

**UNIVERSIDAD NACIONAL DEL CALLAO**  
**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**  
**ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA**



**“CONTROL DE TEMPERATURA DE UNA  
INCUBADORA MICROBIOLÓGICA,  
UTILIZANDO UN CONTROLADOR PID”**

**TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE  
INGENIERO ELECTRÓNICO**

**AUTOR(ES): CAMPOS MARTÍNEZ, LUDEBER JOSMELL  
MINES FERNÁNDEZ, CARLO  
TALLEDO BLEST, ALDO**

**Callao, Diciembre – 2015  
PERU**

## INDICE

CARATULA	
PAGINA DE RESPETO	
HOJA DE REFERENCIA DEL JURADO Y APROBACIÓN	
ÍNDICE.....	1
ÍNDICE DE FIGURAS.....	4
RESUMEN.....	5
ABSTRACT.....	6
I. PLANTEAMIENTO DE LA INVESTIGACIÓN.....	7
1.1 Identificación del problema.....	7
1.2 Formulación del problema.....	7
1.3 Objetivos de la investigación.....	8
1.4 Justificación.....	9
1.5 Limitaciones.....	9
II. MARCO TEÓRICO.....	11
2.1 Antecedentes de estudio.....	11
2.2 Fundamentación ontológica.....	13
2.3 Fundamentación metodológica.....	24
2.4 Fundamentación epistemológica.....	25
2.5 Definición de términos básicos.....	25
III. VARIABLES E HIPÓTESIS.....	27
3.1 Variables de la investigación.....	27
3.2 Operacionalización de variables.....	27

3.3 Hipótesis.....	28
IV. METODOLOGÍA.....	29
4.1 Tipo de investigación .....	29
4.2 Diseño de la investigación .....	29
4.2.1 Modelo matemático de la camara.....	29
4.2.2 Cálculo de las capacidades térmicas.....	33
4.2.3 Función de transferencia del convertidor.....	36
4.2.4 Función de transferencia del transductor de temperatura .....	37
4.2.5 Diseño del control de temperatura en dominio de tiempo.....	38
4.2.6 Diseño del control de temperatura discreto .....	42
4.2.7 Sintonizacion con Prueba de la Regla Ziegler Nichols .....	45
4.2.8 modelo del sistema de calefacción de la camara .....	46
4.2.9 Diseño del circuito de acondicionamiento .....	49
4.2.10 Diseño del circuito de Salida .....	49
4.2.11 Implementacion y simulacion.....	53
4.3 Población y Muestra .....	60
4.4 Técnicas e Instrumentos de Recolección de Datos.....	60
4.5 Procedimiento de recolección de datos.....	60
4.6 Procesamiento estadístico y análisis de datos.....	60
V. RESULTADOS .....	61
5.1 Resultados parciales.....	61
5.2 Resultados finales .....	63

VI. DISCUSIÓN DE RESULTADOS .....	64
6.1 Contratación de hipótesis con los resultados.....	65
6.2 Contratación de hipótesis con otros estudios similares.....	65
VII. CONCLUSIONES.....	68
VIII.RECOMENDACIONES .....	70
IX. REFERENCIAS BIBLIOGRÁFICAS .....	71
<b>ANEXOS</b>	
A. Matriz de consistencia .....	73
B. Programa : Librería LCD.h.....	74
C. Programa : Librería Teclado.h .....	74
D. Programa : Librería control ADC.h .....	74
E. Programa : Algoritmo de control PID.h .....	74
F. Programa : Algoritmo principal controlador_temperatura.h .....	74

## ÍNDICE DE FIGURAS

FIGURA Nº 2.1 DIAGRAMA DE BLOQUES DEL SISTEMA .....	24
FIGURA Nº 4.1 DIAGRAMA ESQUEMÁTICO DE LA INCUBADORA.....	30
FIGURA Nº 4.2 DIAGRAMA DE BLOQUES DEL SISTEMA EN EL PLANO S...	39
FIGURA Nº 4.3 DIAGRAMA DE BODE DE LA FUNCIÓN DE TRANSFERENCIA EN LAZO ABIERTO.....	41
FIGURA Nº 4.4 DIAGRAMA DE BLOQUES DEL SISTEMA EN EL PLANO S..	42
FIGURA Nº 4.5 DIAGRAMA DE BLOQUES DEL SISTEMA EN EL PLANO Z.....	45
FIGURA Nº 4.6 REPRESENTACIÓN BINARIA DEL CROMOSOMA.....	49
FIGURA Nº 4.7 DIAGRAMA DE FLUJO DEL PROGRAMA.....	52
FIGURA Nº 4.8 CIRCUITO DE MEDICIÓN DE TEMPERATURA .....	57
FIGURA Nº 4.9 DIAGRAMA DE BLOQUES DEL CONTROLADOR - CONVERTIDOR.....	58
FIGURA Nº 4.10 DIAGRAMA CIRCUITAL DE LA INTERFASE CONTROLADOR - CONVERTIDOR.....	59
FIGURA Nº 5.1 RESPUESTA A UN ESCALÓN DE REFERENCIA DE TEMPERATURA EN EL PLANO S.....	61
FIGURA Nº 5.2 RESPUESTA A UN ESCALÓN DE REFERENCIA DE TEMPERATURA EN EL PLANO Z.....	62
FIGURA Nº 5.3 RESPUESTA A UN ESCALÓN DE REFERENCIA DE TEMPERATURA.....	63

## RESUMEN

La presente Tesis trata sobre: **"CONTROL DE TEMPERATURA DE UNA INCUBADORA MICROBIOLÓGICA, UTILIZANDO UN CONTROLADOR PID"** cuyos objetivos son:

- a) Diseñar y simular el sistema de control con el Microcontrolador Atmel 169 .
- b) Diseñar y simular el modelo de la incubadora por el método discreto y luego el sistema de control sistema PID discreto usando el software Matlab (Simulink).
- c) Diseñar y simular el lazo de regulación de temperatura usando el algoritmo PID en el plano discreto para varias pruebas utilizando el software Proteus

En este proyecto se diseñó un sistema de control de temperatura de una incubadora microbiológica, que mantuvo una temperatura constante, el circuito de control utiliza controlador PID digital, para lograr controlar la temperatura deseada, el circuito de control PID se sintoniza mediante la regla de Nichols el cual tiene la función de mantener la temperatura constante de, por ejemplo 37°C y no variar más de  $\pm 0.2^{\circ}\text{C}$ .

En Electrónica existen diversas maneras de controlar temperatura y corriente de resistencias calefactoras de una incubadora microbiológica. Sin embargo cuando se utiliza un controlador PID digital se puede controlar en forma precisa la temperatura.

## ABSTRACT

The present Thesis, Therefore the project has been developed "temperature control microbiological incubator Condolatory PID" whose objectives are:

- a) Design and simulate the thyristor power converter to control the heating resistors.
- b) Design and simulate the model of the incubator in the Discrete plane and then discrete control system with PID using the Simulink blocks in Matlab software.
- c) Design and simulate the temperature control loop using the PID algorithm in the Software Proteus.

In this project, a temperature control system of a microbiological incubator that maintained a constant temperature control circuit uses the digital PID controller for the desired temperature control, the PID control circuit is tuned by Nichols which has been designed to maintain constant temperature (37 ° C) and not vary more than  $\pm 0.2$  ° C.

In electronics there are various ways to control temperature and current of heating elements of a microbiological incubator. However, when a digital PID controller is used can precisely control the temperature.

## CAPITULO I

### I. PLANTEAMIENTO DE LA INVESTIGACIÓN

#### 1.1 Identificación del problema

Específicamente, el trabajo de investigación se centra en el Control de Temperatura de una Incubadora Microbiológica, **utilizando un controlador PID basado en un procesador de la Familia Atmel.**

De acuerdo con datos experimentales se encuentra que con un control todo o nada al cortar exactamente a los 37°C, con una temperatura ambiente de 20°C la incubadora presenta una desviación de 1°C aproximadamente, con este algoritmo de control se mantenía oscilando entre más o menos 1°C es decir entre 35 y 38°C en los primeros periodos. Para un control preciso se debe mantener la temperatura constante en 37°C, asimismo, no variar más de  $\pm 0.2^\circ\text{C}$ .

El intervalo de temperatura a controlar fluctúa normalmente entre 35°C y 38°C siendo 37°C la temperatura a la que se ajustan la mayoría de incubadoras microbiológicas.

#### 1.2 Formulación del problema

En diversas aplicaciones biomédicas es importante necesario controlar de manera precisa la temperatura de una incubadora microbiológica. Por tanto, se propone como variables: el sistema de control de temperatura para la incubadora y como siguiente variable, controlar en forma precisa la temperatura de la incubadora microbiológica. Además se le incluyen

lazos de regulación de temperatura para no sobrepasar los límites permisibles máximos de temperatura dentro de la incubadora.

Ante las exigencias requeridas en el diseño, de un control de temperatura de una incubadora microbiológica es que se formula, el problema en forma de pregunta:

¿Es posible resolver el problema de diseño del control de temperatura de una incubadora microbiológica, mediante la aplicación de la técnica de control PID?

### **1.3 Objetivos de la investigación**

#### **a) Objetivo general**

- Diseñar y simular un control de temperatura, para una incubadora microbiológica, diseñado en base a un controlador PID.
- Controlar la incubadora microbiológica, mediante un sistema de control de tipo discreto, en base al procesador Atmel 169 con bloques de control de temperatura.

#### **b) Objetivos específicos**

- Diseñar y simular el modelo de la incubadora y luego el sistema de control discreto usando el software Matlab ( Simulink).
- Diseñar y simular la etapa de potencia (salida) para el control de las resistencias de calefacción.

- Diseñar y simular el lazo de regulación de temperatura usando el algoritmo PID implementando en un Microcontrolador Atmel .
- Diseñar y simular el circuito de medida de temperatura, y mostrar la temperatura.

#### **1.4 Justificación**

Se justifica el desarrollo de la presente Tesis, porque el control de temperatura de la incubadora microbiológica, utilizando un controlador PID, tiene la ventaja de poder controlar la temperatura de manera más precisa, sin embargo el algoritmo de sintonización del sistema de control PID debe ser más sofisticado comparado con un método de sintonización estándar, el cual tiene limitaciones en cuanto al controlar la planta. Por tanto se propone sintonizar el control PID mediante la respuesta en lazo abierto denominado Ziegler Nichols .

En el Perú, el grupo de investigación de equipos médicos y sistemas (GIDEMS) el cual pertenece a la Sección de Electricidad y Electrónica de la Pontificia Universidad Católica del Perú, viene realizando desde 1993 investigaciones y tratamiento de nuevos sistemas de control, ventilación, calefacción, así como la construcción de incubadoras.

#### **1.5 Limitaciones**

- **Teórica.** Existen investigaciones similares que se hayan realizado, para que el usuario pueda determinar las bacterias en sangre y calidad del agua mediante una incubadora microbiológica, cuya

temperatura es controlada mediante el algoritmo PID sintonizado con el método de respuesta en lazo abierto denominado Ziegler Nichols.

## **CAPITULO II**

### **II. MARCO TEÓRICO**

#### **2.1 Antecedentes de estudio**

En el 2001, Valverde y Castro presentaron en el Congreso Nacional de Instrumentación el diseño e implementación de un controlador difuso, para la regulación de temperatura en un horno eléctrico resistivo a un valor constante, en éste se describen las características del controlador difuso, diseñado e implementado con el objetivo, de mantener un valor de temperatura constante dentro del ambiente de un horno eléctrico resistivo. Esto se logra controlando la potencia eléctrica, aplicada al banco de resistencias, del que se compone el horno eléctrico. El controlador difuso se implementó sobre una arquitectura de hardware basada en el microcontrolador M68HC11E9 en su modo expandido.

Un grupo de investigación de la Universidad Técnica Nacional de Argentina presentó el artículo denominado "Controlador difuso multivariable del perfil de temperatura de un horno" en la XI Reunión de Trabajo en Procesamiento de la Información y Control, en el cual desarrollaron un controlador, que permite la descripción del problema usando variables lingüísticas y logrando que el horno eléctrico se comporte, en su conjunto, como si un operador estuviera accionando los controles, para compensar las no linealidades del sistema.

Se presentó en la Universidad Tecnológica de Mixteca, la tesis de licenciatura donde se aborda el problema de monitoreo de un horno de

gas. En esta tesis desarrolló un sistema de monitoreo de temperatura, para el horno de gas. Se utilizó una arquitectura maestro-esclavo con la cualidad de que el esclavo es capaz de asumir el control del monitoreo en caso de que el maestro falle. Se tomó como maestro a una PC, que a través de una interfaz utilizando el software LabVIEW ordena a su esclavo, el microcontrolador (Intel 8032), obtener la medición de temperatura interna del horno mediante un termopar tipo K.

En el 2006 Vásquez y Santiago presentaron en la Universidad Tecnológica de Mixteca, el artículo denominado "Diseño e implementación de un sistema de control de temperatura mediante Instrumentación Virtual", en éste se describe el diseño y la implementación de un controlador PID, para regular la temperatura del sistema AmiraLTR701. Se obtuvo la función de transferencia, para el diseño del controlador, se utilizó la herramienta Simulink y el método de Ziegler-Nichol's, obteniendo las ganancias. La simulación del sistema fue hecha en el software Matlab, para obtener los mejores valores de  $k_p$ ,  $K_i$  y  $K_d$ . Finalmente se usó el software LabVIEW para controlar la planta.

Hanamane y Mudholkar presentaron el artículo denominado "Implementation of fuzzy temperature control using microprocessor", en el que se describe la implementación de un controlador difuso de temperatura, utilizando un microprocesador. Se utilizó un termistor como sensor de temperatura formando el componente resistivo para el circuito del 555, operado en modo astable. La temperatura es leída como función

de la frecuencia y comparada con la referencia de temperatura. El error es mandado al trigger del proceso de inferencia difusa, a través de una Interfaz de Puerto Programable PPI 8255. Para controlar la energía eléctrica del horno, se utiliza la modulación por ancho de pulso (PWM).

En 2009 se realizó en la Universidad Tecnológica de Mixteca, la tesis de licenciatura, en esta tesis se diseñó y construyó el hardware para monitorear y controlar la temperatura utilizando dos sistemas de control, tipo on-off y por ciclos, también se desarrolló un programa con base en una computadora utilizando el software LabVIEW. En este programa se muestra el historial de las variaciones de temperatura y se configura el encendido o apagado, vía puerto serie hacia el módulo de disparo. La digitalización de la temperatura se hace con una tarjeta de adquisición de datos NI USB 6008.

## **2.2 Fundamentación ontológica**

Para cultivar colonias de bacterias existen muchos productos. Como medios de crecimiento se han desarrollado y se venden "reactores" microbiológicos provistos de calefactores eléctricos, agitadores y controladores automáticos de temperatura.

La bacteriemia y la fungemia se producen cuando la multiplicación y llegada de microorganismos a la sangre supera la capacidad del sistema retículo endotelial para eliminarlos. La invasión del torrente sanguíneo se produce desde un foco de infección extravascular, a través de los

capilares o de las vías linfáticas, o desde un foco intravascular como en la endocarditis.

La bacteriemia y la fungemia tienen implicaciones pronosticadas importantes ya que se asocian a una elevada mortalidad que oscila entre el 20 y 40%. El descenso de ésta se encuentra claramente relacionado con la administración de un tratamiento antibiótico adecuado lo más precozmente posible. Por ello, el aislamiento de un microorganismo en los hemocultivos es trascendente porque establece el diagnóstico etiológico de la bacteriemia y permite conocer la sensibilidad del microorganismo causal a los antimicrobianos.

La aplicación de una buena metodología para la extracción y procesamiento de los hemocultivos debe perseguir:

1. El aislamiento de todos los microorganismos productores de bacteriemia. Para ello deben tenerse en cuenta sus diferentes requerimientos nutricionales, temperatura y atmósfera de óptimo crecimiento y período de incubación.
2. La detección precoz de la presencia de bacterias en sangre de forma que las modificaciones del tratamiento que ello conlleva se puedan realizar con la mayor rapidez.
3. La diferenciación, en la medida de lo posible, de los casos de verdadera bacteriemia de aquellos en los que la positividad se debe a un inadecuado procedimiento de extracción y procesamiento de la muestra.

4. La identificación precisa de los agentes causales de la bacteriemia y su sensibilidad a los antimicrobianos.

La incidencia de la bacteriemia depende de la población estudiada y oscila entre 5 y 30 casos por cada 1000 pacientes hospitalizados. Se presenta a cualquier edad y están especialmente predispuestos a padecerla los pacientes con graves enfermedades de base y los sometidos a maniobras que causan alteraciones de los mecanismos generales y locales de defensa frente a la infección.

La bacteriemia puede ser una complicación de infecciones localizadas como infección urinaria, infecciones respiratorias, etc., puede deberse a la irrupción directa de los microorganismos en el torrente cardiovascular como en el caso de pacientes adictos a drogas, portadores de catéteres intravasculares y sometidos a cirugía cardíaca, o bien la puerta de entrada puede ser inoperante.

Las manifestaciones clínicas de las bacteriemias son muy variadas y van desde un cuadro febril, sin ninguna otra manifestación clínica, hasta el shock séptico con fracaso multiorgánico.

La mayoría de los microorganismos son capaces de invadir el torrente sanguíneo. La distribución de los agentes causales de la bacteriemia y la fungemia ha variado en los últimos años y actualmente los microorganismos Gram positivos, especialmente estafilococos y enterococos, igualan o superan en frecuencia a los bacilos Gram negativos.

Los motivos de este cambio no se han establecido con exactitud, pero se atribuyen a la utilización de antibióticos de amplio espectro, al uso generalizado de catéteres intravasculares, a la mayor supervivencia de enfermos graves, a la aparición de resistencia a los antimicrobianos en microorganismos como *Srrophylococcus aureus* o *Enterococcus* que pueden causar brotes de infección hospitalaria, a la mayor agresividad de la cirugía y a la utilización de maniobras terapéuticas invasivas. Por otra parte, el aumento de pacientes con neoplasias o infección por VIH, gravemente inmunodeprimidos, ha provocado la aparición cada vez más frecuente de bacteriemias causadas por agentes que en el pasado eran causa muy rara de infección (*Corynebacterium*,...)

El diagnóstico definitivo de la bacteriemia se establece cuando se aísla el microorganismo causal en la sangre, mediante el cultivo de ésta.

Además, la necesidad de realizar hemocultivos deriva de los siguientes hechos:

1. El aislamiento de una bacteria en los hemocultivos aporta una valiosa información a la hora de elegir el tratamiento antimicrobiano adecuado.
2. El hemocultivo no es una técnica costosa y su obtención no conlleva ningún riesgo para el paciente.
3. Las características clínicas de la bacteriemia y las situaciones en que puede presentarse son tan variadas que no permiten establecer un diagnóstico clínico con suficiente certeza.

Todo ello obliga a mantener un alto índice de sospecha y extraer hemocultivos a muchos de los pacientes que ingresan en el hospital o que consultan por sospecha de infección. Existen dos datos que ilustran este hecho, por una parte, en el 14-25% de los pacientes hospitalizados se sospecha bacteriemia y se les extraen hemocultivos y por otra, en aproximadamente el 14% de los casos en los que se obtienen hemocultivos se establece el diagnóstico de bacteriemia.

Los métodos para el estudio microbiológico de los hemocultivos pueden ser cuantitativos, semicuantitativos y cualitativos.

En los métodos semicuantitativos se sigue un procedimiento de lisis centrifugación. Para ello, se realiza la extracción de sangre en un tubo "sistema vacutainer" con saponinas que rompen las células sanguíneas. La sangre se centrifuga y el sedimento se siembra directamente en las placas de cultivo. Este método permite una valoración semicuantitativa por recuento de las colonias aisladas, consigue a menudo una identificación más rápida de los microorganismos causales y facilita la búsqueda de bacterias como micobacterias o *Legionella* al permitir seleccionar los medios de cultivo más adecuados para aislar dichos microorganismos. Tiene el inconveniente de que la manipulación de la muestra aumenta la posibilidad de contaminación y requiere un procesamiento individualizado con la consiguiente necesidad de personal de laboratorio.

Los métodos cualitativos se realizan cultivando la sangre en frascos con medio líquido o bifásico y son los utilizados de forma rutinaria en la mayoría de los Laboratorios de Microbiología. En éstos el medio de cultivo debe examinarse diariamente para detectar lo antes posible los signos de crecimiento bacteriano. En el procedimiento convencional este examen diario se realiza por inspección visual de los frascos y es el que describiremos en primer lugar en este documento. Existen otros procedimientos de lectura basados en la producción de CO<sub>2</sub> que facilitan aparatos automáticos. Estos se han ido incorporando a los Laboratorios de Microbiología porque detectan con rapidez el crecimiento bacteriano y facilitan el trabajo del personal de laboratorio.

La mayoría de los microorganismos que causan bacteriemia se detectan en los primeros 2 ó 3 días de incubación. Por ello, se recomienda incubar los hemocultivos durante un período de 5 a 7 días. La temperatura óptima de incubación es de 35 a 37°C.

Existen algunos microorganismos que necesitan un período de incubación más largo, como *Bruccella* y bacterias de difícil crecimiento que causan endocarditis, como *Haemophilus*, *Cardiobacterium*, *Eikenella*, *Kingella*... También los hongos necesitan, cuando se sospecha la existencia de estos patógenos, la rutina más extendida es la de prolongar la incubación de los hemocultivos hasta 4 semanas.

Los pacientes con infección por VIH pueden desarrollar bacteriemia por microorganismos con requerimientos especiales para su aislamiento, en

una región de menor temperatura. La transferencia de energía térmica se presenta siempre desde la región de mayor temperatura según la siguiente ecuación:

$$q = -kA \frac{\partial T}{\partial x}$$

Donde:

q = cantidad de calor transferido por conducción

k= conductividad térmica del material

A= Área de transferencia de calor

$\frac{\partial T}{\partial x}$  = gradiente de temperatura en la dirección de flujo de calor.

El signo menos (-) se introduce en la ecuación para que se cumpla la segunda ley de la termodinámica.

Para realizar un determinado trabajo de ciertas características el sistema debe estar realimentado. La función principal de un sistema realimentado es obligar a las magnitudes realimentadas del sistema a conservar los valores tan próximos como sea posible a los de referencia. Estos valores de referencia son a su vez función de las magnitudes de entrada del sistema y las perturbaciones.

Un sistema de control de temperatura, se basa en la utilización de termostatos, bimetalicos o de expansión de fluidos. Para detectar la temperatura, se usan termocuplas, o termistores, siendo los más precisos

las termoresistencias, esto es, resistencias que varían con la temperatura que pueden sensar variaciones de  $0.1^{\circ}\text{C}$

Los controladores de temperatura PID no son termostatos, recordemos que los termostatos encienden la calefacción cuando la temperatura desciende de un cierto valor, y la apagan cuando la temperatura supera otro valor, en síntesis la temperatura se mantiene entre un máximo y un mínimo. En los controladores de temperatura PID, se intenta entregar a la incubadora solo la potencia que hace falta para mantener la temperatura que uno desea.

En la actualidad se producen dispositivos totalmente electrónicos llamados Tiristores, o comúnmente relé de estado sólido. Estos dispositivos, debidamente protegidos, pueden manejar corrientes elevadas entre el encendido y el apagado de la calefacción, y su duración en el tiempo, es muy superior a los termostatos con relé electromecánicos.

Los controles automáticos o adaptativos fueron introducidos para optimizar el control de las plantas industriales. Un ejemplo de esto son los algoritmos genéticos aplicados a los controladores PID que pueden trabajar en un proceso de optimización.

**El desarrollo de controladores PID sintonizado con algoritmos genéticos** es un método no muy usado debido al desconocimiento del uso de las características los algoritmos genéticos en este tipo de aplicaciones.

En la implementación de un controlador PID se debe determinar cuidadosamente tres parámetros como son  $K_p$  ganancia proporcional,  $K_i$  ganancia integral,  $K_d$  ganancia derivativa.

**Acción proporcional:**

La respuesta proporcional es la base de los tres modos de control, si los otros dos, control integral y control derivativo están presentes, éstos son sumados a la respuesta proporcional. "Proporcional" significa que el cambio presente en la salida del controlador es algún múltiplo del porcentaje del cambio en la medición.

Este múltiplo es llamado "ganancia" del controlador. Para algunos controladores, la acción proporcional es ajustada por medio de tal ajuste de ganancia, mientras que para otros se usa una "banda proporcional". Ambos tienen los mismos propósitos y efectos.

**Acción integral:**

La acción integral da una respuesta proporcional a la integral del error. Esta acción elimina el error en régimen estacionario, provocado por el modo proporcional. Por contra, se obtiene un mayor tiempo de establecimiento, una respuesta más lenta y el periodo de oscilación es mayor que en el caso de la acción proporcional.

**Acción derivativa:**

La acción derivativa da una respuesta proporcional a la derivada del error (velocidad de cambio del error). Añadiendo esta acción de control a las anteriores, se disminuye el exceso de sobreoscilaciones.

Existen diversos métodos de ajuste para controladores PID, pero ninguno de ellos nos garantiza que siempre encuentre un PID, que haga estable el sistema. Por lo que el más usado sigue siendo el método de prueba y error, probando parámetros del PID y en función de la salida obtenida, variando estos parámetros.

Se han desarrollado muchas aplicaciones para determinar los parámetros del controlador PID para sistemas SISO. Entre las aproximaciones más conocidas tenemos el método de Ziegler- Nichols, el método de Cohen-Coon.

Para enriquecer las capacidades de las técnicas de sintonización del controlador PID tradicional, se han desarrollado recientemente muchos nuevos métodos como: los algoritmos genéticos, la lógica difusa para sintonizar los controladores PID.

Wang u Kwok han presentado un proceso de neutralización del PH regulado por un controlador PID con sus parámetros optimizados usando un Algoritmo Genético.

Se modeló la planta en este caso la incubadora usando la transformada de La Place, prosiguiendo con la conversión al plano Z y analizando su respuesta en el tiempo, luego se diseñó los controladores PID discreto en el plano Z empleando alguno de los métodos para transformar la función de transferencia del controlador PID del plano S al plano Z, en donde se realizó el análisis de estabilidad del sistema usando el lugar geométrico de las raíces, compensando las inestabilidades del sistema.

- **Estrategias.-** Son acciones propias del investigador para obtener lo que necesita para su investigación.
- **Epistemología.-** Es una disciplina que estudia cómo se genera y se valida el conocimiento de las ciencias.
- **Simulación.-** Es una técnica numérica para conducir experimentos en una computadora digital
- **Modelamiento.-** Reproducción ideal y concreta de un objeto o de un fenómeno con fines de estudio y experimentación.
- **Algoritmo.-** Es un conjunto ordenado de operaciones que permite hallar la solución de un problema.
- **Ontología.-** Es una parte o rama de la filosofía que estudia la naturaleza del ser, la existencia y la realidad, tratando de determinar las categorías fundamentales y las relaciones del "ser en cuanto ser".Las ontologías se aplican también en Inteligencia Artificial para asimilar y codificar el conocimiento, definiendo las relaciones existentes entre los conceptos de un determinado dominio (un área del conocimiento).

## CAPITULO III

### III. VARIABLES E HIPÓTESIS

#### 3.1 Variables de la investigación

Relacionando las variables relevantes que intervienen en el presente problema objeto de estudio, que conllevan a la explicación, demostración y comprobación de la formulada hipótesis, se han identificado las siguientes variables.

- **Variable z** = Controlador PID
- **Variable W** = Controlar en forma precisa la variación de temperatura de la incubadora

#### 3.2 Operacionalización de variables

**Variable Z** = Controlador PID

**Indicadores:** Diseño de un Algoritmo PID

- Estudio del modelo de la incubadora discreta Z<sub>1</sub>
- Obtención de los parámetros del controlador PID Z<sub>2</sub>
- Diseño y simulación del control PID de temperatura Z<sub>3</sub>

**Indicadores:** Diseño del algoritmo con sistema en lazo abierto

- Obtención de los parámetros del algoritmo Z<sub>4</sub>
- Diseño y simulación del algoritmo PID Z<sub>5</sub>

**Variable W** = Controlar en forma precisa la variación de temperatura de la Incubadora.

**Indicadores:** Variación de temperatura de la Incubadora

- mayor a  $0.2\text{ }^{\circ}\text{C}$   $W_1$
- igual a  $0.2\text{ }^{\circ}\text{C}$   $W_2$
- menor a  $0.2\text{ }^{\circ}\text{C}$   $W_3$

### **3.3 Hipótesis**

#### **3.3.1 Hipótesis General**

Un controlador PID discreto basado en un procesador, controla en forma precisa la temperatura.

#### **3.3.2 Hipótesis específica**

Un controlador PID basado en un procesador de la familia Atmel, controla en forma precisa la temperatura, de una incubadora microbiológica.

## **CAPITULO IV**

### **IV. METODOLOGÍA**

#### **4.1 Tipo de investigación**

Ya que se ha planteado la hipótesis a probar y se ha identificado las variables, que actúan sobre el sistema; el proceso de la investigación es de tipo: Aplicada-Transversal, además es Experimental-Tecnológica, este proyecto se realizó en las instalaciones del laboratorio FIEE. UNAC teniendo como referencia el sistema de control de la cámara microbiológica basado en el procesador de la familia ATMEL 169

#### **4.2 Diseño de la investigación**

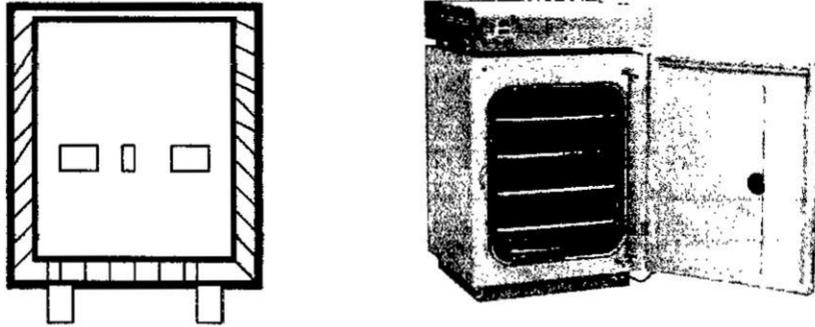
##### **4.2.1 Modelo matemático de la incubadora microbiológica**

Se tiene una incubadora con resistencias eléctricas de calentamiento indirecto, se calienta por efectos de la radiación, convección o una combinación de ambas. La energía calorífica requerida para el calentamiento procede de resistencias eléctricas, dispuestas en la base de la carcasa y en las paredes para asegurar uniformidad y estabilidad de la temperatura dentro de la incubadora.

El problema a resolver es establecer la dinámica de la incubadora. La estrategia del control usada fue el algoritmo PID.

Las dinámicas se establecieron a partir del modelamiento de los respectivos sistemas y luego fueron simuladas.

**FIGURA N° 4.1 DIAGRAMA ESQUEMÁTICO DE LA INCUBADORA**



Las dimensiones de la incubadora microbiológica son:

Dimensiones externas:

Ancho = 28 cm,

Profundidad = 28 cm y

Altura 33 cm.

Dimensiones internas:

Ancho = 23.0 cm,

Profundidad = 22.0 cm y

Altura 20.0 cm.

Capacidad en Volumen = 18 litros

Voltaje nominal de funcionamiento: 220V

Potencia Nominal: 100W

La carcasa y el marco de la puerta, son de acero dulce con recubrimiento epoxi. Posee una ventanilla transparente de plexiglás (con una perfecta

visibilidad) o con ventanilla pintada, opaca de plexiglás para la protección de muestras fotosensibles.

Se determinó el modelo matemático de la incubadora, para obtener las características de la función de transferencia del calor de la incubadora, presentando las relaciones entre la temperatura y la corriente necesaria para producir el flujo de calor que incrementa la temperatura de la incubadora.

La relación entre la transferencia de calor y la temperatura es:

$$Q_1 - (Q_2 - Q_3) = \frac{C_2 dT_3}{dt} \quad (4.2.1.1)$$

Donde:

$T_1$  = Temperatura de las muestras

$T_2$  = Temperatura ambiente

$T_3$  = Temperatura de las resistencias

$T_4$  = Temperatura de la pared interior

$Q_1$  = Calor generado por las resistencias

$Q_2$  = Flujo de calor generado desde las paredes hacia la carga

$Q_3$  = Perdidas de calor a través de las paredes

$C_1$  = Capacidad térmica de las muestras

$C_2$  = Capacidad térmica de los elementos de calentamiento

$C_3$  = Capacidad térmica de las paredes

$R_{t1}$  = Resistencia térmica entre las paredes y la carga

$R_{t2}$  = Resistencia térmica entre las paredes y los exteriores

$\tau = R_{t1} * C_1 =$  Constante de tiempo en que la incubadora alcanza el 63.2% del valor de su estado permanente.

Considerando la conductividad térmica de la carga

$$Q_2 = \frac{T_4 - T_1}{R_{t1}} = \frac{C_1 dT_1}{dt} \quad (4.2.1.2)$$

El flujo de calor a través de las paredes al ambiente es:

$$Q_3 = \frac{T_3 - T_2}{R_{t2}} \quad (4.2.1.3)$$

Si la conductividad térmica de la pared es elevada:

$$Q_3 = \frac{C_3 dT_4}{dt} \quad (4.2.1.4)$$

Sustituyendo (4) en (2) se obtiene:

$$\frac{U^2}{R} - \frac{C_1 dT_1}{dt} - \frac{C_3 dT_4}{dt} = \frac{C_2 dT_3}{dt} \quad (4.2.1.5)$$

Suponiendo una gran aproximación entre  $T_3$  y  $T_4$ :

$$\frac{U^2}{R} - \frac{C_1 dT_1}{dt} = (C_2 + C_3) \frac{dT_4}{dt} \quad (4.2.1.6)$$

De la ecuación (2) se deduce:

$$T_4 = C_1 R_{t1} \frac{dT_1}{dt} + T_1 \quad (4.2.1.7)$$

Sustituyendo (7) en (6) se obtiene:

$$\frac{U^2}{R} - \frac{C_1 dT_1}{dt} = (C_2 + C_3) \frac{d}{dt} \left[ C_1 R_{t1} \frac{dT_1}{dt} + T_1 \right] \quad (4.2.1.8)$$

$C_1 R_{t1} = \tau$ ; es una constante de tiempo. Reemplazando y simplificando

$$\tau \frac{d^2(T_1)}{dt^2} + \frac{(C_1 + C_2 + C_3)}{(C_2 + C_3)} \frac{dT_1}{dt} = \frac{U^2}{R} \frac{1}{(C_2 + C_3)} \quad (4.2.1.9)$$

Comprende la ecuación diferencial que muestra el comportamiento de la temperatura interna, en el tiempo, cuando se le aplica una potencia interna.

#### 4.2.2 Cálculo de las capacidades térmicas

La capacidad térmica se calcula a partir de:

$$C_1 = m_1 * C_{e1} \quad (4.2.2.1)$$

Donde:

$m_1$  = Masa de las muestras

$C_{e1}$  = Calor Masa de las muestras

Considerando  $m_1 = 2.2$  Kg

$$C_{e1} = 0.233 \frac{Wh}{Kg * ^\circ C}$$

Por lo tanto:

$$C_1 = 0.465 \frac{Wh}{^\circ C}$$

La capacidad térmica de la fuente de calor (resistencias), es la capacitancia del volumen de aire caliente, que está en contacto con las muestras.

$$\text{Volumen de aire} = 18 \text{ litros} = 0.018 \text{ m}^3$$

$$\text{Densidad del aire} = 1.18 \frac{\text{Kg}}{\text{m}^3}$$

Masa de aire = densidad x volumen

$$\text{Masa de aire} = 0.021 \text{ Kg}$$

$$\text{Capacidad específica del aire} = 0.28 \frac{\text{Wh}}{\text{Kg} \cdot ^\circ\text{C}}$$

Por lo tanto:

$$C_2 = 5.95 \times 10^{-3} \frac{\text{Wh}}{^\circ\text{C}}$$

La capacidad térmica de las paredes de la incubadora, es la capacidad de las láminas de acero de las paredes internas.

$$\text{Masa de las láminas de acero} = 5 \text{ Kg}$$

$$\text{Capacidad específica del acero} = 0.1255 \frac{\text{Wh}}{\text{Kg} \cdot ^\circ\text{C}}$$

Por lo tanto:

$$C_3 = 0.6275 \frac{\text{Wh}}{^\circ\text{C}}$$

El voltaje de la fuente de alimentación es: 220V

Reemplazando los valores de las capacitancias y voltajes se obtuvo:

$$\tau \frac{d^2(T_1)}{dt^2} + 1.7 \frac{dT_1}{dt} = 350 \frac{dq}{dt} = 350 i(t) \quad (4.2.2.2)$$

Cálculo de la constante de tiempo de la incubadora

El tiempo que demora en calentarse la incubadora es:

$$\Delta t = \frac{\Delta T_c \cdot m_1 \cdot C_{e1}}{A_r \cdot F'_c \cdot \sigma [(Th + 273)^4 - (Tc + 273)^4]} \quad (4.2.2.3)$$

Donde:

$\Delta T_c$  = Incremento de la temperatura de las muestras

$m_1$  = Masa de las muestras

$C_{e1}$  = Capacidad especifica de las muestras

$A_r$  = Superficie de las muestras para la radiación

$F'_c = \varepsilon = 0.4$  Emisividad de las muestras

$Th$  = Temperatura de la incubadora en el instante t

$Tc$  = Temperatura de las muestras en el instante t

$\sigma = 5.67 \times 10^{-8} \frac{W}{m^2 \cdot ^\circ C}$  ; Constante de Stefan-Boltzmann

Reemplazando datos:

$$\Delta t = \frac{12^\circ C \cdot 2Kg \cdot 0.232 \frac{Wh}{Kg^\circ C}}{0.2m^2 \cdot 0.4 \cdot 5.67 \times 10^{-8} \frac{W}{m^2 \cdot ^\circ C} \cdot (310^4 - 295^4)} \quad (4.2.2.4)$$

$$\Delta t = 0.7386 \text{ horas} = 44.3 \text{ min}$$

Por lo tanto se considera la constante de tiempo  $\tau = 44 \text{ minutos}$

Reemplazando datos

$$44 \frac{d^2 T_1}{dt} + 1.74 \frac{dT_1}{dt} = 350 i(t) \quad (4.2.2.5)$$

Aplicando la transformada de La Place se obtuvo la función de transferencia de la planta.

$$44s^2T(s) + 1.74sT(s) = 350i(s)$$

$$\frac{T(s)}{i(s)} = \frac{350}{s(44s + 1.74)} \quad (4.2.2.6)$$

$$\frac{T(s)}{i(s)} = \frac{201.15}{s(25.28s + 1)} \quad (4.2.2.7)$$

#### 4.2.3 Función de transferencia del circuito de salida

El voltaje de control  $V_c$ , varía entre 0 y +5 Voltios. La potencia de la incubadora es 100W y su voltaje de alimentación es de 220V. Por tanto el valor de la resistencia calefactora es:

$$R = \frac{V^2}{P} = \frac{220^2}{100} = 484 \Omega \quad (4.2.3.1)$$

El retardo de tiempo del convertidor es aproximadamente una constante de tiempo de primer orden  $\tau_c$  la cual es la mitad del intervalo entre dos pulsos que se utilizaron para el disparo consecutivo, aplicado al Triac.

$$\tau_c = \frac{1}{2} \left( \frac{16.6}{2} \right) = 4mseg = 6.67 \times 10^{-5}min$$

La función de transferencia es:

$$\frac{i(s)}{V_c(s)} = \frac{i(s)}{V(s)} \frac{V(s)}{V_c(s)} = \frac{1}{R} \frac{V}{(6.67 \times 10^{-5}s + 1)V_c}$$

$$\frac{i(s)}{Vc(s)} = \frac{0.09}{6.67 \times 10^{-5}s + 1} \quad (4.2.3.2)$$

#### **4.2.4 Función de transferencia del sensor de temperatura**

La temperatura es el parámetro más común de medir aunque muchos aspectos de ella son los menos conocidos. Es un parámetro difícil de medir con la precisión que uno espera.

Para obtener precisiones mejores que 0.2°C es necesario gran cuidado. Los errores ocurren debido a la presencia de gradientes de temperatura, corrientes, no linealidades del sensor, un contacto térmico pobre, flujos de calibración, energía radiante y el autocalentamiento del sensor.

Generalmente la precisión de todos los tipos de sensores puede mejorarse mucho por la calibración individual.

En un laboratorio con los equipamientos estándares apropiados es posible medir temperaturas de 0.001°C o incluso mejores. Esto se hace típicamente por interpolación entre dos estándares, usando un sensor de platino de calidad y/o un termopar de tipo S.

Los errores en la calibración de los sensores pueden ser debidos a la compensación, a la escala y a los errores de linealidad. Además, cada uno de estos errores, puede ir a la deriva con el tiempo y ciclo de temperatura. Los sensores RTD de platino, están considerados como los más precisos y estables de los sensores estándar. Los termopares

basados en platino pueden ser tan estables como los sensores RTD de platino y cubrir un gran rango de temperatura.

La capacidad de intercambio del sensor, es a menudo el factor decisivo. Esto se refiere al error de lectura, de la temperatura máxima, que probablemente ocurre al sustituir un sensor, por otro del mismo tipo sin volver a calibrar el sistema.

Elegir una referencia práctica de calibración puede ser importante. Para fines profesionales, un sensor RTD de platino de alta calidad, es el mejor, con un indicador apropiado.

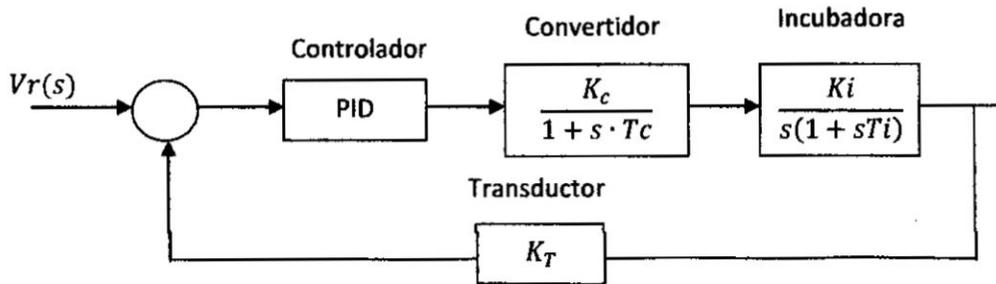
Para este trabajo se seleccionó, una sonda Pt-100, la cual tiene buena linealidad, con respecto a la temperatura. Según el diseño del circuito de acondicionamiento del transductor de temperatura, la función de transferencia del transductor es:

$$\frac{V_T(s)}{T(s)} = 0.1 \quad (4.2.4.1)$$

#### **4.2.5 Diseño del control de temperatura en el dominio del tiempo**

El diseño del controlador de temperatura puede explicarse mediante la figura 4.2 donde se considera la función de transferencia de la incubadora, convertidor y transductor.

FIGURA N° 4.2 DIAGRAMA DE BLOQUES DEL SISTEMA EN EL PLANO S



La función de transferencia en lazo abierto es:

$$F_{OC,T}(s) = G_{C,T} \left( \frac{s \cdot T_{C,T} + 1}{s \cdot T_{C,T}} \right) \left( \frac{Kc}{1 + s \cdot Tc} \right) \left( \frac{Ki}{s(sTi + 1)} \right) (K_T) \quad (4.2.5.1)$$

Uno de los métodos generalmente usados para sintonizar los controladores de PID es el método del Óptimo Simétrico [2]. Este método, consiste en aproximar la función de transferencia, en lazo cerrado, por una de tercer orden, con dos polos complejos conjugados (sistema subamortiguado) con ángulo de  $60^\circ$  y otro más alejado, con lo que se logra que la función de transferencia resultante, sea óptima.

La idea principal consiste en escoger la frecuencia de corte como la media geométrica de las dos constantes de tiempo, de modo que se obtenga el máximo margen de fase. El cual resultará en el factor de amortiguamiento óptimo del control de velocidad, El nombre del método viene del diagrama de Bode que muestra simetría con respecto a la frecuencia de corte, según se muestra en la Figura 4.3. Luego el tiempo integral es:

$$T_{c,w} = a^2 * T_{ie} \quad (4.2.5.2)$$

Donde

$$T_{ie} = T_i + T_c$$

La frecuencia de corte es:

$$\omega_{corte} = \frac{1}{\sqrt{T_{c,T} \cdot T_i}} = \frac{1}{aT_{ie}} \quad (4.2.5.3)$$

De la condición de la frecuencia de corte:

$$|F(j\omega_{corte})| = 1 \quad (4.2.5.4)$$

Con ayuda de las ecuaciones (5.4.1), (5.4.2), (5.4.3) y (5.4.4) se obtiene la ganancia proporcional:

$$G_{c,T} = \frac{1}{aK_i K_c K_T} \left( \frac{1}{T_i} \right)$$

Los polos de la función de transferencia resultante son:

$$s_1 = -\omega_{corte} \dots \dots \dots s_{2,3} = \omega_{corte} \left[ -\frac{a-1}{2} \pm j \sqrt{1 - \left( \frac{a-1}{2} \right)^2} \right]$$

Donde:

$$\xi = \frac{a-1}{2}$$

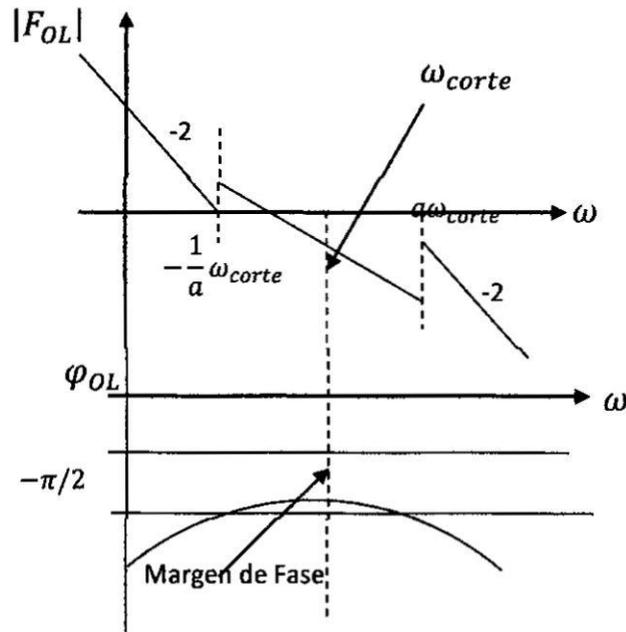
Es el factor de amortiguamiento de la parte oscilatoria de la respuesta.

Asumiendo una respuesta ligeramente sobreamortiguada  $\xi = \frac{1}{\sqrt{2}}$  se

tiene:

$$a = 1 + \sqrt{2} \approx 2.41$$

**FIGURA N° 4.3 DIAGRAMA DE BODE DE LA FUNCIÓN DE TRANSFERENCIA EN LAZO ABIERTO**



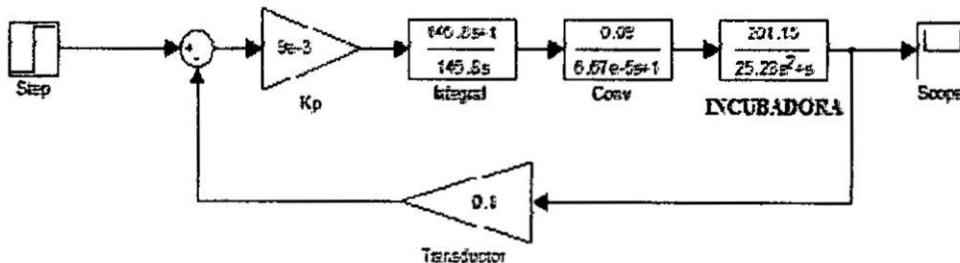
Reemplazando valores:

$$T_{C,T} = a^2(T_i + T_c) = 2.41^2 \times 25.28 = 146.8$$

$$G_{C,T} = \frac{1}{(2.41)(0.09)(201.15)(25.28)0.1} = 9 \times 10^{-3}$$

La figura 4.4, muestra el diagrama de bloques del sistema con los valores de las ganancias y constantes de tiempo hallados.

**FIGURA N° 4.4 DIAGRAMA DE BLOQUES DEL SISTEMA EN EL PLANO S**



#### 4.2.6 Diseño del control de temperatura en tiempo discreto

El control de temperatura en el plano Z se obtiene a partir del sistema de control continuo mostrado en la figura 4.2, hallando el equivalente discreto de cada una de las funciones de transferencia del sistema usando un bloqueador del orden cero para efectuar la conversión del convertidor, horno y transductor de temperatura. Mientras que para el controlador PI se usa la transformación rectangular hacia atrás.

La función de transferencia del bloqueador de orden cero que se usa para la reconstrucción de la señal es:

$$G_{ho} = \frac{1 - e^{-Ts}}{s} \quad (4.2.6.1)$$

Para tener la salida muestreada, de los elementos, cuando hay una entrada muestreada, se necesita la función de transformada 'Z' de la respuesta a los elementos, a una entrada impulso unitario, es igual a la función de transferencia.

De este modo:

$$G_p(z) = z \left[ \frac{1 - e^{-Ts}}{s} \cdot G_p(s) \right] \quad (4.2.6.2)$$

Considerando un tiempo de muestreo menor que la constante de tiempo integral del regulador PI.  $T_s < T_i$

$$T_s = 2 \text{ min}$$

La función de transferencia del conjunto convertidor incubadora es:

$$G_{C,T}(z) = \left\{ \frac{1 - e^{-Ts}}{s} \left( \frac{0.09}{6.67 \times 10^{-5}s + 1} \right) \left( \frac{201.15}{s(25.28s + 1)} \right) \right\}$$

$$G_{C,T}(z) = \frac{1.458z^2 + 1.42z + 2.938 \times 10^{-10}}{z^3 + 1.924z^2 + 0.923z + 3.9 \times 10^{-22}} \quad (4.2.6.3)$$

La expresión de salida de un controlador PID en función del error que es la diferencia que hay entre la señal de entrada y la señal de salida del sistema es dada por:

$$u(t) = K_p \left( e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{d e(t)}{dt} \right) \quad (4.2.6.4)$$

Donde  $K_p$  es la ganancia proporcional,  $T_i$  es el tiempo integral y  $T_d$  es el tiempo derivativo.

Para pequeños periodos de muestreo se puede aproximar el término derivativo a la siguiente expresión:

$$\frac{d e(t)}{dt} = \frac{e(k) - e(k-1)}{T} \quad (4.2.6.5)$$

Existen diferentes formas de integrar una función numéricamente se propone la transformación rectangular hacia atrás.

$$\int e(t) \cong T \sum_0^k e(t)$$

Reemplazando las expresiones anteriores

$$u(k) = K_p \left[ e(k) + \frac{T}{T_i} \sum_0^k e(i) + \frac{T_d}{T} (e(k) - e(k-1)) \right]$$

También se puede escribir como:

$$u(k-1) = K_p \left[ e(k-1) + \frac{T}{T_i} \sum_0^{k-1} e(i) + \frac{T_d}{T} (e(k-1) - e(k-2)) \right]$$

Restando ambas ecuaciones

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

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

Tomando la transformada Z:

$$(1 - z^{-1})U(z) = E(z) \left[ K_p \left( 1 + \frac{T}{T_i} + \frac{T_d}{T} \right) - K_p \left( 1 + 2\frac{T_d}{T} \right) z^{-1} + K_p \left( \frac{T_d}{T} \right) z^{-2} \right]$$

$$\frac{U(z)}{E(z)} = C(z) = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{1 - z^{-1}} \quad (4.2.6.6)$$

Donde:

$$q_0 = K_p \left( 1 + \frac{T}{T_i} + \frac{T_d}{T} \right)$$

$$q_1 = -K_p \left( 1 + \frac{2T_d}{T} \right)$$

$$q_2 = K_p \frac{T_d}{T}$$

Como el control fue un regulador PI simplificando se tiene:

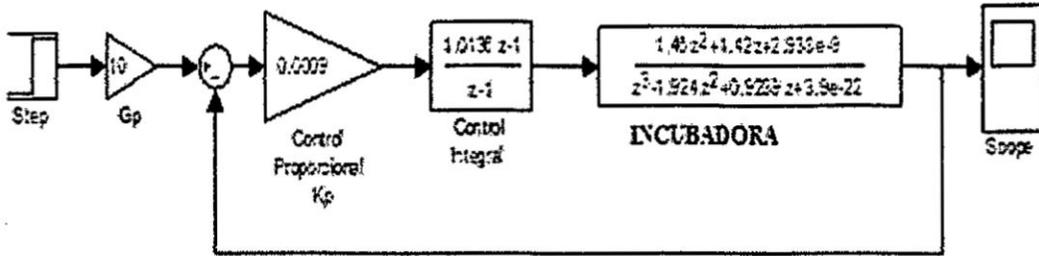
$$G_c(z) = \frac{K_p \left( \left( 1 + \frac{T}{T_i} \right) z - 1 \right)}{z - 1} \quad (4.2.6.7)$$

Reemplazando valores con  $T = 2$  min,  $K_p = 9 \times 10^{-3}$  y  $K_i = 146.8$

$$G_c(z) = \frac{0.009(1.036z - 1)}{z - 1} \quad (4.2.6.8)$$

La figura 4.5 muestra el diagrama de bloques del sistema con los valores de las ganancias y constantes de tiempo hallados.

FIGURA N° 4.5 DIAGRAMA DE BLOQUES DEL SISTEMA EN EL PLANO Z



#### 4.2.7 SINTONIZACIÓN DE CONTROLADOR MEDIANTE ZIEGLER-NICHOLS

.En lazo abierto, muchos procesos pueden definirse según la siguiente función de transferencia:

$$G(s) = \frac{K_0 e^{-s\tau_0}}{1 + \gamma_0 s}$$

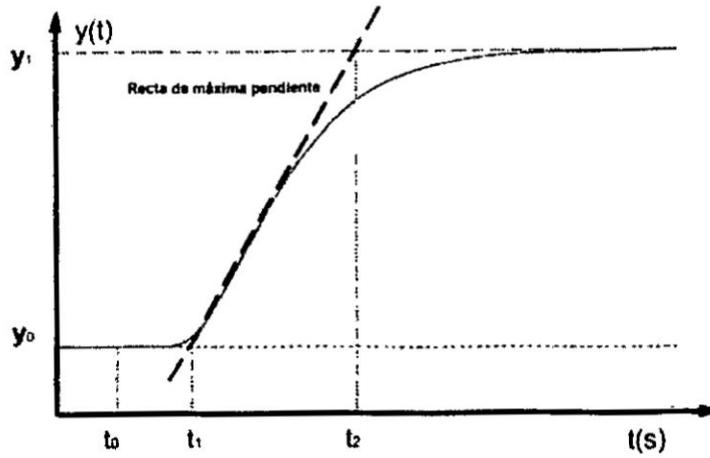
Donde los coeficientes  $K_0$ ,  $\tau_0$  y  $\gamma_0$  se obtienen de la respuesta del sistema en lazo abierto a una entrada escalón.

Se parte del sistema estabilizado en  $y(t) = y_0$  para  $u(t) = u_0$

Se aplica una entrada escalón de  $u_0$  a  $u_1$  (el salto debe estar entre un 10% y un 20% del valor nominal) y se registra la respuesta de la salida hasta que se establezca en el nuevo punto de operación.

Los parámetros se pueden obtener de la respuesta mostrada en la figura siguiente.

Fig. 4.7 : Respuesta de salida ante una entrada escalón



Dónde:

$$\tau_0 = t_1 - t_0$$

$$\gamma_0 = t_2 - t_1$$

$$k_0 = \frac{y_1 - y_0}{u_1 - u_0}$$

Según Ziegler-Nichols, la relación de estos coeficientes con los parámetros del controlador es la siguiente:

$$K_p = 1.2 \frac{\gamma_0}{k_0 \tau_0} \quad T_i = 2\tau_0 \quad T_d = 0.5\tau_0$$

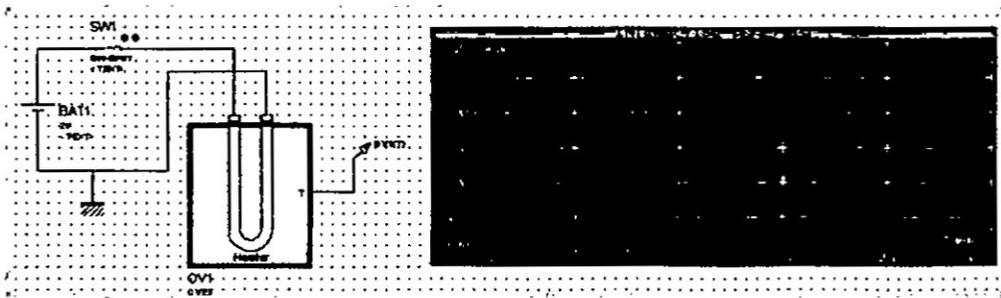
#### 4.2.8 MODELO DEL SISTEMA DE CALEFACCIÓN DE LA CAMARA MICROBIOLÓGICA

Para facilitar la simulación se establecerán los siguientes valores de parámetros funcionales del modelo OVEN:

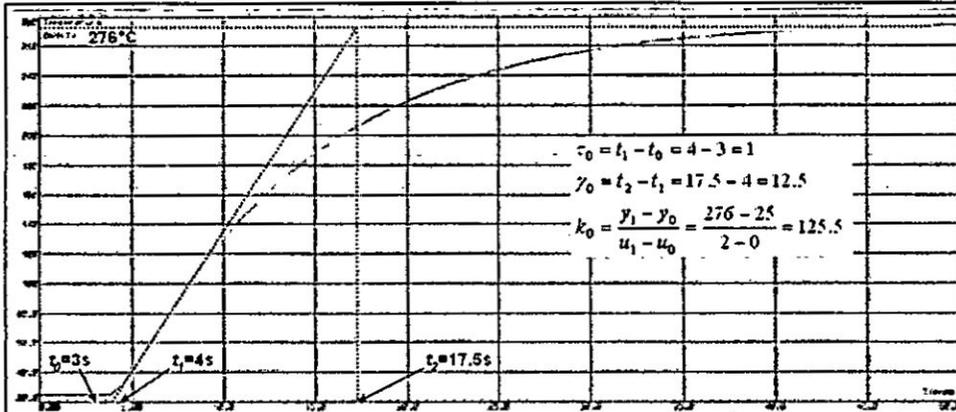
- *Temperature Ambient* (°C)= 25
- *Thermal Resistance to Ambient* (°C/W)= 0.7
- *Oven Time Constant* (sec)= 10

- Heater Time Constant (sec)= 1
- Temperature coefficient (V/°C)= 1
- Heating Power (W)= 120

El modelo esquemático OVEN contiene un terminal sensor T que entrega un voltaje proporcional a la temperatura del sistema. De acuerdo a los parámetros establecidos anteriormente, este terminal entregara 1V/°C, es decir, que para una temperatura de 100°C, el terminal T entregara 100V. Para obtener la respuesta del sistema en lazo abierto ante una entrada escalón (curva de reacción), se utiliza el sistema de análisis interactivo de ISIS Proteus Interactive Analysis (Graph Mode Tool), el cual interactúa con el sistema OVEN mediante el uso de un Voltage Probe1 OV1(T), según se muestra en la Figura . Observar que para realizar la entrada escalón de 0V a 2V se utiliza un interruptor SW1.



De la recta de máxima pendiente se deducen los parámetros  $\tau_0$ ,  $\gamma_0$  y  $k_0$  definidos por el análisis en lazo abierto de Ziegler-Nichols.



Por tanto el modelo del sistema de calefacción queda definido así:

$$G(s) = \frac{K_0 e^{-s\tau_0}}{1 + \gamma_0 s} = 125.5 \frac{e^{-s}}{1 + 12.5s}$$

Los parámetros  $K_p$ ,  $T_i$  y  $T_d$  se calculan según la Regla de Sintonización de Ziegler- Nichols basada en la respuesta al escalón:

$$K_p = 1.2 \frac{\gamma_0}{k_0 \tau_0} = 0.1195 \quad T_i = 2\tau_0 = 2 \quad T_d = 0.5\tau_0 = 0.5$$

Reemplazando los valores de  $K_p$ ,  $T_i$  y  $T_d$  en las ecuaciones dadas anteriormente, y estableciendo un periodo de muestreo  $T = 0.1s$  según criterio  $T < \tau_0/4$ , los parámetros del controlador discreto son:

$$a = K_p = 0.1195 \quad b = \frac{K_p T}{T_i} = 0.0062 \quad c = \frac{K_p T_d}{T} = 0.6215$$

Estos son los valores que utilizaremos cuando programemos el PID en el siguiente capítulo.

#### 4.2.9 Diseño del circuito de acondicionamiento

A la hora de elegir el tipo de sensor de temperatura se tuvo en cuenta los siguientes aspectos:

- a) Elección del rango de temperatura, teniendo en cuenta los rangos de voltaje admitidos por los dispositivos ADC.

Partiendo de las características del sensor RTD Pt – 100:

- DIN Clase A: rango de temperatura de -90°C a 300°C
- Coeficiente de temperatura:  $\alpha = 0.00385 \text{ } ^\circ\text{C}^{-1}$  (de 0°C a 100°C)
- Intensidad de medida: 0.5mA
- Intensidad máxima del sensor: 7mA
- Rango de voltaje admitido con el convertidor ADC: 0 a 5Vdc

La resistencia del sensor RTD está definida por:

$$R_T = R_0(1 + \alpha * T) \quad (4.2.10.1)$$

Según la norma DIN EN 60751, en su tabla de valores básicos en ohmios para sensores de temperatura Pt 100 escalonados de 1°C en 1°C, podemos extraer el valor máximo y mínimo de la resistencia para nuestro rango de temperaturas del sensor, definido anteriormente.

$$R_{T(300^\circ\text{C})} = 212.14 \text{ ohmios}$$

$$R_{T(100^\circ\text{C})} = 138.5 \text{ ohmios}$$

$$R_{T(-90^\circ\text{C})} = 64.3 \text{ ohmios}$$

Calculando el valor máximo posible, aplicado al rango definido:

$$I_{RT} = 0.5 \text{ mA}$$

$$V_{RT(300^\circ\text{C})} = 0.5\text{mA} * 212.14\Omega = 106.07 \text{ mV}$$

$$V_{RT(100^{\circ}C)} = 0.5mA * 138.5 \Omega = 69.25 mV$$

$$V_{REF(ADC)} = 5V$$

El voltaje de salida  $V_{out,max}$  no podrá superar los 5V, configurados como voltaje máximo de referencia del ADC. En el diseño se deberá cumplir la siguiente relación:

$$V_{OUT(MAX)} = V_{RT(100^{\circ}C)} * G$$

Donde:

G = ganancia del amplificador de instrumentación

Reemplazando valores y despejando el valor de G:

$$G = \frac{5V}{69.25} = 72.2$$

Se toma un valor  $G = 72$

De esta manera se obtuvo el valor de voltaje máximo de salida del circuito analógico que se introdujo al ADC.

$$V_{OUT(100^{\circ}C)} = 69.25 * 72 = 4.98V$$

Calculando el valores mínimo de salida del circuito considerando el rango de control entre  $0^{\circ}C$  y  $100^{\circ}C$ .

$$V_{RT(0^{\circ}C)} = 0.5mA * 100 \Omega = 50 mV$$

$$V_{OUT(0^{\circ}C)} = 50mV * 72 = 3.6V$$

b) Intensidad de corriente para evitar el efecto Joule por auto calentamiento.

Inicialmente nos encontramos con el valor de auto calentamiento indicado en la ficha del sensor:

$$E = 0.25^{\circ}\text{C}/\text{mW}$$

El incremento de temperatura por efecto Joule será:

$$\Delta T = (I_{RT})^2 * R_T * E \quad (4.2.10.2)$$

Definiendo el rango de control de temperatura entre 0°C y 100°C y calculando la resistencia del sensor en el rango mencionado

$$R_{Tamb(MAX)} = 138.5 \text{ ohmios}$$

$$R_{Tamb(MIN)} = 100 \text{ ohmios}$$

La intensidad máxima y mínima que circula por el sensor será:

$$I_{RT(MAX)} = \frac{V_{RT(MAX)}}{R_{Tamb(MIN)}}$$

$$I_{RT(MAX)} = \frac{69.25\text{mV}}{100\Omega} = 0.6925\text{mA}$$

$$I_{RT(MIN)} = \frac{V_{RT(MIN)}}{R_{Tamb(MAX)}}$$

$$I_{RT(MIN)} = \frac{50\text{mV}}{138.5\Omega} = 0.36\text{mA}$$

Por lo tanto el incremento máximo de temperatura por Efecto Joule será:

$$\Delta T_{MAX} = (I_{RT(MAX)})^2 * R_{Tamb(MAX)} * E \quad (4.2.10.3)$$

$$\Delta T_{MAX} = (0.6925\text{mA})^2 * 138.5\Omega * 0.25 \frac{^{\circ}\text{C}}{\text{mW}} = 0.0332^{\circ}\text{C}$$

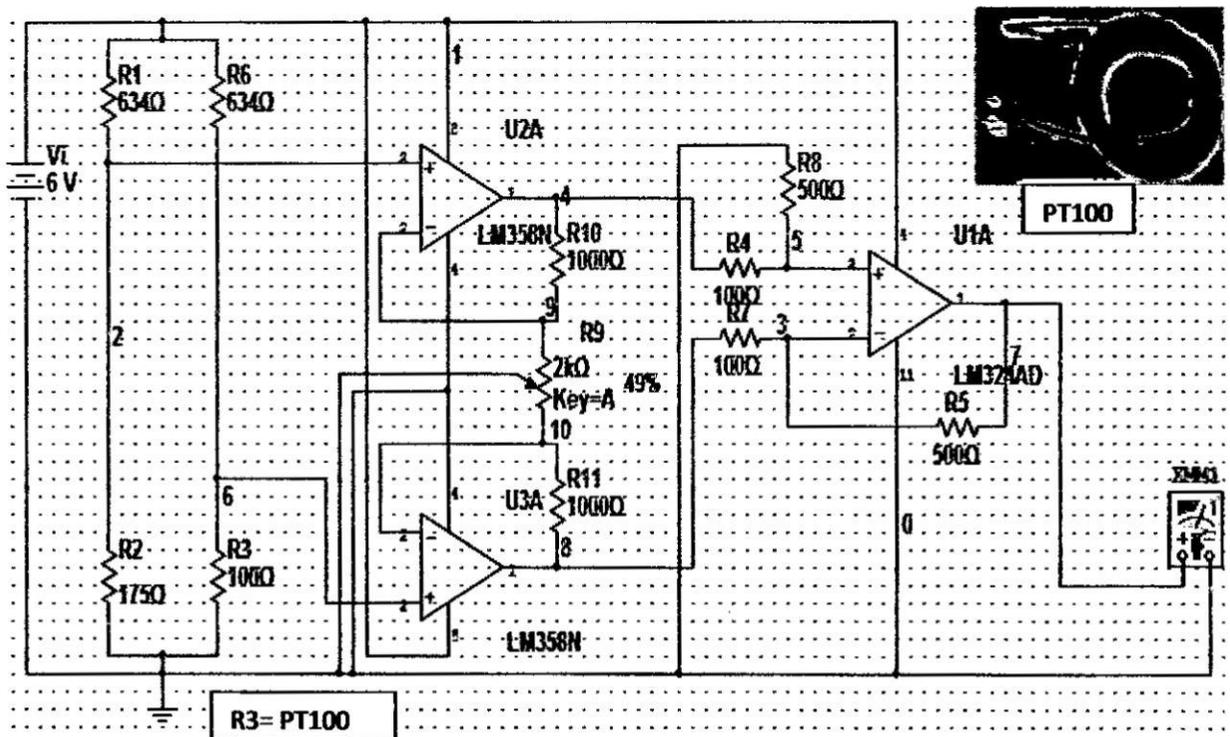
Como podemos apreciar el posible aumento máximo de temperatura provocado por el efecto Joule de auto calentamiento es de 0.0332°C si se trabaja a 100°C, el cual es un valor lo suficientemente bajo para poder influir en la medida real. Si además suponemos que trabajamos a una temperatura de 37°C el incremento por auto calentamiento de temperatura

se verá reducido aún más. Podemos concluir que el diseño planteado no se será afectado por el Efecto Joule de auto calentamiento.

El circuito de acondicionamiento se diseñó en base a una sonda PT-100 situada en puente resistivo, seguido de un amplificador diferencial, el cual se implementado con el amplificador de instrumentación LM324, el cual es un amplificador de instrumentación, cuya estabilidad de la tensión offset, ganancia de tensión en el tiempo y variaciones de temperatura es excelente.

Se utilizó resistencias de precisión , un transformador de 6 voltios

**FIGURA Nº 4.8 CIRCUITO DE MEDICIÓN DE TEMPERATURA**

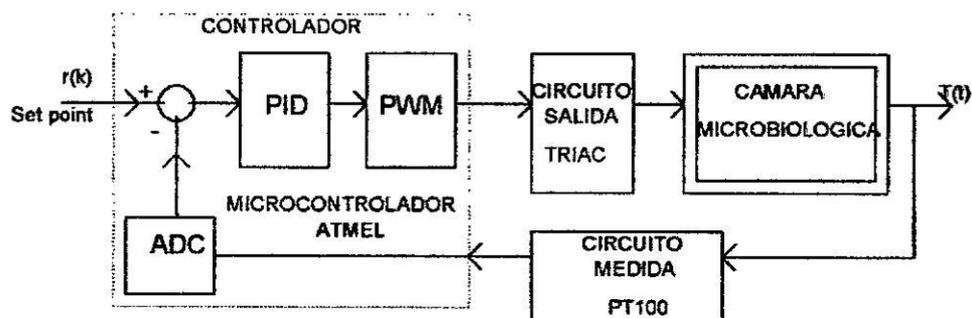


Este circuito se simulo en el programa multisim y luego se implemento

#### 4.2.10 Diseño de la interface del circuito de salida

La potencia que disipa la resistencia calefactora es controlada por un Triac por control de fase, generada por un microcontrolador Atmel 169, el cual se implementa como un generador de pulsos, de disparo.

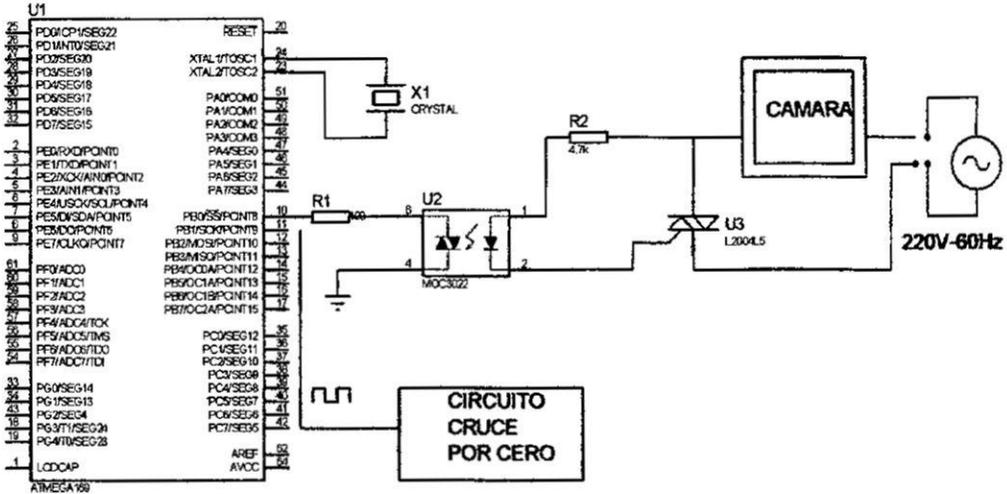
FIGURA N° 4.9 DIAGRAMA DE BLOQUES DEL SISTEMA



El programa en lenguaje C, ejecuta la rutina de control PID cada 100ms. Una rutina de interrupción, que se ejecuta sincronizada, con los cruces por cero de la onda senoidal de entrada, envía los pulsos de disparo al dispositivo de potencia TRIAC.

El programa escrito en lenguaje C usando el compilador CCS se presenta en el anexo m para el microcontrolador Atmel 169 .

**FIGURA N° 4.10 DIAGRAMA CIRCUITAL DE LA INTERFACE  
CONTROLADOR – CIRCUITO DE SALIDA**



Suponiendo que el Triac conduce  $180^\circ$  la corriente eficaz en la carga es:

$$I_{RL} = \frac{220}{484} = 0.45 \text{ Amp}$$

Se seleccionó un Triac el BT-136E, cuya corriente eficaz máxima es 4 amperios y tensión inversa máxima  $V_{DRM} = 600V$ . Según las curvas características del fabricante para una corriente eficaz de 0.45 amperios y un ángulo de conducción de  $180^\circ$  la potencia disipada es  $P_D = 0.8W$ .

La potencia que soporta el Triac sin disipador es:

$$P_{Dmax} = \frac{T_j - T_{amb}}{R_{thj-a}} = \frac{150^\circ C - 50^\circ C}{60^\circ C/W} = 1.67W$$

Por tanto, como  $P_{Dmax} < P_D$  no es necesario disipador.

El microcontrolador controla el ángulo de conducción del Triac, envía los pulsos de disparo usando como interface, un opto acoplador el MOC 3021. El Triac regula la corriente a través de la resistencia calefactora

mediante el controlador PI y un sensor Pt-100 dentro de un sistema en control en lazo cerrado para mantener la temperatura en 37°C.

**El proyecto funcionará de la siguiente manera:**

1. Al inicio, luego de encender la tarjeta por medio de un switch externo, se desplegará en pantalla la temperatura deseada a la cual se desea controlar la planta. Se ha programado una temperatura deseada inicial de 30°C para comodidad del usuario.
2. Después de haber fijado la temperatura deseada por medio del joystick, el controlador analiza la señal de error y envía una señal de tipo PWM al circuito de fuerza, el mismo que regulará la corriente que circula por la bombilla.
3. Luego de un tiempo determinado el sistema logrará estabilizarse con un error de estado estable de  $\pm 1^\circ\text{C}$ .

#### **4.2.11 IMPLEMENTACIÓN Y SIMULACIÓN**

##### **IMPLEMENTACIÓN**

En el anexo se ilustra la codificación de los respectivos módulos del programa tales como LCD.H, Teclado, ADC.H, PID.H y el código principal CONTROLADOR\_TEMPERATURA.C.

Para desarrollar el proyecto hicimos uso de dos tipos de software: AVR Studio, cuyo fin es la programación del ATmega169 y Proteus que nos servirá para la simulación completa del proyecto

Los componentes físicos que utilizamos para la realización del proyecto fueron: el Kit AVR Atmel 169 , el sensor de temperatura PT100 y circuito de acondicionamiento de medida de la temperatura la cámara eléctrica

microbiológica , circuito de salida o de potencia acoplado ópticamente con el Atmel 169.

### **MICROCONTROLADOR ATMEGA169**

El ATmega169 es un Microcontrolador de 8 bits con arquitectura AVR RISC, este posee las siguientes características:

- Arquitectura RISC avanzada.
  - Conjunto de 130 instrucciones ejecutables en un solo ciclo de reloj.
  - 32 x 8 registros de trabajo de propósito general.
  - Rendimiento de hasta 16 MIPS a 16 MHz.
  - Memoria no volátil para Programa y Datos.
  - Flash de 16 K bytes, auto-programable en el sistema.

### **Teclado**

Para operar el AVR atmel 169 se emplea el teclado como una entrada para el usuario. Este opera en cinco direcciones, incluyendo presión en el centro.

### **Modulo LCD**

En las aplicaciones donde es necesaria la interacción con el usuario es muy útil poder mostrar información para el usuario. Una interfaz muy simple para mostrar información podría ser el estado de unos LEDs; mientras que la interacción más compleja puede beneficiarse de una pantalla capaz de desplegar letras, números, palabras o incluso oraciones. Las Pantallas de Cristal Líquido (LCD) son frecuentemente

usadas para desplegar mensajes. Los módulos LCD pueden ser gráficos y se los puede usar para desplegar gráficos y texto, ó pueden ser alfanuméricos capaces de visualizar entre 10 y 80 caracteres. Los módulos LCD alfanuméricos estándar son fáciles de conectar, pero son bastante costosos debido a que tienen incorporado drivers/controladores que se ocupan de la generación de los caracteres/gráficos sobre el vidrio LCD.

El microcontrolador ATmega169 tiene un controlador LCD (LCD Driver) integrado capaz de controlar hasta 100 segmentos. El núcleo altamente eficiente y el consumo de corriente muy bajo de este dispositivo lo hace ideal para aplicaciones energizadas por batería que requieren de una interfaz humana.

### **SIMULACIONES**

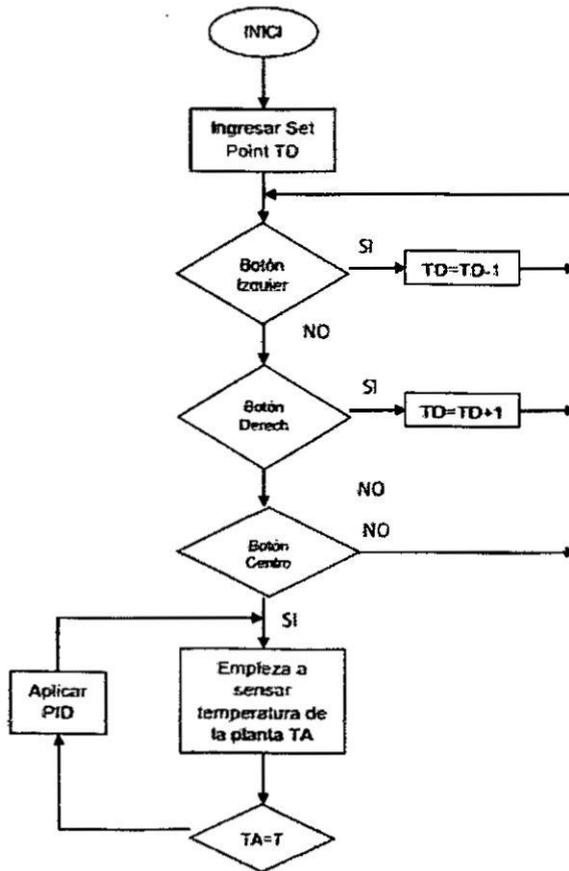
En esta etapa del proyecto se presentarán las respectivas simulaciones en Proteus.

En la gráfica posterior se muestra la temperatura deseada inicial programada de 40°C como se indicó anteriormente.

En las siguientes gráficas se mostrarán las temperaturas deseadas de 40°C, 60°C con sus respectivas respuestas de temperatura del sistema donde se puede observar el tiempo de pico, el valor de temperatura pico, el tiempo de estabilización y el error de estado estable aproximadamente de  $\pm 1^\circ\text{C}$ .

### Diagrama de bloques del algoritmo de control implementado

Fig. 3.1: Diagrama de flujo del sistema de control PID de temperatura.



El proyecto funcionará de la siguiente manera:

1. Al inicio, luego de encender el sistema por medio de un switch externo, se desplegará en pantalla la temperatura deseada a la cual se desea controlar la planta. Se ha programado una temperatura deseada inicial de 30°C para comodidad del usuario.
2. Después de haber fijado la temperatura deseada por medio del teclado, el controlador analiza la señal de error y envía una señal de tipo PWM al circuito de fuerza, el mismo que regulará la

corriente que circula por la resistencia de calefacción de la cámara .

3. Luego de un tiempo determinado el sistema logrará estabilizarse con un error de estado estable de  $\pm 1^{\circ}\text{C}$ .

### Resultado de las simulaciones

Fig. 4.11: Sistema implementado para la simulación en proteus

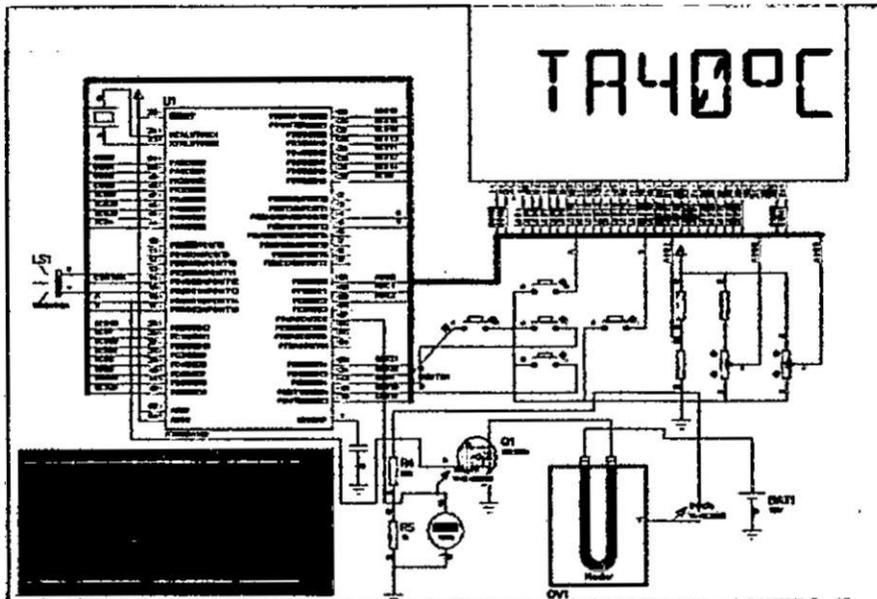
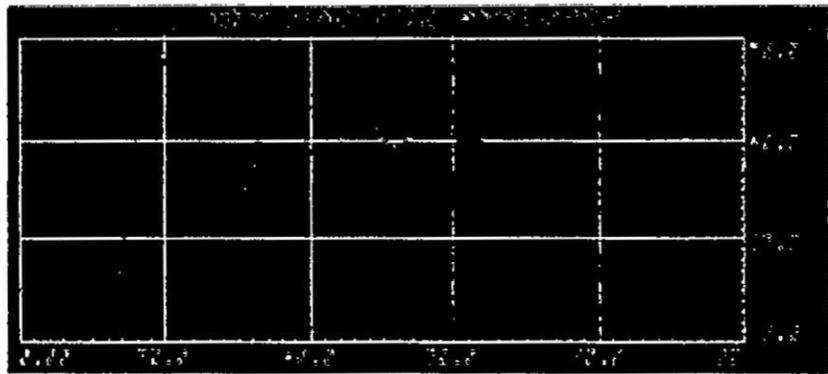
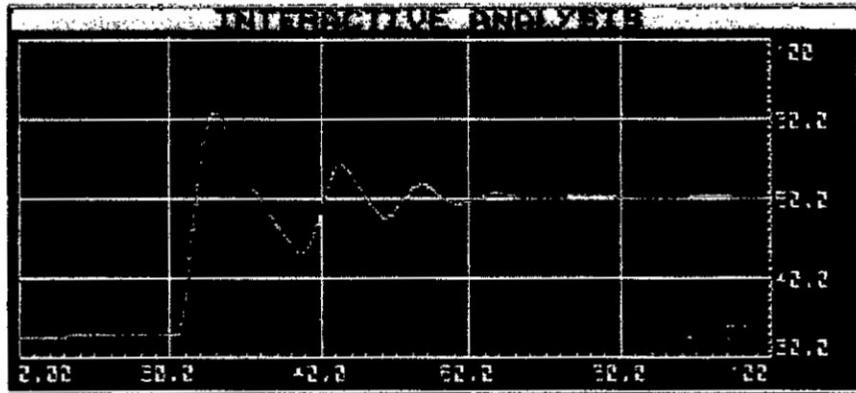


Fig. 4.12: Respuesta de temperatura del sistema a 40°C



**Fig. 4.13: Respuesta de temperatura del sistema a 60°C**



#### **4.3 Población y muestra**

Por el tipo de investigación no corresponde determinar el tamaño de la muestra.

#### **4.4 Técnicas e instrumentos de recolección de datos**

Debido a que no se ha determinado el tamaño de la muestra, no corresponde utilizar las técnicas e instrumentos de recolección de datos.

#### **4.5 Procedimientos de recolección de datos**

No corresponde al presente trabajo.

#### **4.6 Procesamiento estadístico y análisis de datos**

No corresponde utilizar procesamiento y análisis de datos

## CAPITULO V

### V. RESULTADOS

#### 5.1 Resultados parciales

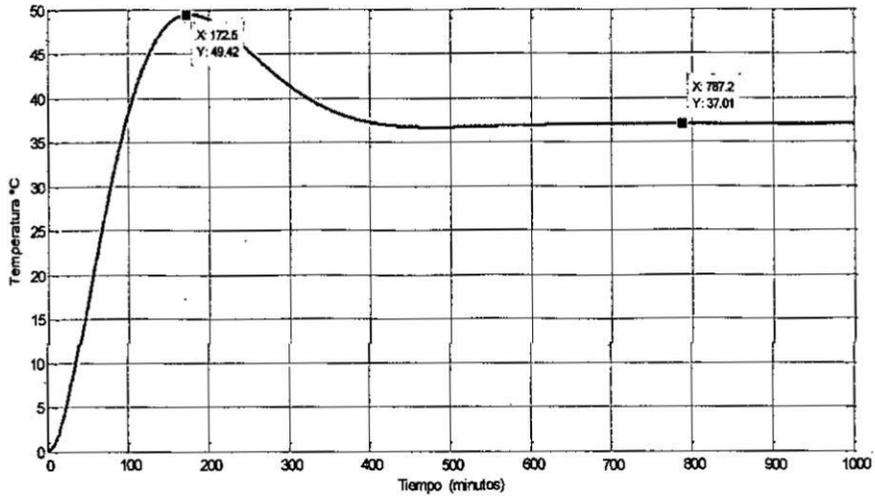
Los datos de temperatura que se muestran en la tabla fueron medidos en el rango de temperatura configurable por el usuario y el estipulado en la norma desde 30 a 39 °C con incrementos de 0.5°C . En la primera columna se encuentran los valores de temperatura correspondiente al sensor PT100 y otros medidores para comparar con sensores tipo cutáneo, fluque (ti25) y por último con un termómetro infrarrojo (fluque 62MAX)

Tabla n Valores de temperatura obtenidos durante las pruebas

Temperatura referencia	Pt100	W10001A	Fluque Ti25	Fluque 62Max
32	31,5	31,0	31,8	31,6
32,5	31,7	31,1	32,1	32,2
<b>33</b>	<b>32,4</b>	<b>31,9</b>	<b>32,6</b>	<b>32,8</b>
33,5	32,7	32,3	33,2	33,0
<b>34</b>	<b>33,5</b>	<b>32,5</b>	<b>33,5</b>	<b>33,4</b>
34,5	33,9	33,4	34,2	34,3
35	34,0	33,5	34,5	34,7
35,5	34,8	34,3	35,3	35,1
<b>36</b>	<b>35,2</b>	<b>34,1</b>	<b>35,8</b>	<b>35,9</b>

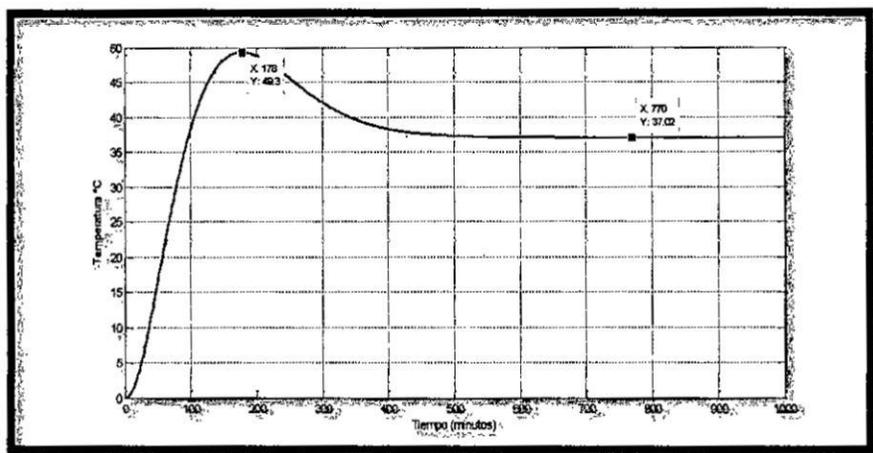
De acuerdo al diagrama de bloques mostrado en la figura N°4.4 (página 39) se procedió a la simulación cuyo resultado se aprecia en la figura N° 5.1 el regulador PI satisface las consideraciones de sobreimpulso y el factor de amortiguamiento requeridos. Sin embargo, para disminuir el sobreimpulso se utilizó el Algoritmo PID; de esta manera se logró un mejor ajuste de los parámetros de control.

**FIGURA N° 5.1 RESPUESTA A UN ESCALON DE REFERENCIA DE TEMPERATURA DEL SISTEMA TIEMPO CONTINUO**



De acuerdo al diagrama de bloques mostrado en la figura N°4.5 se procedió a la simulación cuyo resultado se aprecia en la figura N° 5.2 el regulador PI satisface las consideraciones de sobreimpulso y el factor de amortiguamiento requeridos.

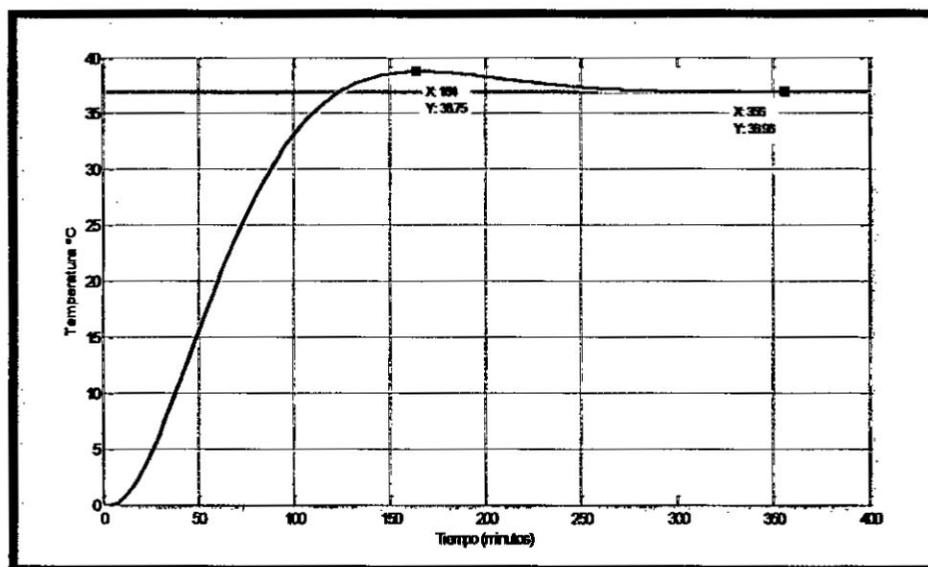
**FIGURA N° 5.2 RESPUESTA A UN ESCALON DE REFERENCIA DE TEMPERATURA DEL SISTEMA CON CONTROL DISCRETO**



## 5.2 Resultados finales

En la figura N° 5.3, se muestra la respuesta a un escalón de temperatura de referencia de 37°C, en el controlador de temperatura usando el algoritmo PID para mejorar la respuesta del sistema, en donde el control PI logra que el sobreimpulso de la temperatura no sea excesivo.

**FIGURA N° 5.3 RESPUESTA A UN ESCALON DE REFERENCIA DE TEMPERATURA**



El controlador PID digital permitió controlar la temperatura en el interior de la incubadora de manera satisfactoria de acuerdo a las condiciones establecidas en la norma internacional IEC 60601-2-19 referente a la seguridad básica y funcionamiento de los equipos médicos, obteniéndose:

- Un tiempo de calentamiento de 35 minutos.
- Un error de estado estable de 0.04°C superando a lo establecido en la norma (0.5°C).
- Un sobreimpulso de 1.75°C superando lo establecida en la norma (2°C).

## CAPITULO VI

### VI. DISCUSIÓN DE RESULTADOS

En el primer capítulo se obtuvo el modelo matemático de la incubadora microbiológica, usando los parámetros inherentes del mismo. A continuación se procedió a calcular la función de transferencia del convertidor y del transductor; consiguiendo de esta manera el diagrama de bloques del sistema incluyendo el controlador PID. Con la función de transferencia de la incubadora en el plano S se calculó los parámetros del control PID mediante el método Sigler Nichols, con el cual se obtuvo en forma aproximada la repuesta deseada. Asimismo transformando la función de transferencia de la incubadora al plano Z se calculó los parámetros del controlador PID digital consiguiendo una repuesta aproximada. Estos parámetros del controlador PID fueron el punto de partida para calcular los parámetros PID con mayor precisión y las simulaciones demostraron que se obtiene una repuesta en el tiempo más preciso.

Asimismo se diseñó un circuito de acondicionamiento que realiza la conversión de la temperatura a señal eléctrica de realimentación. De igual forma se propuso un programa en lenguaje C que ejecuta la acción de control PID obteniéndose la variable manipulada. Esta señal varía el ángulo de disparo del Triac que controla la corriente a través de las resistencias calefactoras regulando la temperatura dentro de la incubadora microbiológica.

### **6.1 Contratación de hipótesis con los resultados**

Según la hipótesis se trata de controlar en forma precisa la variación de temperatura de la incubadora. Como indicador se asume una variación de temperatura menor de  $0.2^{\circ}\text{C}$ . En la figura 5.3 (página 60) se observa que el error de temperatura en estado estacionario es menor de  $0.2^{\circ}\text{C}$  como se estableció en la hipótesis. Asimismo el sobreimpulso que se obtiene es de  $38.75^{\circ}\text{C}$  lo que representa un 5% adicional sobre la temperatura de referencia.

### **6.2 Contratación de hipótesis con otros estudios similares**

Consideramos los antecedentes del estudio indicados en el Capítulo II para la contratación de los resultados obtenidos en la presente tesis:

El trabajo realizado en el 2001, por Valverde y Castro que presentaron en el Congreso Nacional de Instrumentación, el diseño e implementación de un controlador difuso, para la regulación de temperatura en un horno eléctrico resistivo a un valor constante, en éste se describen las características del controlador difuso, diseñado e implementado con el objetivo, de mantener un valor de temperatura constante dentro del ambiente de un horno eléctrico resistivo, se utilizó un controlador difuso, se implementó sobre una arquitectura basada en el microcontrolador M68HC11.

En la presente tesis, se ha utilizado un sistema de control de lazo cerrado, para eliminar cualquier perturbación, que pueda desviar, el efecto final de la temperatura deseada, se utilizó un controlador PID, a la vez un microcontrolador Atmel Atmega169, generador de pulsos de disparo, a través de un optoacoplador, para manejar la resistencia calefactora y controlar su potencia, con un Triac, lo cual no sucede con el trabajo de la referencia.

En el 2006 Vásquez y Santiago presentaron en la Universidad Tecnológica de Mixteca, el artículo denominado "Diseño e implementación de un sistema de control de temperatura mediante Instrumentación Virtual", en éste se describe el diseño y la implementación de un controlador PID, para regular la temperatura del sistema AmiraLTR701. Se obtuvo la función de transferencia, para el diseño del controlador, se utilizó la herramienta Simulink y el método de Ziegler-Nichol's, obteniendo las ganancias. La simulación del sistema fue hecha en el software Matlab, para obtener los mejores valores de  $k_p$ ,  $K_i$  y  $K_d$ . Finalmente se usó el software LabVIEW para controlar la planta.

En la presente tesis, con la función de transferencia de la incubadora microbiológica, en el plano S se calculó los parámetros, del control PID mediante el método de sintonización de Nichol Sigler, con el cual se obtuvo en forma aproximada la respuesta deseada. Asimismo transformando la función de transferencia de la incubadora al plano Z se calculó los parámetros del controlador PID digital, consiguiendo una

## CAPITULO VIII

### VIII. RECOMENDACIONES

1. Se podría mejorar el modelo, o función de transferencia de la incubadora microbiológica, usando el método de identificación de los parámetros ARX.
2. Para el cálculo de los parámetros del controlador PID digital se debe restringir, el rango de valores dentro, de los cuales se halla la respuesta óptima.

## CAPITULO IX

### IX. REFERENCIAS BIBLIOGRÁFICAS

1. ARIAS M. Oswaldo, **Sistema de monitoreo de temperatura para horno de gas**; Universidad Tecnológica de la Mixteca, Tesis de Licenciatura, 2006.
2. HERNANDEZ, J., **Sistema de monitoreo y control de encendido de un horno eléctrico basado en un control de potencia tipo integral**; Universidad Tecnológica de la Mixteca, Tesis de Licenciatura, 2009.
3. KROHLING R.A. and REY P. J., **“Design of optimal disturbance rejection PID controllers using genetic algorithms”**, IEEE Transactions on Evolutionary Computation, Vol. 5, 2001, pp. 78-82.
4. PONCE CRUZ Pedro **“Inteligencia Artificial con aplicaciones a la Ingeniería”** Editorial Alfa Omega, Mexico, 2010, 348p
5. PORTER B. and JONES A.H. **“Genetic Tuning of digital PID controller”** , Electronic Letters Vol 28, 1992, pp 383-384
6. REINOSO Oscar “et al” **Control de Sistemas Discretos**, Editorial McMGRRAW-HILL, 2ª Edición , 2004, 334p.
7. VALVERDE Nicolás y CASTRO Henry, **Controlador difuso para un horno eléctrico resistivo**; Congreso Nacional de Instrumentación (México), 2001.

# ANEXOS

## ANEXO A

### Matriz de Consistencia

**Título: “CONTROL DE TEMPERATURA DE UNA INCUBADORA MICROBIOLÓGICA, UTILIZANDO UN CONTROLADOR PID”**

PROBLEMA	OBJETIVOS	HIPOTESIS	VARIABLES	MÉTODOS
<p><b>General</b></p> <p>¿Qué podemos hacer para controlar la temperatura de una incubadora microbiológica?</p>	<p><b>Objetivo general</b></p> <ul style="list-style-type: none"> <li>- Diseñar y simular un control de temperatura para una incubadora microbiológica, diseñado en base a un controlador PID sintonizado con Regla Nichols-Ziegler.</li> <li>- Controlar la incubadora microbiológica mediante un sistema de control de tipo discreto, en base a la variable Z con bloques de control de temperatura con procesador Atmel 169.</li> </ul> <p><b>Objetivo específico</b></p> <ul style="list-style-type: none"> <li>- Diseñar y simular el modelo de la incubadora en el plano Z y luego el sistema de control discreto con convertidor a tiristores usando el software Matlab con bloques en Simulink.</li> <li>-Diseñar y simular el circuito de salida a tiristores para el control de las resistencias de calefacción.</li> </ul>	<p><b>Objetivo general</b></p> <p>Un controlador de PID sintonizado con Prueba de Ziegler –Nichols controla en forma precisa la temperatura con microcontrolador Atmel 169.</p> <p><b>Objetivo específico</b></p> <p>Un controlador PID sintonizado con Prueba de Ziegler –Nichols controla en forma precisa la temperatura de una incubadora microbiológica.</p>	<p><b>Variables dependientes:</b></p> <p>Controlador PID sintonizado con Prueba Ziegler Nichols.</p> <p><b>Variables independientes:</b></p> <p>Controlar en forma precisa la temperatura de la incubadora.</p>	<p><b>General</b></p> <p>La metodología consiste en el control de temperatura de una incubadora microbiológica, utilizando un controlador PID para la validación correspondiente.</p> <p><b>Específicos</b></p> <ul style="list-style-type: none"> <li>❖ Analizar estructuras ya desarrolladas de controladores de temperatura.</li> <li>❖ Diseñar y simular el controlador aplicando la técnica de control PID con Atmel 169.</li> <li>❖ Efectuar conclusiones en base a los resultados y efectuar algunas recomendaciones para trabajos futuros.</li> </ul>

## Anexo B. Librería LCD.h para manejo del LCD

```
#include <avr/signal.h>

#define pLCDREG ((unsigned char*)(0xEC))

#define TAMANIO_DEL_REGISTRO_LCD 20

#define NUMERO_MAXIMO_DE_CARACTERES 36

char LCD_Data[TAMANIO_DEL_REGISTRO_LCD];

char memo_temp_texto[NUMERO_MAXIMO_DE_CARACTERES];

unsigned char ESCRITURA_DE_CADENA_HABILITADO = 0;

unsigned char LCD_INT_contador = 0;

void inicializar_LCD(void);

void escribir_caracter_en_LCD(char , char );

void escribir_palabras_en_LCD(char *);

void borrar_LCD(void);

void actualizar_LCD(void);

unsigned int tabla_de_caracteres_LCD[] PROGMEM =

    {

        0x0A51, // '*' (?)

        0x2A80, // '+'

        0x0000, // ';' (Sin definir)

        0x0A00, // '-'

        0x0A51, // '.' Signo de grados

        0x0000, // '/' (Sin definir)

        0x5559, // '0'

        0x0118, // '1'

        0x1e11, // '2'
```

0x1b11, // '3'  
0x0b50, // '4'  
0x1b41, // '5'  
0x1f41, // '6'  
0x0111, // '7'  
0x1f51, // '8'  
0x1b51, // '9'  
0x0000, // ':' (Sin definir)  
0x0000, // ';' (Sin definir)  
0x0000, // '<' (Sin definir)  
0x0000, // '=' (Sin definir)  
0x0000, // '>' (Sin definir)  
0x0000, // '?' (Sin definir)  
0x0000, // '@' (Sin definir)  
0x0f51, // 'A' (+ 'a')  
0x3991, // 'B' (+ 'b')  
0x1441, // 'C' (+ 'c')  
0x3191, // 'D' (+ 'd')  
0x1e41, // 'E' (+ 'e')  
0x0e41, // 'F' (+ 'f')  
0x1d41, // 'G' (+ 'g')  
0x0f50, // 'H' (+ 'h')  
0x2080, // 'I' (+ 'i')  
0x1510, // 'J' (+ 'j')  
0x8648, // 'K' (+ 'k')

```

0x1440, // 'L' (+ 'l')
0x0578, // 'M' (+ 'm')
0x8570, // 'N' (+ 'n')
0x1551, // 'O' (+ 'o')
0x0e51, // 'P' (+ 'p')
0x9551, // 'Q' (+ 'q')
0x8e51, // 'R' (+ 'r')
0x9021, // 'S' (+ 's')
0x2081, // 'T' (+ 't')
0x1550, // 'U' (+ 'u')
0x4448, // 'V' (+ 'v')
0xc550, // 'W' (+ 'w')
0xc028, // 'X' (+ 'x')
0x2028, // 'Y' (+ 'y')
0x5009, // 'Z' (+ 'z')
0x0000, // '[' (Sin definir)
0x0000, // '\' (Sin definir)
0x0000, // ']' (Sin definir)
0x0000, // '^' (Sin definir)
0x0000 // '_'
};

void inicializar_LCD(void)
{
borrar_LCD();
LCDCRA = (1<<LCDEN) | (1<<LCDAB);

```

```

LCDCCR =
(1<<LCDDC2)|(1<<LCDDC1)|(1<<LCDDC0)|(1<<LCDCC3)|(1<<LCDCC2)
|(1<<LCDCC1)|(1<<LCDCC0);

```

```

    ASSR = (1<<AS2);

```

```

LCDFRR = (0<<LCDPS0) | (1<<LCDCD1)|(1<<LCDCD0);

```

```

LCDCRB =
(1<<LCDCS)|(1<<LCDMUX1)|(1<<LCDMUX0)|(1<<LCDPM2)|(1<<LCDP
M1)|(1<<LCDPM0);

```

```

    LCD CRA |= (1<<LCDIE);

```

```

}

```

```

void escribir_caracter_en_LCD(char c, char posicion)

```

```

{

```

```

    unsigned int seg = 0x0000;

```

```

    char mascara, nibble;

```

```

    char *ptr;

```

```

    char i;

```

```

    if (posicion > 5) return;

```

```

    if ((c >= '*') && (c <= 'z'))

```

```

    {

```

```

        if (c >= 'a') c &= ~0x20;

```

```

        c -= '*';

```

```

    seg=(unsigned int)

```

```

    pgm_read_word(&tabla_de_caracteres_LCD[(uint8_t)c]);

```

```

    }

```

```

    if (posicion & 0x01)

```

```

        mascara = 0x0F;

```

```

    else

```

```

        mascara = 0xF0;
        ptr = LCD_Data + (posicion >> 1);
        for (i = 0; i < 4; i++)
        {
            nibble = seg & 0x000F;
            seg >>= 4;
            if (posicion & 0x01)
                nibble <<= 4;
            *ptr = (*ptr & mascara) | nibble;
            ptr += 5;
        }
    }

void escribir_palabras_en_LCD(char *palabra)
{
    unsigned char i=0;
    for( i=0;i<NUMERO_MAXIMO_DE_CARACTERES;i++)
        memo_temp_texto[i]='\0';
    LCD_INT_contador = 0;
    ESCRITURA_DE_CADENA_HABILITADO = 1;
    for(i=0;(i<NUMERO_MAXIMO_DE_CARACTERES)&&(*palabra!
    ='\0');i++, palabra++)
        memo_temp_texto[i]=*palabra;
}

void borrar_LCD(void)
{

```

```

        unsigned char i=0;
    for( i=0;i<NUMERO_MAXIMO_DE_CARACTERES;i++)
        memo_temp_texto[i]='\0';
    for (i = 0; i < TAMANIO_DEL_REGISTRO_LCD; i++)
        {
            *(pLCDREG + i) = 0x00;
            *(LCD_Data+i) = 0x00;
        }
    actualizar_LCD();
}

void actualizar_LCD(void)
{
    ESCRITURA_DE_CADENA_HABILITADO = 0;
    for (char i = 0; i < TAMANIO_DEL_REGISTRO_LCD; i++)
        *(pLCDREG + i) = *(LCD_Data+i);
}

SIGNAL(SIG_LCD)
{
    unsigned char letra=0;
    unsigned char i=0;
    if (ESCRITURA_DE_CADENA_HABILITADO==1)
    {
        for(i=0;(i<6);i++)
        {

```

```

        if(!(memo_temp_texto[i+LCD_INT_contador]!='\0'))
            {
            letra = memo_temp_texto[i+LCD_INT_contador];
            escribir_caracter_en_LCD(letra,i);
            }
        else
            {
            escribir_caracter_en_LCD(' ',i);
            }
        _delay_loop_2(20000);
    }
}

if(LCD_INT_contador<NUMERO_MAXIMO_DE_CARACTERES)
    LCD_INT_contador++;
else
    {
        LCD_INT_contador=0;
        ESCRITURA_DE_CADENA_HABILITADO = 0;
    }
}

for (char i = 0; i < TAMANIO_DEL_REGISTRO_LCD; i++)
    *(pLCDREG + i) = *(LCD_Data+i);
}

```

## Anexo C : Librería para manejar Teclado.H

```
/*  
Pines del ATmega169 conectados con el Joystick:  
  
-----  
Bit          7  6  5  4  3  2  1  0  
  
-----  
PORTB   B  A   O  
PORTE   D  C  
  
-----  
PORTB | PORTE  B  A   O  D  C => posición  
  
-----  
*/  
  
#define MASCARA_PINB (1<<PINB4)  
#define MASCARA_PINE ((1<<PINE3)|(1<<PINE2))  
#define IZQUIERDA 2  
#define DERECHA 3  
#define CENTRO 4  
#define NO_VALIDA 5  
#define posicion_C 2 //DERECHA  
#define posicion_D 3 //IZQUIERDA  
#define posicion_O 4 //CENTRO  
#define VERDADERO 1  
#define FALSO 0  
  
volatile unsigned char SELECCION = 0;
```

```

volatile unsigned char SELECCION_VALIDA = 0;

int TD=30;

volatile int enter=0;

void inicializar_joystick(void);

void manejar_interrupcion(void);

void obtener_seleccion(void);

void inicializar_joystick(void)
{
    CLKPR = (1<<CLKPCE);
    CLKPR = (1<<CLKPS3);
    while(CLKPR & (1<<CLKPCE));
    DDRB |= 0xD0;
    PORTB |= MASCARA_PINB;
    DDRE |= 0x0C;
    PORTE |= MASCARA_PINE;
    DDRB = 0;//entrada
    PORTB = MASCARA_PINB;//habilitar PULL-UPs
    DDRE = 0;//entrada
    PORTE = MASCARA_PINE;//habilitar PULL-UPs
    PCMSK1 |= MASCARA_PINB;
    PCMSK0 |= MASCARA_PINE;
    EIFR = ((1<<PCIF1)|(1<<PCIF0));
    EIMSK = ((1<<PCIE1)|(1<<PCIE0));
    DDRD = 0xFF;
    PORTD = 0x00;
}

```

```

}
void manejar_interrupcion(void)
{
    unsigned char joystick;
    unsigned char seleccion;
    joystick = ((~PINB) & MASCARA_PINB);
    joystick |= ((~PINE) & MASCARA_PINE);
    if((joystick & (1<<posicion_C)))
        seleccion = DERECHA;
    else if((joystick & (1<<posicion_D)))
        seleccion = IZQUIERDA;
    else if((joystick & (1<<posicion_O)))
        seleccion = CENTRO;
    else
        seleccion = NO_VALIDA;
    if(seleccion != NO_VALIDA)
    {
        if(!SELECCION_VALIDA)
        {
            SELECCION = seleccion;
            SELECCION_VALIDA = VERDADERO;
        }
    }
    EIFR = ((1<<PCIF1)|(1<<PCIF0));
    obtener_seleccion();
}

```

```

}
void obtener_seleccion(void)
{
    char temperatura_ASCII[]={ '0', '0', '\0' };
    unsigned char seleccion;
    enter=0;
    cli();
    if(SELECCION_VALIDA)
    {
        seleccion = SELECCION;
        SELECCION_VALIDA = FALSO;
    }
    else seleccion = NO_VALIDA;
    if(seleccion != NO_VALIDA)
    {
        switch(seleccion)
        {
            case CENTRO:
                enter=1;
                break;
            case IZQUIERDA:
                TD++;
                enter=0;
        }
        itoa(TD,temperatura_ASCII,10);
        escribir_caracter_en_LCD('T',0);
    }
}

```

```

        escribir_caracter_en_LCD('D',1);

escribir_caracter_en_LCD(temperatura_ASCII[0],2);

escribir_caracter_en_LCD(temperatura_ASCII[1],3);
        escribir_caracter_en_LCD('.',4);
        escribir_caracter_en_LCD('C',5);
        actualizar_LCD();
        break;

        case DERECHA:
                TD--;
                enter=0;
                itoa(TD,temperatura_ASCII,10);
                escribir_caracter_en_LCD('T',0);
                escribir_caracter_en_LCD('D',1);

escribir_caracter_en_LCD(temperatura_ASCII[0],2);

escribir_caracter_en_LCD(temperatura_ASCII[1],3);
        escribir_caracter_en_LCD('.',4);
        escribir_caracter_en_LCD('C',5);
        actualizar_LCD();
        break;
        default:
                break;
}

```

```
}  
sei();  
}  
SIGNAL(SIG_PIN_CHANGE0)  
{  
    manejar_interrupcion();  
}  
SIGNAL(SIG_PIN_CHANGE1)  
{  
    manejar_interrupcion();  
}
```

## Anexo D librería para controlar el ADC.H

```
void Inicializar_ADC(void);
int leer_ADC(void);
void Inicializar_ADC(void)
{
    CLKPR = (1<<CLKPCE);
    CLKPR = (1<<CLKPS1);
    ADMUX= (1<<REFS0)|(1<<MUX2);
    ADCSRA = (1<<ADEN)|(1<<ADPS2);
    leer_ADC();
}
int leer_ADC(void)
{
    char i;
    int ADC_temp;
    int ADCsuma = 0;
    sbi(PORTF, PF3);
    sbi(DDRF, DDF3);
    sbi(ADCSRA, ADEN);
    ADCSRA |= (1<<ADSC);
    while(!(ADCSRA & 0x10));
    for(i=0;i<8;i++)
    {
        ADCSRA |= (1<<ADSC);
        while(!(ADCSRA & 0x10));
```

```
    ADC_temp = ADCL;
    ADC_temp += (ADCH << 8);
    ADCsuma += ADC_temp;
}
ADCsuma = (ADCsuma >> 3);
cbi(PORTF,PF3);
cbi(DDRF,DDF3);
cbi(ADCSRA, ADEN);
return ADCsuma;
}
```

## Anexo E Librería PID.H

```
void PID(void);
void Inicializar_PWM(void);
void leer_temperatura(void);
int valor;

float a=0.1243,b=0.0062,c=0.6215; // Constantes para parámetros de
// controlador PID

float rT,eT,iT,dT,yT,uT,iT0=0,eT0=0; // Variables de controlador PID

float max=1023,min=0; // Variables para anti-windup
extern int TD;
void PID(void)
{
    valor=leer_ADC()/3.1; //Leer ADC
    leer_temperatura();
    yT=valor*10; // Amplificación de la señal de salida (TA)
    rT=TD*10; // Amplificación de la señal de entrada (TD)
    eT=rT-yT; // Cálculo de la señal de error e(kT)
    iT=b*eT+iT0; // Cálculo del término integrativo i(kT)
    dT=c*(eT-eT0); // Cálculo del término derivativo d(kT)
    uT=iT+a*eT+dT; // Cálculo de la señal de control u(kT)
    if (uT>max) // Anti-windup
        uT=max;
    else if (uT<min)
        uT=min;
    OCR1B=uT;
```

```

        iT0=iT;

        eT0=eT;

_delay_loop_2(25000); // Periodo de muestreo T=0.1s
}

void leer_temperatura(void)
{
    char temperatura_ASCII[]={0,'0','\0'};
    itoa(valor,temperatura_ASCII,10);
    escribir_caracter_en_LCD('T',0);
    escribir_caracter_en_LCD('A',1);
    escribir_caracter_en_LCD(temperatura_ASCII[0],2);
    escribir_caracter_en_LCD(temperatura_ASCII[1],3);
    escribir_caracter_en_LCD('.',4);
    escribir_caracter_en_LCD('C',5);
    actualizar_LCD();
}

void Inicializar_PWM(void)
{
    DDRB=0b01000000; // Pin 7 del Puerto B como salida
    TCCR1A=0b00100001;
    TCCR1B=0b00000010;
}

```

## Anexo F Programa del controlador\_temperatura.c

```
#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/pgmspace.h>
#include <compat/deprecated.h>
#include <stdlib.h>
#define F_CPU 1000000L
#include <util/delay.h>
#include "LCD.h"
#include "joystick.h"
#include "ADC.h"
#include "PID.h"

extern volatile int enter;

int main(void)
{
    char temperatura_ASCII[]={0,'0','\0'};
    inicializar_LCD();
    sei();
    inicializar_joystick();
    itoa(TD,temperatura_ASCII,10);
    escribir_caracter_en_LCD('T',0);
    escribir_caracter_en_LCD('D',1);
    escribir_caracter_en_LCD(temperatura_ASCII[0],2);
    escribir_caracter_en_LCD(temperatura_ASCII[1],3);
    escribir_caracter_en_LCD('.',4);
}
```

```
escribir_caracter_en_LCD('C',5);
actualizar_LCD();
while(!enter)
    SMCR = 1;
    EIMSK = ((0<<PCIE1)|(0<<PCIE0));
    Inicializar_ADC();
    Inicializar_PWM();
while(1)
    PID();
return 1;
}
```