



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO



FACULTAD DE CIENCIAS FÍSICAS Y
MATEMÁTICAS

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

TESIS

DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO DIDÁCTICO DE
CONTROL DE TEMPERATURA PARA LA ESCUELA
PROFESIONAL DE INGENIERÍA ELECTRÓNICA EN LA UNPRG

AUTORES

Bach. SANCHEZ CARLOS LUIS ALONSO

Bach. CRUZ IRIGOIN DANNY ADRIAN

PATROCINADOR

Ing. MARTÍN AUGUSTO NOMBERRA LOSSIO

Lambayeque - Perú

2018

UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS

TESIS

DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO DIDÁCTICO DE
CONTROL DE TEMPERATURA PARA LA ESCUELA
PROFESIONAL DE INGENIERÍA ELECTRÓNICA EN LA UNPRG

TESIS PARA OPTAR EL TÍTULO DE
INGENIERO ELECTRÓNICO

APROBADO POR EL SIGUIENTE JURADO:

Ing. HUGO JAVIER CHICLAYO PADILLA
PRESIDENTE

Ing. CARLOS LEONARDO OBLITAS VERA
SECRETARIO

Mag. Ing. OSCAR UCHELLY ROMERO CORTEZ
VOCAL

UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS

TESIS

DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO DIDÁCTICO DE
CONTROL DE TEMPERATURA PARA LA ESCUELA
PROFESIONAL DE INGENIERÍA ELECTRÓNICA EN LA UNPRG

TESIS PARA OPTAR EL TÍTULO DE
INGENIERO ELECTRÓNICO

SUSTENTADO POR:

Bach. LUIS ALONSO SANCHEZ CARLOS
TESISTA

Bach. DANNY ADRIAN CRUZ IRIGOIN
TESISTA

Ing. MARTÍN AUGUSTO NOMBERRA LOSSIO
PATROCINADOR

DEDICATORIA

A Dios, por haberme permitido llegar hasta este punto y haberme dado salud para lograr mis objetivos, además de su infinita bondad y amor.

A mis Padres TEODORO SANCHEZ CARLOS y ARACELLY CARLOS CESPEDES, por haberme apoyado en todo momento, para que yo pudiera lograr mis sueños, por motivarme y creer en Mí y Por el valor mostrado para salir adelante.

A mi familia en general, porque Me han brindado su apoyo incondicional y por compartir conmigo buenos y malos momentos.

LUIS ALONSO SANCHEZ CARLOS

DEDICATORIA

*A Dios, por ser mi guía espiritual,
por darme las fuerzas de seguir
adelante, encarando las adversidades
sin desfallecer en el intento para
alcanzar mis objetivos.*

*A mi padre FAUSTINO CRUZ
AGUIRRE y a mi madre linda MARÍA
ISABEL IRIGOIN GUERRERO,
porque hicieron de mí la persona que
ahora soy, por creer en mí y apoyarme
incondicionalmente en todos los aspectos
posibles, haciendo posible alcanzar este
nuevo objetivo en mi vida profesional.*

*A mis hermanos y de más familiares
por el apoyo y confianza que me
brindaron día a día en el transcurso
de cada año de mi carrera universitaria.*

DANNY ADRIAN CRUZ IRIGOIN

AGRADECIMIENTO

*Expresamos nuestro profundo agradecimiento a:
Todos los familiares, amigos y docentes, que de alguna manera u
otra hicieron posible el
desarrollo de esta presente tesis, que con sus consejos, apoyo y
motivación, alcanzamos
un logro más en nuestras vidas como profesionales.*

RESUMEN

En el mercado industrial actual existe una alta competitividad por elevar la productividad y la calidad de los productos finales. Por estos motivos, se encuentra en una constante búsqueda por mejorar el control de sus procesos y para ello requiere de profesionales que se encuentren mejor capacitados en temas relacionados a la instrumentación y el control de procesos.

Debido a esta necesidad, existen diversas empresas que fabrican módulos didácticos que permiten desarrollar las experiencias de control requeridas para ser competitivo dentro del ámbito industrial actual. Sin embargo, la mayoría de instituciones educativas no cuentan con los recursos económicos suficientes para la adquisición de uno de estos módulos debido al alto costo de inversión que esto significa.

Por ello, el objetivo que se busca al desarrollar este proyecto es diseñar e implementar un módulo didáctico de control de temperatura para el laboratorio de la escuela profesional de Ingeniería electrónica en la UNPRG. Una vez concluido, este módulo permitirá desarrollar conocimientos teórico-prácticos en el campo de instrumentación industrial, así como en el manejo de los equipos involucrados.

Para seleccionar adecuadamente los componentes que forman parte del módulo, se establecieron como principales referentes su funcionabilidad y su costo de adquisición. Una vez realizada la implementación se obtuvo el modelo matemático de la planta y se desarrolló el algoritmo de control PI, debido a que es frecuentemente empleado por las industrias. Este algoritmo fue implementado utilizando conceptos de control por computadora, y le permitió al sistema operar correctamente lazos de control dentro del rango de temperaturas de 25.8 °C a 73.2 °C.

Además, se implementó una interfaz de usuario que permite visualizar gráficamente del comportamiento de las variables del sistema.

ABSTRACT

In the industrial market there is a high competitiveness to raise productivity and the quality of the final product. For these reasons, is in a constant search to improve the control of their processes and for this requires professionals who are better trained in issues related to instrumentation and process control.

Due to this need, there are several companies that manufacture didactic modules that allow to develop the control experiences required to be competitive within the current industrial scope. However, most educational institutions do not have enough financial resources to acquire one of these modules due to the high investment cost that this means.

Therefore, the objective sought in developing this project is to design and implement a didactic module for temperature control for the laboratory of the professional school of electronic engineering at UNPRG. Once concluded, this module will develop theoretical-practical knowledge in the field of industrial instrumentation, as well as in the management of the teams involved.

In order to properly select the components that are part of the module, their functionality and their acquisition cost were established as the main references. Once the implementation was made, the mathematical model of the plant was obtained and the PI control algorithm was developed, because it is frequently used by industries. This algorithm was implemented using computer control concepts, and allowed the system to correctly operate control loops within the temperature range of 25.8 ° C to 73.2 ° C.

In addition, a user interface was implemented that allows graphical visualization of the behavior of the system variables.

ÍNDICE GENERAL

INTRODUCCIÓN	XV
CAPÍTULO I: ASPECTOS DE LA INVESTIGACIÓN	1
1.1. Aspecto informativo	2
1.2. Aspecto de la Investigación	3
1.2.1. Situación Problemática	3
1.2.2. Antecedentes Bibliográficos.....	3
1.2.3. Formulación del problema científico	5
1.2.4. Objetivos	5
1.2.4.1. Objetivo General	5
1.2.4.2. Objetivos específicos.....	6
1.2.5. Justificación e importancia de la investigación	6
1.2.6. Hipótesis.....	7
1.2.7. Diseño metodológico	7
CAPÍTULO II: MARCO TEÓRICO	8
2.1. Teoría de control automático	9
2.1.1. Sistema de control de lazo abierto.....	10
2.1.2. Sistema de control de lazo cerrado.....	11
2.2. Módulos didácticos para el control de procesos industriales	12
2.2.1. Tipos de módulos didácticos de control.....	12
2.2.2. Finalidad de un módulo didáctico de control.....	15
2.3. Sistema SCADA.....	16
2.4. Arduino	19
2.5. Métodos para identificar la función de transferencia	32
2.5.1. Métodos de identificación experimental	33
2.5.1.1. Identificación mediante la respuesta a una entrada escalón	34
2.5.1.1.1. Sistema con respuesta sobreamortiguada	34
2.5.1.1.2. Sistema con respuesta subamortiguada	39
2.5.1.2. Identificación mediante la respuesta en frecuencia	42
2.5.1.3. Identificación mediante mínimos cuadrados.....	45
2.5.2. Determinación de la función de transferencia utilizando software	46
2.6. Método de sintonización de controladores PID	48
2.6.1. Reglas de Ziegler y Nichols	49
2.6.1.1. Primer método	49

2.6.1.2. Segundo método	51
2.6.2. Método de sintonización por ensayo sistemático	53
2.6.2.1. Sintonización de un controlador con acción proporcional	53
2.6.2.2. Sintonización de un controlador con acción proporcional más integral	53
2.6.2.3. Sintonización de un controlador con acción proporcional más derivativo.....	54
2.6.2.4. Sintonización de un controlador con acción proporcional, integral y derivativa	54
2.6.2.5. Efecto de los parámetros de sintonía	55
2.7. Sensores de temperatura	56
2.7.1. Termostato	56
2.7.1.1. Termostato eléctrico	56
2.7.1.2. Termostato mecánico	57
2.7.1.3. Termostato digital	57
2.7.2. Termistores.....	58
2.7.2.1. Termistores PTC	58
2.7.2.2. Termistores NTC.....	60
2.7.3. Termopares	60
2.7.4. RTD (Resistance Temperature Detector)	63
2.8. LabVIEW	65
CAPÍTULO III: DISEÑO DEL SISTEMA	72
3.1. Diagrama de bloques del sistema de control.....	73
3.2. Determinación del modelo matemático de la planta	73
3.2.1. Rango de linealidad de la planta	73
3.2.2. Obtención del modelo mediante la respuesta al escalón.....	75
3.3. Determinación y simulación del controlador en Matlab	79
3.4. Diseño de la estructura del módulo didáctico	83
3.5. Configuración de Arduino	87
3.6. Configuración de LabVIEW.....	88
3.7. Diseño de la interfaz gráfica en LabVIEW	88
CAPÍTULO IV: CONCLUSIONES Y RECOMENDACIONES.....	90
4.1. Conclusiones	91
4.2. Recomendaciones	91
BIBLIOGRAFÍA	92
ANEXOS.....	94

ÍNDICE DE FIGURAS

Figura 1: Sistema de Control en Lazo Abierto	10
Figura 2: Sistema de Control en Lazo Cerrado	11
Figura 3: Ejemplo de un Módulo Didáctico de control	13
Figura 4: Módulo de Control de presión	14
Figura 5: Control de temperatura en un intercambiador de calor	14
Figura 6: Módulo de Control de Nivel de líquido	15
Figura 7: Ejemplo de una pantalla en pvBrowser	17
Figura 8: Ejemplo de una pantalla en Scada Factory Talk	17
Figura 9: Ejemplo de una pantalla en LabVIEW	18
Figura 10: Ejemplo de una pantalla en Wonderware in Touch	18
Figura 11: Ejemplo de pantalla en Vijeo Designer	19
Figura 12: Encapsulado tipo DIP	24
Figura 13: Comunicación I ² C	27
Figura 14: Comunicación SPI	29
Figura 15: Adaptador de alimentación	31
Figura 16: Respuesta a la entrada escalón	35
Figura 17: Respuesta a la entrada escalón	36
Figura 18: Respuesta a la entrada escalón	37
Figura 19: Método de Strejc	38
Figura 20: Tabla de Strejc	38
Figura 21: Respuesta al escalón	39
Figura 22: Respuesta al escalón	40
Figura 23: Respuesta al escalón	41
Figura 24: Gráfica de respuesta en frecuencia - Magnitud	42
Figura 25: Gráfica de respuesta en frecuencia - Fase	42
Figura 26: Gráfica de un sistema con retardo.....	44
Figura 27: Respuesta en frecuencia de un sistema con retardo.....	44
Figura 28: Identificación mediante mínimos cuadrados	45
Figura 29: Señales de prueba.....	45
Figura 30: Importación de datos en Ident	47
Figura 31: Respuesta a la entrada escalón en Ident	47
Figura 32: Función de transferencia en Ident.....	48
Figura 33: Control PID de una planta.....	48

Figura 34: Respuesta a un escalón unitario de una planta	49
Figura 35: Curva de respuesta en forma de S.....	50
Figura 36: Sistema en lazo cerrado con un controlador proporcional	51
Figura 37: Oscilación sostenida con periodo P_{cr}	52
Figura 38: Termostato bimetálico.....	57
Figura 39: Termostato digital.....	58
Figura 40: Curva de respuesta de un PTC	59
Figura 41: Curva de respuesta de un NTC.....	60
Figura 42: Efecto seebeck	60
Figura 43: Termopares industriales	61
Figura 44: Termopar tipo sonda	61
Figura 45: Pt-100.....	64
Figura 46: Conexión a dos hilos	64
Figura 47: Error de medición – conexión a dos hilos.....	64
Figura 48: Conexión a cuatro hilos.....	65
Figura 49: LabVIEW	66
Figura 50: Panel Frontal de LabVIEW.....	67
Figura 51: Paleta de controles en LabVIEW	67
Figura 52: Controles e indicadores numéricos	68
Figura 53: Controles e indicadores Booleanos	68
Figura 54: Controles e indicadores de caracteres.....	69
Figura 55: Ejemplo de un Diagrama de Bloques y Panel Frontal.....	69
Figura 56: Diagrama de Bloques	70
Figura 57: Paleta de funciones.....	71
Figura 58: Sistema en lazo abierto.....	73
Figura 59: Curva estática de la planta.....	74
Figura 60: Curva estática de la planta para un cambio de 0% a 40% de potencia	75
Figura 61: Curva estática de la planta para un cambio de 0% a 90% de potencia	76
Figura 62: Curva estática de la planta para un cambio de 0% a 100% de potencia	77
Figura 63: Código fuente en Matlab	79
Figura 64: Respuesta al escalón del sistema en lazo cerrado.....	80
Figura 65: Código para obtener la respuesta al escalón con $K_p=3$	80
Figura 66: Respuesta al escalón con $K_p=3$	81
Figura 67: Código para obtener la respuesta al escalón con $K_p=3$ y $K_i=4$	81
Figura 68: Respuesta al escalón con $K_p=3$ y $K_i=4$	82

Figura 69: Fuente de poder ATX	83
Figura 70: Conexión entre TT y Arduino	83
Figura 71: Circuito del Dimmer	84
Figura 72: Circuito de control del ventilador	84
Figura 73: Cámara térmica.....	85
Figura 74: Borneras de conexión	85
Figura 75: Interruptor de poder.....	85
Figura 76: Puerto USB	86
Figura 77: Potenciómetro	86
Figura 78: Estructura final del Módulo Didáctico	87
Figura 79: LabVIEW Interface for Arduino - LIFA	87
Figura 80: Buscador del VI Package Manager.....	88
Figura 81: Interfaz de usuario en LabVIEW.....	88
Figura 82: Código en LabVIEW	89

ÍNDICE DE TABLAS

Tabla N° 01: Respuesta escalón de la planta	51
Tabla N° 02: Regla de sintonía con la ganancia crítica K_{cr} y periodo crítico P_{cr}	52
Tabla N° 03: Valores obtenidos haciendo pruebas experimentales.....	74
Tabla N° 04: Valores obtenidos para un cambio de 0% a 40% de potencia	75
Tabla N° 05: Valores obtenidos para un cambio de 0% a 90% de potencia	76
Tabla N° 06: Valores obtenidos para un cambio de 0% a 100% de potencia	77

INTRODUCCIÓN

El continuo desarrollo del mercado a nivel industrial da lugar a la búsqueda constante de la optimización de los procesos de producción. Por ello la industria actual requiere, cada vez con mayor importancia, de profesionales capacitados que puedan realizar adecuadamente el control de los diversos procesos de producción. Por este motivo, la enseñanza de temas relacionados a la instrumentación y al control de procesos requiere necesariamente de un trabajo tipo práctico que le permita al alumno adquirir conocimientos en el manejo y funcionamiento de los equipos que forman parte de un determinado proceso, brindándole una mejor visión de lo que se enseña en clases teóricas.

Debido a esta necesidad, en la actualidad existen diversas empresas que fabrican módulos didácticos en los cuales se desarrolla el control de los procesos que se encuentran con mayor frecuencia en las industrias para un mejor aprendizaje de su comportamiento. Sin embargo, son varias las instituciones educativas que no tienen los recursos suficientes para su adquisición debido al alto costo de inversión que esto implica.

En medio de esta situación surge la posibilidad de realizar, la construcción de un sistema de bajo costo con equipos que se encuentran al alcance en el mercado regional. Este sistema serviría como primera etapa para la implementación de un módulo didáctico de control de temperatura para el laboratorio de la escuela profesional de Ingeniería electrónica en la UNPRG. De esta manera se podrá adquirir conocimientos y experiencia en la instrumentación industrial.

En el capítulo 1 se explica los aspectos de la investigación y el contexto del marco problemático por el cual la mayoría de instituciones educativas no cuentan con estos sistemas para realizar pruebas.

En el capítulo 2 se presenta el marco teórico, como la importancia de los módulos didácticos, teoría del Arduino, métodos para obtener la función de transferencia, teoría del sensor de temperatura.

En el capítulo 3 se desarrolla el diseño del Sistema, a partir del diagrama de bloques del sistema de control, la determinación del modelo matemático de la planta a partir de los valores obtenidos en las pruebas experimentales. Y el desarrollo del algoritmo de control PI. Luego se describe el proceso de implementación de los circuitos de acondicionamiento para el sensor y transmisor. Se presenta además el montaje de los componentes mecánicos del sistema.

En el capítulo 4 se muestran las conclusiones y recomendaciones del proyecto.

ASPECTOS DE LA INVESTIGACIÓN

CAPÍTULO I

1.1 Aspecto informativo

1.1.1 Título

Diseño e implementación de un Módulo Didáctico de Control de Temperatura para la Escuela Profesional de Ingeniería Electrónica en la UNPRG

1.1.2 Personal Investigador

1.1.2.1 Autor

Nombre : Bach. Sanchez Carlos Luis Alonso
Dirección : Av. Túpac Amaru N° 640 - Lambayeque
E – mail : alonso.sc14@gmail.com
Teléfono : 998801255

1.1.2.2 Autor

Nombre : Bach. Cruz Irigoin Danny Adrian
Dirección : Ca. José Pardo N° 480 - Chiclayo
E – mail : cruz_irigoin_danny@outlook.com
Teléfono : 978876459

1.1.3 Asesor

Nombre : Ing. Nombera Lossio Martin
Dirección : Calle Alfredo Lapoint N° 446 - Chiclayo
E – mail : jefristya@hotmail.com
Teléfono: 950408011

1.1.4 Área de Investigación

Ingeniería Electrónica – Control y Automatización

1.1.5 Lugar de ejecución

Laboratorio de Ingeniería Electrónica de la Universidad Nacional Pedro Ruíz Gallo - Lambayeque

1.1.6 Duración

04 meses

1.2 Aspecto de la investigación

1.2.1 Situación Problemática

Un módulo didáctico es una adaptación de una planta productiva industrial a escala reducida cuyo objetivo es permitir la experimentación, con la finalidad de obtener resultados sobre algún proceso físico o químico. Mediante los resultados obtenidos con estos módulos es posible comprobar si la implementación de una planta de mayor tamaño es factible y económicamente viable. Adicionalmente, la planta piloto permite probar nuevos productos, verificar alguna hipótesis o comprobar el buen funcionamiento de un prototipo especial de planta en ingeniería.

El tener un módulo didáctico en un centro de estudios superior facilita a estudiantes y profesores el desarrollo de prácticas de instrumentación industrial, sistemas de control y automatización industrial de procesos. Además, permite el desarrollo de prácticas en forma independiente cuando el estudiante lo requiera, permitiendo así el acercamiento de las universidades o institutos técnicos a la industria.

La Escuela Profesional de Ingeniería Electrónica de la Universidad Nacional Pedro Ruiz Gallo no cuenta con un módulo didáctico de control de temperatura que permita a los estudiantes realizar prácticas para controlar esta variable de proceso que es muy común en procesos industriales.

De allí el interés nuestro por realizar un módulo didáctico de control de temperatura que incentive el interés de los estudiantes en el área de automatización y control de procesos.

1.2.2 Antecedentes Bibliográficos

- **Diseño e implementación de un módulo educativo para el control de temperatura (2017)**

Autores: Fuertes Paucar Yossi Giovanni

Universidad: Pontificia Universidad Católica del Perú

Resumen: El objetivo principal de esta tesis es desarrollar e implementar un sistema de control de temperatura de bajo costo en el módulo educativo de control de nivel que se encuentra actualmente

en el Centro de Tecnologías Avanzadas de Manufactura (CETAM) de la PUCP.

La autora de la tesis hace un profundo análisis para seleccionar adecuadamente los componentes que formaran parte del módulo, estableciendo como principales referentes su funcionabilidad y su costo de adquisición.

Una vez realizada la implementación se obtiene el modelo matemático de la planta y se desarrolla el algoritmo de control PID, debido a que es frecuentemente empleado por las industrias. El algoritmo empleado en esta tesis fue implementado utilizando conceptos de control por computadora, lo que permitió al sistema operar correctamente lazos de control dentro del rango de temperaturas de 25 °C a 36 °C. Además, se implementó una interfaz de usuario que permite visualizar gráficamente el comportamiento de las variables del sistema y modificar de manera sencilla los parámetros del controlador que tienen una relación directa con la respuesta.

Con la implementación de dicho módulo se permitirá a los estudiantes desarrollar conocimientos teórico-prácticos en el campo de control de procesos, así como en el manejo de los equipos involucrados.

- **Diseño e implementación de un módulo didáctico para simular y controlar procesos de empacado de sólidos granulados, para el laboratorio de control de la EIE-CRI (2011)**

Autor: Núñez Núñez Mónica Lilian y Sisa Amaguaya Luis Fernando

Universidad: Escuela Superior Politécnica de Chimborazo

Resumen: Este proyecto de tesis surge como respuesta a la necesidad de implementar un laboratorio de procesos industriales en la Escuela de Ingeniería electrónica en Control y Redes Industriales de la Escuela Superior Politécnica de Chimborazo en sus materias de control y Automatización Industrial.

Para dar solución al problema, los tesisistas proponen el diseño, implementación y respectiva documentación de un módulo didáctico para simular y controlar el empacado de sólidos granulados.

El módulo es diseñado con todos los elementos necesarios para simular procesos industriales (sensores, pulsadores, protecciones, fusibles, PLC, entre otros).

Con la implementación de este módulo se logró que los estudiantes puedan aplicar sus conocimientos de control y automatización a un proceso real.

- **Diseño y construcción de un módulo orientado al desarrollo de prácticas para la materia de control automático bajo la plataforma de Matlab**

Autor: Jurado Valencia Edgar Antonio y Salas Bohórquez Gonzalo Xavier

Universidad: Universidad Politécnica Salesiana – Sede Guayaquil

Resumen: El objetivo de esta tesis es ayudar a los estudiantes a entender cómo trabajan dos diferentes sistemas de control desde la plataforma Simulink del software Matlab.

El primer módulo está formado por el sistema que controla el nivel de líquido en un tanque y el segundo módulo está formado por el sistema que controla la temperatura

Estos módulos se complementan con un amplio análisis que se desarrolla con experimentos y pruebas, cuya finalidad es crear un apropiado sistema de prácticas para los estudiantes de dicha universidad.

1.2.3 Formulación del Problema Científico

¿Cómo diseñar e implementar un Módulo Didáctico de Control de Temperatura para la capacitación de los estudiantes de la Escuela Profesional de Ingeniería Electrónica en la UNPRG?

1.2.4 Objetivos

1.2.4.1 Objetivo general

- Diseñar e implementar un Módulo Didáctico de Control de Temperatura para la Escuela Profesional de Ingeniería Electrónica en la UNPRG

1.2.4.2 Objetivos específicos

- Estudiar los módulos didácticos de control de procesos.
- Diseñar e implementar los circuitos acondicionadores para los sensores.
- Obtener el modelo matemático de la planta.
- Implementar un algoritmo de control PID.
- Desarrollar la interfaz de usuario mediante el software LabVIEW.

1.2.5 Justificación e importancia de la investigación

En la actualidad, el mercado industrial requiere que sus procesos sean controlados y monitoreados adecuadamente. Para ello es necesario que los profesionales de hoy en día se encuentren mejor capacitados en los temas de instrumentación y control de procesos. Esto implica necesariamente que la enseñanza teórica de estos temas se complemente con prácticas de laboratorio que aporten una mayor visión de lo aprendido.

El avance que se viene desarrollando en el campo de los módulos didácticos ha permitido que cada vez tengan mayores funcionalidades, por lo que se incrementa el costo de su adquisición. Esto significa un inconveniente para las instituciones educativas que no disponen de grandes recursos, por eso para algunas de ellas es más factible la implementación de uno de estos módulos con las características básicas para cumplir con su finalidad.

Por lo antes dicho, la importancia del proyecto a desarrollar consiste en diseñar e implementar un módulo didáctico de control de temperatura. Este módulo debe ser sencillo, de bajo costo, con una construcción sólida que resista la exposición a las condiciones a que estará sometido durante su operación, deberá ser de fácil operación y no necesitar mucho mantenimiento. Con este módulo los estudiantes podrán experimentar en el control de temperatura, lo que les permitirá acercarse al mundo de la automatización y el control de procesos industriales.

1.2.6 Hipótesis

Con el Diseño e implementación de un Módulo Didáctico de Control de Temperatura para la Escuela Profesional de Ingeniería Electrónica en la UNPRG, lograremos que los estudiantes puedan realizar prácticas y experimentar en el control de procesos industriales.

1.2.7 Diseño metodológico.

El procedimiento para la realización del proyecto comprende las siguientes fases:

- Estudiar los módulos didácticos de control de procesos
- Diseñar e implementar los circuitos acondicionares para los sensores
- Obtener el modelo matemático de la planta
- Implementar un algoritmo de control PID
- Desarrollar la interfaz de usuario mediante el software LabVIEW
- Implementar el módulo didáctico de acuerdo al diseño
- Analizar el desempeño del módulo didáctico a través de pruebas y ensayos

MARCO TEÓRICO

CAPÍTULO II

2.1. Teoría de control automático [1] [2]

El control automático ha jugado un papel vital en el avance de la ciencia y la ingeniería, resulta esencial su uso en la mayoría de los procesos industriales como control de presión, temperatura, caudal, nivel, etc.

Cualquier estudio sobre control de procesos debe empezar por distinguir los diferentes conceptos de «proceso». Desde el punto de vista de producción, se conoce generalmente como un lugar donde materia, y muy a menudo energía, son tratadas para dar como resultado un producto deseado o establecido. Por ejemplo, son procesos de producción: reactores, hornos, intercambiadores de calor, etc.

Desde un punto de vista de control, su significado es más específico. Un proceso es un bloque que se identifica porque tiene una o más variables de salida de las cuales es importante conocer y mantener sus valores. Como consecuencia estas variables han de ser controladas actuando sobre otra serie de variables denominadas manipuladas.

Asociado al control aparecen tres tipos de variables que es necesario conocer para controlar los procesos:

- *Variable controlada (CV):* Es la característica de calidad o cantidad que se mide y controla. La variable controlada es una condición o característica del medio controlado, entendiendo por tal la materia o energía sobre la cual se encuentra situada esta variable. Por ejemplo, cuando se controla automáticamente la temperatura del agua de un intercambiador de calor, la variable controlada es la temperatura mientras que el agua es el medio controlado.
- *Variable manipulada (MV):* Es la cantidad o condición de materia o energía que se modifica por el controlador automático para que el valor de la variable controlada resulte afectado en la proporción debida. La variable manipulada es una condición o característica de la materia o energía que entra al proceso. Por ejemplo, en un intercambiador de calor cuando el elemento final de control modifica el caudal de vapor al proceso, la variable manipulada es el caudal mientras que el vapor es la energía de entrada.
- *Variable de perturbación (DV):* Es toda variable que tiene influencia sobre la variable controlada pero no puede ser modificada directamente por la variable manipulada. Por ejemplo la temperatura de entrada de agua en un intercambiador de calor. El sistema tiene que esperar a que el cambio en la temperatura del agua alcance la salida del proceso, donde se encuentra

situada la variable controlada, para que ésta realice la acción correspondiente sobre la variable manipulada.

En la actualidad y gracias al avance de la tecnología, se han desarrollado sistemas de control digitales con los cuales se logra un funcionamiento óptimo de los procesos industriales.

2.1.1. Sistemas de control en lazo abierto

Un sistema de control en lazo abierto funciona sin retroalimentación y genera directamente la salida en respuesta a una señal de entrada. En estos sistemas la salida no tiene efecto sobre la acción de control.

En otras palabras, en un sistema de control en lazo abierto no se mide la salida ni se realimenta para compararla con la entrada. Un ejemplo práctico es una lavadora. El remojo, el lavado y el centrifugado en la lavadora operan con una base de tiempo. La máquina no mide la señal de salida, que es la limpieza de la ropa.

En cualquier sistema de control en lazo abierto, la salida no se compara con la entrada de referencia. Así, a cada entrada de referencia le corresponde una condición de operación fija; como resultado de ello, la precisión del sistema depende de la calibración. Ante la presencia de perturbaciones, un sistema de control en lazo abierto no realiza la tarea deseada. En la práctica, el control en lazo abierto sólo se usa si se conoce la relación entre la entrada y la salida y si no hay perturbaciones internas ni externas. Es evidente que estos sistemas no son de control realimentado.

Obsérvese que cualquier sistema de control que opere con una base de tiempo está en lazo abierto. Por ejemplo, el control de tráfico mediante señales operadas con una base de tiempo es otro ejemplo de control en lazo abierto.



Figura 1: Sistema de Control en Lazo Abierto

2.1.2. Sistemas de control en lazo cerrado

Un sistema de control en lazo cerrado es aquel que utiliza una medida de la señal de salida y una comparación con la salida deseada para generar una señal de error que se aplica al actuador.

Los sistemas de control en lazo cerrado se denominan también sistemas de control realimentados. En la práctica, los términos control realimentado y control en lazo cerrado se usan indistintamente. En un sistema de control en lazo cerrado, se alimenta al controlador la señal de error de actuación, que es la diferencia entre la señal de entrada y la señal de realimentación (que puede ser la propia señal de salida o una función de la señal de salida y sus derivadas y/o integrales), con el fin de reducir el error y llevar la salida del sistema a un valor deseado. El término control en lazo cerrado siempre implica el uso de una acción de control realimentado para reducir el error del sistema.

Los sistemas de control realimentados no se limitan a la ingeniería, sino que también se encuentran en diversos campos ajenos a ella. Por ejemplo, el cuerpo humano es un sistema de control realimentado muy avanzado. Tanto la temperatura corporal como la presión sanguínea se conservan constantes mediante una realimentación fisiológica.

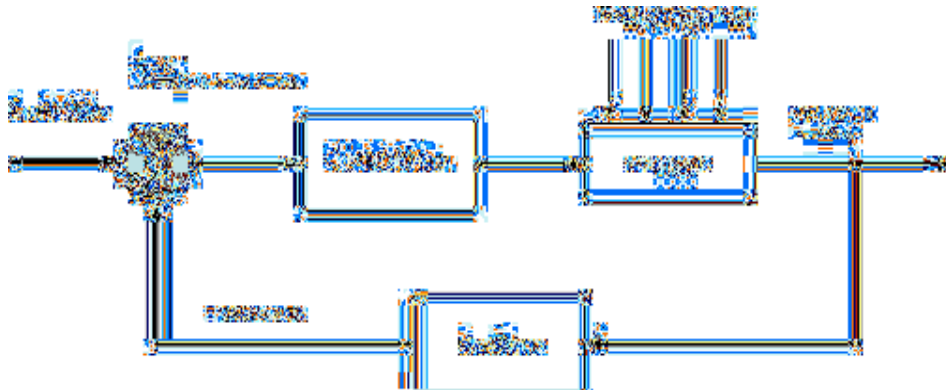


Figura 2: Sistema de Control en Lazo Cerrado.

2.2. Módulos didácticos para el control de procesos industriales [3]

Un módulo didáctico para el control de procesos industriales es una planta de proceso a escala reducida. El fin que se persigue al diseñar, construir y operar una planta piloto es obtener información sobre un determinado proceso físico o químico, que permita determinar si el proceso es técnica y económicamente viable, así como establecer los parámetros de operación óptimos de dicho proceso para el posterior diseño y construcción de la planta a escala industrial.

Un módulo didáctico debe ser mucho más flexible que una planta a escala industrial en cuanto al rango permisible de sus parámetros de operación o variables de proceso, ya que una planta industrial opera siempre en las mismas condiciones, mientras que una planta piloto, por el hecho de estar destinada a la investigación o estudio de un proceso, debe permitir trabajar en un amplio rango de valores de temperatura, presión, etc., de manera que puedan realizarse experimentos o ensayos con valores distintos de las variables de proceso y poder determinar así los valores óptimos.

Los módulos didácticos pueden combinarse y agregar nuevas variables a medir, ya que pueden emular un proceso industrial el cual involucra diferentes sistemas dependiendo del proceso.

El desarrollo de un módulo didáctico está relacionado con una propuesta de enseñanza orientada al desarrollo de competencias. Donde competencia se puede definir como la capacidad de articular eficazmente un conjunto de esquemas (estructuras mentales) y valores, permitiendo movilizar (poner a disposición) distintos saberes en un determinado contexto con el fin de resolver situaciones profesionales.

2.2.1. Tipos de módulos didácticos de control

Un módulo didáctico permite el estudio del control de procesos continuos, soliendo ser las variables típicas a medir: presión, temperatura, humedad, flujo, nivel, etc. El módulo es fabricado con componentes industriales y comerciales reales, fácilmente identificados por los estudiantes y comúnmente utilizados en plantas industriales.

Las principales áreas que se tratan de refinar mediante la experimentación son relativas a producto, materias primas, reacciones, separación y sistemas auxiliares.



Figura 3: Ejemplo de un Módulo Didáctico de Control

2.2.1.1. Módulo didáctico para el control de presión

La presión se define como la fuerza normal que, distribuida uniformemente, actúa sobre una superficie dada.

La fuerza puede estar ocasionada por líquidos, gases o vapores, o por cuerpos sólidos, y su unidad de medida, en el Sistema Internacional (SI), es el Newton.

Conocida esta fuerza y utilizando como unidad de superficie el metro cuadrado se obtiene como unidad de presión el Pascal, nombre debido al físico francés del siglo XVII, Blas Pascal.

La medición de presión dentro de un proceso es importante ya que nos permite ver cuanta potencia nos puede entregar o que tanto más pueden soportar nuestros instrumentos.

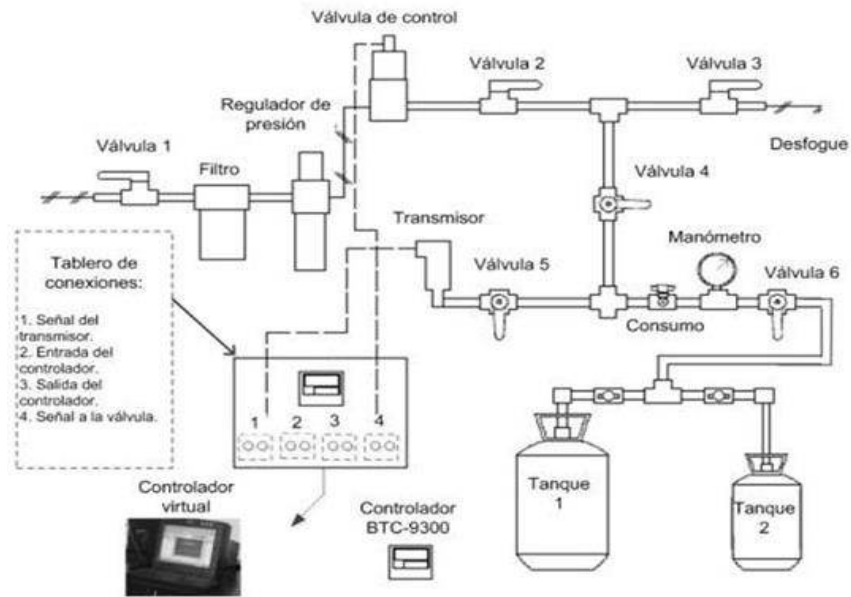


Figura 4: Módulo de Control de presión.

2.2.1.2. Módulo didáctico para el control de temperatura

Una variable importante en los procesos industriales suele ser la de temperatura, ya que esta puede influir de manera importante en el proceso, causando problemas y hasta accidentes si no se controla de manera adecuada.

En la siguiente figura se muestra el diagrama de instrumentación de un módulo didáctico que testea la temperatura en el proceso junto a un intercambiador de calor.



Figura 5: Control de Temperatura en un intercambiador de calor.

2.2.1.3. Módulo didáctico para el control de flujo y nivel

Es común que cuando se requiera mantener en un nivel determinado algún fluido se controle el flujo de este, tanto de entrada como salida, que pasará al recipiente de almacenaje. Es por eso que estas variables suelen medirse juntas.

Un esquema simple de este proceso sería el mostrado a continuación.

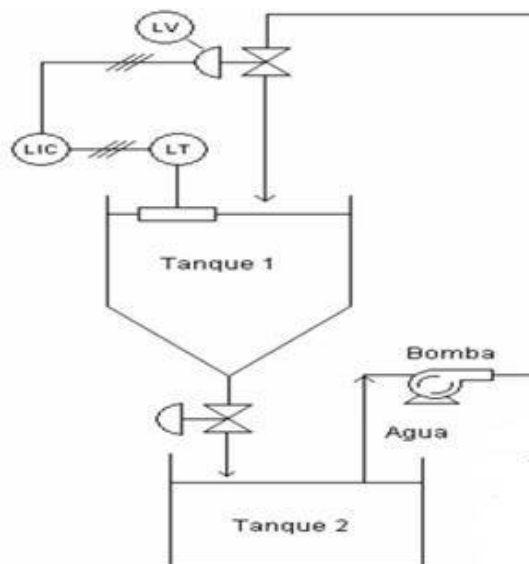


Figura 6: Módulo de Control de Nivel de líquido.

2.2.2. Finalidad de un módulo didáctico de control

La función principal de un módulo didáctico es permitir que los estudiantes puedan controlar variables industriales a través de la experimentación y la práctica. Además, permite llevar a cabo la investigación de parámetros que muestren la información sobre la aplicabilidad de este sistema a nivel industrial.

Una vez que se haya concluido el trabajo experimental en el módulo didáctico, se desprenderá la siguiente información:

- El diagrama de bloques del proceso.
- Reconocimiento de las variables de proceso y como pueden afectar al mismo.
- Reconocimiento de las variables de perturbación y sus efectos sobre el proceso.
- Determinación del tipo de control que se debe aplicar al proceso.
- Los materiales de construcción más adecuados para la planta.
- Los costos estimados de construcción y operación de la planta.

2.3. Sistema SCADA [4]

Un sistema SCADA (Sistema de Control, Supervisión y Adquisición de Datos) es un sistema de control, medición y monitoreo que consiste en una estación maestra (MTU), una o más unidades remotas para control y recolección de datos de campo (RTUs) y un conjunto de paquetes de software usados para monitorear y controlar a los elementos remotamente localizados. Un sistema SCADA provee toda la información que se genera en el proceso productivo a los diversos usuarios.

Un sistema SCADA es usado para controlar y monitorear una planta o equipo, el control puede ser automático o iniciado por comandos del operador. La adquisición de datos es iniciada por las RTUs que hacen un barrido u exploración a alta velocidad de las entradas de campo conectadas a las mismas, posteriormente el MTU hace un barrido de los datos de la RTUs a menor velocidad, de esta manera el dato es procesado.

Un SCADA debe cumplir con varios objetivos para que su instalación sea perfectamente aprovechada:

- Deben ser sistemas de arquitectura abierta, capaces de crecer o adaptarse según las necesidades cambiantes del proceso.
- Debe comunicarse con total facilidad al usuario, con el equipo y con el resto de la planta (redes locales).
- Deben ser programas sencillos de instalar, sin excesivas exigencias de hardware, fáciles de utilizar y con interfaces amigables con el usuario.

La implantación de un sistema SCADA requiere trabajar a tres niveles dentro del entorno piramidal de automatización integrada, permitiendo el intercambio de información entre los elementos de la propia capa o de niveles inmediatamente superior o inferior.

- *Nivel de supervisión:* Este nivel nos permite monitorear el proceso, fijándonos el rendimiento, funcionalidad, entre otros; con lo cual podemos garantizar el correcto funcionamiento.
- *Nivel de control:* En este nivel se encuentran todos aquellos elementos que controlan el proceso, por ejemplo los PLC, controladores PID, etc. Estos están diseñados para manejar el proceso de manera correcta según convenga.
- *Nivel de proceso:* Aquí encontramos a todos los elementos que tienen un contacto directo con el proceso como son los sensores y actuadores, los cuales ejecutan acciones o captan señales que intervienen en el sistema.

Existen diversos programas para elaborar sistemas SCADA, independientes o que vienen asociados a otros programas, algunos son:

- *pvBrowser*: Aplicación GPL para monitorización SCADA con interfaz web.



Figura 7: Ejemplo de una pantalla en pvBrowser

- *FreeSCADA*: Aplicación Open source para proyectos SCADA.
- *Scada Argos*: Proyecto de SCADA para Linux.
- *Scada Factory Talk View SE*: Perteneciente a la empresa Rockwell Automation. Tiene arquitectura escalable desde una sola estación HMI a multi-cliente / sistema de multi-servidor. Puede maximizar la disponibilidad del sistema con una función de detección y recuperación de errores en tiempo real y tendencias históricas con conectividad integrada para Factory Talk Historian.



Figura 8: Ejemplo de una pantalla en Scada Factory Talk.

- *LabVIEW*: Perteneciente a la empresa National Instruments, es un software con múltiples funcionalidades que ofrece dos pantallas para trabajar, una donde se realiza la interfaz de usuario y la otra donde enlazamos cada herramienta con otra, definiendo su función.

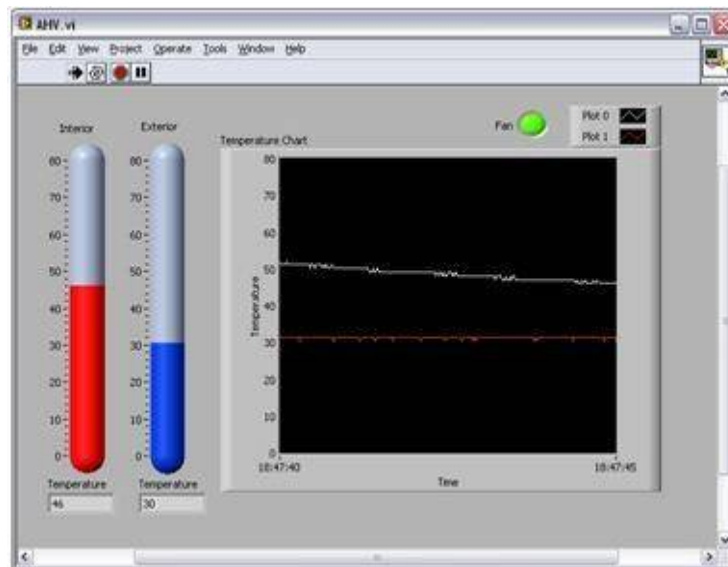


Figura 9: Ejemplo de una pantalla en LabVIEW.

- *Wonderware InTouch HMI*: Software robusto capaz de realizar monitoreo continuo con gráficos de resolución independiente y símbolos inteligentes. Cuenta con un sofisticado sistema de scripting para extender y personalizar aplicaciones.



Figura 10: Ejemplo de una pantalla en Wonderware InTouch.

- *Vijeo Designer*: Perteneciente a la empresa Schneider Electric, dedicado a la configuración de HMI Magelis a través del software Vijeo Designer. Este SCADA es óptimo para el trabajo con los PLC de esta empresa como los Modicon.

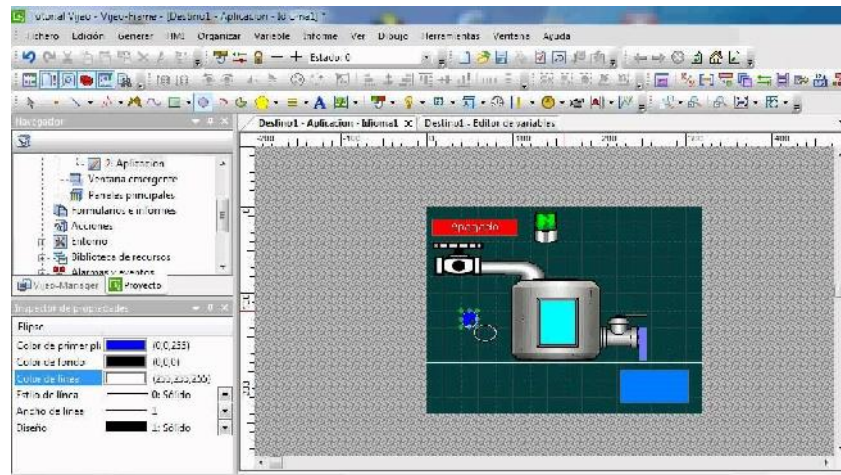


Figura 11: Ejemplo de pantalla en Vijeo Designer.

2.4. Arduino [5]

Cada vez existen más productos domésticos que incorporan algún tipo de microcontrolador con el fin de aumentar sustancialmente sus prestaciones, reducir su tamaño y coste, mejorar su fiabilidad y disminuir el consumo. Así, podemos encontrar microcontroladores dentro de multitud de dispositivos electrónicos que usamos en nuestra vida diaria, como pueden ser desde un simple timbre hasta un completo robot pasando por juguetes, frigoríficos, televisores, lavadoras, microondas, impresoras, el sistema de arranque de un coche, etc.

Uno de los productos basados en microcontrolador que revolucionó el área de la electrónica es Arduino.

Arduino nació en el año 2005 en el Instituto de Diseño Interactivo de Ivrea (Italia), centro académico donde los estudiantes se dedicaban a experimentar con la interacción entre humanos y diferentes dispositivos (muchos de ellos basados en microcontroladores) para conseguir generar espacios únicos, especialmente artísticos. Arduino apareció por la necesidad de contar con un dispositivo para utilizar en las aulas que fuera de bajo coste, que funcionase bajo cualquier sistema operativo y que contase con documentación adaptada a gente que quisiera empezar de cero. La idea original fue fabricar la placa para el uso interno de la escuela.

No obstante, el Instituto se vio obligado a cerrar sus puertas precisamente en 2005. Ante la perspectiva de perder en el olvido todo el desarrollo del proyecto Arduino que se había ido llevando a cabo durante aquel tiempo, se decidió liberarlo y abrirlo a "la comunidad" para que todo el mundo tuviera la posibilidad de participar en la evolución del proyecto, proponer mejoras y sugerencias y mantenerlo vivo. Y así ha sido, la colaboración de muchísima gente ha hecho que Arduino poco a poco haya llegado a

ser lo que es actualmente: un proyecto de hardware y software libre de ámbito mundial.

El principal responsable de la idea y diseño de Arduino, y la cabeza visible del proyecto es el llamado “Arduino Team”, formado por Massimo Banzi (profesor en aquella época del Instituto Ivrea), David Cuartielles (profesor de la Escuela de Artes y Comunicación de la Universidad de Malmö, Suecia), David Mellis (por aquel entonces estudiante en Ivrea y actualmente miembro del grupo de investigación High – Low Tech del MIT Media Lab), Tom Igoe (profesor de la Escuela de Arte Tisch de Nueva York), y Gianluca Martino (responsable de la empresa fabricante de los prototipos de las placas).

Las placas Arduino cuentan con muchas funcionalidades, pero sus características más importantes son:

- Es una placa de hardware libre: Arduino incorpora un microcontrolador reprogramable y una serie de pines hembra (los cuales están unidos internamente a las patillas de E/S del microcontrolador) que permiten conectar allí de forma muy sencilla y cómoda diferentes sensores y actuadores.
- Cuando hablamos de “placa hardware” nos estamos refiriendo en concreto a una PCB (del inglés “Printed Circuit Board”, o sea, placa de circuito impreso). Las PCBs son superficies fabricadas de un material no conductor (normalmente resinas de fibra de vidrio reforzada, cerámica o plástica) sobre las cuales aparecen laminadas pistas de material conductor (normalmente cobre). Las PCBs se utilizan para conectar eléctricamente, a través de los caminos conductores, diferentes componentes electrónicos soldados a ella. Una PCB es la forma más compacta y estable de construir un circuito electrónico, pero una vez fabricada su diseño es bastante difícil de modificar. Así pues, la placa Arduino no es más que una PCB que implementa un determinado diseño de circuitería interna.
- Cuando hablamos de “placa Arduino”, se debe especificar el modelo concreto, ya que existen varias placas Arduino oficiales, cada una con diferentes características (como el tamaño físico, el número de pines hembra ofrecidos, el modelo de microcontrolador incorporado, la cantidad de memoria utilizable, etc.). Conviene conocer estas características para identificar qué placa Arduino es la que nos convendrá más en cada proyecto.
- De todas formas, aunque puedan ser modelos específicos, los microcontroladores incorporados en las diferentes placas Arduino pertenecen todos a la misma “familia tecnológica”, por lo que su funcionamiento en

realidad es bastante parecido entre sí. En concreto, todos los microcontroladores son de tipo AVR, una arquitectura de microcontroladores desarrollada y fabricada por la marca Atmel.

- El diseño hardware de la placa Arduino está inspirado originalmente en el de otra placa de hardware libre preexistente, la placa Wiring. Esta placa surgió en 2003 como proyecto personal de Hernando Barragán, estudiante por aquel entonces del Instituto de Diseño de Ivrea (lugar donde surgió en 2005 precisamente la placa Arduino).
- Es un software gratis, libre y multiplataforma: Arduino es un software (funciona en Linux, MacOS y Windows) que debemos instalar en nuestro ordenador y que nos permite escribir, verificar y guardar en la memoria del microcontrolador de la placa Arduino el conjunto de instrucciones que deseamos que este empiece a ejecutar.
- La manera estándar de conectar nuestro computador con la placa Arduino para poder enviarle y grabarle dichas instrucciones es mediante un simple cable USB, gracias a que la mayoría de placas Arduino incorporan un conector de este tipo.
- Los proyectos Arduino pueden ser autónomos o no. En el primer caso, una vez programado su microcontrolador, la placa no necesita estar conectada a ningún computador y puede funcionar autónomamente si dispone de alguna fuente de alimentación. En el segundo caso, la placa debe estar conectada de alguna forma permanente (por cable USB, por cable de red Ethernet, etc.) a un computador ejecutando algún software específico que permita la comunicación entre este y la placa y el intercambio de datos entre ambos dispositivos. Este software específico lo deberemos programar generalmente nosotros mismos mediante algún lenguaje de programación estándar como Python, C, Java, Php, etc., y será independiente completamente del entorno de desarrollo Arduino, el cual no se necesitará más una vez que la placa ya haya sido programada y esté en funcionamiento.
- Tiene lenguaje de programación libre: Por “lenguaje de programación” se entiende cualquier idioma artificial diseñado para expresar instrucciones (siguiendo unas determinadas reglas sintácticas) que pueden ser llevadas a cabo por máquinas. Concretamente dentro del lenguaje Arduino, encontramos elementos parecidos a muchos otros lenguajes de programación existentes (como los bloques condicionales, los bloques repetitivos, las variables, etc.), así como también diferentes comandos llamados “órdenes” o “funciones” que nos permiten especificar de una forma coherente y sin errores las

instrucciones exactas que queremos programar en el microcontrolador de la placa. Estos comandos los escribimos mediante el entorno de desarrollo Arduino.

- Tanto el entorno de desarrollo como el lenguaje de programación Arduino están inspirado en otro entorno y lenguaje libre preexistente: Processing, desarrollado inicialmente por Ben Fry y Casey Reas.
- Que el software Arduino se parezca tanto a Processing no es casualidad, ya que este está especializado en facilitar la generación de imágenes en tiempo real, de animaciones y de interacciones visuales, por lo que muchos profesores del Instituto de Diseño de Ivrea lo utilizaban en sus clases. Como fue en ese centro donde precisamente se inventó Arduino es natural que ambos entornos y lenguajes guarden bastante similitud. No obstante, hay que aclarar que el lenguaje Processing está construido internamente con código escrito en lenguaje Java, mientras que el lenguaje Arduino se basa internamente en código C/C++.
- Con Arduino se pueden realizar multitud de proyectos de rango muy variado, desde robótica hasta domótica, pasando por monitorización de sensores ambientales, sistemas de navegación, telemática, etc. Realmente, las posibilidades de esta plataforma para el desarrollo de productos electrónicos son prácticamente infinitas y tan solo están limitadas por nuestra imaginación.

Existen muchas otras placas de diferentes fabricantes que, aunque incorporan diferentes modelos de microcontroladores, son comparables y ofrecen una funcionalidad más o menos similar a la de las placas Arduino. Todas ellas también vienen acompañadas de un entorno de desarrollo agradable y cómodo y de un lenguaje de programación sencillo y completo. No obstante, la plataforma Arduino (hardware y software) ofrece una serie de ventajas:

- *Arduino es libre y extensible:* Esto quiere decir que cualquiera que desee ampliar y mejorar tanto el diseño hardware de las placas como el entorno de desarrollo software y el propio lenguaje de programación, puede hacerlo sin problemas. Esto permite que exista un rico “ecosistema” de extensiones, tanto de variantes de placas no oficiales como de librerías software de terceros, que pueden adaptarse mejor a nuestras necesidades concretas.
- *Arduino tiene una gran comunidad:* Muchas personas lo utilizan, enriquecen la documentación y comparten continuamente sus ideas.

- *Su entorno de programación es multiplataforma:* Se puede instalar y ejecutar en sistemas Windows, Mac OS X y Linux. Esto no ocurre con el software de muchas otras placas.
- *Su entorno y el lenguaje de programación son simples y claros:* Son muy fáciles de aprender y de utilizar, a la vez que flexibles y completos para que los usuarios avanzados puedan aprovechar y exprimir todas las posibilidades del hardware. Además, están bien documentados, con ejemplos detallados y gran cantidad de proyectos publicados en diferentes formatos.
- *Las placas Arduino son baratas:* La placa Arduino estándar (llamada Arduino UNO) ya preemsañada y lista para funcionar cuesta alrededor de 20 euros. Incluso, uno mismo se la podría construir (Arduino es hardware libre) adquiriendo los componentes por separado, con lo que el precio total de la placa resultante sería incluso menor.
- *Las placas Arduino son reutilizables y versátiles:* Reutilizables porque se puede aprovechar la misma placa para varios proyectos (ya que es muy fácil de desconectarla, reconectarla y reprogramarla), y versátiles porque las placas Arduino proveen varios tipos diferentes de entradas y salidas de datos, los cuales permiten capturar información de sensores y enviar señales a actuadores de múltiples formas.

Ya se ha comentado anteriormente que existen varios tipos de placas Arduino, cada una con características específicas que hay que conocer para poder elegir el modelo que más nos convenga según el caso. No obstante, existe un modelo “estándar” de placa, y que es el que utilizaremos para el desarrollo del módulo didáctico: la placa Arduino UNO. Desde que apareció en 2010 ha sufrido tres revisiones, por lo que el modelo actual se suele llamar UNO Rev3 o simplemente UNO R3.

El microcontrolador que lleva la placa Arduino UNO es el modelo ATmega328P de la marca Atmel. La “P” del final significa que este chip incorpora la tecnología “Picopower” (propietaria de Atmel), la cual permite un consumo eléctrico sensiblemente menor comparándolo con el modelo equivalente sin “Picopower”, el ATmega328 (sin la “P”). De todas formas, aunque el ATmega328P pueda trabajar a un voltaje menor y consumir menos corriente que el ATmega328 (especialmente en los modos de hibernación), ambos modelos son funcionalmente idénticos.

Al igual que ocurre con el resto de microcontroladores usados en otras placas Arduino, el ATmega328P tiene una arquitectura de tipo AVR, arquitectura desarrollada por Atmel y en cierta medida “competencia” de otras arquitecturas como por ejemplo la PIC del fabricante Microchip. Más concretamente, el ATmega328P

pertenece a la subfamilia de microcontroladores “megaAVR”. Otras subfamilias de la arquitectura AVR son la “tinyAVR” (cuyos microcontroladores son más limitados y se identifican con el nombre de ATtiny) y la “XMEGA” (cuyos microcontroladores son más capaces y se identifican con el nombre de ATxmega).

La siguiente figura muestra esta disposición en el encapsulado de tipo DIP.

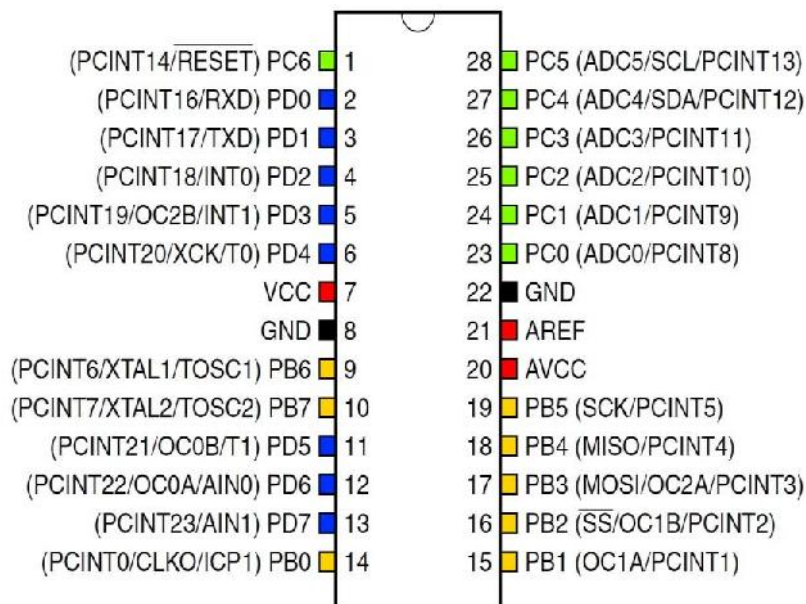


Figura 12: Encapsulado tipo DIP.

Observando la figura anterior se puede saber qué pin es el que recibe la alimentación eléctrica (señalado como “VCC”), qué dos pines están conectados a tierra (los señalados como “GND”), qué pines son los de E/S (señalados como PBx, PCx o PDx) y la existencia de otros pines más específicos como el AVCC (donde se recibe la alimentación suplementaria para el convertidor analógico-digital interno del chip) o el AREF (donde se recibe la referencia analógica para dicho). También se puede observar que junto al nombre de los pines de E/S se indica entre paréntesis las funciones especializadas que cada uno de ellos tiene en particular (además de su función genérica de entrada/salida).

Algunas de estas funciones son por ejemplo la función de “reset” del microcontrolador, o la comunicación con el exterior usando el protocolo serie o el SPI o el I2C, o el uso de interrupciones, o el de las salidas PWM, etc.

Otra cosa que hay que saber de los microcontroladores son los tipos y cantidades de memoria que alojan en su interior. En el caso del ATmega328P tenemos:

- *Memoria Flash:* Memoria persistente donde se almacena permanentemente el programa que ejecuta el microcontrolador (hasta una nueva reescritura si se da el caso). En el caso del ATmega328P tiene una capacidad de 32KB.
En los microcontroladores que vienen incluidos en la placa Arduino no podemos usar toda la capacidad de la memoria Flash porque existen 512 bytes ocupados ya por un código pre programado de fábrica (el llamado “bootloader” o “gestor de arranque”), el cual nos permite usar la placa Arduino de una forma sencilla y cómoda sin tener que conocer las interioridades electrónicas más avanzadas del microcontrolador. Los ATmega328P que podamos adquirir individualmente normalmente no incluyen de fábrica este pequeño programa, por lo que sí ofrecen los 32 KB íntegros, pero a cambio no podremos esperar conectarlos a una placa Arduino y que funcionen sin más ya que les faltará tener grabada esa “pre configuración”.
- *Memoria SRAM:* Memoria volátil donde se alojan los datos que en ese instante el programa (grabado separadamente en la memoria Flash) necesita crear o manipular para su correcto funcionamiento. Estos datos suelen tener un contenido variable a lo largo del tiempo de ejecución del programa y cada uno es de un tipo concreto (es decir, un dato puede contener un valor numérico entero, otro un número decimal, otro un valor de tipo carácter, también pueden ser cadenas de texto fijas u otros tipos de datos más especiales). Independientemente del tipo de dato, su valor siempre será eliminado cuando se deje de alimentar eléctricamente al microcontrolador. En el caso del ATmega328P esta memoria tiene una capacidad de 2KB.
- Si necesitáramos ampliar la cantidad de memoria SRAM disponible, siempre podríamos adquirir memorias SRAM independientes y conectarlas al microcontrolador utilizando algún protocolo de comunicación conocido por este como SPI o I2C.
- *Memoria EEPROM:* Memoria persistente donde se almacenan datos que se desea que permanezcan grabados una vez apagado el microcontrolador para poderlos usar posteriormente en siguientes reinicios. En el caso del ATmega328P esta memoria tiene una capacidad de 1 KB, por lo que se puede entender como una tabla de 1024 posiciones de un byte cada una.
Si necesitáramos ampliar la cantidad de memoria EEPROM disponible, siempre podemos adquirir memorias EEPROM independientes y conectarlas al microcontrolador utilizando algún protocolo de comunicación conocido por este como SPI o I2C. O bien, alternativamente, adquirir tarjetas de memoria de tipo SD (“Secure Digital”) y comunicarlas mediante un circuito específico al

microcontrolador. Las memorias SD son en realidad simples memorias Flash, encapsuladas de una forma concreta; son ampliamente utilizadas en cámaras digitales de foto/vídeo y en teléfonos móviles de última generación, ya que ofrecen muchísima capacidad (varios gigabytes) a un precio barato. La razón por la cual este tipo de tarjetas son capaces de ser reconocidas por el ATmega328P es porque pueden funcionar utilizando el protocolo de comunicación SPI.

Podemos ver que la arquitectura a la que pertenece el chip ATmega328P (y en general, toda la familia de microcontroladores AVR) es de tipo Harvard. En este tipo de arquitectura, la memoria que aloja los datos (SRAM o EEPROM) está separada de la memoria que aloja las instrucciones (Flash), por lo que ambas memorias se comunican con la CPU de forma totalmente independiente y en paralelo, consiguiendo así una mayor velocidad y optimización.

El chip ATmega328P tiene registros de 8 bits. De hecho, todos los microcontroladores que incorporan las diferentes placas Arduino son de 8 bits excepto el incorporado en la placa Arduino Due, que es de 32 bits.

El microcontrolador, a través de algunos de sus pines de E/S, utiliza el sistema de comunicación serie para transmitir y recibir órdenes y datos hacia/desde otros componentes electrónicos. Esto es debido sobre todo a que en una comunicación serie solo se necesita en teoría un único canal (un único "cable"), mientras que en una comunicación en paralelo se necesitan varios cables, con el correspondiente incremento de complejidad, tamaño y coste del circuito resultante. No obstante, no podemos hablar de un solo tipo de comunicación serie. Existen muchos protocolos y estándares diferentes basados todos ellos en la transferencia de información en serie, pero implementando de una forma diferente (como el modo de sincronización entre emisor y receptor, la velocidad de transmisión, el tamaño de los paquetes de datos, los mensajes de conexión y desconexión y de dar paso al otro en el intercambio de información, los voltajes utilizados, etc.). De entre el gran número de protocolos de comunicación serie reconocidos por la inmensa variedad de dispositivos electrónicos del mercado, los que nos interesarán conocer son los que el ATmega328P es capaz de comprender y por tanto, los que podrá utilizar para contactar con esa variedad de periféricos. En este sentido, los estándares más importantes son:

- I²C (Inter-Integrated Circuit, también conocido con el nombre de TWI ("Two-wire", literalmente "dos cables" en inglés): Es un sistema muy utilizado en la industria principalmente para comunicar circuitos integrados entre sí. Su

principal característica es que utiliza dos líneas para transmitir la información: una (llamada línea “SDA”) sirve para transferir los datos (los 0s y los 1s) y otra (llamada línea “SCL”) sirve para enviar la señal de reloj. En realidad también se necesitarían dos líneas más: la de alimentación y la de tierra común, pero estas ya se presuponen existentes en el circuito.

Por “señal de reloj” se entiende una señal binaria de una frecuencia periódica muy precisa que sirve para coordinar y sincronizar los elementos integrantes de una comunicación (es decir, los emisores y receptores) de forma que todos sepan cuándo empieza, cuánto dura y cuándo acaba la transferencia de información. En hojas técnicas y diagramas a la señal de reloj en general se le suele describir como CLK (del inglés “clock”).

Cada dispositivo conectado al bus I²C tiene una dirección única que lo identifica respecto al resto de dispositivos, y puede estar configurado como “maestro” o como “esclavo”. Un dispositivo maestro es el que inicia la transmisión de datos y además genera la señal de reloj, pero no es necesario que el maestro sea siempre el mismo dispositivo: esta característica se la pueden ir intercambiando ordenadamente los dispositivos que tengan esa capacidad.

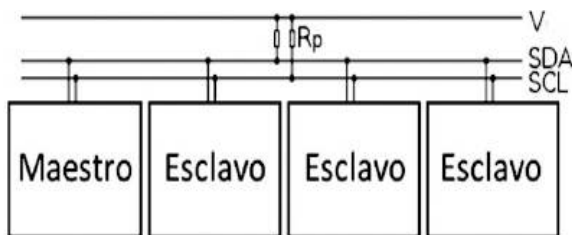


Figura 13: Comunicación I²C.

Tal como se muestra en el diagrama anterior, para funcionar correctamente tanto la línea “SDA” como la “SCL” necesitan estar conectadas mediante una resistencia “pull-up” a la fuente de alimentación común, la cual puede proveer un voltaje generalmente de 5 V o 3,3 V (aunque sistemas con otros voltajes pueden ser posibles).

La velocidad de transferencia de datos es de 100 Kbps en el modo estándar (aunque también se permiten velocidades de hasta 3,4 Mbit/s). No obstante, al haber una única línea de datos, la transmisión de información es “half dúplex” (es decir, la comunicación solo se puede establecer en un sentido al mismo tiempo) por lo que en el momento que un dispositivo empieza a recibir un mensaje, tendrá que esperar a que el emisor deje de transmitir para poder responderle.

- SPI (Serial Peripheral Interface): Al igual que el sistema I²C, el sistema de comunicación SPI es un estándar que permite controlar (a cortas distancias) casi cualquier dispositivo electrónico digital que acepte un flujo de bits serie sincronizado (es decir, regulado por un reloj). Igualmente, un dispositivo conectado al bus SPI puede ser “maestro” o “esclavo”, donde el primero es el que inicia la transmisión de datos y además genera la señal de reloj (aunque, como con I²C, con SPI tampoco es necesario que el maestro sea siempre el mismo dispositivo) y el segundo se limita a responder.

La mayor diferencia entre el protocolo SPI y el I²C es que el primero requiere de cuatro líneas en vez de dos. Una línea (llamada normalmente “SCK”) envía a todos los dispositivos la señal de reloj generada por el maestro actual; otra (llamada normalmente “SS”) es la utilizada por ese maestro para elegir en cada momento con qué dispositivo esclavo se quiere comunicar de entre los varios que puedan estar conectados (ya que solo puede transferir datos con un solo esclavo a la vez); otra (llamada normalmente “MOSI”) es la línea utilizada para enviar los datos 0s y 1s desde el maestro hacia el esclavo elegido; y la otra (llamada normalmente “MISO”) es la utilizada para enviar los datos en sentido contrario: la respuesta de ese esclavo al maestro. Es fácil ver que, al haber dos líneas para los datos la transmisión de información es “full dúplex” (es decir, que la información puede ser transportada en ambos sentidos a la vez).

En la siguiente figura se muestra el esquema de líneas de comunicación existentes entre un maestro y un esclavo y entre un maestro y tres esclavos respectivamente. Se puede observar que, para el caso de la existencia de varios esclavos es necesario utilizar una línea “SS” diferente por cada uno de ellos, ya que esta línea es la que sirve para activar el esclavo concreto que en cada momento el maestro desee utilizar (esto no pasa con las líneas de reloj, “MOSI” y “MISO”, que son compartidas por todos los dispositivos). Técnicamente hablando, el esclavo que reciba por su línea SS un valor de voltaje BAJO será el que esté seleccionado en ese momento por el maestro, y los que reciban el valor ALTO no lo estarán.

Como se puede ver, el protocolo SPI respecto el I²C tiene la desventaja de exigir al microcontrolador dedicar muchos más pines de E/S a la comunicación externa. En cambio, como ventaja podemos destacar que es más rápido y consume menos energía que I²C.

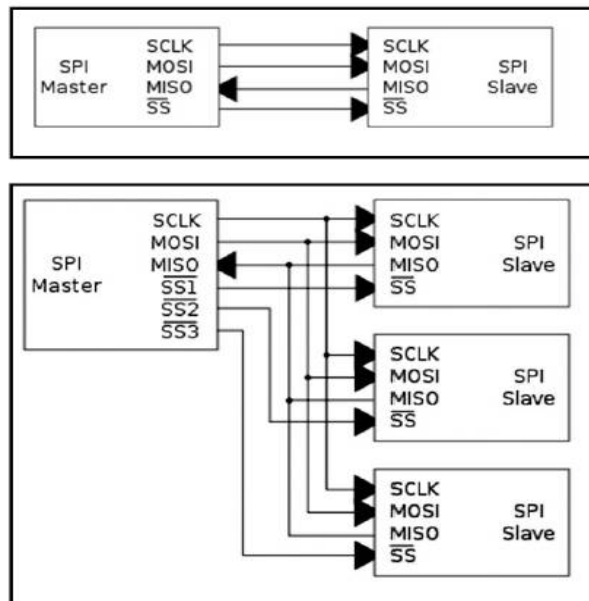


Figura 14: Comunicación SPI.

La placa Arduino UNO, aparte del microcontrolador que incorpora, tiene otras características:

- La alimentación: El voltaje de funcionamiento de la placa Arduino (incluyendo el microcontrolador y el resto de componentes) es de 5 V. Podemos obtener esta alimentación eléctrica de varias maneras:

- Conectando la placa Arduino a una fuente externa, tal como un adaptador AC/DC o una pila: Para el primer caso, la placa dispone de un zócalo donde se puede enchufar una clavija de 2,1 milímetros de tipo “jack”. Para el segundo, los cables salientes de los bornes de la pila se pueden conectar a los pines-hembra marcados como “Vin” y “Gnd” (positivo y negativo respectivamente) en la zona de la placa marcada con la etiqueta “POWER”.

En ambos casos, la placa está preparada en teoría para recibir una alimentación de 6 a 20 voltios, aunque, realmente, el rango recomendado de voltaje de entrada (teniendo en cuenta el deseo de obtener una cierta estabilidad y seguridad eléctrica en nuestros circuitos) es menor: de 7 a 12 voltios. En cualquier caso, este voltaje de entrada ofrecido por la fuente externa siempre es rebajado a los 5 V de trabajo mediante un circuito regulador de tensión que ya viene incorporado dentro de la placa.

- Conectando la placa Arduino a nuestro computador mediante un cable USB: Para ello, la placa dispone de un conector USB hembra de tipo B. La alimentación recibida de esta manera está regulada permanentemente a los 5 V de trabajo y ofrece un máximo de hasta 500 mA de corriente (por lo

tanto, la potencia consumida por la placa es en ese caso de unos 2,5 W). Si en algún momento por el conector USB pasa más intensidad de la deseable, la placa Arduino está protegida mediante un polifusible reseteable que automáticamente rompe la conexión hasta que las condiciones eléctricas vuelven a la normalidad. Una consecuencia de esta protección contra posibles picos de corriente es que la intensidad de corriente recibida a través de USB puede no ser suficiente para proyectos que contengan componentes tales como motores, solenoides o matrices de Led's, los cuales consumen mucha potencia.

Sea cual sea la manera elegida para alimentar la placa, esta es lo suficientemente "inteligente" para seleccionar automáticamente en cada momento la fuente eléctrica disponible y utilizar una u otra sin que tengamos que hacer nada especial al respecto.

Si utilizamos una pila como alimentación externa, una ideal sería la de 9 V (está dentro del rango recomendado de 7 a 12 voltios), y si se utiliza un adaptador AC/DC, se recomienda el uso de uno con las siguientes características:

- El voltaje de salida ofrecido ha de ser de 9 a 12 V DC: En realidad, el circuito regulador que lleva incorporado la placa Arduino es capaz de manejar voltajes de salida (de entrada para la placa) de hasta 20 V, así que en teoría se podrían utilizar adaptadores AC/DC que generen una salida de 20 V DC. No obstante, esta no es una buena idea porque se pierde la mayoría del voltaje en forma de calor (lo cual es terriblemente ineficiente) y además puede provocar el sobrecalentamiento del regulador, y como consecuencia dañar la placa.
- La intensidad de corriente ofrecida ha de ser de 250 mA (o más): Si conectamos a nuestra placa Arduino muchos componentes o unos pocos pero consumidores de mucha energía (como por ejemplo una matriz de LEDs, una tarjeta SD o un motor) el adaptador debería suministrar al menos 500 mA o incluso 1 A. De esta manera nos aseguraremos de que tenemos suficiente corriente para que cada componente pueda funcionar de forma fiable.
- El adaptador ha de ser de polaridad "con el positivo en el centro": Esto quiere decir que la parte externa del cilindro metálico que forma la clavija de 5,5/2,1 mm del adaptador ha de ser el borne negativo y el hueco interior del cilindro ha de ser el borne positivo. Lo más sencillo para estar seguros

de que nuestro adaptador es el adecuado es observar si tiene impreso en algún sitio el siguiente símbolo:

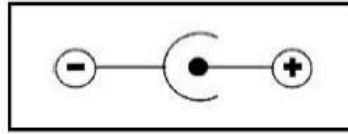


Figura 15: Adaptador de alimentación.

- Por otro lado, dentro de la zona etiquetada como “POWER” en la placa Arduino existe una serie de pines-hembra relacionados con la alimentación eléctrica, como son:
 - GND: Pines-hembra conectados a tierra. Es muy importante que todos los componentes de nuestros circuitos compartan una tierra común como referencia. Estos pines-hembra se ofrecen para realizar esta función.
 - Vin: Este pin-hembra se puede utilizar para dos cosas diferentes: si la placa está conectada mediante la clavija de 2,1mm a alguna fuente externa que aporte un voltaje dentro de los márgenes de seguridad, podemos conectar a este pin-hembra cualquier componente electrónico para alimentarlo directamente con el nivel de voltaje que esté aportando la fuente en ese momento. Si la placa está alimentada mediante USB, entonces ese pin-hembra aportará 5 V regulados. En cualquier caso, la intensidad de corriente máxima aportada es de 40 mA (esto hay que tenerlo en cuenta cuando conectemos dispositivos que consuman mucha corriente, como por ejemplo motores). También podemos usar el pin-hembra “Vin” para otra cosa: para alimentar la propia placa directamente desde alguna fuente de alimentación externa sin utilizar ni la clavija ni el cable USB. Esto se hace conectándole el borne positivo de la fuente (por ejemplo, una pila de 9 V) y conectando el borne negativo al pin de tierra. Si se usa este montaje, el regulador de tensión que incorpora la placa reducirá el voltaje recibido de la pila al voltaje de trabajo de la placa (los 5 V).
 - 5 V: Este pin-hembra se puede utilizar para dos cosas diferentes: tanto si la placa está alimentada mediante el cable USB como si está alimentada por una fuente externa que aporte un voltaje dentro de los márgenes de seguridad, podemos conectar a este pin-hembra cualquier componente para que pueda recibir 5 V regulados. En cualquier caso, la intensidad de corriente máxima generada será de 40 mA. Pero también podemos usar este pin hembra para otra cosa: para alimentar la propia placa desde una

fuentes de alimentación externa previamente regulada a 5 V sin utilizar el cable USB ni la clavija de 2,1mm.

- 3,3 V: Este pin-hembra ofrece un voltaje de 3,3 voltios. Este voltaje se obtiene a partir del recibido indistintamente a través del cable USB o de la clavija de 2,1 mm, y está regulado (con un margen de error del 1%) por un circuito específico incorporado en la placa: el LP2985. En este caso particular, la corriente máxima generada es de 50 mA. Al igual que con los pines anteriores, podremos usar este pin para alimentar componentes de nuestros circuitos que requieran dicho voltaje (los más delicados), pero en cambio, no podremos conectar ninguna fuente externa aquí porque el voltaje es demasiado limitado para poder alimentar a la placa.

2.5. Métodos para identificar la función de transferencia [1]

La función de transferencia de un sistema descrito mediante una ecuación diferencial lineal e invariante en el tiempo se define como el cociente entre la transformada de Laplace de la salida (función de respuesta) y la transformada de Laplace de la entrada (función de excitación) bajo la suposición de que todas las condiciones iniciales son cero.

Considérese el sistema lineal e invariante en el tiempo descrito mediante la siguiente ecuación diferencial:

$$\begin{aligned} a_0 y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} \dot{y} + a_n y \\ = b_0 x^{(m)} + b_1 x^{(m-1)} + \dots + b_{m-1} \dot{x} + b_m x \quad (n \geq m) \end{aligned}$$

Donde “y” es la salida del sistema y “x” es la entrada. La función de transferencia de este sistema es el cociente de la transformada de Laplace de la salida y la transformada de Laplace de la entrada cuando todas las condiciones iniciales son cero, o

$$\begin{aligned} \text{Función de transferencia} = G(s) &= \frac{\mathcal{L}[\text{salida}]}{\mathcal{L}[\text{entrada}] \Big|_{\text{condiciones iniciales cero}}} \\ &= \frac{Y(s)}{X(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} \end{aligned}$$

A partir del concepto de función de transferencia, es posible representar la dinámica de un sistema mediante ecuaciones algebraicas en s. Si la potencia más alta de s en el denominador de la función de transferencia es igual a n, el sistema se denomina sistema de orden n-ésimo.

La aplicación del concepto de función de transferencia está limitada a los sistemas descritos mediante ecuaciones diferenciales lineales invariantes en el tiempo; sin embargo, el enfoque de la función de transferencia se usa extensamente en el análisis y diseño de dichos sistemas. A continuación se presentan algunos comentarios importantes relacionados con la función de transferencia.

- La función de transferencia de un sistema es un modelo matemático porque es un método operacional para expresar la ecuación diferencial que relaciona la variable de salida con la variable de entrada.
- La función de transferencia es una propiedad de un sistema, independiente de la magnitud y naturaleza de la entrada o función de excitación.
- La función de transferencia incluye las unidades necesarias para relacionar la entrada con la salida; sin embargo, no proporciona información acerca de la estructura física del sistema. (Las funciones de transferencia de muchos sistemas físicamente diferentes pueden ser idénticas.)
- Si se conoce la función de transferencia de un sistema, se estudia la salida o respuesta para varias formas de entrada, con la intención de comprender la naturaleza del sistema.
- Si se desconoce la función de transferencia de un sistema, puede establecerse experimentalmente introduciendo entradas conocidas y estudiando la salida del sistema. Una vez establecida una función de transferencia, proporciona una descripción completa de las características dinámicas del sistema, a diferencia de su descripción física.

2.5.1. Métodos de identificación experimental [6]

Mediante el proceso de identificación se pretende obtener un modelo matemático del proceso que se comporte de una manera lo más aproximada posible al sistema real.

Sin embargo si se conocen experimentalmente en un momento dado, la excitación aplicada al sistema y el curso temporal de la respuesta a esa excitación, es posible asociar un modelo al sistema y es posible también determinar la relación entre variables y parámetros del sistema que correspondan al modelo que se le ha asociado a dicho sistema. Al proceso de estructurar un modelo para un sistema y de determinar cuantitativamente a partir de datos experimentales, la interrelación entre las variables y parámetros del proceso, recibe el nombre de "Identificación Paramétrica del Sistema".

La forma más usual de identificar a un sistema es aplicar en su entrada una excitación conocida, obtener la respuesta y después tratar de ajustar un modelo conocido a las particularidades del sistema sujeto a la identificación. Las señales más frecuentemente usadas como excitaciones al hacer una identificación son:

- Señal escalón
- Señal senoidal de amplitud constante y frecuencia variable

A la identificación hecha por medio de una función escalón se le conoce como “Identificación en el dominio del tiempo”, y a la identificación hecha por medio de señales senoidales se le conoce como “Identificación en el dominio de la frecuencia”. Aunque en ambos casos el objetivo sea determinar la función de transferencia que mejor describa el comportamiento del sistema cabe destacar que en la actualidad existen algoritmos computacionales que pueden obtener la función de transferencia de un sistema real con bastante precisión.

2.5.1.1. Identificación mediante la respuesta a una entrada escalón

Para realizar una identificación en el dominio del tiempo es necesario excitar al sistema por medio de una función escalón (de preferencia unitaria) y conocer la respuesta a esta excitación. La mejor manera de conocer la respuesta es contar con una gráfica $[c(t) \text{ vs } t]$ adecuadamente escalada, en la que sea posible observar el curso temporal de la respuesta desde el momento de aplicar la excitación hasta el establecimiento del régimen permanente en el sistema.

Una de las características atractivas del método de identificación en el tiempo es el hecho de que una señal escalón siempre es fácil de generar. Básicamente todos los procesos existentes en la naturaleza pueden clasificarse en dos tipos, sistemas de primer orden y sistemas de segundo orden (muchos de los sistemas de orden superior pueden ser aproximados por sistemas de segundo orden), dentro de los cuales existen muchas variantes.

2.5.1.1.1. Sistemas con respuesta sobreamortiguada

Se tienen los siguientes casos de estudio:

- *Sistemas de primer orden puro*: La respuesta típica de estos sistemas no presenta sobreoscilación, esto quiere decir que nunca llegan al valor exacto de la consigna y por lo tanto,

son sistemas relativamente lentos. Esto se puede apreciar en la figura 16.

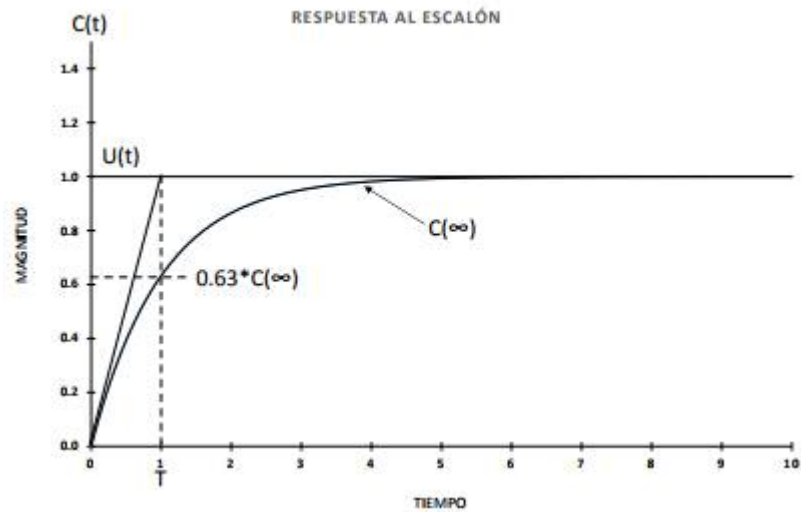


Figura 16: Respuesta a la entrada escalón.

La función de transferencia de un sistema de 1º orden es la siguiente:

$$G(s) = \frac{K}{Ts + 1}$$

Donde K es la ganancia del sistema y T es la constante de tiempo.

El valor de la constante de tiempo se obtiene sobre la gráfica, para ello se observa el tiempo correspondiente a un valor del 63%. Normalmente se trabaja con un factor denominado tiempo de establecimiento, que suele estar comprendido entre un 95 – 98 %. Este factor determina el tiempo en el cual la respuesta se estabiliza entre los límites indicados a ese porcentaje.

- *Sistemas de primer orden con retardo:* La respuesta típica de este tipo de sistemas presenta la misma configuración que un sistema de primer orden puro, en el cual la respuesta presenta un desfase o retardo respecto a la señal de entrada. Lo anterior se puede apreciar en la figura 17.

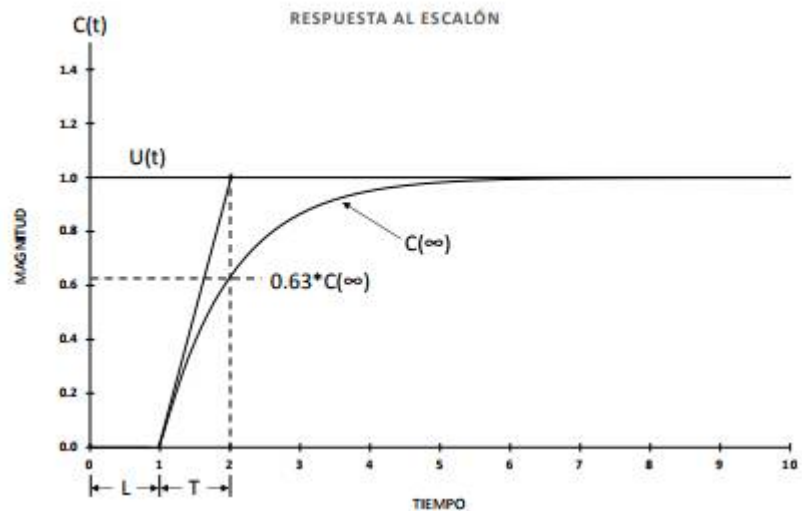


Figura 17: Respuesta a la entrada escalón.

La función de transferencia de un sistema de primer orden con retardo es la siguiente:

$$G(s) = \frac{K}{Ts + 1} e^{-Ls}$$

Donde K es la ganancia del sistema, T es la constante de tiempo y L es el retardo.

Igual que para el caso anterior, el valor de la constante de tiempo se obtiene sobre la gráfica, para ello se observa el tiempo correspondiente a un valor del 63% de $u(t)$.

Normalmente se trabaja con un factor denominado tiempo de establecimiento, que suele estar comprendido entre un 95 – 98 %. Este factor determina el tiempo en el cual la respuesta se estabiliza entre los límites indicados a ese porcentaje.

- *Sistemas de primer orden con polos reales múltiples:* La respuesta de este tipo de sistemas varía según la cantidad de polos existentes, conforme aumenta el número de polos la respuesta es más lenta, teniendo al inicio un arranque con mayor suavidad. Esto se puede apreciar en la figura 18.

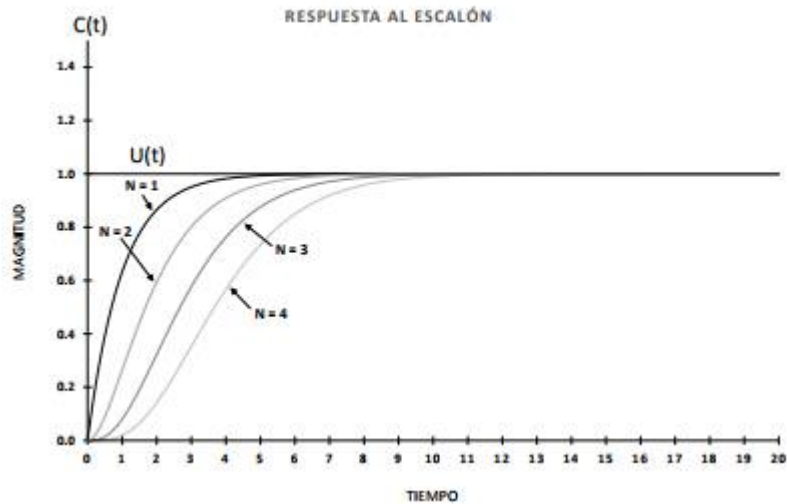


Figura 18: Respuesta a la entrada escalón.

La función de transferencia de un sistema de polos reales múltiples es la siguiente:

$$G(s) = \frac{K}{(Ts + 1)^n}$$

Donde K es la ganancia del sistema, T es la constante de tiempo y “n” es el número de polos del sistema.

Para obtener la función de transferencia de este tipo de sistemas se puede utilizar el “Método de Strejc”. Este método se emplea para la identificación de sistemas de polos múltiples, mediante los parámetros Tu y Ta obtenidos sobre la respuesta del sistema.

Emplea una línea recta de pendiente máxima superpuesta sobre la zona de pendiente, de modo que el valor del parámetro Tu se obtiene con el corte del eje de abscisas y el valor del parámetro Ta se obtiene con el corte de una paralela al eje de abscisas, en el punto donde la respuesta está estable. Lo anterior se muestra en la figura 19.

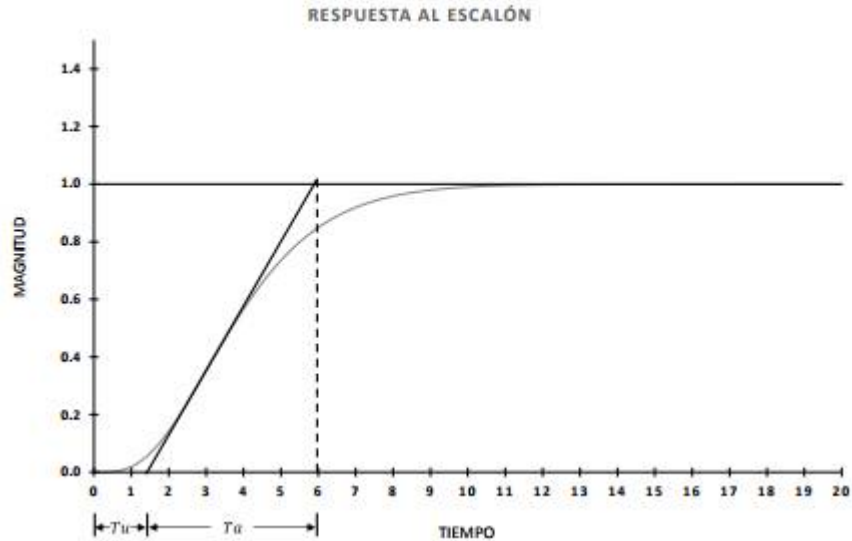


Figura 19: “Método de Strejc”.

Tras obtener el valor de las variables T_u y T_a , se obtiene el valor de T_u/T_a . Con este valor se va a la tabla de Strejc y se toma el valor más próximo, que determina el número de polos múltiples “n”.

n	T_a/τ	T_u/τ	T_u/T_a
1	1	0	0
2	2.7	0.28	0.104
3	3.7	0.8	0.22
4	4.46	1.42	0.32
5	5.12	2.1	0.41

Figura 20: “Tabla de Strejc”.

Se toman los parámetros $\frac{T_a}{\tau}$ y $\frac{T_u}{\tau}$, y se despeja “ τ ” en cada ecuación, si los dos valores no coinciden significa que el sistema no se ajusta bien a un sistema de polos múltiples. Si por el contrario el valor obtenido es muy próximo al de la tabla se tendrá un sistema de orden “n”.

- *Sistemas de primer orden con polos reales distintos:* Si la respuesta es sobreamortiguada, y no se ajusta a ningún sistema visto hasta ahora y < 0.104 ($n = 2$), se puede

ajustar con polos reales distintos. La respuesta típica de una función de transferencia de polos reales distintos se puede apreciar en la figura 21.

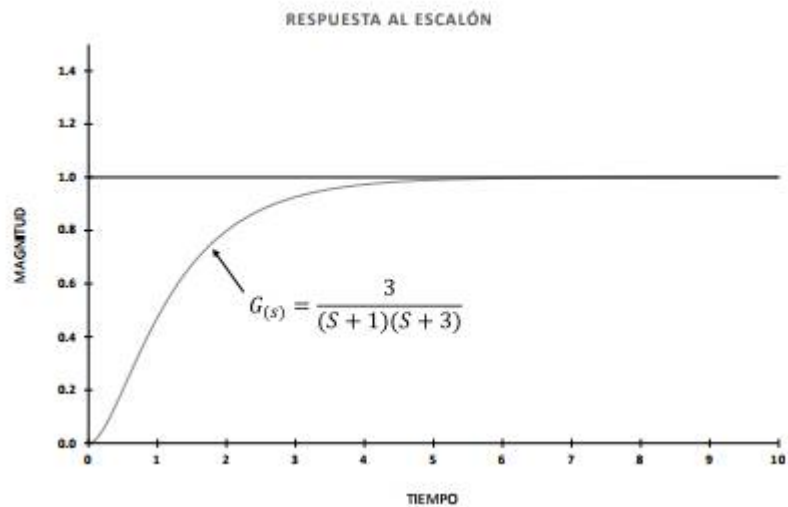


Figura 21: Respuesta al escalón.

La función de transferencia de un sistema con dos polos reales diferentes es la siguiente:

$$G(s) = \frac{K}{(T_1s + 1)(T_2s + 1)}$$

Donde K es la ganancia del sistema, T1 y T2 son las constantes de tiempo.

Un método simple para obtener las constantes de tiempo del sistema de polos distintos es primero aplicar el método de Strejc y luego hacer ajustes en las constantes de tiempo a prueba y error, hasta obtener la respuesta lo más precisa posible.

2.5.1.1.2. Sistemas con respuesta subamortiguada

Este tipo de respuestas presentan sobre oscilación y un periodo transitorio con oscilación, y se deben a sistemas con polos complejos conjugados.

Se tienen los siguientes casos de estudio:

- *Sistemas estándar de segundo orden:* La mayoría de los sistemas industriales se comportan como un sistema de este tipo, en el cual posteriormente el control pretende limitar

parámetros como la sobre oscilación, el tiempo de establecimiento y el error en régimen permanente. La respuesta típica de una función de transferencia de polos complejos se puede apreciar en la figura 22.

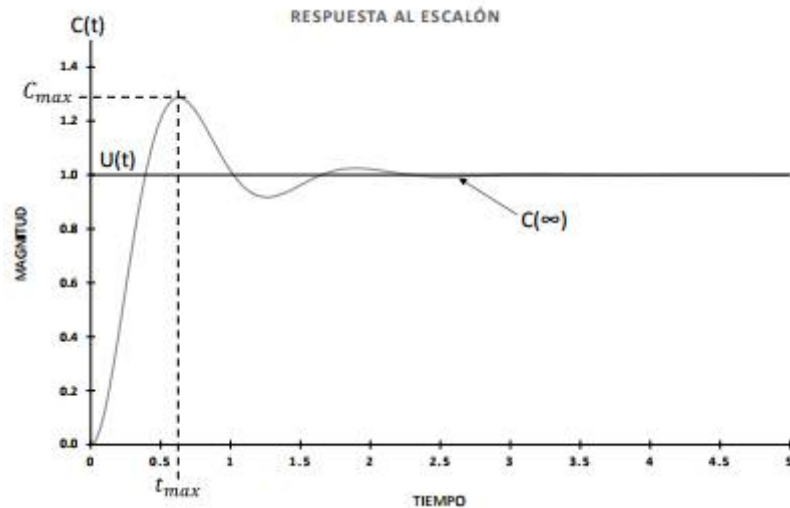


Figura 22: Respuesta al escalón.

La función de transferencia de un sistema estándar de segundo orden está dada por la siguiente expresión:

$$G(s) = \frac{K \omega_n^2}{s^2 + 2\xi \omega_n s + \omega_n^2}$$

Donde K es la ganancia del sistema, ζ es el factor de amortiguamiento y ω_n es la frecuencia natural.

Los parámetros que describen a este tipo de respuesta son los siguientes:

$$S.P = \frac{C_{max} - C(\infty)}{C(\infty)} = e^{-\frac{\xi}{\sqrt{1-\xi^2}}\pi}$$

Dónde:

$$\xi = \frac{|l_n(S.P)|}{\sqrt{\pi^2 + [l_n(S.P)]^2}}$$

El tiempo al cual ocurre el sobretiro máximo está dado por:

$$t_{max} = \frac{\pi}{\omega_n \sqrt{1 - \xi^2}}$$

Dónde:

$$\omega_n = \frac{\pi}{t_{max}\sqrt{1-\xi^2}}$$

- *Sistemas estándar de segundo orden con retardo:* La respuesta típica de una función de transferencia de polos complejos con un retardo se puede apreciar en la figura 23.

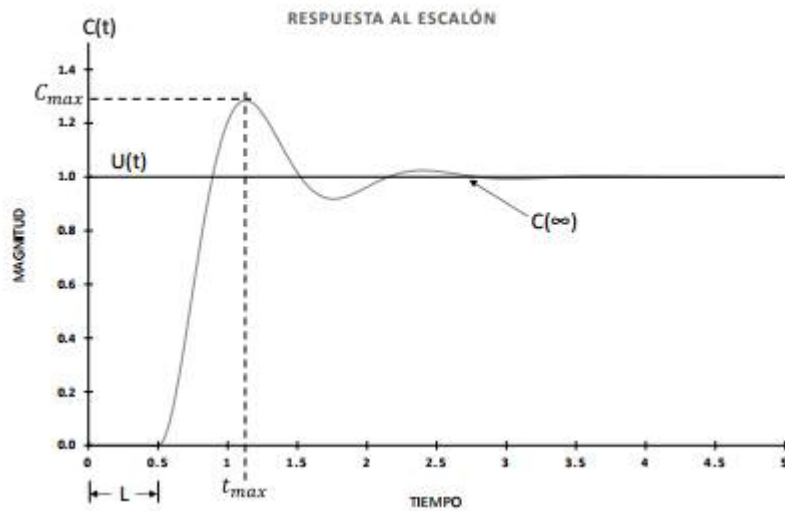


Figura 23: Respuesta al escalón.

La función de transferencia de un sistema estándar de segundo orden está dada por la siguiente expresión:

$$G(s) = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} e^{-Ls}$$

Donde \$K\$ es la ganancia del sistema, \$\xi\$ es el factor de amortiguamiento y \$\omega_n\$ es la frecuencia natural.

Los parámetros que describen a este tipo de respuesta se determinan de la misma forma que para el caso del sistema estándar de segundo orden incluyendo el tiempo de retraso \$L\$, que puede ser determinado de la figura 23.

2.5.1.2. Identificación mediante la respuesta en frecuencia [6]

Dado un sistema con función de transferencia $G(s)$, se define la respuesta en frecuencia como la función de ω : $G(j\omega) = G(s)|_{s=j\omega}$

Si $u(t) = U \cdot \text{sen}(\omega t)$ es una señal senoidal pura de frecuencia ω , entonces $y(t)$ cuando pasa el transitorio es una señal senoidal de la misma frecuencia ω : $y(t) = U \cdot |G(j\omega)| \text{sen}(\omega t + \arg(G(j\omega)))$

La representación de la respuesta en frecuencia mediante los diagramas de Bode contempla dos gráficas. En la primera se representa en el eje de abscisas la frecuencia en escala logarítmica la frecuencia y en el eje de ordenadas la amplitud en decibelios. En la segunda se representa en el eje de abscisas la frecuencia en escala logarítmica la frecuencia y en el eje de ordenadas el argumento o desfase en grados.

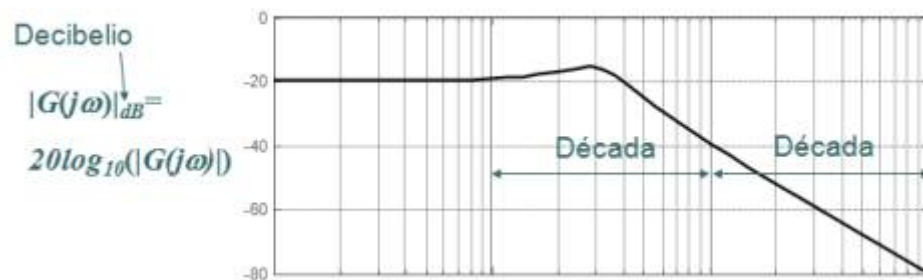


Figura 24: Gráfica de respuesta en frecuencia – Magnitud.

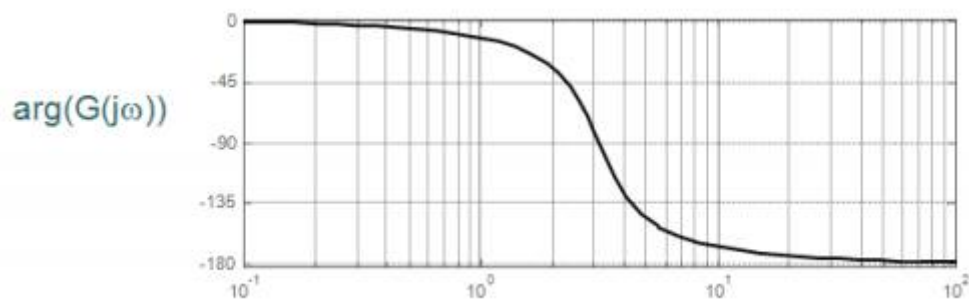


Figura 25: Gráfica de respuesta en frecuencia – Fase.

El método empleado para la identificación de sistemas mediante la respuesta en frecuencia es la aproximación asintótica de líneas rectas sobre la gráfica de amplitud / frecuencia. Las consideraciones a tener en cuenta para la aplicación de este método son:

- *Para el diagrama de amplitud:*
 - Término constante k: añade un valor constante $20\log(k)$.
 - Un polo simple añade pendiente de -20 db/década a partir del valor del polo.
 - Un cero simple añade una pendiente de $+20$ db/década a partir del valor del cero.
 - Un par de polos complejos, añade una pendiente de -40 db/década a partir del valor del módulo de los polos. Si x es pequeño, hay pico de resonancia.
 - Un par de ceros complejos, añade una pendiente de $+40$ db/década a partir del valor del módulo de los ceros. Si x es pequeño, hay pico de resonancia.
 - Un polo en el origen ($1/s$) añade una pendiente de -20 db/década en todas las frecuencias.
 - Un cero en el origen (s) añade con una pendiente de $+20$ db/década en todas las frecuencias.
- *Para el diagrama de fase:*
 - Término constante k: no contribuye.
 - Un polo simple: 0° para $\omega=0$, -90° para $\omega=\infty$.
 - Un cero simple: 0° para $\omega=0$, $+90^\circ$ para $\omega=\infty$.
 - Un par de polos complejos: 0° para $\omega=0$, -180° para $\omega=\infty$.
 - Un par de ceros complejos: 0° para $\omega=0$, $+180^\circ$ para $\omega=\infty$.
 - Un polo en el origen ($1/s$): -90° en todas las frecuencias.
 - Un cero en el origen (s): $+90^\circ$ en todas las frecuencias.
 - Un cero simple positivo: 0° para $\omega=0$, -90° para $\omega=\infty$.
 - Un retardo puro (e^{-sT}): fase = $-\omega T$.

Si se emplea la respuesta en frecuencia para la identificación de un sistema hay que tener presente la zona de identificación posible, esto quiere decir, que no toda la gráfica obtenida es válida para la identificación. A continuación se muestra una gráfica en la cual al final se obtiene una oscilación inesperada, que indica que el sistema presenta retardo.

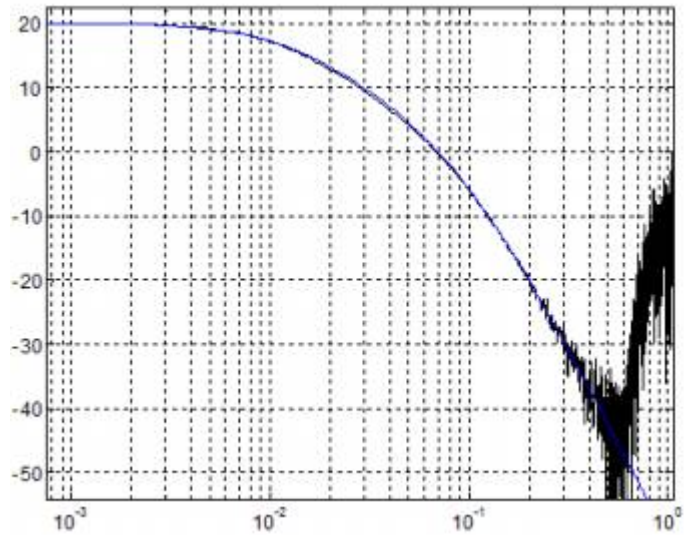


Figura 26: Grafica de un sistema con retardo.

Aplicando retardo, se obtiene una gráfica en la cual se ajusta casi perfectamente la fase esperada.

$$G(s) = \frac{10}{(1 + s/0.011)(1 + s/0.094)^2} e^{-5.15s}$$

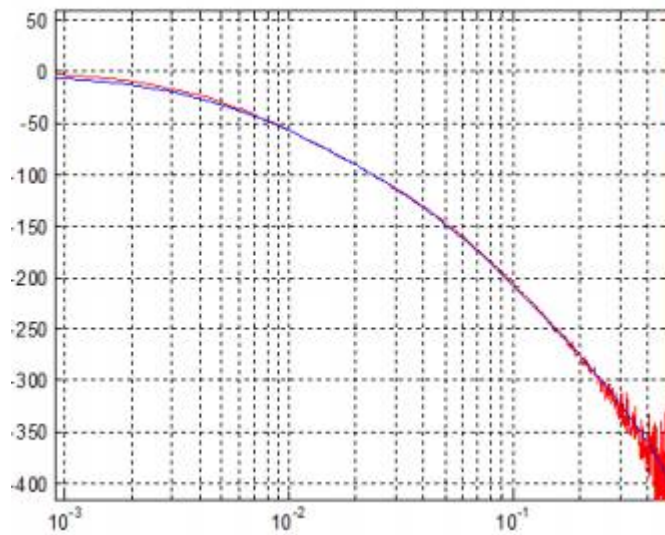


Figura 27: Respuesta en frecuencia de un sistema con retardo.

2.5.1.3. Identificación mediante mínimos cuadrados [6]

Este método de identificación consiste en obtener el equivalente discreto de un sistema continuo lineal. Para ello se debe especificar el periodo de muestreo y el intervalo de valores continuos que equivalen a un mismo valor discreto.

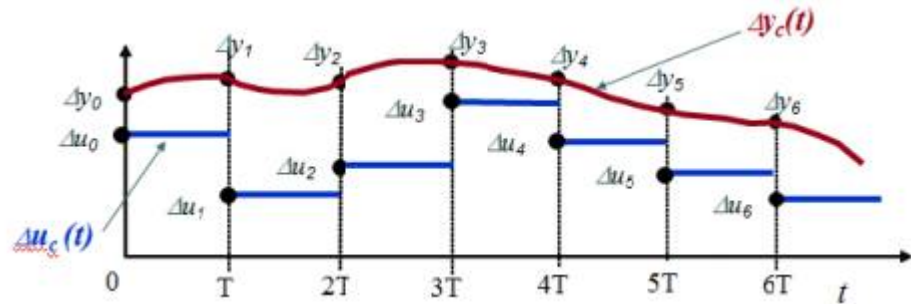


Figura 28: Identificación mediante mínimos cuadrados.

Aplicando la transformada Z se obtiene la función discreta del sistema:

$$G(z) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

$$\Delta y_k + a_1 \Delta y_{k-1} + a_2 \Delta y_{k-2} = b_1 \Delta u_{k-1} + b_2 \Delta u_{k-2}$$

El experimento se lleva a cabo con los siguientes pasos:

- Se aplica una entrada constante hasta que el sistema llega al equilibrio.

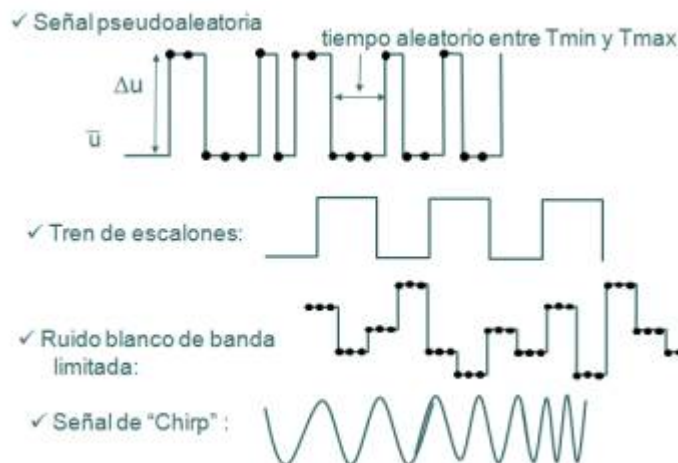


Figura 29: Señales de prueba.

- Se produce el cambio de la entrada para obtener variación de la salida.

- Restando el valor del punto de equilibrio se obtiene la secuencia de entrada y salida, es decir, $\Delta u_0, \Delta u_1, \dots, \Delta u_N$ y $\Delta y_0, \Delta y_1, \dots, \Delta y_N$.
- Se plantea el sistema de ecuaciones a resolver (no tiene solución, porque tiene más ecuaciones que incógnitas).
- Se definen los errores de las ecuaciones como:

$$\begin{cases} e_2 = \Delta y_2 - (-a_1 \Delta y_1 - a_2 \Delta y_0 + b_1 \Delta u_1 + b_2 \Delta u_0) \\ e_3 = \Delta y_3 - (-a_1 \Delta y_2 - a_2 \Delta y_1 + b_1 \Delta u_2 + b_2 \Delta u_1) \\ \vdots \\ e_N = \Delta y_N - (-a_1 \Delta y_{N-1} - a_2 \Delta y_{N-2} + b_1 \Delta u_{N-1} + b_2 \Delta u_{N-2}) \end{cases}$$

- Se reorganiza de la siguiente forma matricial:

$$Y = \begin{bmatrix} \Delta y_2 \\ \Delta y_3 \\ \vdots \\ \Delta y_N \end{bmatrix} \quad X = \begin{bmatrix} -\Delta y_1 & -\Delta y_0 & \Delta u_1 & \Delta u_0 \\ -\Delta y_2 & -\Delta y_1 & \Delta u_2 & \Delta u_1 \\ \vdots & \vdots & \vdots & \vdots \\ -\Delta y_{N-1} & -\Delta y_{N-2} & \Delta u_{N-1} & \Delta u_{N-2} \end{bmatrix} \quad \theta = \begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{bmatrix} \quad E = \begin{bmatrix} e_2 \\ e_3 \\ \vdots \\ e_N \end{bmatrix}$$

$$Y = X \cdot \theta \quad \Rightarrow \quad E = Y - X \cdot \theta$$

- Finalmente se resuelve el sistema de forma que se minimiza la suma de errores al cuadrado, es decir, minimiza: $E^T E = \sum e_i^2$ y la solución se presenta como: $\hat{\theta} = (X^T X)^{-1} X^T Y$

2.5.2. Determinación de la función de transferencia utilizando software [7]

La herramienta "Ident" de Matlab es una aplicación para la construcción de modelos matemáticos de sistemas dinámicos a partir de datos de entrada-salida medidos. Permite crear y usar modelos de sistemas dinámicos que no se modelan fácilmente a partir de los primeros principios o especificaciones. Se pueden usar datos de entrada y salida en el dominio de tiempo y en el dominio de la frecuencia para identificar funciones de transferencia de tiempo discreto y tiempo continuo, modelos de proceso y modelos de espacio de estado. La caja de herramientas también proporciona algoritmos para la estimación de parámetros integrados en línea.

Cuando se preparan datos para identificar modelos, se debe especificar información tal como los nombres de los canales de entrada y salida, el tiempo de muestreo y el comportamiento entre muestras. Ident permite adjuntar esta información a los datos, lo que facilita la visualización de datos, la conversión del dominio y varias tareas de pre procesamiento.

Los datos medidos a menudo tienen desplazamientos, derivas lentas, valores atípicos, valores perdidos y otras anomalías. Ident elimina tales anomalías mediante la realización de operaciones de desviación, filtrado, re muestreo y reconstrucción de datos faltantes. Además, puede analizar la idoneidad de los datos para la identificación y proporcionar diagnósticos sobre la existencia de circuitos de retroalimentación y la presencia de no linealidades.

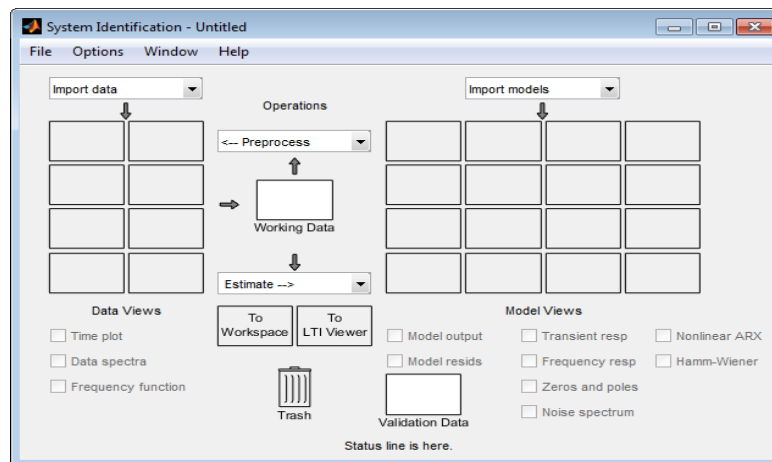


Figura 30: Importación de datos en Ident.

Ident también estima la respuesta al escalón y la respuesta en frecuencia del sistema a partir de los datos medidos. Con estas respuestas, puede analizar las características del sistema, como las constantes de tiempo dominantes, los retardos de entrada y las frecuencias de resonancia. También puede usar estas características para configurar los modelos paramétricos durante la estimación.

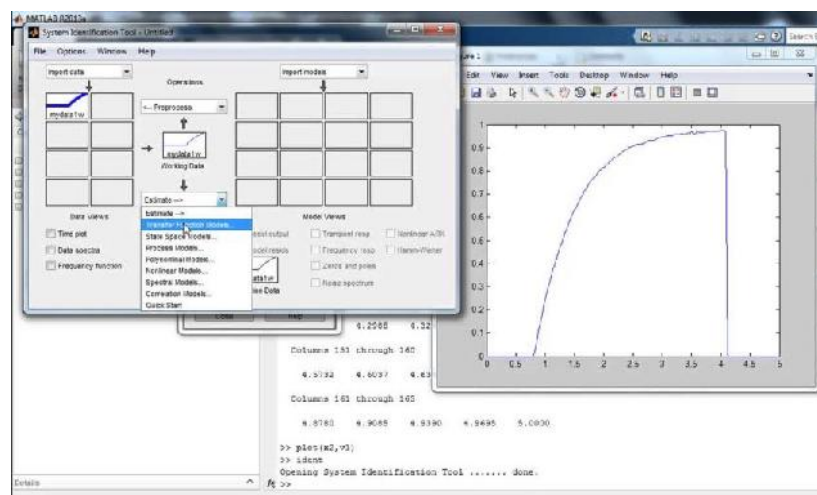


Figura 31: Respuesta a la entrada escalón en Ident.

Los modelos paramétricos, como las funciones de transferencia o los modelos de espacio de estado, usan una pequeña cantidad de parámetros para capturar la

dinámica del sistema. Ident estima los parámetros del modelo y sus incertidumbres a partir de los datos de tiempo-respuesta y frecuencia-respuesta. Se puede analizar estos modelos usando gráficos de tiempo-respuesta y frecuencia-respuesta, como escalón, impulso, diagramas de Bode y mapas de polos y ceros.

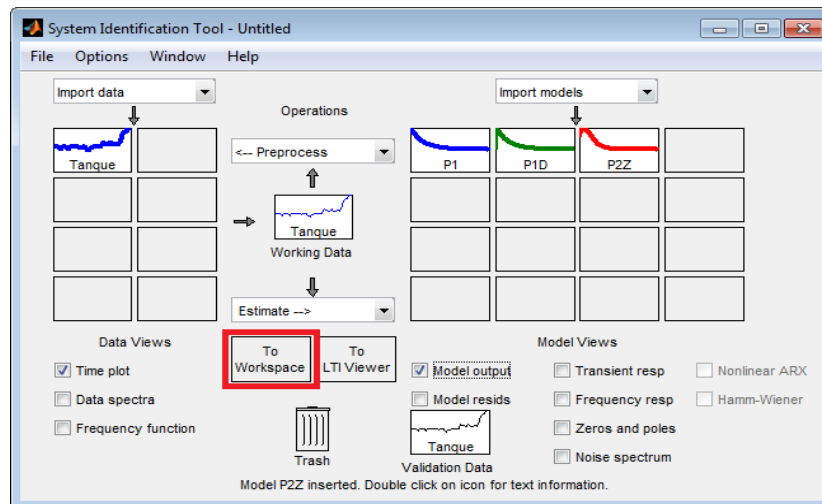


Figura 32: Función de transferencia en Ident.

2.6. Métodos de sintonización de controladores PID [1]

Si se puede obtener un modelo matemático de la planta, es posible aplicar diversas técnicas de diseño con el fin de determinar los parámetros del controlador que cumpla las especificaciones del transitorio y del estado estacionario del sistema en lazo cerrado. Sin embargo, si la planta es tan complicada que no es fácil obtener su modelo matemático, tampoco es posible un método analítico para el diseño de un controlador PID. En este caso, se debe recurrir a procedimientos experimentales para la sintonía de los controladores PID.

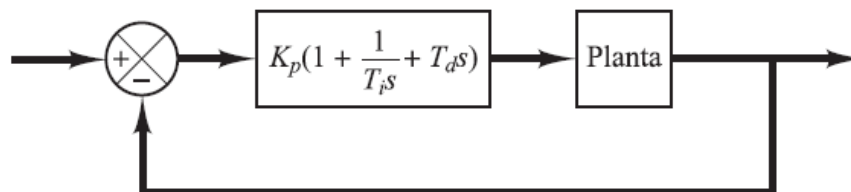


Figura 33: Control PID de una planta

El proceso de seleccionar los parámetros del controlador que cumplan con las especificaciones de comportamiento dadas se conoce como sintonía del controlador.

Ziegler y Nichols sugirieron reglas para sintonizar los controladores PID (esto significa dar valores a K_p , T_i y T_d) basándose en las respuestas escalón experimentales o en el valor de K_p que produce estabilidad marginal cuando sólo se usa la acción de control proporcional. Las reglas de Ziegler-Nichols, que se presentan a continuación, son muy convenientes cuando no se conocen los modelos matemáticos de las plantas. (Por supuesto, estas reglas se pueden aplicar al diseño de sistemas con modelos matemáticos conocidos.) Tales reglas sugieren un conjunto de valores de K_p , T_i y T_d que darán una operación estable del sistema. No obstante, el sistema resultante puede presentar una gran sobre elongación en su respuesta escalón de forma que resulte no aceptable. En tales casos se necesitará una serie de ajustes finos hasta que se obtenga el resultado deseado. De hecho, las reglas de sintonía de Ziegler-Nichols dan una estimación razonable de los parámetros del controlador y proporcionan un punto de partida para una sintonía fina, en lugar de dar los parámetros K_p , T_i y T_d en un único intento.

2.6.1. Reglas de Ziegler y Nichols

Ziegler y Nichols propusieron reglas para determinar los valores de la ganancia proporcional K_p , del tiempo integral T_i y del tiempo derivativo T_d , basándose en las características de respuesta transitoria de una planta dada. Tal determinación de los parámetros de los controladores PID o sintonía de controladores PID la pueden realizar los ingenieros mediante experimentos sobre la planta. (Después de la propuesta inicial de Ziegler-Nichols han aparecido numerosas reglas de sintonía de controladores PID. Estas reglas están disponibles tanto en publicaciones técnicas como de los fabricantes de estos controladores.

Hay dos métodos denominados reglas de sintonía de Ziegler-Nichols: el primero y el segundo método. A continuación se hace una breve presentación de estos dos métodos.

2.6.1.1. Primer método

En el primer método, la respuesta de la planta a una entrada escalón unitario se obtiene de manera experimental, tal como se muestra en la figura 34.

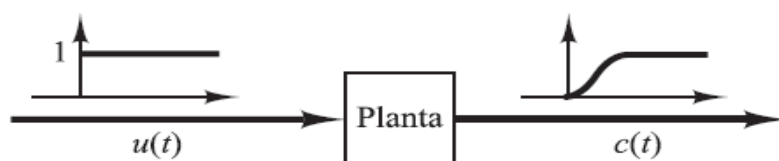


Figura 34: Respuesta a un escalón unitario de una planta.

Si la planta no contiene integradores ni polos dominantes complejos conjugados, la curva de respuesta escalón unitario puede tener forma de S, como se observa en la figura 35.

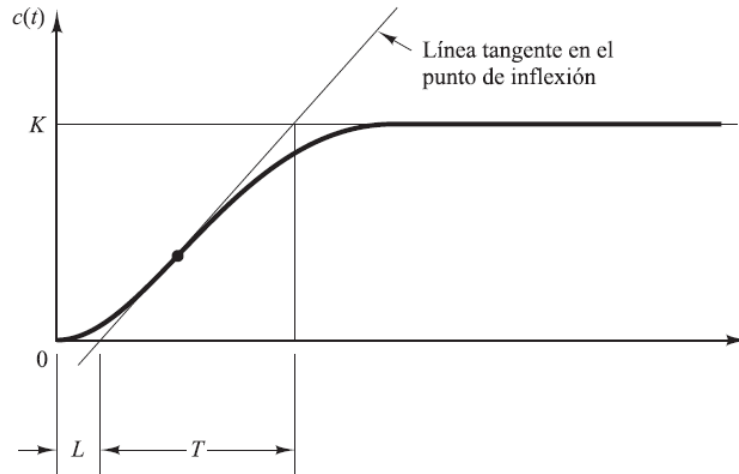


Figura 35: Curva de respuesta en forma de S.

Este método se puede aplicar si la respuesta muestra una curva con forma de S. Tales curvas de respuesta escalón se pueden generar experimentalmente o a partir de una simulación dinámica de la planta.

La curva con forma de S se caracteriza por dos parámetros: el tiempo de retardo L y la constante de tiempo T. El tiempo de retardo y la constante de tiempo se determinan dibujando una recta tangente en el punto de inflexión de la curva con forma de S y determinando las intersecciones de esta tangente con el eje del tiempo y con la línea $c(t) = K$, tal como se muestra en la Figura 18. En este caso, la función de transferencia $C(s)/U(s)$ se aproxima mediante un sistema de primer orden con un retardo del modo siguiente:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts + 1}$$

Ziegler y Nichols sugirieron establecer los valores de K_p , T_i y T_d de acuerdo con la fórmula que se muestra en la Tabla 1.

Tabla 1: Respuesta escalón de la planta.

Tipo de controlador	K_p	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	$2L$	$0.5L$

El controlador PID sintonizado mediante el primer método de las reglas de Ziegler-Nichols produce:

$$\begin{aligned}
 G_c(s) &= K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \\
 &= 1.2 \frac{T}{L} \left(1 + \frac{1}{2Ls} + 0.5Ls \right) \\
 &= 0.6T \frac{\left(s + \frac{1}{L} \right)^2}{s}
 \end{aligned}$$

Por tanto, el controlador PID tiene un polo en el origen y un cero doble en $s = 1/L$.

2.6.1.2. Segundo método

En el segundo método, primero se fija $T_i = \infty$ y $T_d = 0$. Usando sólo la acción de control proporcional (véase la Figura 19), se incrementa K_p desde 0 hasta un valor crítico K_{cr} , en donde la salida presente oscilaciones sostenidas. (Si la salida no presenta oscilaciones sostenidas para cualquier valor que pueda tomar K_p , entonces este método no se puede aplicar.) Así, la ganancia crítica K_{cr} y el periodo P_{cr} correspondiente se determinan experimentalmente.

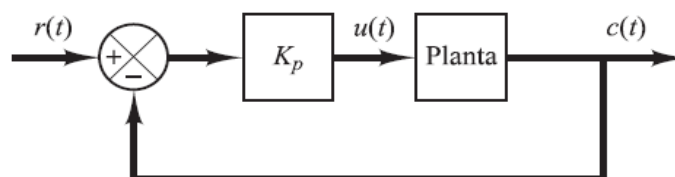


Figura 36: Sistema en lazo cerrado con un controlador proporcional.

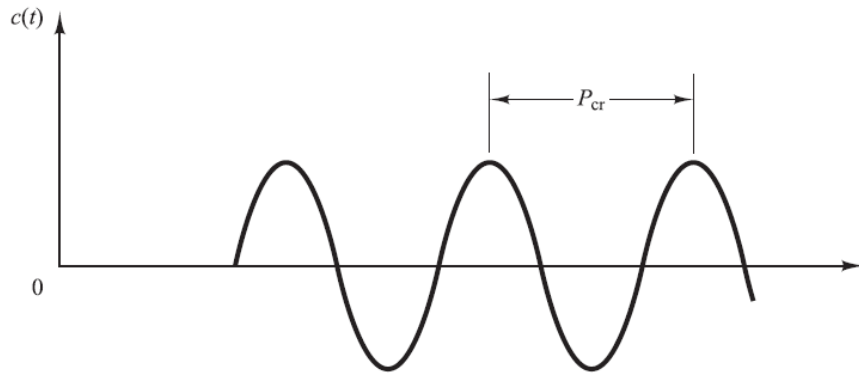


Figura 37: Oscilación sostenida con período P_{cr} .

Ziegler-Nichols sugirieron que se establecieran los valores de los parámetros K_p , T_i y T_d de acuerdo con la fórmula que se muestra en la Tabla 2.

Tabla 2: Regla de sintonía con la ganancia crítica K_{cr} y período crítico P_{cr} .

Tipo de controlador	K_p	T_i	T_d
P	$0.5K_{cr}$	∞	0
PI	$0.45K_{cr}$	$\frac{1}{1.2}P_{cr}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

El controlador PID sintonizado mediante el segundo método de las reglas de Ziegler-Nichols produce:

$$\begin{aligned}
 G_c(s) &= K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \\
 &= 0.6K_{cr} \left(1 + \frac{1}{0.5P_{cr}s} + 0.125P_{cr}s \right) \\
 &= 0.075K_{cr}P_{cr} \frac{\left(s + \frac{4}{P_{cr}} \right)^2}{s}
 \end{aligned}$$

Por tanto, el controlador PID tiene un polo en el origen y un cero doble en $s = -4 / P_{cr}$.

Conviene darse cuenta de que, si el sistema tiene un modelo matemático conocido (como la función de transferencia), entonces se puede emplear el

método del lugar de las raíces para encontrar la ganancia crítica K_{cr} y las frecuencias de las oscilaciones sostenidas ω_{cr} , donde $2\pi/\omega_{cr} = P_{cr}$. Estos valores se pueden determinar a partir de los puntos de cruce de las ramas del lugar de las raíces con el eje $j\omega$. (Obviamente, si las ramas del lugar de las raíces no cortan al eje $j\omega$ este método no se puede aplicar.)

2.6.2. Método de sintonización por ensayo sistemático [2]

Este método de ajuste requiere que el controlador y el proceso estén operando normalmente.

El procedimiento general consiste en empezar con un ajuste ancho de todas las acciones y estrecharlas poco a poco, en pasos sucesivos, hasta obtener la estabilidad deseada. En otras palabras, iniciar el ajuste con acción poco agresiva, o lo que es igual, valores conservadores. Este método emplea la reacción del lazo cerrado en un sistema que está bajo control, siendo el más utilizado en la práctica.

2.6.2.1. Sintonización de un controlador con acción proporcional

Este tipo de controlador tiene ajustable solamente la constante de proporcionalidad (ganancia). Si aumenta la ganancia, aumenta la inestabilidad. Por otro lado, si disminuye la ganancia, la desviación permanente u offset aumenta.

Teniendo en cuenta estas características se debe empezar el ajuste con una ganancia pequeña, aumentándola gradualmente en pasos sucesivos al mismo tiempo que se van realizando cambios de carga. Después de cada ajuste y cambio de carga siguiente, se debe observar la respuesta del proceso hasta que se consiga la estabilidad deseada. Puede considerarse terminado el ajuste del controlador cuando la respuesta de la variable controlada, ante un cambio de carga, tenga una relación de amortiguamiento de 0,25.

2.6.2.2. Sintonización de un controlador con acción proporcional más integral

La mayor parte de los controladores instalados en la industria disponen de estas dos acciones de control, por lo que necesitan dos tipos de ajuste:

- Ganancia proporcional
- Tiempo de reajuste o tiempo integral

El ajuste por ensayo sistemático se realiza en dos pasos:

- *Paso 1:* Con el tiempo integral a su más alto valor, se debe seguir el procedimiento descrito para el controlador proporcional, aumentando la ganancia hasta que se obtenga una relación de amortiguamiento de aproximadamente 0,25. Después se disminuye la ganancia ligeramente.
- *Paso 2:* Mantener la ganancia en el último valor obtenido en el paso 1. Después, empezando con el tiempo integral más alto, se va disminuyendo en pequeños incrementos, creando al mismo tiempo cambios de carga con el punto de consigna hasta que la oscilación se considere excesiva en amplitud. A continuación aumentar ligeramente el tiempo integral.

2.6.2.3. Sintonización de un controlador con acción proporcional más derivativa

Para llevar a cabo este ajuste por el método de ensayo sistemático se requieren dos pasos:

- *Paso 1:* Con el tiempo derivativo a cero, o su mínimo valor, se debe seguir el procedimiento descrito para el controlador proporcional, aumentando la ganancia hasta que se obtenga una relación de amortiguamiento de aproximadamente 0,25.
- *Paso 2:* Mantener la ganancia en el último valor obtenido en el paso 1. Después, empezando con el tiempo derivativo más bajo, se va aumentando en pequeños incrementos, creando al mismo tiempo cambios de carga con el punto de consigna hasta que la oscilación se considere excesiva en amplitud. A continuación reducir ligeramente el tiempo derivativo. Después se puede aumentar ligeramente la ganancia proporcional para mejorar el comportamiento del tiempo derivativo.

2.6.2.4. Sintonización de un controlador con acción proporcional, integral y derivativa

Para realizar este ajuste por el método de ensayo sistemático se requieren tres pasos:

- *Paso 1:* Con el tiempo integral a su máximo valor y el tiempo derivativo a cero se debe seguir el procedimiento descrito para el controlador proporcional, aumentando la ganancia hasta que se obtenga una relación de amortiguamiento de aproximadamente 0,25.

- *Paso 2:* Mantener la ganancia en el último valor obtenido en el paso 1.

Empezando con el tiempo integral más alto se va disminuyendo en pequeños incrementos, creando al mismo tiempo cambios de carga con el punto de consigna, hasta que la oscilación sea excesiva.

- *Paso 3:* Con la ganancia y tiempo integral en los últimos valores obtenidos en los pasos 1 y 2, aumentar el tiempo derivativo en pequeños incrementos, creando al mismo tiempo cambios de carga con el punto de consigna, hasta que la oscilación sea excesiva. A continuación reducir ligeramente el tiempo derivativo.

Con posterioridad se puede aumentar ligeramente la ganancia proporcional para mejorar el resultado del tiempo derivativo.

2.6.2.5. Efecto de los parámetros de sintonía

Un ajuste excesivamente crítico de los parámetros de sintonía en los controladores con acción proporcional, integral y derivativa puede ocasionar oscilación en los ciclos de respuesta. En la práctica, la dificultad está en conocer qué acción tiene un ajuste demasiado crítico, puesto que los ciclos pueden ser debidos a cualquiera de ellas.

Para distinguir los tipos de ciclo se puede utilizar la regla siguiente:

- El ciclo debido a la acción integral tiene un período relativamente largo. Al disminuir el tiempo integral aumenta la oscilación, pero con un período largo.
- El ciclo debido a la acción proporcional tiene un período relativamente moderado. Al aumentar la ganancia proporcional aumenta la oscilación, aumentando su frecuencia.
- El ciclo debido a la acción derivativa tiene un período relativamente corto. Al aumentar el tiempo derivativo aumenta la oscilación. En este caso la frecuencia de oscilación es la más alta de todas.

2.7. Sensores de temperatura [8]

La temperatura cuantifica la actividad de las moléculas de la materia. Está relacionada con la energía cinética, que es la energía asociada a los movimientos de las partículas del sistema. Cuanta mayor energía, mayor será la temperatura.

Intuitivamente, la temperatura es una propiedad de la materia que podemos descubrir cuando la tocamos. Podemos percibir que un material está caliente o frío.

El instrumento clásico con el que se mide la temperatura es el termómetro. Dicho instrumento está calibrado según distintas escalas que dependen de la zona del planeta en el que se encuentre.

Históricamente se desarrollaron una serie de escalas que tomaban distintos puntos de referencia; en la actualidad, hay tres escalas internacionalmente reconocidas, la escala en grados Celsius, la escala en grados Kelvin y la escala en grados Fahrenheit.

El avance de la tecnología ha hecho evolucionar los termómetros, y en la actualidad se encuentran toda una amplia gama de dispositivos que miden la temperatura de distintas formas. Existen los termómetros convencionales, cuyo funcionamiento consiste en mostrar visualmente la temperatura en una escala de valores, tenemos los sensores de temperatura, que ofrecen cambios en magnitudes eléctricas cuando cambia la temperatura y, de esta forma, se pueden conectar a dispositivos eléctricos o electrónicos y por último, tenemos los termostatos, los cuales modifican mecánicamente su comportamiento.

2.7.1. Termostato [9]

Un termostato es el componente de un sistema de control simple que abre o cierra un circuito eléctrico en función de la temperatura.

Su versión más simple consiste en una lámina metálica como la que utilizan los equipos de aire acondicionado para apagar o encender el compresor.

2.7.1.1. Termostato eléctrico

Este tipo de termostato recibe el nombre de bimetalico porque está formado por dos metales diferentes, actuando de manera diferente en función de la temperatura. Cuando se calienta o enfría la unión, en un momento dado, una de las dos piezas dilata o contrae; la otra pieza no ha tenido movimiento. Con ese movimiento se consigue que ambas partes se

unan o separan y por tanto, pueda pasar o no, la corriente eléctrica que cierra el circuito de control.



Figura 38: Termostato bimetalico.

2.7.1.2. Termostato mecánico

En los termostatos mecánicos, el cambio de posición provoca la apertura o cierre de una abertura. Funciona en este caso a modo de válvula que va abriéndose a medida que aumenta la temperatura. Este tipo de termostato es muy usado en automoción para la refrigeración del motor del coche.

Constructivamente, estos termostatos están formados por un gas encerrado en un tubo que al dilatar o contraerse, regula la abertura de una válvula.

2.7.1.3. Termostato digital

Otro tipo de dispositivos a los que comúnmente se les llama termostato son aquellos dispositivos electrónicos que haciendo uso de sensores de temperatura, microprocesadores y relés controlan todo tipo de circuitos. Dado que estos incluyen electrónica programable, el abanico de posibilidades de sus características es mucho más elevado que en las otras dos modalidades de termostato tradicional.

Sus aplicaciones son múltiples, destacando en calefacción, para regular la temperatura de la vivienda y en energía solar, donde los termostatos son diferenciales, pues deben controlar la temperatura tanto de la fuente de calor (las placas solares) como del acumulador.



Figura 39: Termostato digital.

2.7.2. Termistores [10]

Los termistores son sensores de temperatura resistivos que cambian su resistencia de acuerdo con las variaciones de temperatura. Existen dos tipos de termistor, aquellos cuya resistencia aumenta en función de la temperatura, también llamados PTC (Positive Temperature Coefficient) y aquellos cuya resistencia disminuye conforme aumenta la temperatura, llamados NTC (Negative Temperature Coefficient).

El término termistor proviene de Thermally Sensitive Resistor y el material base con que están fabricados es un semiconductor (igual que los transistores) que deja pasar parcialmente la corriente.

La ventaja de los termistores frente a otros sensores de temperatura es el bajo precio de estos, su amplio rango de medida y lo extendidos que se encuentran, lo que permite encontrar dispositivos a los que se pueden conectar sin mayores problemas.

La desventaja principal es que no son lineales, lo que dificulta la adquisición de datos y son complicados de calibrar.

De igual forma que otros sensores resistivos, los termistores acusan el efecto del autocalentamiento por el paso de corriente, por lo que hay que ser cuidadosos en la tensión y corriente que hacemos circular por el sensor para evitar falsos aumentos de temperatura.

2.7.2.1. Termistores PTC

Los termistores PTC son resistencias con un coeficiente de temperatura positivo, esto significa que al aumentar la temperatura también aumentará la su resistividad. El coeficiente de temperatura de un termistor PTC es

único entre unos determinados márgenes de temperaturas. Fuera de estos márgenes, el coeficiente de temperatura es cero o negativo.

Los termistores PTC se utilizan en una gran variedad de aplicaciones, incluyendo limitación de corrientes, como sensor de temperatura, para desmagnetizar las pantallas de tubo de rayos catódicos y para la protección contra el recalentamiento de equipos tales como motores eléctricos. También se utilizan en indicadores de nivel, para provocar retardo en circuitos y como resistores de compensación.

Los encapsulados son muy diversos, podemos encontrarlos en formato tipo disco o lenteja, tipo chip, sonda (cilindro tipo lápiz), axial o tipo perla.

Hasta un determinado valor de voltaje, la característica Intensidad / Voltaje sigue la ley de Ohm, pero la resistencia aumenta cuando la corriente que pasa por el termistor PTC provoca un calentamiento y se alcanza la temperatura de conmutación, temperatura a la cual el valor de la resistencia es igual al doble del que tiene a 25°C. La característica Intensidad / Voltaje depende de la temperatura ambiente y del coeficiente de transferencia de calor con respecto a dicha temperatura ambiente.

Si la temperatura llega a ser demasiado alta, la resistencia del termistor PTC pierde sus propiedades y puede comportarse eventualmente de una forma similar a una resistencia NTC. Por lo tanto, las aplicaciones de una resistencia PTC están restringidas a un determinado margen de temperaturas.

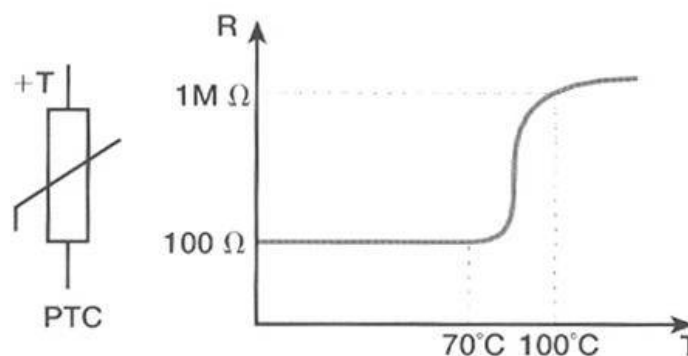


Figura 40: Curva de respuesta de un PTC.

2.7.2.2. Termistores NTC

Los termistores NTC (Negative Temperature Coefficient) son resistencias que varían su valor óhmico de acuerdo a las variaciones de la temperatura. Como su coeficiente de temperatura es negativo, la resistencia será menor cuando la temperatura empiece a aumentar.

Los termistores NTC son baratos, exactos y adecuados para muchas aplicaciones en la industria. Los instrumentos de medición que utilizan NTC no requieren compensación del punto de referencia. Esto los hace ideales para el uso en recintos fríos y en cámaras frigoríficas.

Si se experimenta con un termistor NTC se podrá observar que su comportamiento no es lineal, esto se debe a que la gráfica trazada para unos pequeños incrementos de temperatura toma una forma hiperbólica.

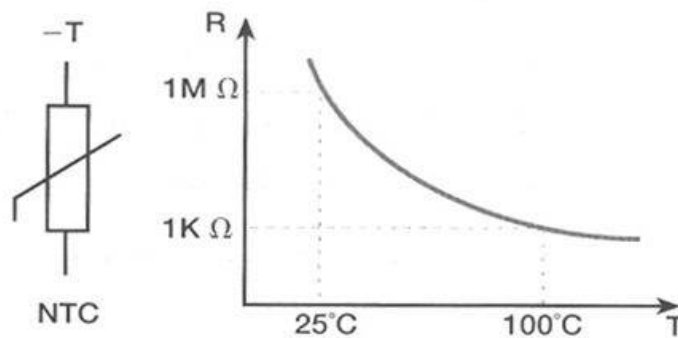


Figura 41: Curva de respuesta de un NTC.

2.7.3. Termopares [11]

Los termopares, también conocidos como termocuplas por la traducción del término inglés Thermocouple, es un transductor formado por la unión de dos metales distintos que produce una diferencia de potencial muy pequeña (del orden de los milivoltios) que es función de la diferencia de temperatura entre uno de los extremos denominado «punto caliente» o «unión caliente» o de «medida» y el otro llamado «punto frío» o «unión fría» o de «referencia» (efecto Seebeck).

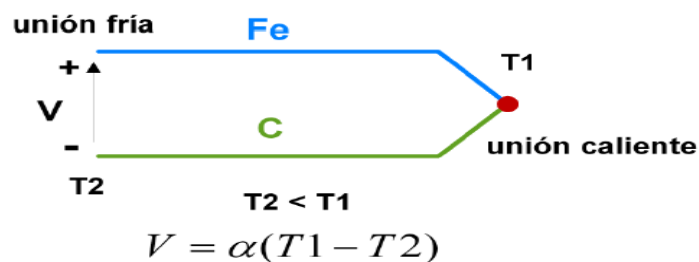


Figura 42: Efecto Seebeck.

Normalmente los termopares industriales están compuestos por un tubo de acero inoxidable u otro material. En un extremo del tubo está la unión, y en el otro el terminal eléctrico de los cables, protegido dentro de una caja redonda de aluminio (cabezal).



Figura 43: Termopares industriales.

En instrumentación industrial, los termopares son usados como sensores de temperatura. Son económicos, intercambiables, tienen conectores estándar y son capaces de medir un amplio rango de temperaturas. Su principal limitación está en la exactitud, pues es fácil obtener errores del sistema cuando se trabaja con precisiones inferiores a un grado Celsius.

Los termopares están disponibles en diferentes formatos, como sondas. Estas últimas son ideales para variadas aplicaciones de medición, por ejemplo, en la investigación médica, sensores de temperatura para los alimentos, en la industria y en otras ramas de la ciencia, etc.



Figura 44: Termopar tipo sonda.

A la hora de seleccionar una sonda de este tipo debe tenerse en consideración el tipo de conector. Los dos tipos son el modelo «estándar», con pines redondos y el modelo «miniatura», con pines chatos, siendo estos últimos (contradictoriamente al nombre de los primeros) los más populares.

Otro punto importante en la selección es el tipo de termopar, el aislamiento y la construcción de la sonda. Todos estos factores tienen un efecto en el rango de temperatura a medir, precisión y fiabilidad en las lecturas.

A continuación, se habla brevemente de los tipos de termopares que existen en el mercado:

- *Tipo K (cromel/alumel):* Con una amplia variedad aplicaciones, está disponible a un bajo costo y en una variedad de sondas. El cromel es una aleación de Ni-Cr, y el alumel es una aleación de Ni-Al. Tienen un rango de temperatura de $-200\text{ }^{\circ}\text{C}$ a $+1372\text{ }^{\circ}\text{C}$ y una sensibilidad $41\text{ }\mu\text{V}/^{\circ}\text{C}$ aproximadamente. Posee buena resistencia a la oxidación.
- *Tipo E (cromel/constantán):* No son magnéticos y gracias a su sensibilidad, son ideales para el uso en bajas temperaturas, en el ámbito criogénico. Tienen una sensibilidad de $68\text{ }\mu\text{V}/^{\circ}\text{C}$.
- *Tipo J (hierro/constantán):* Su rango de utilización es de $-270/+1200\text{ }^{\circ}\text{C}$. Debido a sus características se recomienda su uso en atmósferas inertes, reductoras o en vacío, su uso continuado a $800\text{ }^{\circ}\text{C}$ no presenta problemas, su principal inconveniente es la rápida oxidación que sufre el hierro por encima de $550\text{ }^{\circ}\text{C}$; y por debajo de $0\text{ }^{\circ}\text{C}$ es necesario tomar precauciones a causa de la condensación de vapor de agua sobre el hierro.
- *Tipo T (cobre/constantán):* Ideales para mediciones entre -200 y $260\text{ }^{\circ}\text{C}$. Resisten atmósferas húmedas, reductoras y oxidantes y son aplicables en criogenia. El tipo termopar de T tiene una sensibilidad de cerca de $43\text{ }\mu\text{V}/^{\circ}\text{C}$.
- *Tipo N (nicrosil/nisil):* Es adecuado para mediciones de alta temperatura gracias a su elevada estabilidad y resistencia a la oxidación, y no necesita del platino utilizado en los tipos B, R y S, que son más caros.
Por otro lado, los termopares tipo B, R y S son los más estables, pero debido a su baja sensibilidad ($10\text{ }\mu\text{V}/^{\circ}\text{C}$ aprox.) generalmente son usados para medir altas temperaturas (superiores a $300\text{ }^{\circ}\text{C}$).
- *Tipo B (Pt-Rh):* Son adecuados para la medición de altas temperaturas superiores a $1800\text{ }^{\circ}\text{C}$. Los del tipo B presentan el mismo resultado a $0\text{ }^{\circ}\text{C}$ y $42\text{ }^{\circ}\text{C}$ debido a su curva de temperatura/voltaje, limitando así su uso a temperaturas por encima de $50\text{ }^{\circ}\text{C}$.

- *Tipo R (Pt-Rh)*: Adecuados para la medición de temperaturas de hasta 1300 °C. Su baja sensibilidad (10 $\mu\text{V}/^\circ\text{C}$) y su elevado precio quitan su atractivo.
- *Tipo S (Pt/Rh)*: Ideales para mediciones de altas temperaturas hasta los 1300 °C, pero su baja sensibilidad (10 $\mu\text{V}/^\circ\text{C}$) y su elevado precio lo convierten en un instrumento no adecuado para el uso general. Debido a su elevada estabilidad, el tipo S es utilizado para la calibración universal del punto de fusión del oro (1064,43 °C).

2.7.4. RTD (Resistance Temperature Detector) [11]

Un RTD es un sensor de temperatura basado en la variación de la resistencia de un conductor con la temperatura. Los metales empleados normalmente como RTD son platino, cobre, níquel y molibdeno. De entre los anteriores, los sensores de platino son los más comunes por tener mejor linealidad, más rapidez y mayor margen de temperatura.

El Pt100 es un sensor de temperatura que a 0°C tiene una resistencia de 100 ohmios. Al aumentar la temperatura, la resistencia eléctrica también aumentará. Este sensor es sensible a la temperatura de cualquier termómetro de resistencia.

El incremento de la resistencia del Pt100 no es lineal pero si es creciente, de tal forma que mediante tablas es posible encontrar la temperatura exacta a la que corresponde.

Normalmente las sondas Pt100 industriales se fabrican encapsuladas en la misma forma que los termopares, es decir dentro de un tubo de acero inoxidable u otro material (vaina). En un extremo está el elemento sensible (Sensor RTD) y en el otro está el terminal eléctrico de los cables protegido dentro de una caja redonda de aluminio (cabezal).

Dependiendo de la forma de construcción, la parte bobinada del Pt100 va encapsulada en un cuerpo cerámico o de vidrio, o bien se encuentra como fina capa sobre una plaquita cerámica. Los terminales del elemento de medida están unidos a la parte resistiva activa de forma que resistan a las vibraciones.

Su rango de medida es del orden de -200°C a 600°C. Su exactitud es bastante buena y la tolerancia de su valor nominal R0 puede ser de 0,06 Ω . La velocidad de respuesta depende del formato en que se presente y el fabricante dará su valor en las hojas de características.



Figura 45: Pt-100.

La forma más simple de medir temperatura con un Pt100, es hacer pasar una corriente constante a través de él y medir la tensión en sus extremos. Como la corriente es fija y la resistencia depende de la temperatura, la tensión medida será proporcional a la temperatura.

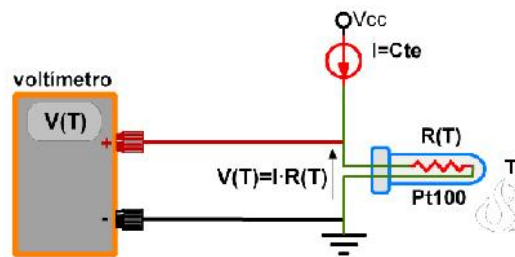


Figura 46: Conexión a dos hilos.

Al hacer circular corriente por el Pt100 este se calienta, lo que introduce un error en la medida llamado “error por autocalentamiento”. Hay que consultar los datos del fabricante para mantener este error lo más bajo posible según la exactitud que queramos conseguir. Los fabricantes dan tablas para calcular el error cometido según la corriente que circula por la resistencia.

Otro problema que surge habitualmente es que el Pt100 y el equipo electrónico donde está el acondicionador de señal están a una cierta distancia, por lo que la pequeña resistencia de los cables que unen el sensor al equipo, provoca una caída de tensión al ser atravesados por la corriente, que supone un error en la medida.

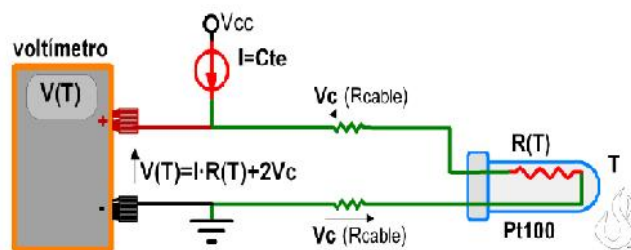


Figura 47: Error de medición - Conexión a dos hilos.

Una solución para este problema es utilizar una conexión de 4 hilos. En la figura se ve el esquema de medida a 4 hilos. El medidor no ve la caída de tensión en los cables por los que circula la corriente $V / (R_{\text{cable}})$, por lo que no se introduce error en la medida. Los cables de medida también tienen resistencia, pero no influyen en la medida porque por ellos no circula corriente.

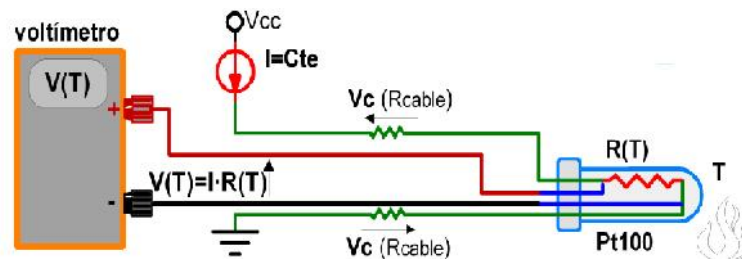


Figura 48: Conexión a cuatro hilos.

2.8. LabVIEW

LabVIEW es un software de ingeniería de sistemas que requiere pruebas, medidas y control con acceso rápido a hardware y análisis de datos.

Este programa fue creado por National Instruments (1976) para funcionar sobre máquinas MAC y salió al mercado por primera vez en 1986. Ahora está disponible para las plataformas Windows, UNIX, MAC y GNU/Linux.

El entorno de programación de LabVIEW simplifica la integración de hardware para aplicaciones de ingeniería, así se tiene una manera consistente de adquirir datos desde hardware de NI o de otros fabricantes.

LabVIEW reduce la complejidad de la programación y además permite visualizar resultados inmediatamente con la creación integrada de interfaces de usuario y visualizadores de datos integrados. Para garantizar la compatibilidad con otras herramientas de ingeniería, LabVIEW puede interactuar o reutilizar bibliotecas de otros software y lenguajes de fuente abierta.

Los programas desarrollados con LabVIEW se llaman Instrumentos Virtuales (VI), y su origen proviene del control de instrumentos, aunque hoy en día se ha expandido ampliamente no sólo al control de todo tipo de electrónica (Instrumentación electrónica) sino también a la programación embebida, comunicaciones, matemáticas, etc.

Entre sus objetivos están el reducir el tiempo de desarrollo de aplicaciones de todo tipo (no sólo en ámbitos de Pruebas, Control y Diseño) y el permitir la entrada a la informática a profesionales de cualquier otro campo. LabVIEW consigue combinarse con todo tipo de software y hardware, tanto del propio fabricante (tarjetas de adquisición de datos, PAC, Visión, instrumentos y otro Hardware) como de terceros.

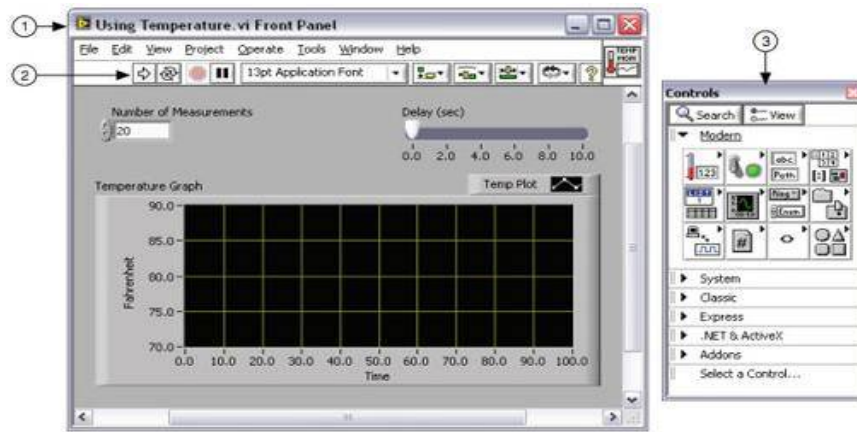
Su principal característica es la facilidad de uso, válido para programadores profesionales como para personas con pocos conocimientos en programación que pueden hacer programas relativamente complejos, imposibles de hacer con lenguajes tradicionales. También es muy rápido hacer programas con LabVIEW y cualquier programador, por experimentado que sea, puede beneficiarse de él.



Figura 49: LabVIEW.

Como se ha dicho antes, LabVIEW es una herramienta gráfica de programación, esto significa que los programas no se escriben, sino que se dibujan, facilitando su comprensión. Al tener ya pre diseñados una gran cantidad de bloques, se le facilita al usuario la creación del proyecto, con lo cual en vez de estar una gran cantidad de tiempo en programar un dispositivo o bloque, se le permite invertir mucho menos tiempo y dedicarse un poco más en la interfaz gráfica y la interacción con el usuario final. Cada VI consta de dos partes diferenciadas:

- *Panel Frontal:* El Panel Frontal es la interfaz con el usuario, se utiliza para interactuar con el usuario cuando el programa se está ejecutando. Los usuarios podrán observar los datos del programa actualizados en tiempo. En esta interfaz se definen los controles (pueden ser botones, marcadores etc.) e indicadores (pueden ser gráficas).



(1) Ventana de Panel Frontal | (2) Barra de Herramientas | (3) Paleta de Controles

Figura 50: Panel Frontal de LabVIEW.

La paleta de controles contiene los controles e indicadores que se utilizan para crear el panel frontal. Se puede tener acceso a la paleta de controles de la ventana del panel frontal al seleccionar la opción View » Controls Palette o al dar clic con botón derecho en cualquier espacio en blanco en la ventana del panel frontal. La paleta de controles está dividida en varias categorías; podemos exponer algunas o todas estas categorías para cumplir con nuestras necesidades. La siguiente figura muestra la paleta de controles con todas las categorías expuestas.

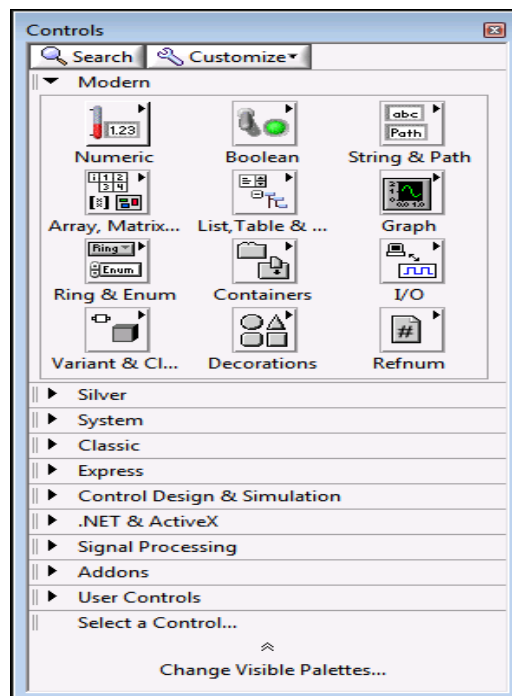


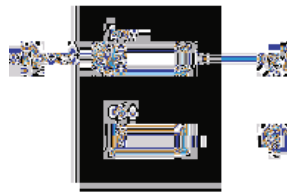
Figura 51: Paleta de controles en LabVIEW.

Cuando se interactúa con un panel frontal como una interfaz de usuario, se pueden modificar los controles para transmitir entradas y ver los resultados en los indicadores. Los controles definen las entradas y los indicadores muestran las salidas.

Los controles generalmente son perillas, botones, perillas, deslizadores y secuencias. Simulan dispositivos de entrada de instrumentos y suministran datos al diagrama de bloques del VI. Los indicadores generalmente son gráficas, tablas, LEDs y secuencias de estado. Los indicadores simulan dispositivos de salida de instrumentos y muestran los datos que el diagrama de bloques adquiere o genera.

Cada control o indicador tiene un tipo de datos asociado con él. Los tipos de datos utilizados con mayor frecuencia son numéricos, Booleano y cadena de caracteres.

El tipo de datos numérico pueden representar números de varios tipos como un entero o real. Los dos objetos numéricos comunes son el control numérico y el indicador numérico, como se muestra en la siguiente figura. Los objetos como medidores y perillas también representan datos numéricos.



(1) Botones de Incremento/Reducción | (2) Control Numérico | (3) Indicador Numérico

Figura 52: Controles e Indicadores numéricos.

El tipo de datos Booleano representa datos que solamente tienen dos estados posibles, como TRUE y FALSE u ON y OFF. Los controles e indicadores Booleanos se usan para proporcionar y visualizar valores Booleanos. Los objetos Booleanos simulan interruptores, botones y LEDs. El interruptor de encendido vertical y los objetos LED redondos se muestran en la siguiente figura.



Figura 53: Controles e Indicadores Booleanos.

El tipo de dato cadena de caracteres es una secuencia de caracteres ASCII. Los controles en cadena se usan para recibir texto desde el usuario como una contraseña o nombre. Los indicadores en cadena se usan para mostrar texto al usuario. Los objetos en cadena más comunes son tablas y cuadros de texto, como se muestra en la siguiente figura.

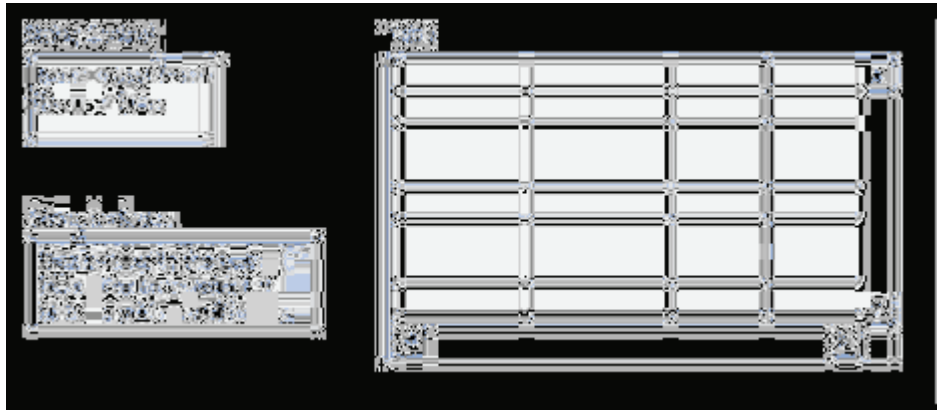
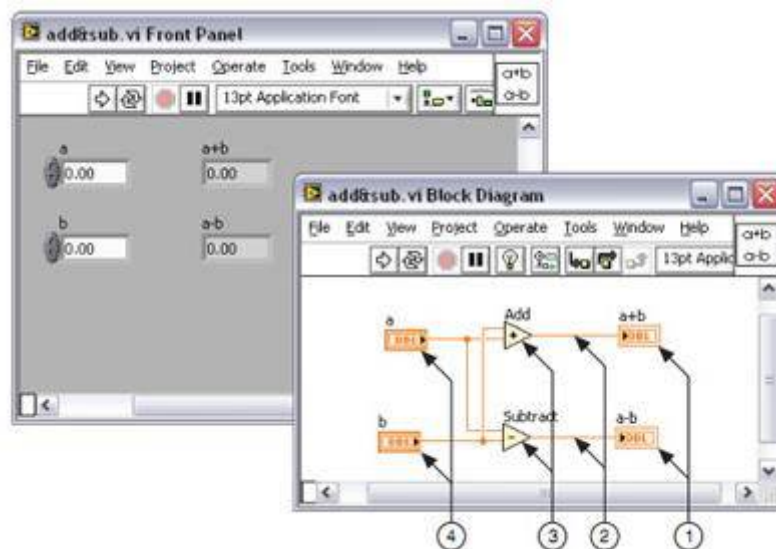


Figura 54: Controles e Indicadores de caracteres.

- Diagrama de Bloques:** Es el programa propiamente dicho, donde se define la funcionalidad del proyecto. Aquí se colocan íconos que realizan una determinada función y se interconectan (el código que controla el programa). Suele haber una tercera parte icono/conector que son los medios utilizados para conectar un VI con otros.



(1) Terminales de Indicador | (2) Cables | (3) Nodos | (4) Terminales de Control

Figura 55: Ejemplo de un Diagrama de Bloques y Panel Frontal.

Después de que se crea la ventana del panel frontal, se añade código usando representaciones gráficas de funciones para controlar los objetos del panel frontal.

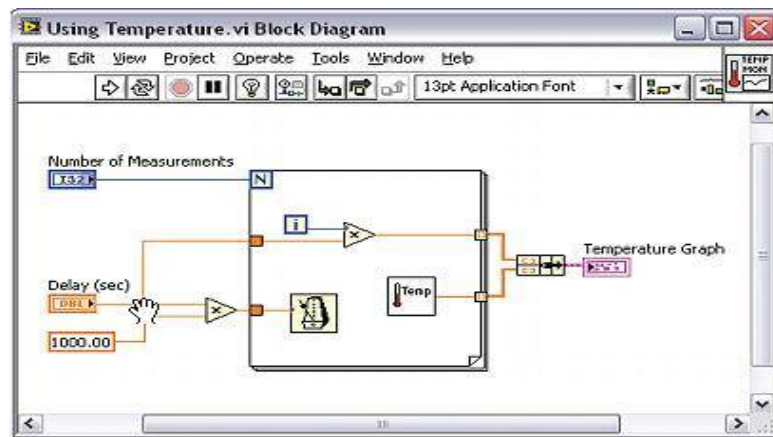


Figura 56: Diagrama de Bloques.

Los objetos en la ventana del panel frontal aparecen como terminales en el diagrama de bloques. Los terminales son puertos de entrada y salida que intercambian información entre el panel frontal y el diagrama de bloques. Son análogos a los parámetros y constantes en lenguajes de programación basados en texto.

Los controles, indicadores y constantes se comportan como entradas y salidas del algoritmo del diagrama de bloques.

Los nodos son objetos en el diagrama de bloques que tienen entradas y/o salidas y realizan operaciones cuando el VI se ejecuta. Son análogos a instrucciones, operaciones, funciones y sub rutinas en lenguajes de programación basados en texto. Los nodos pueden ser funciones, subVIs, Express VIs o estructuras. Las estructuras son elementos de control de procesos, como estructuras de Casos, Ciclos For o Ciclos While.

La paleta de funciones contiene los VIs, funciones y constantes que se pueden utilizar para crear el diagrama de bloques. Tenemos acceso a la paleta de funciones del diagrama de bloques al seleccionar View » Functions Palette. La paleta de funciones está dividida en varias categorías; podemos mostrar y esconder categorías para cumplir con nuestras necesidades. La siguiente figura muestra la paleta de funciones con todas las categorías expuestas y la categoría de Programación expandida.

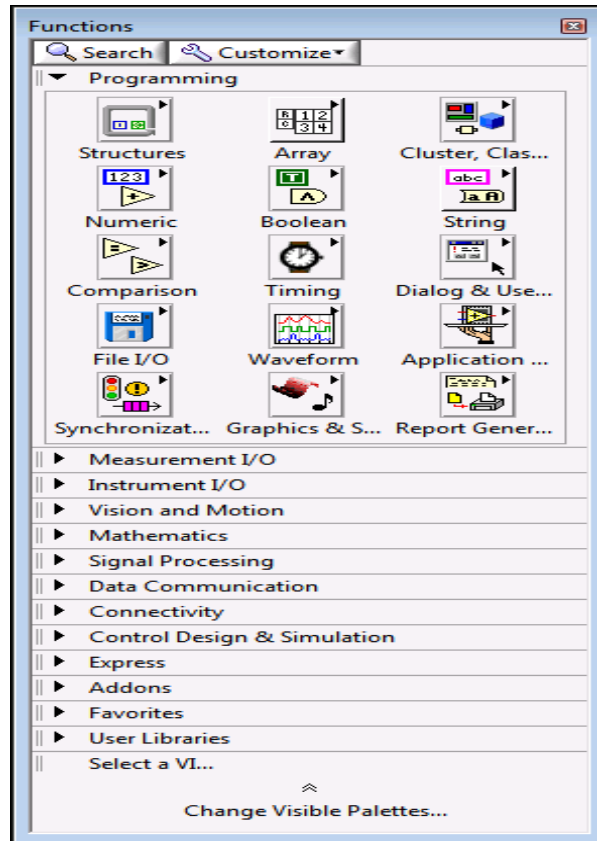


Figura 57: Paleta de funciones.

DISEÑO DEL SISTEMA

CAPÍTULO III

3. Diseño del módulo didáctico de control de temperatura

3.1. Diagrama de bloques del sistema de control

En el diagrama de bloques que se muestra en la figura 58 se puede observar que el sensor Pt – 100 toma datos de la temperatura de la planta y los envía a la computadora mediante una tarjeta de adquisición de datos (Arduino) que se encarga de digitalizar la lectura. La computadora envía una señal análoga a través de la tarjeta de adquisición de datos para la activación del foco para aumentar la temperatura y una señal digital para activar el ventilador y extraer el calor del módulo.

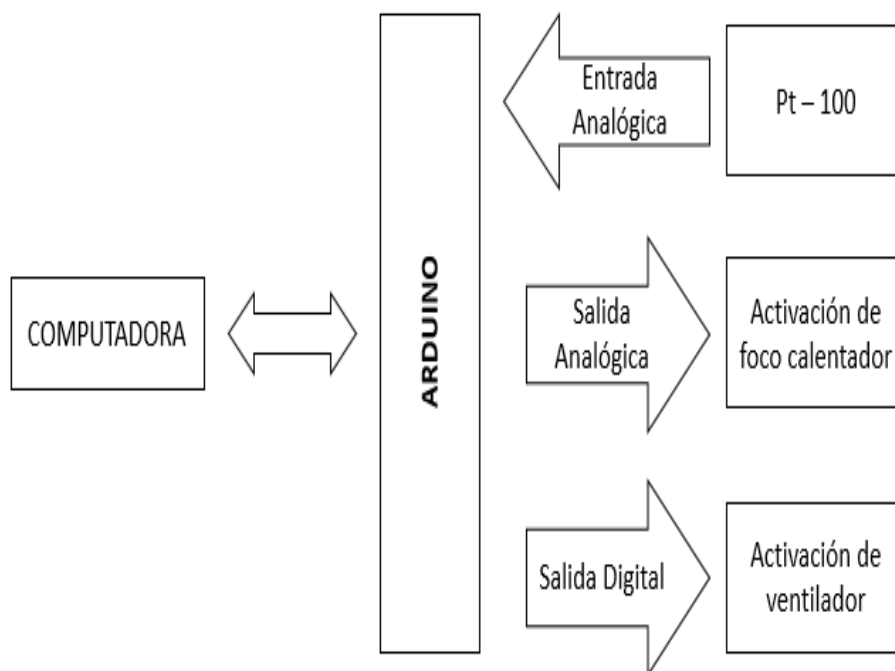


Figura 58: Sistema en lazo abierto.

3.2. Determinación del modelo matemático de la planta

3.2.1. Rango de linealidad de la planta

Haciendo las respectivas pruebas a la planta en lazo abierto como se muestra en el diagrama de bloques la figura 58, se aplicaron diferentes porcentajes de potencia eléctrica en la resistencia, la cual es proporcional al flujo de calor proporcionado por el foco, y se midió el valor de la temperatura en estado estacionario. Los valores obtenidos se muestran en la Tabla 3.

Tabla 3: Valores obtenidos haciendo pruebas experimentales.

Voltios (AC)	%	POTENCIA	Temp (T°)
0,0	0	0	26,10
62,9	10	8,18	35,80
89,0	20	16,36	42,80
109,0	30	24,55	48,50
125,9	40	32,73	55,50
140,7	50	40,91	58,00
154,1	60	49,09	61,00
166,5	70	57,27	64,50
178,0	80	65,46	67,00
188,8	90	73,64	68,80
199,0	100	81,82	73,20

Con los valores obtenidos se graficó la curva estática de la planta, la cual se muestra en la figura 59, y nos ayuda a determinar su rango lineal.

Se observa que presenta 2 zonas lineales comprendidas entre 0% y 40% y entre 40% y 90% de la potencia disipada por el foco, una vez excedido este valor la temperatura de la planta pues el ventilador no puede controlar la disipación.

Por esta razón, para las pruebas realizadas posteriormente se trabajó con valores de referencia entre 27.00 °C y 68.80 °C asegurando la operación dentro del rango lineal de la planta.

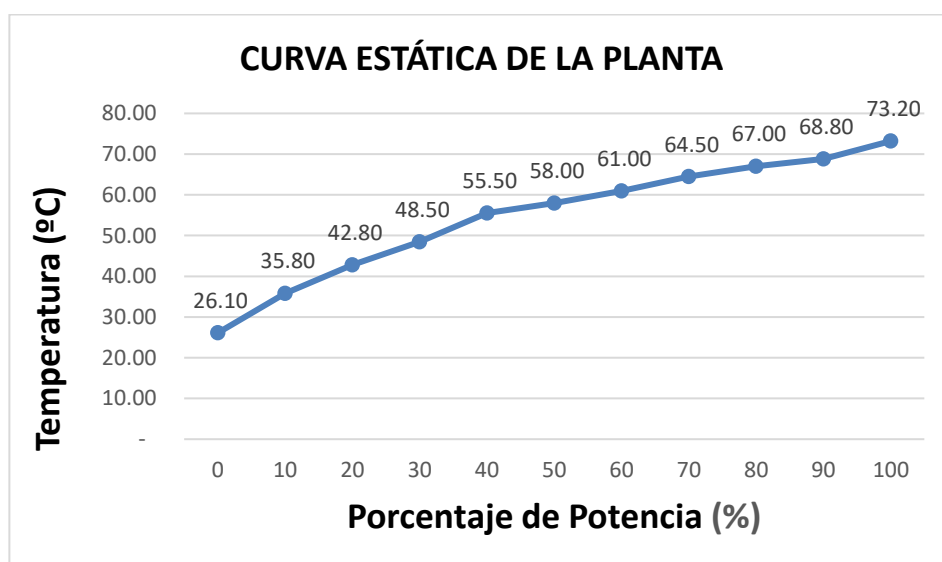


Figura 59: Curva estática de la planta.

3.2.2. Obtención del modelo mediante la respuesta al escalón

Para obtener el modelo matemático del sistema se realizaron pruebas basadas en la curva de respuesta. Estas pruebas se desarrollaron en lazo abierto según el diagrama mostrado en la figura 58, en esta configuración la computadora aún no tiene la función de controlador y es utilizada únicamente para monitorear la variable de temperatura y generar la señal de activación del foco.

Para un cambio de 0% a 40% de potencia disipada, se obtiene la siguiente respuesta:

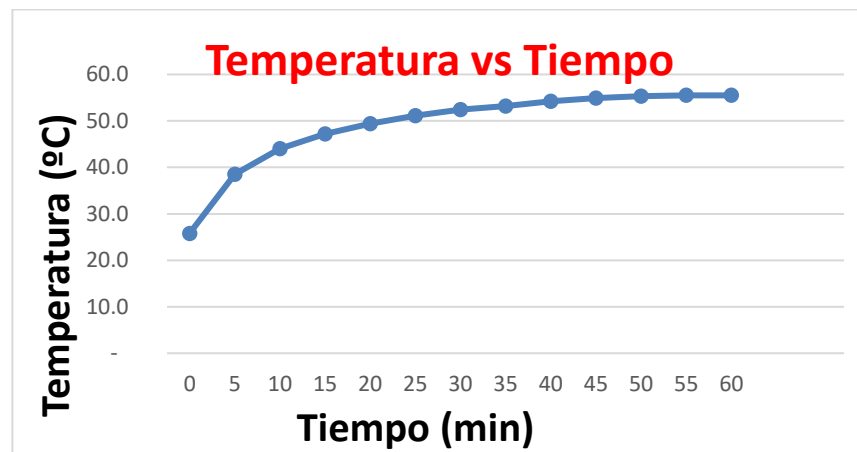


Figura 60: Curva estática de la planta para un cambio de 0% a 40% de potencia.

Tabla 4: Valores obtenidos para un cambio de 0% a 40% de potencia.

TIEMPO (Min)	T (°C)
0	25,8
5	38,5
10	44,0
15	47,2
20	49,4
25	51,1
30	52,4
35	53,2
40	54,2
45	54,9
50	55,3
55	55,5
60	55,5

Para un cambio de 0% a 90% de potencia disipada, se obtiene la siguiente respuesta:

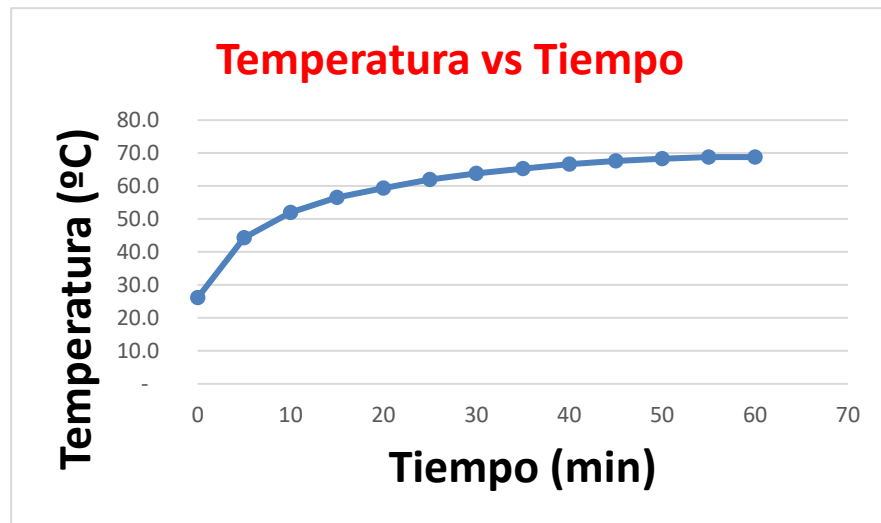


Figura 61: Curva estática de la planta para un cambio de 0% a 90% de potencia.

Tabla 5: Valores obtenidos para un cambio de 0% a 90% de potencia.

TIEMPO (Min)	T (°C)
0	26,1
5	44,3
10	52,0
15	56,5
20	59,4
25	62,0
30	63,8
35	65,3
40	66,6
45	67,6
50	68,3
55	68,8
60	68,8

Para un cambio de 0% a 100% de potencia disipada, se obtiene la siguiente respuesta:

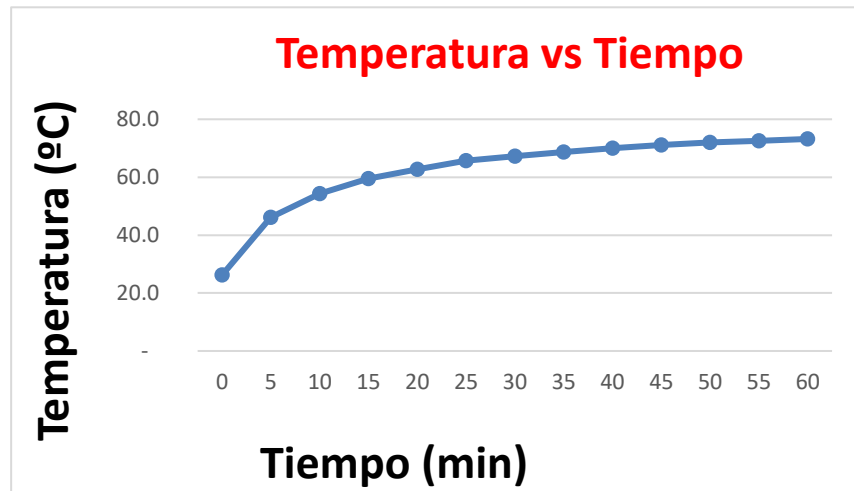


Figura 62: Curva estática de la planta para un cambio de 0% a 100% de potencia.

Tabla 6: Valores obtenidos para un cambio de 0% a 100% de potencia.

TIEMPO (Min)	T (°C)
0	26,2
5	46,1
10	54,3
15	59,5
20	62,7
25	65,7
30	67,2
35	68,7
40	70,0
45	71,1
50	72,0
55	72,6
60	73,2

Una vez obtenidas estas curvas de respuesta se puede identificar que el proceso se comporta como un sistema de primer orden.

Determinación de la constante de tiempo

Para todos los casos observados en los diferentes porcentajes de potencia se observa que la constante de tiempo multiplicada por cinco es de 55 minutos que es el tiempo en que demora en estabilizarse la temperatura de la cámara hecha la prueba en lazo abierto.

Por lo tanto:

$$5 \tau = 55 \text{ min}$$

Entonces:

$$\tau = 11 \text{ min}$$

Determinación de la ecuación de la curva de respuesta

La ecuación de la curva tendrá la siguiente forma:

$$f(t) = V_o - (V_o - V_i) * e^{(-t/\tau)}$$

Esta ecuación resulta de evaluar la curva de respuesta al 40% de la potencia ya que se encuentra dentro del rango de linealidad.

Donde:

Vo = Valor final, Vi = Valor inicial y "t" es el tiempo

Reemplazando los valores de la curva de respuesta obtenemos los siguientes valores:

$$V_o = 55.5 \text{ }^\circ\text{C}, V_i = 25.8 \text{ }^\circ\text{C} \text{ y } T = 11 \text{ min}$$

Por lo tanto la ecuación final es la siguiente:

$$f(t) = 55.5 - 29.7 * e^{(-t/11)}$$

Modelo Matemático final

Aplicando la transformada de Laplace a la ecuación de la curva de respuesta obtenemos la siguiente ecuación:

$$F(s) = \frac{55.5}{s} - \frac{29.7}{s + 0.0909}$$

Al dividir esta ecuación por $U(s) = \frac{29.7}{s}$, podremos hallar la función de transferencia de la planta.

La función de transferencia calculada sería:

$$H = \frac{25.8s + 5.04}{s + 0.0909}$$

3.3. Determinación y simulación del controlador en Matlab

Una vez que se obtiene la función de transferencia de la planta, se procederá a obtener los parámetros de un controlador PID utilizando el software Matlab.

El algoritmo desarrollado en Matlab estará basado en los métodos de Ziegler y Nichols ya que son los más utilizados cuando se quiere obtener un modelo matemático. Adicionalmente, podremos observar en graficas la respuesta al escalón de la planta, la ubicación de los polos y una comparación entre la respuesta original del sistema y la respuesta con el controlador PID.

```

1 - clear all
2 - clc
3 - num = [ 25.8 5.04 ]
4 - den = [ 1 0.0909 ]
5
6 - Gp = tf( num, den )
7 - H = [ 1 ]
8
9 - M = feedback( Gp, H )
10 - step(M)
11 - hold on
12
13 - %%%%%%%%%%%5
14 - Kp = 0
15 - Ki = 0
16 - Kd = 0
17
18 - Gc = pid( Kp, Ki, Kd )
19 - Mc = feedback( Gc*Gp, H )
20 - step(Mc)
21 - grid on
    
```

Figura 63: Código fuente en Matlab.

Al ejecutar el código anterior obtenemos la ecuación en lazo cerrado con realimentación unitaria:

$$HLC = \frac{25.8s + 5.04}{26.8s + 5.131}$$

Luego podemos observar que se genera una imagen con la respuesta al escalón como se muestra en la figura 63.

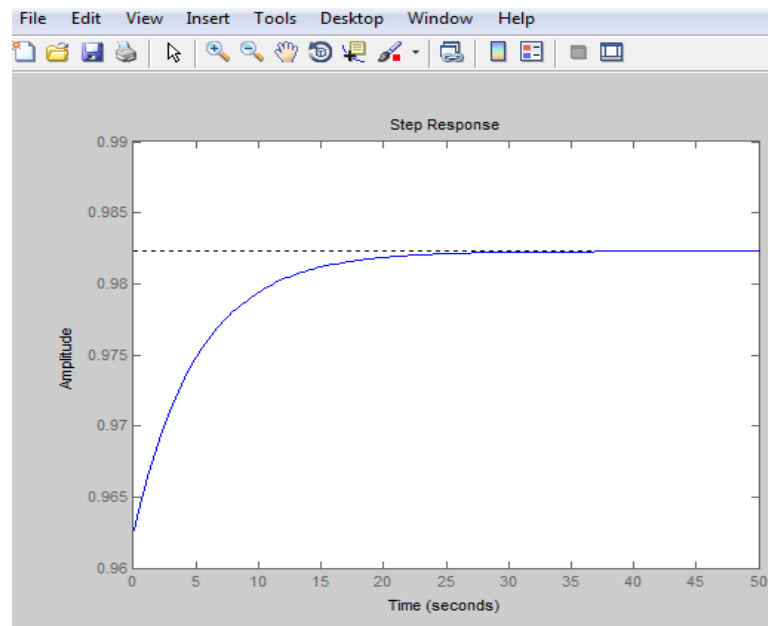


Figura 64: Respuesta al escalón del sistema en lazo cerrado.

En la figura anterior se puede observar que no se llega a la referencia deseada y además el tiempo de establecimiento es de 30 segundos aproximadamente.

Para mejorar esta respuesta se irán incrementando los parámetros del controlador PID (K_p , K_i , K_d), teniendo en cuenta el efecto que producen sobre la respuesta de la planta.

```

1 - clear all
2 - clc
3 - num = [ 25.8 5.04 ]
4 - den = [ 1 0.0909 ]
5
6 - Gp = tf( num, den )
7 - H = [ 1 ]
8
9 - M = feedback( Gp, H )
10 - step(M)
11 - hold on
12
13 - %%%%%%%%%%%5
14 - Kp = 3
15 - Ki = 0
16 - Kd = 0
17
18 - Gc = pid( Kp, Ki, Kd )
19 - Mc = feedback( Gc*Gp, H )
20 - step(Mc)
21 - grid on
    
```

Figura 65: Código para obtener la respuesta al escalón con $K_p = 3$.

Al ejecutar el código anterior se generan las siguientes curvas de respuesta al escalón. Los valores de K_p fueron 1, 2 y 3 y las curvas se muestran en la figura 65 con colores negro, amarillo y azul, respectivamente.

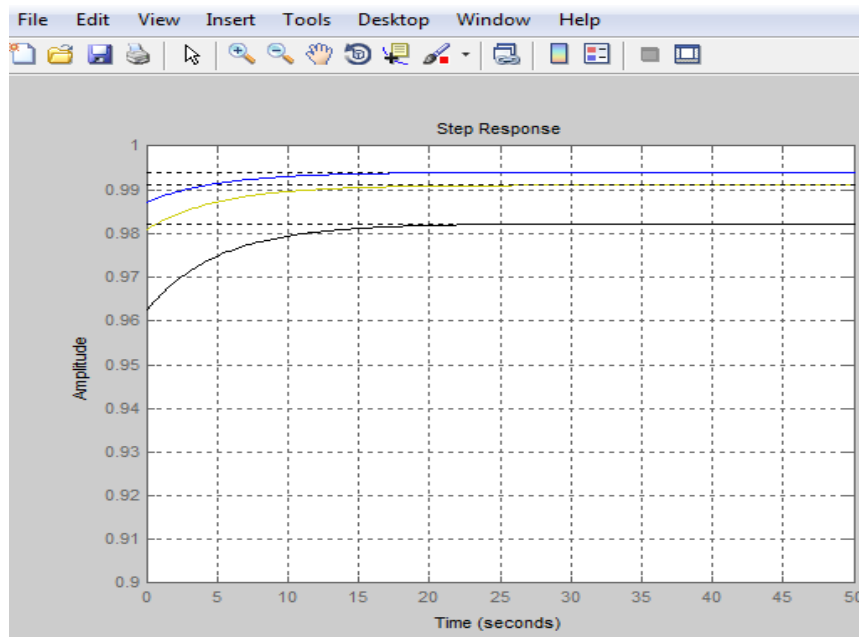


Figura 66: Respuesta al escalón con $K_p = 3$.

Para mejorar la respuesta del sistema, se procedió a incrementar la ganancia integral de uno en uno para ir analizando el comportamiento de la planta.

```

EDITOR      PUBLISH      VIEW
+   Folder Save Find Files
New Open Compare Print EDIT NAVIGATE Breakpoints Run Run and Run and Run Section
FILE          BREAKPOINTS      RUN
PID.m* x
1 - clear all
2 - clc
3 - num [ 25.8 5.04 ]
4 - den [ 1 0.0909 ]
5
6 - Gp tf( num, den )
7 - H [ 1 ]
8
9 - M feedback( Gp, H )
10 - step(M)
11 - hold on
12
13 #####5
14 - Kp 3
15 - Ki 4
16 - Kd
17 - Gc 1.0
18 - Mc 1.1 *Gp, H )
19 - step(Mc)
20 - grid on
21
22
    
```

Figura 67: Código para obtener la respuesta al escalón con $K_p = 3$ y $K_i = 4$.

Al ejecutar el código anterior se generan las siguientes curvas de respuesta al escalón. El valor de K_p se mantiene en 3 mientras que los valores de K_i fueron 1, 2, 3 y 4. Las curvas se muestran en la figura 67 con color celeste, amarillo y azul y rojo, respectivamente.

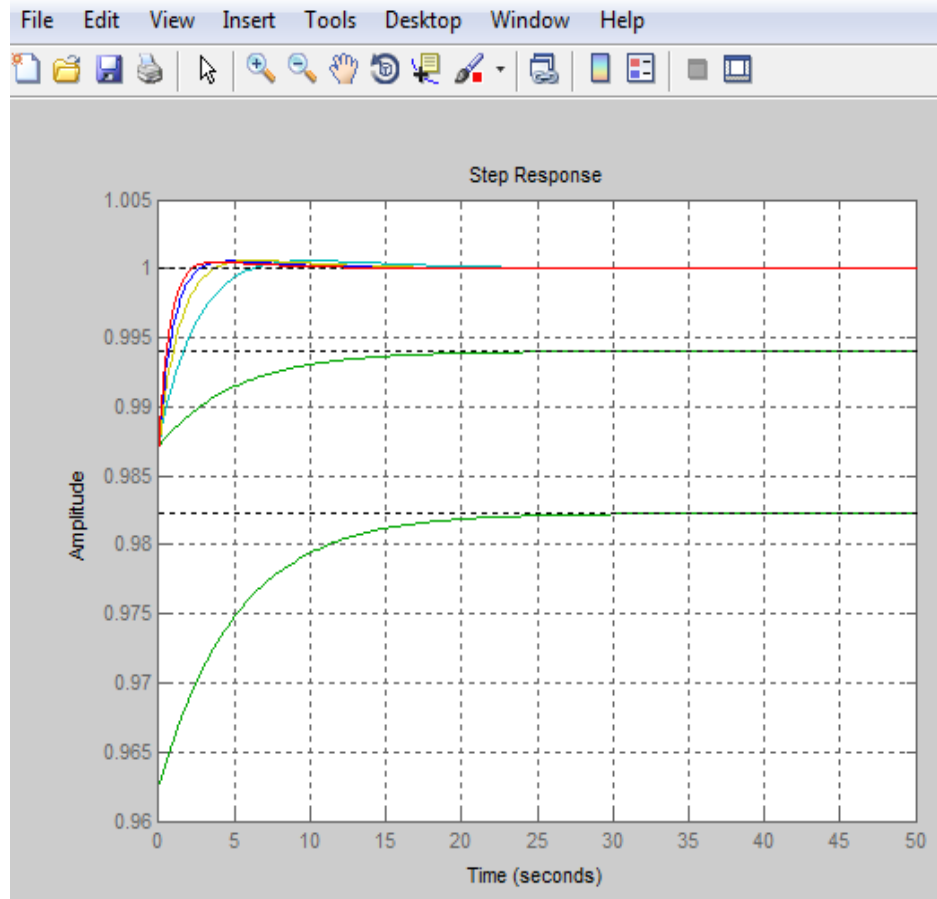


Figura 68: Respuesta al escalón con $K_p = 3$. Y $K_i = 4$.

Con $K_p = 3$, $K_i = 4$ y $K_d = 0$ se obtiene la curva roja que se muestra en la figura anterior, esta respuesta es mejor que con valores más bajos de K_i debido a que se llega a la referencia en menos de 10 segundos.

Por lo tanto la ecuación del controlador PID es la siguiente:

$$G = K_p + K_i * \frac{1}{s}$$

$$G = 3 + \frac{4}{s}$$

La ecuación de todo el sistema en lazo cerrado y con el controlador PID es la siguiente:

$$M = \frac{77.4s^2 + 118.3s + 20.16}{78.4s^2 + 118.4s + 20.16}$$

3.4. Diseño de la estructura del módulo didáctico

La parte interna del Módulo Didáctico de Control de temperatura contará con cuatro compartimientos:

- **Compartimento de fuente de poder:** Se utiliza una fuente de PC de la cual se utilizan cables con voltajes de 24V y de 12V DC.



Figura 69: Fuente de poder ATX.

- **Compartimento de circuitos:** En él se encuentran los circuitos que configuran la correcta instalación del transmisor de temperatura (T T) (de 0-200 °C / 4-20 mA) con el sensor Pt - 100 y la conexión del Arduino por el pin A0 para leer el voltaje que produce la corriente al pasar por la resistencia de 250 ohmios.

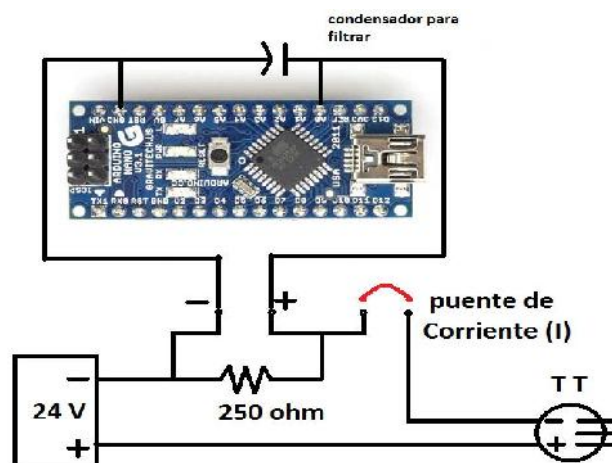


Figura 70: Conexión entre el TT y Arduino.

Este compartimento también alojara al Dimmer que alimentará la bombilla que influirá en la variación de la temperatura.

=

Figura 71: Circuito del Dimmer.

Por último se encuentra el Driver que controla el encendido del Cooler para enfriar la cámara térmica.

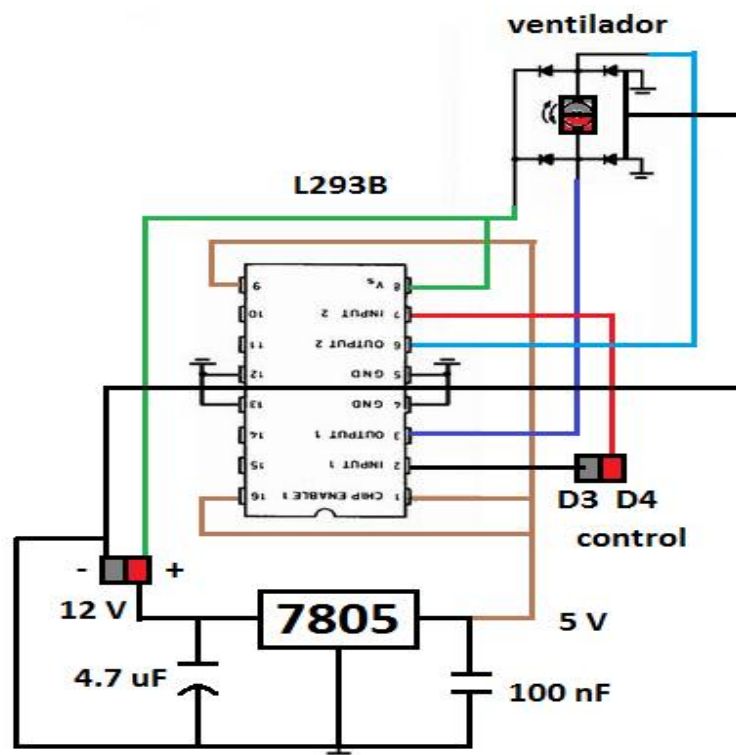


Figura 72: Circuito de control del ventilador.

- **Compartimento de la cámara térmica:** En ella se encuentran la bombilla de 100W a 220VAC, el Cooler (ventilador) de 12VDC y el Pt - 100.

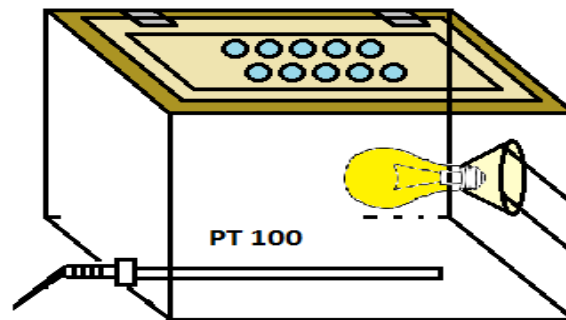


Figura 73: Cámara térmica.

- **Compartimento de cable:** Aquí se guardan los cables a utilizar para la conexión de las borneras y el de la conexión de poder a los 220VAC.



Figura 74: Borneras de conexión.

La parte externa del Módulo Didáctico de Control de temperatura contará con los siguientes elementos:

- **Interruptor de Poder:** Con el que se enciende o se apaga el Módulo de control de temperatura.



Figura 75: Interruptor de poder.

- **Puerto USB:** Permite la comunicación entre el Arduino y la interfaz creada en el software LabVIEW. Además, sirve para energizar al Arduino con 5 VDC.

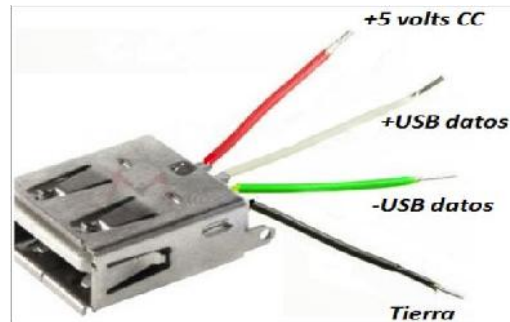


Figura 76: Puerto USB.

- **Perilla de Regulación:** Está formado por un potenciómetro el cual regula la potencia que el Dimmer entrega la bombilla.



Figura 77: Potenciómetro.

- **Borneras para medir voltaje:** Son dos Borneras conectadas en paralelo con la bombilla de la cámara térmica. Esto permite medir el voltaje que cae sobre la bombilla a cada instante.
- **Borneras para medir Corriente:** Está conectada en serie entre el polo Negativo del TT (del PT100) y a la resistencia de 250 ohmios. Las Borneras estarán puenteadas para tener un buen funcionamiento.
- **Borneras de Voltaje DC:** Están conectadas en paralelo a la resistencia de 250 ohmios para medir el voltaje que cae en ella, ya que es el mismo voltaje que es leído por el pin Analógico A0 del Arduino Nano.
- **Borneras del Pt-100:** Estas borneras están conectadas en paralelo al sensor al TT y al Pt-100 para poder medir su resistencia.

En la siguiente imagen se muestra la estructura final del Módulo Didáctico de Control de temperatura.

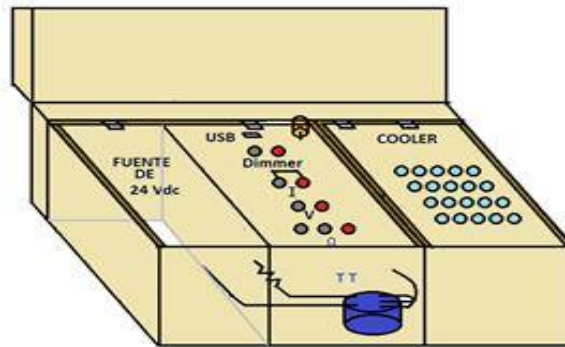


Figura 78: Estructura final del Módulo Didáctico.

3.5. Configuración de Arduino

Para comenzar, instalaremos el programa desde la página Web de National Instruments. A continuación, conectaremos el Arduino al ordenador a través del cable con el que le cargamos los programas.

Esta interfaz, la encontraremos en la carpeta donde hayamos instalado LabVIEW. A continuación, entramos en vi.lib y a continuación en LabVIEW Interface for Arduino. Tras esto, en Firmware y, finalmente, en LVIFA_Base. Cuando estemos dentro de la carpeta, abriremos el software de Arduino y cargaremos los programas. Con esto ya se puede utilizar Arduino con LabVIEW.

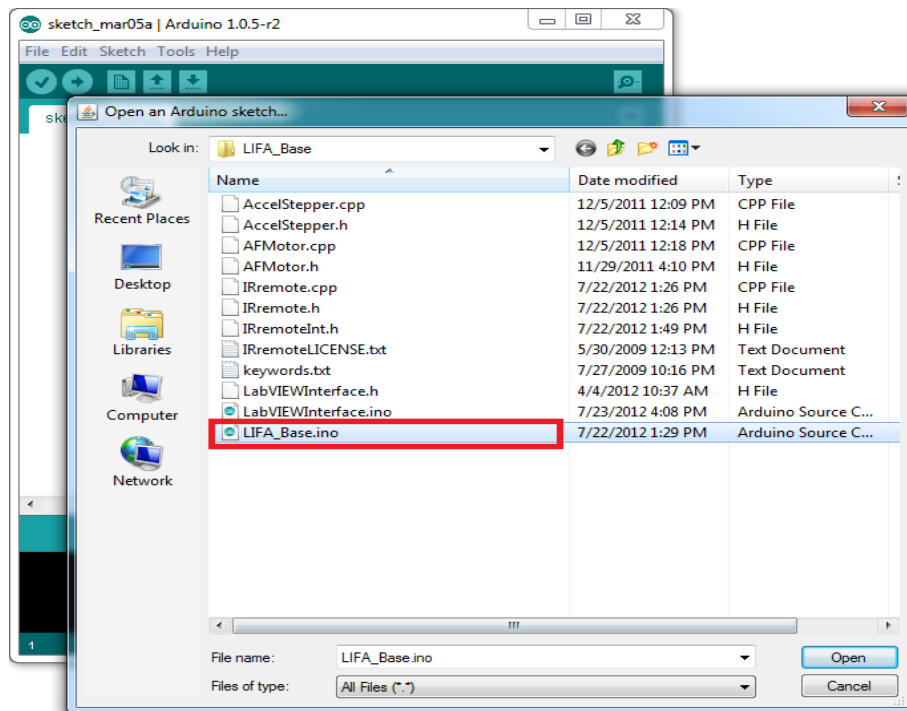


Figura 79: LabVIEW Interface for Arduino - LIFA.

3.6. Configuración de LabVIEW

Después de instalar la plataforma LabVIEW, debemos abrir VI Package Manager (el VI Package Manager viene incluido al instalar LabVIEW) y colocar en el Buscador “Arduino” y elegir “LabVIEW Interface for Arduino” para poder descargarlo (no cerrar hasta que se termine la instalación).

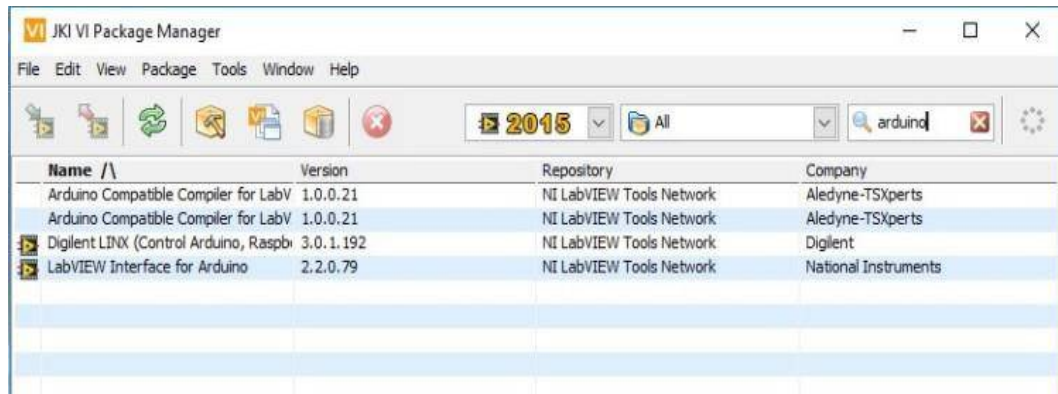


Figura 80: Buscador del VI Package Manager.

3.7. Diseño de la interfaz gráfica en LabVIEW

La interfaz de usuario permitirá elegir el puerto COM asignado a la tarjeta de adquisición de datos (Arduino), la visualización de la temperatura medida a través de un gráfico (Tiempo vs Temperatura) y además nos brindará la opción de encender y apagar el ventilador para extraer el calor del módulo. Adicionalmente, se podrá observar el voltaje medido en la entrada analógica del Arduino.

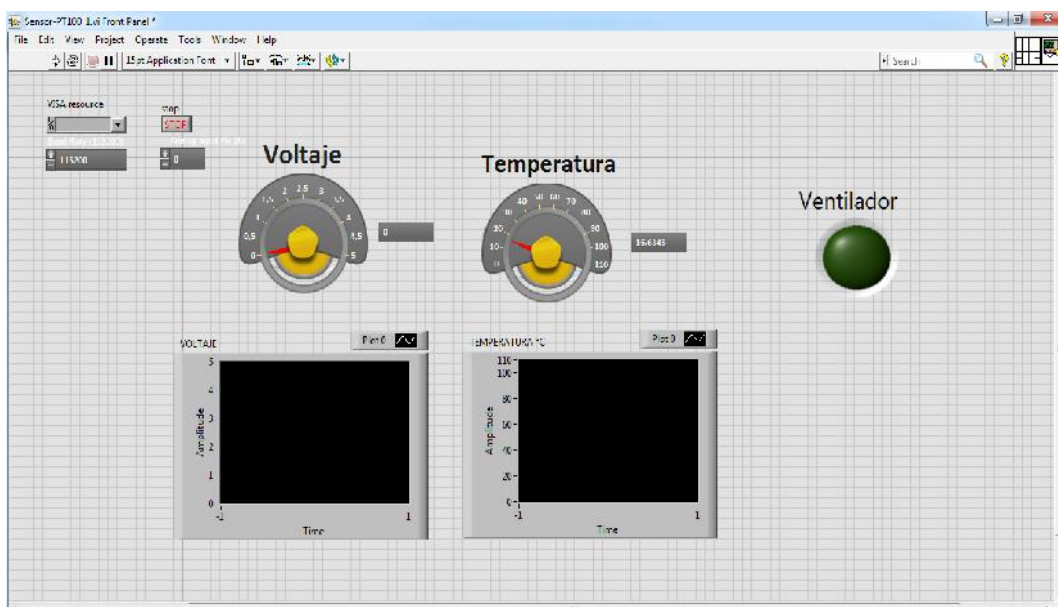


Figura 81: Interfaz de usuario en LabVIEW.

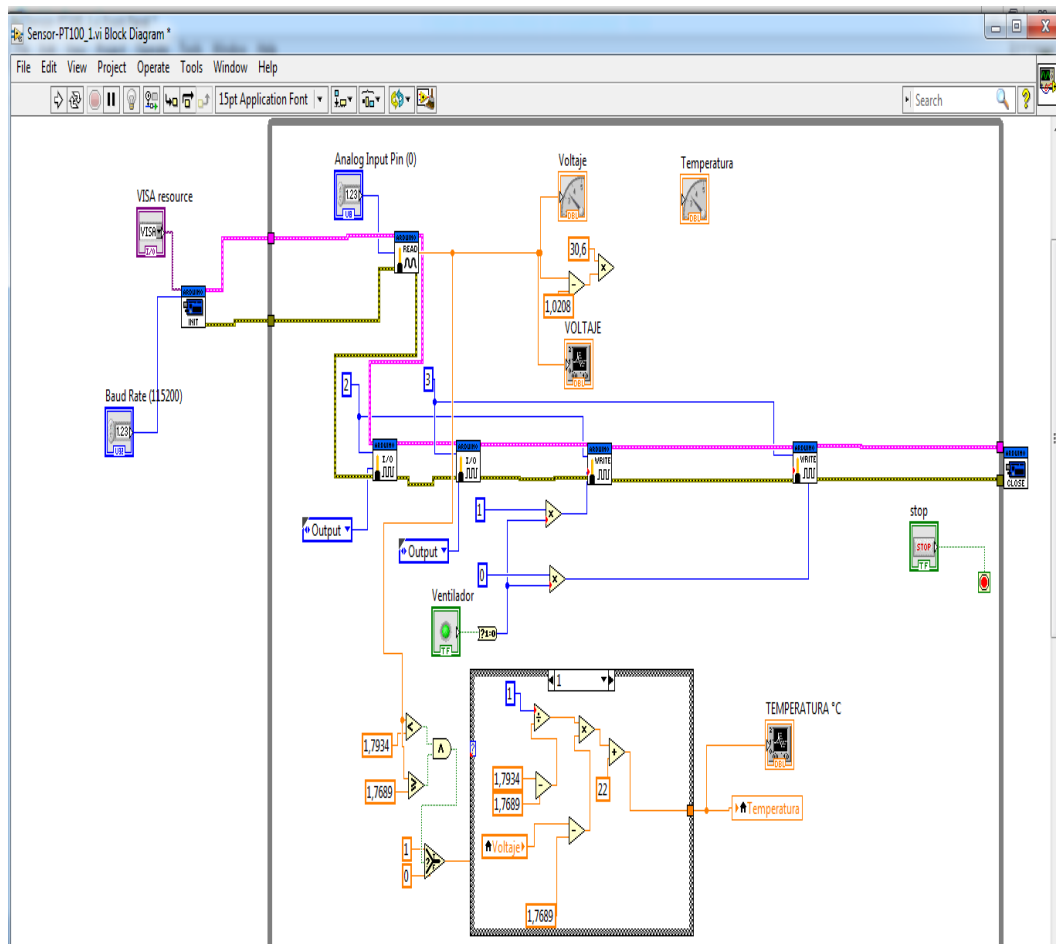


Figura 82: Código en LabVIEW.

CONCLUSIONES Y RECOMENDACIONES

CAPÍTULO IV

4.1. Conclusiones

- Se desarrolló satisfactoriamente el módulo de proceso de un sistema de control de temperatura.
- Se seleccionó correctamente el sensor de temperatura y la tarjeta de adquisición de datos en base a los requerimientos del sistema y a la evaluación de diferentes soluciones que ofrece el mercado local por cada uno.
- Se seleccionó con éxito el circuito acondicionador para el sensor de temperatura (Transmisor de temperatura). Esto permitió tener una lectura precisa del sensor al pasar por las etapas de amplificación y filtrado. Además, el Dimmer diseñado hizo posible el control de potencia del foco mediante una salida analógica de la tarjeta de adquisición de datos
- Se obtuvo correctamente el modelo matemático aproximado del sistema en base a su curva de respuesta. Esto permitió identificar el comportamiento del sistema como uno de primer orden. Además, se validó el modelo obtenido haciendo uso del software Matlab.
- Se consiguió desarrollar una interfaz de usuario intuitiva y funcional en el software LabVIEW que permite observar el cambio de temperatura en el módulo. Además, proporciona las gráficas del comportamiento en el tiempo de las variables de control y de proceso del sistema para un mayor análisis del lazo de control.

4.2. Recomendaciones

- Para obtener una respuesta adecuada del sistema es necesario que durante las pruebas se mantenga bien cerrada la cámara térmica de modulo, puesto que al ingresar aire del exterior significaría una variación en el modelo matemático de la planta.
- Al momento de realizar las medidas de temperatura en el módulo, es recomendable utilizar un termómetro digital para verificar que las lecturas que se muestran en la interfaz gráfica de LabVIEW sean correctas.
- Se recomienda realizar una limpieza periódica del módulo a fin de evitar suciedad, hongos y obstrucciones en la cámara térmica.

BIBLIOGRAFÍA

REFERENCIAS BIBLIOGRÁFICAS

[1] Ingeniería de control moderna, 5ta edición. Autor: Katsuhiko Ogata.

[2] Control Avanzado de Procesos, 2da edición. Autor: José Acedo Sánchez.

[3] Universidad de Almería – UAL, Plantas Piloto.

https://w3.ual.es/portales/ingenieriaquimica/ing_quimica_act.html

[4] Introducción a los Sistemas de Control, Supervisión y de Adquisición de Datos

http://instrumentacionycontrol.net/Descargas/Documentacion/HMI_SCADA/lyCnet_Introduccion_Sistemas_SCADA.pdf

[5] Arduino: Curso Práctico de Formación, 1ra edición. Autor: Oscar Torrente.

[6] Introducción a los Sistemas de Control y Modelo Matemático

<http://dea.unsj.edu.ar/control1b/teoria/unidad1y2.pdf>

[7] System Identification Toolbox - MATLAB - MathWorks

<https://www.mathworks.com/products/sysid.html>

[8] El calor y la temperatura

http://www.quimicaweb.net/grupo_trabajo_ccnn_2/tema3/tema3.htm

[9] Termostato

<https://www.luna.ovh/planeta/es/Termostato>

[10] Termistor, protección para motores

<http://www.ingmecafenix.com/electricidad-industrial/termistor-proteccion-motores/>

[11] Instrumentos de medición

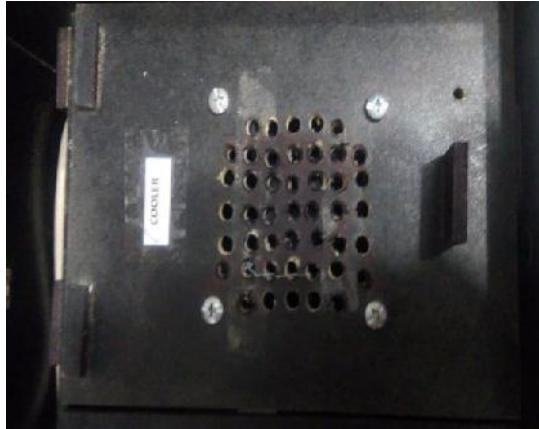
<https://www.scribd.com/document/238327822/INSTRUMENTOS-DE-MEDICION-docx>

ANEXOS

- Diseño final del Módulo de Control de Temperatura



- **Diseño de la Cámara Térmica del Módulo de Control de Temperatura**



- **Diseño de los circuitos del Módulo de Control de Temperatura**

