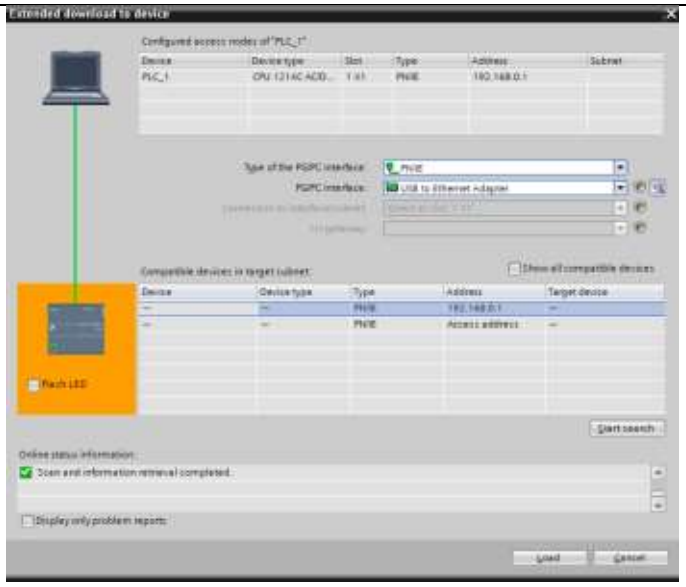


Figura 30. Cargando programa en el PLC

- En la siguiente ventana clic en "Load"

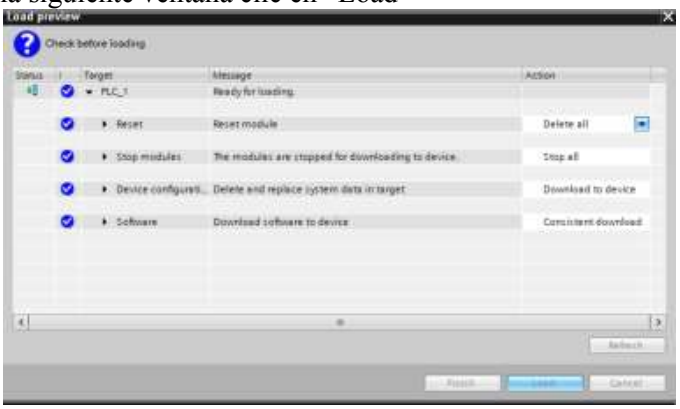


Figura 31. Cargando programa en el PLC

- En la siguiente ventada clic en "Finish"

Figura 32. Cargando programa en el PLC

4. Configuración del software KEPServerEX

- Abrir el acceso directo del software KEPServerEX6
- Clic derecho en "Connectivity" (Conectividad), clic en "New Channel" (Nuevo Canal)

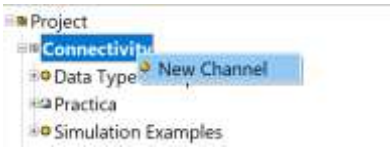


Figura 33. Nuevo Canal

- Seleccionar "Siemens TCP/IP Ethernet"

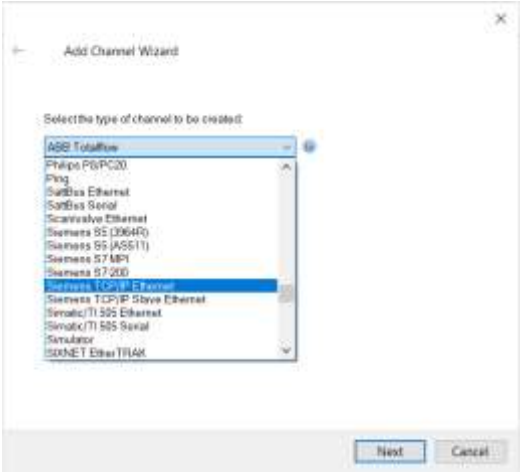


Figura 34. Selección del tipo de canal

- Se coloca el nombre de la practica en este caso "Practica_PID"

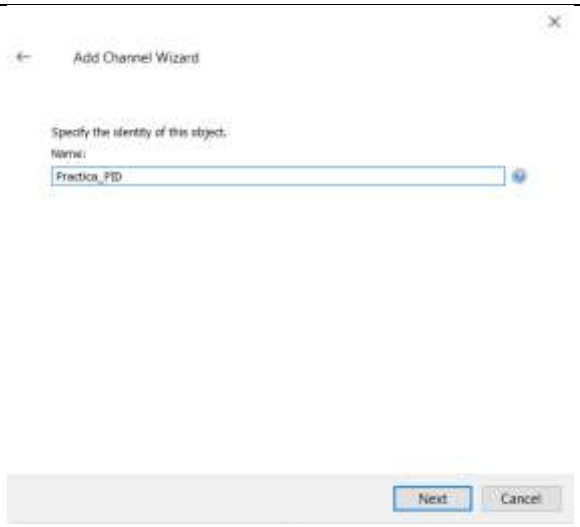


Figura 35. Nombre de la práctica

- Desplegar el menú de los adaptadores dando clic en "…" (véase Figura 36)

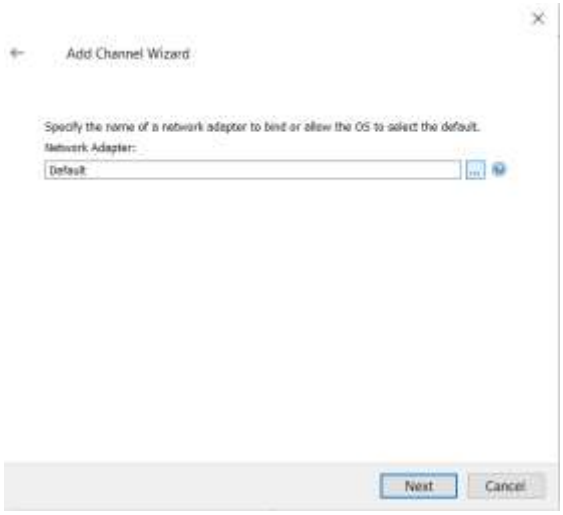


Figura 36. Selección de dirección IP

- Se escoge la opción con la dirección IP "192.168.0.8"

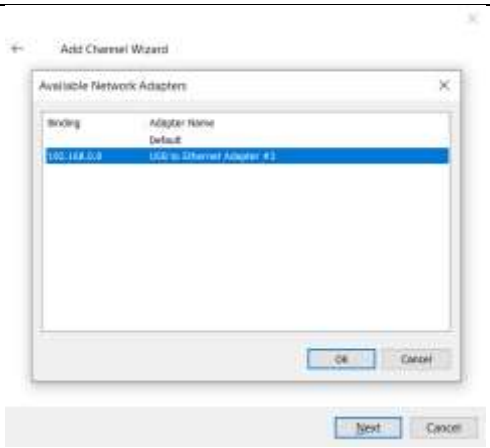


Figura 37. Dirección IP

- Seleccionar "Next" hasta llegar a la ventana de resumen, (véase Figura 38)
- Clic en "Finish"

Figura 38. Ventana de resumen

- Seleccionar "Clic to add a device" (Clic para añadir un dispositivo) en el canal creado (véase Figura 39)

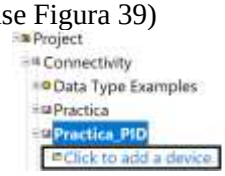


Figura 39. Anadir un dispositivo

- Colocar el nombre del dispositivo, clic en "Next"

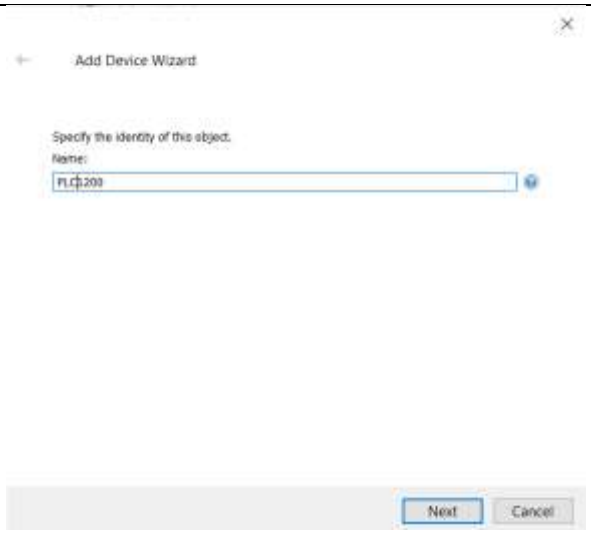


Figura 40. Nombre del dispositivo

- Seleccionar el dispositivo “S7-1200”, clic en “Next”

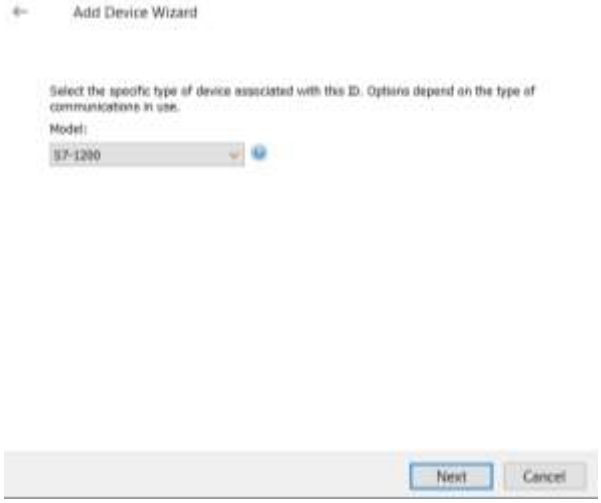


Figura 41. Seleccionar el modelo del dispositivo

- Se coloca la dirección IP del dispositivo “192.168.0.1”, clic en “Next”

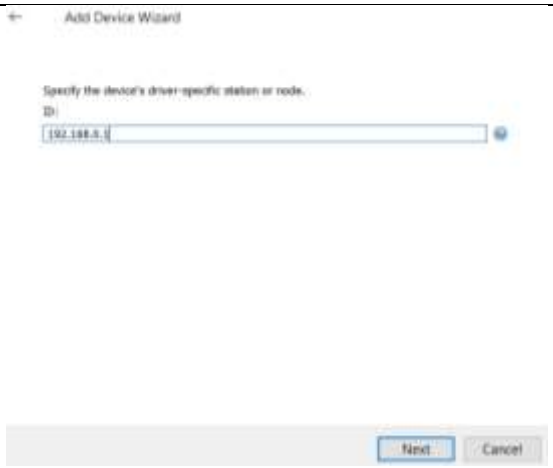


Figura 42. Dirección IP del dispositivo

- Seleccionar “Next” hasta la ventana de resumen.
- Clic en “Finish”

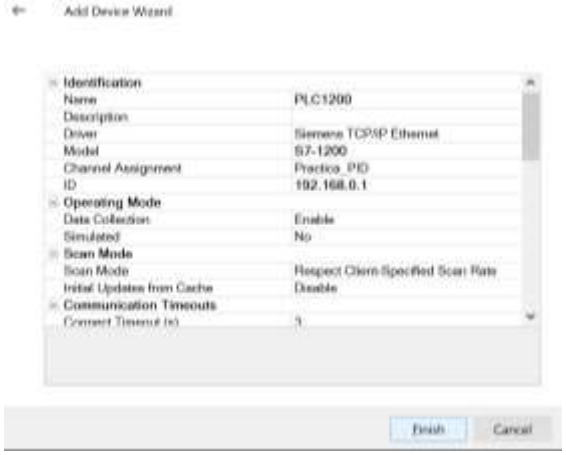


Figura 43. Ventana de Resumen

- Agregamos una nueva variable (véase Figura 44)

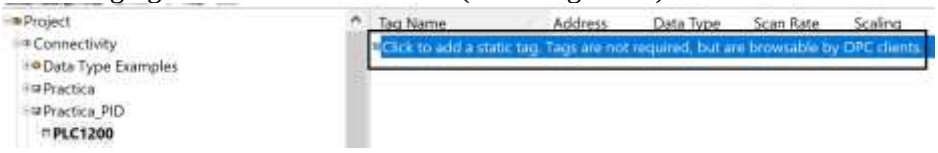


Figura 44. Se agregan variables

Tag Name	Address	Data Type
SalidaControlador	MD1	Float
Setpoit	MD115	Float
Tcamara	MD110	Float

Figura 45. Variables agregadas

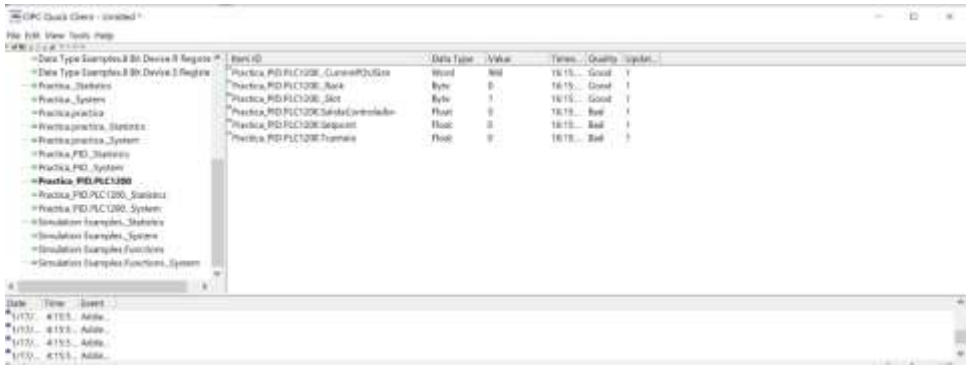


Figura 46. OPC Quick Client

5. Abrir el acceso directo del software LabVIEW 2019.

- Dar clic en el botón "Create New Project" (Crear un nuevo proyecto) (véase Figura 47).

Figura 47. Pantalla de inicio.

- Seleccionar "Desktop", seguido de "Blank Project" y clic en "Finish"

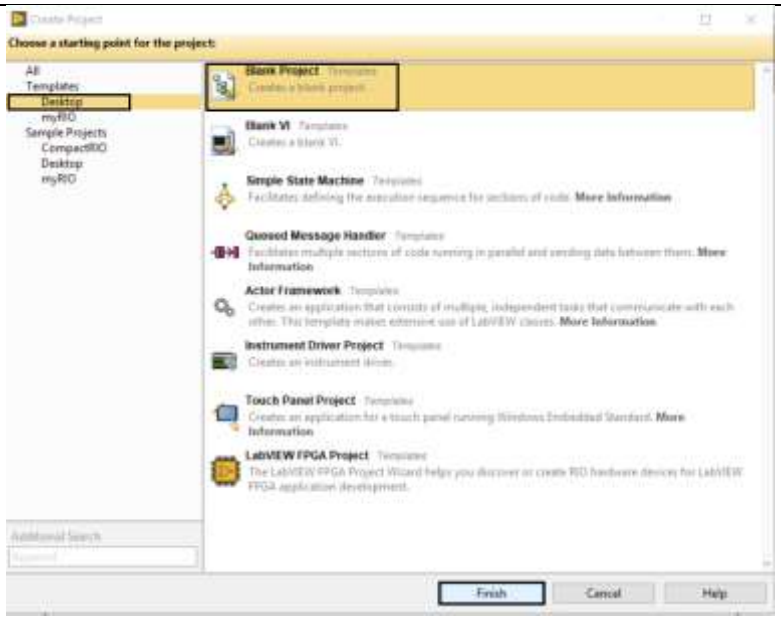


Figura 48. Creación del proyecto

- Clic derecho en "Computer", seleccionar "New" seguido de "VI" (véase Figura 49)

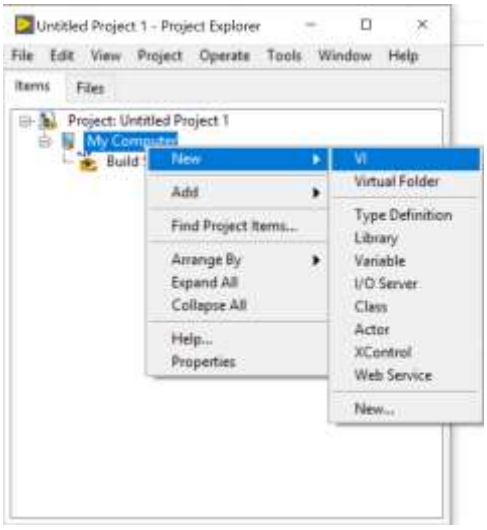


Figura 49. Agregar VI.

- Aparecerá la ventana de "Block Diagram" y "Front Panel".

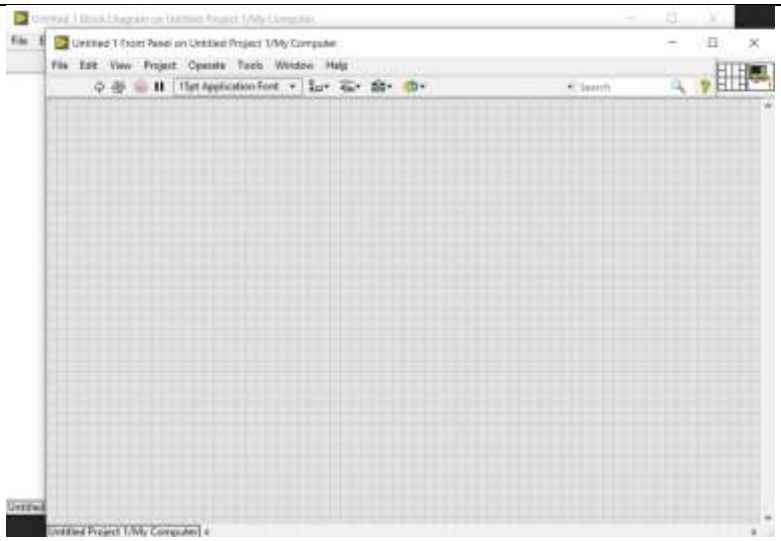


Figura 50. Ventanas del VI.

Agregando variables

- Clic derecho en "My Computer", seleccionar "New" seguido de "I/O Server"

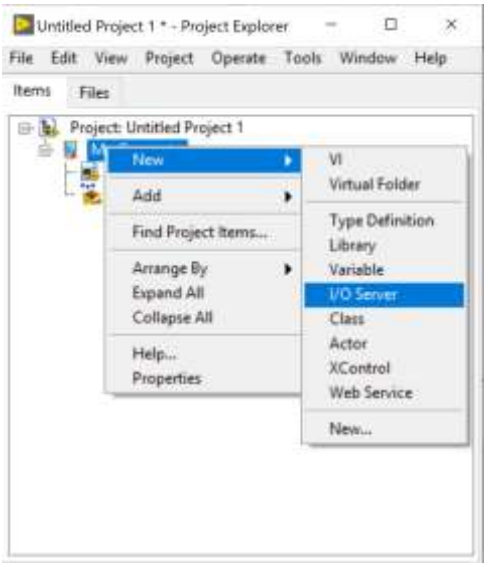


Figura 51. I/O Server

- Seleccionar "OPC Server" y clic en "Continue…"

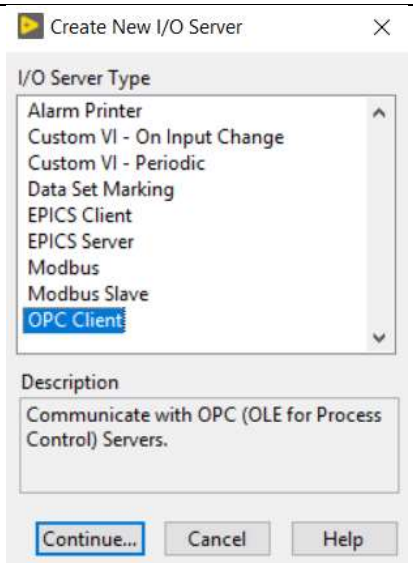


Figura 52. Creando un nuevo I/O Server

- Seleccionar “Kepware.KEPServerEX.V6”, clic en “OK”

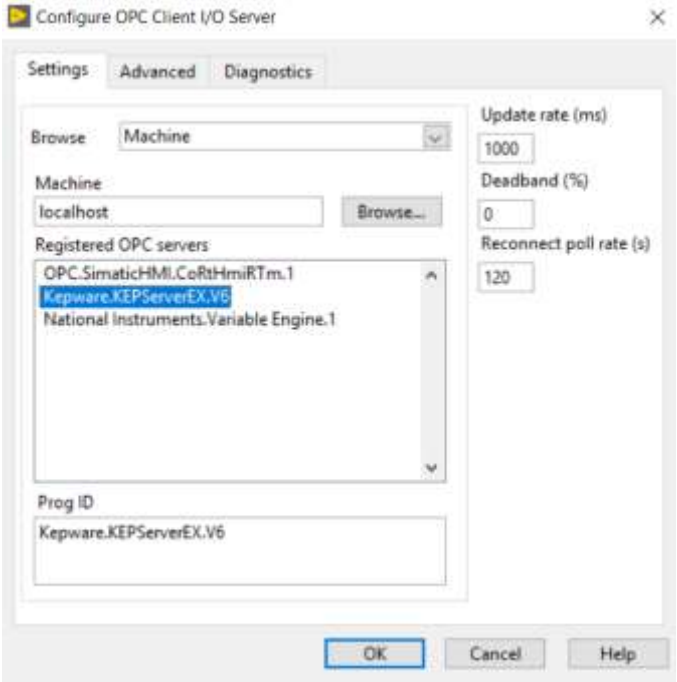


Figura 53. Configurando OPC Client I/O Server

- En la ventana principal del árbol del proyecto, clic derecho en OPC1 y seleccionar “Create Bound Variables” (véase Figura 54)

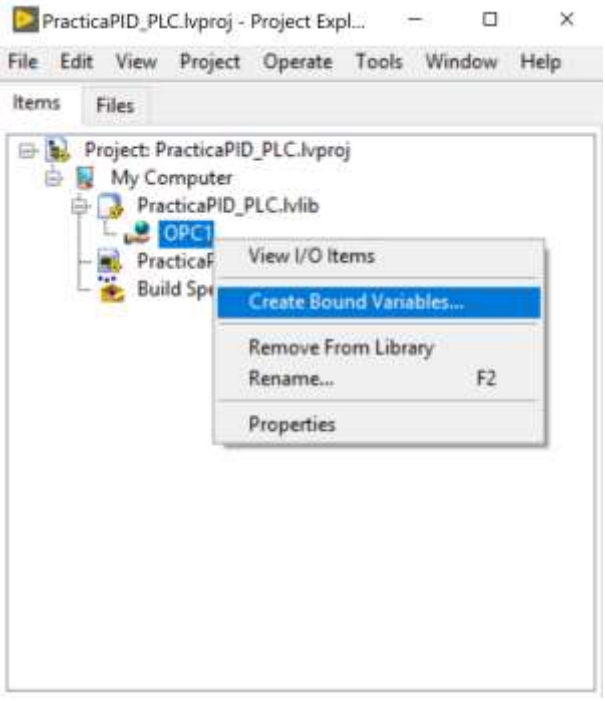


Figura 54. Create Bound Variables

- En la ventana emergente llamada “Create Bound Variables” dar clic en “Project: PracticaPID_PLC.lvproj”, seguido de “My Computer”, seleccionar “PracticaPID_PLC.lvlib”, clic en OPC1, seleccionar “Practica_PID”, clic en “PLC1200”, seleccionar “_System”, escoger la variable “SalidaControlador”, clic en “Add”
- Repetir “Add” para las variables “Setpoint” y “Tcamara”
- Clic en “OK”

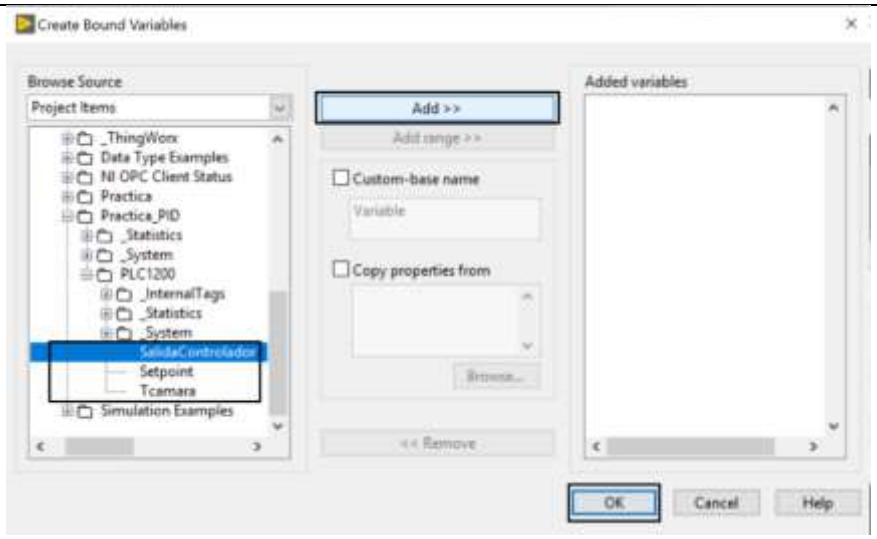


Figura 55. Añadiendo variables

- En la ventana principal del proyecto aparecen las variables agregadas (véase Figura 56).

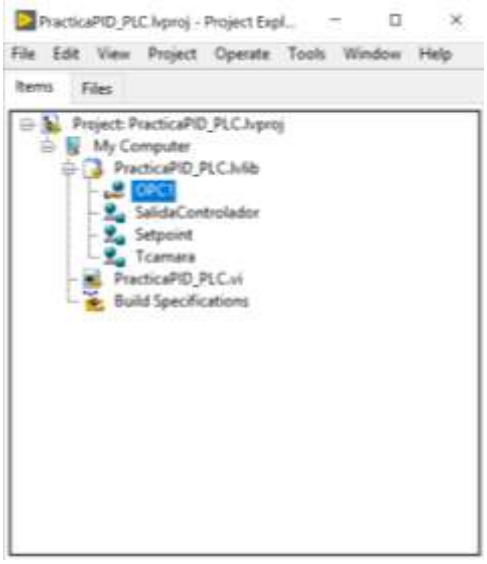


Figura 56. Ventana principal del proyecto

- Arrastrar cada variable (SalidaControlador, Setpoint, Tcamara) a la ventana “Block Diagram” (véase Figura 57)

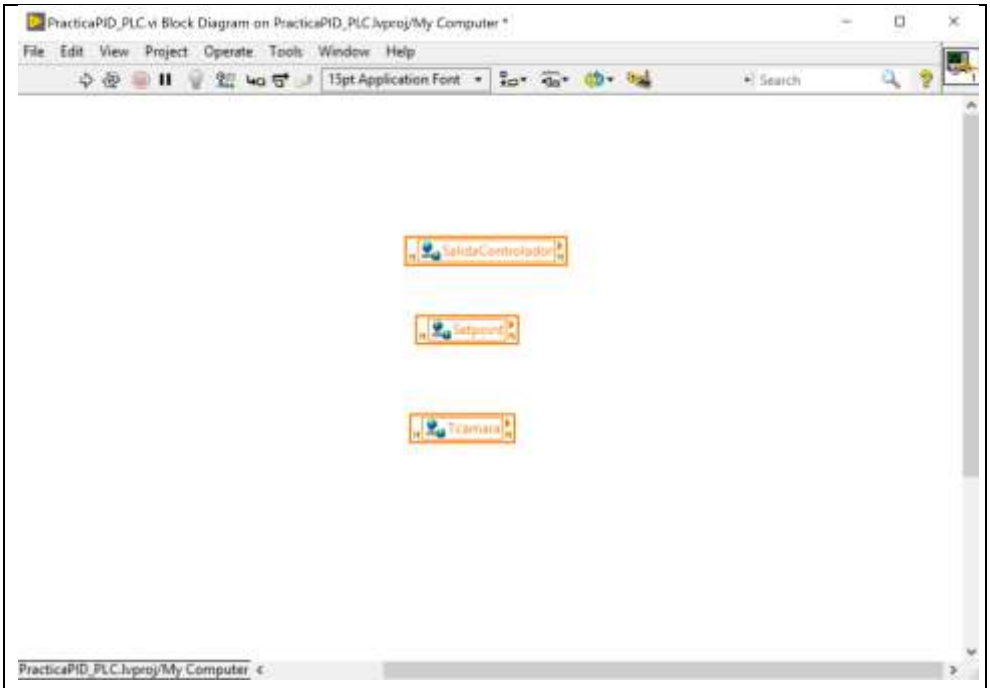


Figura 57. Agregando las variables en la ventana "Block Diagram"

- Clic derecho en la ventana "Block Diagram", seleccionar "Structures" y arrastrar el bloque "While Loop" (véase Figura 58).
- Dibujar el tamaño del "While Loop" con el cursor, las variables agregadas deberán estar dentro del mismo (véase Figura 59).

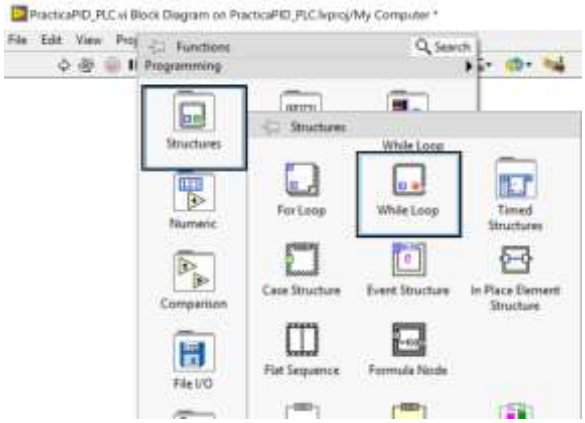


Figura 58. Agregar bloque "While Loop"

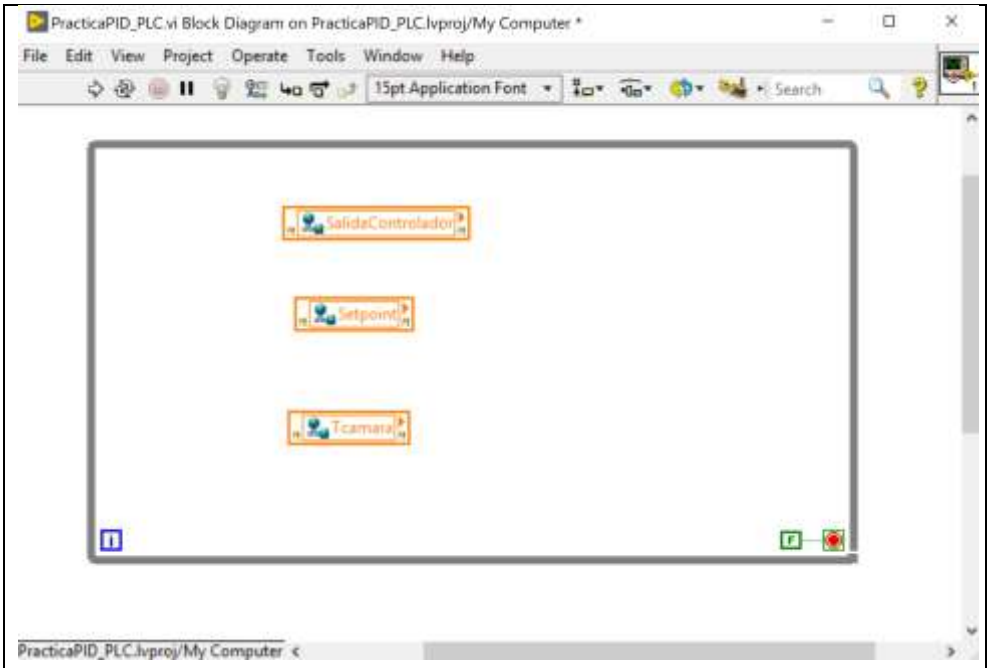


Figura 59. Agregando el bloque "While Loop"

- Clic derecho en la variable "Setpoint", seleccionar "Access Mode" y escoger "Write" (véase Figura 60)

Figura 60. Estableciendo las variables como salidas y entradas.

- Repetir para las variables "Tcamara" y "SalidaControlador" como "Read"
- Clic derecho en la salida de la variable "Tcamara", seleccionar "Create Indicador" (véase Figura 61).
- Repetir para "SalidaControlador"

Figura 61. Creación de Indicadores en las variables

- Clic derecho en la entrada "Setpoint", seleccionar "Create Control" (véase Figura 62)

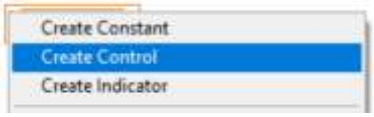


Figura 62. Creación de Controles en las variables

- Clic derecho en la ventana "Front Panel", seleccionar "Numeric" y arrastrar a la ventana el bloque "Thermometer (Termómetro) (véase Figura 63)

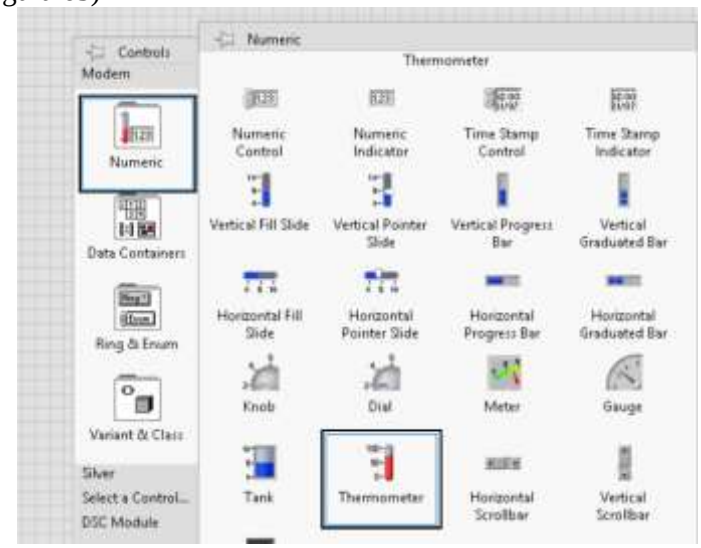


Figura 63. Termómetro

- Se agrega el bloque para guardar los datos al hacer clic derecho en la ventana "Block Diagram", seleccionar "File I/O" y arrastrar el bloque "Write Meas File" a la ventana.

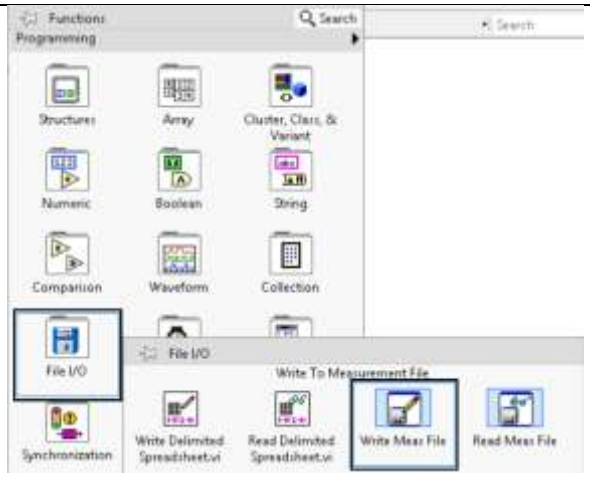


Figura 64. Bloque "Write Meas File"

- Al arrastrar el bloque la ventana de configuraciones de este se abre, seleccionar la ubicación deseada para guardar el archivo y configurar el bloque como se muestra (véase Figura 66)

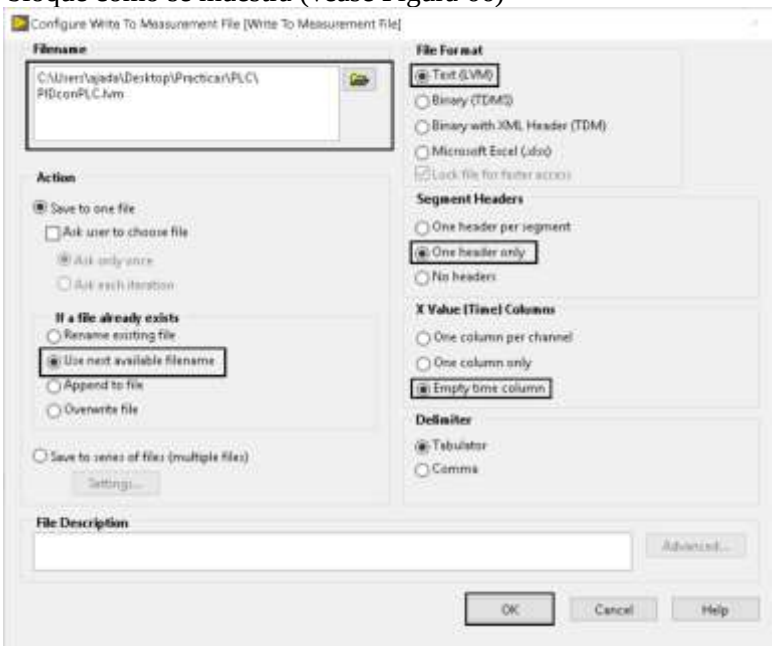


Figura 65. Configuración del bloque para guardar los datos

- Se conectan las señales como se muestra en la figura 66.

APÉNDICES

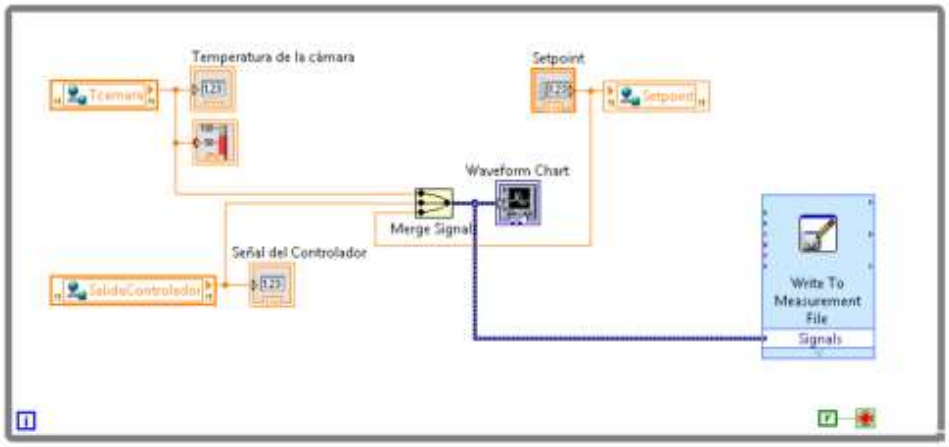


Figura 66. Ventana "Block Diagram"

Figura 67. Ventana "Front Panel"

- Se inicia el programa y se coloca el valor "60" en "Setpoint"

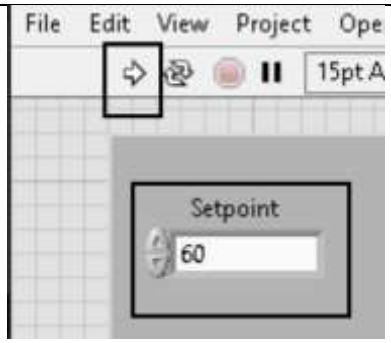


Figura 68. Ejecución del programa

6. Ejecución del bloque PID

- En el programa TIA Portal, ubicarse en "Cyclic Interrupt", seleccionar "Start CPU" y seleccionar "Monitoring on/off"

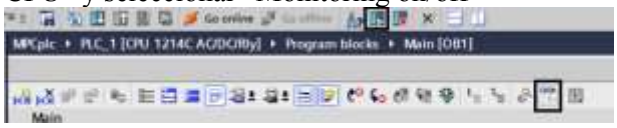


Figura 69. Iniciando el controlador

- Doble clic en la puesta en marcha del controlador (véase Figura 70)

Figura 70. Puesta en marcha del controlador

- En la ventana "PID_Compact_1", seleccionar en "Sampling time", "1s", luego clic en "Start"

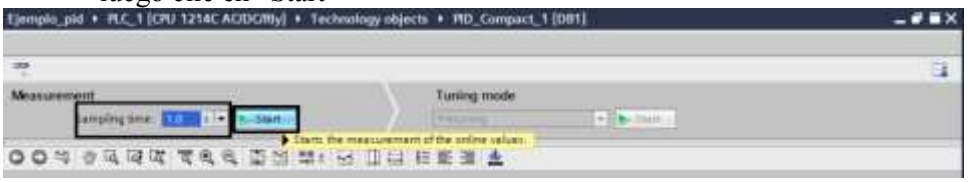


Figura 71. Iniciando controlador

- Para iniciar el autotuning del PID, seleccionar “Start PID_Compact”, luego en las opciones de “Tuning mode” escoger “Fine tuning”, clic en “Start”. Esperar hasta que se complete el proceso de auto sintonía.

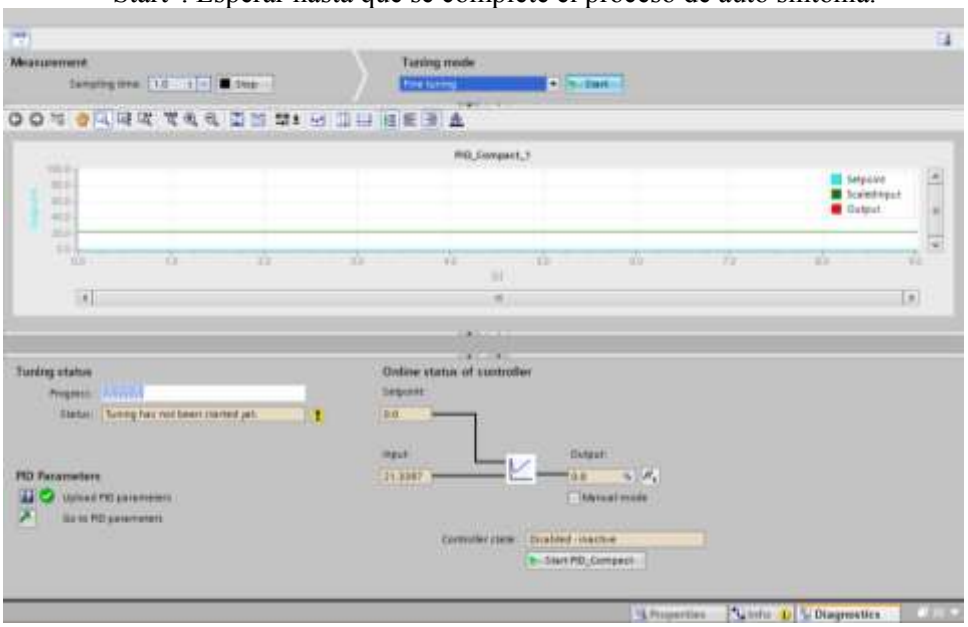


Figura 72. Iniciando Autotuning

- Una vez que se finalizado el progreso, dar clic en “Upload PID parameters”, si se desea observar los parámetros del controlador dar clic en “Go to PID parameters”

Figura 73. Actualizando parámetros obtenidos por el autotuning

- Se puede comprobar el funcionamiento del controlador al colocar diferentes valores en “Setpoint” en el programa de LabVIEW (véase Figura 74)

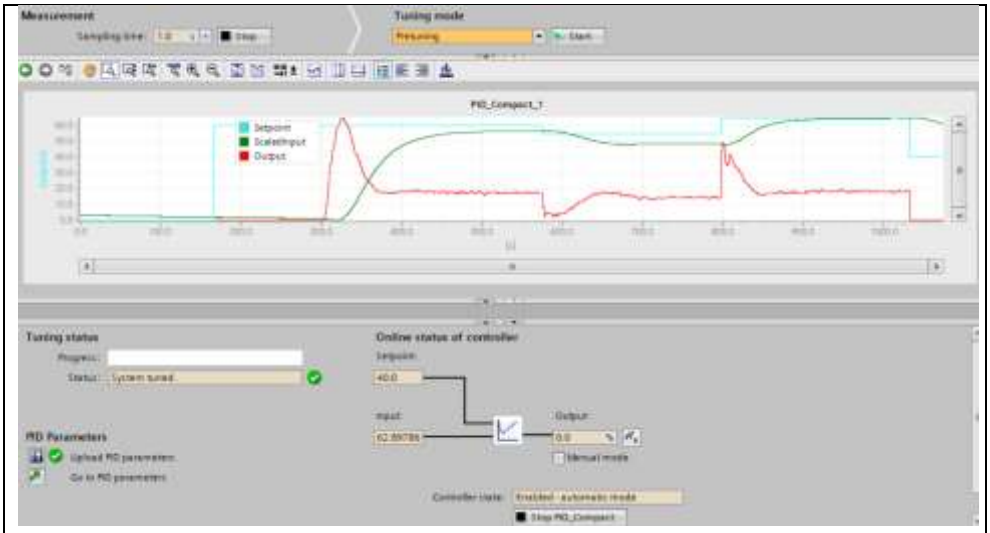


Figura 75. Comprobando el controlador

7. Gráfica de datos obtenidos en el software Matlab

- Abrir el software Matlab R2017b.

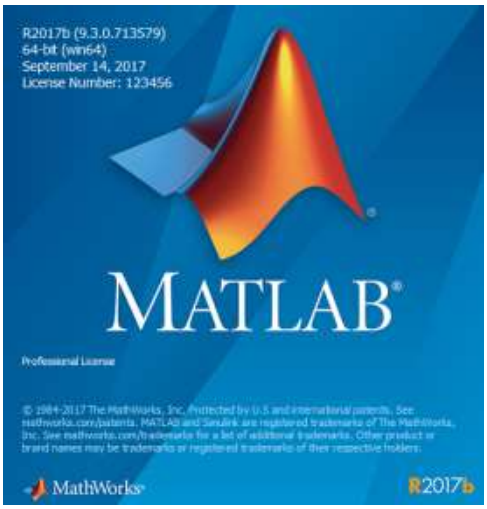


Figura 76. Software MATLAB R2017b.

- Clic en la ventana "HOME"
- En la sección "VARIABLE" (Variable), seleccionar "Import Data" (Importar Datos) (véase Figura 77).

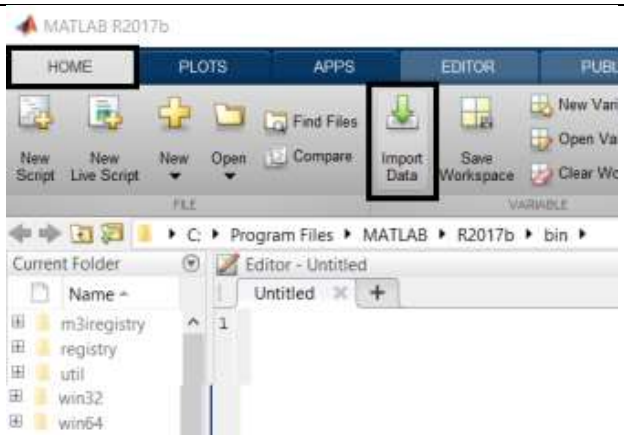


Figura 77. Importar datos adquiridos.

Se abre la ventana "IMPORT"

- En la sección "IMPORTED DATA", en la opción "Output Type" se escoge "Column vectors" (véase Figura 78).
- Para cambiar el nombre de cada columna, clic sobre la celda del nombre y escribir el nombre respectivo (véase Figura 78).
- Dar clic en "Import Vector".

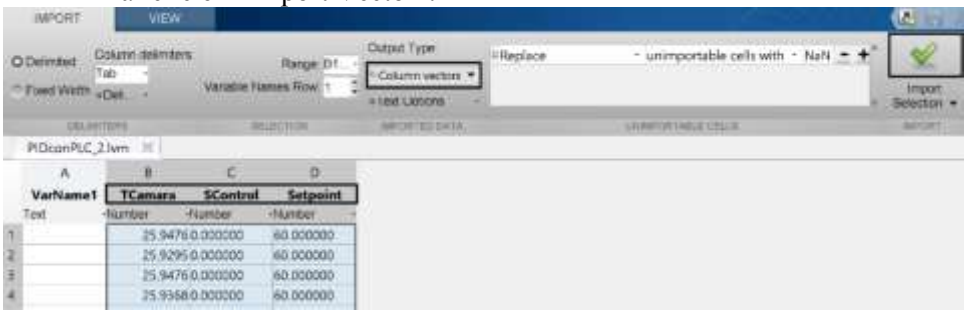


Figura 78. Importando datos

<u>Guardar los datos ubicados en el Workspace</u>

Figura 79. Variables Workspace

- En la pestaña “HOME”, en la sección “VARIABLE”, seleccionar “Save Workspace” (Guardar Espacio de Trabajo) (véase Figura 80).

Nota: Guardar los datos en la misma carpeta que el “Script”, el nombre del archivo en este caso se llamará datos.mat.

Figura 80. Guardando los datos

- Se crea un nuevo proyecto escogiendo “New Script” (Nuevo Script)

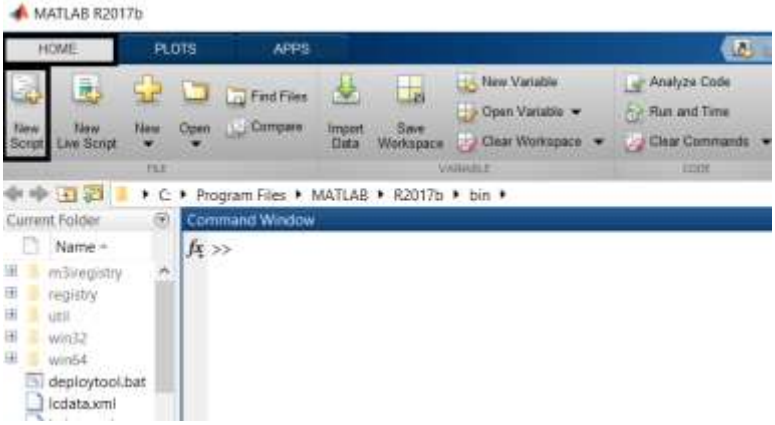


Figura 81. Creando “New Script”

- Los datos guardados se grafican con el siguiente código, estos se observan en la Figura 82

```
load datos.mat
ds=1;
t=0:ds:length(Setpoint)-ds;
figure(1)
subplot(211);
plot(t,Setpoint,'b','LineWidth',1.2); xlabel('Time(s)'); ylabel('Temperatura (ºC)');
hold on; grid on
plot(t,TCamara,'r','LineWidth',1.2);
title('Controlador PID')
legend('Referencia','Temperatura de la cámara')
subplot(212);
plot(t,SControl,'k','LineWidth',1.2);
xlabel('Time(s)');ylabel('Porcentaje'); grid on
legend('Señal de Control')
```

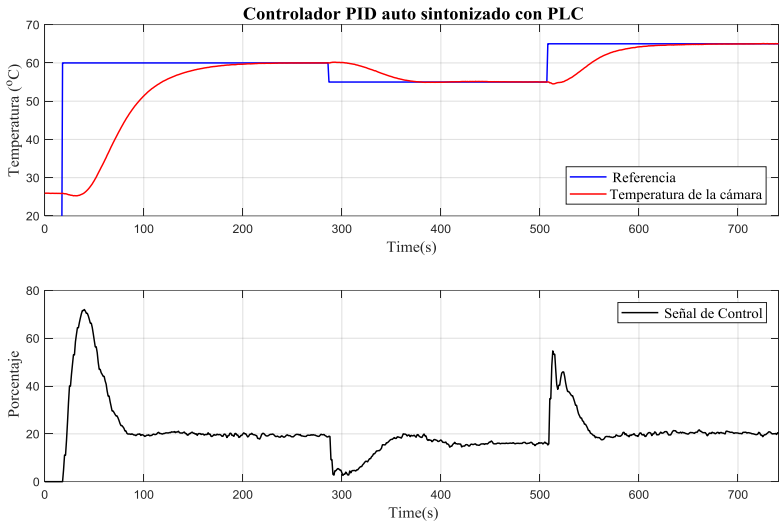


Figura 82. Control PID con auto sintonía del PLC

8. Esquema de conexión

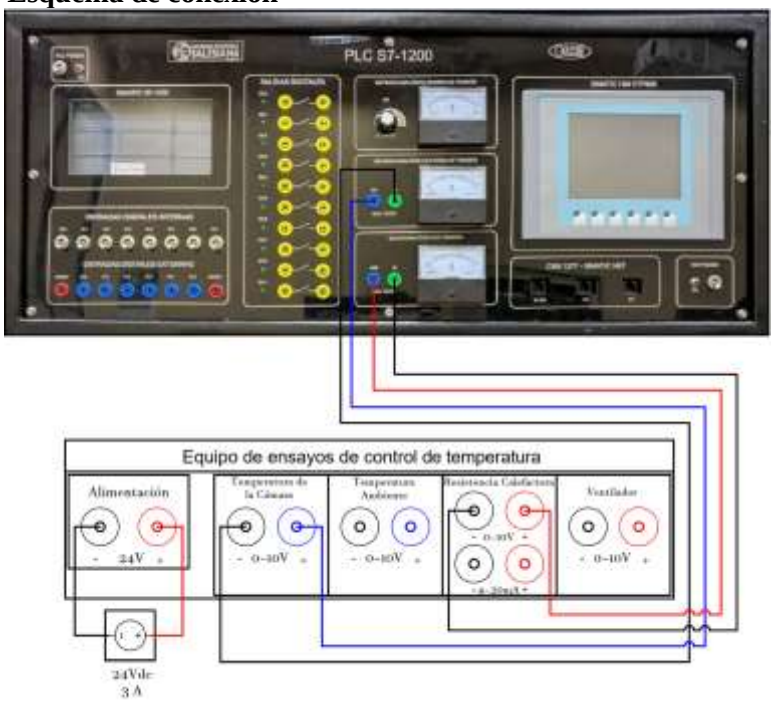


Figura 89. Diagrama de conexión

APÉNDICES

RESULTADO(S) OBTENIDO(S):
CONCLUSIONES:
RECOMENDACIONES:

APÉNDICES

- **GUÍA DE PRÁCTICA 5: IMPLEMENTACIÓN DE UN CONTROLADOR PID AUTO SINTONIZADO USANDO EL MÓDULO TK4S-T4RC Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA**

FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: Electrónica y automatización	**ASIGNATURA**: Teoría de control

NRO. PRÁCTICA:	5	**TÍTULO PRÁCTICA**: **IMPLEMENTACIÓN DE UN CONTROLADOR PID AUTO SINTONIZADO USANDO EL MÓDULO TK4S-T4RC Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA**

OBJETIVO

1. Conocer las características generales del módulo TK4S-T4RC
2. Implementar un controlador PID de auto sintonía
3. Monitorear las señales del sistema con el equipo NI myRIO.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

a) Estructura de un sistema de control en lazo cerrado.
b) Manejo de sistemas embebidos.
c) Familiarización con el software LabVIEW 2019 y el dispositivo NI myRIO

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Módulo TK4S-T4RC	1	Autonics	-
Equipo de ensayos de control de temperatura	1	CTI Lab	-
Fuente DC 24 volts 3.2 Amps	1	-	-
NI myRIO	1	National Instruments	
Computadora S.O. Windows 7 o			

superior, 32 o 64 bits				
Software LabVIEW				
Cables varios				

Tabla 1. Equipos, instrumentos y software

3. Exposición

Módulo TK4S-T4RC: controlador de temperatura de alta precisión, alcanza un tiempo de muestreo de 50ms, soporta control por calentamiento y enfriamiento, así como un modo manual. Posee dos salidas de control destinadas a un elemento calentador y uno enfriador. [21]

Figura 1. Módulo Autonics TK4S-T4RC. [18]

***Kit NI MyRio*:** proporciona entradas y salidas configurables que permite ensenar e implementar múltiples conceptos de diseño con un dispositivo.

Incluye entradas analógicas, salidas analógicas, líneas E/S digitales, Leds, un push-Button, un acelerómetro interno, un FPGA Xilinx y un procesador dual-Core ARM Cortex-A9. [18]

Figura 1. Módulo NI MyRio. [18]

Datos técnicos:

NI myRIO	
Memoria	No volátil (512 MB) DDR3 (256 MB)
FPGA	Xilinx Z-7010
Wireless Characteristics	Radio (IEEE 802.11 b,g,n). Frecuencia de banda (ISM 2.4 GHz). Ancho de banda (20MHz). Canales (USA 1 A 11, Internacional 1 a 13). TX (+10dBm max). Outdoor range (Hasta 150m). Directividad de la antena (Omnidireccional). Seguridad (WPA, WPA2, WPA2-Enterprise).
USB Ports	USB host-device port (2.0 Hi-Speed)
Acelerómetro	Número de ejes (3). Rango (± 8g).

	Resolución (12bits). Taza de muestreo (800 S/s). Ruido (3.9 mg(rms) a 25ºC)
Requerimientos de potencia	Rango de voltaje de la fuente de alimentación (6 a 16 VCC). Consumo máximo de energía (14 W). Consumo de energía inactivo típico (2.6 W).

Tabla 2. Características del dispositivo NI-myRIO

***Equipo de ensayo de control de temperatura**:* este dispositivo permite realizar ensayos de control de temperatura con picos de hasta 55 °C, equipado con una niquelina fabricada con alambre de nicromo, tres ventiladores 3010 para perturbaciones, sensores de temperatura tipo RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C.

El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.

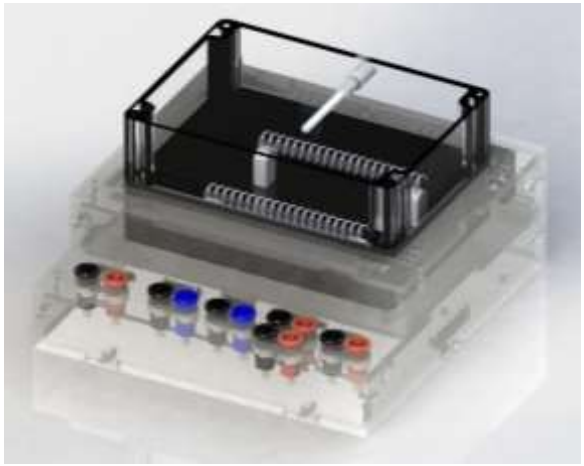


Figura 2. Equipo de ensayo de control de temperatura.

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. **Configuración de parámetros del modulo**

- Restauramos los valores por defecto del dispositivo, para ello mantenemos pulsadas las teclas de navegación hasta que visualicemos "Init" (Inicializar) seleccionamos "yes" y confirmamos pulsando la tecla "mode" (véase figura 3).

Figura 3. Restaurar valores por defecto

- Mantenemos pulsada la tecla "Mode" durante 3 segundos hasta visualizar "Par1" (Grupo de parámetros 1).

Figura 4. Grupo de parámetros 1

- Pulsamos la tecla "Arriba" hasta ubicarnos en "Par3" (Grupo de parámetros 3) y pulsamos "mode" para acceder a los parámetros del grupo 3(véase Figura 5).

Figura 5. Grupo de parámetros 3

- Dentro de la sección "In-t" (Entrada), pulsamos "abajo" hasta visualizar "Au1" (Entrada analógica 0-10v) y guardamos cambios con "mode" (véase figura 6)

Figura 6. Menú de opciones de entrada

- Pulsamos mode hasta visualizar “L-rG” (límite inferior de la entrada analógica) y establecemos su valor en 00.00 (véase Figura 7).

Figura 7. Límite inferior de la entrada analógica

- Nos ubicamos en “H-rG” (límite superior de la entrada analógica) y establecemos su valor en 10.00 (véase figura 8).

Figura 8. Límite superior de la entrada analógica

- Pulsamos “mode”, hasta visualizar “dot” (punto decimal) y establecemos su valor 0.0 (véase figura 9).

Figura 9. Ubicación del punto decimal

- Pulsamos "mode" y nos ubicamos en "L-SC" (límite inferior de la escala) y establecemos su valor en 000.0 (véase Figura 10).

Figura 10. Límite inferior de la escala de la entrada

- Pulsamos "mode" y nos ubicamos en "H-SC" (límite superior de la escala) y establecemos su valor en 100.0 (véase Figura 11).

Figura 11. Límite superior de la escala de la entrada

- Pulsamos "mode" y nos ubicamos en "d.Unt" (unidades mostradas) y establecemos su valor en °C (véase Figura 12).

Figura 12. Unidades mostradas en el indicador

- Pulsamos "mode" hasta visualizar "o-Ft" (tipo de control) y establecemos su valor en "PI d" (véase Figura 13).

Figura 13. Comportamiento de la salida del dispositivo

- Pulsamos “mode” hasta visualizar “H-t” y establecemos su valor en 120.0 (véase Figura 14).

Figura 14. Tiempo de acción de la salida 1

- Mantenemos pulsado “mode” durante 3 segundos para regresar al menú anterior e ingresamos al grupo de parámetros 5 “Par5” (véase Figura 15).

Figura 15. Grupo de parámetros 5

- Pulsamos "mode" hasta visualizar "USEr" (Nivel de Usuario) y establecemos su valor en "HIGH" (véase Figura 16).

Figura 16. Nivel de usuario

- Mantenemos pulsado la tecla "mode" durante 3 segundos para regresar al menú anterior, y nos ubicamos en el grupo de parámetros 4 (véase Figura 17).

Figura 17. Grupo de parámetros 4

- Pulsamos la tecla “mode” hasta visualizar “Al-1” (Operación de la salida de alarma 1) y establecemos su valor en “off” (véase Figura 18).

Figura 18. Comportamiento de alarma 1

- Pulsamos “mode” hasta visualizar “Ao.ñ2” y establecemos su valor en “H-ñu” (véase Figura 19).

Figura 19. Usar salida 2 como salida de corriente

- Mantenemos pulsado la tecla "mode" durante 3 segundos para regresar al menú anterior (véase Figura 20).

Figura 20. Grupo de parámetros 1

- Mantenemos pulsado la tecla "mode" durante 3 segundos para regresar al estado de espera (véase Figura 21).

Figura 21. Pantalla principal

- Utilizando las flechas establecemos una referencia entre 30 y 50 ℃ (véase Figura 22).

Figura 22. Establecer una referencia

- Pulsamos la tecla “mode” durante 3 segundos y nos ubicamos en el “par2” (grupo de parámetros 2).

Figura 23. Grupo de parámetros 2

- Pulsamos la "mode" hasta visualizar "At" (Autotuning) y establecemos su valor en "on" (véase Figura 24).

Figura 24. Activar autotuning

- Esperamos unos minutos hasta que se termine el proceso de autotuning (véase Figura 25).

Figura 25. Pantalla principal con autotuning activado

2. Monitoreo de señales con LabVIEW

- Abrimos LabVIEW e iniciamos un nuevo proyecto (véase Figura 26).

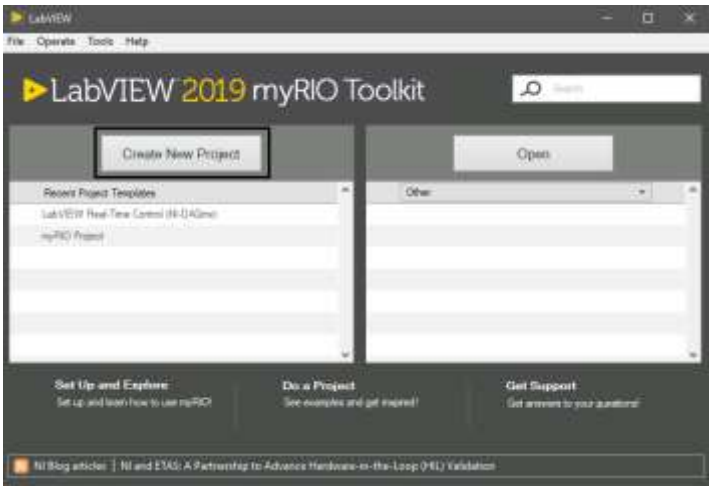


Figura 26. Ventana de inicio LabVIEW 2019

- Creamos un proyecto en blanco

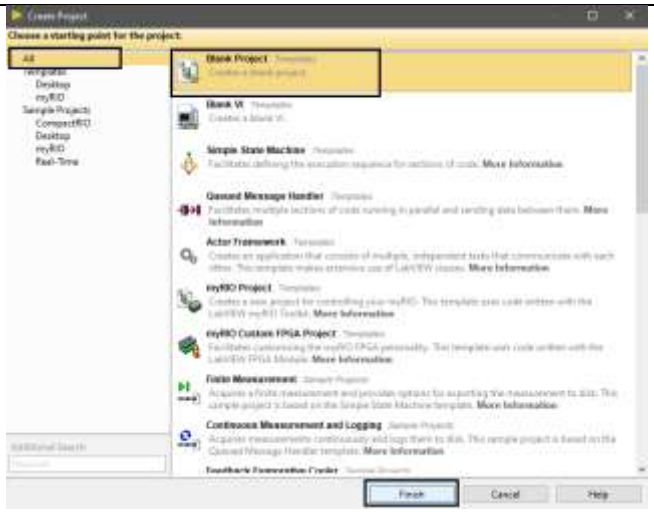


Figura 27. Crear nuevo proyecto en blanco

- En el explorador del proyecto, damos clic derecho en la carpeta raíz, seleccionamos "New" (Nuevo) y damos clic en "Targets and Devices" (Objetivos y dispositivos) (Véase Figura 28).

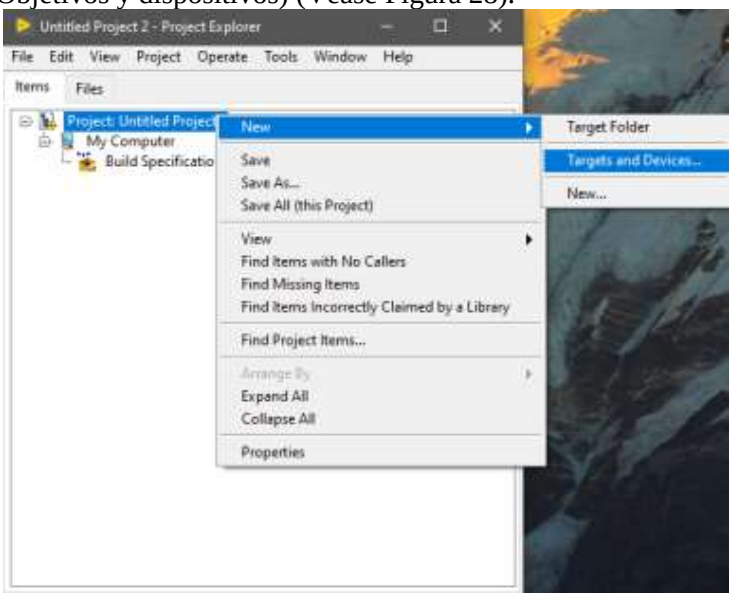


Figura 28. Agregar un dispositivo

- Seleccionamos "New Target or Device" (Nuevo objetivo o dispositivo), desplegamos la carpeta myRIO, seleccionamos myRIO-1900 y damos clic en "OK" (véase Figura 29).

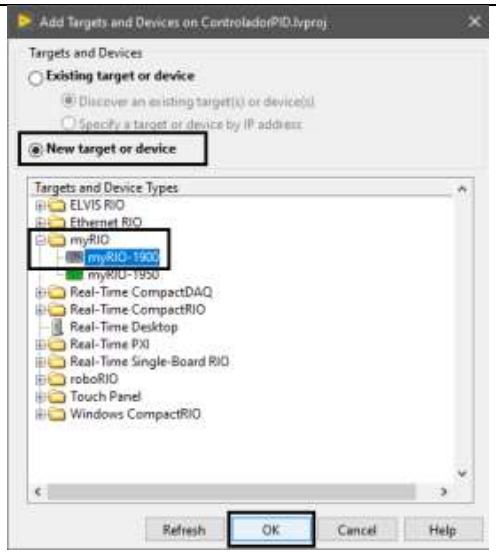


Figura 29. Agregar myRIO-1900

- Damos clic derecho en "RT myRIO Target" (Dispositivo myRIO), seleccionamos "New" (Nuevo) y damos clic en "VI" (Interfaz Gráfica) (véase Figura 30).

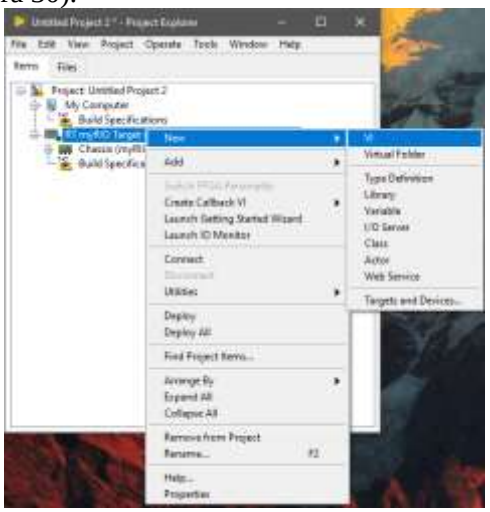


Figura 30. Agregar un VI

- Agregamos una estructura while, una demora y un control para detener el proceso. Al agregar el bloque "Wait" (Esperar), aparecerá una ventana. Debemos configurar el "Counter units" (Unidades del contador) en milisegundos (véase Figura 31).

Figura 31. Estructura while

- Damos clic derecho y accedemos al buscador de bloques de LabVIEW. (véase Figura 32).

Figura 32. Acceder al buscador de LabVIEW

- Usando el buscador agregamos el bloque "Analog input" (Entrada Analógica), al agregarlo aparecerá un asistente en el que modificaremos el nombre del bloque, así como las entradas físicas que vamos a utilizar, al finalizar pulsamos "OK". (véase Figura 33).

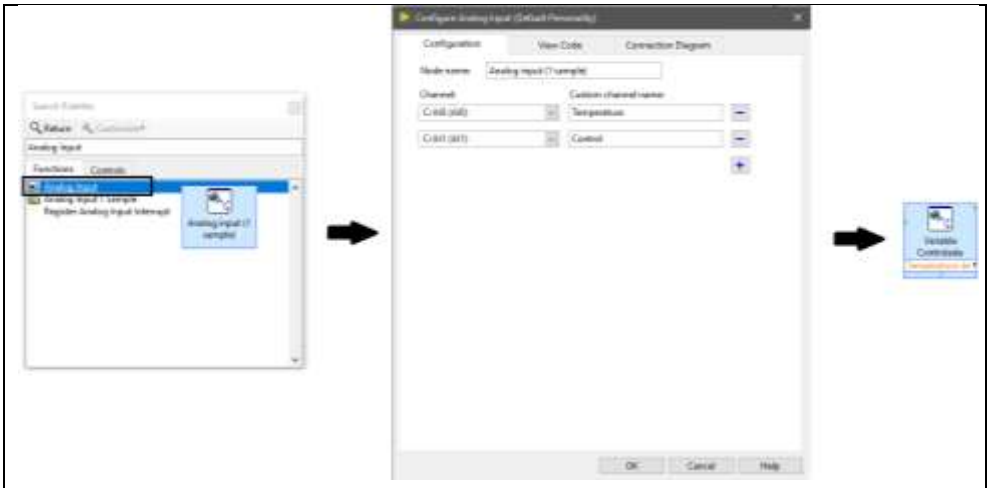


Figura 33. Configuración de entradas analógicas

- Agregamos el bloque "Set Dynamic Data Atributes" (Establecer atributos de los datos) y modificamos sus parámetros (véase Figura 34).

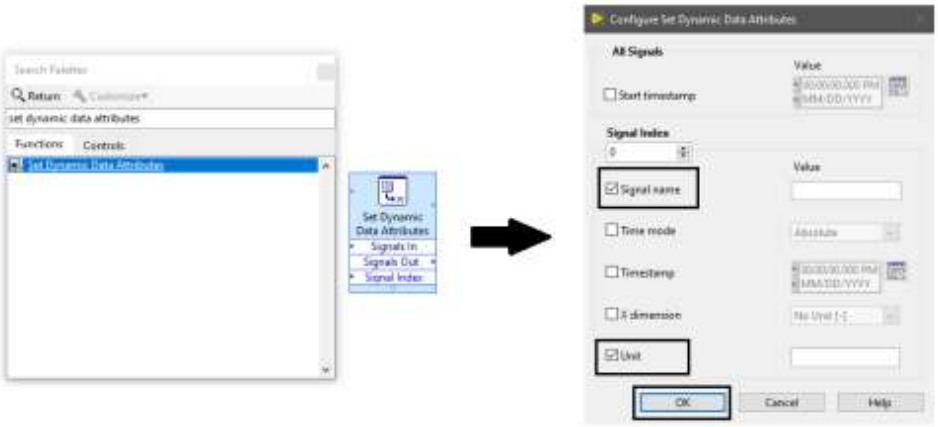


Figura 34. Configuración del bloque "Set Dynamic Data Atributes"

- Conectamos los bloques "Set Dynamic Data Atributes" y añadimos los atributos requeridos en las señales (véase Figura 35)

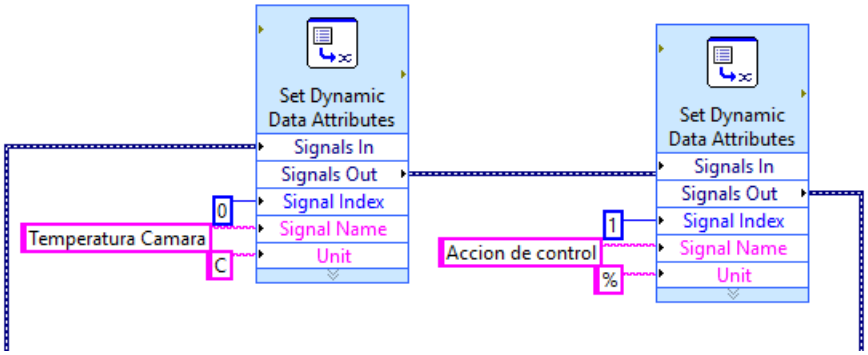


Figura 35. Conexión de bloques "Set Dynamic Data Atributes"

- Agregamos el bloque "Write to measurement file" (Escribir a archivo de datos) (véase Figura 36).

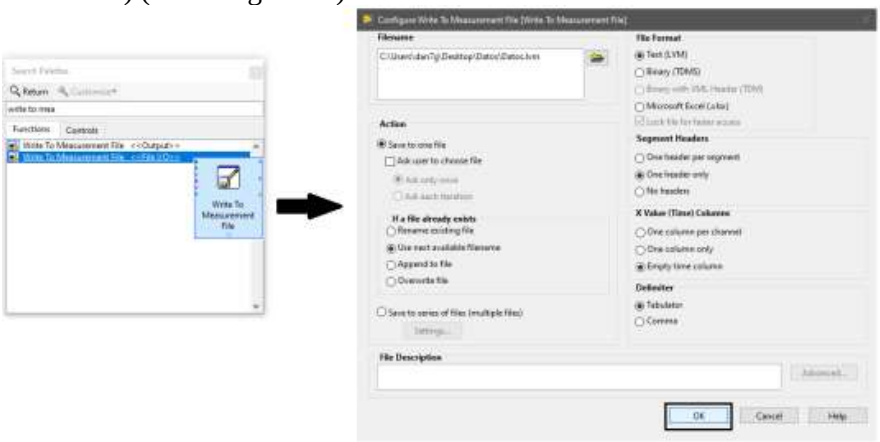


Figura 36. Agregar el bloque "Write to measurement file"

- En la ventana de configuración seleccionaremos la dirección de destino del archivo de datos (véase Figura 37).

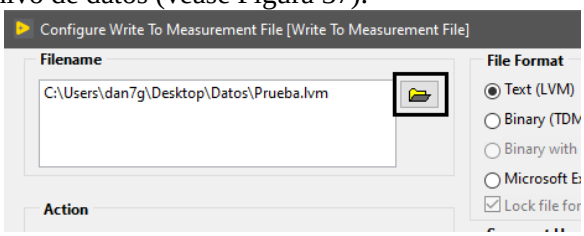


Figura 37. Selección de dirección de archivo de datos

- Se debe asignar un nombre al archivo de datos (véase Figura 38)

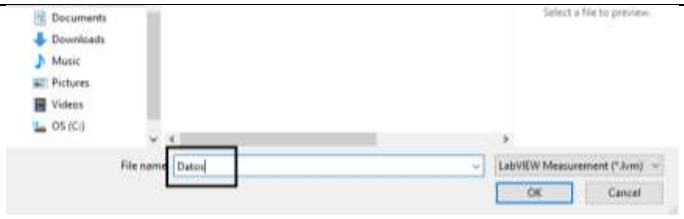


Figura 38. Nombre del archivo de datos

- Configuramos los parámetros restantes (véase Figura 39).

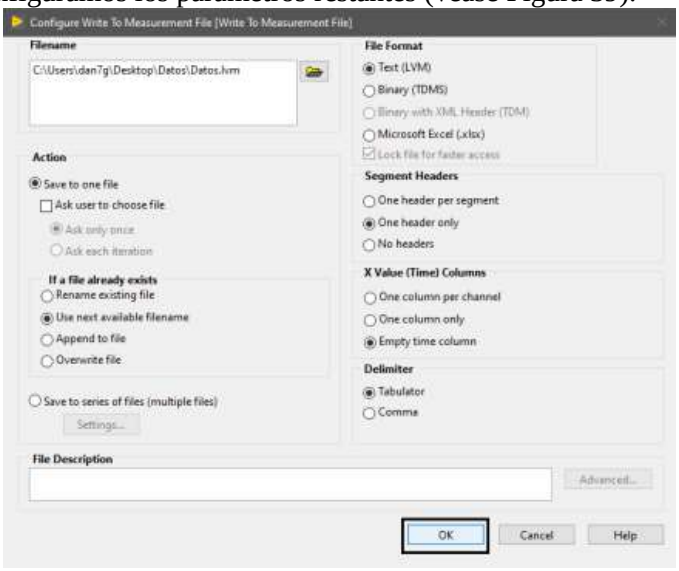


Figura 39. Configuración del bloque "Write to measurement file"

- Utilizando el bloque "Merge Sygnals" agrupamos las señales de interés y las conectamos el conjunto como se muestra en la Figura 40.

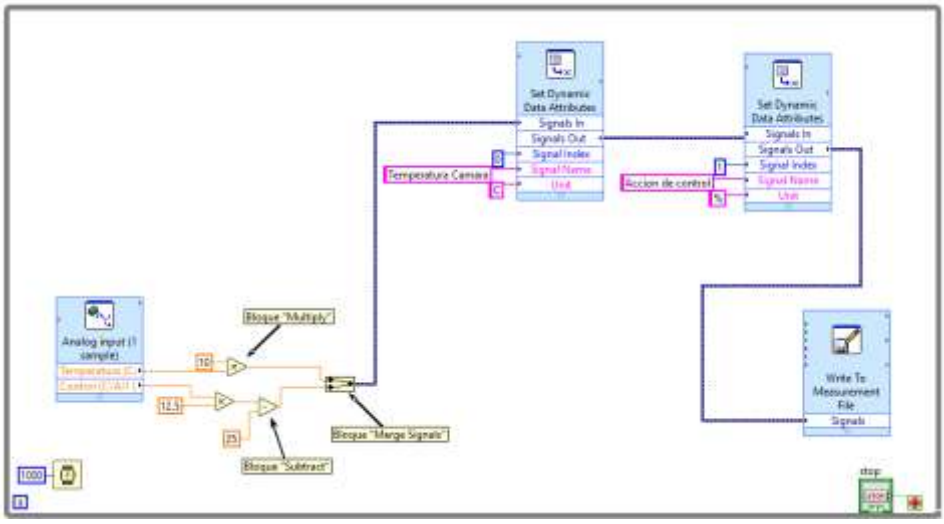


Figura 40. Conexión de secciones del programa

- Hacemos clic derecho, nos ubicamos en la carpeta "Graph" y seleccionamos "Waveform Chart" (véase Figura 41).

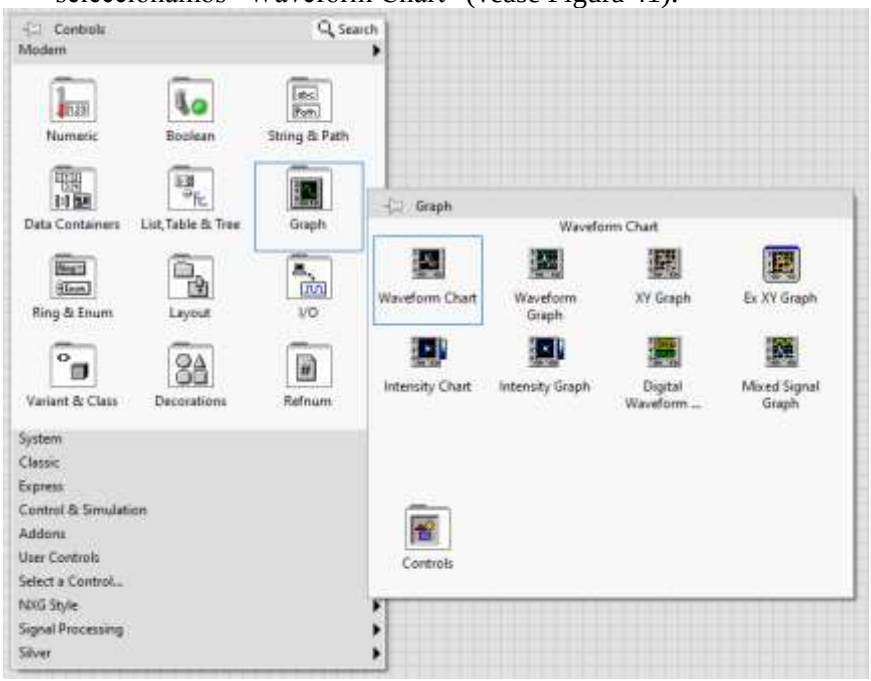


Figura 41. Agregar un "Waveform Chart"

- En el diagrama de bloques conectamos el bloque “Waveform Chart” con las señales de interés (véase Figura 42)

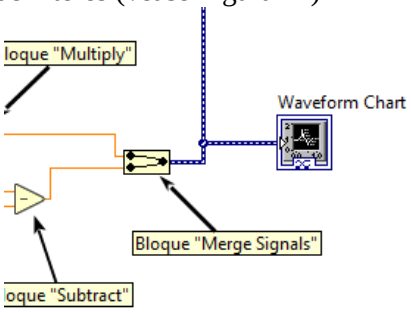


Figura 42. Conexión del bloque “Waveform Chart”

- Acomodamos los elementos de la interfaz gráfica para una mejor visualización (véase Figura 43)

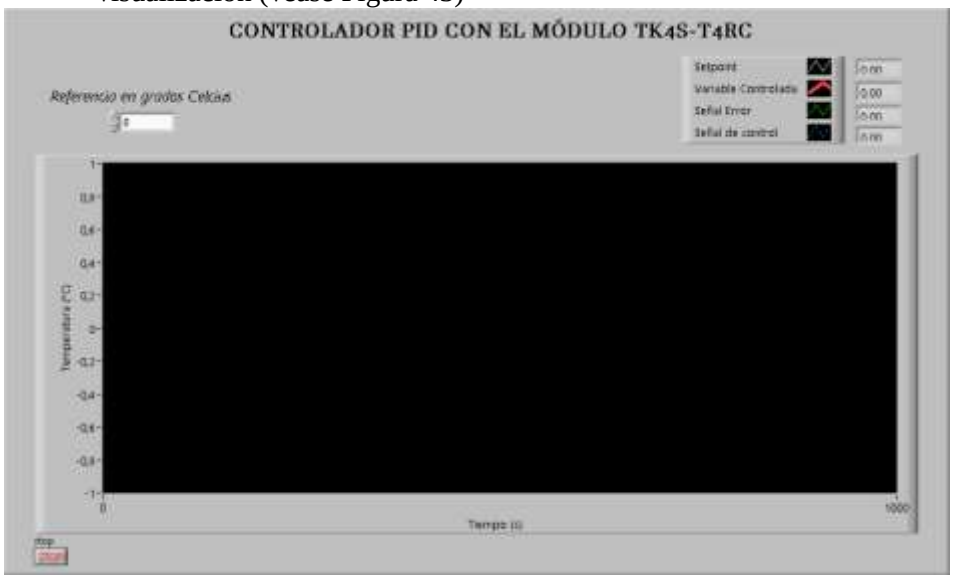


Figura 43. Interfaz grafica

3. Cableado de dispositivos

- El esquema de conexión se muestra a continuación (véase Figura 44).

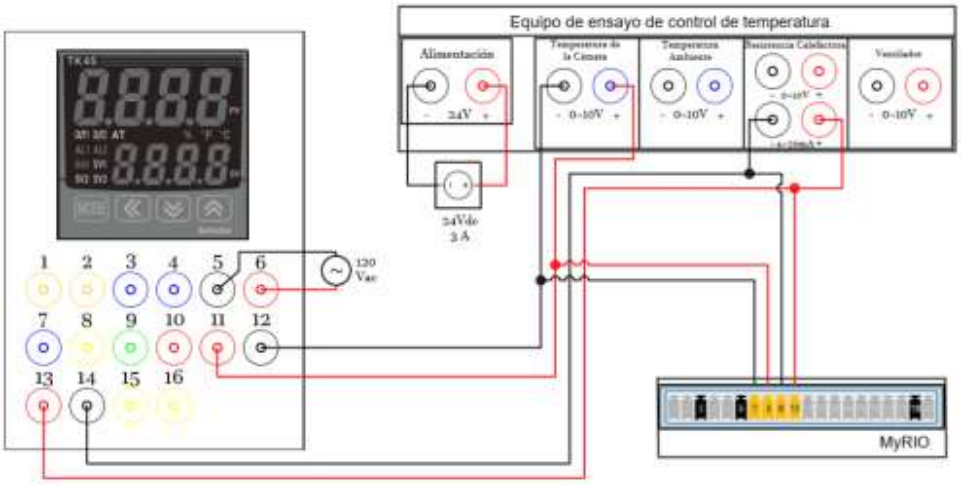


Figura 44. Diagrama de conexión

- Al conectar Ni myRIO al PC aparecerá la ventana mostrada en la figura 45, debemos seleccionar en "Do Nothing" (No hacer nada).
- La dirección IP mostrada será utilizada más adelante.

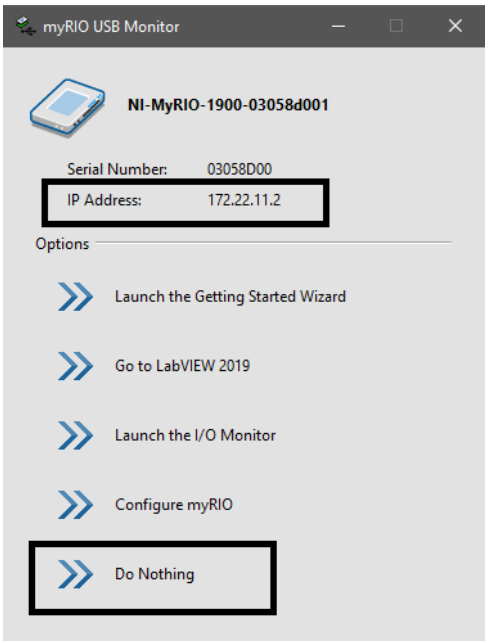


Figura 45. Ventana de inicio de NI myRIO

- En el explorador del proyecto, damos clic derecho en “RT myRIO target 0.0.0.0” y seleccionamos “Properties” (véase Figura 46).

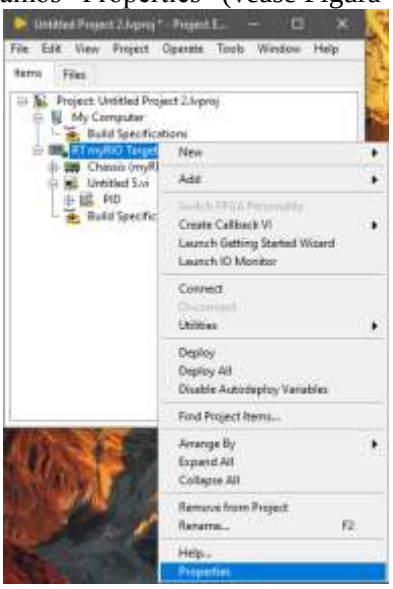


Figura 46. Propiedades de NI myRIO

- En la sección “General” cambiamos la dirección IP con la mostrada anteriormente y damos clic en “OK” (véase Figura 47).

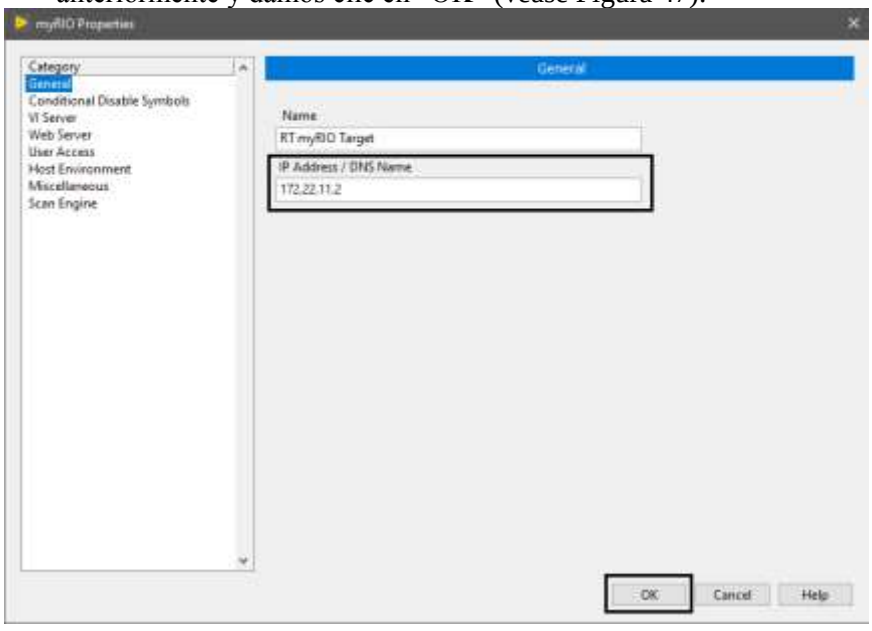


Figura 47. Asignar dirección IP

- Ejecutamos el código cuando se desee capturar las señales (véase Figura 48).

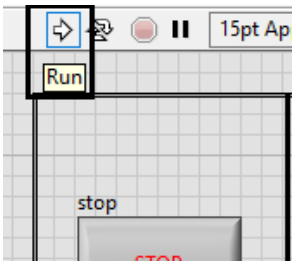


Figura 48. Ejecutar código

- Para acceder al archivo de datos debemos utilizar un navegador de internet, donde ingresaremos la dirección IP de la NI myRIO seguido de "/files" (véase Figura 49).

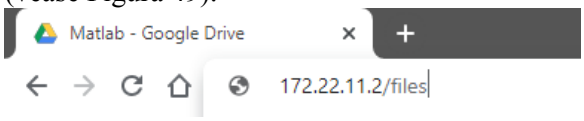


Figura 49. Ingresar dirección IP

- En el campo de usuario ingresamos "admin" y dejamos el campo de contraseña en blanco (véase Figura 50).

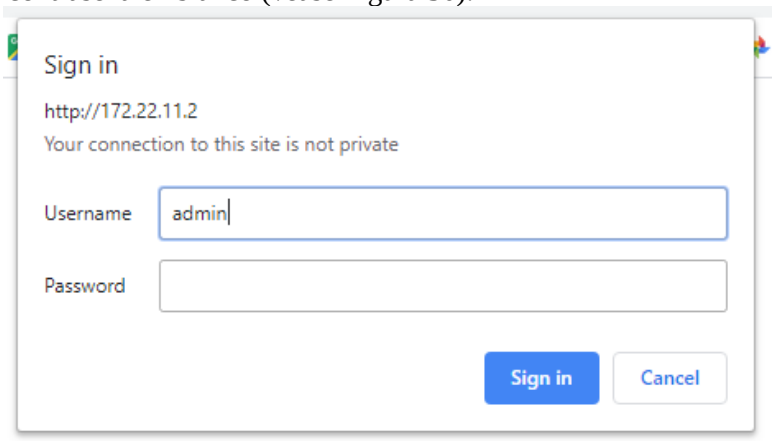


Figura 50. Ingresar credenciales

- Accedemos al directorio seleccionado anteriormente y descargamos el archivo. En caso de que simplemente se muestre en el navegador debemos dar clic derecho sobre los datos y guardarlo como texto. (véase Figura 51)

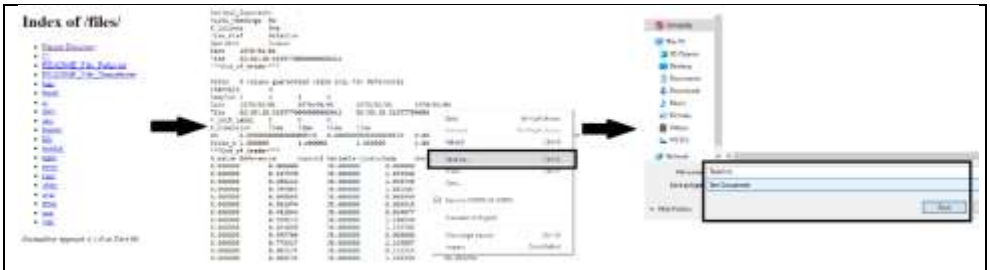


Figura 51. Guardar datos

4. Grafica de resultados en Matlab

- Abrimos Matlab r2017b y en hacemos clic en el botón “Import Data” (Importar datos). (véase Figura 52)

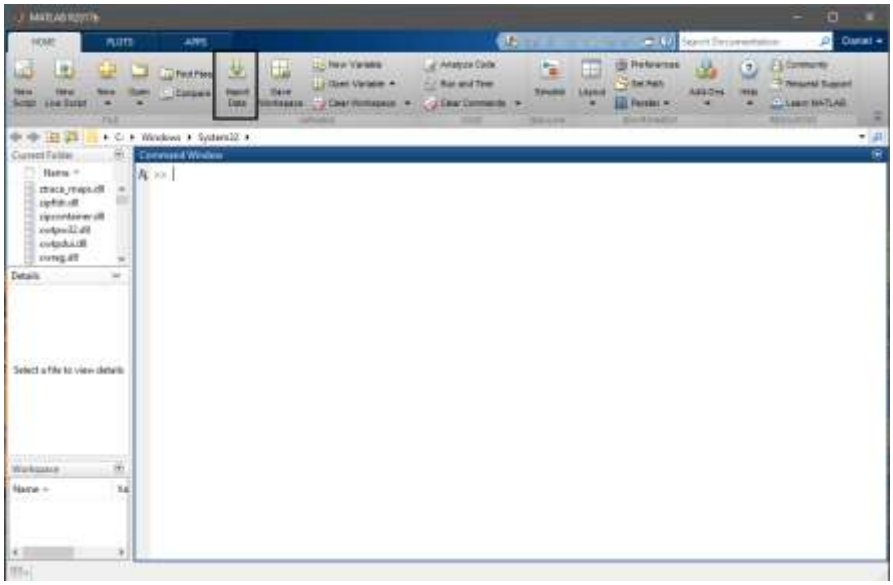


Figura 52. Importar datos

- En la ventana emergente nos ubicamos en la pestaña “Import” (Importar) y seleccionamos el delimitador seleccionado en LabVIEW en este caso “Tab” (Tabulador).
- Seleccionamos el rango de la matriz que contiene los datos junto con la fila en la que se encuentran los nombres de las variables
- Seleccionamos el tipo de salida como “Column vectors” (Vectores Columna) y finalmente damos clic en “Import Selection” (Importar selección) (véase figura 53).

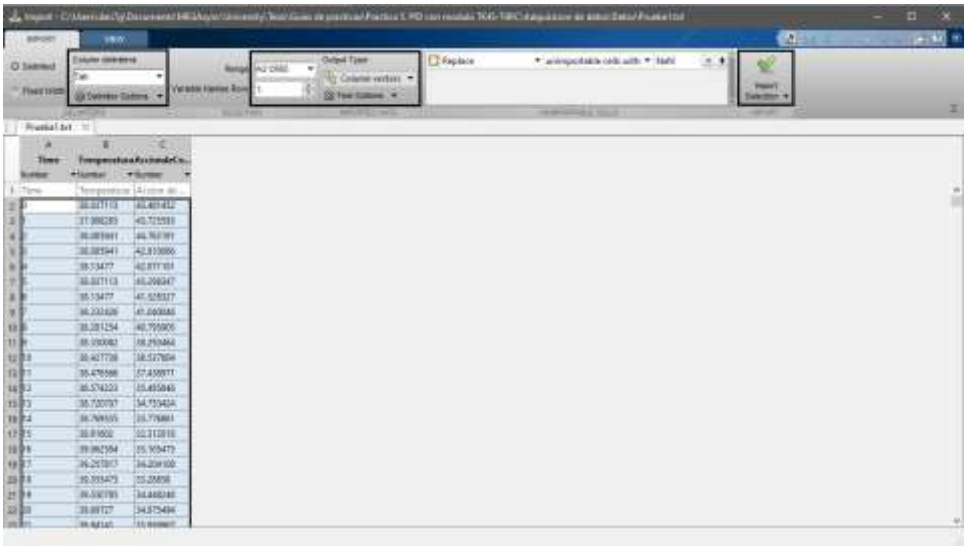


Figura 53. Configurar parámetros

- Las variables aparecerán en el workspace

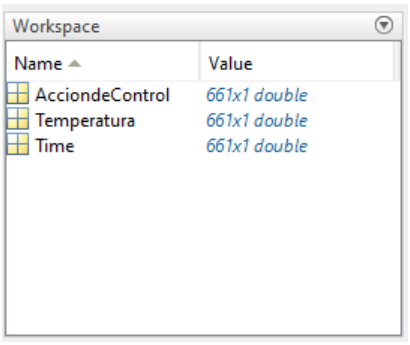


Figura 54. Variables en el workspace

- Con los datos importados podemos graficar el comportamiento del sistema con controlador, en la Figura 55.

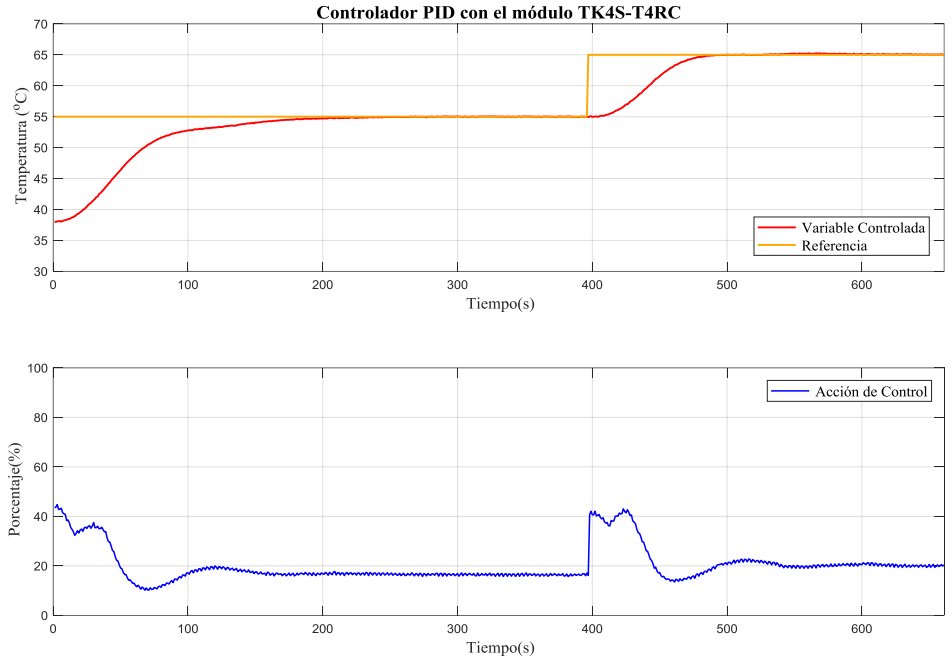


Figura 55. Graficas del comportamiento del sistema

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

APÉNDICES

- **GUÍA DE PRÁCTICA 6: DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR EN ESPACIO DE ESTADOS UTILIZANDO UNA NI myDAQ Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA**

FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: Ingeniería Electrónica		**ASIGNATURA**: Teoría de control
NRO. PRÁCTICA:	6	**TÍTULO PRÁCTICA**: **DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR EN ESPACIO DE ESTADOS UTILIZANDO UNA NI MYDAQ Y EL EQUIPO DE ENSAYOS DE CONTROL DE TEMPERATURA**

OBJETIVO

1. Proporcionar las herramientas necesarias para que el estudiante pueda implementar un controlador en el software Matlab r2017b que permita el manejo de un sistema térmico.
2. Conocer las características generales del equipo NI-DAQ.
3. Conocer la estructura de un sistema SISO.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

a) Creación de un proyecto y familiarización con el software Matlab r2017b
b) Manejo básico de Simulink
c) Respuesta de un sistema de segundo orden

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Computadora S.O. Windows 7 o superior, 32 o 64 bits	1	-	-
Software Matlab r2017b	1	MathWorks	-
NI-DAQ	1	National Instruments	-

Equipo de ensayos de control de temperatura	1	CTI Lab	-
Fuente DC 24 volts 3.2 Amps	1	Siemens	-

Tabla 1. Equipos, instrumentos y software

3. Exposición

***NI-DAQ*:** es un equipo multifunción que contiene entradas analógicas, digitales y dos contadores de 32 bits. Cuenta con NI signal streaming para transferencia de datos bidireccional a alta velocidad mediante USB. Este dispositivo es ideal para aplicaciones de control, monitoreo de campo, adquisición de datos en general y aplicaciones académicas.

Figura 1. Modulo NI-DAQ [18]

***Equipo de ensayos de control de temperatura**:* este dispositivo permite realizar ensayos de control de temperatura con picos de hasta 55 °C, equipado con una

niquelina fabricada con alambre de nicromo, dos ventiladores 3010 para perturbaciones, sensores de temperatura tipo RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C.

El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.

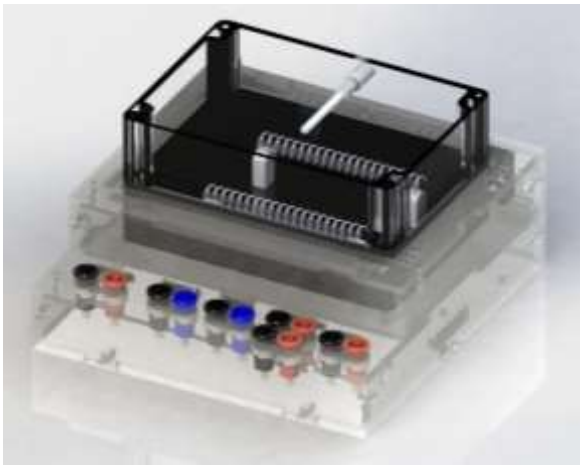


Figura 2. Equipo de ensayo de control de temperatura

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. **Preparar dispositivos**

- Conectamos el dispositivo NI-DAQ al computador, el led incorporado comenzara a parpadear. Esta acción se debe realizar previo a abrir el programa, de otra manera Matlab no reconocerá al dispositivo.
- Abrimos Matlab r2017b

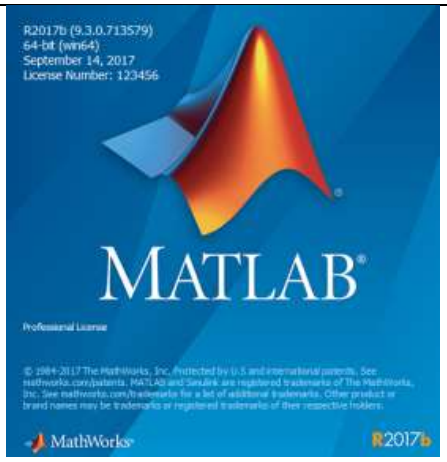


Figura 3. Matlab

- En la Ventana de Comandos (Command Window) escribimos "sldrtkernel -install" (véase Figura 4)

Figura 4.Instalando kernel para Real Time de Simulink

- Colocar "y" en el mensaje (véase Figura 5).

```
You are going to install the Simulink Desktop Real-Time kernel.
Do you want to proceed? [y] :
```

Figura 5. Mensaje para proceder con la instalación del Real-Time kernel

- Debemos observar el siguiente mensaje en la ventana de comandos (véase figura 6)

```
The Simulink Desktop Real-Time kernel has been successfully installed.
```

Figura 6. Mensaje de instalación exitosa

2. Análisis de controlabilidad y observabilidad

- En un nuevo script ingresamos el modelo mostrado en la ecuación 1, realizamos la aproximación de Pade y el modelo resultante lo pasamos a espacio de estados en la forma canónica controlable. El código se muestra a continuación.

$$G = \frac{0.00004936}{s^3 + 0.01448 \cdot s^2 + 0.00637 \cdot s + 0.00001842} \quad [1]$$

```
clear all
clc
% Planta Camara
s=tf('s');
G=(4.936e-5*exp(-17*s))/(s^3+0.01448*s^2+0.00637*s+1.842e-5);
%Aproximacion de Pade, para eliminar el exponencial
G_p=pade(G,1);
[num,den]=tfdata(G_p,'v');
%Obtener matrices del sistema en espacio de estados
[A,B,C,D]=tf2ss(num,den);
%Forma canonica controlable
A=rot90(A,2);
B=rot90(B,2);
C=rot90(C,2);
m=ss(A,B,C,D);
```

- Realizamos el análisis de observabilidad y controlabilidad. El código se muestra a continuación.

```
% Analisis de controlabilidad y observabilidad
Controlab=ctrb(A,B);
if rank(Controlab)==size(A,1)
    disp('El sistema es controlable')
else
    disp('El sistema NO es controlable')
end
Observab=obsv(A,C);
if rank(Observab)==size(A,1)
    disp('El sistema es observable')
else
    disp('El sistema NO es observable')
end
```

- La respuesta de Matlab luego de ejecutar el código se muestra en la figura 7.

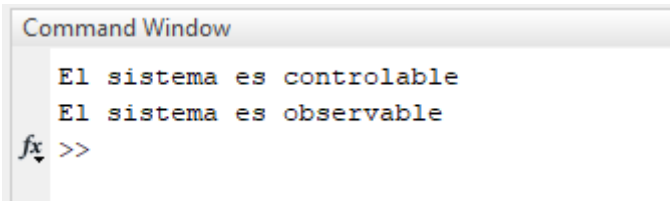


Figura 7. Sistema observable y controlable

3. Diseño del controlador

- La técnica de diseño empieza con la determinación de los polos en lazo cerrado deseados a partir de los requerimientos establecidos por el diseñador tales como velocidad, razón de amortiguamiento, o ancho de banda, al igual que los requisitos en estado estacionario. Para este ejemplo hemos dispuesto lo siguiente:

  - Tiempo de establecimiento

  $$ts = 650\ s$$

  - Sobre impulso máximo

  $$Mp = 8\%$$

  Para calcular el coeficiente de amortiguamiento utilizamos la ecuación 2.

  $$Mp = e^{\frac{-\pi\xi}{\sqrt{1-\xi^2}}} \qquad [2]$$

  $$0.08 = e^{\frac{-\pi\xi}{\sqrt{1-\xi^2}}}$$

  $$\xi = 0.62657$$

  Para calcular la frecuencia natural utilizamos la ecuación 3.

  $$ts = \frac{4}{\xi w_n} \qquad [3]$$

  $$650 = \frac{4}{0.6265 w_n}$$

  $$w_n = 0.009822$$

- Los polos dominantes del sistema se obtienen al resolver la ecuación 4.

  $$s^2 + 2\xi w_n s + w_n^2 = 0 \qquad [4]$$

  $$s^2 + 0.0123s + 0.00009647 = 0$$

  $$s_{1,2} = -0.00615 \pm 0.007658j$$

- A pesar de que el sistema es de tercer orden, es suficiente con calcular los dos polos más lentos ya que estos regirán el comportamiento del sistema, en este caso dos complejos conjugados. El resto de los polos deberán estar más alejados del eje jw, para mermar su influencia.

- Para asegurar las condiciones en estado estacionario se ha incrementado un integrador. Los polos no dominantes del sistema se muestran a continuación.

$$s_{3,4,5} = -0.123$$

- Debido a que las variables de estado del sistema no están disponibles, es necesario la implementación de un observador de estados de orden completo. Por norma general los polos del observador deben ser de dos a cinco veces más rápidos que los del controlador. Para este ejemplo se han escogido los siguientes polos:

$$s_{1,2,3} = -0.3075$$

- El código necesario para el cálculo de las matrices de ganancias, con el integrador embebido, se muestra a continuación.

```
%%
% Integrador
Ai=[A zeros(4,1);-C 0];
Bi=[B;0];

% Matriz de ganancias del controlador
pC=[-0.00615+0.007658i -0.00615-0.007658i -0.123 -0.123 -0.123];
k=acker(Ai,Bi,pC)

% Matriz de ganancias del observador
pO=[-0.3075 -0.3075 -0.3075 -0.3075];
ke=acker(A',C',pO)'
```

- Las matrices resultantes son:

$$k = [-2.0989 \cdot 10^{-6} \quad -0.0008 \quad -0.0071 \quad -0.0829 \quad -7.5683 \cdot 10^{-5}]$$

$$ke = \begin{bmatrix} -4.1783 \cdot 10^{3} \\ -306.6581 \\ -11.8833 \\ 7.1119 \end{bmatrix}$$

4. **Simulación del controlador en Simulink**

- Para comprobar el funcionamiento del controlador es muy recomendable simular el sistema, el esquema utilizado se muestra en la figura 8.

APÉNDICES

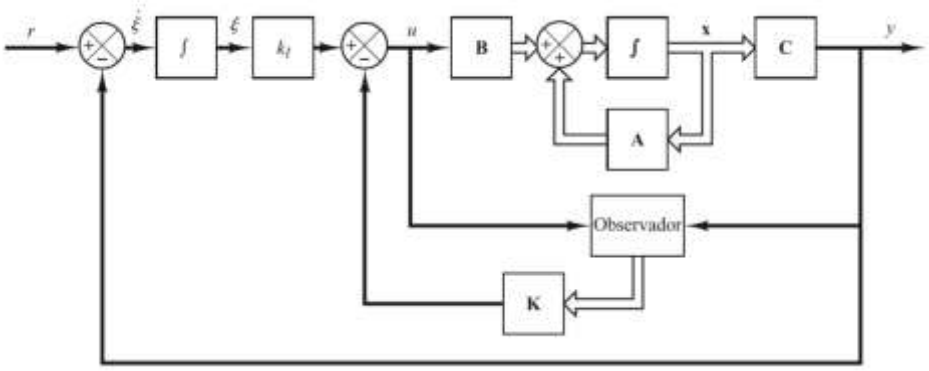


Figura 8. Esquema de servosistema tipo 1

- El observador de estados se muestra en la figura 9.

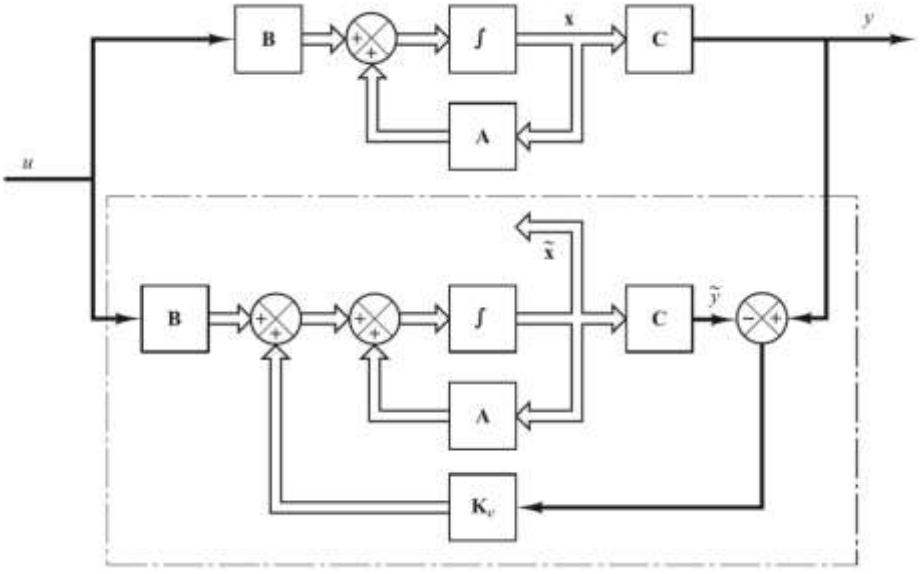


Figura 9. Observador de estados

- El esquema en Simulink se muestra en la figura 10.

APÉNDICES

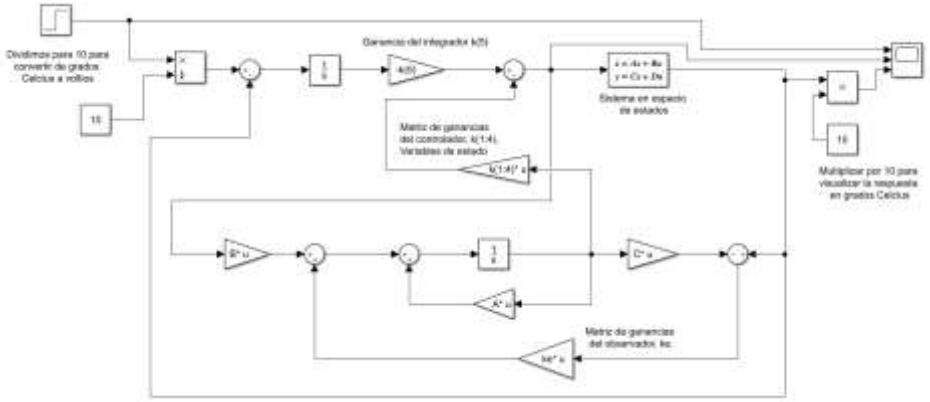


Figura 10. Modelos de servosistema tipo 1 en Simulink

- Para configurar un bloque de ganancia damos doble clic, ingresamos la matriz de ganancia correspondiente, seleccionamos el tipo de multiplicación y pulsamos ok. La matriz de ganancia debe estar definida en el workspace. (véase figura 11)

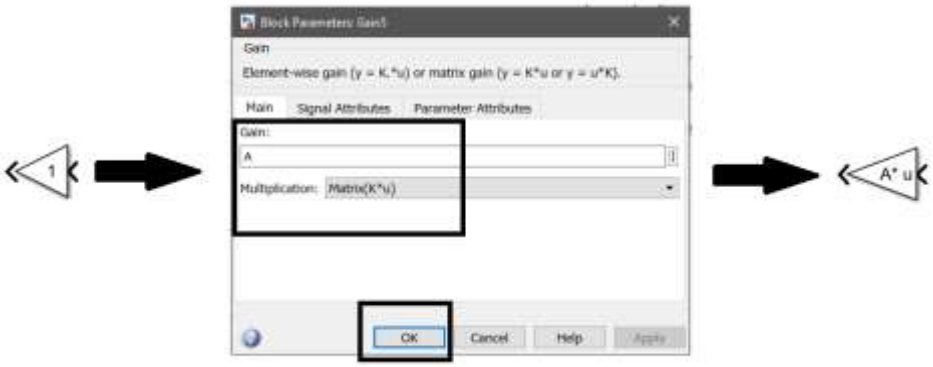


Figura 11. Agregar y configurar un bloque de ganancia

- Repetimos el proceso con todos los bloques de ganancia.
- Para configurar el sistema en espacio de estados, damos doble clic en el bloque e ingresamos las matrices A, B, C y D. Las matrices deben estar definidas en el workspace. (véase figura 12)

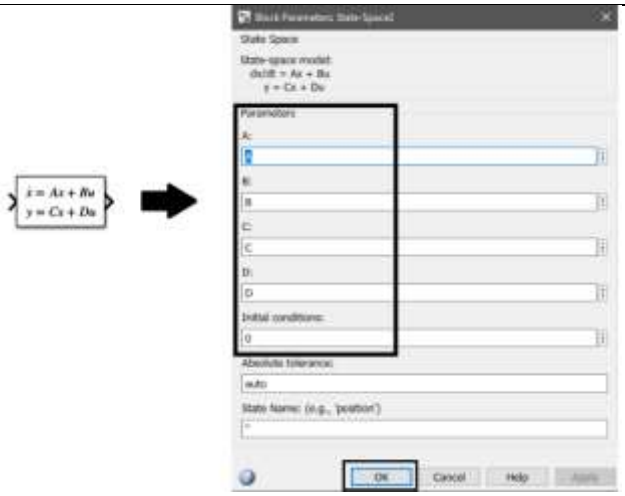


Figura 12. Configurar parámetros de bloque en espacio de estados

- Establecemos el tiempo de simulación en 1500 y ejecutamos el modelo. (véase figura 13)

Figura 13. Iniciar simulación

- Para visualizar la curva de respuesta damos doble clic en el bloque scope.

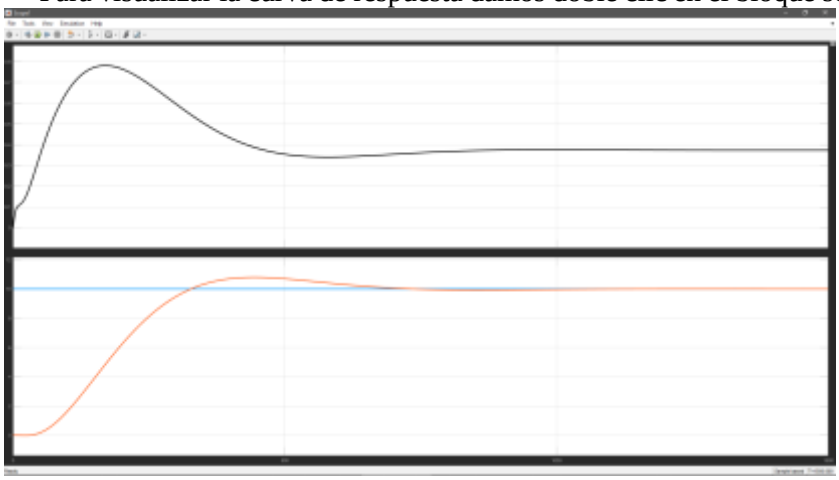


Figura 14. Respuesta simulada del sistema

5. Implementación del controlador

- Para implementar el controlador con el equipo de ensayos de control de temperatura debemos sustituir el bloque de espacio de estados por una entrada y una salida física. Estos bloques están dentro de la librería "Data Acquisition Toolbox" (véase figura 15)

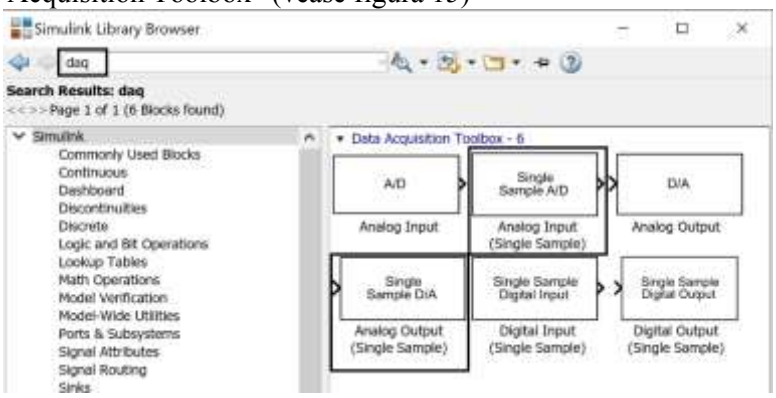


Figura 15.Anadir bloques de entrada y salida

- Dar doble clic en el bloque "Analog Input", en la ventana emergente damos clic en "Unselect All" (deseleccionar todas) (véase Figura 16)

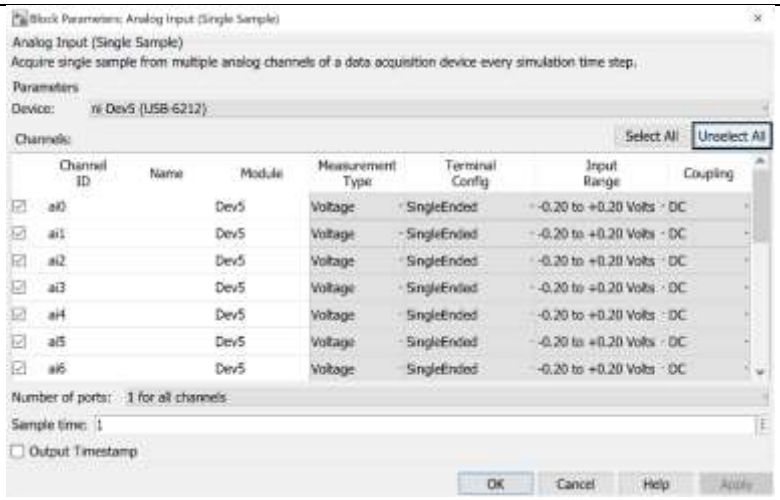


Figura 16137. Configuración del bloque de entradas analógico

- Seleccionamos la entrada utilizada, en este casi "ai1" en la casilla "Input Range" seleccionamos "-10 to +10 Volts" y en la casilla "Terminal Config" colocamos "Differential". Desplegamos el submenú "Number of ports" (número de puertos) y seleccionaos "1 per channel" (Uno por canal). Finalmente, en "Sample time" (Tiempo de muestreo) colocamos "1" y damos clic en "OK" (véase Figura 17)

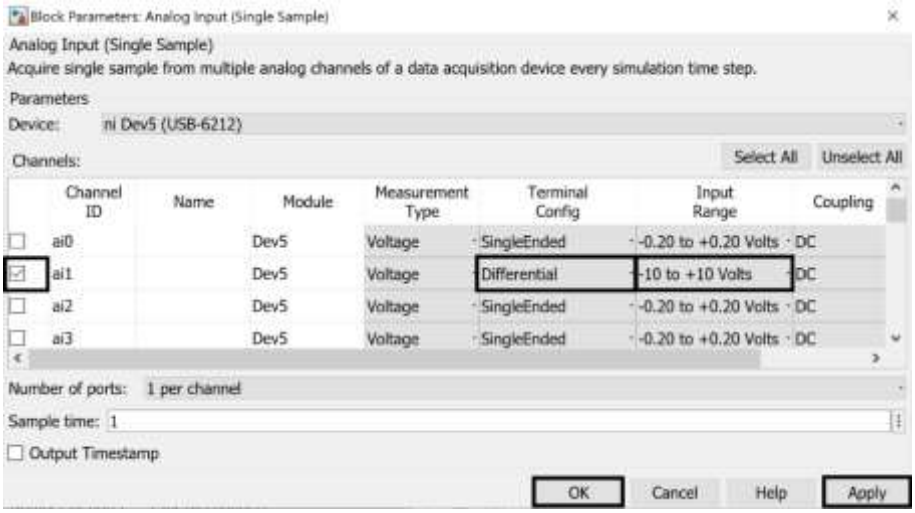


Figura 17. Configuración del bloque de entradas analógicas

- Damos doble clic en "Analog Output" y seleccionamos la salida utilizada en este caso "ao1". En "Input Range" seleccionamos "-10 to +10 Volts".

En “Number of ports” seleccionamos “1 per channel”, ingresamos “1” en el tiempo de muestreo y pulsamos “OK” (véase figura 18)

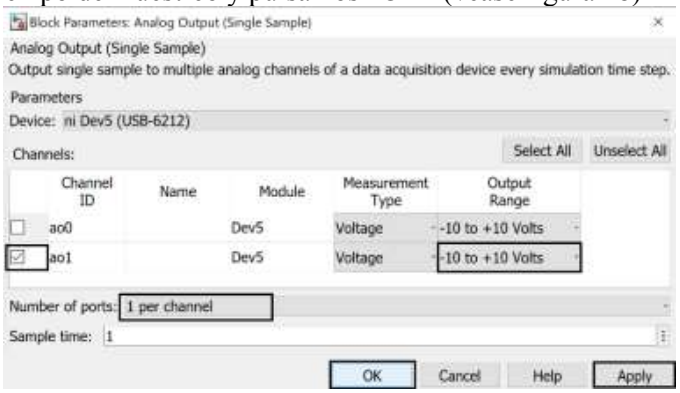


Figura 18. Configuración del bloque de salidas analógicas

- Añadimos un bloque de saturación para limitar la salida del controlador de 0 a 10volts. De otra forma Simulink marcará un error. (véase figura 19)

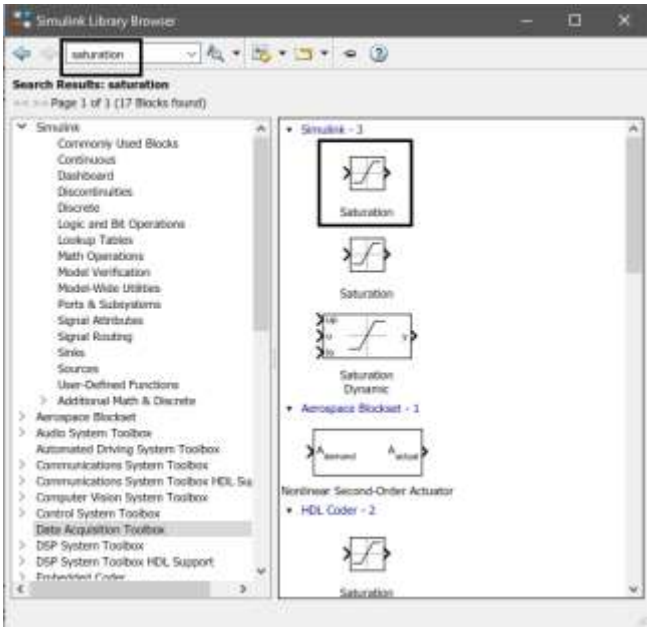


Figura 19. Añadir bloque de saturación

- Una vez con el bloque ubicado damos doble clic en el bloque e ingresamos el límite superior e inferior y pulsamos “OK” (véase Figura 20)

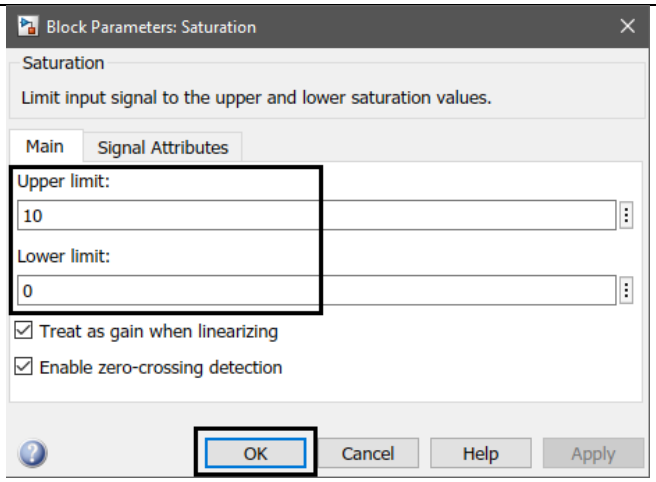


Figura 20. Configurar bloque de saturación

- Añadimos la librería real time synchronization. (véase figura 21)

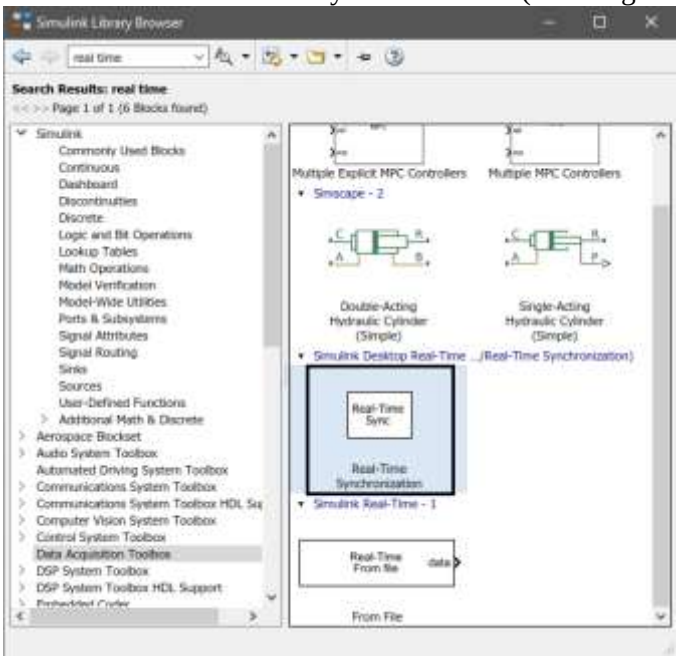


Figura 21. Añadir bloque "Real-Time Synchronization"

- Damos doble clic en el bloque y modificamos los parámetros como se muestra en la Figura 22

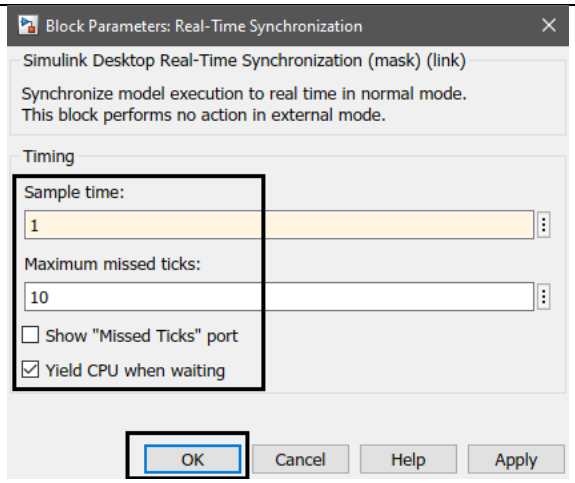


Figura 22. Configurar bloque Real-Time Synchronization

- Añadimos el bloque “To Workspace” para recolectar los datos de las señales. Debemos conectar cada bloque a una señal de interés, en este caso: señal de control, respuesta del sistema y referencia.

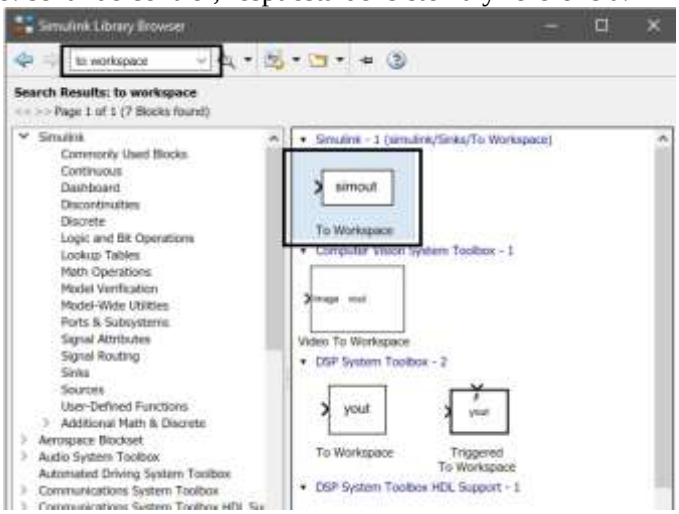


Figura 23. Anadir bloque "To Workspace"

- Damos doble clic en los bloques “To workspace” para modificar el nombre de la variable, el formato y el tiempo de muestreo. Repetimos el proceso para las demás variables (véase Figura 24).

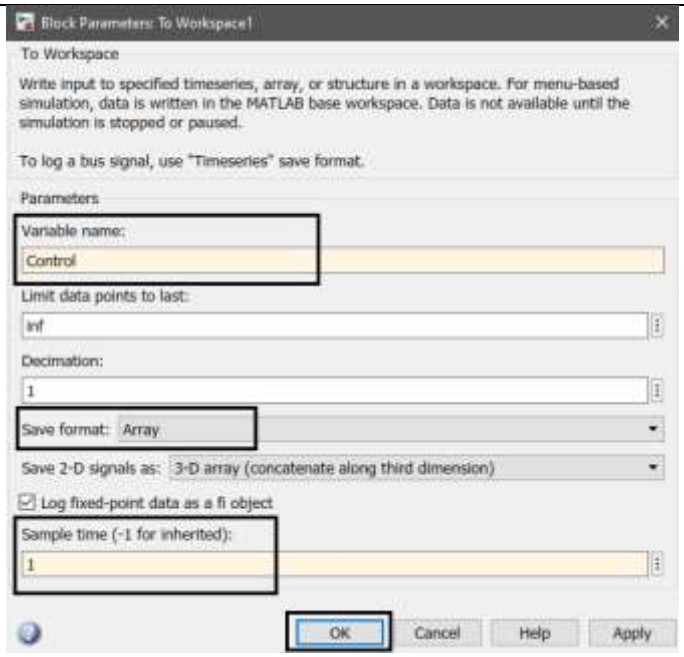


Figura 24. Configurar bloque "To Workspace"

- El esquema final se muestra en la figura 25

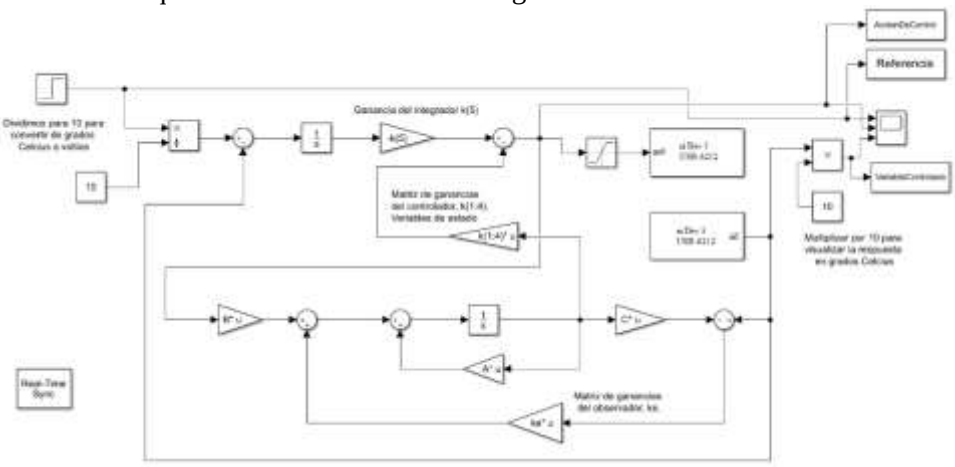


Figura 25. Esquema completo del controlador en Simulink

6. Cableado de dispositivos

- El esquema de conexión se muestra a continuación (véase Figura 26).

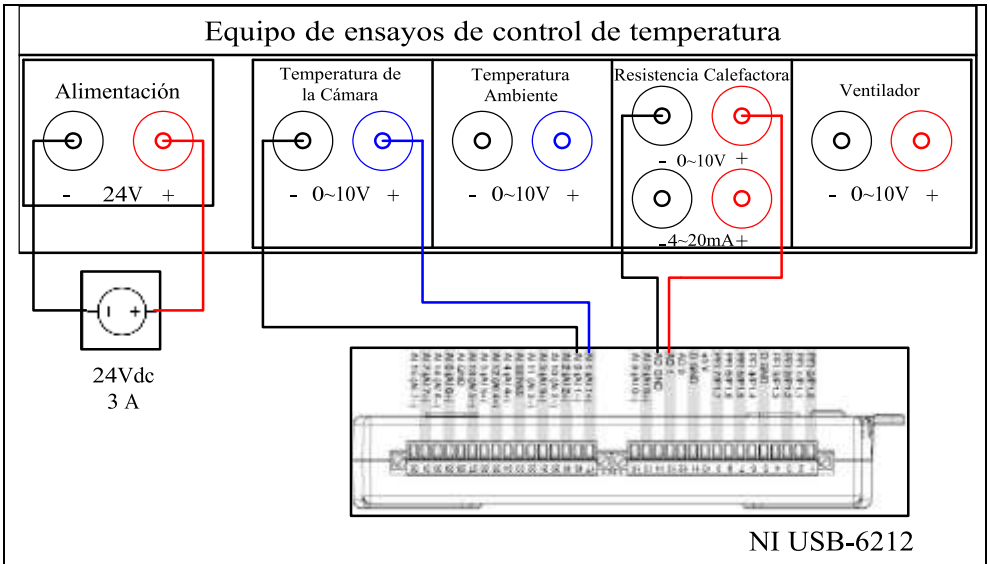


Figura 26. Diagrama de conexión

- Establecemos el tiempo de simulación en 1500 y ejecutamos el modelo. (véase figura 27)

Figura 27. Iniciar proceso

- Podemos observar el comportamiento del controlador en tiempo real, debemos dar doble clic al scope.

7. Grafica de resultados en Matlab

- En el workspace encontraremos los datos recolectados del ensayo, por lo que no es necesario importarlos.
- La respuesta del sistema se muestra en la figura 28.

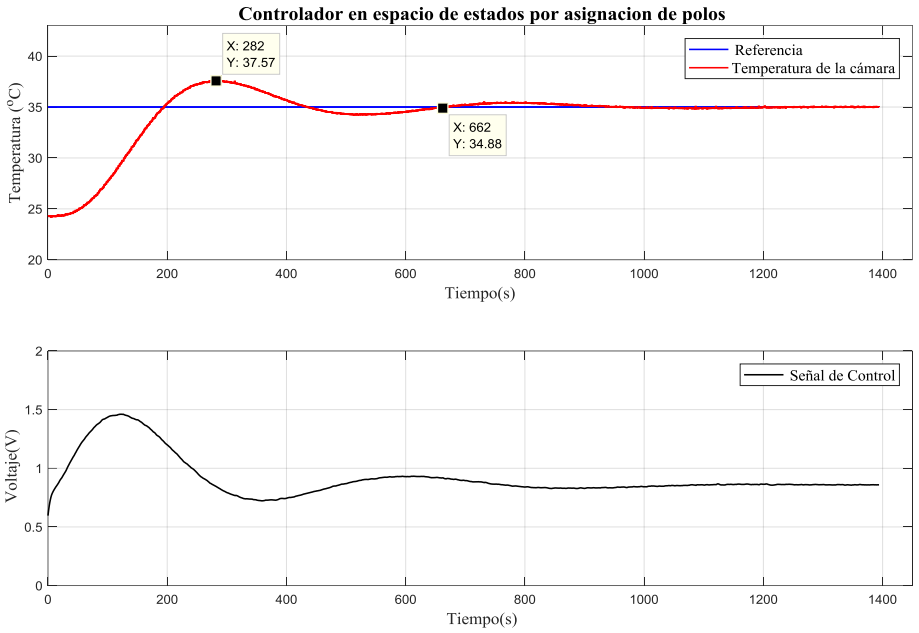


Figura 28. Respuesta del sistema

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

APÉNDICES

- **GUÍA DE PRÁCTICA 7: CONTROLADOR PREDICTIVO BASADO EN EL MODELO (MPC) IMPLEMENTADO EN EL EQUIPO DE ENSAYO DE CONTROL DE TEMPERATURA MEDIANTE SIMULINK Y NI myDAQ.**

FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: INGENIERÍA ELECTRÓNICA	**ASIGNATURA**: TEORÍA DE CONTROL

NRO. PRÁCTICA:	7	**TÍTULO PRÁCTICA**: **CONTROLADOR PREDICTIVO BASADO EN EL MODELO (MPC) IMPLEMENTADO EN EL EQUIPO DE ENSAYO DE CONTROL DE TEMPERATURA MEDIANTE SIMULINK Y NI MYDAQ.**

OBJETIVO

1. Proporcionar las herramientas necesarias para que el estudiante pueda crear un proyecto en el software LabVIEW 2019 que permita el manejo de un equipo de ensayo de control de temperatura mediante el dispositivo NI-DAQ.
2. Conocer las características generales del equipo de ensayo de control de temperatura, su rango de trabajo y el dispositivo NI-DAQ.
3. Conocer la herramienta Simulink de Matlab para la adquisición de datos con el dispositivo NI-DAQ.

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

a) Controlador Predictivo basado en el Modelo (MPC)

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Computadora S.O. Windows 10, 64 bits	1	-	-
Software Matlab 2017	1	-	-
NI DAQ USB-6212	1	Nacional Instrument	-

Equipo de ensayo de control de temperatura	1	CTILab	-
Cables bananas	Varios	-	-

Tabla 1. Equipos, instrumentos y software

3. Exposición

NI DAQ USB-6212: es un equipo multifunción que contiene entradas y salidas analógicas, digitales y dos contadores de 32 bits. Cuenta con NI signal streaming para transferencia de datos bidireccional a alta velocidad mediante USB. Este dispositivo es ideal para aplicaciones de control, monitoreo de campo, adquisición de datos en general y aplicaciones académicas.

Figura 1. NI DAQ USB-6212

Equipo de ensayos de temperatura*:* este dispositivo permite realizar ensayos de control de temperatura con

picos de hasta 80 °C, equipado con una niquelina fabricada con alambre de nicromo, dos ventiladores 3010 para perturbaciones, sensores de temperatura tipo RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C.

El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.

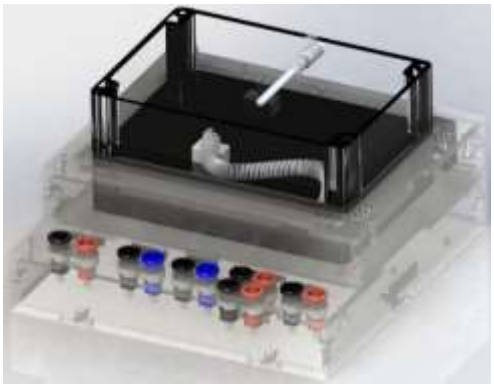


Figura 2. Equipo de ensayos de control de temperatura

ACTIVIDADES POR DESARROLLAR

1. **Instalación de drivers**
   - Descargar los drivers del dispositivo NI-DAQ USB-6212 del siguiente link https://www.ni.com/es-cr/support/downloads/drivers/download.ni-daqmx.html#333268
2. **Agregar herramientas necesarias del software Matlab para la simulación en tiempo real**
   - Abrir el software Matlab 2017b y conectar la NI-DAQ en el puerto USB del computador

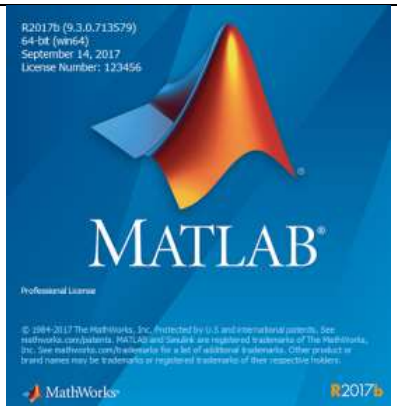


Figura 3. Matlab

- En la Ventana de Comandos (Command Windows) escribir "`sldrtkernel -install`"' (véase Figura 4)

Figura 4. Instalando Kernel para Real Time de Simulik

- Colocar "y" en el mensaje (véase Figura 5).

```
You are going to install the Simulink Desktop Real-Time kernel.
Do you want to proceed? [y] :
```

Figura 5. Mensaje para proceder con la instalación del Real-Time Kernel

```
The Simulink Desktop Real-Time kernel has been successfully installe
```

Figura 6. Mensaje de instalación exitosa

3. Diseño del controlador MPC

- Se crea un nuevo proyecto escogiendo "New Script" (Nuevo Script)

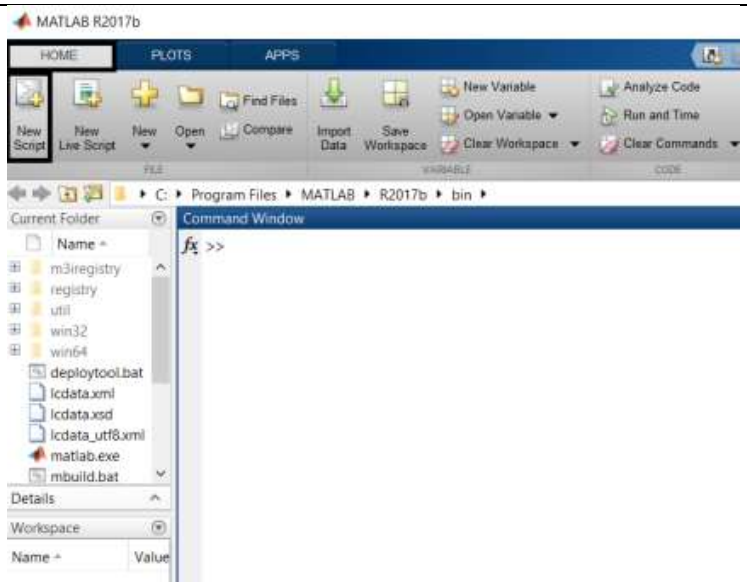


Figura 7. Creando "New Script"

- El siguiente código permite convertir la planta del sistema en espacio de estados y discretizarla. El tiempo de muestreo es 1 segundo.

```
% Planta Camara
Ts=1;
s=tf('s');
G2=(0.0002453*s+4.212e-7)/(s^3+0.02612*s^2+0.0001594*s+1.529e-7);
% Obteniendo el modelo en espacio de estados
[num,den]=tfdata(G2,'v');
[A,B,C,D]=tf2ss(num,den);
%Forma canonica controlable
A=rot90(A,2);
B=rot90(B,2);
C=rot90(C,2);
%Sistema en espacio de estados
CS=ss(A,B,C,D);
%Se discretiza el sistema
C3D=c2d(CS,Ts);
```

- En la Ventana de Comandos (Command Windows) escribir "mpcDesigner"' (véase Figura 8)

Figura 8. Abriendo MPC Designer

Se abrirá la ventana "MPC Designer"

- En la pestaña "MPC DESIGNER" dar clic en "MPC Structure" (véase Figura 9)

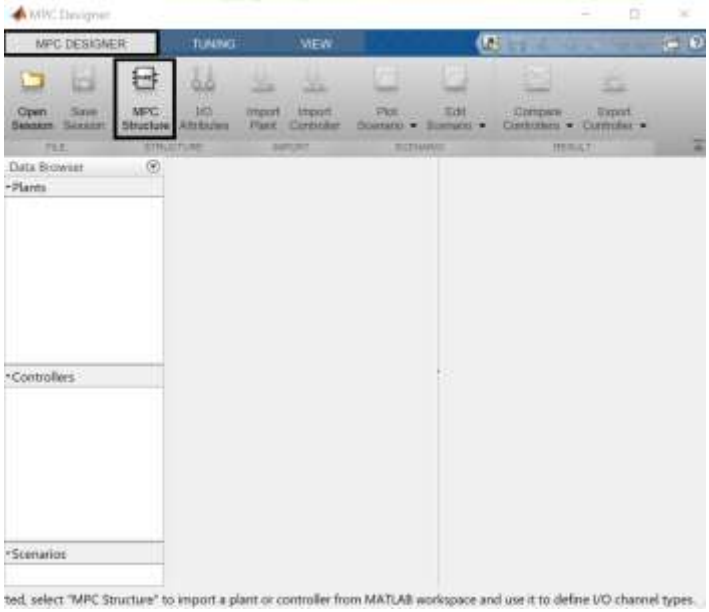


Figura 9. MPC Designer

- Se abrirá la ventana "Define MPC Structure By Importing", seleccionar el nombre de la planta obtenido anteriormente en espacio de estados en este caso "C3D", en "Specify MPC controller Sample time" colocar "1". La variable manipulada (MV) y salida medida (MO) se deja el valor por defecto (1). Clic en "Define and Import" (véase Figura 10).

APÉNDICES

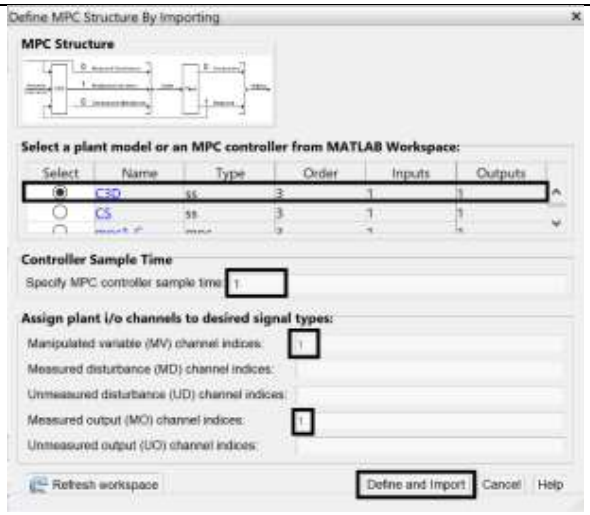


Figura 10. Importando planta a controlar

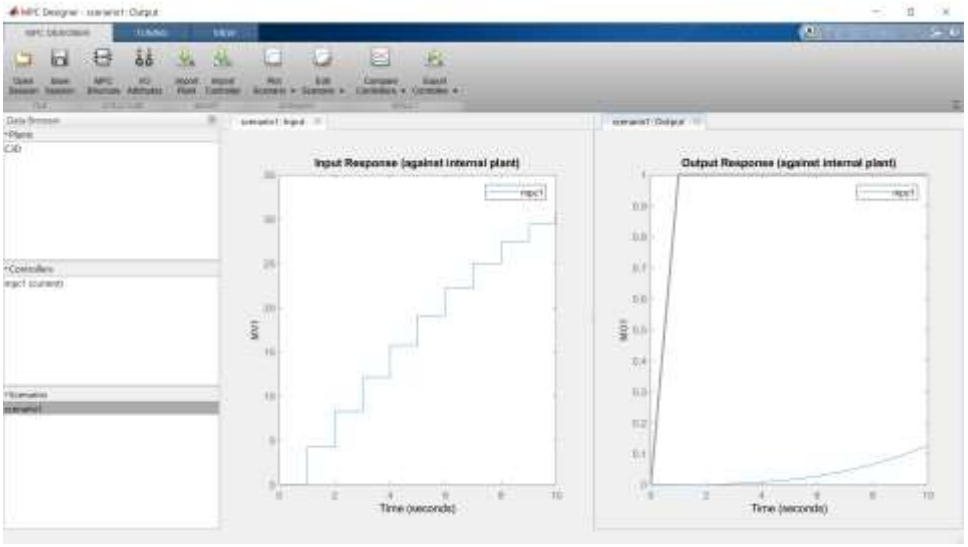


Figura 11. Planta ingresada

- En la pestaña “MPC DESIGNER” seleccionar el símbolo “▼” de “Edit Scenario”, clic en “scenario1”

Figura 12. Editando escenario

- En la ventana "Simulation scenario" coloca al menos 500s en la duración de la simulación, clic en "Apply", clic en "OK" (véase Figura 13). En esta ventana es posible cambiar el tamaño del escalón y su tiempo de inicio.

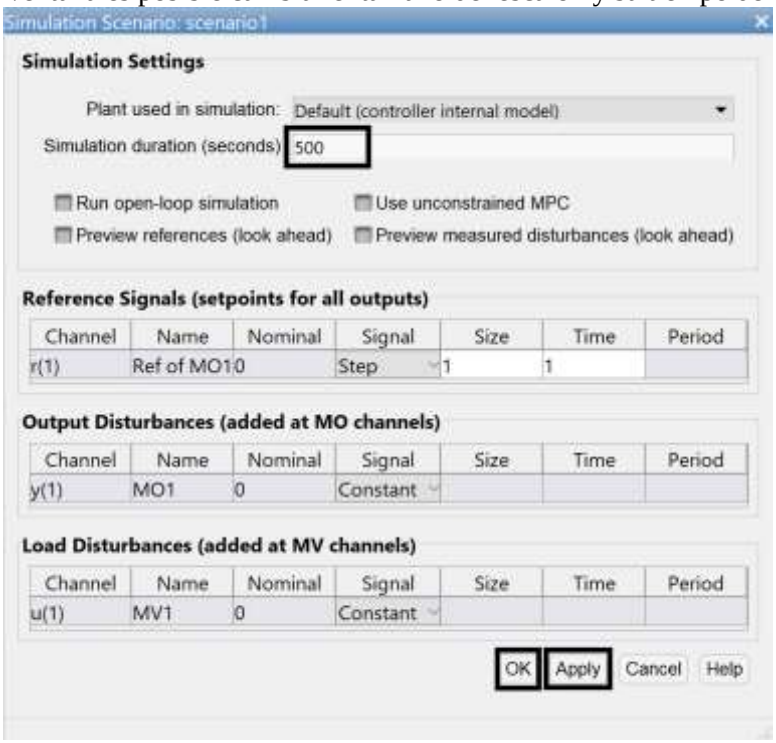


Figura 12. Editando tiempo de simulación

- En la pestaña "MPC DESIGNER", en la seccion "STRUCTURES", seleccionar "I/O Atributes". En la ventana emergente colocar "0.5" en el valor nominal de entrada "u(1)" y "2.6" en el valor nominal de salida "y(1)". Seleccionar "Apply" y "OK" (vease Figura 14).

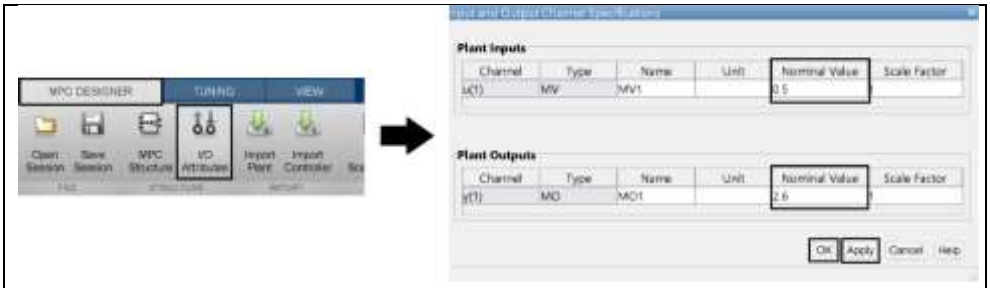


Figura 14. Colocando condiciones iniciales del controlador

- En la pestaña "TUNING", se puede cambiar los parámetros de la sección "HORIZON" para estabilizar el sistema (véase Figura 15).

Figura 15. Modificando parámetros para estabilidad del sistema

- En la pestaña "TUNING" dar clic en "Constraints" (véase Figura 16)

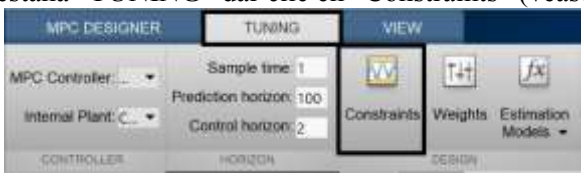


Figura 16. Constraints

- En la ventana emergente "Constraints" ubicarse en "Input Constraints", colocar "0" y "10" en "Min" y "Max" respectivamente, clic en "Apply" y "OK" (véase Figura 17).

Nota: Los parámetros RateMin y RateMax pueden modificarse para mejorar la respuesta del sistema.

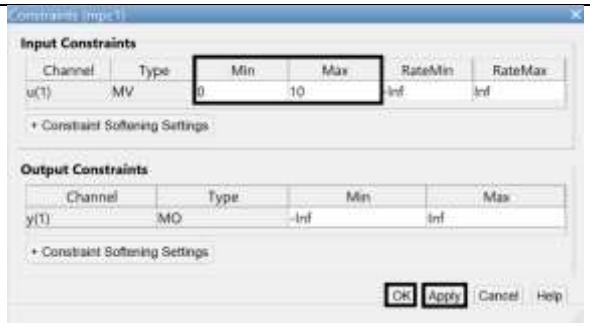


Figura 17. Constraints

- Se puede mejorar la respuesta del sistema modificando los de "PERFORMANCE TUNING" (véase Figura 18).

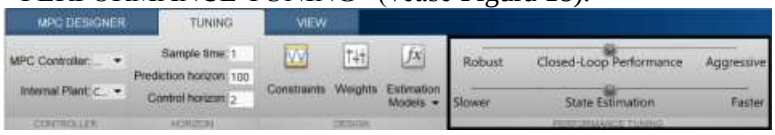


Figura 18. Performance Tuning

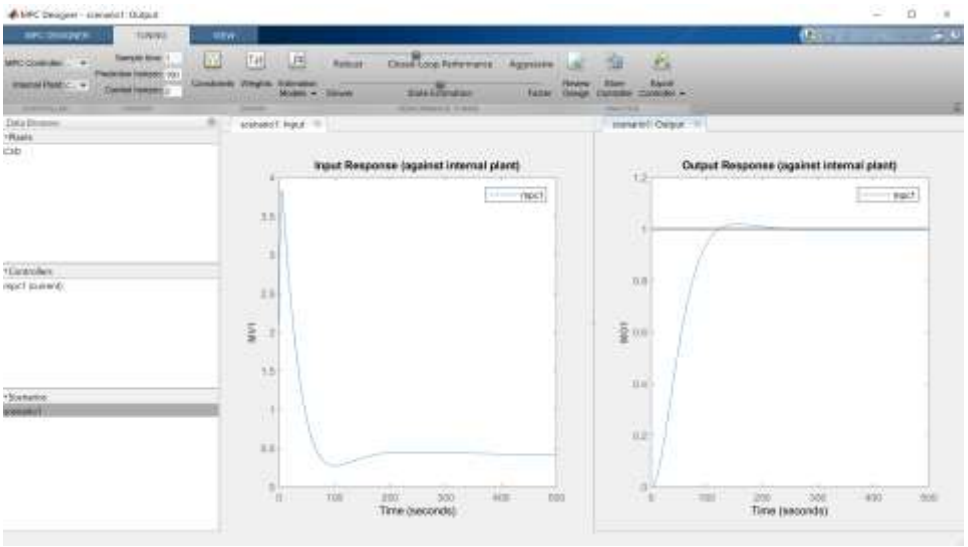


Figura 19. Simulación del modelo estabilizado

- En la pestaña "TUNING" en la sección "ANALYSIS", seleccionar el símbolo "▼" de "Export Controller", clic en "Generate a Simulink Model" (véase Figura 20). Se abrirá la ventana mostrada en la Figura 21.

Figura 20. Generando Modelo de Simulink

4. Conexión con los bloques DAQ

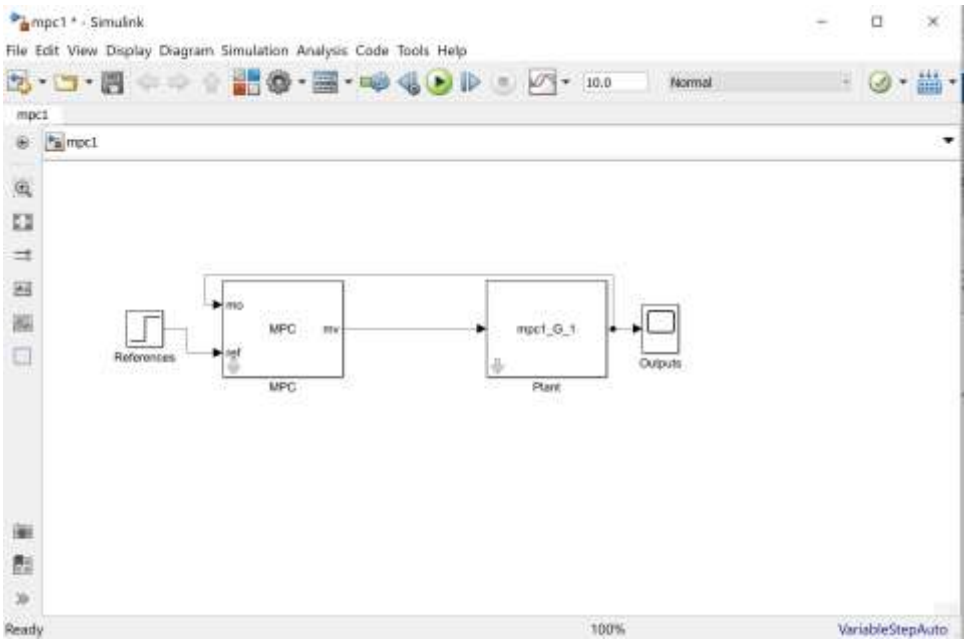


Figura 21. Modelo Creado con MPC

- Dar clic en "Library Browser" ubicado en la parte superior izquierda del modelo creado.

Figura 22. Abriendo librerías.

- En el buscador escribir “daq” y arrastrar al modelo los bloque “Analog Input (Single Sample)” y “Analog Output (Single Sample)” (véase Figura 23). Colocar el tiempo de muestreo de 1 segundo.

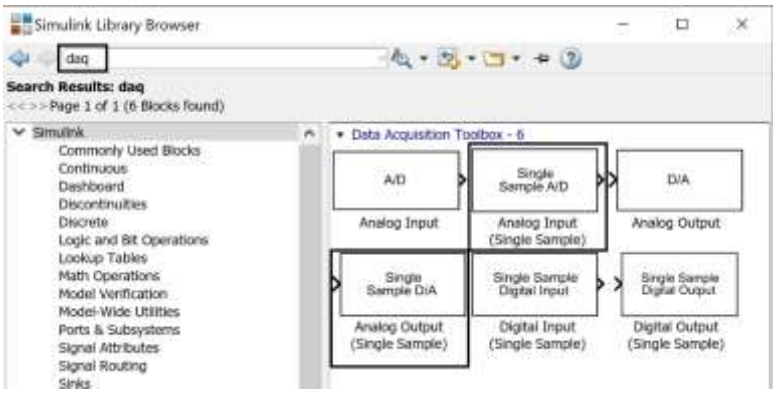


Figura 23. Agregando bloques para la DAQ

- Dar doble clic en el bloque “Analog Input”, en la ventana emergente dar clic en “Unselect All” para desmarcar todas las entradas (véase Figura 24).

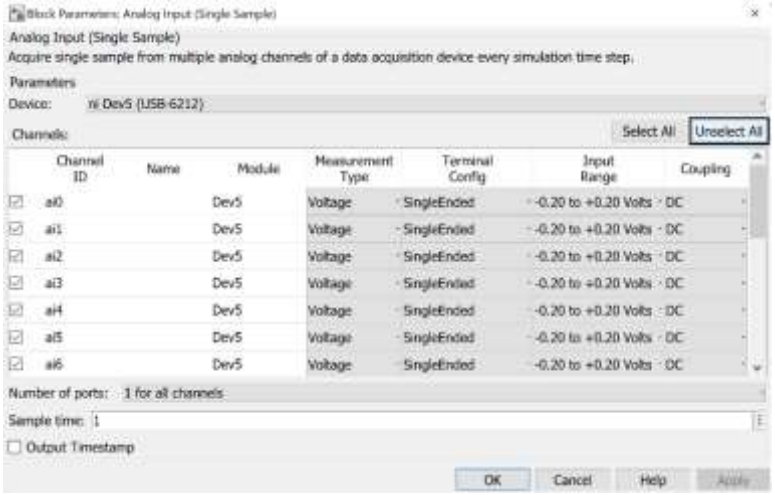


Figura 24. Configuración del bloque de entradas analógico

- Una vez desmarcadas las entradas marcar la entrada “ai1”; desplegar el menú del parámetro “Input Range” y seleccionar “-10 to +10 Volts”; Del menú del parámetro “Terminal Config” escoger “Differential”; en las

opciones de “Number of ports” seleccionar “1 per channel”; en “Sample time” colocar “1”. Clic en “Apply” y “OK” (véase Figura 25).

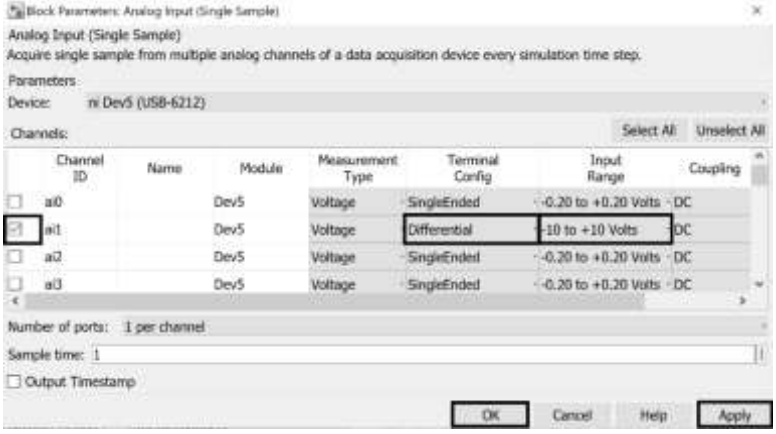


Figura 25. Configuración del bloque de entradas analógicas

- Doble clic en “Analog Output” marcar únicamente la salida “ao1”; desplegar el menú del parámetro “Input Range” y seleccionar “-10 to +10 Volts”; en las opciones de “Number of ports” seleccionar “1 per channel”; en “Sample time” colocar “1”. Clic en “Apply” y “OK” (véase Figura 26).

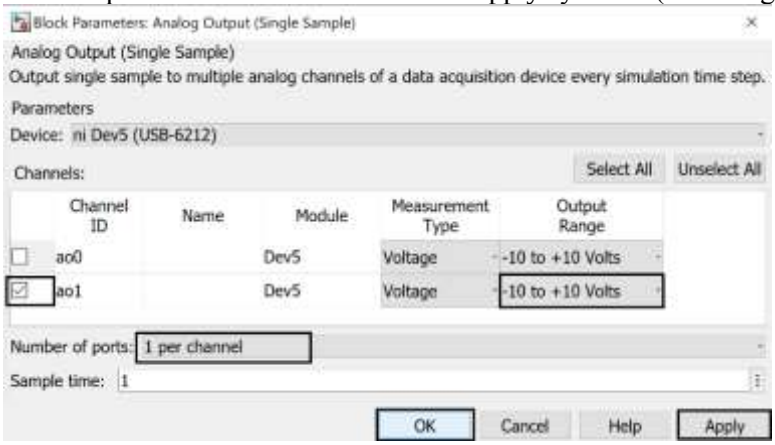


Figura 26. Configuración del bloque de salidas analógicas

- Se elimina el bloque de la planta del Modelo generado, se conecta el bloque de entrada analógica (Analog Input) a la señal del bloque MPC (mo) y la señal del bloque MPC (mv) al bloque de salida analógica (Analog Output) (véase Figura 27).

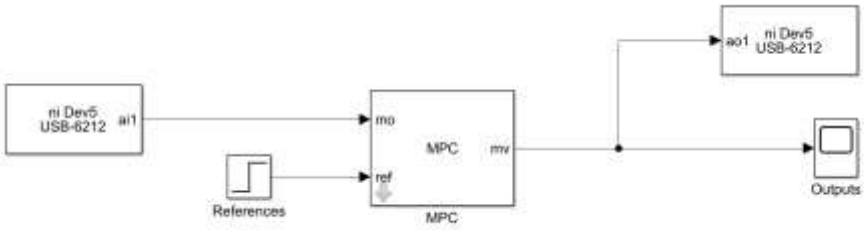


Figura 27. Conexión de la DAQ y el bloque MPC

- En el buscador de librerías escribir "real time" y arrastrar el bloque "Real Time Synchronization" a la ventana de Simulink (véase Figura 28).

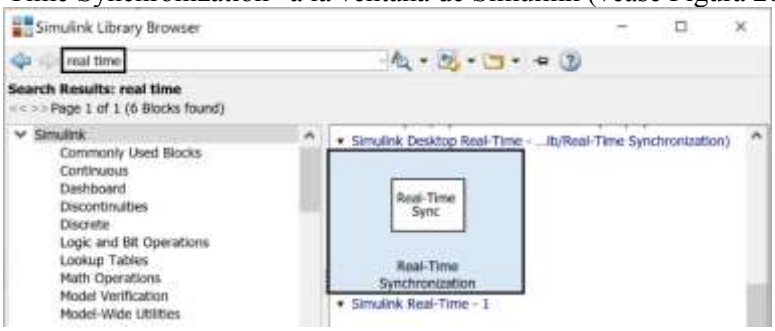


Figura 28. Bloque Real-Time Sync

- Doble clic en el bloque "Real-Time Sync" y colocar "1" en "Sample time". Clic en "Apply" y en "OK" (véase Figura 29).

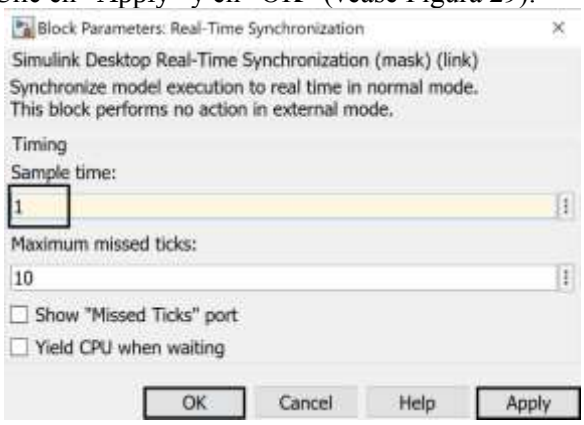


Figura 29. Configuración del tiempo de muestreo

- En el buscador de librerías se busca “display”, “To Workspace”, “product”, “divide” y “constant” para observar el comportamiento del controlador (véase Figura 30)

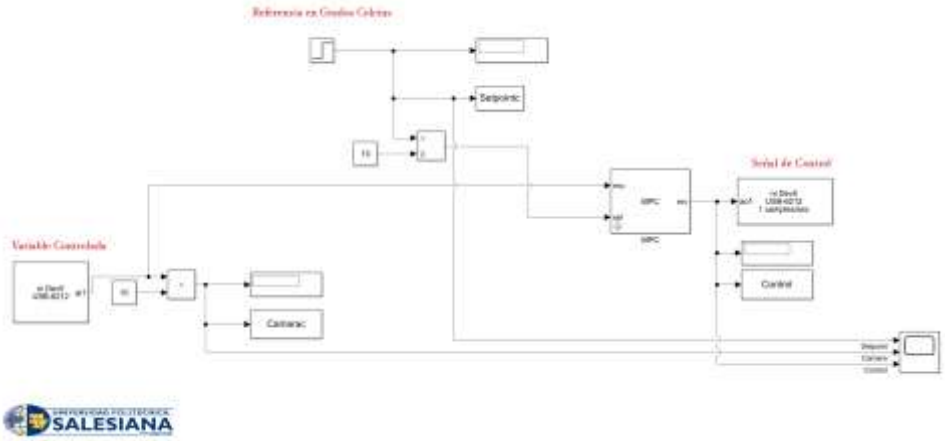


Figura 30. Controlador MPC

- Abrir el buscador de librerías, seleccionar “Dashboard” y arrastrar los bloques “Half Gauge” y “Slider” a la ventana de Simulink (véase Figura 31 y 32).

Figura 31. Bloque “Half Gauge”

Figura 32. “Slider”

- Una vez agregados los bloques dar doble clic sobre el bloque "Half Gauge", donde se abrirá la ventana de configuración de este (véase Figura 33).

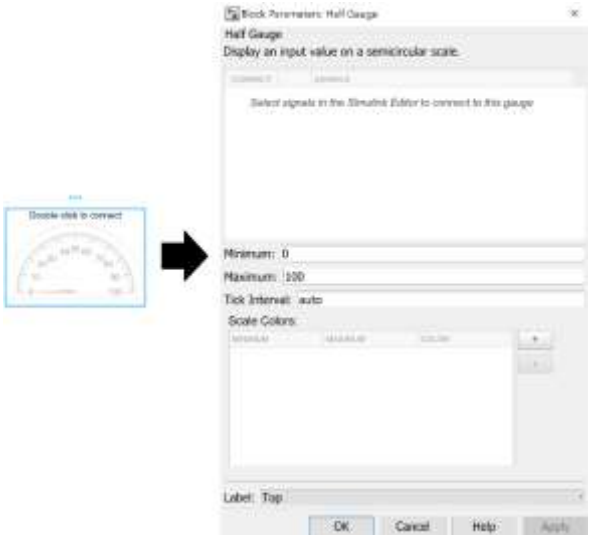


Figura 33. Parámetros del bloque "Half Gauge"

- Seleccionar el bloque "Analog Input" ya conectado; una vez seleccionado aparecerá la señal "Analog Input (Single Sample):1" en las configuraciones del bloque. Seleccionar la señal y dar clic en "OK" (véase Figura 34).

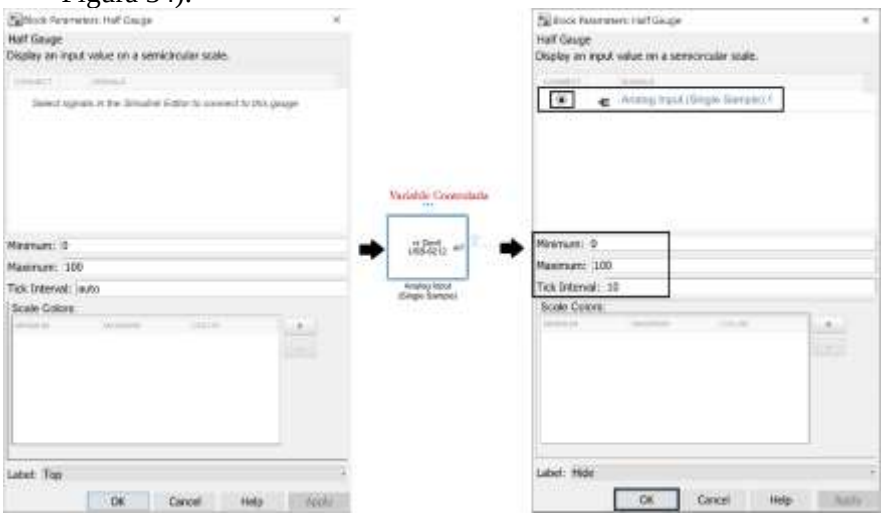


Figura 34. Seleccionando señal del bloque "Half Gauge"

- Doble clic sobre el bloque “Slider”, donde se abrirá la ventana de configuración de este (véase Figura 35)

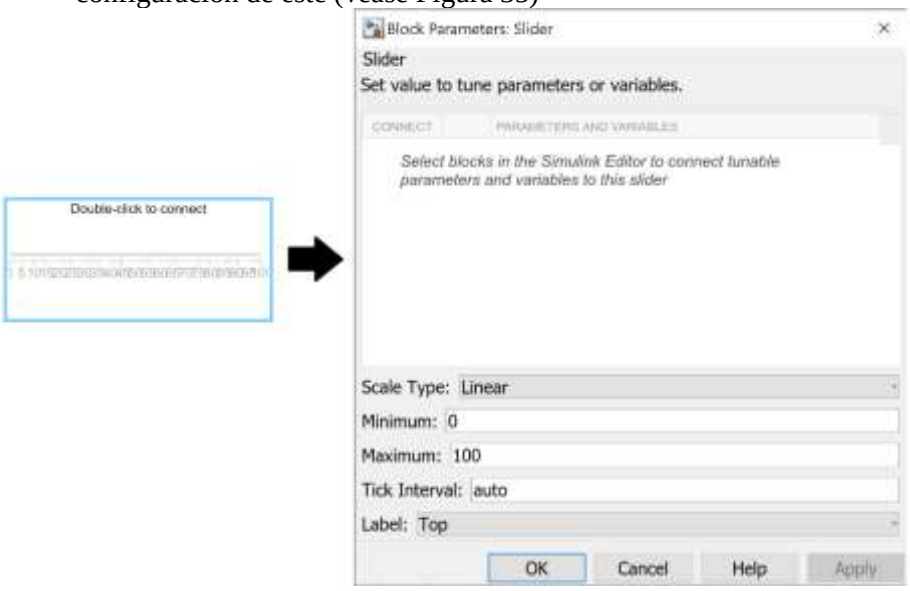


Figura 35. Parámetros del bloque “Slider”

- Seleccionar el bloque “Step” destinado a colocar la referencia; una vez seleccionado aparecerá el parámetro “Step: After” en las configuraciones del bloque. Seleccionar el parámetro y dar clic en “OK” (véase Figura 36).

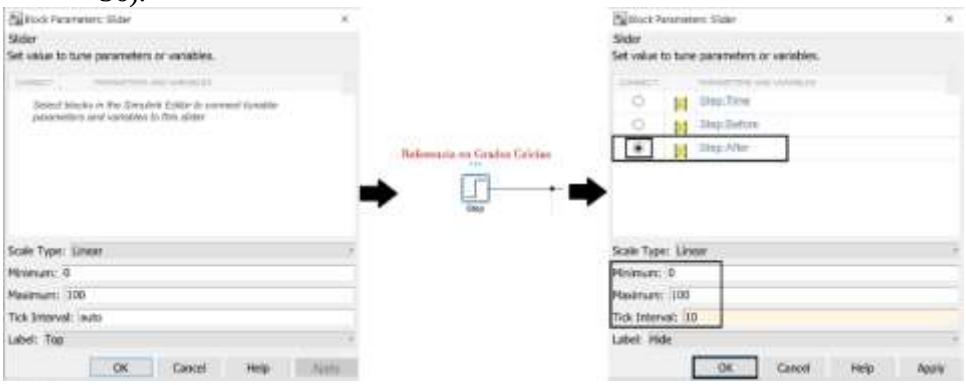


Figura 36. Seleccionando señal del bloque “Slider”

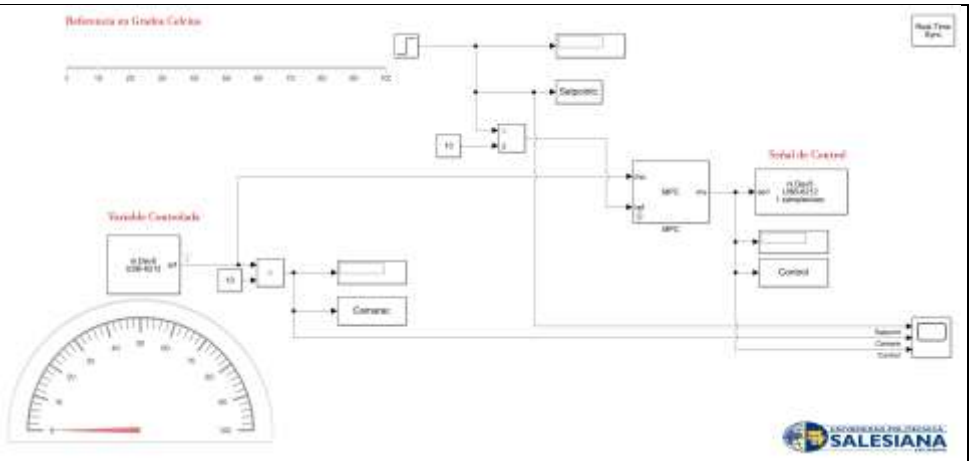


Figura 37. Configuración de bloques

- Colocar "inf" en la duración de la simulación y dar clic en "Run"

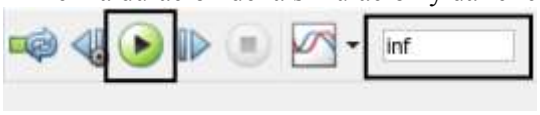


Figura 38. Ejecutando simulación

- Se coloca la referencia en el valor deseado en este caso 40, se puede variar la referencia al llegar a la estabilidad.
- Una vez comprobado el funcionamiento del controlador, parar el programa y guardar los datos del "Workspace"

Figura 39. Variables Workspace

- En la pestaña "HOME", en la sección "VARIABLE", seleccionar "Save Workspace" (Guardar Espacio de Trabajo) (véase Figura 40).

Nota: Guardar los datos en la misma carpeta que el "Script", el nombre del archivo en este caso se llamará datos.mat.

Figura 40. Guardando los datos

- Los datos guardados se grafican con el siguiente código, estos se observan en la Figura 41

```
load datos.mat
ds=1;
t=0:ds:length(Setpointc)-ds;
figure(1)
subplot(211);
plot(t,Setpointc,'b','LineWidth',1.2); xlabel('Time(s)'); ylabel('Temperatura (°C)');
hold on; grid on
plot(t,Camarac,'r','LineWidth',1.2);
title('Controlador PID')
legend('Referencia','Temperatura de la cámara')
subplot(212);
plot(t,SControl,'k','LineWidth',1.2);
xlabel('Time(s)');ylabel('Porcentaje'); grid on
legend('Señal de Control')
```

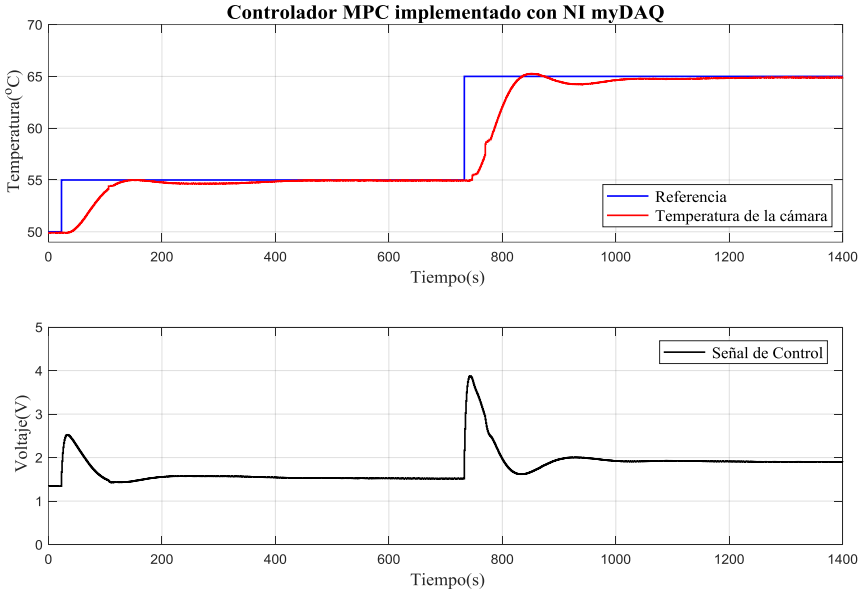


Figura 41. Controlador MPC

5. Diagrama de conexión

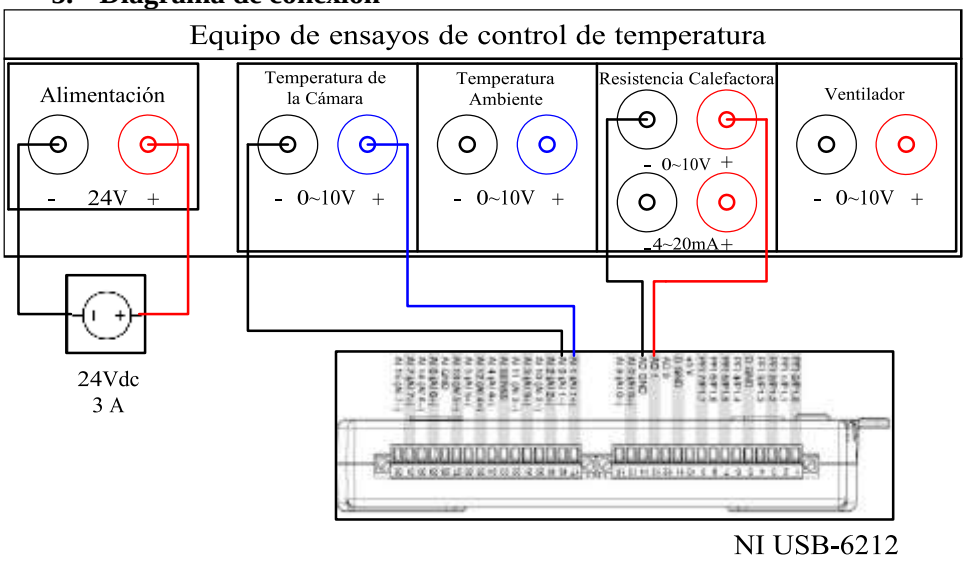


Figura 42. Diagrama de conexión

RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

APÉNDICES

- **GUÍA DE PRÁCTICA 8: CONTROLADOR PREDICTIVO BASADO EN EL MODELO (MPC) IMPLEMENTADO EN EL EQUIPO DE ENSAYO DE CONTROL DE TEMPERATURA MEDIANTE SIMULINK Y PLC.**

FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: INGENIERÍA ELECTRÓNICA		**ASIGNATURA**: TEORÍA DE CONTROL
NRO. PRÁCTICA:	8	**TÍTULO PRÁCTICA**: **CONTROLADOR PREDICTIVO BASADO EN EL MODELO (MPC) IMPLEMENTADO EN EL EQUIPO DE ENSAYO DE CONTROL DE TEMPERATURA MEDIANTE SIMULINK Y PLC.**

OBJETIVOS

1. Conocer las características generales de la herramienta MPC Designer de Matlab
2. Conocer las características generales de la comunicación OPC
3. Implementar un controlador predictivo basado en el modelo (MPC)

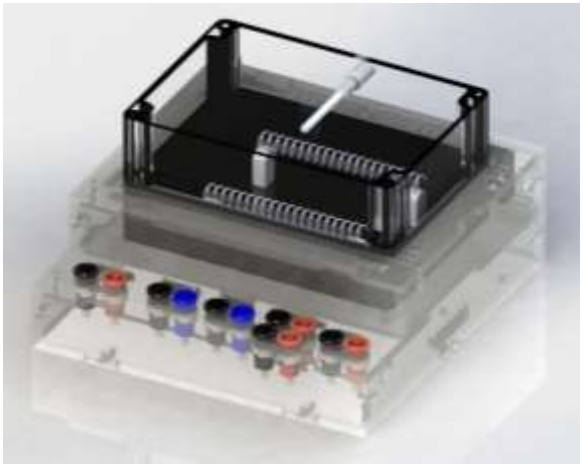
INSTRUCCIONES
(Detallar las instrucciones que se dará al estudiante):

1. Requisitos y conocimientos previos

a) Creación de un Proyectos en TIA Portal y Matlab

2. Equipos, instrumentos y software

Descripción	Cantidad	Marca	Identificación / serie
Computadora S.O. Windows 10, 64 bits	1	-	-
Tia Portal v13.	1	-	-
Software Matlab 2017	1	-	-
PLC S7-1200	1	SIEMENS	-
Equipo de ensayo de control de temperatura	1	CTILab	-
Cables	Varios	-	-

	Tabla 1. Equipos, instrumentos y software
	3. Exposición ***Equipo de ensayos de temperatura****:* este dispositivo permite realizar ensayos de control de temperatura con picos de hasta 55 °C, equipado con una niquelina fabricada con alambre de nicromo, dos ventiladores 3010 para perturbaciones, sensores de temperatura tipo RTD Pt100 junto con transmisores calibrados para proporcionar una señal de 0 a 10 volts en el rango 0 a 100 °C. El dispositivo admite señales continuas de 0 a 10 volts para el control de la potencia entregada al ventilador como a la niquelina.  *Figura 1. Equipo de ensayo de control de temperatura.* **SIMATIC S7-1200:** estos dispositivos ofrecen la automatización de varias aplicaciones industriales gracias a su flexibilidad. Tiene los más altos requisitos de la industria para aplicaciones que requieren de alta tecnología. Cuenta con un diseño compacto para el ahorro de espacio puesto que contiene entradas y salidas integradas, además, permite incrementar placas de montaje adicionales como entradas y salidas, la comunicación RS485 y la placa de batería al controlador. El controlador

	SIMATIC S7-1200 es capaz de manejar tareas estándar y relacionadas con la seguridad. Es posible programar el CPU para la comunicación con equipos como SIMATIC HMI. El equipo SIMATIC S7-1200 admite conmutación serial, PROFIBUS, IO-Link, AS-Interface, entre otros [20]. *Figura 2. Equipo SIEMENS S7-1200* [20].

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. **Conexión del PLC**

- En el buscador de Windows escribir “Change adapter options”
- En la ventana “Network Connections” clic derecho en el puerto de red conectado al PLC, clic en “Propiedades”

Figura 3. Propiedades del puerto ethernet

- Doble clic en "Internet Protocol Version 4 (TCP/IPv4)"

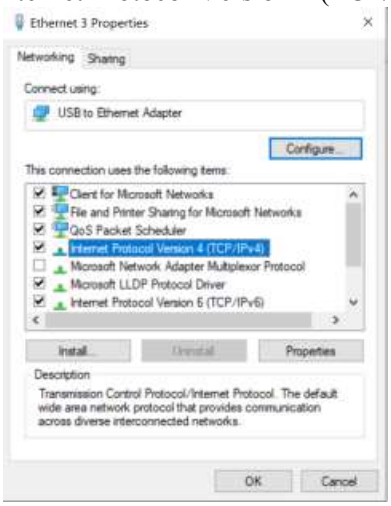


Figura 4. Propiedades del puerto

- Colocar la dirección IP "192.168.0.2", la máscara de red "255.255.255.0", puerta de enlace "192.168.0.1".
- Clic en "OK"

Figura 5. Configuración de dirección IP

2. Abrir el programa Tia Portal v13

- Seleccionar "Create New Project" y colocar el nombre del proyecto y la ubicación del archivo finalmente damos clic en "Create"

Figura 6. Creando nuevo proyecto

- Seleccionamos "PLC programming" y hacemos clic en "add a new device"

Figura 7. Añadiendo un nuevo dispositivo

- Seleccionamos "CPU 1214C AC/DC/RLy", "6ES7 214-1BE30-0XB0", "V1.0" y damos clic en "OK"

Figura 8. Colocando un dispositivo

- Hacemos doble clic en “Main” para abrir la ventana de programación

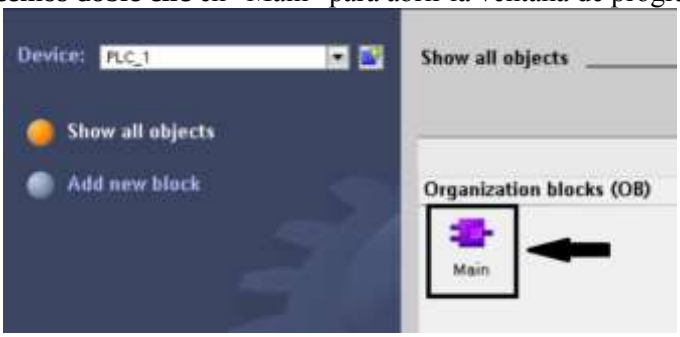


Figura 9. Programando PLC

- En el Árbol del Proyecto dar clic en “Device configuration”

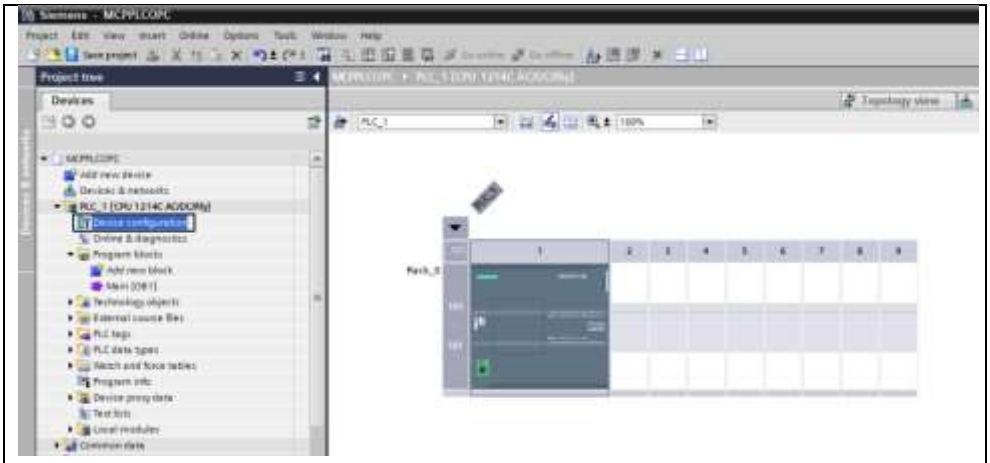


Figura 10. Device configuration

- En la ventana “Hardware catalog” ubicado al lado derecho de la interfaz, desplegamos “Signal boards”, “AQ” y “AQ 1x12BIT” y damos doble clic en “6ES7 232-4HA30-0XB0”

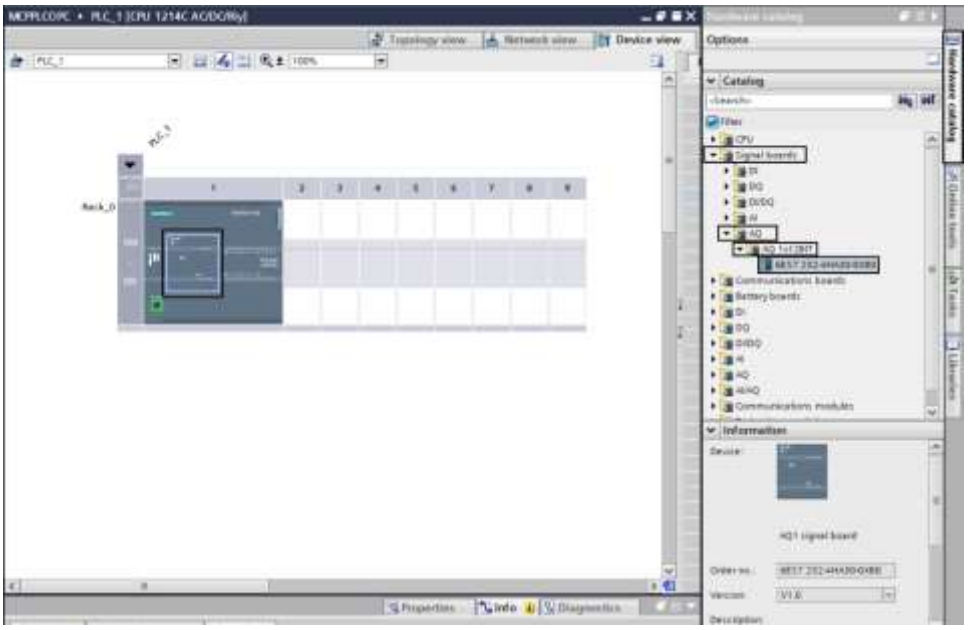


Figura 11. Añadiendo bloque de salidas analógicas

- Reanudamos a la sección “main” dando doble clic.

Figura 12. Programando el proyecto

- Desplegamos la pestaña “Basic Instructions” ubicado al lado derecho de la interfaz
- Desplegamos “Convesion operations” y arrastrar el bloque “NORM_X” y “SCALE_X” al “Network 1” y “Network 2” (véase Figura 13)

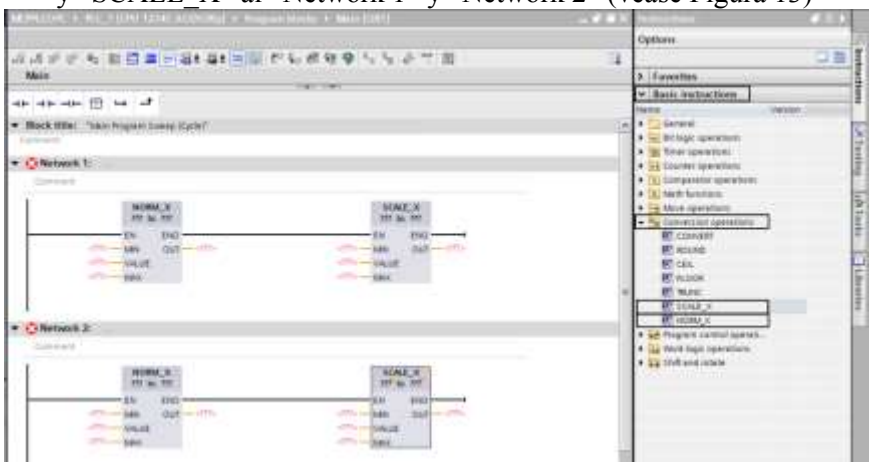


Figura 13. Bloques para programación

- Colocar “%IW66” para la entrada de señales analógicas y “%QW80” para la salida de señales analógicas. (véase Figura 14)

APÉNDICES

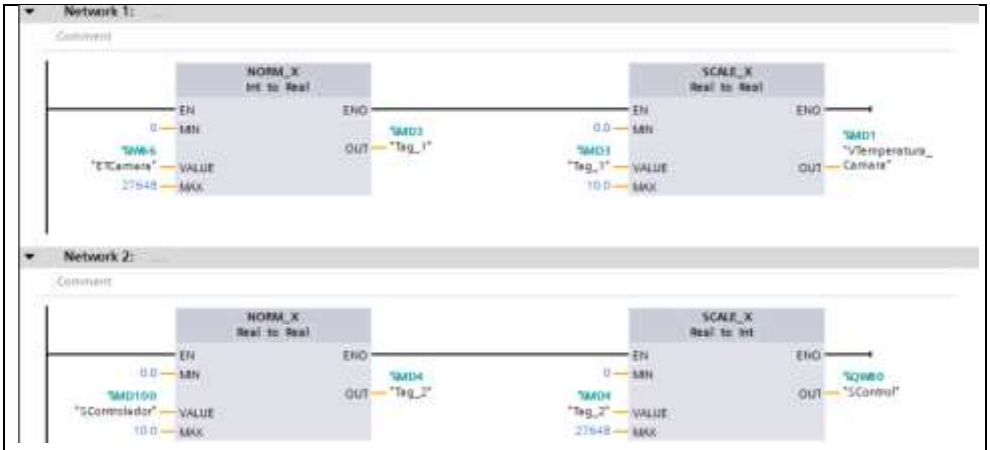


Figura 14. Configuración de bloques

- Clic en “Download to device” para cargar el programa en el PLC

Figura 15. Cargando programa

- En el menú “Type of the PG/PC interface” seleccionar “PN/IE”
- EN el menú “PG/PC interface” seleccionar “USB to Ethernet Adapter”
- Clic en “Start search”
- Clic en “192.168.0.1”
- Clic en “Load”

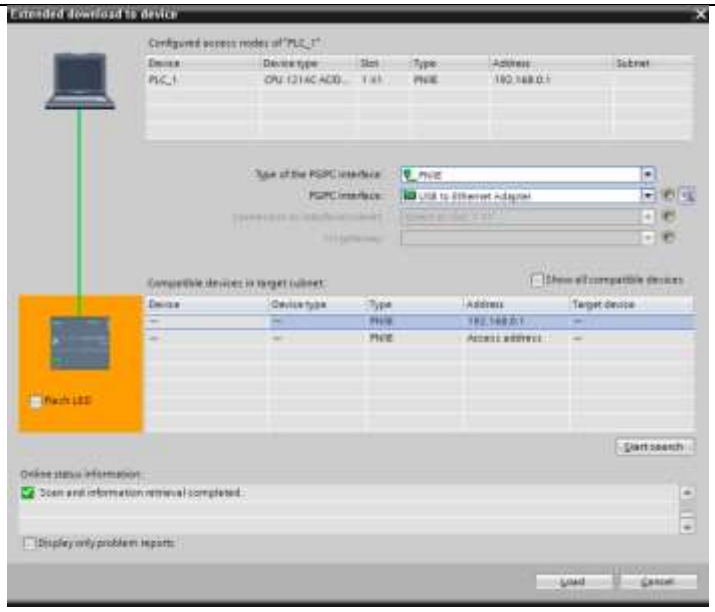


Figura 16. Cargando programa en el PLC

- En la ventana emergente damos clic en “Load”

Figura 17. Cargar programa en el PLC

- Marcamos la casilla “Start all” y damos clic en “Finish”

Figura 18. Cargando programa en el PLC

3. Descarga e instalación del software KEPServerEX

- Descargar el software KEPServerEX del siguiente link https://www.kepware.com/en-us/products/kepserverex/ e instalar
- Abrir el software KEPServerEX como administrador
- Clic derecho en "Connectivity", clic en "New Channel"

Figura 19. Creando Nuevo Canal

- Colocar el nombre del canal, clic en "Next"

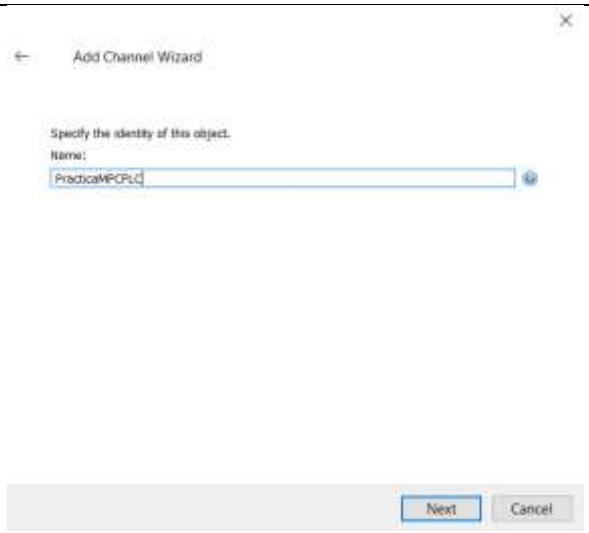


Figura 20. Colocando el nombre del canal

- Escoger la opción del adaptador conectado al PLC

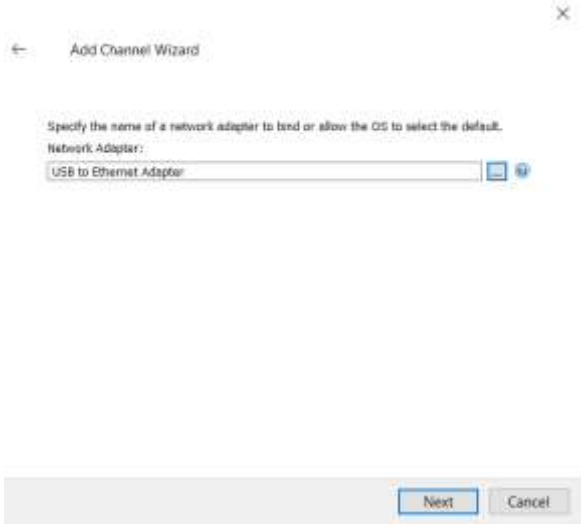


Figura 21. Adaptador de red

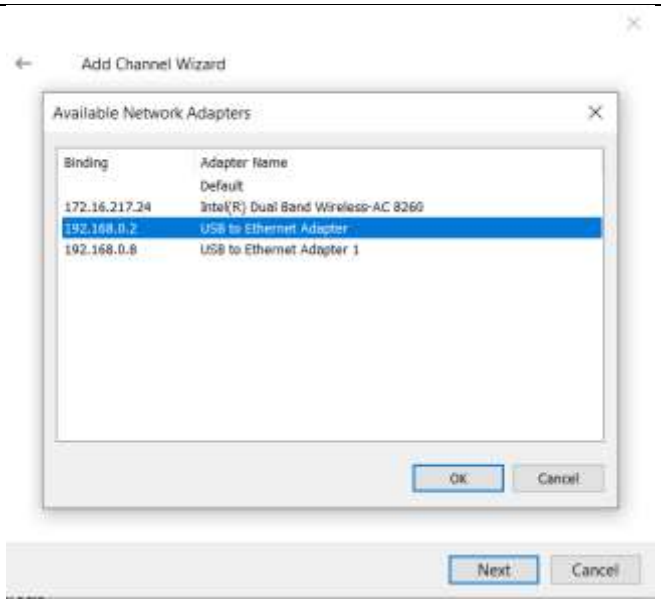


Figura 22. Adaptador de red

- Clic en "Next" hasta llegar a la ventana de resumen, clic en "Finish"

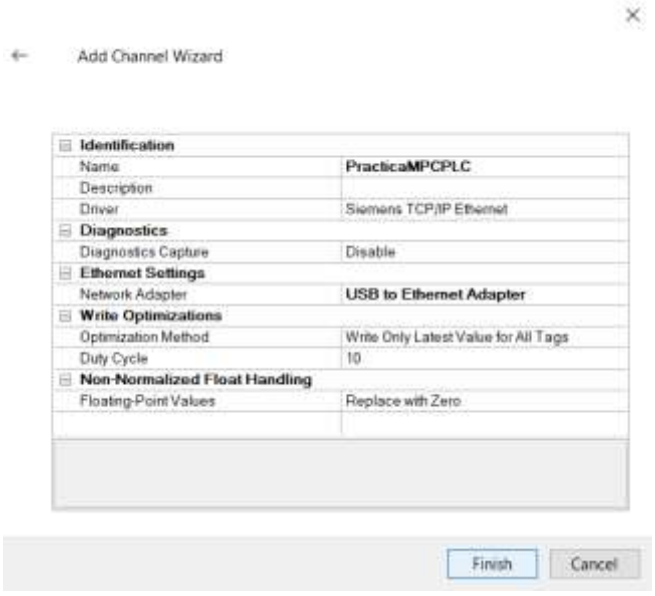


Figura 23. Ventana de resumen

- Clic derecho sobre el nombre del canal, escoger “New Device”

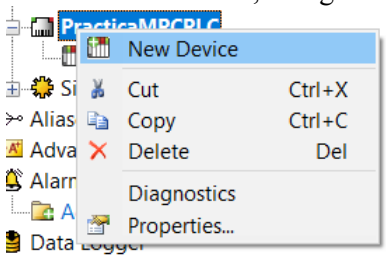


Figura 24. Creando un nuevo dispositivo

- Colocar el nombre del dispositivo

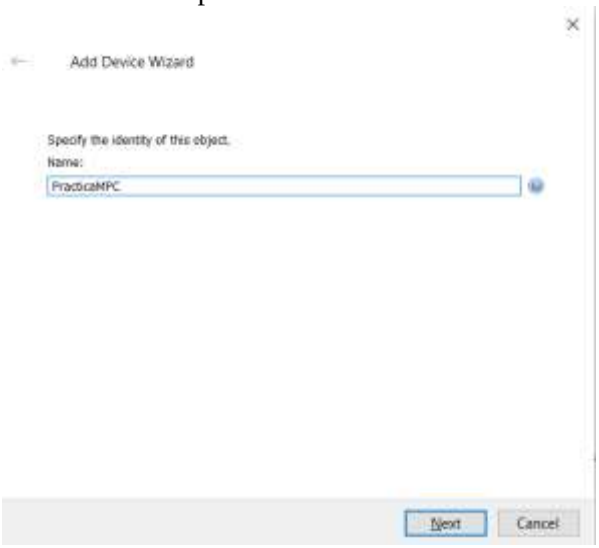


Figura 25. Colocando el nombre al dispositivo’

- Escoger el modelo del dispositivo “S7-1200”

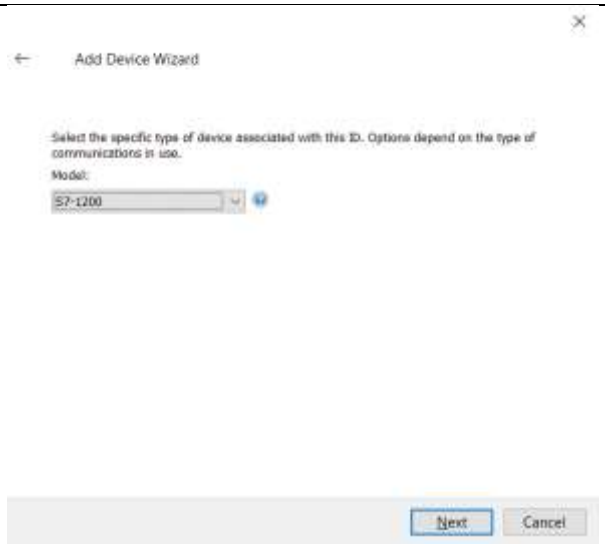


Figura 26. Modelo del dispositivo

- Colocar la dirección IP del dispositivo “192.168.0.1”

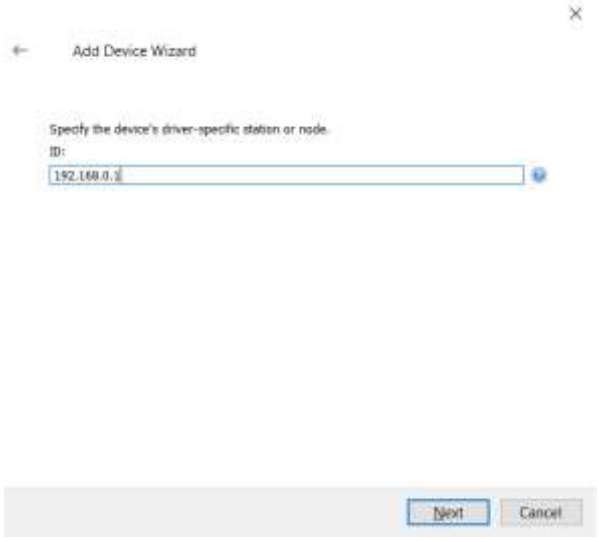


Figura 27. Dirección IP del dispositivo

- Clic en “Next” en todas las pantallas hasta llegar a la ventana de resumen.

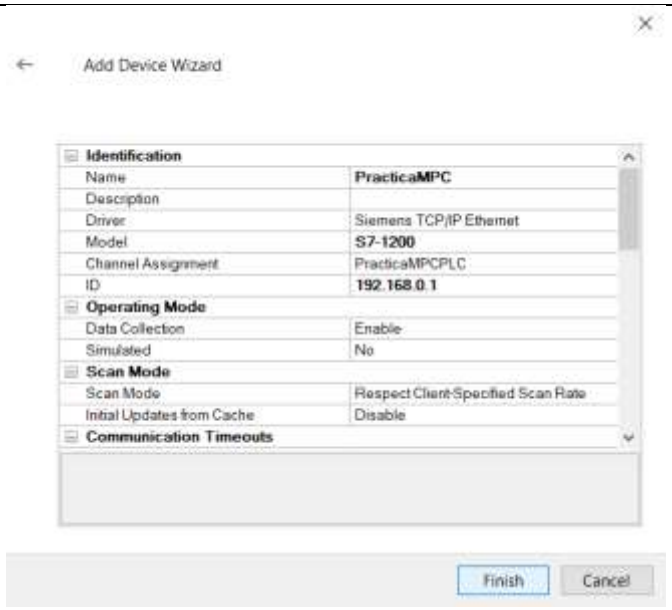


Figura 28. Ventana de resumen

- Abrir KepServer y dar clic en “Click to add a static tag”

Figura 29. Agregar un “static tag”

- Agregar las variables para la entrada de temperatura y la salida de control

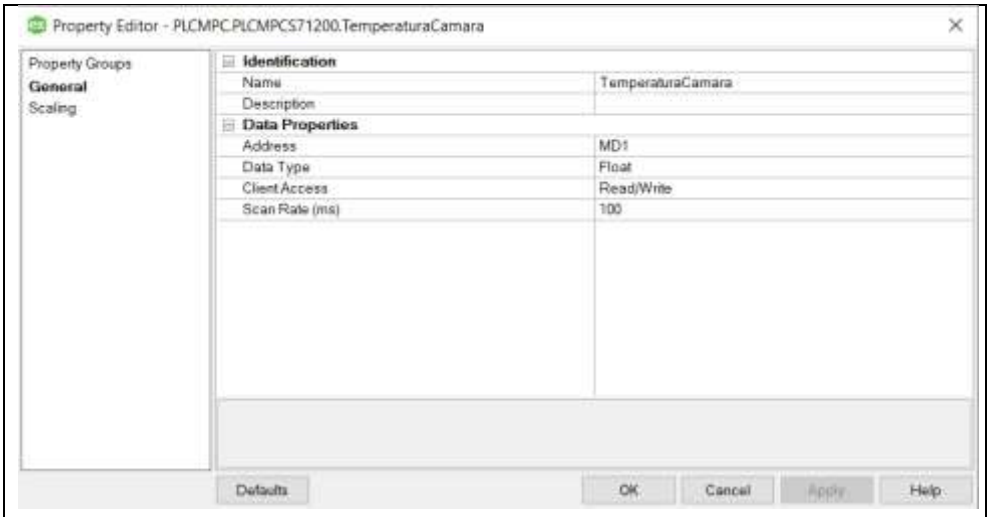


Figura 30. Variable Temperatura cámara

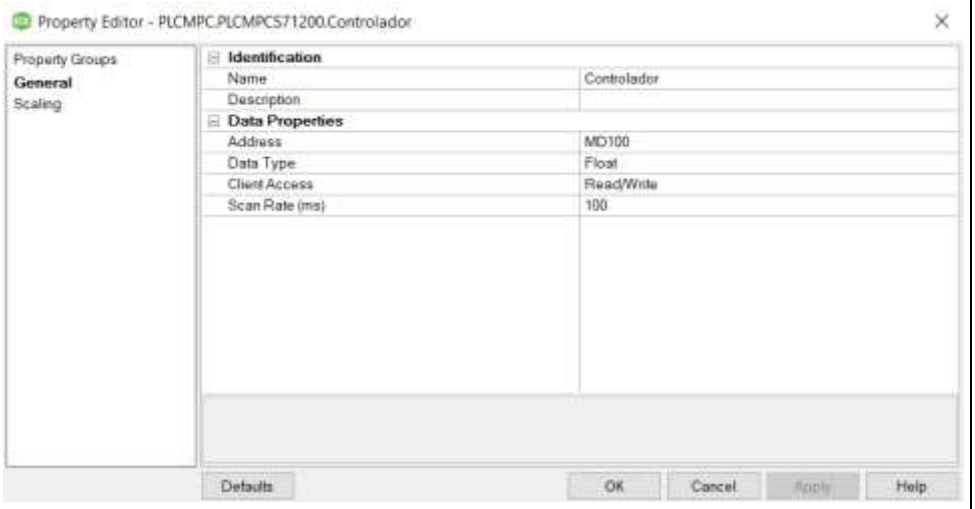


Figura 31. Variable "Controlador" (Señal de control)

Tag Name	Address	Data Type
Controlador	MD100	Float
TemperaturaCamara	MD1	Float

Figura 32. Tabla de resumen de variables

- Clic en "Quick Client" donde se podrá observar el estado de las variables

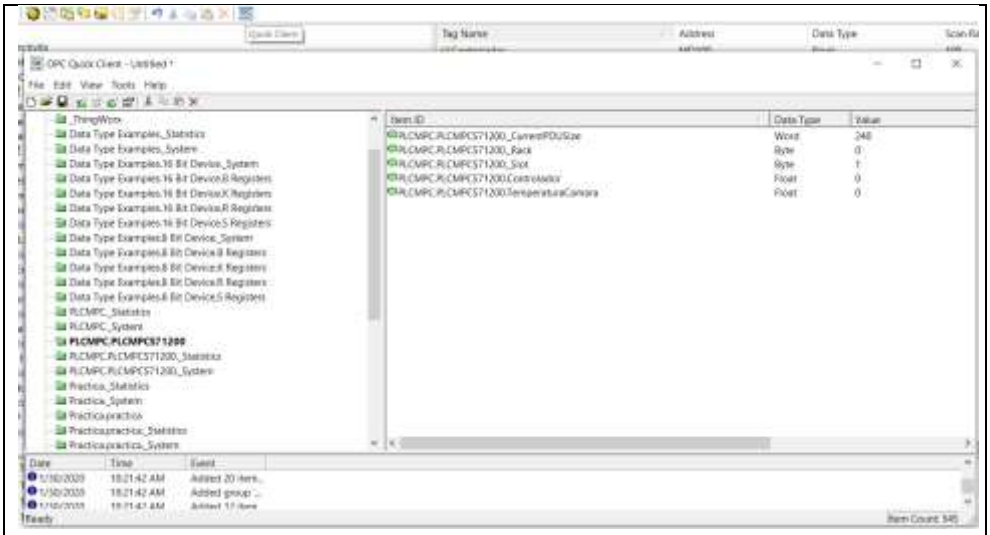


Figura 33. Ventana de estado de variables

4. Diseño del controlador

- Abrir Matlab 2017b

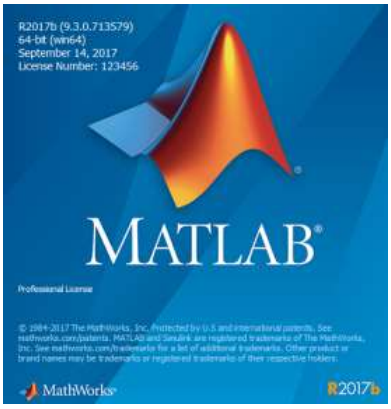


Figura 34. Matlab

- En la Ventana de Comandos (Command Windows) escribir "`sldrtkernel -install`"

Figura 35. Instalando kernel para Real Time de Simulink

- Colocar "y" en el mensaje

```
You are going to install the Simulink Desktop Real-Time kernel.
Do you want to proceed? [y] :
```

Figura 34. Mensaje para proceder con la instalación del Real-Time kernel

```
The Simulink Desktop Real-Time kernel has been successfully installe
```

Figura 35. Mensaje de instalación exitosa

- En la Ventana de Comandos (Command Windows) escribir "mpcDesigner"

Figura 36. Abriendo MPC Designer

- En la ventana emergente "MPC DESIGNER" dar clic en "MPC Structure"

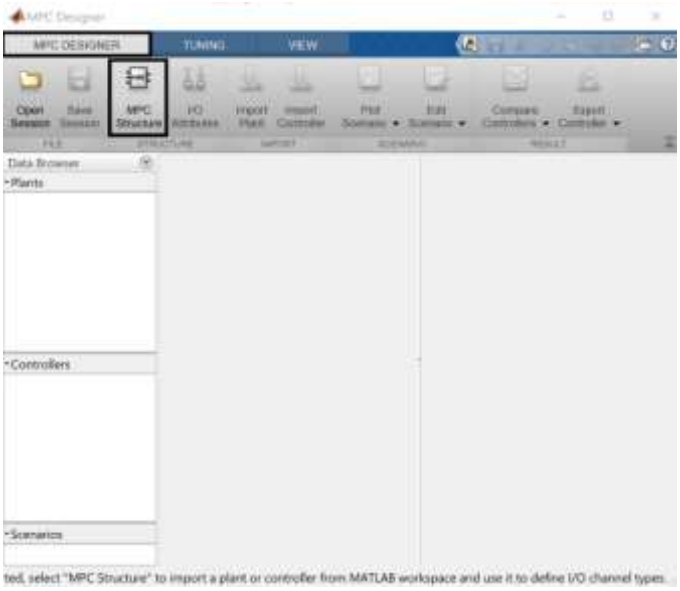


Figura 37. MPC Designer

- En la ventana "Define MPC Structure By Importing"
- Clic en el nombre de la planta obtenido anteriormente en espacio de estados en este caso "CS"

- En “Specify MPC controller Sample time” colocar “1”
- La variable manipulada (MV) y salida medida (MO) serán “1”
- Clic en “Define and Import”

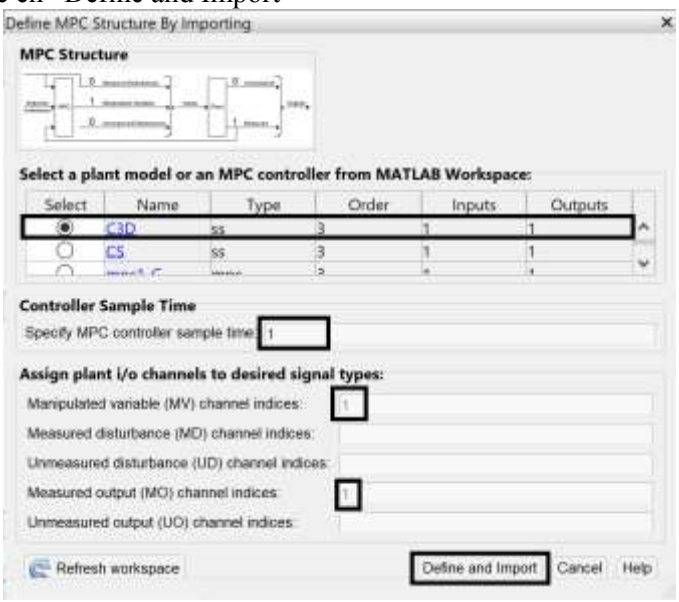


Figura 38. Importando planta a controlar

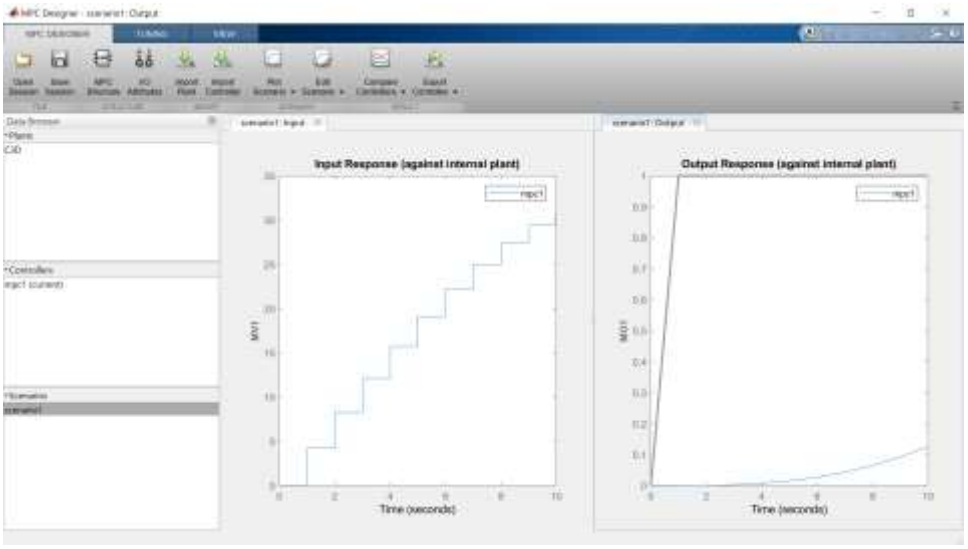


Figura 39. Planta ingresada

- En la pestaña “MPC DESIGNER” clic en “Edit Scenario”, clic en “scenario1”

Figura 40. Editando escenario

- En la ventana “Simulation scenario” coloca al menos 500s en la duración de la simulación, clic en “Apply”, clic en “OK”. En esta ventana es posible cambiar el tamaño del escalón y su tiempo de inicio.

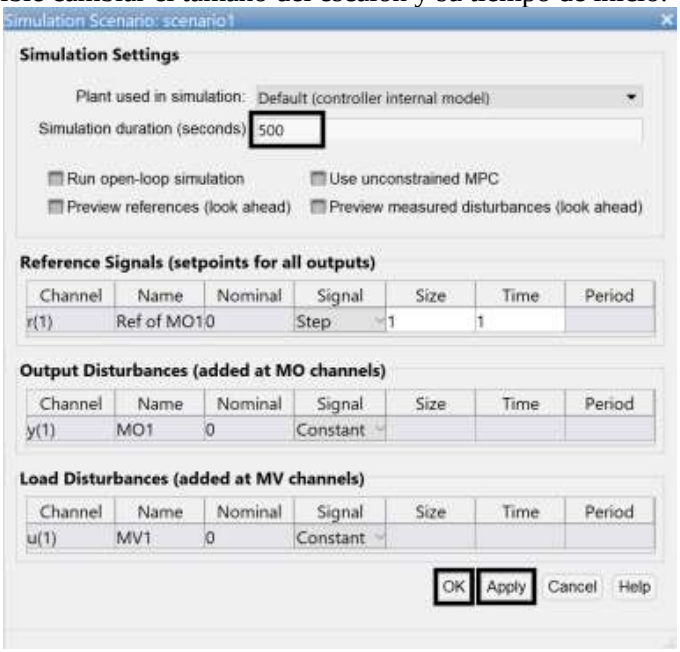


Figura 41. Editando tiempo de simulación

- En la pestaña “TUNING”, se puede cambiar los parámetros de la sección “HORIZON” para estabilizar el sistema.

Figura 42. Modificando parámetros para estabilidad del sistema

- En la pestaña “TUNING” dar clic en “Constraints”

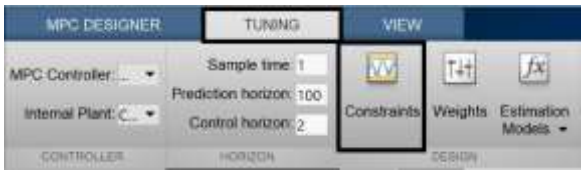


Figura 43. Constraints

- En la ventana “Constraints” ubicarse en “Input Constraints”, colocar “0.01” y “10” en “Min” y “Max” respectivamente.
- Clic en “Apply”, clic en “OK”

Nota: Los parámetros RateMin y RateMax pueden modificarse para mejorar la respuesta del sistema

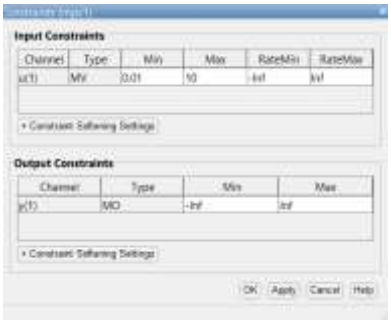


Figura 44. Constrainsts

- Se puede mejorar la respuesta del sistema modificando los de “PERFORMANCE TUNING”.

Figura 45. Performance Tuning

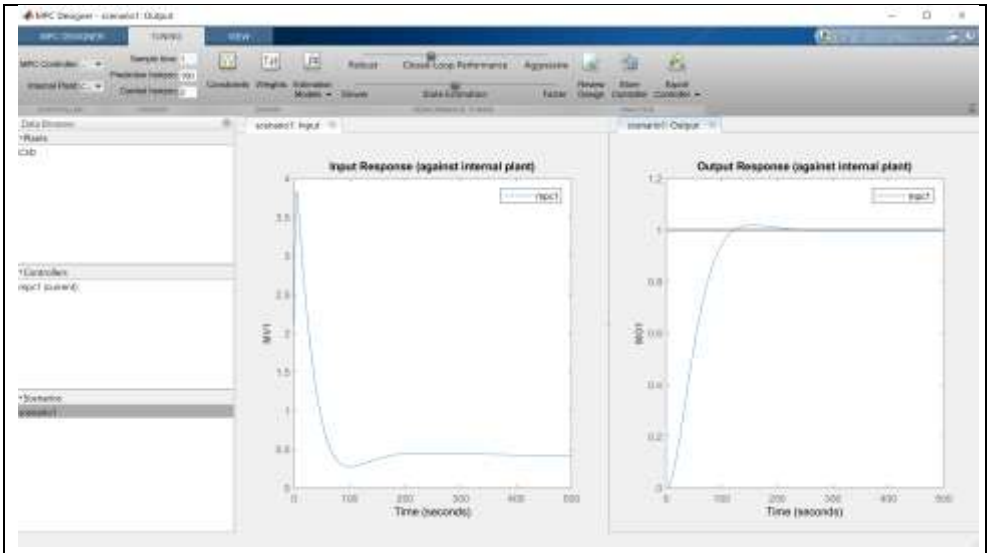


Figura 46. Simulación del modelo estabilizado

- En la pestaña "TUNING" en la sección "ANALYSIS", clic en "Export Controller", clic en "Generate a Simulink Model"

Figura 47. Generando Modelo de Simulink

5. Modelo generado en simulink del controlador MPC

Figura 48. Modelo Creado con MPC

- Dar clic en “Library Browser” ubicado en la parte superior izquierda del modelo creado.

Figura 49. Abriendo librerías.

- En el buscador escribir “opc” y arrastrar al modelo los bloque “OPC Read y “OPC Write”. Colocar el tiempo de muestreo de 1 segundo.

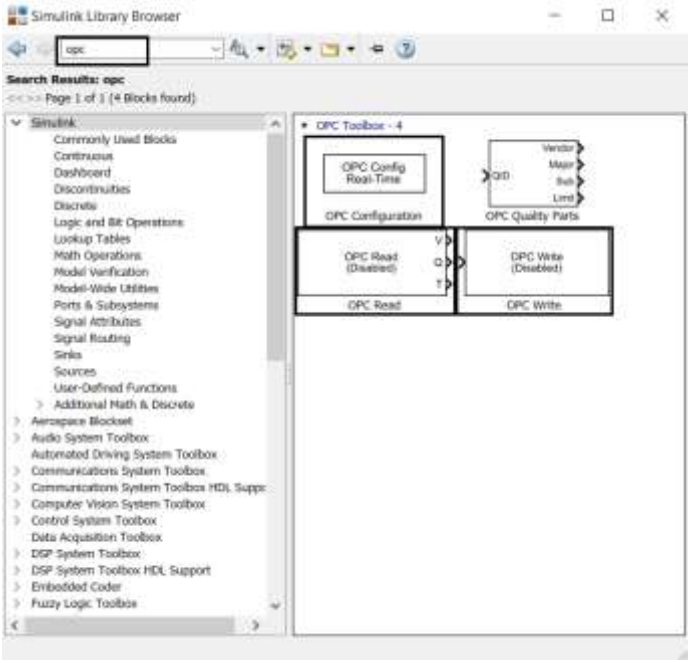


Figura 50. Agregando bloques para la comunicación OPC

- Dar doble clic en el bloque “Real-Time”
- En la ventana abierta dar clic en “Configure OPC Clients”

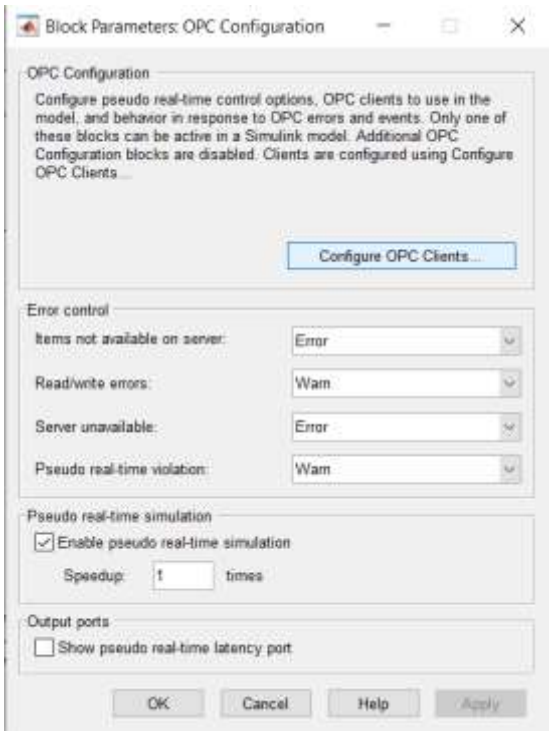


Figura 51. Configuración del bloque "OPC Configuration"

- Clic en "Add"
- Clic en "Select"
- Seleccionar "Kepware.KEPServerEX.V6"
- Clic en "OK"

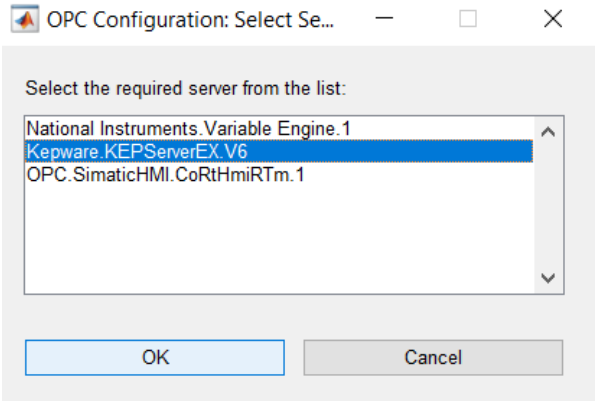


Figura 52. Selección del servidor OPC

- Doble clic en el bloque “OPC Read”
- Clic en “Add Items”
- Seleccionar el nombre del canal y buscar la variable “Temperatura de la cámara”
- Clic en “OK”

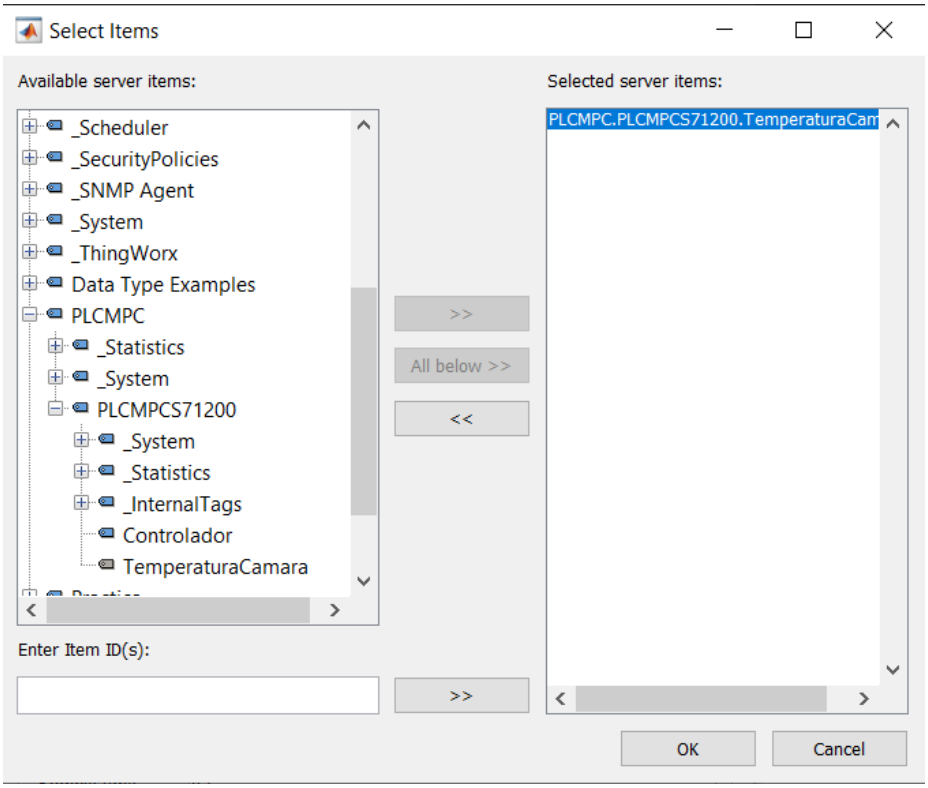


Figura 53. Agregar variable compartida “TemperaturaCamara”

- Configurar tiempo de muestreo en el bloque “OPC Read”, colocar “1” en “Sample time” (véase Figura 54)
- Clic en “OK”

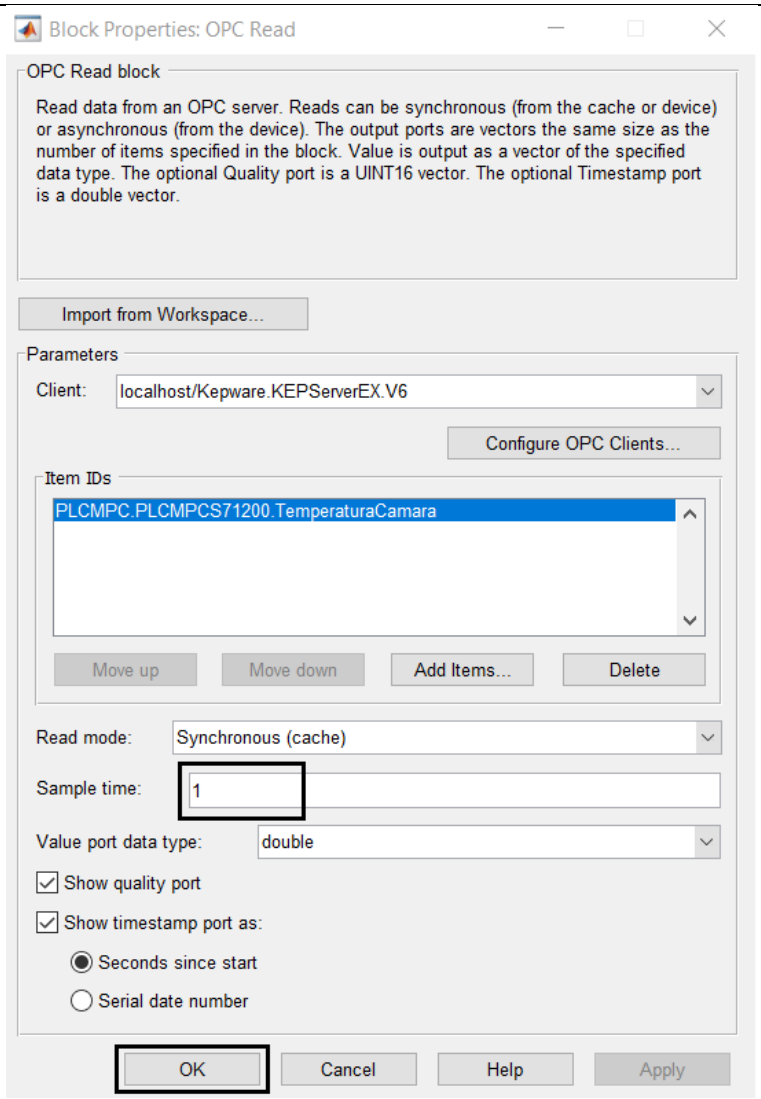


Figura 54. Configuración del tiempo de muestreo

- Doble clic en el bloque “OPC Write”
- Clic en “Add Items”
- Seleccionar el nombre del canal y buscar la variable “Controlador”
- Clic en “OK”

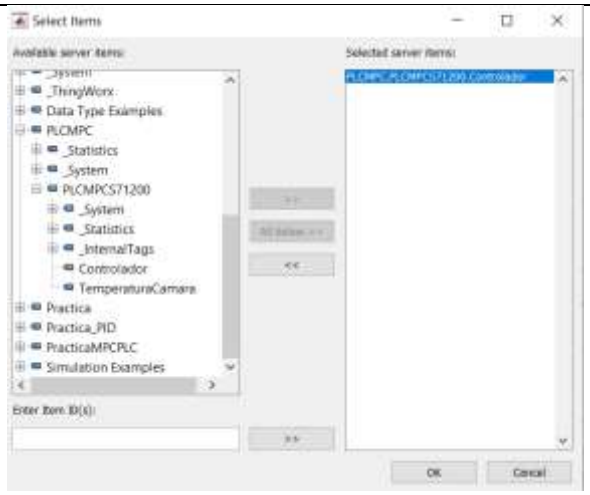


Figura 54. Agregar variable compartida "Controlador"

- Configurar tiempo de muestreo en el bloque "OPC Write", colocar "1" en "Sample time" (véase Figura 55)
- Clic en "OK"

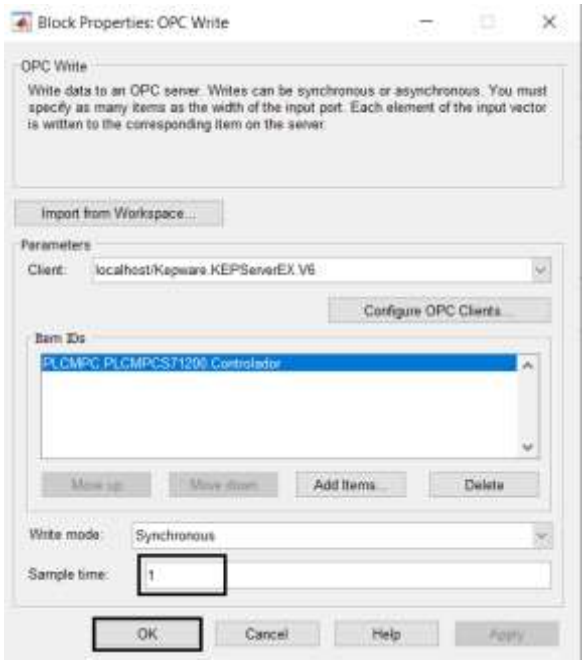


Figura 55. Configuración del tiempo de muestreo

- En el buscador de librerías escribir “real time” y arrastrar el bloque “Real Time Syncronization”

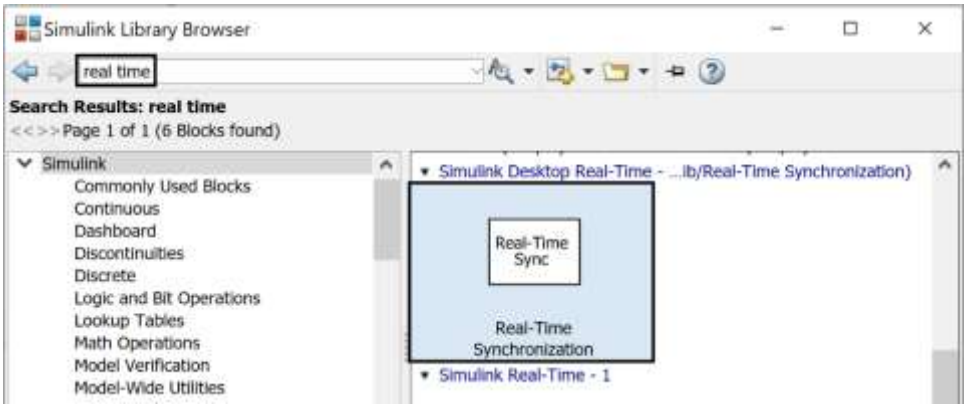


Figura 56. Bloque Real-Time Sync

- Doble clic en el bloque “Real-Time Sync” y colocar “1” en “Sample time”
- Clic en “Apply”, clic en “OK”

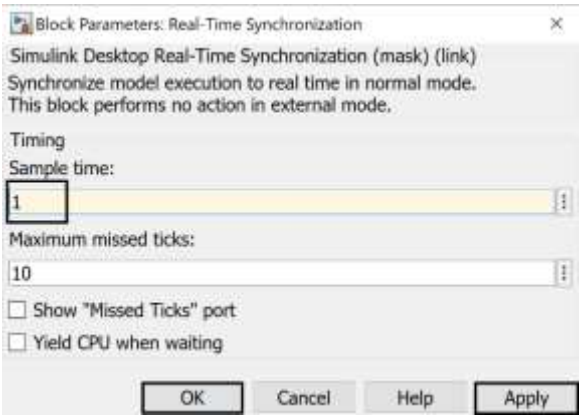


Figura 57. Configuración del tiempo de muestreo

- En el buscador de librerías se busca “display”, “To Workspace”, “product”, “divide” y “constant” para observar el comportamiento del controlador.

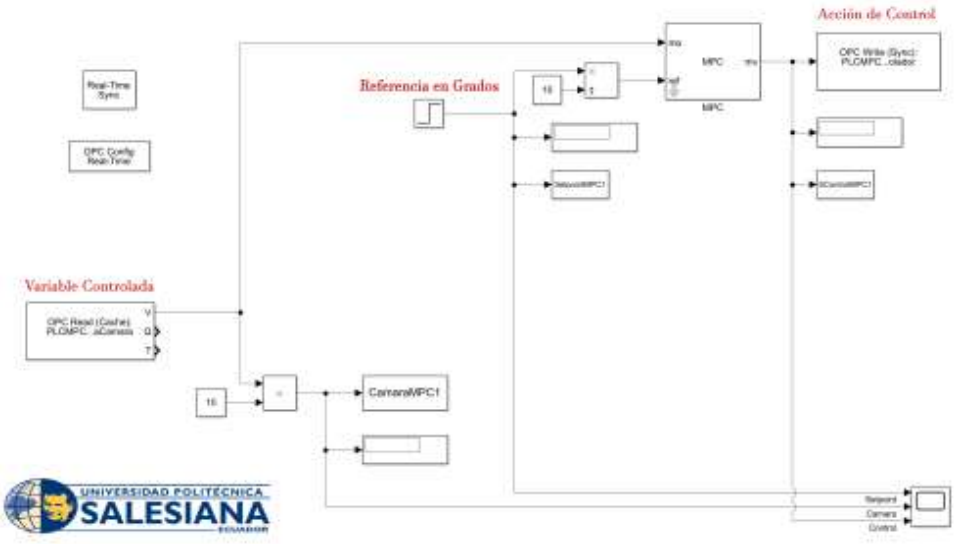


Figura 58. Controlador MPC

- Para una simulación continua colocan "inf" en el tiempo de duración de la simulación
- Clic en "Play"

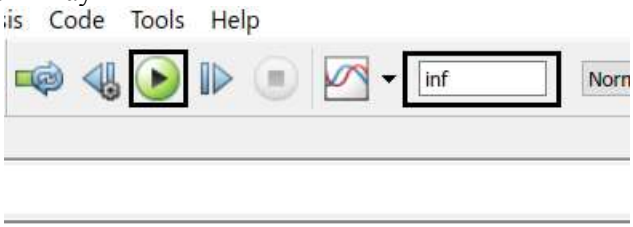


Figura 59. Ejecución del programa

6. Grafica de resultados en Matlab

- Los datos estarán en el workspace, de modo que no es necesario importarlos. En la figura 60 se muestra el comportamiento del sistema

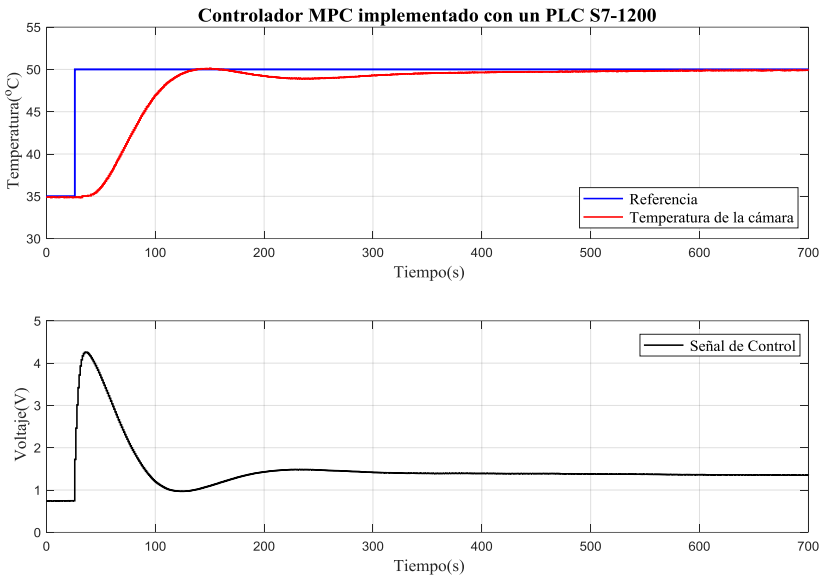


Figura 60. Respuesta del sistema en la implementación del controlador MPC con PLC S7-1200

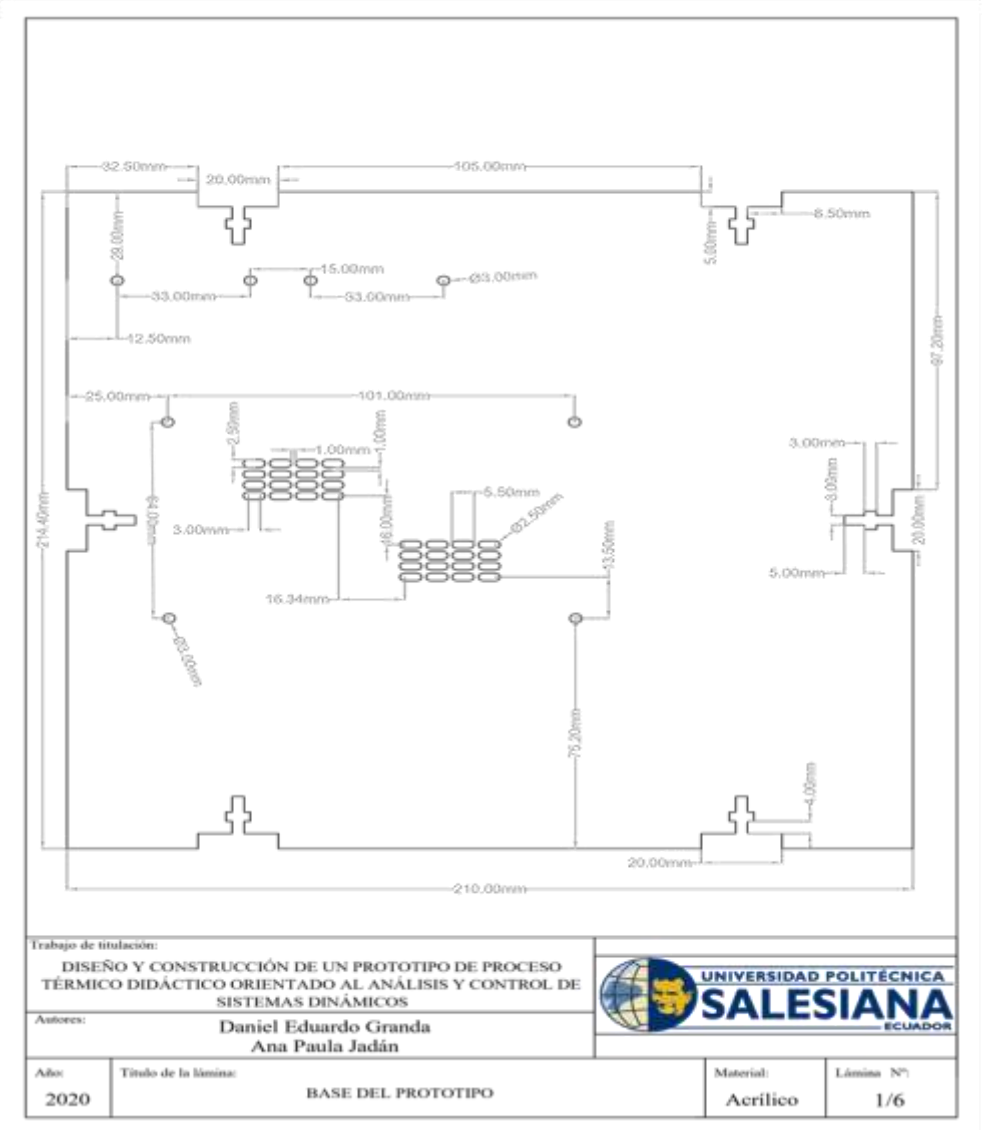
RESULTADO(S) OBTENIDO(S):

CONCLUSIONES:

RECOMENDACIONES:

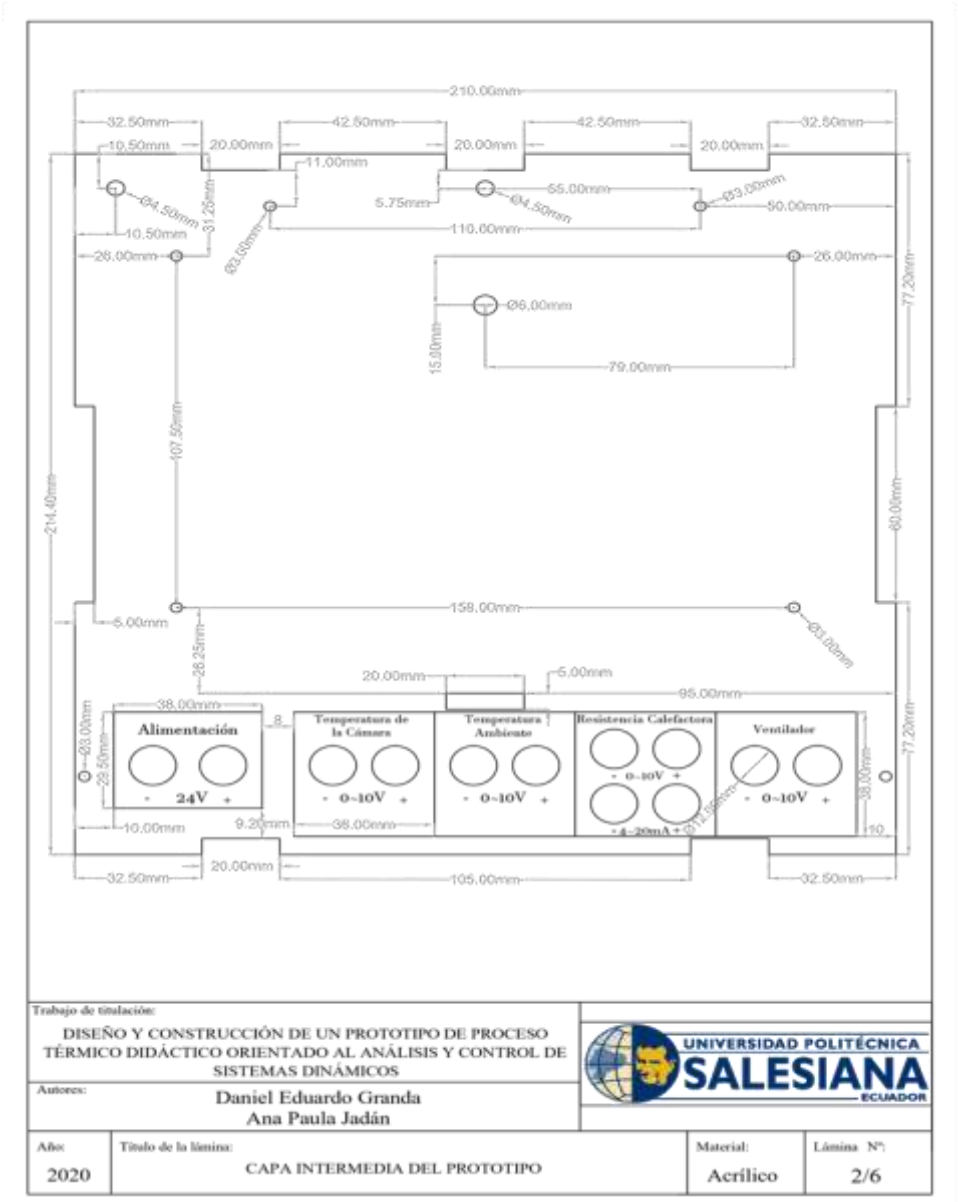
APÉNDICE B: PLANOS DEL PROTOTIPO

Pieza base del prototipo



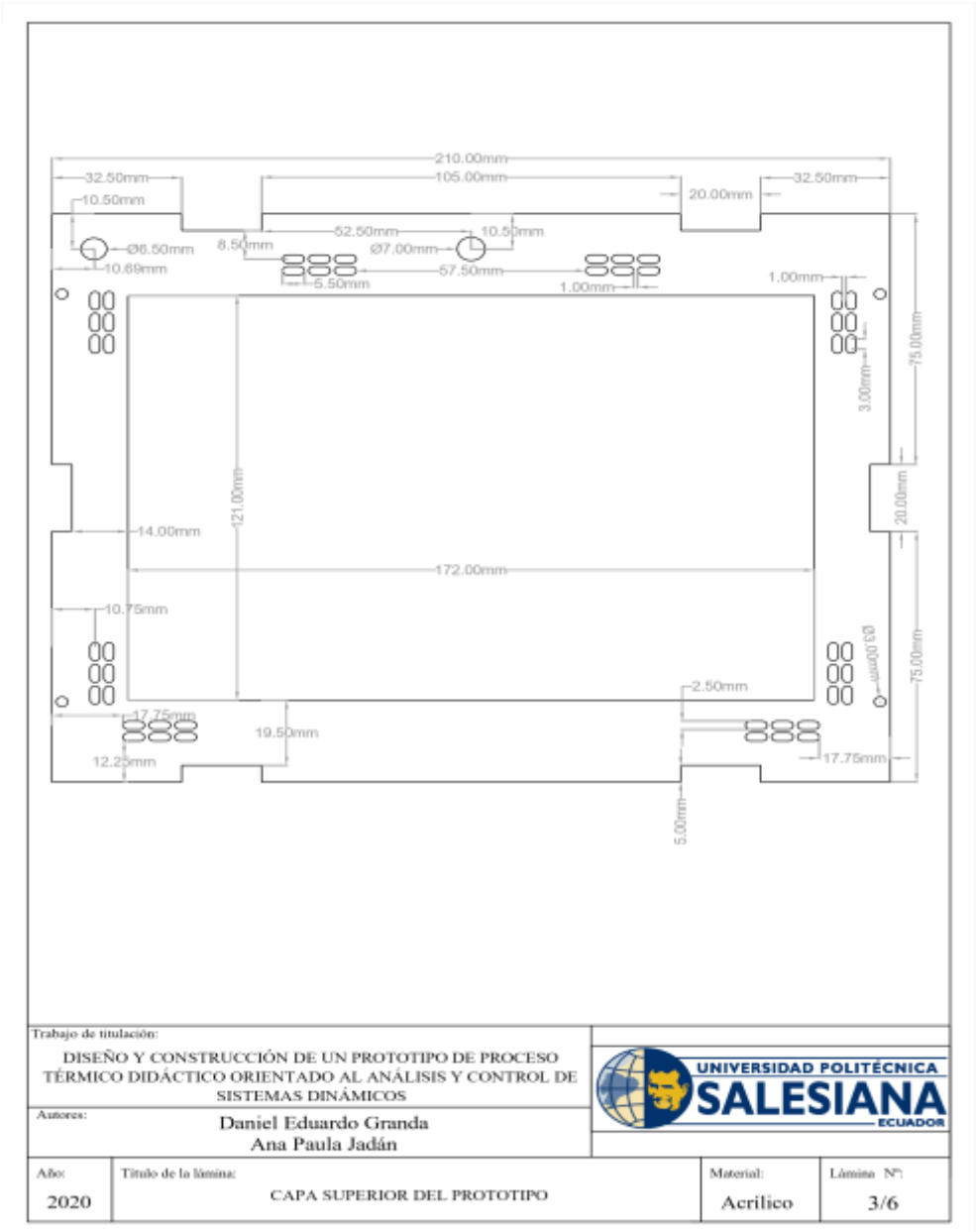
APÉNDICES

Pieza intermedia del prototipo



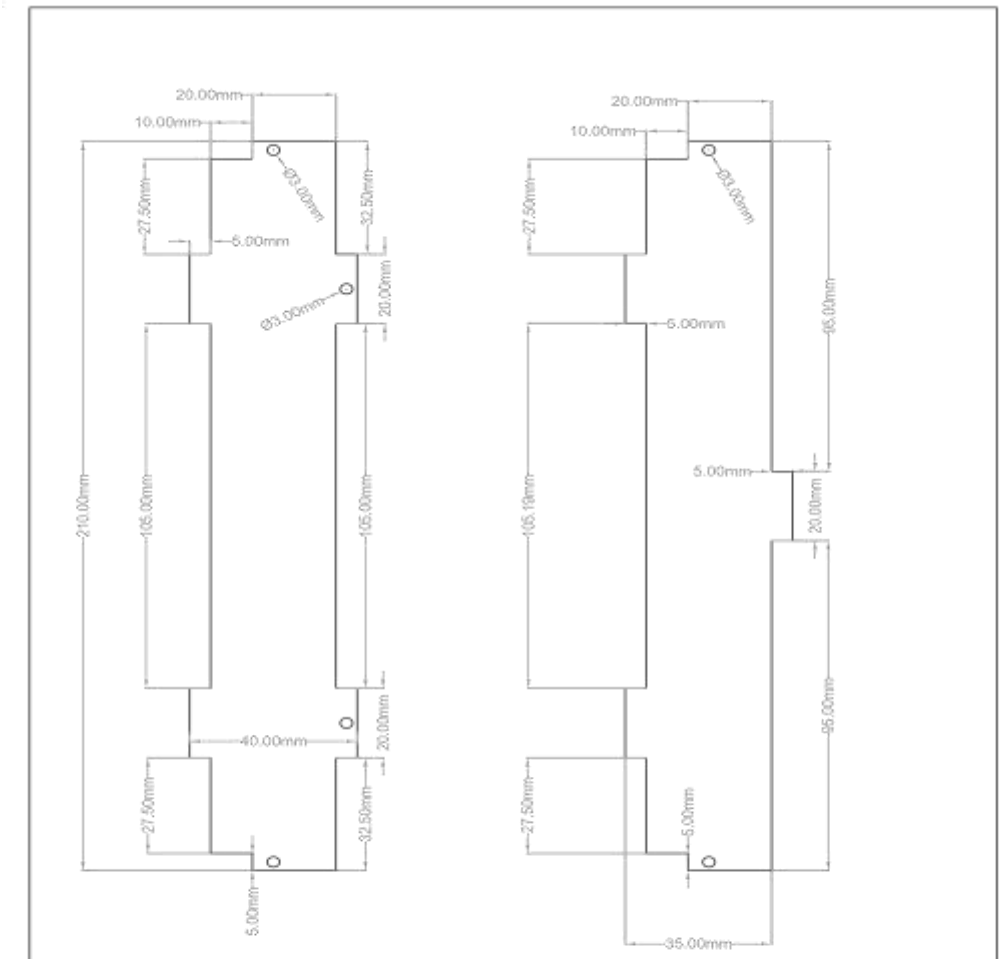
APÉNDICES

Pieza superior del prototipo



APÉNDICES

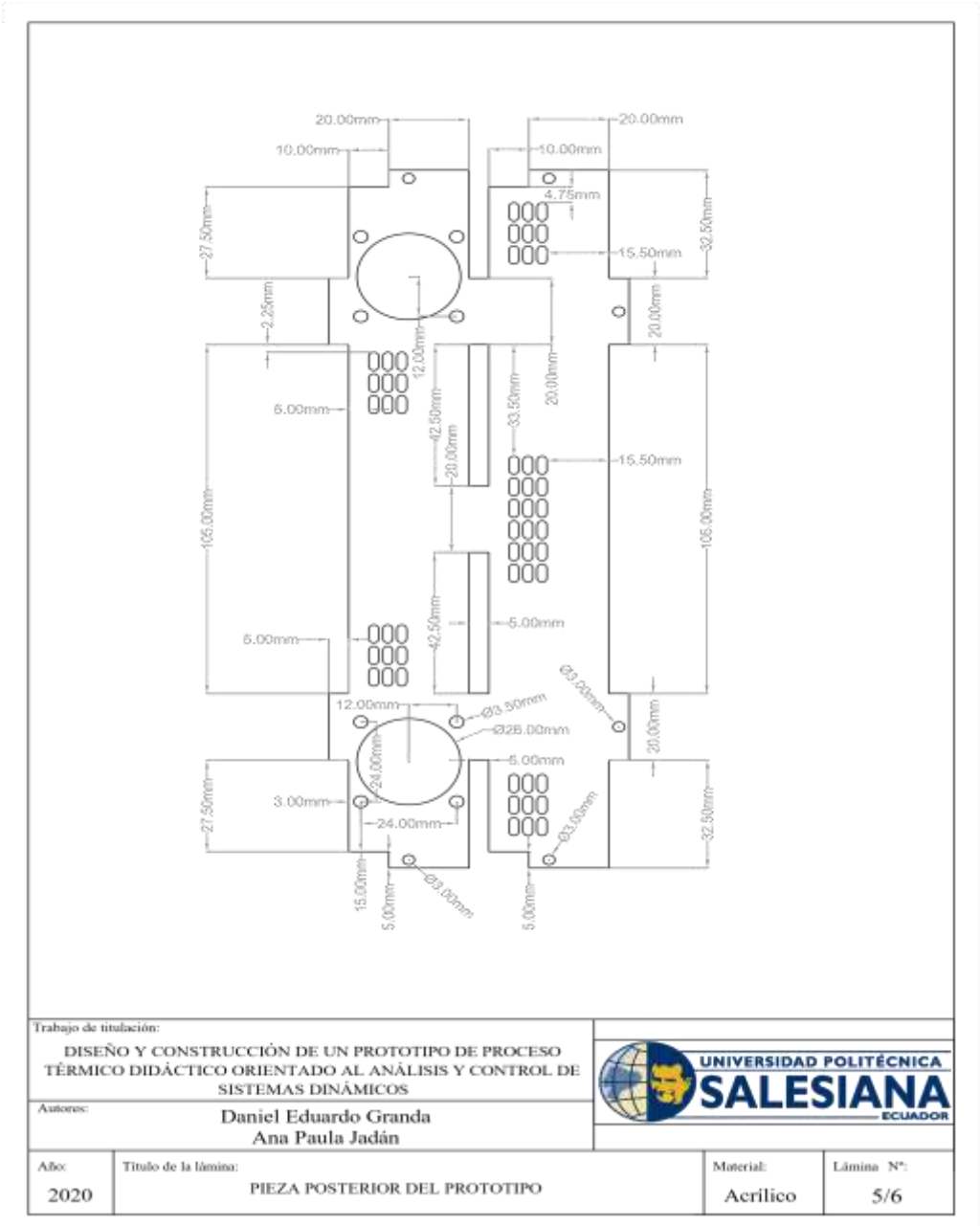
Piezas frontales del prototipo

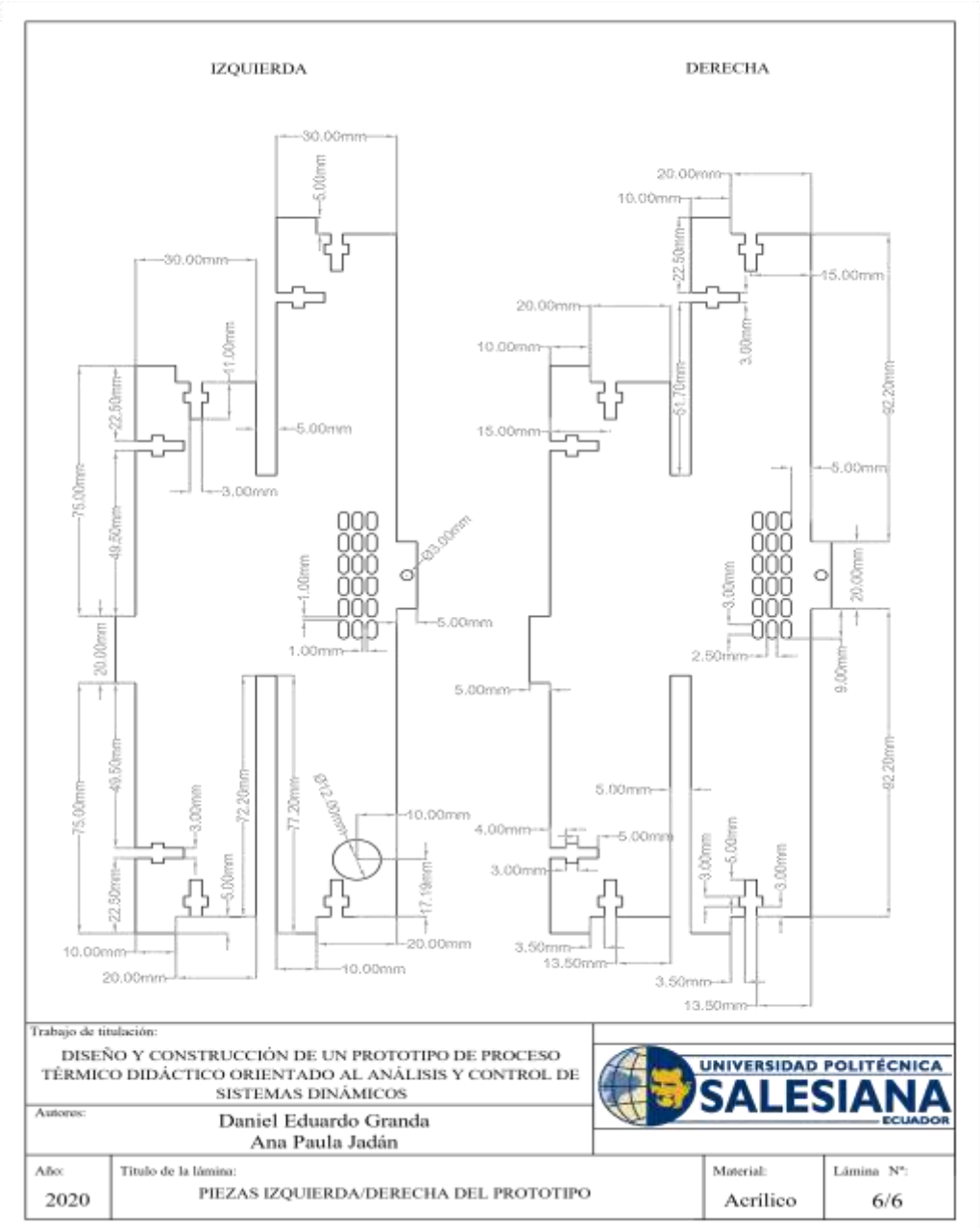


Trabajo de titulación: DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE PROCESO TÉRMICO DIDÁCTICO ORIENTADO AL ANÁLISIS Y CONTROL DE SISTEMAS DINÁMICOS			
Autores: Daniel Eduardo Granda, Ana Paula Jadán			
Año: 2020	Título de la lámina: PIEZAS FRONTALES DEL PROTOTIPO	Material: Acrilico	Lámina N°: 4/6

APÉNDICES

Pieza posterior del prototipo



APÉNDICES

Pieza izquierda/derecha del prototipo